



UNIVERSIDADE DO ESTADO DO AMAZONAS
ESCOLA SUPERIOR DE TECNOLOGIA
ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Franklin Mendes Lima

**IDENTIFICAÇÃO DA ORIENTAÇÃO DE
OBJETOS RETÂNGULARES CONDUZIDOS
POR UMA ESTEIRA UTILIZANDO
SENSORIAMENTO ÓPTICO**

Manaus
2016

Franklin Mendes Lima

**IDENTIFICAÇÃO DA ORIENTAÇÃO DE
OBJETOS RETÂNGULARES CONDUZIDOS
POR UMA ESTEIRA UTILIZANDO
SENSORIAMENTO ÓPTICO**

Trabalho de Conclusão de Curso submetido à Coordenação do curso de Engenharia de Controle e Automação da Universidade do Estado do Amazonas como parte dos requisitos necessários para a obtenção do grau de Engenheiro.

Orientador Dr. Israel Mazaira Morales

Manaus
2016

Franklin Mendes Lima

**IDENTIFICAÇÃO DA ORIENTAÇÃO DE OBJETOS
RETANGULARES CONDUZIDOS POR UMA
ESTEIRA UTILIZANDO SENSORIAMENTO
ÓPTICO**

Trabalho de Conclusão de Curso submetido à Coordenação do curso de Engenharia de Controle e Automação da Universidade do Estado do Amazonas como parte dos requisitos necessários para a obtenção do grau de Engenheiro.

Aprovado em 17 de junho de 2016.

BANCA EXAMINADORA

Dr. Israel Mazaira Morales
Orientador

Msc. Charles Luiz da Silva Melo
Presidente da Banca

Dr. Daniel Guzman del Rio
Convidado 2

Dr. Luis Delfin Rojas Puron
Convidado 3

Manaus
2016

Agradecimentos

Primeiramente gostaria de agradecer a Deus pelas oportunidades concedidas durante a vida e pela determinação para concluir os estudos universitários.

Um agradecimento especial aos meus pais, Antonia e Francisco, pelo incentivo e apoio dado aos estudos e no dia a dia.

Agradeço à Universidade do Estado do Amazonas, à coordenação de Engenharia de Controle e Automação, ao meu orientador, Dr. Israel Mazaira Morales, e a todos os professores e demais funcionários que nos ajudaram a superar esta etapa da vida.

Um grande agradecimento pelo apoio e motivação dos meus amigos formados nesta jornada: Nilteomar Nascimento, Daniel Oliveira, Dilermando Ferreira, Eiji Maeda, Rodrigo Santa Rita, Kenny Caldas, Isabelle Stoco, Rafael Goulart, Felipe Máximo, Izoneide Leite, Antonio Medeiros, Wellington Di Tomasso, Lucas Bonetti, Letícia Silva, Érica Kido, Renan Baima, Álvaro Belota, Cassio Rodrigues, Anderson Hiago, Guilherme Linardi, Alexandre Gomes, Dianny Machado, Talita Timbó, Victor Amorim e Luanda Marinho.

Resumo

A colisão de uma ferramenta robótica acarreta em interrupções no processo produtivo gerando prejuízos e perda de competitividade. Braços robóticos são amplamente utilizados para a coleta de objetos na indústria, porém, o monitoramento de mudanças na orientação preestabelecida de objetos retangulares conduzidos em uma esteira objetivando a correção e adaptação de trajetória de coleta sem interrupções na linha de produção é pouco empregado. Este projeto visa a propor uma solução para esta problemática projetando um sistema de percepção robótico capaz de identificar tais mudanças utilizando sensores ópticos e um CLP que informa nos seus terminais de saída em formato Booleano a orientação do objeto relativo a um eixo preestabelecido e controla o acionamento e desligamento da esteira.

Palavras-chaves: Sistema de percepção robótico, identificar orientação, coletar objetos retangulares

Abstract

The collision of a robotic tool entails to interruptions in the production process causing damage and loss of competitiveness. Robotic arms are widely used for the collection of objects in the industry, however, the monitoring of changes in the predetermined orientation of rectangular objects carried on a treadmill aiming at the collecting trajectory correction and adaptation nonstopping the production line is rarely used. This project aims to propose a solution to this problem by projecting a robotic perception system capable of identifying such changes using optical sensors and a PLC that informs on its outputs terminals in Boolean format the object orientation relative to a predetermined axis and controls the treadmill activation and shutdown.

Key-words: Robotic perception system, identify orientation, collect rectangular objects

Lista de ilustrações

Figura 1 – Relações trigonométricas básicas	15
Figura 2 – Aplicação da trigonometria em um quadrado	16
Figura 3 – Aplicação da trigonometria em um retângulo	16
Figura 4 – Módulo de E/S discretas com 16 entradas e 16 saídas	19
Figura 5 – Ciclo de execução de um CLP típico	20
Figura 6 – Instruções básicas da Linguagem Ladder	20
Figura 7 – Exemplo de programa em Ladder	21
Figura 8 – Conversão de Numeral Decimal para Binário	22
Figura 9 – Componentes básicos de um robô	24
Figura 10 – Eixos de movimentação de um robô com seis juntas rotacionais	24
Figura 11 – Cinco principais geometrias robóticas	26
Figura 12 – Espaços de trabalho típicos	26
Figura 13 – Sistemas de referência robóticos	27
Figura 14 – Sensor ultrassônico testado no projeto	30
Figura 15 – Sensor óptico utilizado para detectar passagem de objetos conduzidos pela esteira	31
Figura 16 – CLP TSX 3722001	31
Figura 17 – Esquema de ligação do módulo TSX DMZ 28DR	32
Figura 18 – Fonte de energia secundária DC FA-3003 empregada	33
Figura 19 – Captura da tela das entradas programadas em Ladder	33
Figura 20 – Captura da tela das saídas programadas em Ladder	33
Figura 21 – Interface de programação do software PL7 PRO®	34
Figura 22 – Configuração idealizada dos sensores ópticos	35
Figura 23 – Definindo o período de interrupção do feixe de S_2 (ΔT_{I2})	36
Figura 24 – Determinando ΔT_{I2} por triangulação	36
Figura 25 – Estabelecendo ΔT_{I2} por triangulação quando $\theta \geq 0^\circ$	37
Figura 26 – Estabelecendo ΔT_{I2} por triangulação quando $\theta < 0^\circ$	38
Figura 27 – Variações de $ \theta $ para diferentes ΔT_{I2} quando $ \theta = \gamma$ e $V = 1$ [cm/s]	39
Figura 28 – Definindo o período de interrupção do feixe de S_1 (ΔT_{I1})	40
Figura 29 – Determinando ΔT_{I1} por triangulação	40
Figura 30 – Estabelecendo ΔT_{I1} por triangulação quando $\theta \geq 0^\circ$ e $ \theta \geq \alpha$	41
Figura 31 – Estabelecendo ΔT_{I1} por triangulação quando $\theta \geq 0^\circ$ e $ \theta < \alpha$	42
Figura 32 – Definindo ΔT_{I1} por triangulação quando $\theta < 0^\circ$ e $90^\circ - \theta \geq \alpha$	42
Figura 33 – Determinando ΔT_{I1} por triangulação quando $\theta < 0^\circ$ e $90^\circ - \theta < \alpha$	43
Figura 34 – Definindo ΔT_{I1} por triangulação quando $ \gamma = 45^\circ$ e $ \theta < 45^\circ$	44
Figura 35 – Fluxograma do projeto	45

Figura 36 – Cabine do controlador Yasnac MRC	46
Figura 37 – Braço Robótico Motoman SK6	46
Figura 38 – Acionamento da esteira	47
Figura 39 – Atribuição de C , L , V e α	47
Figura 40 – Calculando γ e γ'	48
Figura 41 – Evento ocorrido na borda de subida do sinal de S_1	48
Figura 42 – Contagem de ΔT_{I1}	49
Figura 43 – Borda de descida de S_1 e cálculo de ΔT_{I1}	49
Figura 44 – Acionamento do temporizador ΔT_{Off}	50
Figura 45 – Borda de descida de S_2 e cálculo de ΔT_{I2} e β	50
Figura 46 – Cálculo do módulo de θ	51
Figura 47 – Determinação do sinal de θ parte 1	51
Figura 48 – Determinação do sinal de θ parte 2	52
Figura 49 – Determinação do sinal de θ parte 3	52
Figura 50 – Determinação do sinal de θ parte 4	53
Figura 51 – Determinação do sinal de θ parte 5	53
Figura 52 – Conversão para booleano do sinal de teta	54
Figura 53 – Conversão do módulo de θ de Inteiro para Booleano equivalente	54
Figura 54 – Transmissão de θ nos terminais de saída do CLP parte 1	55
Figura 55 – Transmissão de θ nos terminais de saída do CLP parte 2	55
Figura 56 – Habilitando o robô e zerando as variáveis no fim do ciclo do programa	56
Figura 57 – Sistema de percepção robótico proposto	57

Lista de tabelas

Tabela 1 – Comparação da área de trabalho dos tipos de configuração de robôs . . . 27

Sumário

1	INTRODUÇÃO	11
1.1	Formulação do problema	11
1.2	Justificativa	11
1.3	Motivação	12
1.4	Objetivos	12
1.4.1	Objetivos Específicos	12
1.5	Metodologia	12
1.6	Organização do trabalho	13
2	REFERENCIAL TEÓRICO	14
2.1	Sensores	14
2.1.1	Sensores Ópticos	14
2.2	Trigonometria	15
2.3	Controlador Lógico Programável	17
2.3.1	Arquitetura do CLP	17
2.3.1.1	Fonte de Alimentação	17
2.3.1.2	Unidade Central de Processamento (UCP)	18
2.3.1.3	Memória	18
2.3.1.4	Módulos de Entrada e Saída ou Módulos de (E/S)	18
2.3.1.5	Terminal de Programação	19
2.3.2	Funcionamento de um CLP Típico	19
2.3.3	Linguagem de Programação Ladder	20
2.4	Conversões entre Numerais Decimal e Binário e Lógica Booleana	21
2.5	Robótica	22
2.5.1	Robô e Manipulador Robótico	22
2.5.2	Classificação de Robôs	24
2.5.3	Sistemas de Referência de um Robô	27
2.5.4	Percepção Robótica	28
2.5.5	Cinemática Direta e Inversa	28
3	TRABALHOS RELACIONADOS	29
3.1	Agarre Preciso de Malha Aberta com Mãos Subatuadas Inspi- rado pela Estratégia de Manipulação Humana	29
4	MATERIAIS E MÉTODOS	30
4.1	Identificação do Sistema de Percepção Robótico	30

4.1.1	Sensor Ultrassônico Arduino HC-SR04	30
4.1.2	Sensor Óptico Retro Reflexivo BEN5M-MFR	30
4.1.3	CLP Telemecanique® Modicon TSX 3722	31
4.1.3.1	Módulo de Entradas e Saídas Digitais TSX DMZ 28DR	32
4.1.4	Software PL7 PRO®	34
4.2	Determinando a Orientação de Coleta do Objeto Retangular	34
4.2.1	Cálculo do Módulo de θ	35
4.2.2	Estabelecendo o Sinal de θ	39
4.3	Controlando Componentes do Sistema	44
4.4	Programação do Cálculo da Orientação de Coleta de Objetos Retangulares	46
4.4.1	Acionamento da Esteira	47
4.4.2	Leitura dos Sensores Ópticos e Definição da Orientação de Coleta	48
4.4.3	Conversão dos Resultados Obtidos de Numeral Inteiro para Lógica Bo- oleana	53
4.4.4	Habilitar Robô e Zerar Variáveis	56
5	RESULTADOS E DISCUSSÕES	57
5.1	Determinação do Sistema de Percepção Robótico	57
5.2	Modelagem e Cálculo da Orientação de Agarre a partir de Variáveis do Sistema	57
5.3	Programação em Linguagem Ladder do Cálculo da Orientação de Coleta de Objetos Retangulares Aleatoriamente Orientados	59
6	CONCLUSÕES	61
6.1	Trabalhos futuros	62
	REFERÊNCIAS	63

1 INTRODUÇÃO

Os equipamentos robóticos clássicos apresentam uma programação fixa e não dispõem de sensoriamento a fim de ajustar suas trajetórias e de cumprir sua tarefa sob condições não previstas durante a etapa de sintonização. Indústrias e empresas comercializam e utilizam amplamente essas máquinas robóticas clássicas em operações de coleta e identificação de objetos, fresagem, usinagem e pintura manipulando-as por meio de ferramentas de controle clássico ou até mesmo somente se baseando nas configurações de fábrica desconsiderando variações de desempenho causadas por desgaste de peças, distúrbios e ruídos dos equipamentos localizados no processo produtivo inserido.

A colisão de uma ferramenta robótica gera desperdícios de matéria prima, danifica equipamentos ou até mesmo leva à sua própria inutilização acarretando em interrupções no processo produtivo gerando prejuízos e perda de competitividade no mercado. Visando à solução destes gargalos, há uma busca no desenvolvimento e elaboração de novas técnicas de controle e automação com a finalidade de fornecerem um desempenho funcional ótimo a equipamentos sob condições adversas e incertezas encontradas no ambiente real de trabalho por meio da elaboração de novos softwares e dispositivos, bem como, o desenvolvimento de aplicações utilizando recursos disponíveis.

O presente projeto visa a desenvolver um procedimento automatizado capaz de identificar e informar a orientação de objetos retangulares em relação a um eixo predeterminado enquanto são transportados por uma esteira.

1.1 Formulação do problema

Robôs pré-programados que não dispõem de um sistema de percepção robótico geram custos e atrasos desnecessários de produção quando não operam em condições ideais determinadas durante as fases de programação e sintonização.

1.2 Justificativa

Manipuladores robóticos são amplamente utilizados para a coleta de objetos na indústria, porém, a aplicação de técnicas de percepção robótica no monitoramento de mudanças na orientação de objetos transportados por uma esteira visando à correção e adaptação da trajetória de trabalho sem interrupções na linha de produção é pouco empregada. No desenvolvimento deste projeto foram utilizados conhecimentos nos ramos da Instrumentação Industrial, Robótica, Modelagem e Simulação de Sistemas Discretos, Redes Industriais de Comunicação, Controle e Automação.

1.3 Motivação

Existem fatores que comprometem a aplicação da visão robótica como: Ausência de iluminação adequada, intolerância à presença de corpos entre o elemento de aquisição de imagens e o objeto alvo, requisição de elevada capacidade de processamento de dados e necessidade de parar a esteira e, em seguida, coletar imagens para, somente depois, começar a processar as informações desejadas. Tendo em vista estas limitações, tem-se como motivação propor uma solução alternativa à visão robótica objetivando a solução da problemática de coleta de objetos aleatoriamente orientados.

1.4 Objetivos

O presente projeto tem como objetivo projetar um sistema de percepção robótico capaz de identificar e informar nos seus terminais de saída em formato booleano a orientação de objetos retangulares relativo a um eixo preestabelecido enquanto são conduzidos por uma esteira utilizando sensores ópticos e um Controlador Lógico Programável (CLP), que por sua vez, em um projeto futuro, indicará ao controlador do braço robótico a orientação de coleta do objeto a fim de adaptar a sua trajetória de trabalho, caso necessário.

1.4.1 Objetivos Específicos

- a) Identificar um conjunto de sensores apropriados para trabalhar em conjunto com um CLP na percepção de mudanças na orientação de objetos.
- b) Determinar a modelagem do cálculo da orientação do objeto a partir da identificação de variáveis relevantes do sistema.
- c) Planejar a interação do CLP com a esteira e os sensores.
- d) Elaborar, durante o desenvolvimento prático do projeto, um programa em Linguagem Ladder estabelecendo o controle do sistema por meio do CLP.
- e) Realizar testes entre a comunicação do CLP com a esteira e os sensores.
- f) Executar testes com os componentes integrados.
- g) Estabelecer limites operacionais deste projeto.

1.5 Metodologia

Foram realizadas pesquisas bibliográficas a respeito de sensores disponíveis no mercado e trabalhos relacionados ao desenvolvimento de sistemas de percepção robóticos.

Formulamos um referencial teórico explicando temas importantes à compreensão deste projeto: Sensores ópticos, trigonometria, CLP, Linguagem Ladder, conversões entre numerais decimal e binário e lógica Booleana, manipulador robótico, controlador do robô e percepção robótica.

Definimos o conjunto de sensoriamento o qual é composto por dois sensores ópticos retro reflexivos que informam ao CLP a duração da interrupção dos seus respectivos sinais causada pela transição do objeto conduzido pela esteira. Identificamos a modelagem da orientação de agarre do objeto em função de variáveis do sistema.

Criamos um programa em Linguagem Ladder capaz de receber a duração da interrupção de cada um dos sensores e informar em booleano a orientação do objeto nos seus terminais de saída.

Estabelecemos quando o CLP aciona e desliga a esteira.

Convertemos a informação em decimal da orientação do objeto para binário e, em seguida, armazenamos em bits de memória o seu equivalente booleano a fim de, em projetos futuros, informar ao controlador Yasnac MRC do braço robótico Motoman SK6 disponível no laboratório de Robótica da Escola Superior de Tecnologia da Universidade do Estado do Amazonas (EST-UEA) objetivando adaptar a trajetória de coleta de objetos retangulares aleatoriamente orientados.

Realizamos testes manuais após programar o CLP e ao integrá-lo aos sensores objetivando averiguar o correto funcionamento e interação entre eles.

1.6 Organização do trabalho

Este trabalho foi dividido em seis capítulos para melhor compreensão do leitor.

O capítulo 1 menciona uma breve introdução sobre a problemática da programação clássica de robôs a qual não abrange conceitos de percepção robótica e contextualiza o projeto por meio de justificativa e motivação. Também estabelece objetivos a serem cumpridos e a metodologia aplicada visando a alcançá-los.

Elaboramos no capítulo 2 um referencial teórico necessário ao entendimento e desenvolvimento do projeto explicando temas importantes tendo como fundamento o conhecimento de autores de obras consultadas e de trabalhos científicos relacionados às teorias e tecnologias aplicadas durante este trabalho.

No capítulo 3 realizamos resumo de trabalho relacionado a esta monografia descrevendo as principais ideias dos autores e citando semelhanças e diferenças relevantes.

Já no capítulo 4, há uma sistematização dos materiais e métodos aplicados para a determinação do sistema de percepção robótico empregado, bem como, a sua interligação e inteiração com os demais componentes utilizados.

Em seguida, no capítulo 5 relatamos e discutimos os resultados obtidos.

Por fim, no capítulo 6 citamos as conclusões deduzidas no decorrer da elaboração deste projeto e propomos trabalhos futuros relacionados a este tema.

Disponibilizamos a programação realizada em Linguagem Ladder em formato digital.

2 REFERENCIAL TEÓRICO

Para o desenvolvimento do presente trabalho, realizamos uma pesquisa bibliográfica teórica relacionada à: trigonometria, sensores industriais, cinemática direta e inversa de robôs, programação em Linguagem Ladder, conversão de números do formato inteiro em binário e programação de robôs.

Buscamos informações em manuais sobre o princípio de funcionamento de dispositivos como: sensores ópticos, controlador lógico programável, manipulador robótico e controlador do robô utilizado, bem como, as formas de interação entre si.

2.1 Sensores

Sensores são “dispositivos sensíveis a alguma forma de energia do ambiente que pode ser luminosa, térmica, cinética, relacionando informações sobre uma grandeza que precisa ser medida como: pressão, velocidade, [...] posição” (THOMAZINI; ALBUQUERQUE, 2007). Sensor “é um dispositivo que detecta a variação de uma grandeza física” (AGUIRRE, 2007b).

Os sensores podem ser divididos em analógicos ou digitais. O sensor analógico se caracteriza por poder indicar qualquer valor no seu sinal de saída no decorrer do tempo quando utilizado dentro de suas capacidades operacionais. O sensor digital, conhecido também como sensor discreto, “pode assumir um número finito de valores em determinada escala” (ROSÁRIO, 2005). O sensor binário é uma classe especial de sensor digital, pois pode transmitir somente dois valores no seu sinal de saída: 0 ou 1, “ligado” ou “desligado”.

Os sensores discretos podem ser divididos em dois grupos:

a) De contato mecânico: é necessário um contato físico entre o objeto e sensor a fim de detectar a presença do objeto.

b) Sem contato mecânico ou de proximidade: o objeto é detectado quando atinge certa proximidade com o sensor.

2.1.1 Sensores Ópticos

É utilizado na detecção de objetos em longas distâncias e há diversos modelos no mercado tornando possível detectar quaisquer tipos de materiais. Os sensores ópticos “emitem um feixe de luz e detectam as alterações da intensidade de luz recebida em consequência do movimento de objetos opacos. Possuem um emissor de impulsos rápidos de luz infravermelha e um receptor” (MORAES; CASTRUCCI, 2010).

“O emissor de luz óptico pode ser um diodo emissor de luz (LED) ou uma lâmpada. O receptor é um componente fotossensível, isto é, sensível à luz, como o fototransistor, os

fotodiodos ou resistores variáveis pela luz” (ROSÁRIO, 2005).

O emissor gera o feixe de luz de forma pulsada a uma certa frequência de modo que o receptor o distinga da luz do ambiente. O sinal recebido pelo receptor é enviado para um filtro passa faixa, onde somente uma faixa de frequência específica é convertida em tensão contínua que passa por um comparador o qual verifica se a tensão gerada corresponde à desejada. Caso seja, a saída do sensor é acionada.

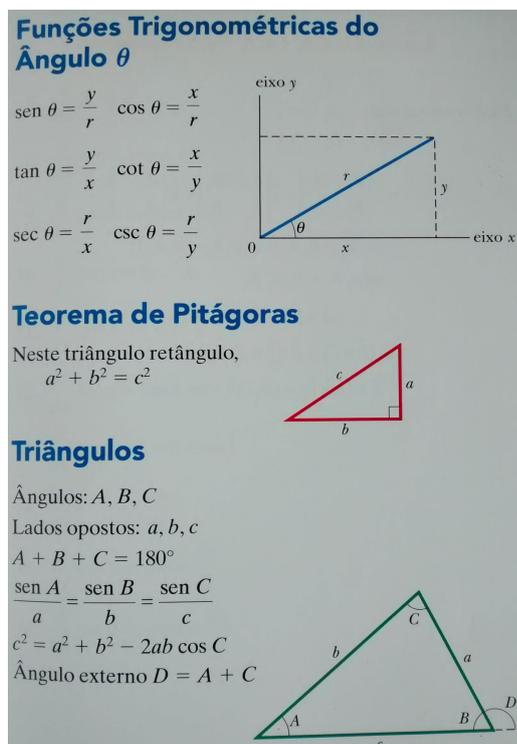
As vantagens do uso de sensores ópticos se dão pela capacidade de detectar pequenos objetos a longas distâncias, funcionam em ambientes sujos, demandam pouca manutenção e já existem alguns exemplares capazes de detectar objetos transparentes, porém, o alinhamento entre emissor e receptor é crucial ao correto funcionamento.

2.2 Trigonometria

A trigonometria é um ramo da matemática dedicada a estudar a razão de comprimento entre dois lados de um triângulo retângulo, o qual possui um dos seus três ângulos, A, medindo 90 graus [°]. Um dos ângulos agudos, B ou C, pode assumir valores acima de 0° a abaixo de 90° devendo satisfazer a equação 2.1 conforme pode ser visto na figura 1:

$$A + B + C = 180^\circ \tag{2.1}$$

Figura 1 – Relações trigonométricas básicas



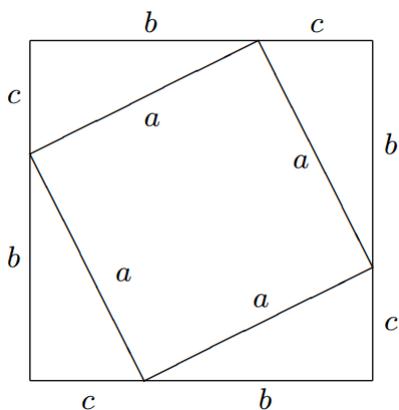
Fonte: (HALLIDAY; RESNICK; WALKER, 2010)

Podemos observar da figura 1 a possibilidade de obter o componente x de uma reta aplicando a função cosseno e a y, pela fórmula seno.

É importante destacar a impossibilidade de um ângulo dentro de um triângulo possuir um valor negativo. Contudo, quando medimos a inclinação relativa entre duas retas, podemos escolher uma delas como referência e, desta forma, a inclinação da segunda relativa à primeira pode assumir valores positivos ou negativos.

A trigonometria pode ser empregada em problemáticas envolvendo quadrados ou retângulos conforme ilustrado na figura 2:

Figura 2 – Aplicação da trigonometria em um quadrado

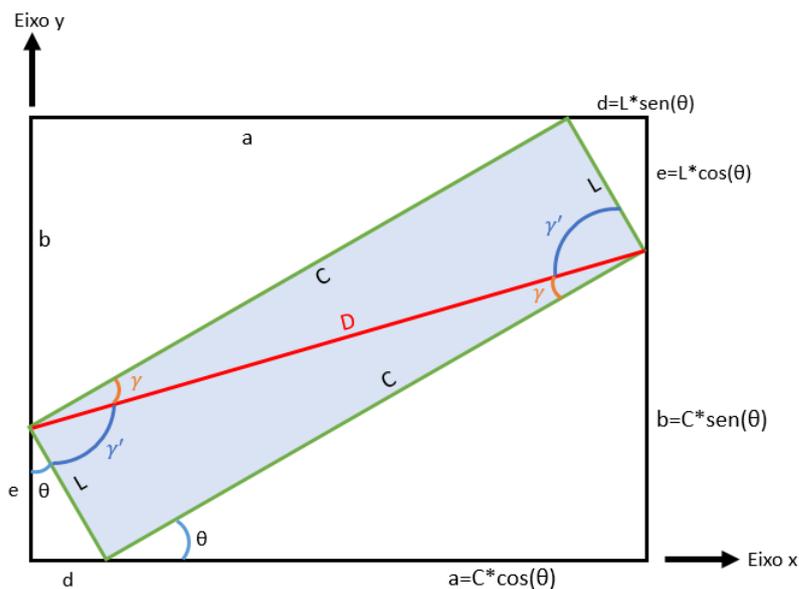


Fonte:

http://www.matematica.pucminas.br/profs/web_fabiano/calculo1/trigonometria.pdf

O retângulo, figura 3, é uma generalização do quadrado, o qual apresenta quatro lados iguais, a ou A, possuindo um comprimento, C, diferente de sua largura, L.

Figura 3 – Aplicação da trigonometria em um retângulo



Fonte: Autor

Em ambos os casos, podemos calcular a projeção desse objeto sobre eixos arbitrários por meio das fórmulas 2.2 e 2.3 e das mostradas na figura 1:

$$C * \operatorname{sen} (|\theta|) \pm L * \operatorname{cos} (|\theta|) = D * \operatorname{sen} (|\theta| \pm \gamma') [mm] \quad (2.2)$$

$$C * \operatorname{sen} (|\theta|) \pm L * \operatorname{cos} (|\theta|) = D * \operatorname{cos} (|\theta| \mp \gamma) [mm] \quad (2.3)$$

Onde:

$$D = \sqrt{C^2 + L^2} [mm] \quad (2.4)$$

$$\gamma = \operatorname{tg}^{-1} \left(\frac{L}{C} \right) = 90^\circ - \gamma' [^\circ] \quad (2.5)$$

$$\gamma' = \operatorname{tg}^{-1} \left(\frac{C}{L} \right) = 90^\circ - \gamma [^\circ] \quad (2.6)$$

2.3 Controlador Lógico Programável

O Controlador Lógico Programável (CLP) “é um dispositivo físico, eletrônico, que possui uma memória interna programável capaz de armazenar sequências de instruções lógicas binárias, além de outros comandos” (ROSÁRIO, 2005). O CLP “pode executar funções equivalentes às de um painel de relés ou de um sistema de controle lógico, também realizando operações lógicas e aritméticas, manipulação de dados e comunicação em rede, sendo utilizado no controle de sistemas automatizados” (MORAES; CASTRUCCI, 2001).

Há “várias décadas o [...] CLP é um dos equipamentos mais utilizados para controlar e automatizar equipamentos e processos, especialmente os presentes na indústria” (AGUIRRE, 2007a).

2.3.1 Arquitetura do CLP

Para MORAES e CASTRUCCI (2010), um CLP típico é constituído dos seguintes elementos:

2.3.1.1 Fonte de Alimentação

De acordo com MORAES e CASTRUCCI (2010), a fonte de alimentação realiza a conversão de corrente alternada em contínua para alimentar o controlador. Há dois tipos de fontes, a interna ao controlador, *source*, e a externa ao controlador, *sink*.

2.3.1.2 Unidade Central de Processamento (UCP)

Ainda segundo MORAES e CASTRUCCI (2010), a UCP é responsável pela execução dos programas de aplicação do usuário e pela atualização da memória de dados e da memória-imagem das entradas e saídas.

2.3.1.3 Memória

Memória é uma classificação de dispositivos capazes de armazenar dados e programas de forma temporária ou permanente. A unidade básica de memória é o dígito binário, ou bit, o qual pode conter duas informações, '0' ou '1'. Alguns modelos de memória permitem ao usuário somente a leitura de dados, *Read Only Memory* (ROM), e outros permitem também a escrita, *Random Access Memory* (RAM).

Conforme estabelecido por MORAES e CASTRUCCI (2010), as memórias mais utilizadas em CLPs são descritas abaixo:

a) memória EPROM: *Erasable Programmable Read Only Memory*, ou Memória Programável e Apagável Somente de Leitura, armazena dados e um programa responsável pela inicialização do controlador e gerencia sequências de operações;

b) memória do usuário: armazena o programa do usuário e possui dois modos de operação: em execução, ou RUN, responsável pela varredura cíclica do programa aplicativo criado pelo usuário fazendo o CLP operar em tempo real e o parado, PROG, utilizado quando realiza a transferência do programa aplicativo entre CLP e computador ou vice-versa;

c) memória de dados: mantém dados referentes ao processamento do programa aplicativo do usuário, ou seja, uma tabela de valores manipuláveis;

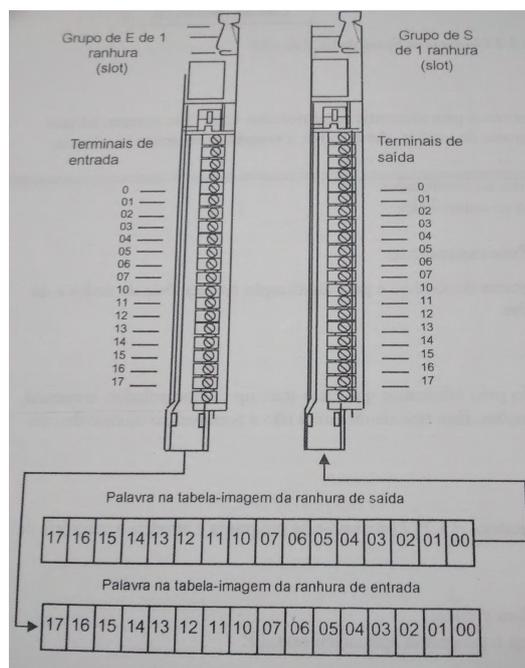
d) memória-imagem das entradas e saídas: reproduz digitalmente o estado dos equipamentos periféricos de entrada e saída.

2.3.1.4 Módulos de Entrada e Saída ou Módulos de (E/S)

“Correspondem a um conjunto de módulos que agrupam determinada categoria de sinais de E/S. Podem ser encontrados módulos para os mais diversos tipos de sensores, atuadores e dispositivos tipicamente utilizados em processos industriais, envolvendo o condicionamento de sinais e proteção” (AGUIRRE, 2007a).

Os módulos de entrada recebem dados analógicos ou digitais de eventos externos por meio de sensores, botões, geradores de pulsos, entre outros e os módulos de saída podem gerar sinais contínuos e discretos os quais podem ser direcionados a atuadores, indicadores, motores, válvulas, etc. “Basicamente, os módulos de saída dos controladores são acionados por três métodos: Saída a Relé, a Triac e a Transistor” (MORAES; CASTRUCCI, 2010). A figura 4 ilustra um módulo de E/S:

Figura 4 – Módulo de E/S diretas com 16 entradas e 16 saídas



Fonte: (AGUIRRE, 2007a)

2.3.1.5 Terminal de Programação

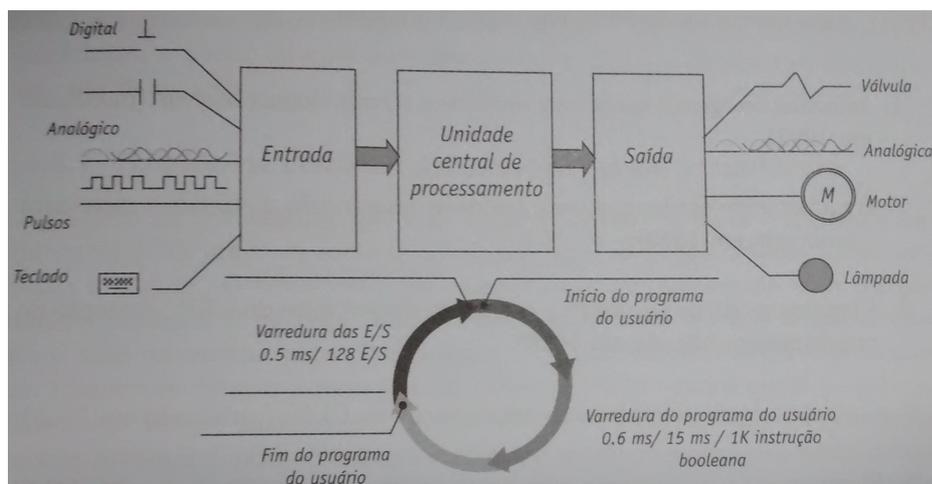
“É um dispositivo direcionado à comunicação entre usuário e controlador permitindo autodiagnóstico, alterações online, programação de instruções, monitoração, gravamento e apagamento de memória" (MORAES; CASTRUCCI, 2010).

2.3.2 Funcionamento de um CLP Típico

Para AGUIRRE (2007a), os CLPs típicos permitem a sua operação em dois modos operacionais básicos: o Modo de Programação e o Modo de Execução e o momento dado ao processo de executar o programa armazenado no CLP é Ciclo de Execução o qual é realizado continuamente e abrange as seguintes etapas sistematizadas na figura 5:

- leitura das informações das interfaces de entrada e armazenamento na memória, imagem das entradas;
- processamento das instruções do programa aplicativo criado pelo programador a partir das imagens de entrada gerando na memória os valores a serem atribuídos às saídas, imagens das saídas;
- atualização das saídas mediante o envio da respectiva imagem para respectivos módulos de saída sendo, esses três procedimentos, repetidos indefinidamente caso não seja interrompido o modo de operação, modificado o programa ou interrompido o fornecimento de energia ao CLP.

Figura 5 – Ciclo de execução de um CLP típico



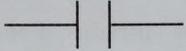
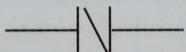
Fonte: (SOUZA, 2005)

2.3.3 Linguagem de Programação Ladder

A Linguagem Ladder é parecida com os diagramas de relés os quais eram utilizados pela indústria antes do advento do CLP. Essa semelhança visava à uma adaptação ágil dos profissionais em programar CLPs e à rápida aceitação de investidores empresários e industriários.

De acordo com AGUIRRE (2007a), esta é uma linguagem gráfica que utiliza símbolos de contato Normalmente Aberto (NA) e Normalmente Fechado (NF) que pode ser adequadamente combinados em série e paralelo para realizar a função lógica “E” e a função lógica “OU”. Quando desejado, o resultado da lógica é armazenado em uma instrução bobina a qual pode representar uma saída digital externa ou uma memória interna do CLP. As instruções de contato e bobina básicas podem ser vistas na figura 6:

Figura 6 – Instruções básicas da Linguagem Ladder

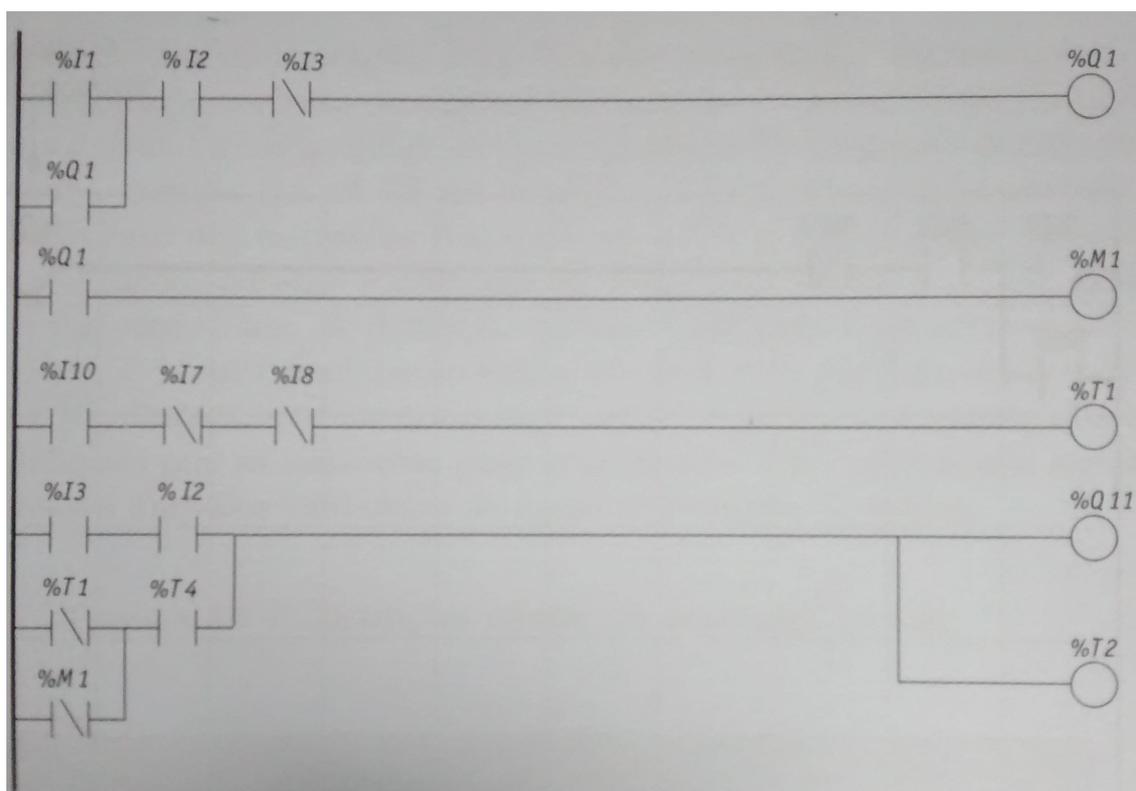
Instrução	Símbolo gráfico	Semântica
Instrução normalmente aberta	xx 	Verifica se o endereço xx é igual a 1 (continuidade lógica)
Instrução normalmente fechada	yy 	Verifica se o endereço yy é igual a 0 (continuidade lógica)
Instrução bobina	zz -()- ou 	Armazena o resultado da lógica de acordo com a continuidade lógica

Fonte: (AGUIRRE, 2007a)

Muitos CLPs possuem recursos mais avançados como memórias, armazenam informações, temporizadores, medem a duração do acionamento de suas entradas, contadores, verificam quantas vezes suas entradas foram acionadas, blocos de operações algébricas, manipulam informações de números inteiros e pontos flutuantes, etc.

A declaração dos principais elementos desta linguagem é geralmente identificada pelos seguintes comandos: contator (%I), bobina (%Q), memória (%M), temporizador (%Tm ou %T) e contador (%C). Um programa desta linguagem é mostrado na figura 7:

Figura 7 – Exemplo de programa em Ladder



Fonte: (AGUIRRE, 2007a)

Esta linguagem foi utilizada neste projeto para realizar o controle de componentes periféricos por meio de um CLP.

2.4 Conversões entre Numerais Decimal e Binário e Lógica Booleana

Os numerais de base decimal possuem dez possibilidades de valores (0, 1, 2, 3, 4, 5, 6, 7, 8, 9 ou 0) para cada casa numérica, enquanto os numerais de base binária só podem assumir os valores (0 ou 1) para cada casa numérica. Um dado Booleano unitário pode assumir um valor “verdadeiro” ou “falso”, equivalentes aos numerais binários ‘1’ ou ‘0’, respectivamente. Neste projeto, um relé de saída conduz energia quando há um bit de

memória associado em série possuindo um valor “verdadeiro”, ao passo que quando este bit assumir um valor “falso”, há uma interrupção da passagem de energia pelo relé do CLP.

Para converter um numeral decimal, como por exemplo “156”, em binário, ver figura 8, realizamos sucessivas divisões por dois e escrevemos ao lado o resto de cada divisão, os numerais em azul na figura 8. Repetimos este processo até o numeral se tornar ‘0’ ou ‘1’ e em seguida escreve-se também este numeral restante. Por último, escrevemos os numerais da fileira azul de baixo para cima, obtendo, desta forma, o equivalente binário de um decimal. Logo, “10011100” em binário equivale a “156” em decimal.

Figura 8 – Conversão de Numeral Decimal para Binário

$2 \overline{)156}$	<i>Remainder:</i>
$2 \overline{)78}$	0
$2 \overline{)39}$	0
$2 \overline{)19}$	1
$2 \overline{)9}$	1
$2 \overline{)4}$	1
$2 \overline{)2}$	0
$2 \overline{)1}$	0
	1
$156_{10} = 10011100_2$	

Fonte: <http://pt.wikihow.com/Converter-de-Decimal-para-Bin%C3%A1rio>

2.5 Robótica

“A robótica envolve o estudo da engenharia mecânica, da engenharia elétrica e da inteligência artificial, entre outras disciplinas” (ROSÁRIO, 2005).

Segundo Niku (2013), robótica é a arte, a base de conhecimento e *know-how* de concepção, aplicação e uso de robôs em atividades humanas. Sistemas robóticos consistem não apenas em robôs, mas também em outros dispositivos e sistemas utilizados em conjunto com os robôs.

“A robótica é um assunto interdisciplinar que se beneficia das engenharias mecânica, elétrica e eletrônica, da ciência da computação, das ciências cognitivas, da biologia e de muitas outras disciplinas” (NIKU, 2013).

2.5.1 Robô e Manipulador Robótico

Consoante ROSÁRIO (2005), com respeito à estrutura, um robô é um sistema mecânico de geometria variada, composto por corpos rígidos, articulados entre si, com a tarefa

de sustentar e posicionar/orientar a ferramenta terminal, garra mecânica ou ferramenta especializada, a qual fica em contato direto com o processo.

“Um dos sistemas robóticos mais encontrados na indústria, na academia e até na área espacial, são os robôs manipuladores industriais, ou chamados simplesmente de manipuladores” (AGUIRRE, 2007c).

Um manipulador industrial é “uma máquina [...] com vários graus de liberdade, controlada automaticamente, reprogramável, multifuncional, que pode ter base fixa ou móvel para a utilização em aplicações de automação industrial” (ROMANO, 2002).

De acordo com AGUIRRE (2007c), um manipulador está composto por estrutura física constituída por elos ligados por juntas, figura 9, possuindo: um braço atrelando mobilidade, um punho fornecendo destreza, um efetuator possibilitando a execução de tarefas específicas, atuadores gerando movimento por meio da atuação das juntas, motores tipicamente elétricos, hidráulicos e/ou pneumáticos, sensores mensurando o estado dos manipuladores e, em alguns modelos de robôs, adquirindo informações do ambiente externo e um sistema de controle encarregado de controlar e supervisionar os movimentos do robô. Consoante Niku (2013), um robô possui um processador, geralmente um computador, objetivando calcular os movimentos das articulações determinando quanto e quão rápido cada articulação deve se mover para atingir o local e as velocidades desejadas, supervisiona as ações coordenadas do controlador e dos sensores e requer um sistema operacional, programas, equipamentos periféricos, como um monitor e no robô há três grupos de programas de *software*:

- a) sistema operacional responsável por operar o processador;
- b) o responsável pelo cálculo dos movimentos necessários a serem atribuídos para cada junta com base nas equações da cinemática do robô e enviar tais informações ao controlador;
- c) um conjunto de rotinas orientadas a aplicações e programas desenvolvidos para o uso do robô ou de seus periféricos visando à realização de tarefas específicas.

“A mobilidade do manipulador é resultado de uma série de movimentos elementares, independentes entre si, denominados graus de liberdade do robô” (AMANDA, 1995). A quantidade de juntas de um robô é igual ao seu número de graus de liberdade.

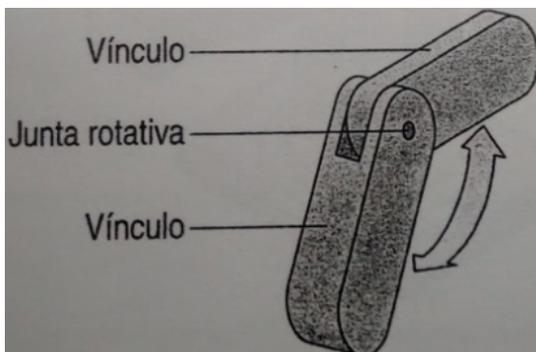
Para ROSÁRIO (2005), um robô industrial possui seis graus de liberdade, sendo três para determinar o posicionamento e outras três a fim de estabelecer a orientação da ferramenta terminal inserida em um ambiente tridimensional. Com menos de seis graus de liberdade não se alcança todos os pontos do local de trabalho. Um manipulador com mais de seis juntas é caracterizado como redundante, pois tem um ou mais graus de liberdade a mais do que o mínimo necessário para a execução da tarefa.

Um braço robótico geralmente é formado por sete segmentos de metal conhecidos como elos ou vínculos e unido por seis junções denominadas juntas, uniões, eixos ou articulações. Uma junta pode ser prismática, realiza um deslocamento linear de uma junta em relação

ao vínculo anterior, ou rotacional, gera um deslocamento angular entre duas junções sucessivas.

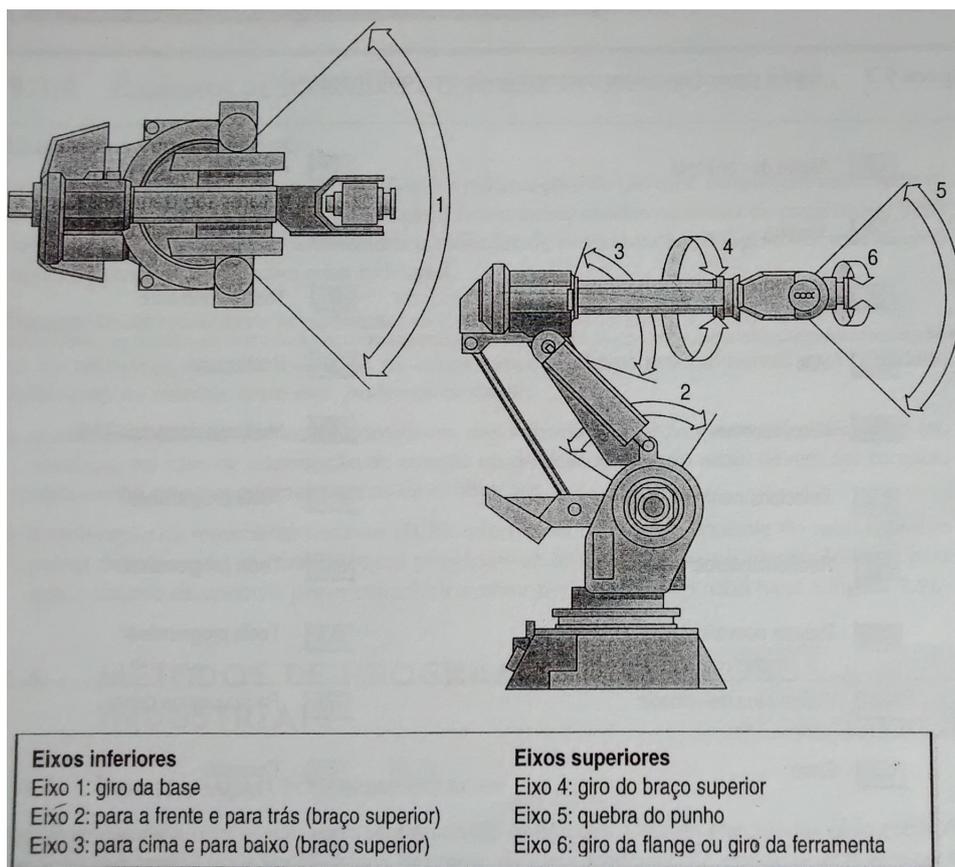
Um modelo de robô industrial é ilustrado na figura 10.

Figura 9 – Componentes básicos de um robô



Fonte: (ROSÁRIO, 2005)

Figura 10 – Eixos de movimentação de um robô com seis juntas rotacionais



Fonte: (ROSÁRIO, 2005)

2.5.2 Classificação de Robôs

A Associação Japonesa de Robôs Industriais, Japanese Industrial Robot Association (JIRA) estabeleceu a seguinte classificação de robôs:

a) Classe 1: dispositivo de movimentação manual: um dispositivo com vários graus de liberdade sendo acionado por um operador.

b) Classe 2: robô de sequência fixa: um mecanismo que executa as etapas sucessivas de uma tarefa de acordo com um determinado método o qual é de difícil modificação pelo usuário.

c) Classe 3: robô de sequência variável: Idêntico à classe 2, mas fácil de modificar pelo utilizador.

d) Classe 4: robô de reprodução: um operador humano realiza a tarefa manualmente, levando o robô, o qual registra os movimentos para reprodução posterior sendo capaz de reproduzir os mesmos movimentos de acordo com as informações gravadas.

e) Classe 5: robô de controle numérico: o operador fornece ao robô um programa de movimentos em vez de ensinar a tarefa manualmente.

f) Classe 6: robô inteligente: um autômato com os meios de compreender seu ambiente e a capacidade de concluir com êxito uma tarefa, apesar das mudanças nas condições do ambiente em que se deva ser concluída.

O Instituto Americano de Robótica, Robotics Institute of America (RIA), não considera as classes 1 e 2 citadas acima como robôs. A Associação Francesa de Robótica (ARF) determinou a seguinte classificação:

a) Tipo a: dispositivos de manipulação com controle manual para telerobótica.

b) Tipo b: mecanismos automáticos de manipulação com ciclos predeterminados.

c) Tipo c: programável, robôs servos controlados com trajetórias contínuas ou ponto a ponto.

d) tipo d: Semelhante a C, mas com capacidade de adquirir informações do seu ambiente.

As articulações robóticas podem ser de translação ou prismáticas (P), rotação ou de Revolução (R) e esféricas. Por exemplo, um robô com três juntas prismáticas e três rotacionais pode ser designado como (PPRRR) ou (3P3R).

Quanto ao tipo de elo, há cinco modelos de robôs bastante difundidos:

a) cartesiana, retangular ou pórtico (PPP);

b) cilíndrica (PRP);

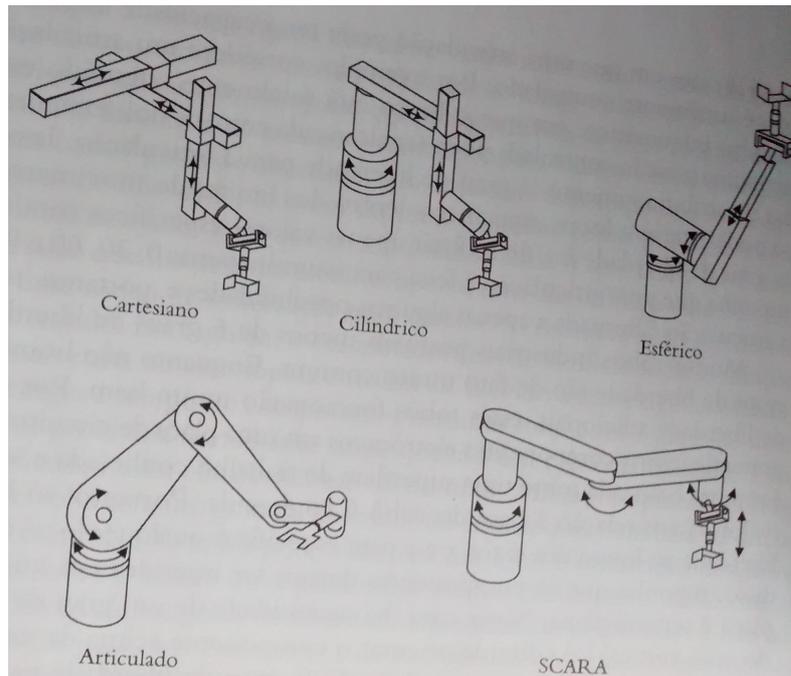
c) esférica (RRP);

d) articulada ou antropomórfica (PPP);

e) SCARA: possui duas ou três juntas rotacionais paralelas permitindo ao manipulador se mover em um plano horizontal, além articulações prismáticas adicionais.

O formato típico destes modelos podem ser conferidos na figura 11:

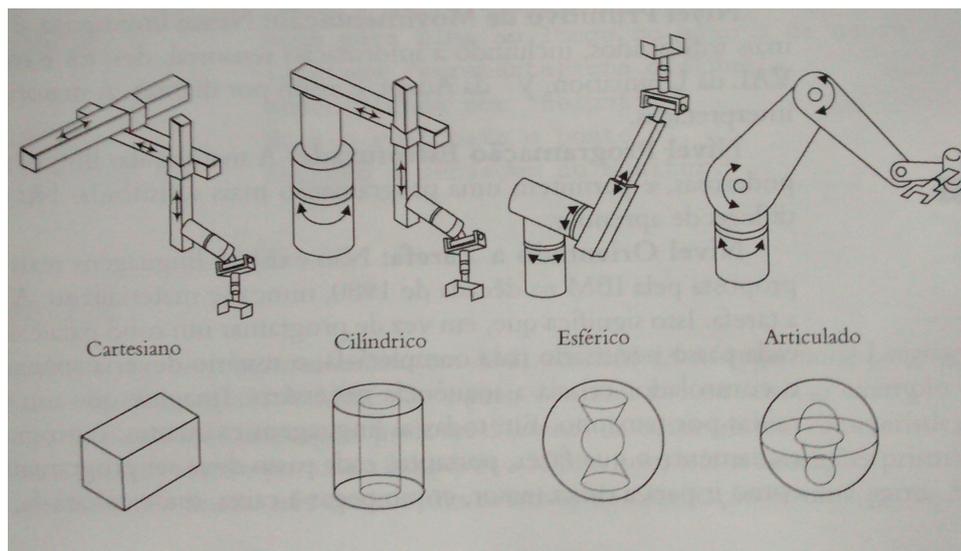
Figura 11 – Cinco principais geometrias robóticas



Fonte: (NIKU, 2013)

Os quatro primeiros modelos citados antes do SCARA normalmente possuem 3 juntas rotacionais adicionais a fim de orientar a garra ou mão do robô e estas combinações das três primeiras juntas resultam em áreas de trabalho distintas conforme exemplificado na figura 12 e na tabela 1:

Figura 12 – Espaços de trabalho típicos



Fonte: (NIKU, 2013)

Tabela 1 – Comparação da área de trabalho dos tipos de configuração de robôs

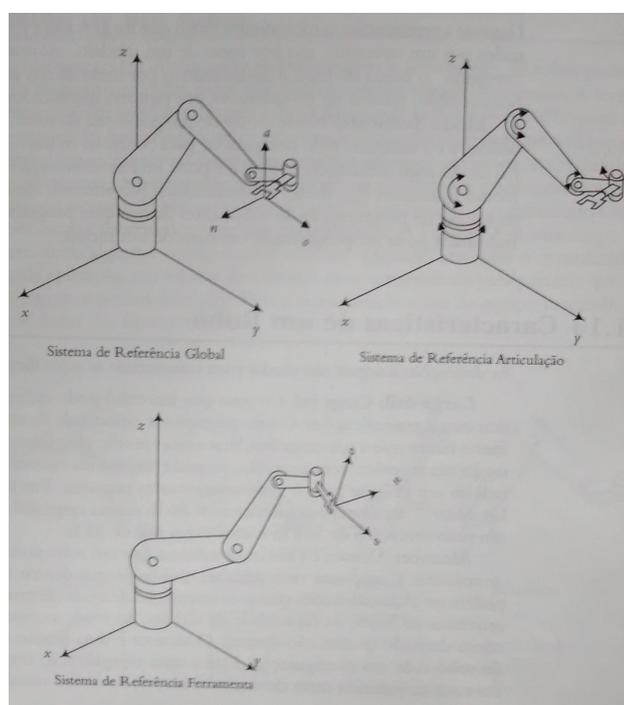
Tipo de configuração das juntas robóticas	Capacidade quanto à área de trabalho
Cartesiano	Alcançam qualquer ponto de um cubo de lado L.
Cilíndrico	Alcançam qualquer ponto em um cilindro de altura L e raio 2L, exceto os pontos do cilindro interno de raio L e altura L.
Esférico	Alcançam qualquer ponto de uma esfera de raio 2L, exceto a esfera interna de raio L.
Articulado	Alcançam qualquer ponto de um cilindro de raio 2L e altura L.
SCARA	Alcançam qualquer ponto de uma esteira de raio 2L.

Fonte: (ROSÁRIO, 2005)

2.5.3 Sistemas de Referência de um Robô

Segundo Niku (2013), os movimentos do robô geralmente são realizados nos sistemas coordenados exibidos na figura 13:

Figura 13 – Sistemas de referência robóticos



Fonte: (NIKU, 2013)

a) sistema de referência global: é um sistema universal de coordenadas definido pelos eixos x, y e z. As articulações do robô se movem simultaneamente de uma maneira coordenada para deslocar o efetuador ou a garra em relação aos três primeiros eixos

principais. É usado para mensurar movimentos realizados pelo manipulador relativos à sua base ou a outros componentes;

b) sistema de referência das articulações: cada elo é acessado e movido individualmente, portanto, procura-se replicar um deslocamento do efetuador em uma única junta;

c) sistema de referência ferramenta: especifica os movimentos da mão do robô em relação ao sistema de referência da própria mão.

2.5.4 Percepção Robótica

Para AGUIRRE (2007c), os sensores integrados ao robôs podem ser:

a) internos ou propioceptivos: fornecem informações sobre o estado do robô;

b) externos ou exteroceptivos: visam a obter informações de uma área ao redor do robô, para ser realizada como realimentação pelo seu sistema de controle.

Consoante AGUIRRE (2007c), sensores externos possibilitam a utilização de técnicas de controle de malha fechada objetivando conferir ao robô a capacidade de modificar a forma de realizar sua tarefa, de acordo com a percepção do ambiente em que se encontra, permitindo a realização de tarefas complexas, tais como agarrar objetos aleatoriamente posicionados e orientados, seguir objetos móveis em um ambiente tridimensional, realizar a montagem de dispositivos mecânicos, cumprir a inspeção e controle de qualidade de peças, efetuar a busca e identificação de objetos, etc.

2.5.5 Cinemática Direta e Inversa

A cinemática de um robô manipulador é o estudo da posição, orientação e velocidade angular e linear do seu efetuador e de seus ligamentos.

Segundo MAZAIRA (2008), o problema central da cinemática é como definir a posição do robô o qual pode ser dividido em duas partes, na cinemática direta, encontrar a posição e orientação da ferramenta no espaço cartesiano da base a partir das posições das articulações e na cinemática inversa, definir possíveis posições das juntas dada uma posição e orientação desejada para o efetuador, caso existirem.

3 TRABALHOS RELACIONADOS

Neste capítulo serão abordados alguns trabalhos científicos que, de alguma maneira, se assemelham com o tema desta monografia. O objetivo é encaixá-la no contexto científico, descrevendo as semelhanças e diferenças com os trabalhos já publicados.

3.1 Agarre Preciso de Malha Aberta com Mãos Subatuadas Inspirado pela Estratégia de Manipulação Humana

“Agarre Preciso de Malha Aberta com Mãos Subatuadas Inspirado pela Estratégia de Manipulação Humana”, em tradução livre para a Língua Portuguesa do título “*Open-Loop Precision Grasping With Underactuated Hands Inspired by a Human Manipulation Strategy*”, é um projeto científico realizado por (ODHNER; MA; DOLLAR, 2013).

Neste artigo, os autores demonstram um projeto de dedos subatuados e um método de agarre para a manipulação e coleta precisa de pequenos objetos baseando-se na estratégia da anatomia humana para pegar objetos a partir de uma superfície plana. Introduziram uma de tarefa virar e agarrar o objeto, sendo que um dedo se apoia na superfície plana, enquanto o outro dedo pega um objeto fino lançando-o em uma configuração estável de agarre entre dois dedos.

Há um foco em realizar a coleta de objetos desconhecidos e observar as estratégias tomadas por um manipulador robótico enquanto é determinado uma configuração do braço robótico como um todo apoiado sobre a superfície por um dedo ao passo que ambos os dois dedos tentam virar e agarrar o corpo.

Foi realizada uma modelagem matemática considerando as dimensões da mão robótica e da força aplicada sobre o corpo.

Analisaram a variação de desempenho em função das dimensões do objeto a ser coletado e da configuração da posição do manipulador.

Este robô precisa de contato físico para poder determinar a localização do objeto alvo e em relação a si mesmo.

Uma vantagem em projetar um sistema de percepção robótico com sensores a laser é a não necessidade de realizar contato diretamente com o corpo para poder determinar sua orientação.

4 MATERIAIS E MÉTODOS

Neste capítulo há uma sistematização dos materiais e métodos aplicados para a determinação do sistema de percepção robótico empregado, bem como, a sua interligação e comunicação estimada com uma esteira e um controlador de manipulador robótico.

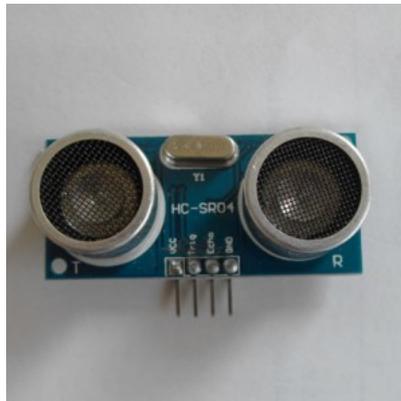
4.1 Identificação do Sistema de Percepção Robótico

Foi realizada uma pesquisa bibliográfica abrangente sobre sensores disponíveis no mercado. O sensor deve ser capaz, em conjunto com o CLP, de detectar a orientação de objetos, ou seja, os deslocamentos angulares, relativos a um eixo preestabelecido.

4.1.1 Sensor Ultrassônico Arduino HC-SR04

Efetuamos testes com um sensor ultrassônico Arduino HC-SR04, figura 14, capaz de calcular a distância de objetos situados entre 2 e 400 cm do sensor. Porém, quando o ângulo de inclinação do objeto em relação ao sensor era superior a 45° , as medidas obtidas foram bastante imprecisas sendo, portanto, desconsiderado do projeto.

Figura 14 – Sensor ultrassônico testado no projeto



Fonte: Autor

4.1.2 Sensor Óptico Retro Reflexivo BEN5M-MFR

Neste projeto foram utilizados dois sensores ópticos retro reflexivos BEN5M-MFR, figura 15, e seus respectivos espelhos refletor:

Figura 15 – Sensor óptico utilizado para detectar passagem de objetos conduzidos pela esteira



Fonte: Autor

O espelho refletor tem a finalidade de direcionar ao receptor o feixe de luz infravermelho gerado pelo emissor. O BEN5M-MFR pode ser alimentado tanto com fonte de corrente contínua quanto com alternada suportando diferença de tensões entre 24 e 240 Volts (V). É capaz de detectar objetos opacos a uma distância de 0,1 a 5 metros.

Há dois modos de operação selecionáveis, o Modo Claro (Light Mode) e o Modo Escuro (Dark Mode). No Modo Claro, o contato do relé do sensor óptico fica em estado normalmente aberto. Isso significa que a luz indicadora verde do sensor somente irá acender ao detectar o objeto. No Modo Escuro, o contato do relé do sensor fica em estado normalmente fechado, sendo que a luz indicadora verde irá apagar somente ao detectar o objeto. Uma luz vermelha indica se o sensor está ligado.

Este sensor tem cinco fios possuindo as seguintes funções:

- a) marrom: transmite energia da fonte ao sensor;
- b) azul: fio neutro do sensor;
- c) cinza: transmite o sinal de saída do contato em modo Normalmente Fechado (NF);
- d) preto: envia o sinal de saída do contato em modo Normalmente Aberto (NA);
- e) branco: direciona a energia da fonte ao relé do sensor.

4.1.3 CLP Telemecanique® Modicon TSX 3722

Entre os controladores lógicos programáveis disponíveis na universidade, foi escolhido o CLP Telemecanique® Modicon TSX 3722, figura 16:

Figura 16 – CLP TSX 3722001



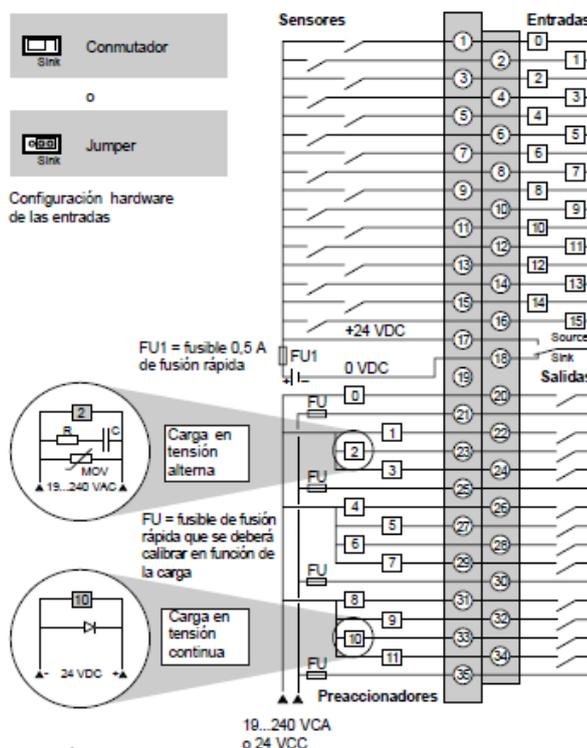
Fonte: Autor

Segundo o manual (ELETRIC, 2001), o TSX 3722 não possui módulos de entrada e saída embutidos, mas é adaptado para conectar tanto módulos analógicos quanto digitais, contém um visualizador do estado das entradas e saídas acoplados, necessita de uma fonte de corrente alternada primária com tensão entre 100 Volts (V) e 240 V ou uma alimentação de corrente contínua de 24 V com tensões limites entre 19 V e 30 V. Os módulos de E/S necessitam de uma fonte secundária para conduzir energia nos seus terminais.

4.1.3.1 Módulo de Entradas e Saídas Digitais TSX DMZ 28DR

O módulo de E/S TSX DMZ 28DR, figura 17, possibilita a comunicação do CLP TSX 3722 com outros dispositivos como sensores, atuadores, esteiras e outros controladores. Possui 28 ligações digitais as quais 16 são entradas e 12 são saídas a relé. As entradas podem ser usadas na configuração *Sink* (equivalente ao modo PNP) ou *Source* (o mesmo que modo NPN). Para que ocorra condução de energia, é necessária, além da fonte primária ligada ao CLP, uma fonte de corrente contínua secundária de 24 V, figura 18, conectada aos terminais do DMZ 28DR.

Figura 17 – Esquema de ligação do módulo TSX DMZ 28DR



Fonte: (ELETRIC, 2001)

Figura 18 – Fonte de energia secundária DC FA-3003 empregada



Fonte: Autor

Foram utilizados 3 dos 16 terminais de entrada, figura 19, e 9 dos 12 terminais de saída, figura 20, neste projeto.

Figura 19 – Captura da tela das entradas programadas em Ladder

Address	Type	Symbol	Comment
%CH1MOD	CH		
%I1MOD.ERR	EBOOL		
%Mv1MOD	WORD		
%Mv1MOD.1	WORD		
%Mv1MOD.2	WORD		
%Mv1MOD.3	WORD		
%Kv1MOD	WORD		
%Kv1MOD.1	WORD		
%Kv1MOD.2	WORD		
%Kv1MOD.3	WORD		
%I1.0	EBOOL		
%I1.0.ERR	BOOL		
%I1.1	EBOOL		
%I1.1.ERR	BOOL		
%I1.2	EBOOL		
%I1.2.ERR	BOOL		
%I1.3	EBOOL		
%I1.3.ERR	BOOL		
%I1.4	EBOOL	Ligar	CLP liga a esteira ao iniciar seu programa
%I1.4.ERR	BOOL		
%I1.5	EBOOL		
%I1.5.ERR	BOOL		
%I1.6	EBOOL	Sinal_sensor_s1	O sensor S1 conduz energia à entrada enquanto seu feixe de luz é interrompido
%I1.6.ERR	BOOL		
%I1.7	EBOOL	Sinal_sensor_s2	O sensor S2 conduz energia à entrada enquanto seu feixe de luz é interrompido

Fonte: Autor

Figura 20 – Captura da tela das saídas programadas em Ladder

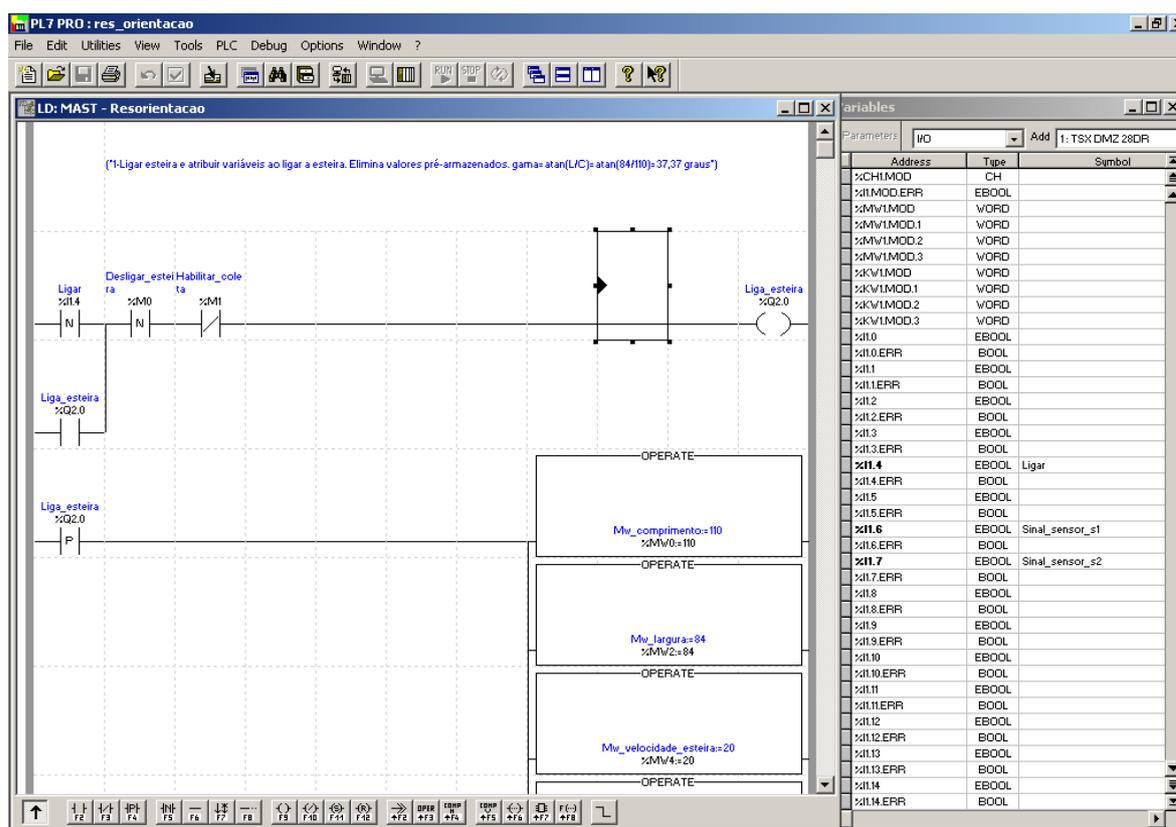
Address	Type	Symbol	Comment
%CH2MOD	CH		
%I2MOD.ERR	EBOOL		
%Mv2MOD	WORD		
%Mv2MOD.1	WORD		
%Mv2MOD.2	WORD		
%Mv2MOD.3	WORD		
%Kv2MOD	WORD		
%Q2.0	EBOOL	Liga_esteira	Controla o acionamento da esteira
%I2.0.ERR	BOOL		
%Q2.1	EBOOL	Brk_0	
%I2.1.ERR	BOOL		
%Q2.2	EBOOL	Brk_1	
%I2.2.ERR	BOOL		
%Q2.3	EBOOL	Brk_2	
%I2.3.ERR	BOOL		
%Q2.4	EBOOL	Brk_3	
%I2.4.ERR	BOOL		
%Q2.5	EBOOL	Brk_4	
%I2.5.ERR	BOOL		
%Q2.6	EBOOL	Brk_5	
%I2.6.ERR	BOOL		
%Q2.7	EBOOL	Brk_6	
%I2.7.ERR	BOOL		
%Q2.8	EBOOL	Brk_7	

Fonte: Autor

4.1.4 Software PL7 PRO®

O *software* PL7 PRO®, figura 21, é compatível com o CLP TSX 3722 sendo programável e executável no sistema operacional Windows XP. Este *software* é utilizado para programar controladores de acordo com o padrão IEC Standard 1131-3. O programa permite projetar aplicações nas Linguagens: Ladder, Grafset, Lista de Instruções ou Instruction List (SL) e Texto Estruturado ou Structured Text (ST).

Figura 21 – Interface de programação do software PL7 PRO®

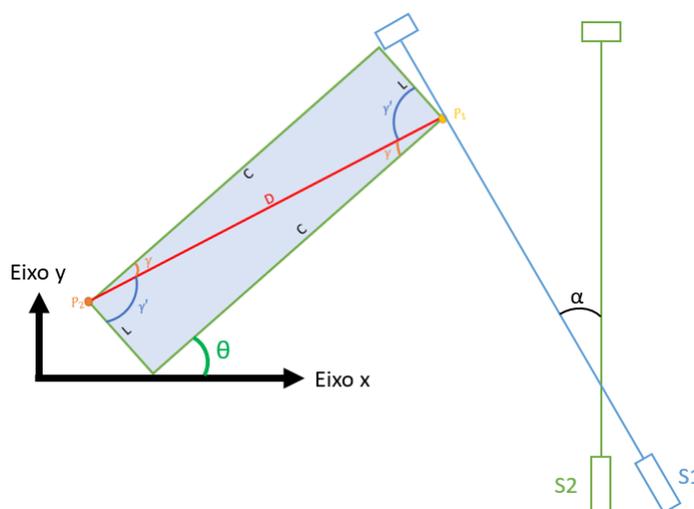


Fonte: Autor

4.2 Determinando a Orientação de Coleta do Objeto Retangular

O projeto de sensoriamento proposto visando à determinação da orientação de um objeto retangular conduzido por uma esteira a uma velocidade V , em [mm/s], é mostrado na figura 22:

Figura 22 – Configuração idealizada dos sensores ópticos



Fonte: Autor

Onde:

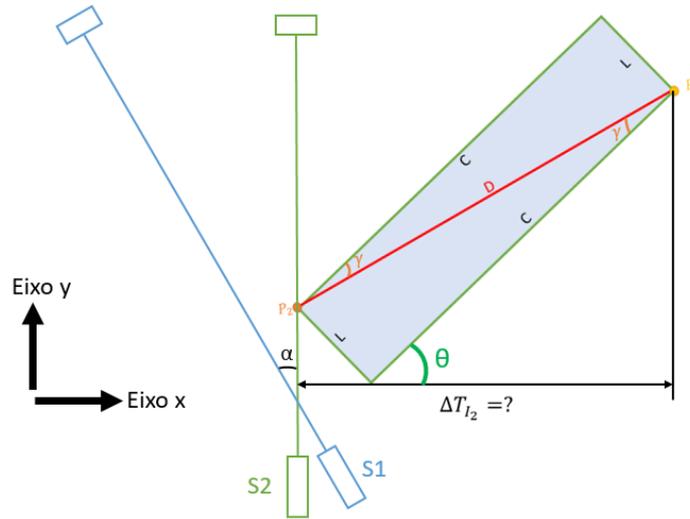
 S_1 e S_2 : sensores ópticos 1 e 2, respectivamente; P_1 : ponto do objeto o qual causa a interrupção da luz enviada pelo sensor S_1 nos cálculos de ΔT_{I1} ou pelo sensor S_2 nas fórmulas envolvendo ΔT_{I2} ; P_2 : último ponto do objeto a interromper a luz enviada pelo sensor S_1 nas equações com ΔT_{I1} ou pelo sensor S_2 nas operações com ΔT_{I2} ; C : comprimento do corpo retangular em [mm]; L : largura do corpo retangular em [mm]; D : diagonal do corpo retangular em [mm]; θ : ângulo formado pelo objeto em relação ao eixo x em [°]; α : deslocamento angular do sensor S_1 relativo ao sensor fixo S_2 em [°]; ΔT_{I1} : intervalo de nível alto da saída de S_1 ou período de interrupção do feixe de S_1 em [mm]; ΔT_{I2} : intervalo de nível alto da saída de S_2 ou período de interrupção do feixe de S_2 em [mm]; ΔT_{Off} : período entre a borda de subida do sinal do sensor S_2 e o desligamento da esteira em [s].

4.2.1 Cálculo do Módulo de θ

O objetivo central sistema de percepção robótico é obter o valor de θ a partir ΔT_{I1} e ΔT_{I2} fornecidos pelos sensores S_1 e S_2 , respectivamente, e dos dados informados durante a elaboração do programa em Linguagem Ladder: velocidade da esteira (V), comprimento do objeto (C), largura do objeto (L) e deslocamento angular do sensor S_1 relativo ao sensor fixo S_2 (α).

Primeiramente, calcula-se somente o módulo de θ ($|\theta|$) a partir de ΔT_{I2} , V , C e L . A formulação matemática relacionando $|\theta|$ com ΔT_{I2} , V , C e L é ilustrada em um exemplo de objeto com ângulo $\theta \geq 0^\circ$:

Figura 23 – Definindo o período de interrupção do feixe de S_2 (ΔT_{I2})



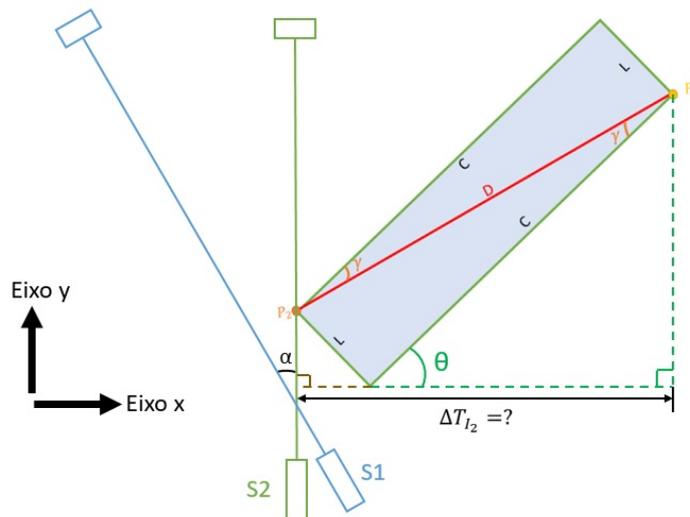
Fonte: Autor

Como mostrado na figura 23, ΔT_{I2} é o intervalo entre a passagem de P_2 pelo feixe de S_2 e o cruzamento de P_1 pelo feixe de S_2 . Como S_2 está alinhado com o eixo Y, ΔT_{I2} está relacionado à projeção do objeto sobre o eixo X (X_0) pela fórmula 4.1.

$$\Delta T_{I2} = X_0/V [s] \tag{4.1}$$

Podemos determinar ΔT_{I2} por meio de triangulações, figura 24:

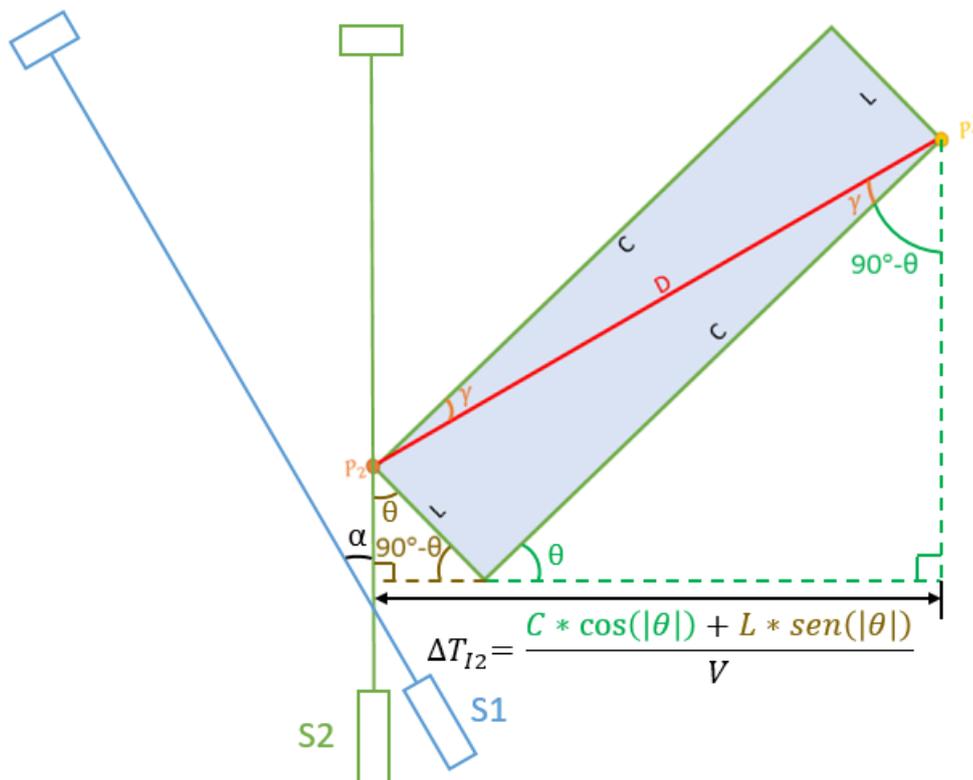
Figura 24 – Determinando ΔT_{I2} por triangulação



Fonte: Autor

Definindo os ângulos internos da figura 24, nota-se que a reta X_0 é composta pela projeção do comprimento e largura do objeto sobre o eixo X, figura 25 e equação 4.2:

Figura 25 – Estabelecendo ΔT_{I2} por triangulação quando $\theta \geq 0^\circ$



Fonte: Autor

Encontramos:

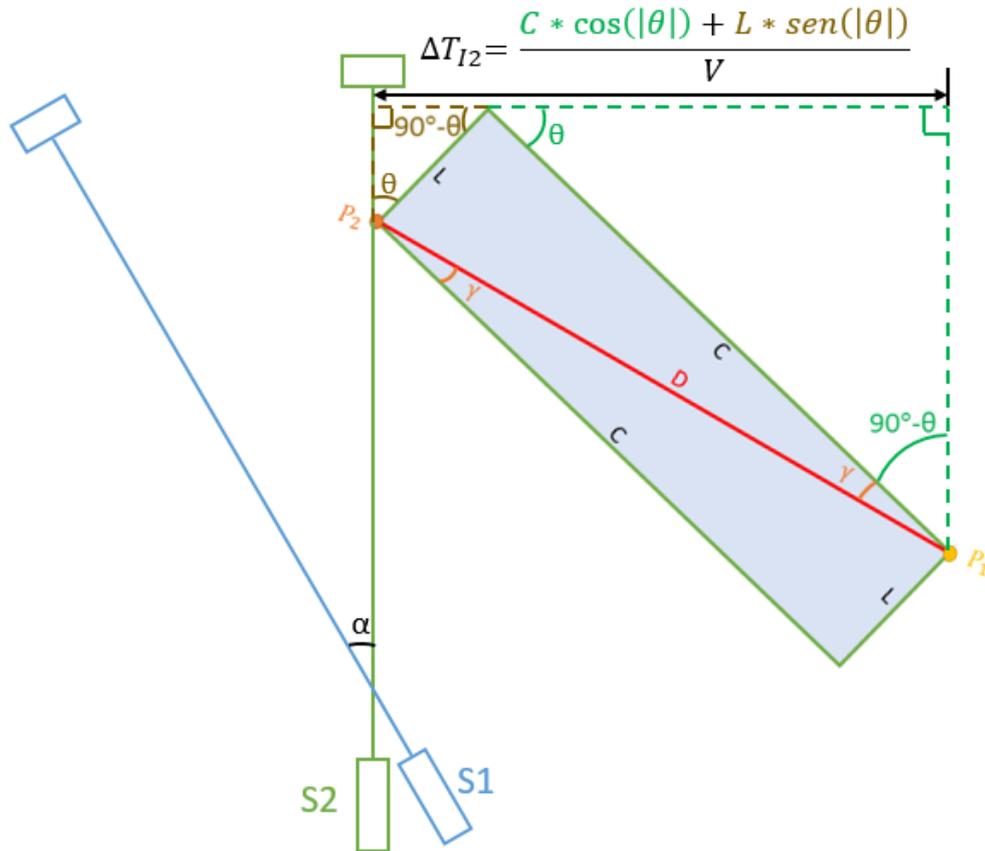
$$X_0 = C * \cos(|\theta|) + L * \text{sen}(|\theta|) \text{ [mm]} \quad (4.2)$$

$$\Delta T_{I2} = [C * \cos(|\theta|) + L * \text{sen}(|\theta|)] / V \text{ [s]} \quad (4.3)$$

É importante lembrar que em triângulos só existem ângulos positivos. Por isso, somente utilizamos o valor absoluto dos deslocamentos angulares entre as retas dos triângulos.

Ao aplicarmos novamente os passos citados anteriormente quando $\theta < 0^\circ$, vemos que as equações 4.2 e 4.3 são aplicáveis também:

Figura 26 – Estabelecendo ΔT_{I2} por triangulação quando $\theta < 0^\circ$



Fonte: Autor

Podemos isolar $|\theta|$ da equação 4.3, a partir das fórmulas 2.2 ou 2.3, obtendo as equações 4.4 ou 4.5, respectivamente:

$$|\theta| = \text{sen}^{-1}(V * \Delta T_{I2}/D) - \gamma' [^\circ] \quad (4.4)$$

$$|\theta| = \text{cos}^{-1}(V * \Delta T_{I2}/D) + \gamma [^\circ] \quad (4.5)$$

Contudo, ao aplicar as equações 4.4 e 4.5 de forma arbitrária, podemos encontrar resultados negativos para $|\theta|$. Mas ao verificar a seguinte condição:

$$\beta = \text{sen}^{-1}(V * \Delta T_{I2}/D) \geq \gamma' [^\circ] \quad (4.6)$$

Se a equação 4.6 for verdadeira, para calcular $|\theta|$, deve-se utilizar a equação 4.4. Se não for, a equação 4.5 torna-se aplicável.

Um caso crítico na determinação de $|\theta|$ foi encontrado. Quando $|\theta| = \gamma$, pequenas mudanças de ΔT_{I2} , advindas de oscilações da esteira ou de pouca precisão dos sensores, podem causar grandes variações no módulo de θ , figura 27:

Figura 27 – Variações de $|\theta|$ para diferentes ΔT_{I2} quando $|\theta| = \gamma$ e $V = 1$ [cm/s]

C= 3 [cm], L= 1 [cm]				C= 3 [cm], L= 2 [cm]			
$ \theta =\gamma=18,435^\circ$				$ \theta =\gamma=33,69^\circ$			
ΔT_{I2}	$ \theta $	Varição de ΔT_{I2}	Varição de $ \theta $	ΔT_{I2}	$ \theta $	Varição de ΔT_{I2}	Varição de $ \theta $
$\sqrt{10}$ [s]	18,435°	-	-	$\sqrt{13}$ [s]	33,69°	-	-
3,16 [s]	16,260°	0,072%	2,175°	3,603 [s]	31,515°	0,072%	2,175°
3,15 [s]	13,385°	0,3883%	5,050°	3,592 [s]	28,640°	0,3883%	5,050°
3,13 [s]	10,242°	1,021%	8,193°	3,569 [s]	25,497°	1,021%	8,193°
3,10 [s]	7,045°	1,969%	11,39°	3,535 [s]	22,300°	1,969%	11,39°
3,00 [s]	0°	5,132%	18,435°	3,421 [s]	15,255°	5,132%	18,435°
3,00 [s]	0°	5,132%	18,435°	3,00 [s]	0°	16,795%	33,69°
1,00 [s]	90°	68,377%	71,565°	2,00 [s]	90°	44,530%	56,31°

Fonte: Autor

Sendo que:

$$L/V \leq \Delta T_{I2} \leq D/V [s] \quad (4.7)$$

Na condição $|\theta| = \gamma$, um erro de 0,1% na definição de ΔT_{I2} acarreta um desvio superior a 2° em $|\theta|$ para qualquer $0^\circ < \gamma \leq 45^\circ$ sendo aconselhável, portanto, um sensor capaz de mensurar ΔT_{I2} com elevada precisão.

Caso deseje-se modificar o valor das variáveis V , C , L e/ou α , não é necessário parar a execução do programa ou desligar os equipamentos, mas estas alterações só podem ocorrer antes de o objeto interromper o feixe de luz de um dos sensores e/ou depois do comando de habilitar o robô.

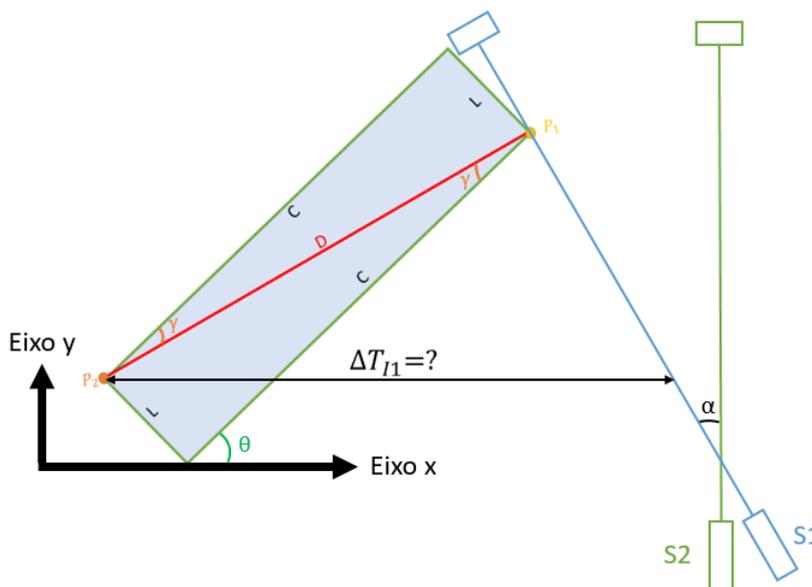
4.2.2 Estabelecendo o Sinal de θ

As equações 4.3 a 4.6 não são suficientes para determinarmos a orientação do objeto. Isso se deve porque um mesmo corpo retangular com orientação θ ou $-\theta$ vai interromper o feixe de S_2 em um mesmo intervalo de tempo ΔT_{I2} fazendo ser fundamental a presença do sensor S_1 com um deslocamento angular (α) e a análise de ΔT_{I1} .

Se faz necessário distinguir o intervalo de interrupção informado pelo sensor S_1 (ΔT_{I1}) com o estimado matematicamente em função de dados do sistema ($\Delta T'_{I1}$), uma vez que será necessário comparar ambos para determinar θ .

A formulação matemática relacionando sinal e módulo de θ com ΔT_{I1} , V , C , L , θ , γ e α é ilustrada em um exemplo de objeto com ângulo $\theta \geq 0^\circ$ e $|\theta| \geq \alpha$:

Figura 28 – Definindo o período de interrupção do feixe de S_1 (ΔT_{I1})



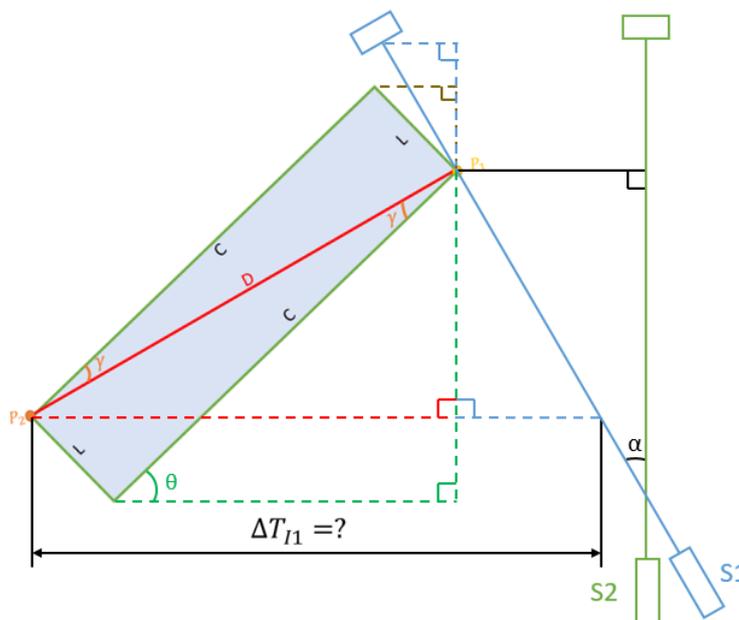
Fonte: Autor

Como mostrado na figura 28, ΔT_{I1} é o período entre a interrupção do feixe de S_1 pelo ponto P_2 e a passagem de P_1 pelo feixe de S_1 . ΔT_{I1} está relacionado com a distância horizontal do ponto P_2 ao feixe de S_1 quando este é interrompido por P_1 (X_1) pela fórmula 4.8:

$$\Delta T_{I1} = X_1/V [s] \tag{4.8}$$

Analogamente ao ΔT_{I2} , estimamos triangulações para determinar ΔT_{I1} , figura 29:

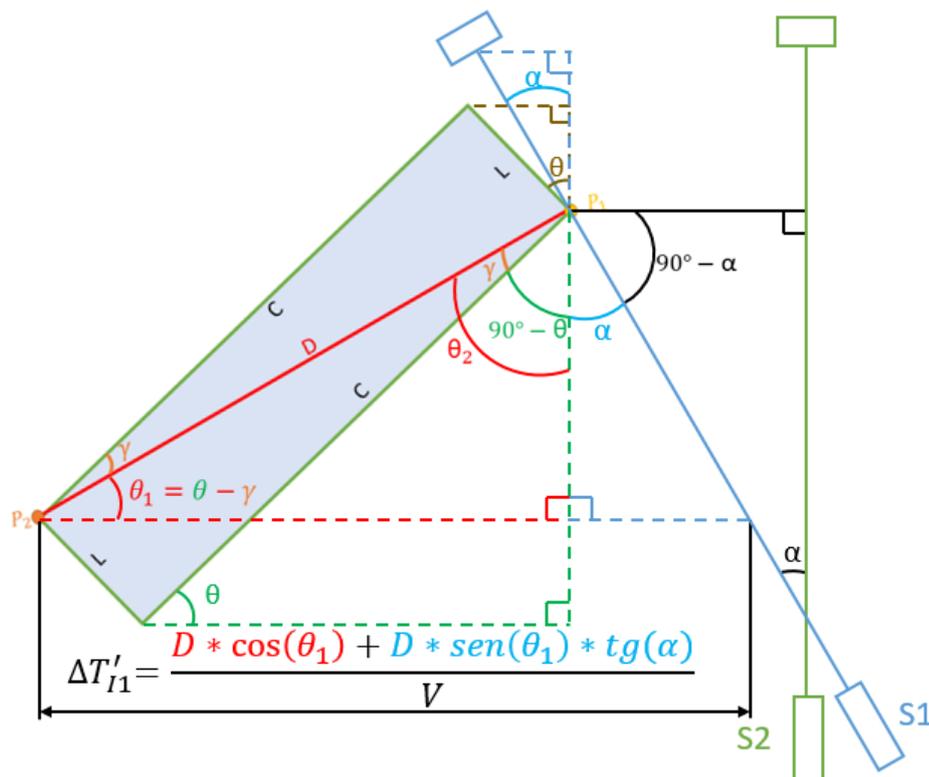
Figura 29 – Determinando ΔT_{I1} por triangulação



Fonte: Autor

Pela figura 28 é possível notar que o comprimento da reta X_1 pode ser dividido em duas partes: uma reta vermelha formada pela projeção da diagonal do objeto sobre um eixo paralelo ao eixo x determinado no instante em que P_1 interrompe o feixe de S_1 sendo, a extensão desta reta, delimitada pelo ponto P_2 e o feixe de S_1 e por uma reta azul em função de α e da diagonal do objeto. Definindo os ângulos internos dos triângulos estabelecidos por triangulação obtemos X_1 , figura 30 e fórmula 4.9:

Figura 30 – Estabelecendo $\Delta T'_{I1}$ por triangulação quando $\theta \geq 0^\circ$ e $|\theta| \geq \alpha$



Fonte: Autor

Temos:

$$X_1 = D * \cos(\theta_1) + D * \text{sen}(\theta_1) * \text{tg}(\alpha) \text{ [mm]} \quad (4.9)$$

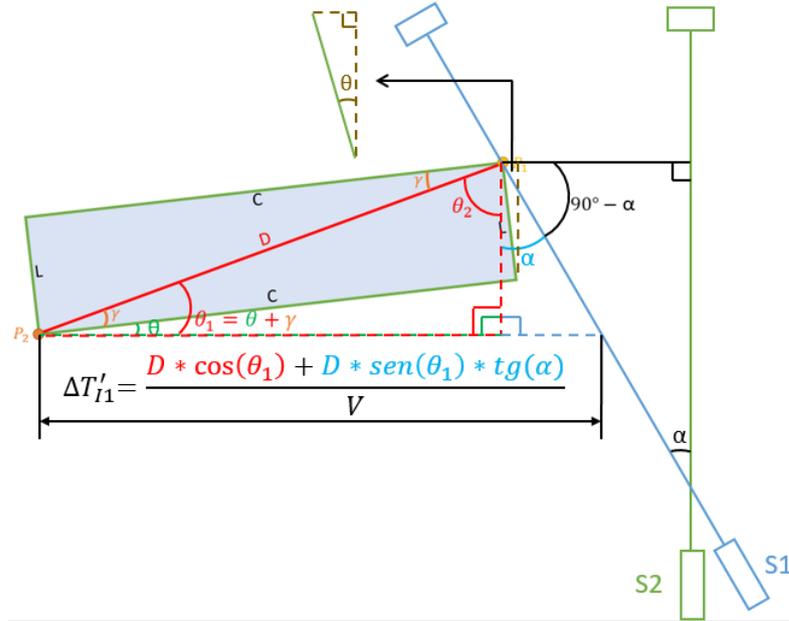
$$\Delta T'_{I1} = [D * \cos(\theta_1) + D * \text{sen}(\theta_1) * \text{tg}(\alpha)] / V \text{ [s]} \quad (4.10)$$

Onde:

$$\theta_1 = |\theta| - \gamma \text{ [}^\circ\text{]} \quad (4.11)$$

De modo análogo temos quando um objeto possui ângulo $\theta \geq 0^\circ$ e $|\theta| < \alpha$:

Figura 31 – Estabelecendo $\Delta T'_{I1}$ por triangulação quando $\theta \geq 0^\circ$ e $|\theta| < \alpha$



Fonte: Autor

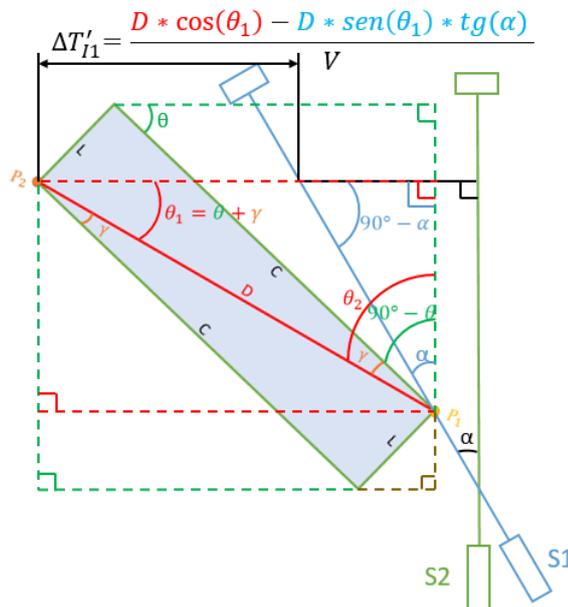
Se $\theta \geq 0^\circ$ e $|\theta| < \alpha$, as fórmulas 4.9 e 4.10 são aplicáveis, mas:

$$\theta_1 = |\theta| + \gamma [^\circ] \quad (4.12)$$

Percebe-se que X_1 é igual para qualquer orientação no intervalo $0^\circ \leq \theta \leq 90^\circ$. Isto não ocorre quando θ assume valores negativos como pode ser visto nos dois exemplos a seguir.

Para um corpo retangular com deslocamento linear $\theta < 0^\circ$ e $90^\circ - |\theta| \geq \alpha$ relativo ao eixo X:

Figura 32 – Definindo $\Delta T'_{I1}$ por triangulação quando $\theta < 0^\circ$ e $90^\circ - |\theta| \geq \alpha$



Fonte: Autor

Descobrimos:

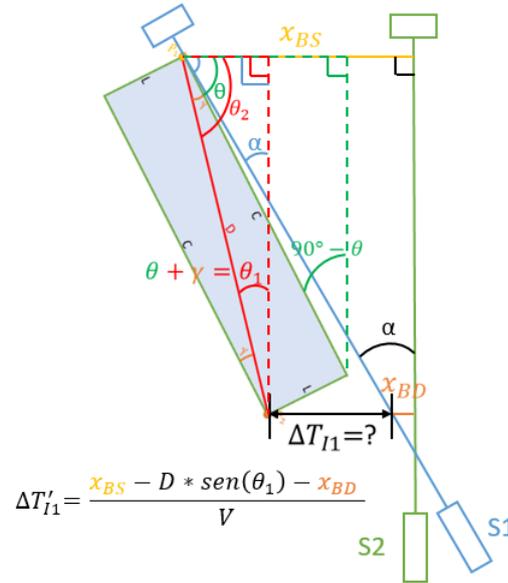
$$X_1 = D * \cos(\theta_1) - D * \text{sen}(\theta_1) * \text{tg}(\alpha) \text{ [mm]} \quad (4.13)$$

$$\Delta T'_{I1} = [D * \cos(\theta_1) - D * \text{sen}(\theta_1) * \text{tg}(\alpha)]/V \text{ [s]} \quad (4.14)$$

Sendo que θ_1 é definido pela fórmula 4.12.

Ainda é possível termos o caso em que $\theta < 0^\circ$ e $90^\circ - |\theta| < \alpha$:

Figura 33 – Determinando $\Delta T'_{I1}$ por triangulação quando $\theta < 0^\circ$ e $90^\circ - |\theta| < \alpha$



Fonte: Autor

Deste maneira estabelecemos:

$$X_1 = x_{BS} - D * \text{sen}(\theta_1) - x_{BD} \text{ [mm]} \quad (4.15)$$

$$\Delta T'_{I1} = [x_{BS} - D * \text{sen}(\theta_1) - x_{BD}]/V \text{ [s]} \quad (4.16)$$

De tal forma que θ_1 é definido pela equação 4.12 e:

x_{BS} : deslocamento horizontal do ponto P_1 entre a borda de subida do sinal dos sensores S_1 e S_2 ;

x_{BD} : deslocamento horizontal do ponto P_2 entre a borda de descida do sinal dos sensores S_1 e S_2 .

Tendo em vista a complexidade de determinar ΔT_{I1} para $\theta < 0^\circ$ e as equações 4.9 e 4.10 servirem para qualquer $0^\circ \leq \theta \leq 90^\circ$, é mais fácil verificar se θ é positivo aplicando a fórmula:

$$0.95 * \Delta T_{I1} < \Delta T'_{I1} < 1.05 * \Delta T_{I1} \text{ [s]} \quad (4.17)$$

Sendo que ΔT_{I1}^i é definido pela equação 4.10. Caso a fórmula 4.17 seja verdadeira, ou seja, se o desvio de ΔT_{I1} for até de $\pm 5\%$ em comparação com ΔT_{I1}^i :

$$\theta = |\theta| [^\circ] \quad (4.18)$$

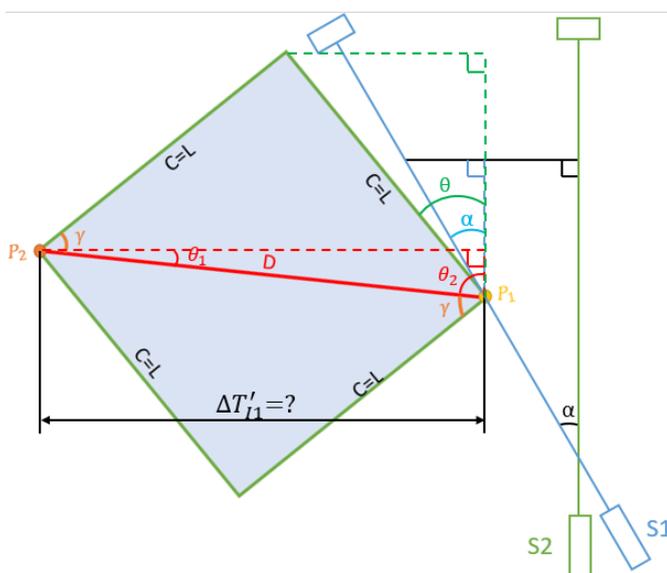
Senão:

$$\theta = -|\theta| [^\circ] \quad (4.19)$$

A margem de erro de $\pm 5\%$ da equação 4.17 visa a compensar os erros associados aos dispositivos como cálculos do CLP realizados com casas decimais limitadas, oscilações indesejadas na velocidade da esteira e medições imprecisas de sensores.

Se o objeto for retangular e $|\theta| < 45^\circ$:

Figura 34 – Definindo ΔT_{I1}^i por triangulação quando $|\gamma = 45^\circ$ e $|\theta| < 45^\circ$



Fonte: Autor

Não é possível determinar ΔT_{I1}^i por triangulação quando P_2 está acima de P_1 . Contudo, este problema não ocorre quando $|\theta| \geq 45^\circ$. Portanto, se $\gamma = 45^\circ$ e $|\theta| < 45^\circ$, aplicamos a fórmula 4.20 antes das fórmulas 4.8 a 4.19 tendo em vista que se um objeto retangular possuir uma orientação θ , esta equivale a $-(90^\circ - \theta)$.

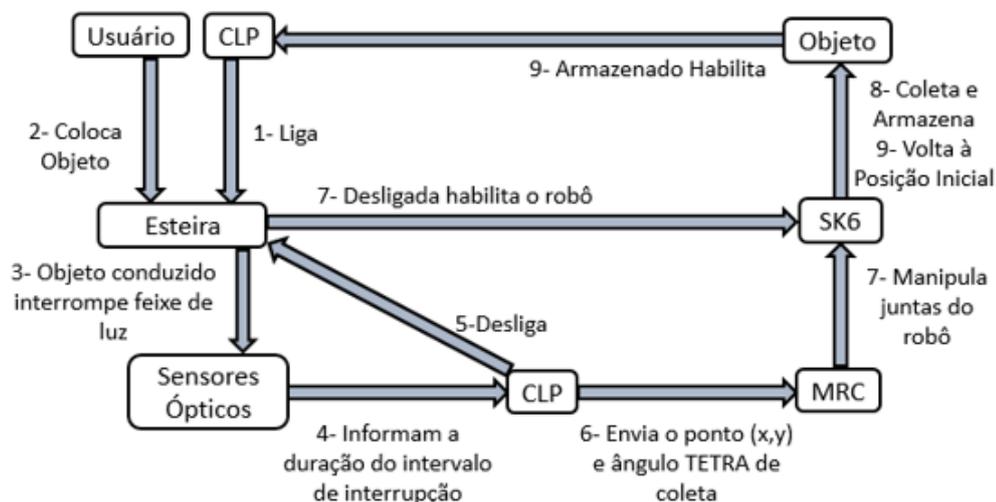
$$|\theta| = 90^\circ - |\theta| [^\circ] \quad (4.20)$$

4.3 Controlando Componentes do Sistema

Neste projeto estão sendo utilizados uma esteira, um Controlador Lógico Programável (CLP) Telemecanique TSX3722001, um computador executando o *software* PL7 PRO®, dois sensores ópticos BEN5M-MFR e caixas retangulares com comprimento de 134 mm,

largura de 68 mm e altura de 54 mm . A interação dos componentes deste projeto é mostrada na figura 35:

Figura 35 – Fluxograma do projeto



Fonte: Autor

Ao inicializar a programação elaborada por meio do programa PL7 Pro® e armazenada na memória do TSX, o CLP liga a esteira. Como não há uma máquina para colocar o objeto sobre a esteira, como ocorre em uma linha de produção, é necessário um usuário realizar essa função.

O objeto conduzido pela esteira interrompe o feixe de luz de cada um dos sensores S_1 e S_2 os quais informam a duração do intervalo de interrupção de seu respectivo feixe de luz, ΔT_{I1} e ΔT_{I2} , ao CLP que, com estes dados e mais os informados durante a elaboração do programa em Linguagem Ladder: velocidade da esteira (V), comprimento do objeto (C), largura do objeto (L) e deslocamento angular do sensor S_1 relativo ao sensor fixo S_2 (α), realiza os cálculos programados para obter a orientação do produto retangular. Em seguida, converte o módulo e sinal de θ de numeral inteiro para booleano. Se o bit associado à cada terminal de saída do CLP receber a lógica booleana “verdadeiro”, haverá condução de energia, se receber o valor “falso”, permanecerá a não ter passagem de energia.

Desta forma, em um projeto futuro, ao conectar os terminais de saída do CLP a entradas do controlador Yasnac MRC, figura 36, será possível informar a orientação real da caixa retangular a fim de o controlador realizar cálculos de cinemática inversa adequando a trajetória de coleta do robô Motoman SK6, figura 37, caso seja necessário. Um temporizador é acionado tendo em vista a necessidade de fornecer um intervalo para o robô coletar e armazenar o objeto antes de acionar novamente a esteira para repetir todo o processo.

Figura 36 – Cabine do controlador Yasnac MRC



Fonte: Autor

Figura 37 – Braço Robótico Motoman SK6



Fonte: Autor

4.4 Programação do Cálculo da Orientação de Coleta de Objetos Retangulares

Elaboramos um programa em Linguagem Ladder com o *software* PL7 PRO® objetivando validar a formulação teórica desenvolvida neste projeto. Foi descrita as partes mais relevantes do programa aplicativo.

Para melhor esclarecimento, listamos a nomenclatura dos principais elementos utilizados em Ladder:

%I1.N: entrada número N;

%Q2.N: saída número N;

%MWN: memória de 16 bits capaz de armazenar um número inteiro;

%MFN: memória capaz de armazenar um ponto flutuante, ou seja, um número real;

%TMN: temporizador;

%MNN: monoestável (gera um pulso por um período determinado);

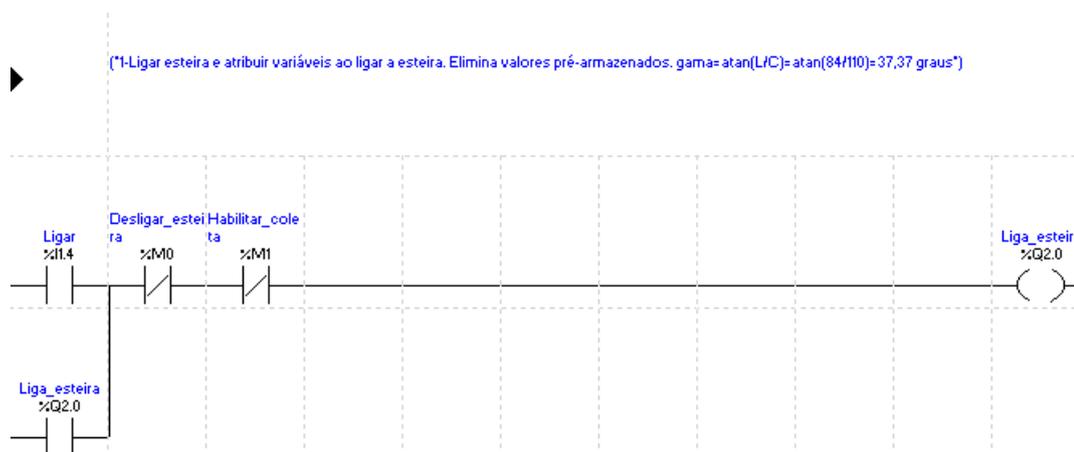
%CN: contador de eventos;

%CN.V: armazena o valor atual da contagem do contador.

4.4.1 Acionamento da Esteira

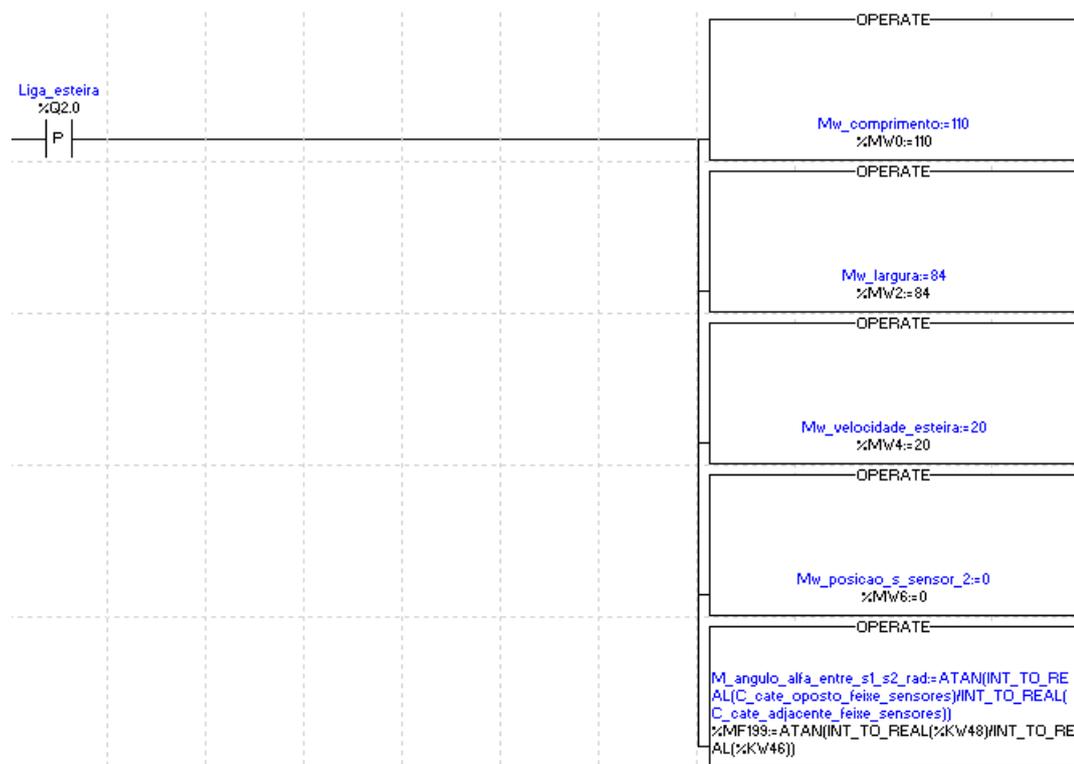
A esteira é acionada e C , L , V e α são declarados quando o programa entra em modo de execução:

Figura 38 – Acionamento da esteira



Fonte: Autor

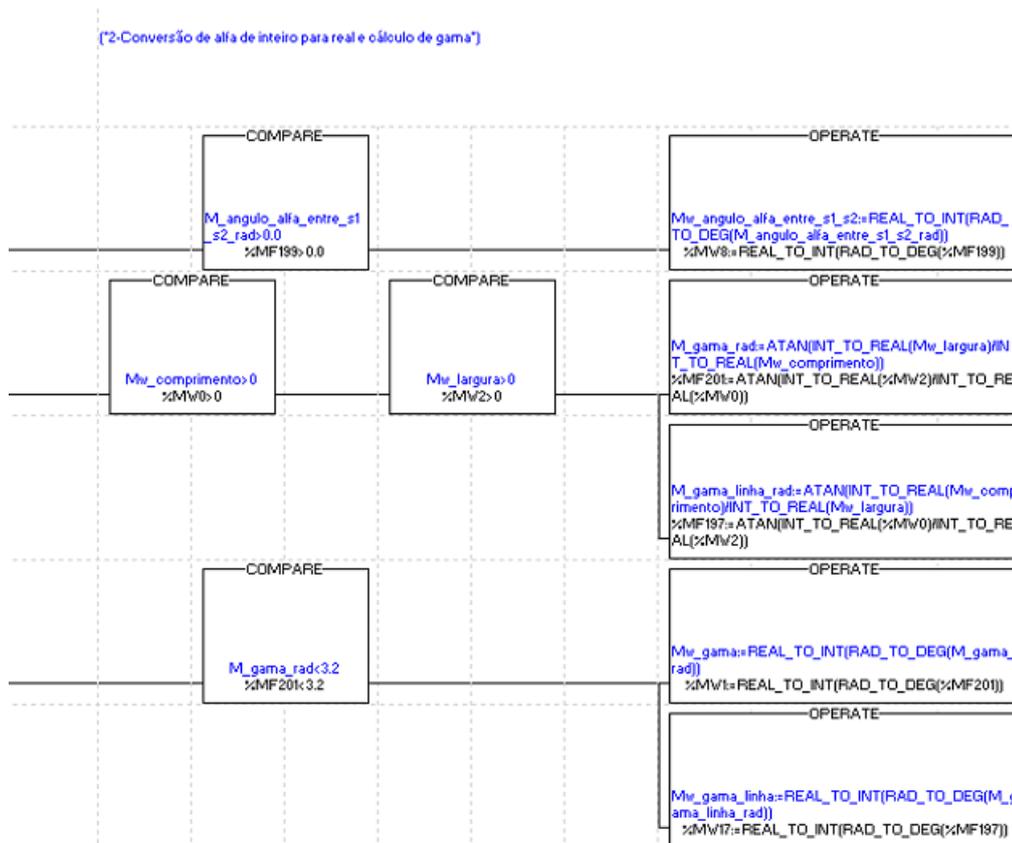
Figura 39 – Atribuição de C , L , V e α



Fonte: Autor

Em seguida podemos obter γ e γ' :

Figura 40 – Calculando γ e γ'



Fonte: Autor

4.4.2 Leitura dos Sensores Ópticos e Definição da Orientação de Coleta

A borda de subida do sinal do sensor S_1 aciona a contagem de ΔT_{I1} :

Figura 41 – Evento ocorrido na borda de subida do sinal de S_1

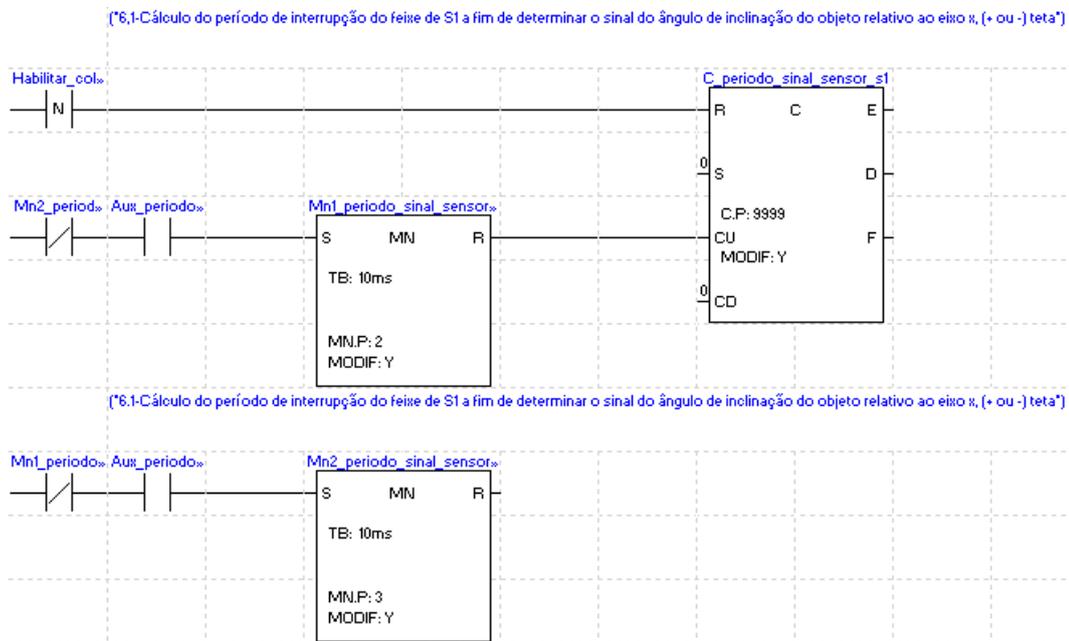


Fonte: Autor

Tentamos utilizar um temporizador na entrada conectada ao sensor S_1 e outro na do sensor S_2 a fim de obter a leitura do valor armazenado para realizar os cálculos. Contudo, o valor armazenado no temporizador era zerado na borda de descida dos sinais dos sensores. Por isso, encontramos um meio alternativo para obter ΔT_{I1} e ΔT_{I2} utilizando um contador com um monoestável em sua entrada de contagem crescente. O acionamento do primeiro monoestável está associada ao desacionamento do segundo monoestável do mesmo modo que o acionamento do segundo está interligado ao desacionamento do primeiro gerando, dessa fora, um ciclo de pulsos contáveis por um contador enquanto os contadores de

entrada estiverem acionados. Ao terminar o período de coleta do robô, no fim do ciclo da programação, o contador é reiniciado (é realizado um *reset*).

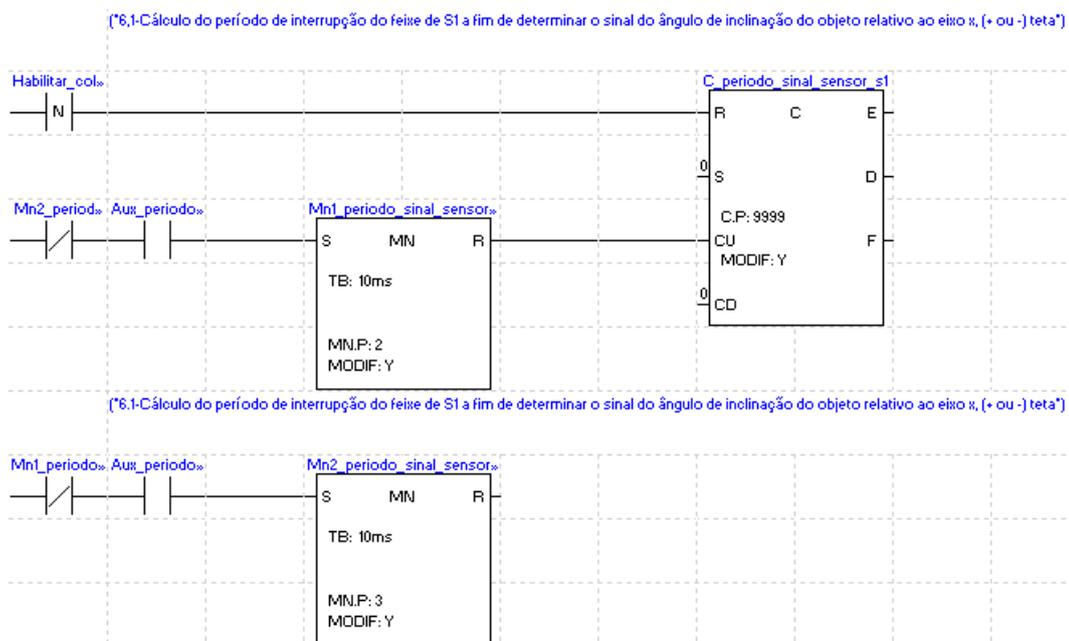
Figura 42 – Contagem de ΔT_{I1}



Fonte: Autor

A borda de descida do S_1 desabilita os dois monoestáveis associados ao contador $C_perodo_sinal_sensor_s1$, figura 42. Em seguida, ΔT_{I1} é calculado:

Figura 43 – Borda de descida de S_1 e cálculo de ΔT_{I1}

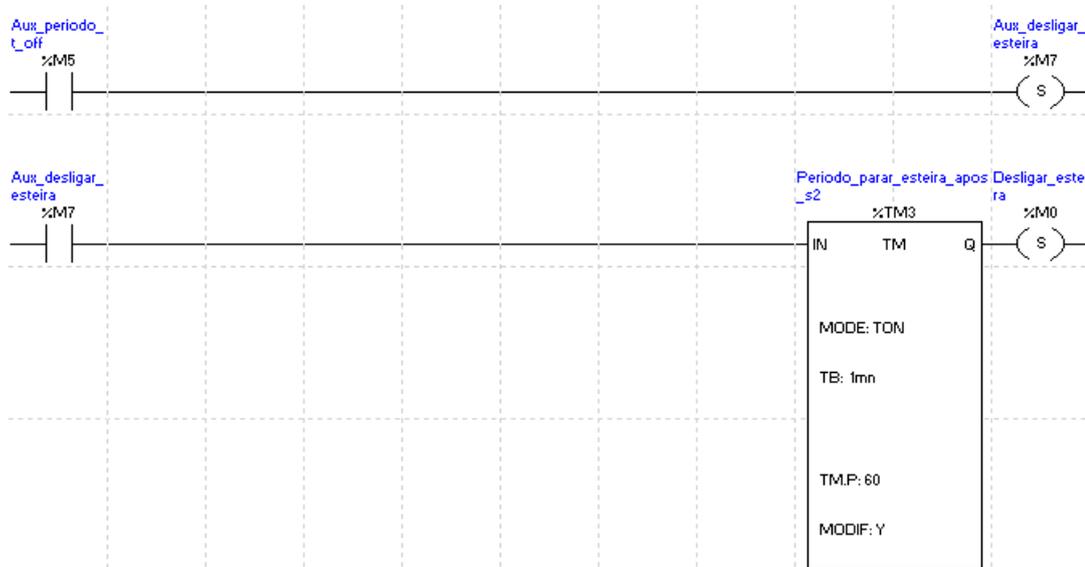


Fonte: Autor

Na borda de subida de S_2 há o acionamento de um contador e dois monoestáveis para a contagem de ΔT_{I2} de forma análoga à contagem de ΔT_{I1} e o temporizador com o valor

de ΔT_{Off} armazenado é iniciado objetivando garantir a passagem completa do objeto pelo feixe de S_2 .

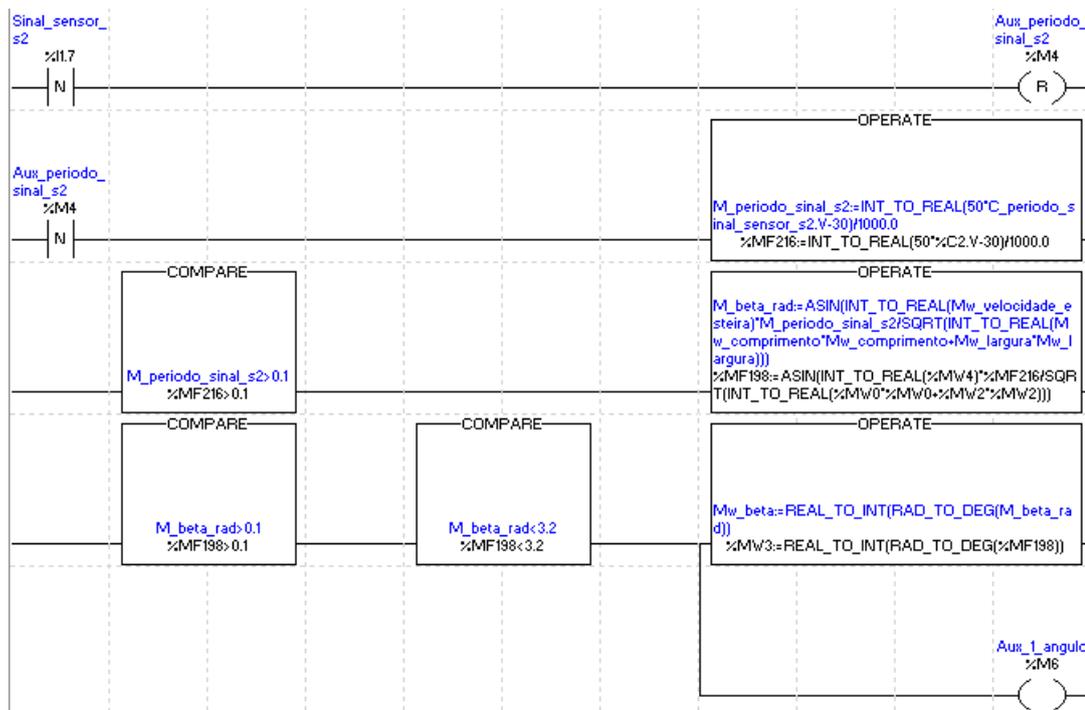
Figura 44 – Acionamento do temporizador ΔT_{Off}



Fonte: Autor

A borda de descida do S_2 desabilita a contagem de ΔT_{I2} de forma similar ao ocorrido com ΔT_{I1} . Logo após, ΔT_{I2} e β são calculados:

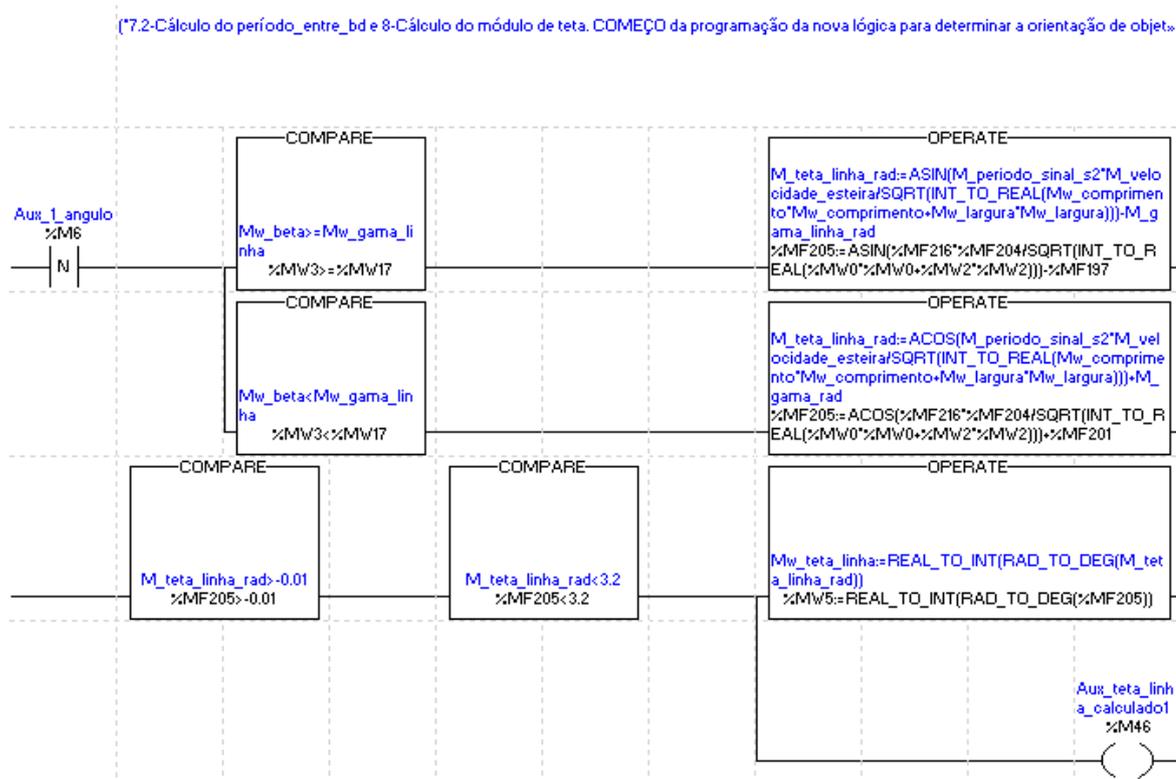
Figura 45 – Borda de descida de S_2 e cálculo de ΔT_{I2} e β



Fonte: Autor

β é utilizado na fórmula 4.6 para determinamos $|\theta|$:

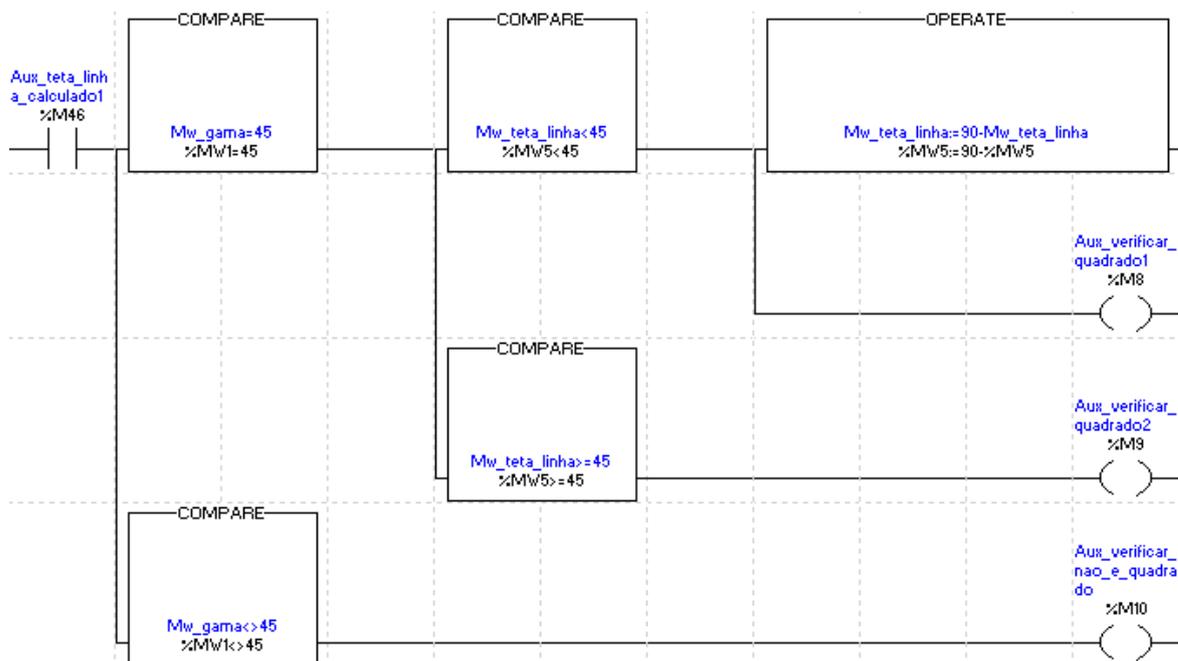
Figura 46 – Cálculo do módulo de θ



Fonte: Autor

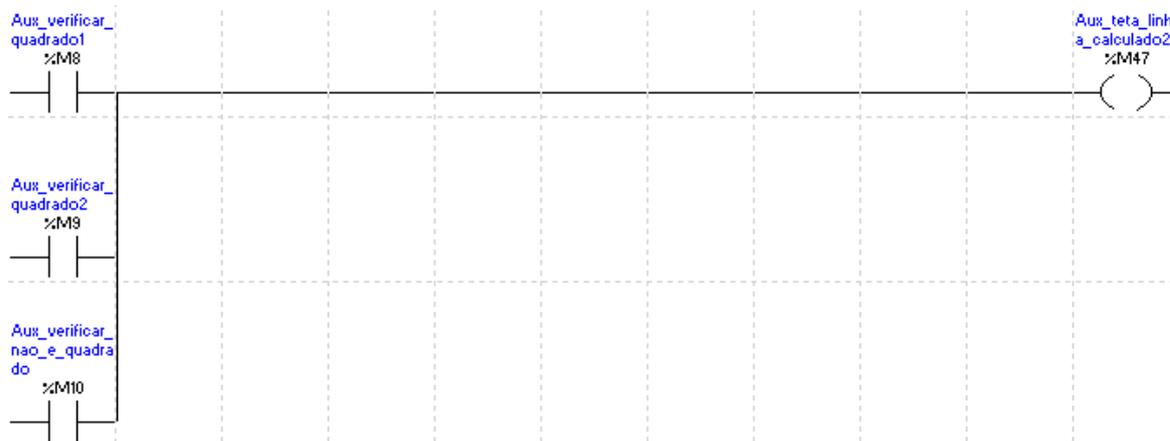
Agora podemos determinar o sinal de θ pelas fórmulas 4.9, 4.10, 4.17, 4.18, 4.19 e 4.20:

Figura 47 – Determinação do sinal de θ parte 1



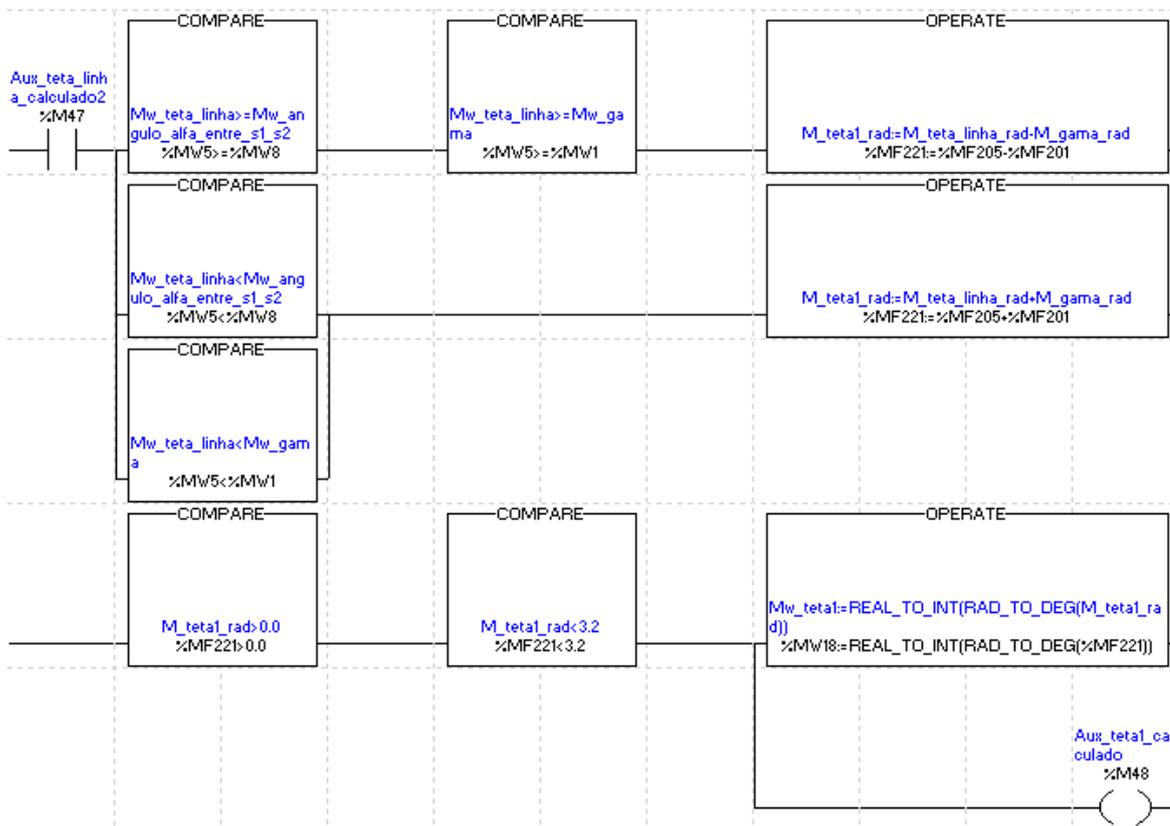
Fonte: Autor

Figura 48 – Determinação do sinal de θ parte 2



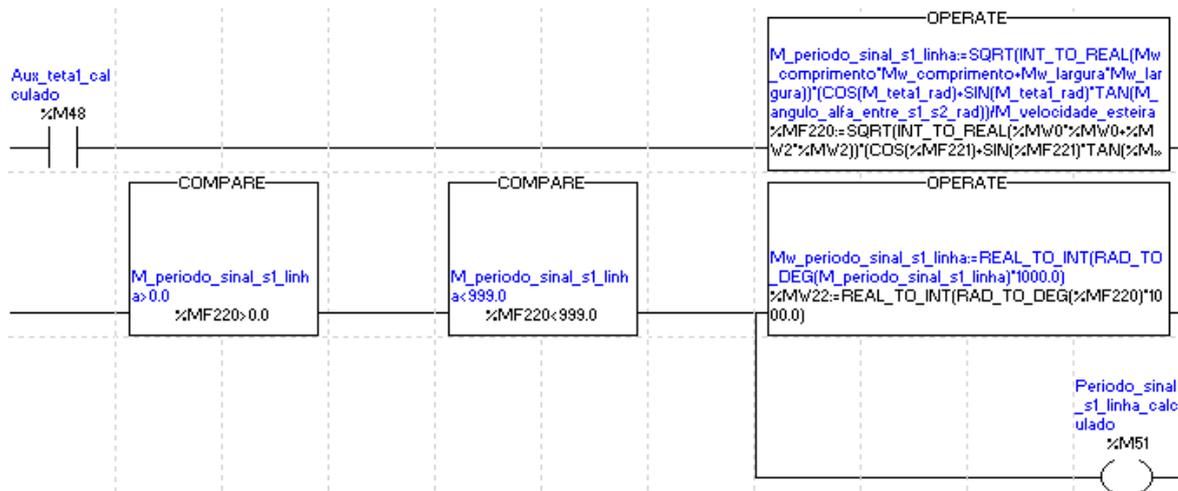
Fonte: Autor

Figura 49 – Determinação do sinal de θ parte 3



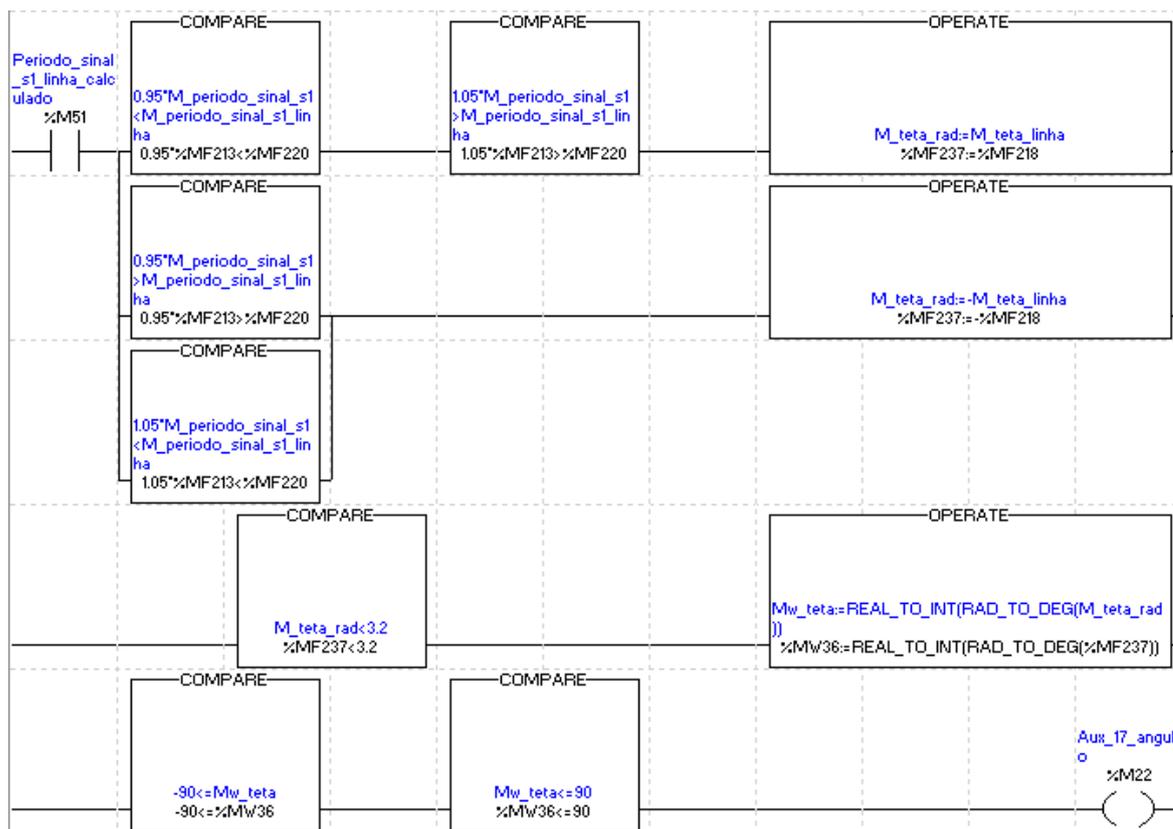
Fonte: Autor

Figura 50 – Determinação do sinal de θ parte 4



Fonte: Autor

Figura 51 – Determinação do sinal de θ parte 5



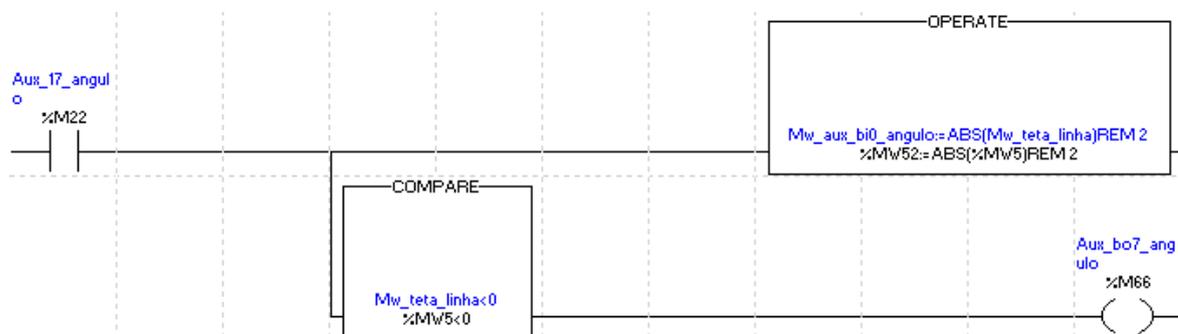
Fonte: Autor

4.4.3 Conversão dos Resultados Obtidos de Numeral Inteiro para Lógica Booleana

Após obtermos a orientação nos formatos numeral real e inteiro, podemos converter o sinal de θ para booleano atribuindo ao bit Aux_bo7_angulo o valor lógico “verdade” caso

seja negativo ou o valor lógico “falso” se for positivo:

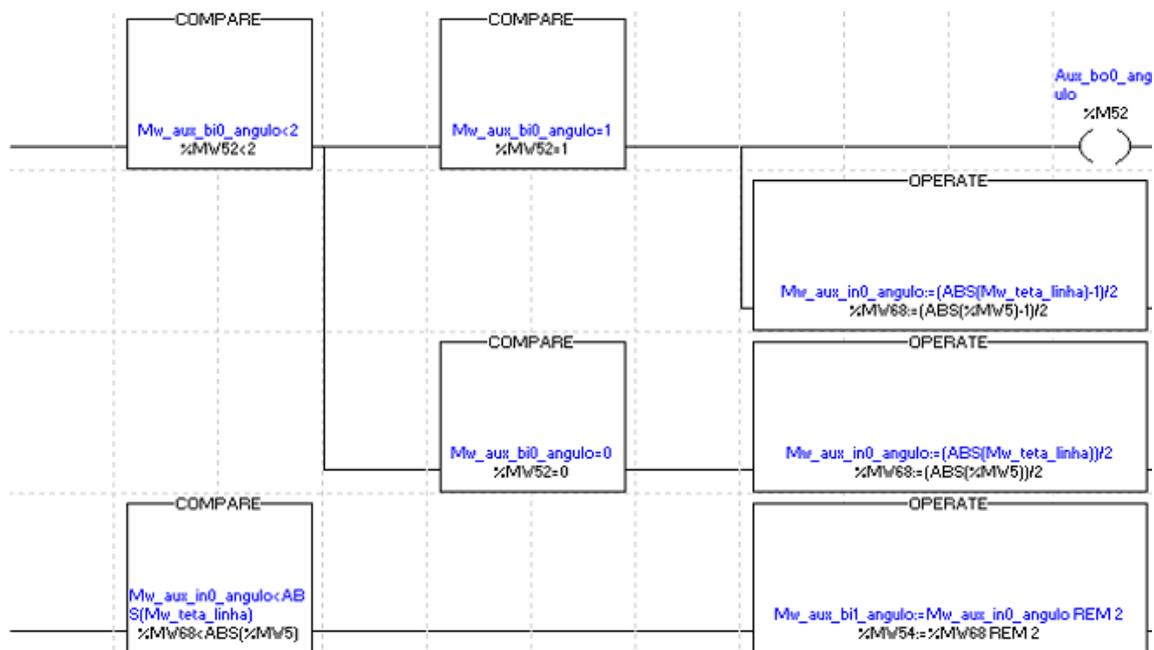
Figura 52 – Conversão para booleano do sinal de teta



Fonte: Autor

Logo após, o módulo de θ é convertido de numeral inteiro para numeral binário e lógica booleana:

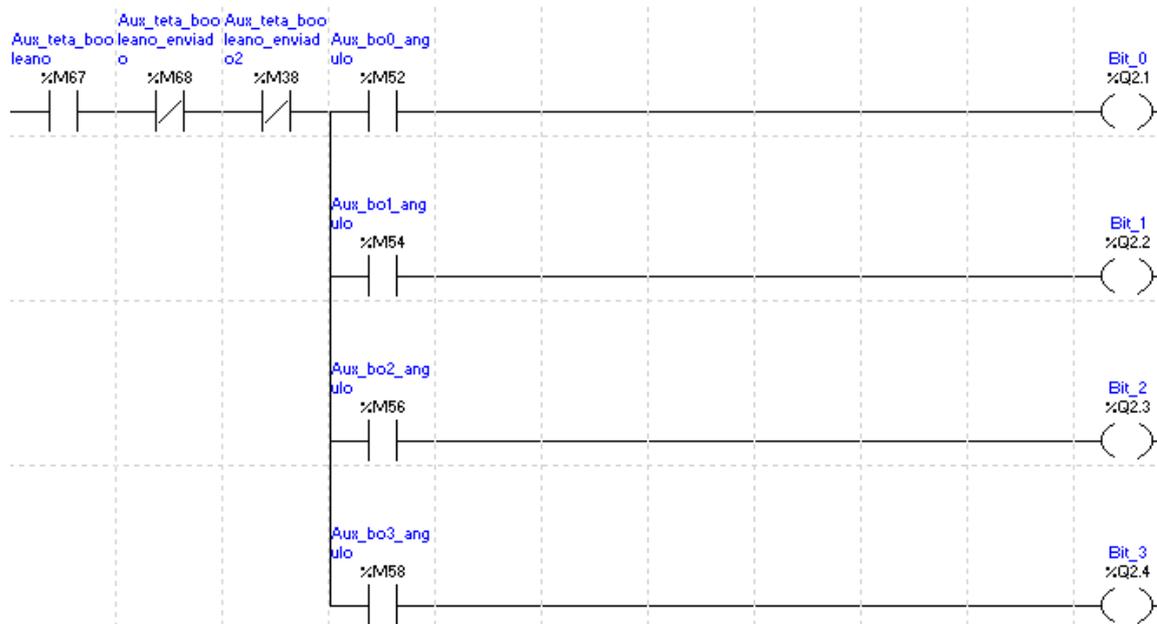
Figura 53 – Conversão do módulo de θ de Inteiro para Booleano equivalente



Fonte: Autor

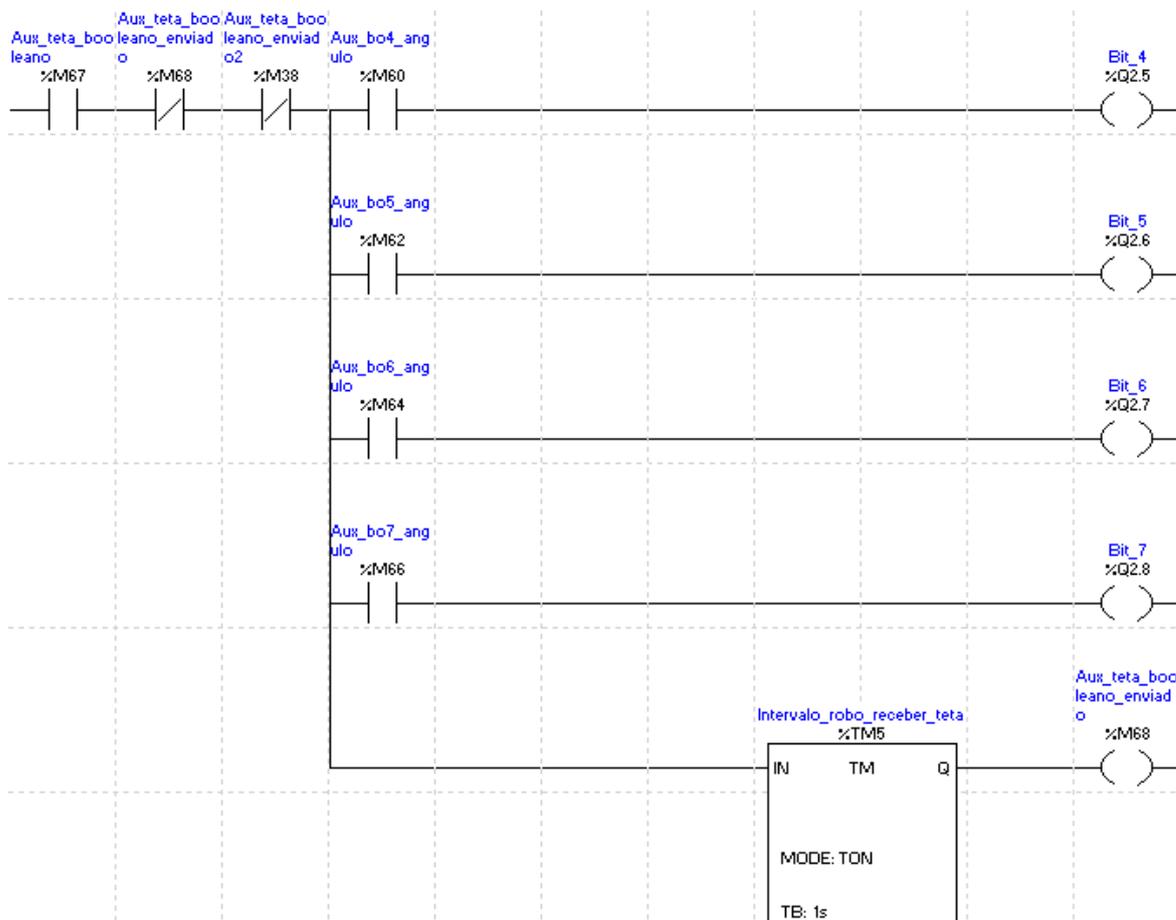
Os dados referentes à orientação são disponibilizados nos terminais de saída Bit_0 a Bit_7 da seguinte forma:

Figura 54 – Transmissão de θ nos terminais de saída do CLP parte 1



Fonte: Autor

Figura 55 – Transmissão de θ nos terminais de saída do CLP parte 2



Fonte: Autor

Estes dados permanecem legíveis nos terminais Bit_0 a Bit_7 durante o acionamento

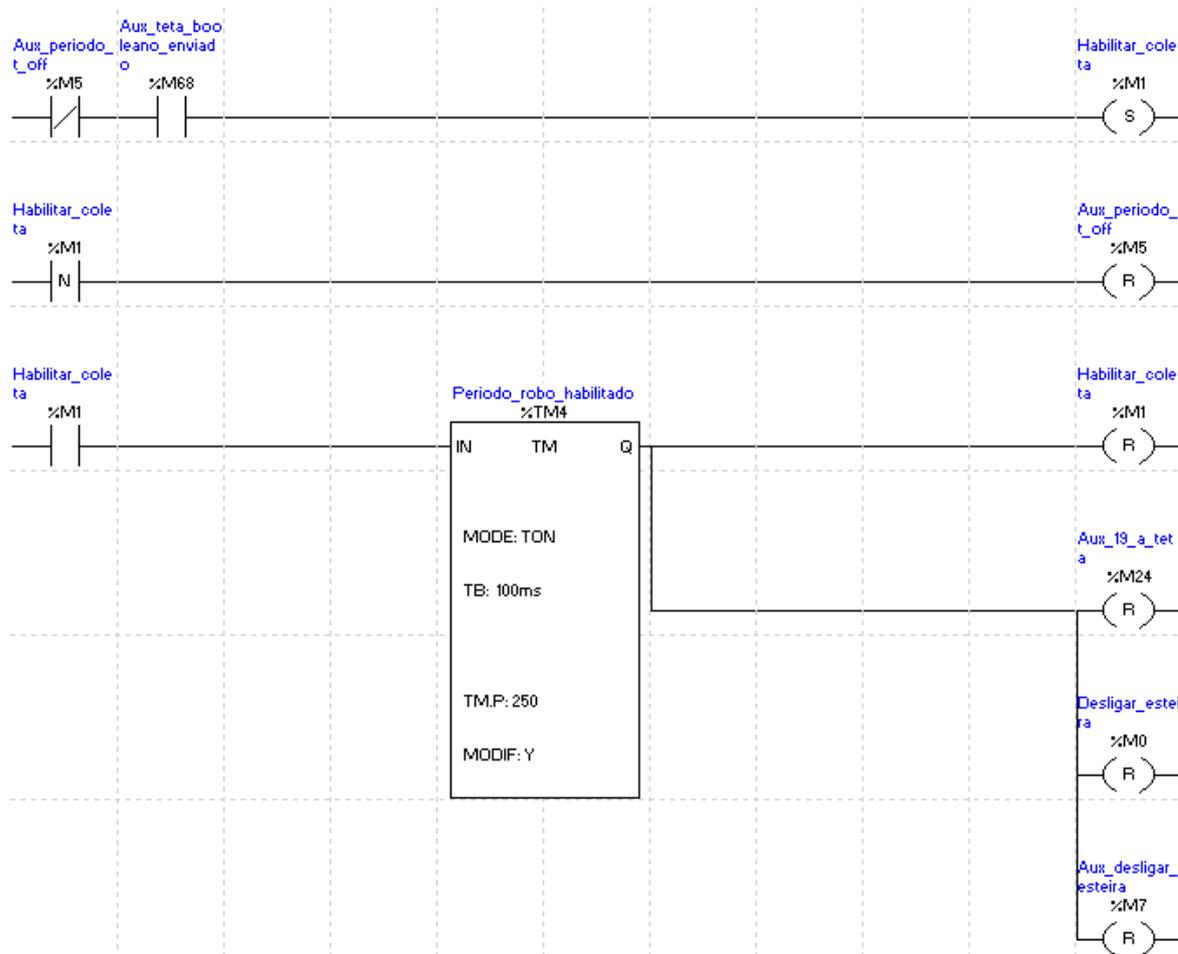
do temporizador Intervalo_ robo_ receber_ teta, figura 55.

Movendo a caixa manualmente conseguimos confirmar o cálculo da orientação de θ verificando os terminais Bit_0 a Bit_7.

4.4.4 Habilitar Robô e Zerar Variáveis

Se a esteira estiver parada e o intervalo definido pelo temporizador Intervalo_ robo_ receber_ teta objetivando fornecer um período suficiente para um controlador de robô possa ler os bits Bit_0 a Bit_7. A memória Habilitar_ coleta, figura 56, é desabilitada objetivando desacionar a operação de coleta do robô, apagar os dados de variáveis armazenados em memórias, ligar novamente a esteira e iniciar um novo ciclo preparando o programa a realizar novamente todas as etapas descritas.

Figura 56 – Habilitando o robô e zerando as variáveis no fim do ciclo do programa



Fonte: Autor

5 RESULTADOS E DISCUSSÕES

Neste capítulo abordamos e relatamos os resultados obtidos por meio da metodologia aplicada durante o planejamento e execução deste projeto.

5.1 Determinação do Sistema de Percepção Robótica

Na seção 4.1 propomos os componentes necessários para montar um sistema de percepção robótica a fim de indentificar a orientação de objetos retangulares aleatoriamente orientados enquanto são conduzidos por uma esteira a partir do período de interrupção dos sinais dos sensores ópticos S_1 e S_2 , ΔT_{I1} e ΔT_{I2} , respectivamente.

Figura 57 – Sistema de percepção robótica proposto



Fonte: Autor

Na seção 4.3 definimos sistematicamente o princípio de funcionamento do sistema da figura 57 na determinação da orientação de corpos retangulares relativo a um eixo fixo (θ) enquanto são conduzidos por uma esteira, o controle do acionamento de uma esteira e a disponibilização de θ em formato Booleano nos terminais de saída do CLP Telemecanique TSX3722001 para que, em um projeto futuro, um controlador de manipulador robótico possa receber a informação do valor de θ a fim de adaptar a orientação de agarre do manipulador.

5.2 Modelagem e Cálculo da Orientação de Agarre a partir de Variáveis do Sistema

Determinamos a orientação θ de objetos retangulares aleatoriamente orientados conduzidos por uma esteira em função de variáveis dos mecanismos os quais compõem o

sistema utilizado neste trabalho como: intervalo de nível alto da saída de S_1 ou período de interrupção do feixe de S_1 (ΔT_{I1}), intervalo de nível alto da saída de S_2 ou período de interrupção do feixe de S_2 (ΔT_{I2}), velocidade da esteira (V), comprimento (C) e largura (L) do objeto, gama (γ) e deslocamento angular do sensor S_1 relativo ao sensor fixo S_2 (α).

Primeiramente, obtemos o módulo da orientação ($|\theta|$) a partir de um ΔT_{I2} calculado a partir do período em que o sensor S_2 permanece em estado alto:

$$|\theta| = \text{sen}^{-1}(V * \Delta T_{I2}/D) - \gamma' [^\circ] \quad (5.1)$$

$$|\theta| = \text{cos}^{-1}(V * \Delta T_{I2}/D) + \gamma [^\circ] \quad (5.2)$$

Sendo que:

$$D = \sqrt{C^2 + L^2} [mm] \quad (5.3)$$

$$\gamma = \text{tg}^{-1}\left(\frac{L}{C}\right) = 90^\circ - \gamma' [^\circ] \quad (5.4)$$

$$\gamma' = \text{tg}^{-1}\left(\frac{C}{L}\right) = 90^\circ - \gamma [^\circ] \quad (5.5)$$

Porém, se utilizarmos as fórmulas 5.1 e 5.2 de forma arbitrária, podemos encontrar resultados negativos para $|\theta|$. Portanto, é necessário verificar a seguinte condição:

$$\beta = \text{sen}^{-1}(V * \Delta T_{I2}/D) \geq \gamma' [^\circ] \quad (5.6)$$

Caso a equação 5.6 seja verdadeira, $|\theta|$ é obtido da equação 5.1. Se não for, a equação 5.2 é empregada.

Se o objeto for quadrado e $|\theta| < 45^\circ$, utilizamos a fórmula 4.20 tendo em vista que nem sempre é possível determinar ΔT_{I1} por triangulação e esta limitação não ocorre quando $|\theta| \geq 45^\circ$.

Distinguimos o período em que o sensor S_1 permanece em estado alto (ΔT_{I1}) com o estimado matematicamente em função de dados do sistema (ΔT_{I1}^i), haja vista que será necessário comparar ambos para determinar θ .

Então, calculamos ΔT_{I1}^i a partir de V , C , L , $|\theta|$, γ e α :

$$\Delta T_{I1}^i = [D * \text{cos}(\theta_1) + D * \text{sen}(\theta_1)]/V [s] \quad (5.7)$$

Onde, se $|\theta| \geq \alpha$ e $|\theta| \geq \gamma$:

$$\theta_1 = |\theta| - \gamma [^\circ] \quad (5.8)$$

Ou se $|\theta| < \alpha$ ou $|\theta| < \gamma$:

$$\theta_1 = |\theta| + \gamma [^\circ] \quad (5.9)$$

Por último, determinamos o sinal de θ a partir de $|\theta|$, ΔT_{I1} e ΔT_{I1} verificando a seguinte condição:

$$0.95 * \Delta T_{I1} < \Delta T_{I1} < 1.05 * \Delta T_{I1} [s] \quad (5.10)$$

Se a condição 5.10 for verdadeira, então:

$$\theta = |\theta| [^\circ] \quad (5.11)$$

Senão:

$$\theta = -|\theta| [^\circ] \quad (5.12)$$

A margem de erro de $\pm 5\%$ da equação 5.10 objetiva compensar as imprecisões dos dispositivos como cálculos do CLP realizados com casas decimais limitadas e oscilações indesejadas na velocidade da esteira.

5.3 Programação em Linguagem Ladder do Cálculo da Orientação de Coleta de Objetos Retangulares Aleatoriamente Orientados

Convertemos a modelagem a orientação θ de objetos retangulares aleatoriamente orientados em um programa aplicativo para o CLP TSX3722, como pode ser visto em formato digital e no tópico 4.4.

Após armazenarmos a orientação θ nos formatos numeral real e inteiro em memória, convertemos o sinal de θ para booleano atribuindo a um bit de memória o valor lógico “verdade” caso seja negativo ou o valor lógico “falso” se for positivo. Em seguida, armazenamos em seis bits o equivalente Booleano de $|\theta|$ em formato numeral inteiro.

Programamos o CLP para reter a informação em Booleano de θ nos terminais Bit_0 a Bit_7 durante o acionamento do temporizador Intervalo_robô_receber_teta, figura 55, para que um controlador de um manipulador robótico possa ler o valor de θ . Logo depois, este temporizador aciona o bit de memória Aux_teta_booleano_enviado, figura 55.

Um comando de habilitar a coleta do robô é acionado quando a esteira estiver parada e a memória Aux_teta_booleano_enviado estiver acionada, figura 56. Após completado o período de coleta do robô definido pelo temporizador Periodo_robô_habilitado, figura 56, a esteira é acionada novamente e as memórias são apagadas permitindo o programa a realizar novamente todas as etapas descritas.

Movendo a caixa manualmente obtivemos resultados satisfatórios ao verificar os terminais Bit_0 a Bit_7.

Houve alguns imprevistos e contratempos durante a elaboração do programa em Linguagem Ladder. Atribuímos um numeral constante à memória %KF0. Ao declarar um numeral na memória %KF1, perdia-se a informação na memória %KF0. O mesmo ocorria com as demais memórias de constantes pares e ímpares. Caso houvesse algum dado na memória %MF0 ou %MF1, ao atribuir um valor à memória %KF0 ou %KF1, os dados de %MF0 e %MF1 eram perdidos. Logo, foi decidido em não utilizar memórias %K. A diferença entre memórias %K e %M é percebida durante a execução do programa. As do tipo %M podem ser lidas e escritas, enquanto as %K só podem ser lidas.

6 CONCLUSÕES

O sensor ultrassônico Arduino HC-SR04 não é compatível com a proposta de determinar a orientação de objetos retangulares. Por outro lado, a aplicação de dois sensores ópticos retro reflexivos BEN5M-MFR e seus respectivos refletores com um CLP Telemecanique® Modicon TSX 3722 acoplado a um módulo de E/S TSX DMZ 28DR e conectado a um computador com sistema operacional Windows XP com o *software* PL7 PRO® mostrou-se ser um sistema de percepção robótico capaz de identificar a orientação de corpos retangulares (θ) conduzidos por uma esteira.

Os objetos retangulares situados sobre a esteira devem possuir o mesmo comprimento e largura. Caso deseja-se obter a orientação de objetos retangulares com comprimento e/ou largura diferentes ou mudar a inclinação do sensor S_1 relativa ao sensor S_2 , basta mudar o valor das suas respectivas constantes, declaradas na programação em Ladder, mas estas alterações só podem ocorrer antes de o objeto interromper o feixe de luz de um dos sensores e/ou depois do comando de habilitar o robô. Dessa forma, os métodos do processamento dos cálculos automaticamente são ajustados sem precisar desligar o robô ou o CLP ou desconectá-los.

Determinamos θ em função de variáveis dos equipamentos os quais compõem o sistema utilizado neste trabalho como: intervalo de nível alto da saída de S_1 ou período de interrupção do feixe de S_1 (ΔT_{I1}), intervalo de nível alto da saída de S_2 ou período de interrupção do feixe de S_2 (ΔT_{I2}), velocidade da esteira (V), comprimento (C) e largura (L) do objeto, gama (γ) e deslocamento angular do sensor S_1 relativo ao sensor fixo S_2 (α).

Convertemos a modelagem da orientação θ em um programa aplicativo de Linguagem Ladder por meio do *software* PL7 PRO®. Desta forma, quando os sensores informam ΔT_{I1} e ΔT_{I2} ao CLP, este informa θ em formato Booleano nos seus terminais de saída Bit_0 a Bit_7 por um intervalo de tempo arbitrário visando a possibilitar a leitura de θ por um controlador de um manipulador robótico. No programa aplicativo foram inseridos comandos a fim de o CLP acionar e parar a esteira, habilitar a coleta do objeto retangular pelo robô e apagar os dados armazenados em memória possibilitando o programa repetir ciclicamente a etapas descritas neste projeto.

Caso $\Delta T_{I1} = \Delta T_{I2}$ e se forem idênticos tanto quando θ for positivo ou negativo, é possível determinar somente o módulo da orientação do objeto $|\theta|$.

Quando $|\theta| = \gamma$, pequenas mudanças de ΔT_{I2} geram grandes variações de $|\theta|$ para qualquer $0^\circ < \gamma \leq 45^\circ$ tornando necessário um sensor capaz de mensurar ΔT_{I2} com boa precisão.

Se $|\gamma = 45^\circ$ e $|\theta| < 45^\circ$, nem sempre é possível realizar triangulações para determinar ΔT_{I1} .

Tentamos utilizar memórias do formato %K em conjunto com as do tipo %M objetivando facilitar a programação, mas ao declarar memórias %K, perdíamos informações armazenadas tanto no padrão %M quanto no %K. Logo, descartamos o uso de memórias %K.

Deslocando o objeto manualmente obtivemos resultados satisfatórios ao verificar os terminais Bit_0 a Bit_7.

6.1 Trabalhos futuros

a) A partir das informações obtidas na modelagem realizada visando ao agarre de objetos aleatoriamente posicionados e orientados, buscar técnicas de controle visando a uma solução mais simples para esta problemática;

b) Determinar o posicionamento de objetos retangulares com o auxílio da modelagem da orientação relativo a um eixo fixo desenvolvido neste projeto.

c) Modelar o posicionamento e orientação de agarre de objetos não retangulares a partir de informações sobre mudanças de estado dos mecanismos utilizados;

d) Criar um sensor inteligente capaz de ajustar automaticamente a orientação do sensor S1 relativa ao sensor S2 e processar as informações tratadas pelo CLP;

e) Utilizar, alternativamente, um sensor capaz de mensurar distâncias no desenvolvimento das quatro propostas de trabalho citadas anteriormente;

f) Empregar um manipulador com sensor capaz de confirmar o agarre do objeto e/ou uma esteira com recursos para enviar ao CLP o deslocamento angular ou linear realizado.

Referências

- AGUIRRE, L. A. *Enciclopédia de Automática: Controle e Automação, Volume I*. [S.l.]: Blucher, 2007a. 17, 18, 19, 20, 21
- AGUIRRE, L. A. *Enciclopédia de Automática: Controle e Automação, Volume II*. [S.l.]: Blucher, 2007b. 14
- AGUIRRE, L. A. *Enciclopédia de Automática: Controle e Automação, Volume III*. [S.l.]: Blucher, 2007c. 23, 28
- AMANDA, M. A. *Controle de Robôs. XV Curso de Automática en la Indústria*. [S.l.]: Aguadulce, 1995. 23
- ELETRIC, S. *Micro: Configuración del Modicon TSX Micro*. 2001. Disponível em: <<http://www.tecnologia-tecnica.com.ar/plc/Micro.pdf>>. 32
- HALLIDAY; RESNICK; WALKER, P. *Fundamentos de Física, Volume 3: Eletromagnetismo*. [S.l.]: LTC, 2010. 15
- MAZAIRA, I. *Robótica Industrial*. [S.l.], 2008. 28
- MORAES, C. C. d.; CASTRUCCI, P. d. L. *Engenharia de Automação Industrial*. [S.l.]: LTC, 2001. 17
- MORAES, C. C. d.; CASTRUCCI, P. d. L. *Engenharia de Automação Industrial*. [S.l.]: LTC, 2010. 14, 17, 18, 19
- NIKU, S. B. *Introdução à Robótica: Análise, Controle e Aplicações*. 2.ed. ed. [S.l.]: LTC, 2013. 22, 23, 26, 27
- ROMANO, V. *Robótica Industrial: Aplicações na Indústria de Manufatura e de Processo*. [S.l.]: Blucher Ltda, 2002. 23
- ROSÁRIO, J. M. *Princípios de Mecatrônica*. [S.l.]: Prentice Hall, 2005. 14, 15, 17, 22, 23, 24, 27
- SOUZA, L. E. *Arquitetura de Sistemas de Automação - Módulo 1*. Crti training center. Itajubá, 2005. 20
- THOMAZINI, D.; ALBUQUERQUE, P. U. B. d. *Sensores Industriais: Fundamentos e Aplicações*. 4.ed. ed. [S.l.]: Érica, 2007. 14