



TRABALHO DE GRADUAÇÃO

**CONSTRUÇÃO DE UMA GRUA PARA COLETA
DE MATERIAL FERROSO CONTROLADA POR CLP**

Daniel Fraianeli

Thiago Alberto Ferreira Adnet

Brasília, dezembro de 2009

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia

TRABALHO DE GRADUAÇÃO

**CONSTRUÇÃO DE UMA GRUA PARA COLETA
DE MATERIAL FERROSO CONTROLADA POR CLP**

Daniel Fraianeli

Thiago Alberto Ferreira Adnet

*Relatório submetido ao Departamento de Engenharia
Elétrica como requisito parcial para obtenção
do grau de Engenheiro de Controle e Automação*

Banca Examinadora

Prof. Geovany Araujo Borges, ENE/UnB
Orientador

Prof. João Yoshiyuki Ishihara, ENE/UnB
Co-orientador

Prof. Ricardo Zelenovsky, ENE/UnB
Examinador interno

Dedicatórias

à minha família

Thiago Alberto Ferreira Adnet

à minha família.

Daniel Fraianeli

Agradecimentos

Agradeço primeiramente aos meus pais que sempre me deram todo o apoio, orientação e confiança não só na faculdade, mas em toda a minha jornada até aqui. Agradeço também aos professores Geovany Araújo Borges, que me deu o suporte necessário pra realização deste projeto, e Guilherme Caribé de Carvalho, coordenador do projeto. Além disso agradeço à Rockwell Automation, que forneceu os materiais e oportunidade de realização deste projeto. Também agradeço a todos os outros professores que contribuíram para a minha formação durante o curso, sem esquecer os funcionários do SG11, SG09 e GRACO, que sempre estiveram dispostos a ajudar e ensinar. Por último, agradeço aos meus colegas que deram a ajuda e momentos de alegria que fizeram esse curso memorável.

Daniel Fraianeli

Agradeço a meus pais e a meu irmão - pessoas que me deram todo o apoio durante esse curso. Também sou grato aos professores, que me ensinaram, principalmente ao professor Geovany. E agradeço aos colegas de turma, cuja amizade e apoio foram essenciais para chegar até aqui.

Thiago Alberto Ferreira Adnet

RESUMO

Este trabalho visa à construção e desenvolvimento de uma plataforma de aprendizado destinada ao laboratório de controle pra automação. A plataforma consiste em um guindaste em escala movido por um servomotor, sendo o guindaste provido de um eletro-ímã para coleta de material ferroso simulando uma planta de fundição.

O controle do sistema é feito com o uso de um controlador Rockwell ControlLogix 5560 e a detecção da presença de material a ser coletada é feita por sensores indutivos. Há ainda a utilização de sensores fotoelétricos para fim de curso, da interface SERCOS para comunicação com o inversor de frequência Rockwell Kinetix e o uso de enconders para se determinar posição do guindaste e da haste.

ABSTRACT

This work aims at building and developing a learning platform for the automation laboratory. The platform consists of a crane driven by a servomotor, and the crane equipped with an electro-magnet for collecting ferrous material simulating a smelting plant.

The control system is made using a Rockwell ControlLogix controller 5560 and the presence of material to be collected is detected by inductive sensors. There is also the use of photoelectric sensors for limit switches, the SERCOS interface for communication with the variable frequency drive Rockwell Kinetix and the use of incremental encoders to determine position of the crane and the stem.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	CONTEXTUALIZAÇÃO	1
1.2	DEFINIÇÃO DO PROBLEMA	2
1.3	OBJETIVOS DO PROJETO	3
1.4	RESULTADOS OBTIDOS	3
1.5	APRESENTAÇÃO DO MANUSCRITO	4
2	FUNDAMENTAÇÃO TÉCNICA	5
2.1	INTRODUÇÃO	5
2.2	CONTROLE	5
2.2.1	DISPOSITIVOS DE CONTROLE INDUSTRIAL	5
2.3	SENSORES	9
2.3.1	SENSORES INDUTIVOS	9
2.3.2	SENSORES FOTOELÉTRICOS	10
2.3.3	ENCODERS	13
2.4	ACIONAMENTO	17
2.4.1	MOTOR DE INDUÇÃO	17
2.4.2	KINETIX	20
2.5	COMUNICAÇÃO	24
2.5.1	DEVICEĀNET	24
2.5.2	SERCOS	27
3	DESENVOLVIMENTO E RESULTADOS EXPERIMENTAIS	33
3.1	INTRODUÇÃO	33
3.2	CONCEPÇÃO DO SISTEMA	34
3.2.1	DESCRIÇÃO DA PLANTA DE FUNDIÇÃO	34
3.2.2	ARQUITETURA DO CONTROLE	37
3.3	MONTAGEM MECÂNICA	40
3.3.1	ESTRUTURA	40
3.3.2	SUPORTE DO MOTOR	43
3.3.3	ACOPLADOR	44
3.4	MONTAGEM ELÉTRICA	44
3.5	PROGRAMAÇÃO	47

4	CONCLUSÕES	54
	REFERÊNCIAS BIBLIOGRÁFICAS	56
	ANEXOS	57
I	DIAGRAMAS ESQUEMÁTICOS	58
I.1	MATERIAIS DO PROJETO	58
I.1.1	MATERIAIS FALTANTES	59
I.2	ESQUEMÁTICOS DE LIGAÇÃO	59
I.2.1	LIGAÇÕES KINETIX	59
I.2.2	IDENTIFICAÇÃO DA PINAGEM E DOS CABOS	60
II	DESCRIÇÃO DO CONTEÚDO DO CD	63

LISTA DE FIGURAS

1.1	Planta de Fundição da Romi em Santa Bárbara D'Oeste	2
1.2	Projeto da planta de simulação	3
2.1	ControlLogix, apenas o módulo de controle.....	6
2.2	ControlLogix ligado num rack completo com diferentes módulos de expansão	7
2.3	Softwares de programação do ControlLogix.....	8
2.4	Esquemático da construção de um sensor indutivo	9
2.5	Vários exemplos de sensores indutivos	10
2.6	Esquemático de um sensor de fecho transmitido em operação	11
2.7	Características do emissor e receptor	12
2.8	Sensor 42 CF e sua ligação	13
2.9	Encoder incremental de baixa resolução e sinais de saída correspondentes aos dois canais	14
2.10	Esquemático de um circuito de avaliação de direção, utilizando um flip flop D ligado à saída dos canais do encoder	14
2.11	Disco de encoder absoluto em código de gray e circuito de conversão	16
2.12	Esquemático da transmissão padrão Stegmann	17
2.13	Campo magnético em um motor de indução	18
2.14	Analogia para motor de indução.....	20
2.15	Motor de indução do experimento e outro motor desmontado	20
2.16	Kinetix com 1 IAM e 3 AMs	22
2.17	Diagrama de um circuito variador de frequência	22
2.18	Exemplo de saída de uma circuito variador de frequência	23
2.19	RSNetWorx	24
2.20	Esquemático do barramento DeviceNet	25
2.21	Esquemático do conector DeviceNet	25
2.22	Exemplo de conector FSMA, utilizado na interface SERCOS.....	29
2.23	Exemplo de rede SERCOS com 2 anéis	30
3.1	Detalhes do sistema	34
3.2	Diagrama de blocos do sistema	34
3.3	Rede de Petri - Resumo do sistema.	36
3.4	Rede de Petri - Coleta de uma peça.	36
3.5	Rede de Petri para definição de referências de posição.....	37
3.6	Arquitetura projetada.	38

3.7	Arquitetura do sistema.....	38
3.8	Desenho da vista frontal do Kinetix 6000.	39
3.9	Guia linear	40
3.10	Projeto do suporte	41
3.11	Projeto em SolidWorks.....	41
3.12	Cantoneira de fixação.....	42
3.13	Vista frontal da estrutura.....	42
3.14	Estrutura no Laboratório de Automação.....	42
3.15	Motor e suporte.	43
3.16	Projeto do suporte.....	43
3.17	Desenho do acoplador motor-guia.	44
3.18	RSLinx.	45
3.19	Driver do RSLinx.....	45
3.20	Esquema do barramento DeviceNet.....	46
3.21	Ligações Kinetix-LIM.....	46
3.22	Conector da entrada I/O do Kinetix.....	47
3.23	RSNetWorx - rede com sensores indutivos.	48
3.24	Modificando o endereço do sensor.	48
3.25	RSNetWorx - Input.	49
3.26	Download to Device.....	50
3.27	RSLinx com a rede DeviceNet configurada.....	50
3.28	RSLogix - Path.....	51
3.29	Scan List - DeviceNet.....	51
3.30	Numeração para priorização.	52
3.31	Ladder para os sensores indutivos.	52
3.32	Variáveis do ladder - 'Program Tags'.	53
I.1	Esquemático de ligação do Kinetix com o LIM	59
I.2	Ligação das fibras ópticas entre o Kinetix e o controlador	60
I.3	Exemplo de um Low Profile Conector (Auxiliary Feedback)	61

LISTA DE TABELAS

2.1	Código de gray de 3 bits.	16
2.2	Distâncias máximas do tronco DeviceNet	26
2.3	Exemplos de configurações SERCOS	30
I.1	Materiais da remessa original.....	58
I.2	Materiais que não foram enviados na remessa original. Encomendados	59
I.3	Pinos da entrada I/O do Kinetix.	61
I.4	Pinos da entrada Auxiliary Feedback do Kinetix. (todos os fios estão no cabo 845-CA-C-10 a chegar)	61
I.5	Pinos da entrada Motor Feedback do Kinetix (todos os fios estão no cabo 2090-XXNFMP-S09).....	62

LISTA DE SÍMBOLOS

Símbolos Latinos

N	velocidade	[rpm]
f	frequência	[Hz]
P	Número de Pólos	[#]
s	escorregamento	[%]
V	Tensão	[V]

Subscritos

s	síncrona
r	rotor
i	entrada
o	saída

Siglas

AM	Axis Module
AT	Amplifier Telegram
AWG	American Wire Gauge
CAN	Controller Area Network
CIP	Common Industrial Protocol
CLP	Controlador Lógico Programável
DO	Dark Operate
HCS	Hard Clad Silica
IAM	Integrated Axis Module
IDN	Identification Number
LIM	Line Interface Module
LO	Light Operate
MDT	Master Data Telegram
MST	Master Synchronization Telegram
OSI	Open Systems Interface
PAC	Programmable Automation Controller
POF	Plastic Optic Fiber
PWM	Pulse Width Modulation
RTR	Remote Transmission Request

Capítulo 1

Introdução

Nos últimos anos a preocupação com a segurança e saúde dos funcionários em indústrias passou a ocupar uma posição de destaque na lista de prioridades de empresas de manufatura. Indústrias de fundição têm ambientes naturalmente hostis à presença humana e a diminuição da necessidade da presença de operários no local dos fornos se faz necessária.

1.1 Contextualização

Fundição é o processo de fabricação de peças onde um material derretido é colocado pra solidificação em moldes que darão a forma da peça ao final do processo. A fundição é um dos processos mais comuns na produção de peças, pois é comumente utilizada como etapa inicial na fabricação de peças que serão usinadas ou forjadas, ou mesmo no preparo do material bruto a ser levado pra indústrias. De fato, cerca de 90% das peças fabricadas envolveram fundição em alguma etapa da fabricação e cerca de 200 milhões de toneladas de peças fundidas são produzidas anualmente[1]. O exemplo mais comum de peça fundida é o motor do carro, onde o bloco, a biela, acopladores e outras peças são todas forjadas.

A indústria de fundição costuma ser um ambiente hostil à presença humana. O ferro, por exemplo, possui ponto de fusão de 1538 °C em estado puro. Naturalmente, ambientes e materiais que atinjam temperaturas desse nível, mesmo que isolados, não são apropriados pra presença e manipulação direta humana. A fundição ainda pode envolver a manipulação contínua de grande quantidade de cargas que podem ter tamanho e peso colossais, tornando o trabalho ainda mais perigoso e menos convidativo a presença humana. Assim, a automação de plantas de fundição tem sido procurada de modo a evitar os perigos do ambiente industrial e ainda permitir um aumento de produtividade associado ao trabalho ininterrupto de máquinas automatizadas.

1.2 Definição do problema

Este trabalho busca, por meio de uma grua em escala, simular o processo de automatização do processo de coleta de material a ser levado aos fornos de fundição. Uma grua equipada com eletroímã e com movimento axial, deverá buscar material ferroso sempre que sua presença for detectada por sensores indutivos. O movimento axial é feito com o uso de um motor de indução trifásico acionado por um inversor de frequência e é usado pra se levar o carro que contém o guindaste até a posição da peça a ser coletada. Toda a integração e automação da planta é feita utilizando-se um PAC (*Programmable Automation Controller*), i.e., um controlador destinado a automação, que agrega as funções lógicas de um CLP às funções de processamento capazes de implementar um controlador PID, ou de se comunicar com outros equipamentos através do uso de protocolos industriais. Um exemplo de situação que se deseja simular se encontra na figura 1.1 ¹, com a diferença que a coleta de material é feita por um gancho.



Figura 1.1: Planta de Fundição da Romi em Santa Bárbara D'Oeste

O esquemático do que será construído pra simulação de processos semelhantes ao da figura 1.1 é apresentado na figura 1.2².

¹fonte: http://www.romi.com.br/fu_noticias.0.html?&L=0

²adaptado de: proposta inicial do projeto

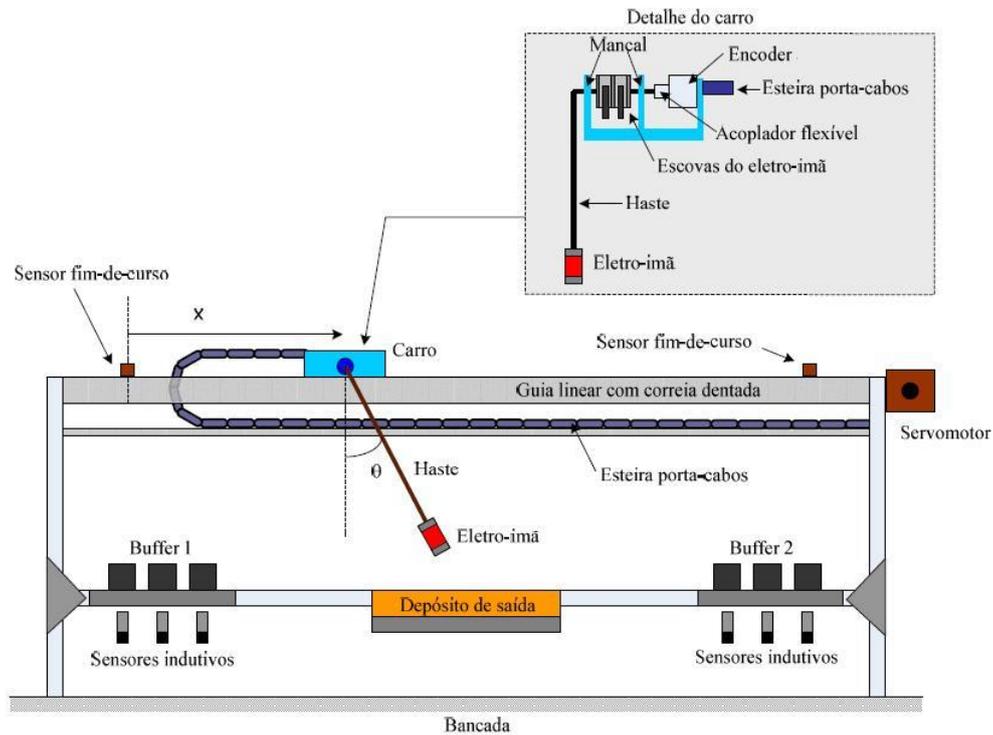


Figura 1.2: Projeto da planta de simulação

1.3 Objetivos do projeto

O objetivo deste projeto é construir um planta de simulação de um ambiente de uma indústria de fundição, incluindo num primeiro momento a construção da estrutura e ligação elétrica, para depois ocorrer a ligação de sensores e redes de comunicação e por último, o projeto de um controle de posição para a grua.

O projeto se destina a ser um experimento do laboratório de automação da Universidade de Brasília, servindo para ensino de CLPs, protocolos de comunicação industrial, acionamento de motores entre outras funções.

1.4 Resultados Obtidos

Da parte de montagem mecânica, o suporte da grua foi construído com sucesso e a peça de suporte do motor a ser colocada na estrutura se encontra em fabricação. Devido a necessidade de primeiro se fazer uma avaliação de como a haste do guindaste vai ficar, os sensores indutivos não foram presos porque a distância que eles devem ficar não é conhecida. O quadro elétrico não foi construído devido à falta da fonte específica do variador de frequência Kinetix e a possibilidade de o local do quadro ser repensado devido ao tamanho do novo componente. Também não foi avaliado qual o tamanho das fontes externas seriam necessárias. O acoplador para ligar a guia ao motor foi adquirido e se encontra ligado à guia. O acoplador pra ligar o encoder à haste foi encomendado e não chegou.

Da parte de montagem elétrica, a rede DeviceNet foi ligada com sucesso. Até o momento, somente os 6 sensores indutivos se encontram conectados, mas a rede suporta até 57 novos equipamentos DeviceNet mediante a obtenção de mais cabos e conectores (descritos no capítulo 2.5.1). A tomada trifásica foi instalada, mas como a fonte específica do Kinetix (*Line Interface Module*, que inclui fusíveis, filtros, transformador para outras ligações que não a ligação direta trifásica, e outras funcionalidades descritas no capítulo 3.4) e o Power Rail (descrito no capítulo 2.4.2 que é essencial para o funcionamento do Kinetix não chegaram, optou-se por manter os equipamentos em suas caixas para proteção, já que eles não podem funcionar. Os 2 pares de sensores elétricos são destinados a serem ligados à entrada específica do Kinetix, e como o último não está em condições de operar, os sensores não foram ligados. Por último, o encoder externo também não foi ligado uma vez que não só o Kinetix não está em condições de operação mas também o cabo específico pra ele não foi enviado, embora já tenha sido encomendado.

Diante dos empecilhos relacionados no parágrafo anterior, a programação de funcionalidades envolvendo o Kinetix não pode ser testada e desenvolvida. Uma programação para a rede DeviceNet desenvolvida, de modo que os sensores sejam lidos para a verificação da presença de material na posição correta. Depois, os sensores com material na posição correta são testados com base numa prioridade arbitrariamente definida e ativam ou desativam variáveis do tipo boolean que indicam qual deverá ser a posição do carro pra buscar o material na posição correspondente. Uma vez que a posição dos sensores não foi completamente definida, a variável foi mantida como boolean, mas poderá ser usada pra determinar outra variável que determina a referência exata para o PID do carro.

1.5 Apresentação do manuscrito

No capítulo 2 é feita uma revisão bibliográfica sobre o tema de estudo, incluindo seções sobre os equipamentos utilizados, explicações sobre as redes e protocolos de comunicação utilizados além de informações sobre o funcionamento dos sensores e motores. Em seguida, o capítulo descreve a metodologia empregada no desenvolvimento do projeto, detalhando a maneira como a situação a ser simulada foi transformada em planta de simulação, uma descrição da lógica de automação e outros resultados experimentais são discutidos no capítulo 3, seguido das conclusões no capítulo 4. Os anexos contêm material complementar como a lista de materiais e a conexões elétricas faltantes.

Capítulo 2

Fundamentação Técnica

2.1 Introdução

A Revisão Bibliográfica foi dividida de acordo com elementos de uma planta de controle. Primeiro se discute a central de processamento e controle, que consiste no ControlLogix 5560, um PAC (*Programmable Automation Controller*) de fabricação Rockwell Automation. Essa unidade corresponde ao Controlador, uma vez que além do comando o ControlLogix também é responsável pelo cálculo do sinal de controle. Em seguida os sensores são discutidos. Há sensores de realimentação na planta de controle (Encoder) e sensores de comando lógico (indutivos e fotoelétricos). O bloco de acionamento da planta de controle corresponde ao motor de indução e variador de frequência. O processo a ser controlado futuramente será visto no capítulo 3. Toda a comunicação do sistema, incluindo a automação lógica e a planta de controle é feita utilizando-se duas redes de comunicação industrial, a SERCOS interface e a rede DeviceNet, cujas especificações também são descritas nessa seção.

2.2 Controle

2.2.1 Dispositivos de Controle Industrial

CLP (Controlador Lógico Programável) é um computador digital empregado largamente em processos de automação de sistemas eletromecânicos, desde controle de iluminação, passando por controle de temperatura de ambientes chegando à automação de grandes plantas industriais. Embora alguns autores prefiram diferenciar entre CLPs propriamente ditos como sendo aqueles que possuem apenas entradas e saídas binárias, chamando os CLPs incrementados com funções analógicas de PAC (Programmable Automation Controller), é comum tratar todos esses dispositivos pela sigla CLP. Neste texto, qualquer referência a CLP ou PAC, se refere ao ControlLogix, um modelo de PAC da Rockwell Automation mostrado na figura 2.1 ¹.

¹fonte: <http://www.ab.com/programmablecontrol/pac/>



Figura 2.1: ControlLogix, apenas o módulo de controle

Diferentemente de computadores comuns, os CLPs possuem grande parte de seu processamento executado em paralelo com outras funções, em especial, a atualização dos valores nas suas entradas. Além disso, os CLPs costumam possuir um grande número de entradas e saídas tanto analógicas como digitais, de modo que se possa acessar muitos pontos e variáveis diretamente. Ainda, esses equipamentos em geral são projetados pra operar em ambientes normalmente hostis a computadores. Os CLPs são resistentes ao calor, a interferências eletromagnéticas, a vibrações, e são feitos pra não produzirem faíscas (para ambientes com vapores inflamáveis).

Os CLPs possuem projeto modular expansível. Isso significa que eles são flexíveis para atender às necessidades da aplicação e mantêm processamento paralelo, permitindo facilidade de manutenção, expansão e remodelamento. Esses módulos costumam ser encaixados em racks construídos especialmente para o modelo a ser utilizados. O rack tem um circuito que contem trilhos para alimentação e comunicação dos módulos, além de slots para o encaixe do módulo. Somente um módulo de controle é utilizado por rack, e os outros módulos mais comuns são os de entrada e saída em diferentes configurações (digitais, analógicas ou mistura das duas opções), módulos de protocolos de redes industriais (Industrial Ethernet, DeviceNet, ControlNet, SERCOS...) e alguns módulos pra aplicações específicas. O módulo de controle não é responsável pelo comando de nenhum desses módulos, sendo responsável apenas por buscar as informações já preparadas em posições específicas da memória e pela lógica envolvendo essas variáveis. Os módulos sempre possuem um processamento independente de modo que possam fazer suas funções como fazer leituras de pontos ou comandar a rede sem a necessidade do módulo de controle. A figura 2.2 mostra um rack com diversos módulos ².

²fonte: <http://www.ab.com/programmablecontrol/pac/>



Figura 2.2: ControlLogix ligado num rack completo com diferentes módulos de expansão

Os CLPs podem ser programados em 5 diferentes tipos de linguagem: FBD (*Function Block Diagram*), ST (*Structured Text*, igual ou aproximadamente igual a linguagens como C ou Pascal), IL (*Instruction List*, similar a Assembly), SFC (*Sequential Function Chart*, semelhante a redes de petri) e LD (*Ladder Logic*, algumas vezes conhecido como lógica relé). Essas categorias de linguagem foram definidas como padrão para CLPs, mas em geral CLPs mais complexos modificam essas linguagens de modo a se obter instruções próprias e específicas.

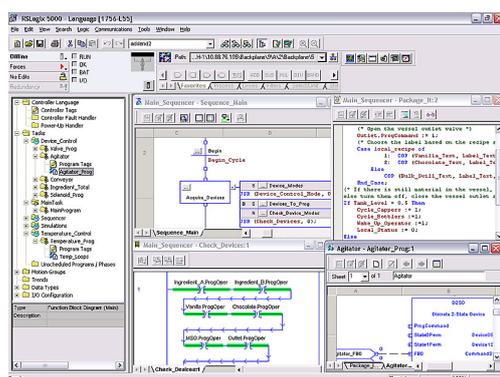
ControlLogix é a família de CLPs mais complexa da Rockwell Automation. Com memórias de programa indo até 8 Mb e integração fina entre o controlador e seus módulos de comunicação e I/O, o ControlLogix é capaz de processar qualquer situação em indústrias [2].

Além das funções comuns a praticamente todos CLPs descritas até agora, o ControlLogix fornece funcionalidades próprias. O controlador pode ser programado em 3 diferentes linguagens: SFC, Ladder e ST. Em qualquer dos casos, a arquitetura disponibiliza funções próprias e específicas que podem ser incorporadas em qualquer das linguagens. Essas instruções vão de simples timers, passando por blocos de controle PID e fuzzy, chegando a instruções específicas para a família de variadores de frequência Kinetix, para os controladores integrados com o módulo de interface SERCOS. A lista de funções é tão extensa que não será discutida aqui, manuais estão disponíveis gratuitamente em www.ab.com. O controlador também permite a definição de rotinas próprias, que podem ser usadas como blocos dentro de qualquer programa, seja o principal, sejam programas executados automaticamente em ocasiões especiais como o desligamento ou ocorrência de alarmes pré-definidos.

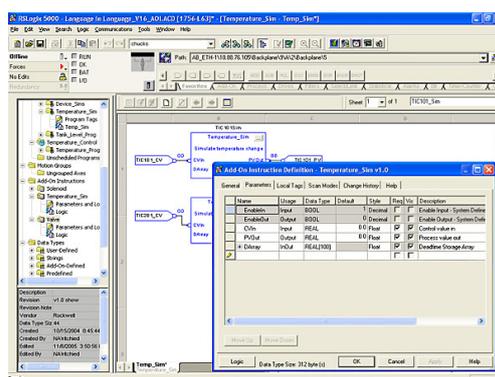
O controlador inclui ainda outras características e vantagens específicas. É possível se utilizar um segundo controlador como backup, sendo que CLP de backup começará a atuar assim que houver a falha no primário, e terá todas as suas variáveis atualizadas como se estivesse em funcionamento pleno. Ainda sobre segurança, o controlador e seus módulo testam as suas entradas e saídas constantemente, de modo a verificar mau funcionamento, gerar avisos ou rotinas excepcionais. O ControlLogix possui ainda módulos de precisão temporal de qualidade, que provêm

varreduras em até 100 ms, alta repetibilidade de tempo e sincronia temporal entre controladores via GPS.

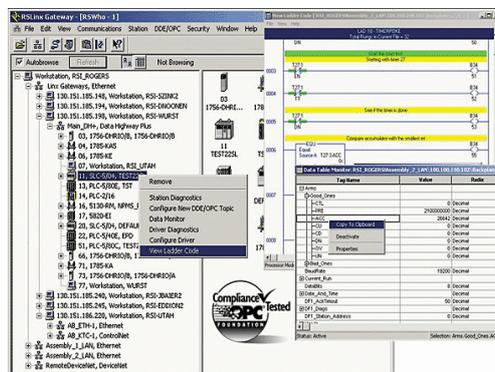
Os softwares responsáveis pela criação e manutenção de um sistema com o ControlLogix são o RSLogix5000 e o RSLinx. O RSLogix desenvolve programas simbólicos em qualquer das linguagens aceitas pelo ControlLogix com a possibilidade de criação de variáveis de diferentes tipos, incluindo arrays e estruturas (*structs*). É nesse programa que se acessam as funções específicas da plataforma ControlLogix e que são desenvolvidas funções próprias pra reutilização nos programas. O software permite ainda a visualização das variáveis (*tags*) concomitantemente com a leitura pelo controlador. Com isso não só o processo de debug é simplificado, mas também é possível a correção de falhas com mudanças sem o desligamento do controlador e download apenas do programa alterado. O RSLogix é o programa que gerencia as redes de comunicação diretas com o controlador. Isso significa que os módulos do rack com o controlador são visualizados e declarados para o controlador aqui. Sem o RSLinx, o controlador não é capaz de saber com quem "conversa". O RSLinx também permite a integração com outros programas, como o Factory Talk para terminais de acesso remoto para operadores do sistema e o RSNetWorx para configuração de redes entre módulos e sensores. Outros programas podem ser usados como o RSEmulate para simulação de programas, mas estes programas não são necessários. O RSNetWorx (que é o terceiro e último programa essencial) será introduzido no capítulo 2.5.1. Informações e tutoriais também são disponibilizados no site www.ab.com.³



(a) RSLogix: programação em diferentes linguagens



(b) RSLogix: definição de funções próprias



(c) RSLinx

Figura 2.3: Softwares de programação do ControlLogix

³fonte: <http://www.rockwellautomation.com/rockwellsoftware/products/>

2.3 Sensores

Sensores de presença são os sensores utilizados pra detectar a existência de objetos próximos ao sensor. Em geral, sensores de presença podem medir pequenas distâncias além da capacidade de apenas detectar objetos, e podem ser adaptados pra medições de nível de líquido, de contagem de peças, vibração, e outras funções. Os sensores de presença podem ser divididos em duas categorias, os sensores que necessitam contato físico e os que não necessitam.

Sensores de presença sem a necessidade de contato possuem diversas vantagens sobre os sensores que requerem o toque pra detecção de presença de materiais. Os sensores sem contato podem detectar a presença de materiais frágeis sem a possibilidade de danificá-los, pois sem o toque não há risco de um impacto. Essa característica também protege o próprio sensor de impactos ou desgaste. Os sensores sem contato ainda são praticamente livres de histerese além de boa resposta dinâmica[3]. Os sensores sem contato mais comuns são os fotoelétricos, indutivos e capacitivos.

2.3.1 Sensores Indutivos

Usados para detectar objetos metálicos e não metálicos, o sensor indutivo trabalha pelo princípio da indução eletromagnética. Há mais de um tipo de sensor indutivo, mas o tipo mais comum consiste em duas bobinas, sendo uma delas ligada a uma alimentação elétrica de corrente alternada. Com a passagem de corrente, haverá a indução de um campo magnético que por sua vez induzirá uma corrente elétrica na segunda bobina. O processo se repete e a segunda bobina também produzirá um campo magnético que será o campo efetivo do sensor, de maneira semelhante à figura 2.4 [3].

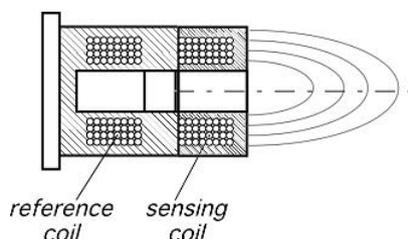


Figura 2.4: Esquemático da construção de um sensor indutivo

Uma vez que haja um objeto imerso no campo magnético gerado pelo sensor a relutância magnética se alterará devido a presença de correntes geradas no objeto e o efeito será um desbalanceamento entre as bobinas do sensor, indicando que há a presença de material[4]. Para se utilizar o sensor como medidor de distâncias, é necessário se equacionar a quantidade de alteração na relutância magnética e assim calcular a distância a partir de outros dados. Vários exemplos de sensores podem ser vistos na figura 2.5 ⁴.

⁴fonte: <http://www.automationdirect.com.au/Welcome/>



Figura 2.5: Vários exemplos de sensores indutivos

Outras considerações são necessárias. Embora a histerese desses sensores seja naturalmente pequena, às vezes é necessário se aumentar essa histerese propositadamente. Isso é útil em situação onde um objeto pode estar no limiar de detecção do sensor e com uma vibração suficiente para que o objeto entre e saia do limiar continuamente. Nesse caso, a histerese impede que o objeto seja detectado muitas vezes, quando na verdade há apenas um objeto. Outra consideração é a presença ou não de blindagem. A blindagem não afeta o funcionamento do sensor, afeta apenas a concentração do campo. Com a blindagem, o campo magnético fica direcionado o que reduz a área que o sensor pode ser usado como detector. Por outro lado, evita que suportes ou outras peças indesejáveis sejam detectadas. Para se aumentar a distância que o sensor é útil, é necessário se aumentar a distância que o campo magnético atinge e isso só pode ser feito aumentando-se o diâmetro das bobinas do sensor, o que naturalmente impede que sensores indutivos sejam usados para grandes distâncias.

O sensor indutivo utilizado no experimento é um sensor inteligente, ou seja, ele tem um processamento local das informações (o que é importante, uma vez que ele transmite informações no protocolo DeviceNet). Embora o sensor possa operar em modo de medição de distâncias, a sua aplicação se resume a detecção de presença. A sua resposta digital inclui um bit para detecção de presença, um bit para indicar se o material está muito próximo do sensor e um bit que indica se o material está muito longe. Esses bits serão usados para indicar se o material está presente e na posição correta. O sensor possui ainda bits de auto-diagnóstico para evitar leituras provenientes de um sensor defeituoso.

2.3.2 Sensores Fotoelétricos

Sensores fotoelétricos são sensores que se baseiam no princípio da emissão de elétrons quando submetidos à radiação eletromagnética de ondas de alta frequência, como a luz. Estes sensores têm variadas aplicações como medir a luminosidade ambiente, detectar cores de objetos, ou mesmo como sensores de presença (aplicação deste projeto).

Para o uso como detectores de presença, estes sensores precisam do material sensível à luz e de uma fonte de luz permanente. Basicamente, uma vez que fonte de luz é bloqueada, o sensor acusa a presença de alguém ou algo na sua frente. Os sensores fotoelétricos de presença podem ser de 3 tipos, difusão, reflexão e fecho transmitido.

O modo fecho transmitido é o modo onde o emissor de luz e o receptor se encontram em compartimentos separados. As duas unidades são posicionadas em frente uma à outra, de modo que receptor fique constantemente recebendo o fecho emitido pela outra unidade, o emissor. Assim, um objeto que bloqueie esse fecho de luz, fará com que o receptor mude estado indicando que há a presença de algo. Esses sensores são semelhantes aos sensores de intrusão presente em sistemas de segurança de bancos, museus e outros edifícios. A figura 2.6 [5] mostra um esquemático do sensor de fecho transmitido.



Figura 2.6: Esquemático de um sensor de fecho transmitido em operação

As principais vantagens desse tipo de sensor são a distância máxima de medição e a grande resistência a poeira e outros contaminantes, diferentemente dos sensores por reflexão que dependem de um trajeto de ida e volta (o que dobra a distância que o fecho percorre) e também de poucas perdas durante a reflexão (o que requer limpeza constante). As distâncias efetivas desse tipo de sensor podem chegar a mais de 200m e alguns sensores podem continuar a operar satisfatoriamente mesmo que 99,67% da soma das áreas do emissor e receptor esteja coberta por poeira se estiver a 3m ou menos um do outro [5]. Outra vantagem é a capacidade de se detectar objetos com grande reflexão sem problemas, uma vez que o receptor estará atrás do objeto. Por outro lado, esses sensores não podem ser usados pra detectar objetos transparentes. E ainda, precisam de um bom alinhamento entre o receptor e o emissor, ou não funcionarão corretamente.

Em geral, são utilizados LEDs no emissor, já que esses dispositivos são mais confiáveis e de menor tamanho que lâmpadas. Os LEDs podem ser de praticamente qualquer cor, desde que a onda tenha frequência suficiente para causar o efeito fotoelétrico. No entanto, ondas de mais baixa frequência são preferidas pois consomem menos energia e assim o infravermelho (< 400 THz) é a frequência mais comum, seguido do LED de cor vermelha (> 400 THz e < 480 THz) em aplicações onde a visibilidade do fecho é importante. Os LEDs, como qualquer outra fonte de luz, emitem luz em todas as direções o que dispersa a intensidade, diminuindo o alcance e podendo interferir em sensores em outros lugares. Por esse motivo, os sensores incluem lentes que dirigem o fecho na direção que importa. Uma vez que a energia no fecho fica toda em uma única direção, o fecho de luz consegue chegar a maiores distâncias e podem ser usados pra detectar objetos menores. Fechos de área muito grande não serão efetivamente bloqueados por objetos pequenos. Por outro lado, o concentrar o fecho diminui a área em que o sensor é efetivo e dificulta o alinhamento do sensor e receptor. É importante lembrar que o que importa no receptor é energia contida no fecho, e não a área do mesmo. A figura 2.7(a)[5] mostra o efeito das lentes. O emissor precisa ainda de modular a sua luz emitida. Isso é feito por duas razões. A primeira, é que a transmissão contínua de fechos de grande intensidade requerem grande corrente o que gera muito calor, então uma transmissão intermitente (da ordem de 5 KHz) é usada pra diminuir o calor gerado. Segundo, por motivos de segurança contra interferência externas como outras fonte de luz ou a presença de reflexos, os

emissores emitem um padrão de luz próprio pra que o receptor reconheça se a fonte de luz é do emissor ou outra qualquer [5]. Em sistemas de segurança o padrão de modulação é complexo pra evitar que pessoas imitem a fonte e "enganem" os sensores, mas em indústrias apenas a transmissão intermitente e a sincronia do receptor costumam ser suficientes. O emissor transmite um pulso a cada 0,2 ms conforme mostrado na figura 2.7(b)[5]. O receptor irá responder somente a esse padrão de transmissão, ou de outro modo outra fonte de luz poderia ativar o sensor indevidamente.

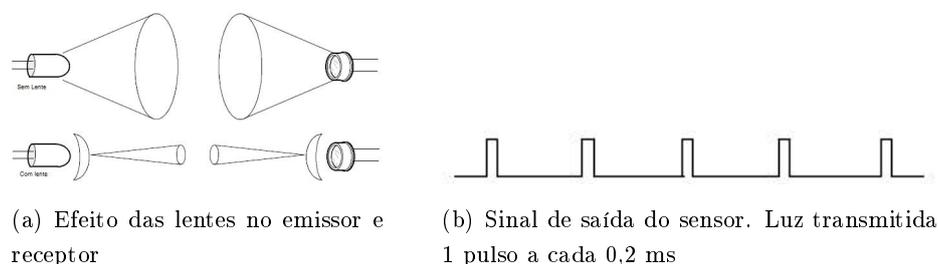


Figura 2.7: Características do emissor e receptor

Os receptores são constituídos de um fotodiodo e um circuito auxiliar pra indicação da medição. O fotodiodo opera como um diodo comum, mas o material do qual ele é feito é sensível ao efeito fotoelétrico. Isso significa que quando os fótons atingem o diodo, elétrons são excitados dando origem a portadores de carga móveis (o elétron e a lacuna) que darão origem a uma corrente que independe da fonte de tensão, ou que afeta características como o efeito de avalanche. O sensor 42CF da Rockwell Automation que é utilizado no experimento utiliza a operação foto condutiva de corrente. Quando o diodo é diretamente polarizado, o aumento de corrente devido ao efeito foto elétrico não é significativo. No caso da polarização reversa, a corrente da fonte assume valores mínimos (da ordem de $50 \mu\text{A}$ para 5V), mas a corrente causada pelo efeito fotoelétrico se torna bem maior (da ordem de $500 \mu\text{A}$ para 5V e 3000 lux) [6]. Em geral, um circuito adicional é utilizado pra converter essa corrente em alguma saída útil a PLCs. No caso do experimento, o sensor tem saída PNP, e embora o circuito interno do sensor não seja disponível em datasheets, a saída se comporta de forma esperada.

Os sensores da família 42CF podem ser de difusão ou fecho transmitido e podem ser PNP ou NPN. No caso dos sensores do experimento, todos são NPN de fecho transmitido. O sensor funciona como um transistor npn no modo chaveamento, e uma vez que a presença de material é detectada, o sensor chaveia a saída para da alimentação para a referência. O sensor possui ainda a possibilidade de chavear a saída quando o receptor recebe não mais recebe luz (*dark operate*), ou quando a recepção é restabelecida (*light operate*). A figura 2.8(b) mostra o esquema de ligação do sensor. O símbolo Load, corresponde ao Kinetix 6000 que possui uma conexão para entrada e saída que inclui fontes 24VDC. Quando o sensor acusa a presença de um objeto, o fio preto assumirá a tensão da referência fazendo com que a carga (load, no caso do experimento, se trata do ControlLogix indiretamente através do Kinetix 6000) tenha uma diferença de potencial e assim identifique o sinal do sensor. A figura 2.8[5] mostra o sensor utilizado no experimento (emissor e receptor iguais) e a sua fiação.

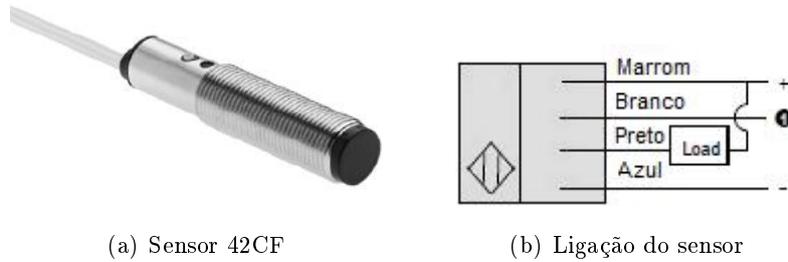


Figura 2.8: Sensor 42 CF e sua ligação

2.3.3 Encoders

Encoders são dispositivos utilizados na transformação de movimento linear ou angular em um sinal elétrico que possa ser usado pra inferir dados de tal movimento. Embora não sejam os únicos equipamentos que possam realizar essa tarefa, a facilidade de interfaceamento com o circuito de processamento tornou este tipo de sensor muito comum em movimentação de eixos de máquinas, robôs articulados e outras aplicações não necessariamente ligadas à indústria.

Encoders são construídos baseados em diferentes princípios (magnetismo, toque, luz), mas todos consistem em uma peça acoplada ao movimento com marcações compatíveis com o princípio no qual são baseados, e uma peça que fará a medição dessas marcas. Os dois encoders utilizados no projeto são ópticos (de marcações geométricas), e por isso qualquer referência a encoder se refere a este modelo e apenas ele será descrito aqui (embora os encoders baseados em outros princípios sejam muito semelhantes). Independentemente do princípio no qual se baseiam, os encoders podem ser lineares ou angulares, sendo esses últimos, os modelos empregados neste trabalho. Naturalmente, os encoders angulares podem ser usados para medir movimentos lineares se o seu movimento for atrelado ao movimento a ser medido. Essa característica é comumente utilizada em situações onde um motor move uma esteira, ou uma roda se move no plano.

Encoders ópticos angulares são construídos com um disco, normalmente de material transparente, acoplados ao eixo de movimento. No disco são feitas marcações opacas de modo que sejam distintas do resto do disco. O posicionamento das marcações depende da aplicação e do tipo do encoder, mas em qualquer caso, a informação vai ser extraída da presença ou não dessas marcações. Um laser é posicionado transversalmente ao disco, e à medida que o eixo onde está acoplado o disco gira, o laser fará a leitura baseada no fato de a luz passar (uma parte não marcada do disco está em frente à fonte de luz) ou não (uma marca está em frente ao laser). O contrário também é possível (disco opaco, marcas transparentes).

O disco e o feixe de laser são instrumentos usados pra detectar o movimento, um circuito ou software adicional é necessário pra se extrair mais informações do movimento e depende do tipo de encoder. O posicionamento das marcas também depende do tipo de encoder, ou seja, se o encoder é incremental ou absoluto. Neste projeto, o encoder usado pra medir a posição da haste é um encoder incremental. O encoder utilizado na planta de controle como feedback de posição, é um encoder Stegmann Hyperface, um encoder de projeto proprietário que agrega características de encoder absoluto e incremental. Mais detalhamento sobre a aplicação desses encoders são fornecidas no

2.3.3.1 Encoders Incrementais

Os encoders incrementais só podem determinar a posição com base em um referencial. O referencial costuma ser uma marcação única e de pequenas dimensões numa trilha interna ao disco. Uma fonte de luz específica pra essa trilha interna é adicionada, e uma vez que a marcação é detectada, a posição pode ser determinada a partir da contagem de quantas marcações da trilha externa foram detectadas. O sinal da medição do laser é usado como sinal de relógio em um contador digital, o valor da contagem é multiplicado pela distância que uma marca significa (depende do sistema sendo medido). O sinal de referência é o reset para o relógio. A figura 2.9 [7] mostra um encoder incremental de baixa resolução (1/4 de volta e o movimento mínimo detectado) com dois canais A e B, que detectam suas próprias trilhas.

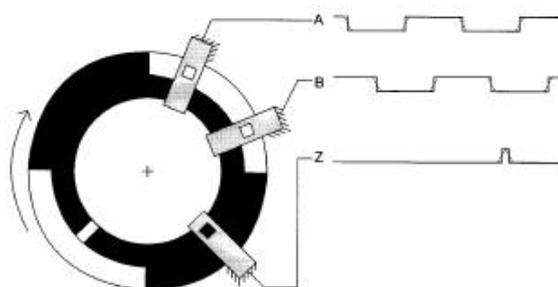


Figura 2.9: Encoder incremental de baixa resolução e sinais de saída correspondentes aos dois canais

O encoder descrito até agora é capaz de detectar movimento, e pode incrementar a posição a partir do seu próprio referencial, mas não pode detectar direção do movimento. Na verdade, se houver uma rotação pro outro lado, o circuito auxiliar ao encoder continuaria a incrementar a posição embora o correto fosse decrementar a posição. Pra solução desse problema, adiciona-se um segundo laser de medição conforme indicado na figura 2.9. Um circuito ou software faz a detecção da direção e com isso a decisão se é pra incrementar ou decrementar o contador. No caso da figura 2.9, se o sinal do laser A for acionado primeiro, o encoder está no sentido horário. Normalmente o circuito com flip flop D da figura 2.10 é usado (ou emulado em software) pra detecção de direção. Neste experimento, a detecção direção é feita automaticamente pela entrada *auxiliary feedback* do Kinetix (conforme descrição no capítulo 3, então não será necessário circuito ou programação adicional uma vez que a função já está implementada).

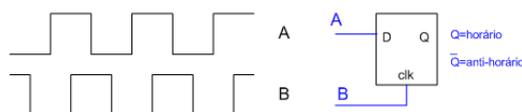


Figura 2.10: Esquemático de um circuito de avaliação de direção, utilizando um flip flop D ligado à saída dos canais do encoder

A resolução do encoder é determinada pela densidade das marcas das trilhas externas. Se o disco

for metade opaco, metade translúcido, o movimento somente será detectado quando meia volta do encoder for completada, uma vez que apenas nessa situação o laser terá uma nova mudança de estado. Quanto mais marcas alternadas, menor será o movimento necessário do disco pra que haja uma mudança no laser seja detectada. A resolução não afeta a acurácia, ou seja, a proximidade do valor medido com o real não é alterada com o número de marcas. A acurácia do sensor é determinada pela focalização do laser, pela qualidade da marcação (igualdade na distância e tamanho das marcas, diferença entre zona opaca e transparente), resistência à vibração entre outras.

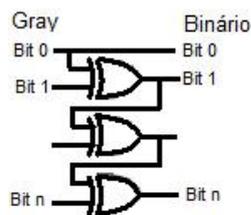
2.3.3.2 Encoders Absolutos

Ao contrário dos encoders incrementais que fazem uma contagem de marcas pra saber a posição, os encoders absolutos determinam a posição diretamente a partir da marcação sendo lida no momento. Cada marcação é posicionada de modo que a combinação das trilhas signifique um número constituído de bits, seja esse número em codificação binária, em código de gray ou qualquer codificação utilizada. Os encoders absolutos em geral utilizam uma leitura paralela, de modo que a interpretação do número de seja simplificada, o que significa que cada trilha do disco corresponde a um bit do número a ser lido. O resultado disso é a necessidade de quantos lasers quantos forem o número de bits do encoder, ao invés de apenas 3 como no encoder incremental, o que torna os encoders absolutos mais caros. Por outro lado, caso haja uma perda de energia, o encoder absoluto saberá sua posição uma vez que a energia retorne, enquanto que o incremental terá que encontrar a sua referência pra então fazer um novo cálculo de posição. O encoder absoluto possui ainda menos erros acumulados. O modelo incremental acumula erros quando é utilizado, já que uma marca perdida significa um acúmulo de erro dali em diante, enquanto que uma leitura errada no modelo absoluto torna não necessariamente se repetirá na próxima leitura.

Diferentes codificações já foram propostas pra leitura do encoder absoluto. A mais imediata é a codificação binária, o que permite uma lógica de interpretação do valor lido imediata. Essa codificação por outro lado, causa um risco de um erro de leitura em um único bit significar uma grande diferença na interpretação do número. Em um encoder de 4 bits (16 posições possíveis) por exemplo, um erro no bit mais significativo pode mudar da posição 8 (1000) pra 0 (0000), o que corresponde a meia volta de diferença. Esse erro de leitura pode causar movimentação e vibração excessiva, o que pode danificar o equipamento ou pior, causar ferimentos a alguém próximo do local. Pra contorno desse problema, a codificação de gray se tornou uma das mais comuns nesse tipo de encoder. A codificação de gray é uma das variadas codificações que codificam números vizinhos com a mudança de apenas um bit, mas essa codificação se tornou popular em encoders porque é facilmente convertida em código binário (usado no processamento) e também porque suas marcações mais estreitas têm o dobro da espessura da codificação binária [7](uma vez que pela codificação formada, nunca ocorre a sequência 0-1-0-1... em qualquer dos bits do número, verifique na tabela 2.1 e na figura 2.11(a) [8]).



(a) Disco com marcações Geométricas em código de Gray



(b) Circuito de Conversão para código binário

Figura 2.11: Disco de encoder absoluto em código de gray e circuito de conversão

Tabela 2.1: Código de gray de 3 bits.

Número decimal	Em código de Gray	Em código binário
0	000	000
1	001	001
2	011	010
3	010	011
4	110	100
5	111	101
6	101	110
7	100	111

A resolução deste tipo de sensor é determinada pelo número de trilhas e consequentemente o número de bits da medição do sensor. Um sensor de 12 bits por exemplo, fornece 4096 medições pra cada volta. Assim como nos encoders incrementais, a acurácia não é diretamente proporcional à resolução, sendo ligada a construção mecânica do sensor e à sua conexão com processo sendo medido (ou seja, se não há folgas no acoplamento entre o eixo do motor e a parte que tem movimento).

Independentemente da resolução, encoders absolutos podem ser de uma única volta (*singleturn*) ou de múltiplas voltas (*multiturn*). Ambos operam de maneira semelhante, mas o encoder de múltiplas voltas possui um contador ou algum outro aparato pra contagem de quantas voltas foram completadas, e assim pode medir posições em locais além dos alcançados com uma única volta do encoder. O encoder utilizado no projeto é do tipo única volta.

2.3.3.3 Stegmann Hiperface

Stegmann Hiperface é um projeto de encoders ópticos proprietário da empresa SickStegmann, uma empresa que fabrica produtos industriais com parceria com diversas empresas do ramo, incluindo a Rockwell Automation. O encoder embutido no motor, que serve de realimentação de posição do carro do guindaste, segue esse padrão Stegmann Hiperface. O nome provém do nome da marca (Stegmann) e High PERform interFACE, o que mostra que o padrão também inclui diretivas sobre a transmissão de informações. A transmissão padrão é resumida na figura 2.12[9],

e é feita com o uso de um cabo padrão fornecido pela Rockwell Automation (descrição na tabela I.5).

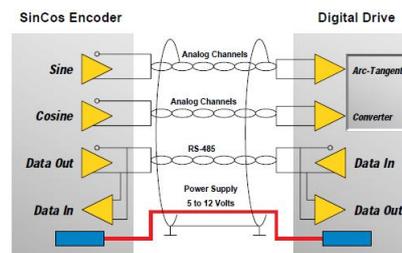


Figura 2.12: Esquemático da transmissão padrão Stegmann

O padrão define sensores com ambas as características, incremental (usado pra realimentação de velocidade) e absoluto (usado pra posição)[9]. As trilhas externas formam um código específico (porém derivado do código de gray) para o encoder absoluto, que é usado logo após a alimentação do encoder como referência, e sua informação pode ser lida novamente a qualquer momento para eliminação de erros acumulados. Os sensores de cada trilha podem ser lidos separadamente. O encoder do motor é de única volta (*single-turn*), o que significa que em uma volta, todas as posições do código da trilha são lidas, e é necessária uma solução extra no software para leituras que envolvam múltiplas voltas do encoder.

Concomitante com a leitura das trilhas com o código absoluto, as trilhas internas funcionam fornecendo a leitura em seno e cosseno, funcionando como encoder incremental. As entradas seno e cosseno podem ser transformadas em canais A e B respectivamente, funcionando conforme explicado no capítulo 2.3.3.1 sem a necessidade de um aparato externo. O uso de seno e cosseno por outro lado, facilita a transmissão de dados uma vez que é necessário menos banda pra transmissão [9], o que dá mais flexibilidade para o circuito que fará a leitura desses sinais.

2.4 Acionamento

2.4.1 Motor de Indução

O motor de indução é o tipo de motor mais comumente usado na indústria por diversas razões. A primeira, é que a sua construção simples dispensa o uso de peças complexas que requerem manutenção constante como escovas, comutadores ou anéis girantes de contato. A segunda é que esse motor pode ter sua velocidade variada, pois ela está ligada a frequência da excitação do motor. Assim, equipamentos que variem essa frequência podem acionar o motor variando a sua velocidade. Além disso, os motores de indução não requerem dupla excitação externa. Alguns motores DC e motores síncronos necessitam uma fonte extra de voltagem para excitação da armadura e do rotor (para geração do campo magnético, e da corrente que irá interagir com o mesmo). Por último, o uso em larga escala desses motores já criou um grande número de fornecedores de equipamentos e acessórios, além de um vasto conhecimento na aplicação desses itens em aplicações industriais, tornando os motores de indução os mais comuns em aplicações industriais.

Como todos os motores elétricos, os motores de indução se baseiam na interação entre o campo magnético e o campo elétrico conforme explicado pelas leis de Faraday e Lenz. Dessa maneira, utiliza-se a interação entre campos magnéticos originado pela corrente elétrica e o campo magnético induzido no rotor por esse mesmo campo magnético originado pela corrente elétrica, pra dar origem a torque e conseqüentemente a movimento. Todos os motores elétricos são constituídos de duas partes principais, o rotor que é a parte girante e a armadura, que no caso dos motores de indução, é parte que recebe a excitação elétrica. A excitação elétrica alternada na armadura irá causar o campo magnético que dará origem ao movimento.

A armadura do motor consiste em um cilindro onde são adicionados condutores com espaçamento dependendo do tipo de motor. Em motores trifásicos, os condutores são distantes 120 graus entre si. O cilindro é concêntrico com o eixo que se move, o que equivale a dizer que os condutores circundam o eixo. Esses condutores são isolados entre si e cada um deve ser ligado a uma fase, sendo que pro correto funcionamento do motor, as correntes também devem estar defasadas de 120 graus. Para ilustração de como o campo magnético que causa movimento é gerado, observe a figura 2.13[10]⁵.

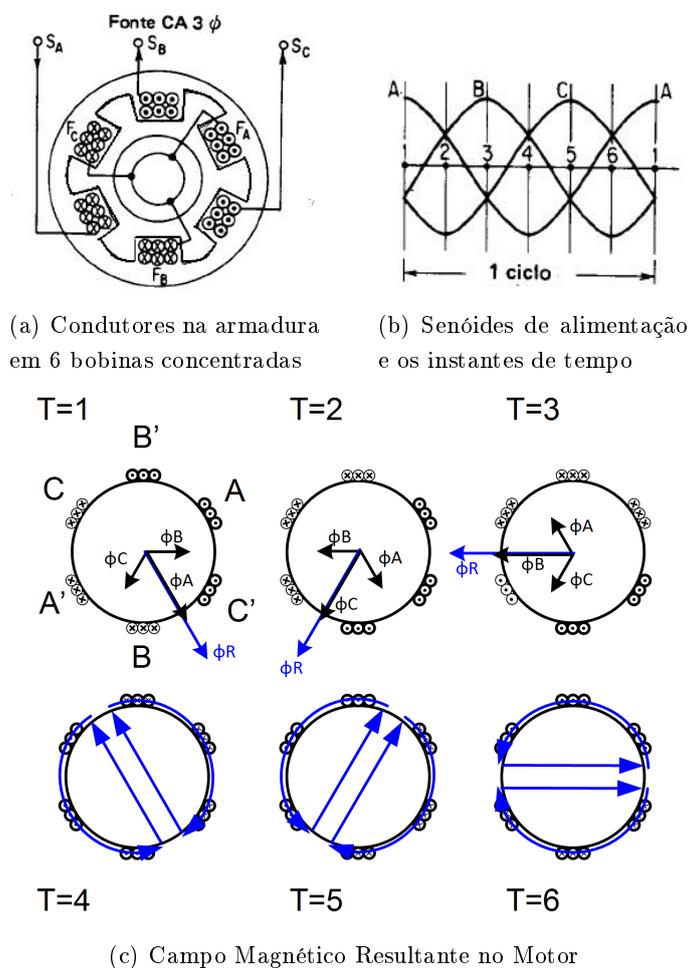


Figura 2.13: Campo magnético em um motor de indução [10]

⁵com adaptações

A figura 2.13(a) mostra a ligação das bobinas concentradas do motor. As bobinas concentradas não são usadas comercialmente pois não usam eficientemente o fluxo do entreferro da armadura, mas foram usadas porque são mais claras na observação do efeito do campo magnético. No entanto, os enrolamentos distribuídos geram um efeito semelhante no campo magnético. Cada conjunto de bobinas tem seu respectivo ponto de "entrada" S ligado a fase correspondente da alimentação trifásica e seu ponto de "saída" F ligado a um ponto comum. A figura B nos mostra o gráfico das três fases em 6 instantes de tempo e a figura c, mostra o campo magnético resultante em cada um dos 6 instantes destacados na figura b. No primeiro instante, a fase A está com valor máximo e o fluxo magnético resultante dos seus enrolamentos tem maior magnitude. As fases B e C têm o mesmo valor e pelo instante 1 da figura 2.13(c), vemos que o fluxo tem componentes na direção do fluxo de A e componentes em direções opostas que se anulam devido ao valor de corrente igual. Assim, o fluxo resultante está na direção do fluxo gerado pela fase A. No instante 2, a situação é semelhante, com a diferença que a fase C tem maior valor de corrente e conseqüentemente a resultante dos fluxos se dá na direção do fluxo gerado por C. O processo segue com a fase B atingindo o maior valor e com os fluxos de A e C se anulando em direções que não sejam as mesmas do fluxo de B. O processo se repete com o sentido oposto quando as fases atingem o valor mínimo (valor máximo em magnitude, mas no outro sentido) e assim, um ciclo é formado fazendo que o fluxo resultante seja giratório. A máquina mostrada na figura 2.13 tem somente dois pólos (o norte e o sul). Dependendo do número de bobinas usado por fase, máquinas com mais pólos podem ser construídas, mas o funcionamento segue o princípio do campo giratório mesmo assim.

É imediato observar que ao operar como gerador, a frequência de saída do motor depende da velocidade de rotação e do número de pólos. A saída do motor completará um ciclo completo, ou seja, o fluxo magnético da fase estará no valor máximo e alinhado com o resultante em um momento (e mais uma vez na direção oposto), uma vez a cada par de pólos, o que nos mostra que a frequência depende de $P/2$. O motor do experimento tem 2 pólos A frequência também dependerá da velocidade de rotação diretamente, já que uma rotação gera um ciclo para um par de pólos. Assim a frequência em Hz é proporcional ao número de rotações por minuto, dividido por 60 pra converter pra rotações por segundo. Juntando as duas fórmulas e isolando a velocidade, já que é a grandeza que queremos obter, temos a equação 2.1[10]:

$$N = \frac{120 \times f}{P} \quad (2.1)$$

O princípio de funcionamento do rotor pode ser entendido utilizando-se o exemplo descrito por Irving Kosow em seu livro Máquinas Elétricas e Transformadores. Considere uma placa de aço com um disco de alumínio ou cobre pivotado num mancal, que está preso a essa placa de aço. Sobre esse disco e placa, há um ímã permanente em U de modo que o circuito magnético será completado pela placa de aço. Quando o ímã girar, correntes parasitas serão induzidas no disco abaixo dele. De acordo com a lei de Lenz, o sentido das correntes será tal que se oponha ao movimento que deu origem às correntes, causando então o aparecimento de um pólo sul sob o pólo norte do ímã, e um pólo norte sob o pólo sul do ímã. Desse modo, haverá atração dos pólos e o disco girará no sentido do ímã. Nesse experimento, o ímã permanente corresponde à armadura que produz o campo giratório quando excitada, o disco é o rotor da máquina e a placa de aço corresponde ao restante da estrutura do motor. A figura 2.14[10] mostra a analogia com o motor. O rotor mais

comum em motores de indução é o tipo gaiola de esquilo, que consiste em dois anéis que são ligados por barras de cobre ou alumínio.

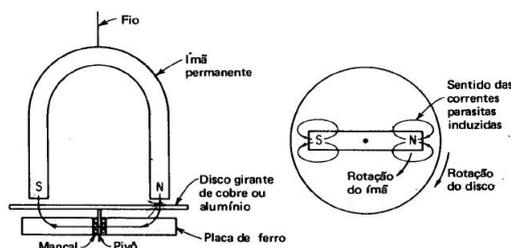
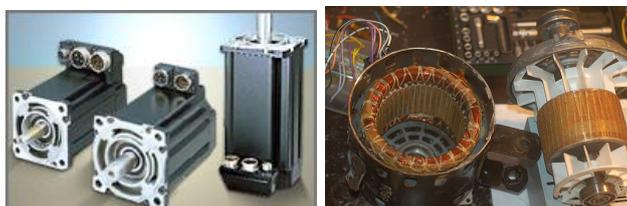


Figura 2.14: Analogia para motor de indução

É importante observar que o disco não pode girar na velocidade do ímã, ou então não haverá movimento relativo dos dois, a corrente induzida no disco será zero e não haverá pólos opostos pra haver atração e movimento. A velocidade do disco, no entanto, é proporcional à velocidade síncrona, podendo ser calculada pela equação 2.2[10]:

$$N_r = N_s \times s \quad (2.2)$$

onde N_s é a velocidade síncrona, N_r é a velocidade do rotor e s é uma porcentagem do escorregamento, ou seja, a velocidade do rotor é uma fração da velocidade síncrona. A figura 2.15⁶ mostra o motor de indução do experimento e um motor semelhante por dentro.



(a) Motor de indução do ex- (b) Motor de indução des-
perimento (MPL-A310F) montado

Figura 2.15: Motor de indução do experimento e outro motor desmontado

2.4.2 Kinetix

Kinetix é a mais nova família de variadores de frequência da Rockwell Automation desenvolvido com o objetivo de minimizar o espaço e fiação utilizados em quadros de automação industrial. A família Kinetix somente se comunica através da interface SERCOS, que conforme explicado no capítulo 2.5.2, é um protocolo de comunicação desenvolvido tendo em vista acionamento de eixos. A família Kinetix inclui ainda entradas próprias para ligação do sinal de feedback do motor, sinal de um encoder externo extra e ainda uma entrada para sensores que incluem fontes 24VDC que minimizam a necessidade de comunicação com módulos de expansão do controlador e eliminam a necessidade de fontes externas.

⁶fonte: <http://www.ab.com/motion/servomotors/lowinertia.html>

A família Kinetix é composta por 3 linhas de modelos, o Kinetix 2000, 6000 e 7000 que se diferenciam principalmente pela potência fornecida. Em todos os casos, os modelos são vendidos podem ser IAM (*Integrated Axis Module*) e AM (*Axis Module*). O IAM é composto de duas partes, uma de controle e uma de potência, vendidos em um único módulo. Os AMs são semelhantes à parte de potência dos IAMs. Ainda são disponibilizados equipamentos obrigatórios ou acessórios para o uso com essa família de variadores.

O LIM (*Line Interface Module*) é uma fonte para o Kinetix. Embora o Kinetix possa ser ligado diretamente à tomada trifásica, diversos componentes externos são necessários, como filtros de linha trifásicos e monofásicos, fusíveis, uma contatora específica externa pra Ligação do Contactor Enable (as ligações do Kinetix são detalhadas no capítulo 3.4). O LIM é um equipamento projetado, assim como a família Kinetix, para redução de espaço e fiação em quadros e este equipamento faz isso incorporando todo esse aparato num único módulo. O Power Rail também desenhado pra diminuir fiação e espaço. Este equipamento consiste num trilho que tem dupla função: encaixe e suporte dos módulos de potência e ainda a transmissão de alimentação e sinais de controle. Dentre os sinais de controle existem sinais de mau funcionamento de módulos de potência e outros parâmetros destinados à supervisão, pois o comando é feito pela interface SERCOS, sendo o mestre dessa comunicação o ControlLogix e não o módulo de controle do IAM. Estes dois equipamentos descritos são obrigatórios (o LIM pode ser substituído pelos seus equipamentos internos separadamente). Para auxiliar o Kinetix existem ainda filtros para o Power Rail, Shunt Modules, dispositivos externos para freio resistivo, entre outros, mas uma vez que esses equipamentos não fazem parte deste experimento, eles não serão discutidos aqui.

O módulo de controle do IAM coordena até 16 módulos de potência, indicando quais os parâmetros de atuação de cada módulo, como voltagem e frequência. A coordenação é feita a partir dos comandos recebidos do controlador ControlLogix, a partir do módulo SERCOS. O Kinetix recebe qual instrução deve executar, que vão desde comandos para desligamento de algum módulo de potência, passando por funções de posicionamento em algum ponto de referência, posicionamento na posição *home*, instruções de fiscalização ou refinamento de um dos módulos, e outras muitas funções disponíveis no ControlLogix e descritas no manual ControlLogix Motion Instructions disponibilizado no CD deste projeto. A comunicação entre o módulo de controle e os módulos de potência é feita através do Power Rail, que é um trilho usado pra transferir alimentação e alguns dados entre o IAM e seus módulos AMs (mesmo o IAM incluindo um módulo de controle e potência num único invólucro, ele também requer o uso do Power Rail). A figura 2.16⁷ mostra o Kinetix 6000 com 1 módulo IAM e 3 AMs.

⁷fonte:<http://www.ab.com/motion/servodrives/kinetix6000.html>



Figura 2.16: Kinetix com 1 IAM e 3 AMs

Os AMs são os módulos de potência e incluem entradas para feedback. Cada um desses módulos possui as suas próprias entradas, de modo que cada um possa operar a sua operação e malha de controle independentemente. A primeira entrada é o feedback do motor, que é destinada a encoders Stegmann Hiperface (podendo aceitar encoders TTL também), aceitando entradas seno, cosseno e de dados (que indica sobrecarga no motor, por exemplo), além de possuir uma fonte 9 V ou 5 V para a alimentação do encoder. A segunda entrada é semelhante à primeira e também foi destinada a encoders, mas neste experimento ela é usada com um encoder TTL ao invés de Stegmann como a primeira. A última entrada se destina a entradas binárias de sensores externos. As entradas podem ser usadas pra qualquer função, mas há instruções que procurarão automaticamente por determinadas entradas, como a instrução que retorna o objeto acionado pelo motor para a posição padrão (*home*). Mais de uma instrução usa diferentes entradas, e devido ao grande número de possibilidade, recomenda-se que se utilize o manual ControlLogix Motion Instructions. As entradas com utilização prevista neste experimento são a *Overtravel+* e *Overtravel-*. Os sensores de presença fotoelétricos serão ligados a esses pinos como medida de segurança para evitar que o carro do guindaste atinja e danifique os limites da correia linear.

O circuito de potência que efetivamente realiza a variação da frequência é mostrado na figura 2.17[11].

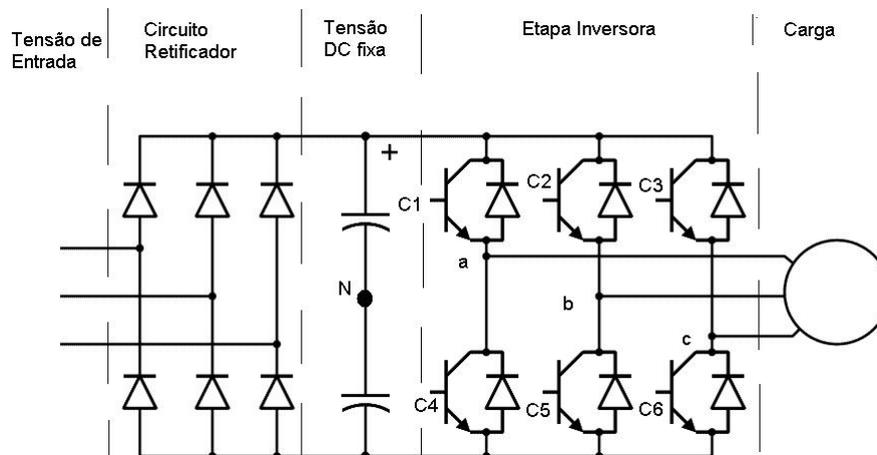


Figura 2.17: Diagrama de um circuito variador de frequência

O funcionamento do circuito é simples. A etapa retificadora transforma a entrada de corrente alternada em corrente contínua. É importante observar que embora essa parte do circuito indique que o Kinetix possa funcionar com uma única fase (ou seja, com uma tomada comum), esse modo de operação não é possível devido à lógica do Kinetix que indica um erro de ligação e impede o funcionamento do aparelho. Normalmente a decisão de se usar 1 fase ou 3 fases está relacionada à potência necessária na carga. Uma vez que a saída é trifásica, a potência é (para fases equilibradas e defasadas de 120°) 3 vezes maior, e uma vez que o circuito não tem uma fonte extra de potência, o fato poderia causar uma sobrecarga na alimentação monofásica, ou erros de operação na saída. Após a etapa retificadora, são adicionados 2 capacitores em série. O capacitor diminui o efeito de ripple, e o uso de 2 capacitores se deve somente ao fato de que o ponto entre eles normalmente é usado como ponto neutro na ligação. Os transistores da etapa inversora operam como chaves acionadas por uma entrada em PWM. O PWM consistirá numa onda quadrada com pulsos de largura variante de acordo com a forma de onda a ser reproduzida na saída. Como a saída senoidal é desejada, a largura do pulso de PWM varia também senoidalmente, causando na saída uma réplica do PWM, sendo que o valor médio do PWM será equivalente ao sinal senoidal conforme mostrado na figura 2.18[11], onde V_o indica o valor médio que é o valor "sentido" pela carga e V_i indica a saída real simulando o PWM. O valor negativo é em referência ao ponto N.

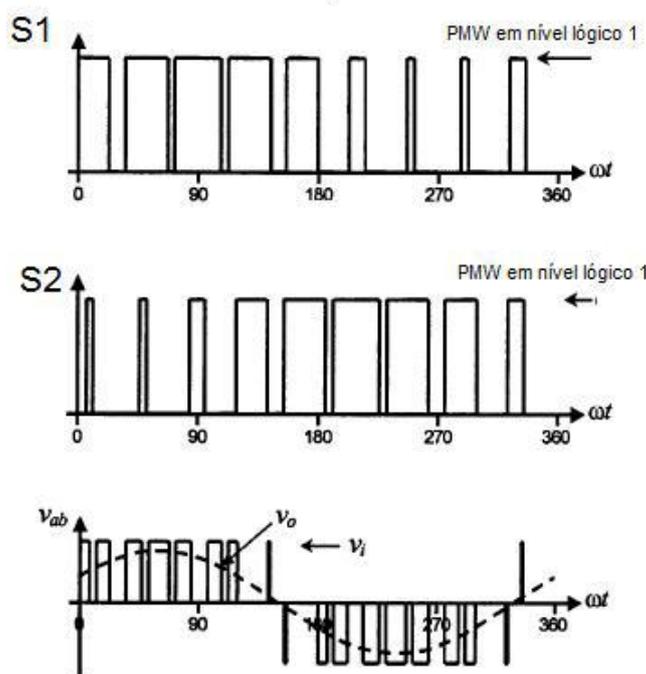


Figura 2.18: Exemplo de saída de uma circuito variador de frequência

Além do circuito apresentado, há a parte lógica (que inclui a origem do PWM) e outras etapas que fazem funções como eliminar harmônicas e controlar a qualidade da saída independentemente da carga. Essas funcionalidades são prometidas pela Rockwell Automation em sua descrição do produto [12], porém nenhum detalhe é fornecido sobre esses circuitos, sendo considerado um segredo relacionado à marca.

2.5 Comunicação

2.5.1 DeviceNet

DeviceNet é um protocolo de comunicação industrial criado pela Allen Bradley com o objetivo de conectar componentes de baixo nível (em geral sensores) com componentes de alto nível (controladores). O protocolo DeviceNet implementa as camadas 1, 2 e 7 da pilha OSI (*Open Systems Interconnection*), ou seja, o protocolo define uma infra-estrutura básica de rede e uma interface com o software de aplicação.

Embora tenha sido desenvolvido por uma empresa privada, o protocolo foi tornado aberto pra facilitar a interconexão de sistemas de diferentes procedências. Foi então criada a ODVA (*Open DeviceNet Vendors Association*) para manutenção das especificações do protocolo e para avaliação de equipamentos que implementam DeviceNet, além de fornecer assistência pra fabricantes que desejem produzir equipamentos DeviceNet.

A Rockwell Automation disponibiliza o software RSNNetWorx para criação e manutenção de uma rede DeviceNet (também cria redes Ethernet/IP e ControlNet). O software realiza a função de declarar quais dispositivos estão na rede, quais parâmetros o módulo DeviceNet deve usar pra se comunicar com os mesmos, além de quais informações o módulo deve buscar nesses sensores. Sem o RSNNetWorx, o módulo DeviceNet não sabe com quem conversa, pois a varredura apenas identifica os dispositivos no endereço da rede, e não faz a configuração necessária pra comunicação. O RSNNetWorx (mostrado na figura 2.19⁸) serve ainda pra serviços de manutenção da rede, como obter diagnósticos de funcionamento dos sensores ou calculo de banda passante restante.

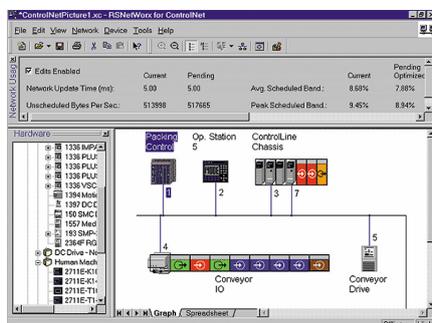


Figura 2.19: RSNNetWorx

2.5.1.1 Camada física

DeviceNet define o barramento como topologia de rede e este barramento inclui cabos de alimentação, comunicação e o cabo de aterramento. Os fios de comunicação devem ser brancos e azuis, enquanto que os cabos de alimentação são pretos e vermelhos. O quinto cabo, o de aterramento, não tem cor pré-definida. O barramento não pode ser dividido em ramos a não ser com o uso de um equipamento específico (TAP). A figura 2.20 [13] mostra um exemplo de rede semelhante ao usado no experimento, exceto pelo fato de não serem necessários TAPs. O resistor

⁸<http://www.rockwellautomation.com/rockwellsoftware/products/>

de terminação é 121Ω .

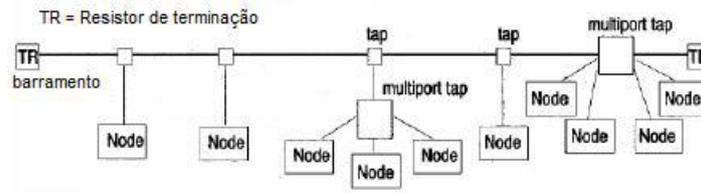


Figura 2.20: Esquemático do barramento DeviceNet

O protocolo DeviceNet também define algumas especificações sobre conectores, embora eles não sejam únicos. Os conectores são necessariamente [14]:

- Todos os nós devem ter conectores machos;
- Um nós deve poder ser desconectado sem comprometer o sistema ou afetar qualquer outro nó;
- Conectores devem suportar no mínimo, 8A, 24V ou 200VA;
- No mínimo 5 pinos de conexão devem existir. Um sexto pino pode existir para fins de aterramento da carcaça do conector e/ou equipamento;

Um tipo comum de conector, que será usado nas conexões deste projeto, pode ser visto na figura 2.21[14].

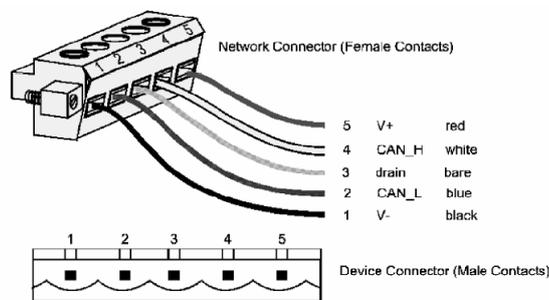


Figura 2.21: Esquemático do conector DeviceNet

Os fios a serem usados na construção do barramento podem ser de 3 tipos: espesso, fino ou flat, ou existe a opção de se combinar os 3 tipos de cabo. O cabo espesso e o flat costumam ser usados em longas distâncias devido à maior resistência mecânica, enquanto que o fino costuma ser usado apenas em conexões locais. Os cabos espesso e fino são de par trançado individualmente estanhados. O cabo espesso possui diâmetro aproximado de 10.8mm e 3 voltas/ft. Para o par de sinal, o tamanho do fio é #18 (19x30AWG), que o mesmo tamanho do fio de aterramento. A impedância típica é de 120Ω a 1MHz, a capacitância é de 12pF/ft e o atraso é de 1.36ns/ft. Para o par de energia o que muda é o fio que é #15 (19x28AWG) e deve conduzir até 8A. O cabo é finalizado com uma blindagem de Mylar em cada cabo interno. O cabo fino de sinal tem diâmetro

médio 6,13mm com 4,8 voltas/ft. O tamanho do fio é #24(19x36AWG) e as características de impedância, capacitância e atraso são as mesmas do cabo espesso. O fio do cabo interno de energia tem tamanho de #22 (19x34AWG) e deve conduzir até 3A, que é o mesmo tamanho do cabo interno de aterramento. O cabo fino também é finalizado com blindagem de Mylar. O cabo flat deve ser feito de Santoprene ou TPE, deve caber em eletrodutos de 25mm e deve poder conduzir 600V ou 8A. Não há outras especificações pra este cabo [14].

A rede DeviceNet suporta até 64 nós independentemente da velocidade de comunicação usada. A comunicação no entanto, afeta a distância máxima que o tronco pode ter entre o controlador da rede DeviceNet e o nó. Quanto maior a velocidade, menor a distância máxima, que também depende do tipo de cabo escolhido. A tabela 2.2 resume as distâncias máximas do tronco dependendo do tipo de fiação e da taxa de transmissão (todas as 3 possíveis) [14].

Tabela 2.2: Distâncias máximas do tronco DeviceNet

Taxa de transmissão	125kbps	250kbps	500kbps
Fio espesso	500m	250m	100m
Fio fino	100m	100m	100m
Cabo flat	420m	200m	75m

A rede DeviceNet possui transmissão diferencial, sendo o par de fios de transmissão identificados como CAN-H (branco) e CAN-L(azul). Dois estados do fio representam os sinais lógicos, estado recessivo e o dominante. O CAN-H muda de 2,5V(recessivo) para 4,0V(dominante) e o CAN-L muda de 2,5V (recessivo) para 1,5V (dominante). Todas as voltagens são tomadas em relação ao V- da rede. O 1 lógico é interpretado quando ambos os fios estão em estado recessivo e o 0 lógico será interpretado quando ambos se encontram em estado dominante. O estado sem transmissão da rede é o recessivo. Como a rede é de barramento em paralelo, isso resulta em uma configuração AND. Quando qualquer nó muda pra 0 lógico que é o estado dominante, toda a rede entrará nesse estado, daí o nome dominante. Para transmissão de 1 lógico, toda a rede tem que estar em estado recessivo.

2.5.1.2 Camada de Enlace

O protocolo DeviceNet define o CAN (Controller Area Network) como protocolo dessa camada e assim o formato do frame é o mesmo. O primeiro bit identifica o início da transmissão. Os próximos 12 bits, incluem o identificador e o *Remote Transmission Request* (RTR). Esse 12 bits são usados como indicador de prioridade e são usados quando a há colisão, explicado mais adiante. Os próximos 6 bits são conhecidos como bits de controle. Os 2 primeiros desses 6 são fixos, e os outros 4 indicam o número de bytes de dados a serem recebidos e esse número pode variar de 1 a 8 bytes, que vem em sequência aos 6 bits de controle. Após os bytes de dados, 16 bits são pra detecção de erros de acordo com o algoritmo *Cyclic Redundancy Check*. Os próximos 9 bits são de *acknowledge* e fim de frame. O tempo de transmissão entre frames é de até 3 tempos de bit.

O acesso ao meio é definido como *Carrier Sense Multiple Access*. Isso significa que qualquer nó verifica se há alguma transmissão em andamento antes de transmitir seus dados. Uma vez

havendo uma transmissão, o nó espera até 3 tempos de bit e tenta novamente até que possa transmitir. Se houver uma transmissão simultânea antes que os nós possam detectar a transmissão do outro, os bits de identificador e RTR farão a arbitragem de quem tem maior prioridade. Dado que o barramento possui um efeito de porta AND, o nó que transmitir mais 0 lógicos transmitirá enquanto que os outros sofrerão interferência. Quando a prioridade definida é mais alta, o valor numérico nos bits dos campos identificador e RTR será mais baixo, o que garante a transmissão deste frame quando há frames de prioridade mais baixa sendo transmitidos. Os nós que detectar que sua transmissão está sofrendo interferência irá abortar a transmissão antes dos 7 bits de fim de frame e tentará novamente mais tarde.

2.5.1.3 Camada de Aplicação

DeviceNet define o protocolo CIP (*Common Industrial Protocol*) como protocolo da camada de aplicação. O protocolo CIP é orientado a objeto, i.e., o protocolo define que cada elemento da rede tenha definido objetos contendo atributos (ou dados), serviços (ou comandos) e comportamento (resposta a comandos). De fato, cada nó é enxergado como uma coleção de objetos no protocolo CIP, o protocolo sequer reconhece dados que não estejam contidos em objetos. Os objetos e seus atributos são endereçados por um esquema padronizado que consiste em um endereço MAC ID (*Media Access Control IDentification*), que é um inteiro que identifica o nó. O *Class Identifier* é um inteiro que identifica qual classe de objeto é buscada. O *Instance Identifier* identifica a instância do objeto. Como objetos podem ter atributos ou serviços, o próximo identificador pode ser um *Attribute ID* ou *Service Code*, e ambos consistem em um inteiro identificador.

Os objetos do protocolo CIP são padronizados de modo que a comunicação seja garantida independentemente do fabricante (ou até do protocolo, já que CIP também é o protocolo de aplicação de outros padrões como Ethernet/IP e ControlNet)[15]. A lista padrão pode ser estendida pelo fabricante. Um dos principais objetos padrão é o Communication Object, que contém dados da comunicação como o tamanho máximo das mensagens, e também possuem os comandos e comportamentos necessários pra se iniciar uma conexão. Devido à presença deste objeto, DeviceNet é um protocolo orientado a conexão. Outro objeto sempre presente é o objeto de identidade, que contém informações sobre a natureza do nó, por exemplo, sensores, válvulas e motores. Outros objetos podem depender da natureza do equipamento e dependem do que o equipamento faz. Sensores devem possuir objetos ligados ao tipo de respostas como objeto *analog output* e acionadores podem ter objetos como *soft starts*.

2.5.2 SERCOS

SERCOS é um protocolo de comunicação industrial desenvolvido por duas organizações alemãs, a ZVEI e a VDW e foi desenvolvido como um protocolo aberto destinado a aplicações que envolve movimento, em especial o movimento em eixos coordenados. A motivação inicial foi a substituição de drivers de acionamento de movimentação analógicos por um driver digital, que oferecesse um baixíssimo tempo de ciclo, alta imunidade a ruído e principalmente, sincronia precisa entre dois ou mais eixos.

SERCOS corresponde à *SE*rial *Re*al-time *CO*munication *S*ystem (numa tradução livre, Sistema de comunicação serial em tempo real), e foi o protocolo registrado como uma marca SERCOS interface™, embora seja um protocolo totalmente livre (na verdade, é o único protocolo de sincronia de alto desempenho totalmente livre [16]).

O protocolo SERCOS tem 3 versões diferentes, nomeadas de SERCOS seguido do numeral romano correspondente. As duas primeiras versões (I e II) são baseadas em comunicação via fibra óptica, e diferem principalmente no circuito usado e nas taxas de transferência de bits possíveis. As duas utilizam um circuito desenvolvido especificamente para o protocolo, e a versão II é na verdade uma atualização da primeira versão. A versão III não foi criada com o intuito de substituir a versão II, mas de diversificar as aplicações do protocolo e é baseada em Ethernet. Uma vez que a versão II será usada no projeto, todas as informações a seguir são referentes a essa versão.

É importante notar que embora o protocolo permita outras funções, como I/O, a interface SERCOS não é destinada ser um barramento de campo pra leitura de sensores ou qualquer outra função que não seja o comando de movimentações com precisão. A inclusão de outras possibilidades se deve à simplificação da rede, evitando que fosse necessária outra rede pra fazer essas leituras, o que também criaria a necessidade de se sincronizar essas redes. É comum que se use a interface apenas para leitura de sensores de posição ou informações pertinentes ao movimento que precisem de sincronia com o mesmo, mas em geral outras redes são usadas para a leitura de outros sensores não essenciais ao movimento.

Assim como o protocolo DeviceNet, o protocolo SERCOS define as camadas 1, 2 e 7 da pilha OSI. Isso significa que é previamente definido o meio de comunicação, o compartilhamento desse meio e as mensagens reconhecidas.

2.5.2.1 Camada Física

As versões I e II são baseadas em fibra óptica. Um único laço de fibra óptica é usado pra transportar toda a informação de comando e de I/O necessária.

A fibra óptica é naturalmente imune a ruídos eletromagnéticos e facilita a fiação ao dispensar conduítes e conectores complexos. Por outro lado, a fibra óptica é cara e de difícil reparo, o que requer um planejamento inteligente da rede, de modo que as fibras tenham comprimento minimizado e estejam protegidas.

O conector para transmissores e receptores de mensagens é definido como FSMA (IEC 60874-2). Além disso, o conector deve ter um nível de qualidade mínimo de 5 e deve ter um anel conector metálico. Um exemplo de conector pode ser visto na figura 2.22 ⁹.

⁹<http://www.hubersuhner.com.br/co-br/pt/hs-p-fo-con-fsma-image1>



Figura 2.22: Exemplo de conector FSMA, utilizado na interface SERCOS

Tanto a fibra óptica plástica, como a fibra óptica de fibra de vidro são aceitas pela interface SERCOS, mas ambas devem operar como ondas de comprimento de 650 nm (o que corresponde à luz vermelha no espectro). A fibra óptica de plástico (POF) deve ter 1 mm de diâmetro e a distância máxima permitida em todos os nós da rede é de 10000 m, considerando a distância máxima entre nós como 40m. A fibra de vidro (HCS) deve ter $200\mu\text{m}$ de diâmetro, distância máxima 50000 m e distância máxima entre nós de 200m. Em ambos os casos, o diâmetro da fibra interna que é especificado, não há especificações para o diâmetro do cabo. Dentre os materiais em uso para fibra óptica, a atenuação máxima permitida é 220 dB/km para o plástico e 6 dB/km para a fibra de vidro. [16]

Os transmissores de receptores devem respeitar todas as especificações acima e devem ser autorizados pela SERCOS International e.V., que é a associação responsável pela manutenção do protocolo e avaliações de conformidade. O site da associação (www.sercos.com) fornece datasheets dos equipamentos aprovados. Não há repetidores de fibra óptica aprovados para o uso em redes sercos

O protocolo SERCOS determina que a rede precisa ser construída em anel, sendo que o controlador da rede manda mensagens para o primeiro nó, que retransmite para o próximo nó e assim por diante até que se retorne ao canal de recepção do controlador da rede. Um controlador de rede pode comandar mais de um anel, dependendo da sua capacidade computacional. Uma vez que o processamento das informações deve ser feito em cada nó da rede, a complexidade computacional se refere apenas ao número de nós na rede, uma vez que o controlador é responsável somente por tarefas de comunicação. A comunicação é feita baseada no algoritmo mestre-escravo. O Controlador da rede é o mestre, o que significa que apenas ele pode originar mensagens. Todos os outros nós da rede escravos, e somente a interface SERCOS III permite a comunicação entre nós da rede. A figura 2.23[16] mostra um exemplo de rede SERCOS.

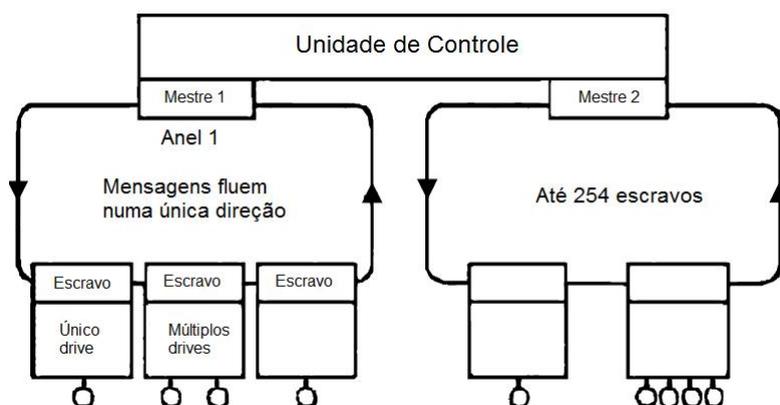


Figura 2.23: Exemplo de rede SERCOS com 2 anéis

Cada anel deve ter no máximo 254 nós, mas dependendo da necessidade de desempenho da rede, o anel pode ter o número de nós limitado. A rede pode operar em 4 diferentes taxas de transmissão: 2, 4, 8 e 16 Mbits/s. A rede opera em ciclos, conforme será visto na próxima subseção, e o tempo entre cada ciclo também pode ser definido, podendo ser: 62.5, 125, 250 e 500 μ s ou múltiplos de 1 ms. Quanto maior a taxa de transmissão, maior a possibilidade de erros, mas também maior a densidade de nós em um anel, ou maior o número de anéis possíveis. O tempo de ciclo maior também permite a maior densidade de nós no anel, mas diminui a quantidade de informação a ser enviada pra cada nó. Uma consideração sobre a quantidade de escravos a ser adicionada e sobre o desempenho da rede devem ser feitas para a escolha destes parâmetros. A tabela 2.3[16] resume alguns exemplos de configurações.

Tabela 2.3: Exemplos de configurações SERCOS

Tempo de Ciclo	Tamanho da Mensagem	Taxa de Transmissão	Número de Escravos	Taxa de dados (não-cíclicos)	Tempo de Ciclo Remanescente
2ms	32 bytes	2 Mbits/s	8	8 kbits/s (2 bytes)	390 μ s
1ms	32 bytes	4 Mbits/s	8	16 kbits/s (2 bytes)	125 μ s
1ms	36 bytes	8 Mbits/s	15	32 kbits/s (4 bytes)	208 μ s
0.5ms	36 bytes	16 Mbits/s	14	128 kbits/s (8 bytes)	113 μ s
2ms	Telegrama Padrão 2,3,4	16 Mbits/s	112	8 kbits/s (2 bytes)	330 μ s

2.5.2.2 Camada de Enlace

Uma das principais características do protocolo SERCOS é a sua capacidade de manter a sincronia entre o mestre e seus escravos. O feito é obtida a partir do envio regular de mensagens chamadas telegramas. Um telegrama é uma sequência rígida de bits contendo informações de dados e sincronia, e pode ser de três tipos:

- **Master Synchronization Telegram (MST):** é a partir deste telegrama que os escravos adquirem o seu relógio e realizam correções na sua sincronia. Este telegrama é diferente

durante a inicialização do sistema, e nesse caso ele contém dados sobre o modo de operação de cada drive (posição, velocidade ou torque). Durante a comunicação normal, este telegrama não contém dados.

- **Amplifier Telegram (AT):** Resposta dos escravos ao MST, contendo, no campo de dados, 3 tipos de informações. 8 bits são destinados ao status do escravo, informando se ele está pronto pra operar e no modo correto. 16 bits contem informações de tempo não-crítico, como limites de torque, limites de posição, constantes de tempo ou ganhos no loop. A última informação consiste em 1 a 16 IDNs (serão detalhadas na próxima seção) com informações de feedback, como posição ou velocidade atuais.
- **Master Data Telegram (MDT):** Provê dados para todos os escravos. A sua estrutura é semelhante a do MST, mas aqui o campo de dados sempre inclui informações. As informações são organizadas em campos fixos como no AT. Os primeiros 8 bits habilitam ou desabilitam algum escravo. O segundo campo contém informações de setup como ganhos de loop, ou funções específicas como homing (enviar o equipamento de volta a posição original). O último campo contém IDNs (comandos) a cada escravo, dependendo da aplicação.

Todos os telegramas consistem em 5 campos, sendo 2 delimitadores (um no início e outro no final, ambos contendo a mensagem 01111110), um de endereço, um de dados e um de checagem de erro. Os bits são identificados no tempo, isto é, o sinal permanece em 1 enquanto o bit for 1, e o mesmo vale pro valor zero (ou seja, o sinal não retorna pra um valor padrão entre bits iguais). Pra evitar problemas de sincronia, o controlador insere um bit de valor 0 toda vez em que ocorra uma sequência de cinco bits com valor 1.

Como meio de se ordenar o compartilhamento do meio, a interface SERCOS delimita tempos de transmissão pra cada nó da rede. Um escravo irá responder ao comando do mestre apenas em tempos pré-determinados, de modo que dois escravos não realizam uma transmissão ao mesmo tempo. A sincronia é renovada periodicamente a cada ciclo, que envolve uma troca padrão de mensagens entre o mestre e seus escravos. O tempo de ciclo é determinado pelo operador, conforme descrito anteriormente. Cada ciclo contém os seguintes passos [16]:

1. O mestre envia um MST, que contém as informações de tempo para os escravos enviarem e receberem novas informações
2. Após um tempo decorrido ao final do MST, o primeiro escravo transmite o seu AT, contendo informações pertinentes a ele. Cada nó então segue transmitindo seus próprios ATs de acordo com o cronograma definido no passo 1.
3. Após a última transmissão de AT, o mestre envia o MDT contendo informações para todos os escravos. Cada escravo sabe previamente a posição correspondente à sua informação.
4. O ciclo recomeça do passo 1.

2.5.2.3 Camada de Aplicação

Todas as mensagens na interface SERCOS são caracterizadas por seu *identification number* ou IDN. O IDN é um número que é associado a uma função específica pré-definida na interface SERCOS. Existem 32767 IDNs pré-definidos na especificação, mas os equipamentos SERCOS não precisam incluir necessariamente todos eles, apenas um subconjunto compatível com a sua aplicação. Existe ainda a opção de incluir IDNs específicos de um fabricante (até 32767 IDNs, totalizando 65534), mas somente em casos em que não existe um IDN padrão para a execução da função. Alguns IDNs definidos por fabricantes já foram ou estão em processo de se tornar parte do padrão SERCOS, de modo a expandir a funcionalidade universal da interface.

O IDN é na verdade o conteúdo do campo de dados no telegrama descrito na 2.5.2.2, seja o telegrama um AT ou MDT, e assim, toda a informação é transmitida nesse padrão, desde ganhos de loop, a informações de erro como sobreaquecimento de equipamento. A lista completa de todos os IDNs é muito extensa e é obtida por fabricantes interessados em produzir equipamentos SERCOS.

Capítulo 3

Desenvolvimento e Resultados Experimentais

3.1 Introdução

O sistema pode ser dividido em duas partes: mecânica e elétrica. A montagem mecânica engloba os aspectos da forma do sistema como uma grua para coleta de materiais. Já a parte elétrica envolve controle e redes de comunicação.

Na montagem mecânica, é necessário o projeto de uma estrutura metálica, onde ficarão todos os componentes do sistema (com exceção do computador). Nessa estrutura, deve haver suportes para o motor, os buffers com as peças, o depósito de saída, o guia-linear e a correia porta-cabos. Também são necessários o projeto de uma peça de fixação do motor e a definição das características do acoplador motor-guia.

Para o melhor entendimento do sistema, sua modelagem é necessária. Uma primeira descrição em redes de Petri é mostrada aqui. Também se pode fazer um modelo utilizando autômatos.

A montagem e configuração da rede DeviceNet deve ser feita para se obter informações dos sensores indutivos. As medições dos encoders e dos sensores fotoelétricos serão feitas pelo controlador, através do Kinetix.

É necessária também a configuração e utilização dos softwares RSLogix 5000, RSLinx e RS-NetWorx.

Algumas modificações do projeto original foram feitas e podem ser vistas comparando-se as figuras 3.6 e 3.7.

3.2 Concepção do Sistema

3.2.1 Descrição da Planta de Fundição

A figura 3.1 mostra o projeto inicial, em escala, que contém todos os elementos para a concepção do sistema.

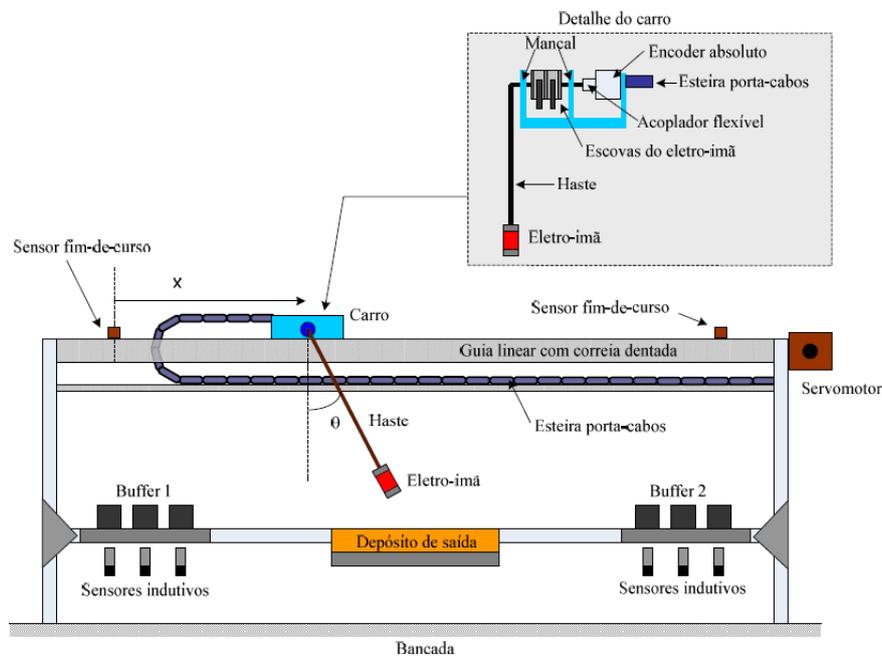


Figura 3.1: Detalhes do sistema

Os elementos necessários são a estrutura de suporte, buffers para as peças, depósito de saída, servomotor, guia linear com correia dentada, esteira porta-cabos, sensores indutivos, sensores fotoelétricos, carro do guia linear com um encoder absoluto, haste e eletroímã.

A haste carrega o eletroímã, que é responsável pela coleta das peças. O movimento do carro que leva o encoder com a haste é feito pelo servo motor. A correia dentada do guia linear utiliza o torque do motor para a movimentação do carro.

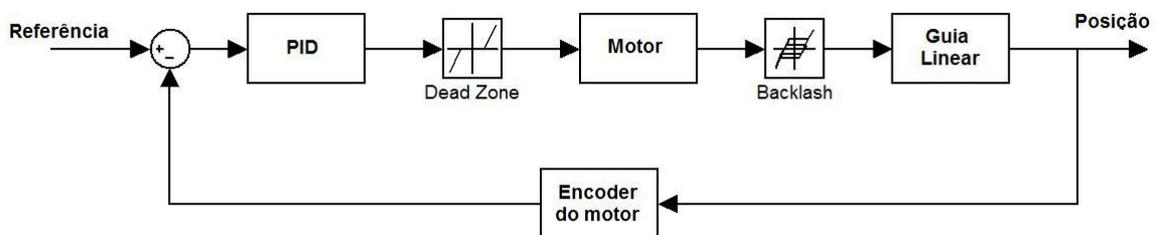


Figura 3.2: Diagrama de blocos do sistema

Há seis posições pré-definidas em relação aos buffers onde ficam as peças. Os sensores indutivos, ao detectarem a presença de uma peça, mandam essa informação ao controlador. Este, por sua vez, passa essa referência ao controlador PID. Esse controlador é feito a partir de um bloco próprio do RSLogix 5000. Para o correto projeto do PID, é necessário que o sistema esteja todo montado e funcionando.

A posição do carro é medida com a utilização do encoder do motor e a relação entre a posição angular do eixo (medida com esse encoder) e a distância percorrida pela guia linear. A saída do sistema é justamente a posição do carro, que deve estar sobre um dos seis buffers de peças, para que estas possam ser coletadas com o eletroímã.

Há, nesse sistema, algumas não-linearidades, como pode ser visto no diagrama da figura 3.2. Primeiramente há a zona morta do motor. Existe também folgas no acoplamento do eixo do motor com o eixo do guia linear. As folgas podem ser minimizadas com a escolha de um acoplador adequado e com o correto posicionamento entre o eixo do motor e o guia.

Os sensores fotoelétricos são utilizados como sensores de fim-de-curso do carro da haste, em relação ao guia linear. Esses sensores definem uma 'saturação' da posição que o carro pode atingir. O comprimento total do guia linear é 1,85 metro, mas o espaço que se pode percorrer nela é um pouco menor, inclusive por medida de segurança (para não deixar que o carro bata nos limites).

Os sensores indutivos - que são ligados por uma rede DeviceNet - informam o controlador sobre a presença ou ausência de peças em seus respectivos reservatórios. Com a informação de onde há peças e onde não as há, o CLP calcula, utilizando-se de um sistema de prioridades, a referência para a posição da haste.

O controlador do sistema - o CLP - tem a função de coletar dados do sistema através dos sensores para realizar o controle do motor. O controlador se comunica com a rede DeviceNet da planta, onde estão os seis sensores indutivos.

Com o correto posicionamento do carro e ângulo da haste, aciona-se o eletroímã para a coleta da peça. Ele só será desligado quando a haste estiver na posição para liberar a peça no depósito de saída.

A figura 3.3 mostra uma rede de petri que é um resumo do funcionamento do sistema: se há uma referência definida, busca a peça naquele local; quando a peça chegar à saída, define-se nova referência.

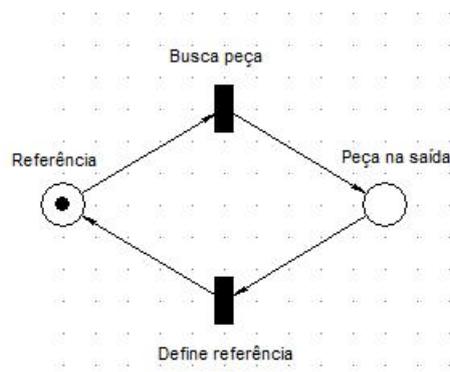


Figura 3.3: Rede de Petri - Resumo do sistema.

Pode-se ver a ordem de ações tomadas pelo controlador na rede de petri da figura 3.4.

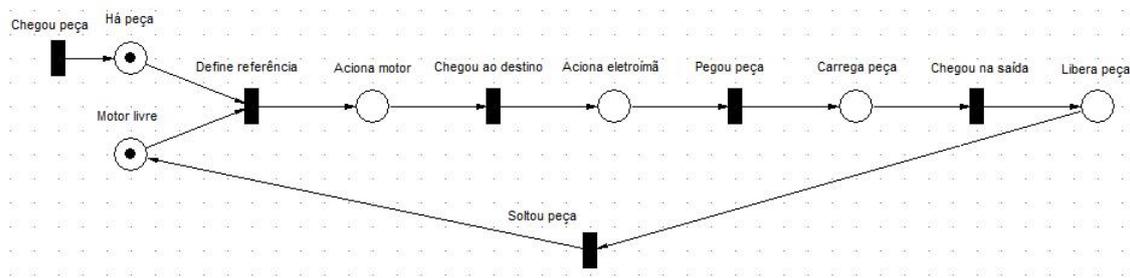


Figura 3.4: Rede de Petri - Coleta de uma peça.

Os dois lugares com marcas - 'Há peça' e 'Motor livre' - são os fatos que desencadeiam o processo. Se há uma peça a ser coletada e, se o motor está livre, define-se uma referência nova para o controlador. Este aciona o motor que, ao chegar ao seu destino, provoca o acionamento do eletroímã. Com a peça presa, ela é carregada para o buffer de saída e, ao chegar lá, é liberada. Sem a peça pendurada na haste, o motor está livre para realizar outro percurso, caso haja outra peça.

O controlador deve obedecer a uma prioridade na seleção de qual buffer terá a peça coletada. Primeiramente, deve haver somente uma referência para o controlador. A planta pode também realizar uma priorização das peças de acordo com alguma regra pré-definida.

A figura 3.5 mostra uma rede de petri que faz a seleção entre dois caminhos com prioridades diferentes.

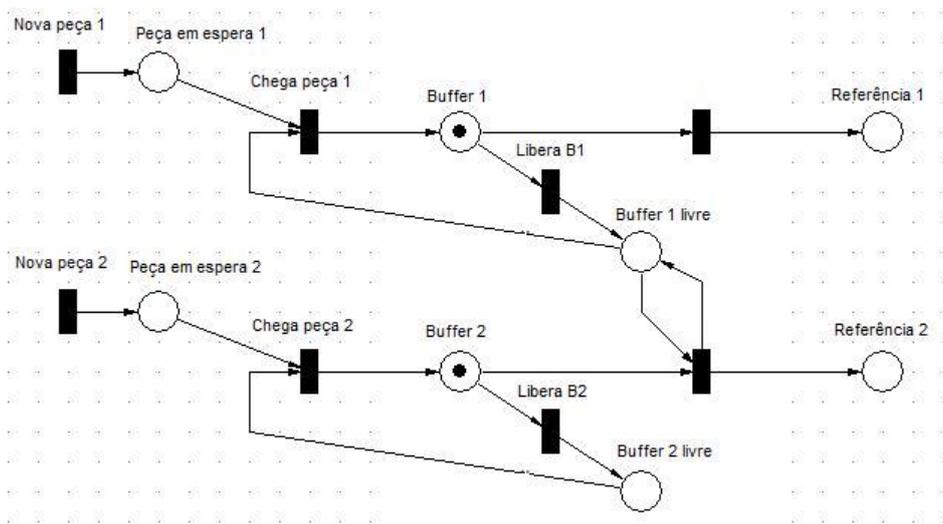


Figura 3.5: Rede de Petri para definição de referências de posição.

Como se pode ver na figura 3.5, basta que haja uma peça no buffer 1 - representada por uma marca em 'Buffer 1' - para que a referência 1 seja definida - marca em 'Referência 1'. Já para se definir a referência 2, além de haver peça no 'Buffer 2', também é necessário que não haja peça no buffer 1 - marca em 'Buffer 1 livre'.

Conclui-se que as peças do buffer 1 têm maior prioridade no transporte do que as peças do buffer 2.

Para mais peças (mais posições de referência), basta haver as mesmas relações entre as peças dos buffers 1 e 2. Se não houver peças em 1 e 2, libera-se o 3. E assim sucessivamente.

As redes de Petri das figuras 3.3, 3.4 e 3.5 foram feitas no software Visual Object Net 2.7. Os arquivos dessas redes estão no CD do projeto.

3.2.2 Arquitetura do Controle

A figura 3.6 mostra a arquitetura original desse sistema.

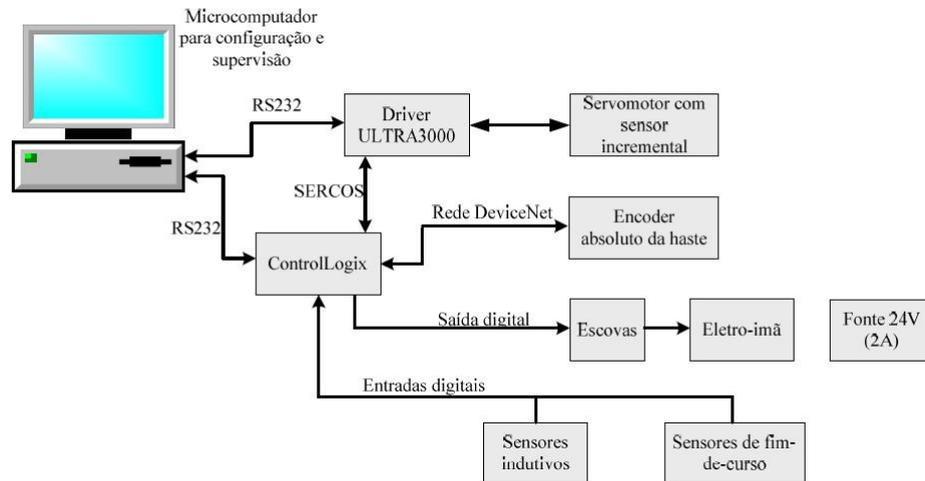


Figura 3.6: Arquitetura projetada.

Apesar de se manterem a maioria dos elementos do sistema da figura 3.6, algumas modificações foram feitas - tanto troca de equipamento, quanto mudança de conexões.

A figura 3.7 mostra as modificações feitas.

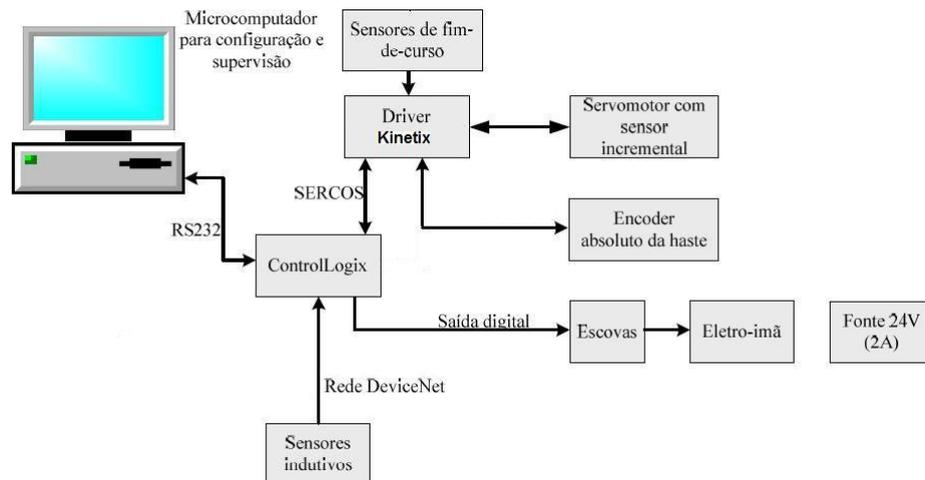


Figura 3.7: Arquitetura do sistema.

O controlador ControlLogix 5560 é o centro desse sistema. Ele se comunica com um microcomputador - por uma interface RS232 -, onde se fará a programação do controle utilizando-se a linguagem Ladder. Esse modelo de controlador já vem com uma interface SERCOS acoplada a ele. É essa interface que faz a comunicação do controlador com o Kinetix - responsável pelo acionamento do motor.

O controlador possui poder computacional suficiente para gerenciar toda essa planta de automação, além de possuir a capacidade de se comunicar com outros módulos que serão necessários - DeviceNet, I/O e a própria SERCOS.

A SERCOS interface foi selecionada primeiramente para que se utilize um driver digital, ao

invés de um driver de movimentação analógico. Essa interface faz comandos de movimentação com precisão e faz uma sincronia precisa entre eixos.

O Kinetix - um inversor de frequência - é o driver utilizado para o acionamento do motor. Além da alimentação e da comunicação com o controlador e com o motor, ele possui três conexões importantes (figura 3.8).

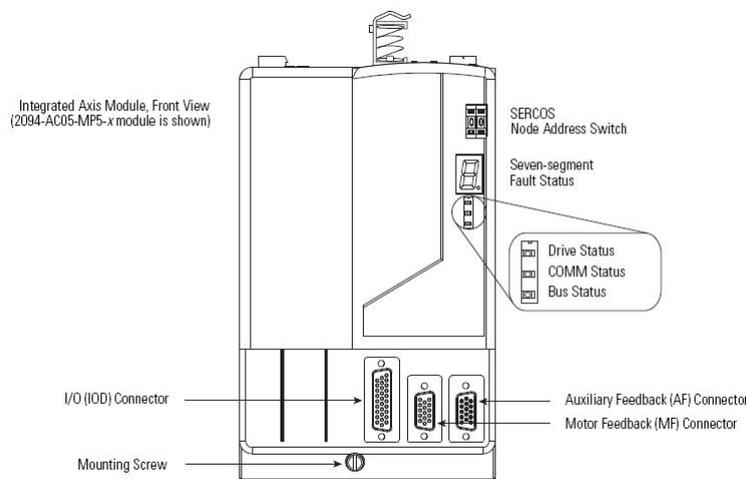


Figura 3.8: Desenho da vista frontal do Kinetix 6000.

Essas três conexões são 'Motor Feedback', 'Auxiliary Feedback' e 'I/O'.

O 'Motor Feedback' recebe a realimentação do motor. É aí que se faz a leitura do encoder do motor.

O sistema possui um outro encoder, que é o da haste. O cabo desse encoder se liga à entrada 'Auxiliary Feedback'. Pode-se ver outra modificação entre o projeto original (figura 3.6) e o modificado (figura 3.7): o encoder da haste se comunica com o bloco do Kinetix, não com o do ControlLogix.

A terceira conexão é a de 'I/O'. A figura 3.7 mostra uma ligação entre os sensores de fim-de-curso e o Kinetix. Essa ligação é feita pela porta 'I/O'. Essa porta possui uma utilização genérica. Por exemplo, em vez de se utilizar um módulo de I/O juntamente ao ControlLogix, pode-se utilizar essas entradas da porta 'I/O' para se fazer o acionamento do eletroímã.

O sistema possui uma rede DeviceNet, que se comunica também com o controlador. Essa rede possui seis sensores indutivos, os quais, além de detectarem a presença de material ferromagnético, possuem dados importantes para a detecção de possíveis erros na operação do sistema.

3.3 Montagem Mecânica

3.3.1 Estrutura

O primeiro passo para a montagem mecânica do sistema foi o projeto da estrutura metálica de suporte. As figuras 3.10 e 3.11 mostram os desenhos feito em SolidWorks para esse projeto.

Definiu-se lugares na estrutura para cada parte do sistema. No topo da estrutura, uma viga de comprimento 1,85m foi colocada para se fixar o guia linear, que tem esse comprimento total. O guia - além de carregar encoder, haste, peça metálica, cabos e conectores - estará sujeito a movimentação rápida, vibrações e esforços. Por esse motivo, além das barras onde se fixou o guia, outras duas barras foram colocadas embaixo daquelas, para ajudar a suportar os esforços.

Cada barra possui uma seção transversal de 30x60 mm. As barras de fixação do guia, juntamente com as duas barras de suporte, possuem, juntas, uma seção transversal de 60x90 mm, que suportará os esforços. Esse conjunto de barras é o suficiente para impedir a flexão do suporte do guia, o que provocaria mais não-linearidades no sistema.

Estimou-se um comprimento de aproximadamente 70 cm para a haste de coleta de materiais. Pode-se observar, na figura 3.11, uma distância de 600 mm entre as barras de suporte do guia e as barras onde ficarão os buffers com as peças. Somando a esses 600 mm a espessura das barras, a altura do guia e do encoder - onde ficará a haste -, há espaço suficiente para a movimentação do carro, sem que a haste bata na estrutura.

As barras horizontais inferiores são as responsáveis pela fixação dos sensores e dos buffers das peças. Além disso, também fica aqui o depósito de saída das peças.

Na figura 3.11, observa-se à direita um bloco com a altura de 670 mm. É ali que se fixará o suporte do motor. Ele deve ficar nessa posição para que seu eixo se alinhe com o eixo de fixação do guia linear (figura 3.9).



Figura 3.9: Guia linear

Um outro conjunto de barras horizontais que pode ser visto nas figuras 3.10 e 3.11 - no meio da estrutura - servirá de apoio à esteira porta-cabos. Não se definiu a altura dessas barras para que elas possam ser ajustadas na hora da montagem final do sistema. Não há a necessidade de uma altura exata, já que a esteira é flexível. Deve-se apenas tomar cuidado na montagem para que haja espaço suficiente para a sua movimentação.

Os 'pés' da estrutura foram projetados para que se possa aparafusá-la à mesa.

Todas as barras foram fixadas umas às outras por cantoneiras (figura 3.12).

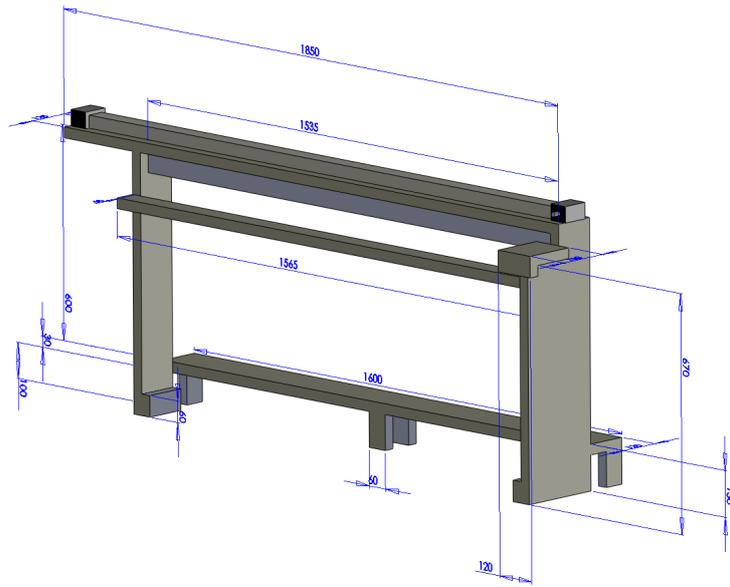


Figura 3.10: Projeto do suporte

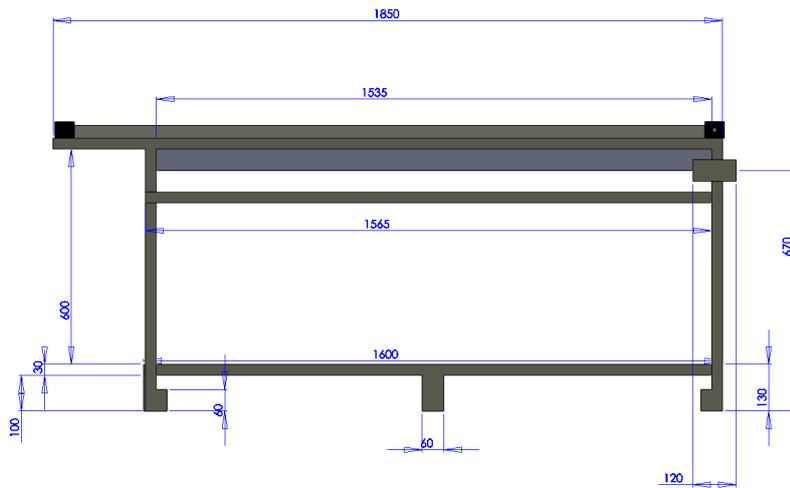


Figura 3.11: Projeto em SolidWorks



Figura 3.12: Cantoneira de fixação.

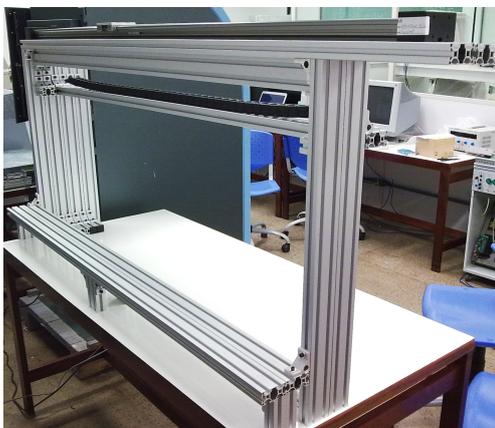
Basta encaixar uma cantoneira em duas barras perpendiculares e fixá-la com parafusos e porcas.

Esse tipo de cantoneira foi selecionado porque garante uma boa rigidez às conexões da estrutura, já que esta estará sujeita a vibrações e esforços da movimentação do carro do guia linear.

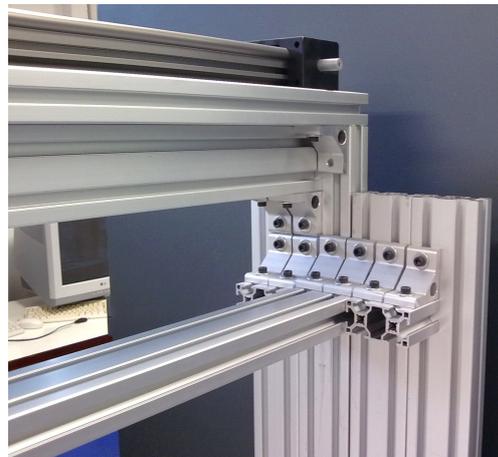
As peças para essa montagem foram compradas da Famak. O resultado da montagem pode ser visto nas figuras 3.13 e 3.14.



Figura 3.13: Vista frontal da estrutura.



(a) Outra vista.



(b) Suporte da esteira e canto do motor.

Figura 3.14: Estrutura no Laboratório de Automação.

3.3.2 Suporte do Motor

O motor precisa de um suporte específico. Ele deve ter quatro furos que se alinhem aos quatro furos do motor, para que este seja fixado por quatro parafusos. Além desses, há a necessidade de um furo maior, por onde passará o acoplador que ligará o motor ao guia linear. O diâmetro externo do acoplador é de aproximadamente 40 mm. O diâmetro do furo central do suporte foi definido em 50 mm para se evitar possíveis choques devidos a vibrações. A figura 3.15 mostra o motor junto com o suporte.



Figura 3.15: Motor e suporte.

Uma chapa de alumínio foi utilizada para a montagem dessa peça. Sua espessura é de 10 mm - suficiente para agüentar o peso e os esforços do motor.

A chapa foi cortada e soldada para ficar no formato em 'L'. Os furos foram feitos numa furadeira de coordenadas.

A peça marrom vista na figura 3.16 é um calço, que pode ser de madeira ou alumínio. Essa peça será feita depois de pronto o suporte, para que se possa fazer um melhor ajuste e alinhamento entre os eixos do motor e do guia linear. A figura 3.16 também mostra mais uma vista do projeto da peça, feito em SolidWorks.

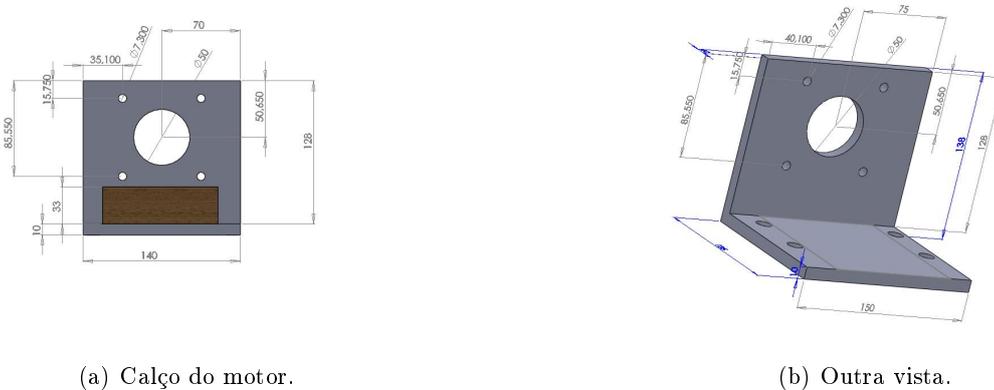


Figura 3.16: Projeto do suporte.

3.3.3 Acoplador

Para a conexão do motor ao guia linear, um acoplador é necessário.

O eixo do motor possui as seguintes medidas: diâmetro do eixo, 10 mm; diâmetro interno, 4 mm; largura da chaveta, 5mm; eixo do motor + chaveta, 18 mm.

Além de atender a essas especificações, o acoplador deve ser capaz de minimizar as folgas e compensar desalinhamentos, que podem ser problemas na montagem.

O modelo Rotex GS backlash-free (figura 3.17) foi o escolhido, por se tratar de um acoplador que minimiza folgas e efeito de vibrações, e ser de fácil montagem.

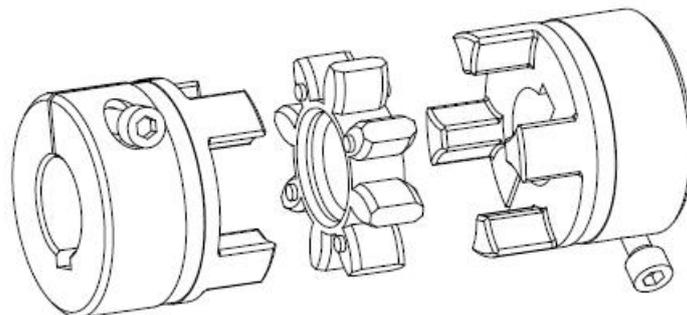


Figura 3.17: Desenho do acoplador motor-guia.

3.4 Montagem Elétrica

Primeiramente, ligou-se o CLP ao computador. Com o software RSLinx, é possível se visualizar tudo o que está conectado ao controlador. A figura 3.18 mostra que há três elementos conectados ao rack de componentes do controlador. O quadro da direita mostra esses elementos. O 00 é o próprio ControlLogix 5560, o 01 é o módulo da interface SERCOS e o 02 corresponde ao módulo da rede DeviceNet.

Para a utilização do RSLinx, é necessária a configuração do driver de comunicação. A aba 'Communications' possui a opção 'Configure Drivers'. Seleciona-se o 'RS-232 DF1 devices' e clica-se em 'Add new...' (figura 3.19).

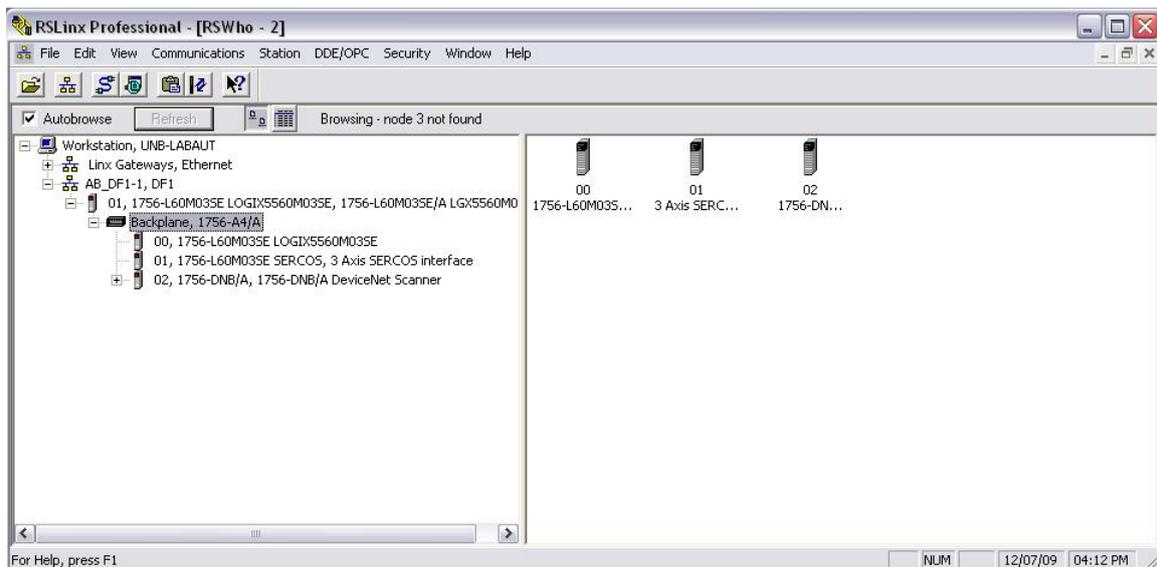


Figura 3.18: RSLinx.

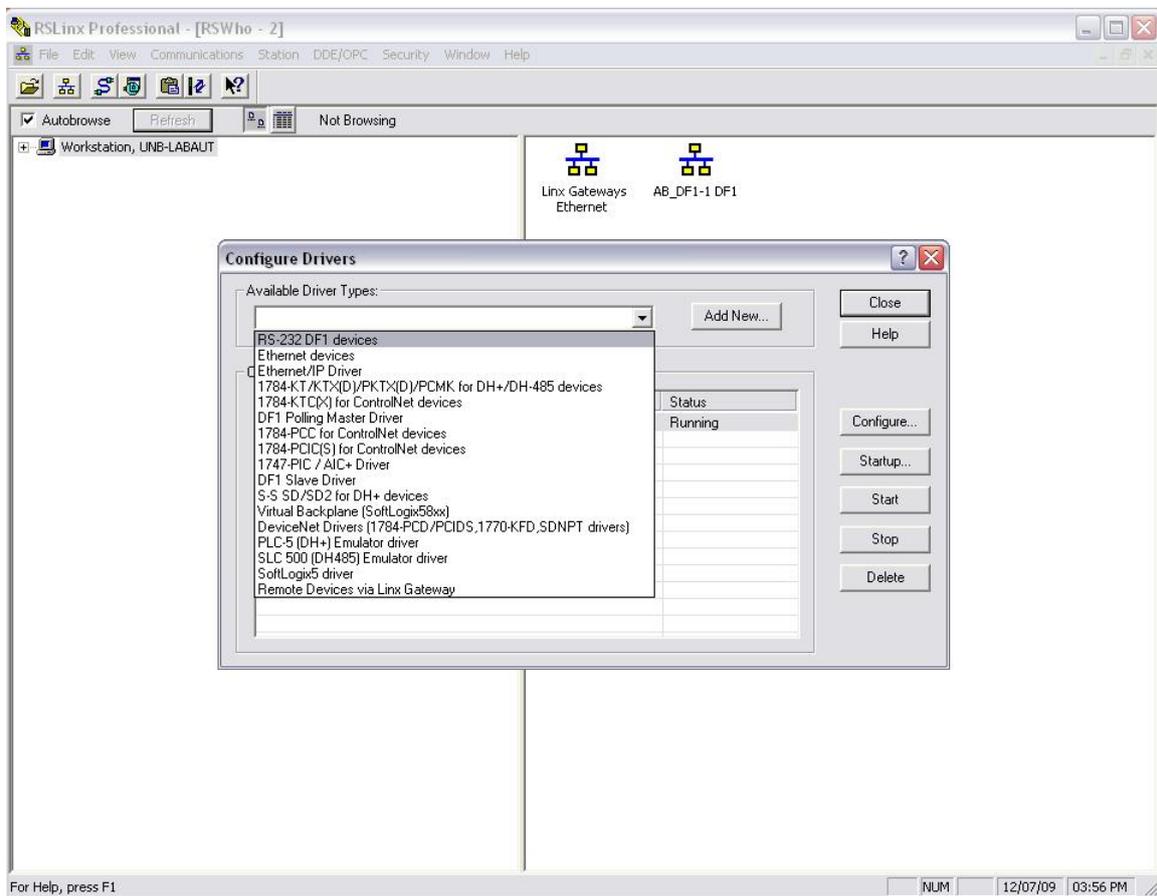


Figura 3.19: Driver do RSLinx.

Os seis sensores indutivos foram conectados à rede, conforme explicado na seção 2.5.1. Além do scanner DeviceNet - módulo que fica acoplado ao lado do ControlLogix - também é necessária uma fonte de alimentação externa de 24V. Essa fonte é que alimenta o barramento DeviceNet (figura

3.20).

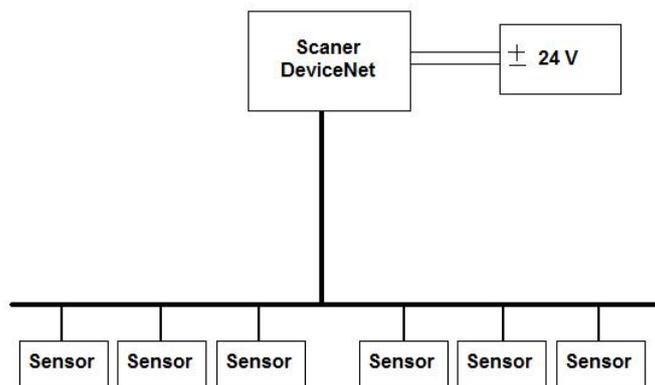


Figura 3.20: Esquema do barramento DeviceNet.

As ligações do Kinetix serão feitas com a ajuda de um LIM - Line Interface Module. Ele engloba várias ligações necessárias ao funcionamento do inversor, como filtros, transformadores e isolamentos. Essa ligação pode ser vista na figura 3.21.

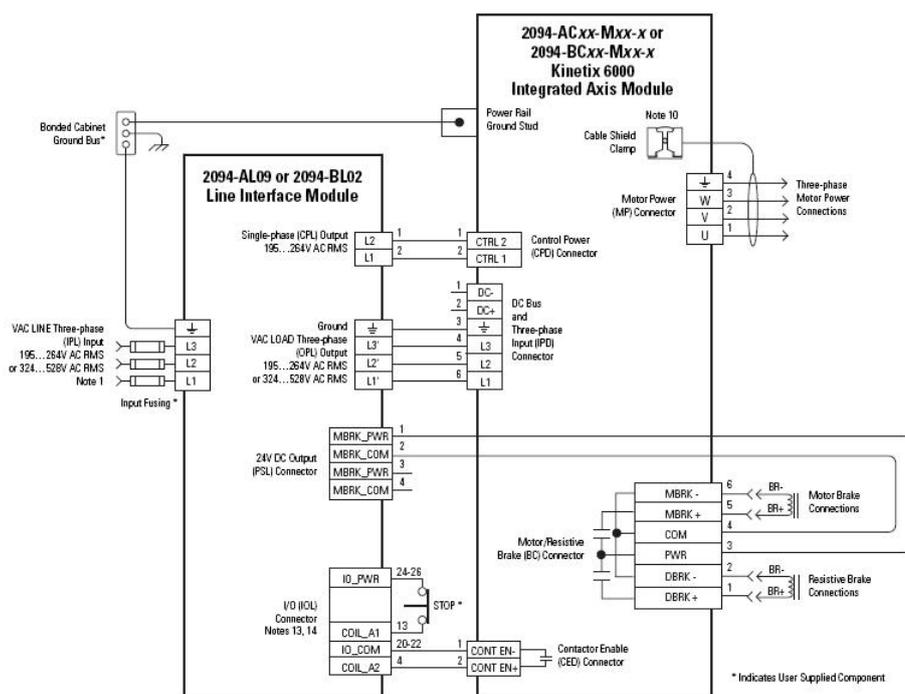


Figura 3.21: Ligações Kinetix-LIM.

Os quatro pontos na parte superior direita do Kinetix são as ligações de alimentação do motor.

Como mostrado na figura 3.8, a entrada 'Motor feedback' recebe a realimentação do motor, a 'Auxiliary feedback' recebe a do encoder da haste. Já a entrada 'I/O' possui um conector específico - 2090-K6ck-d26m -, figura 3.22.

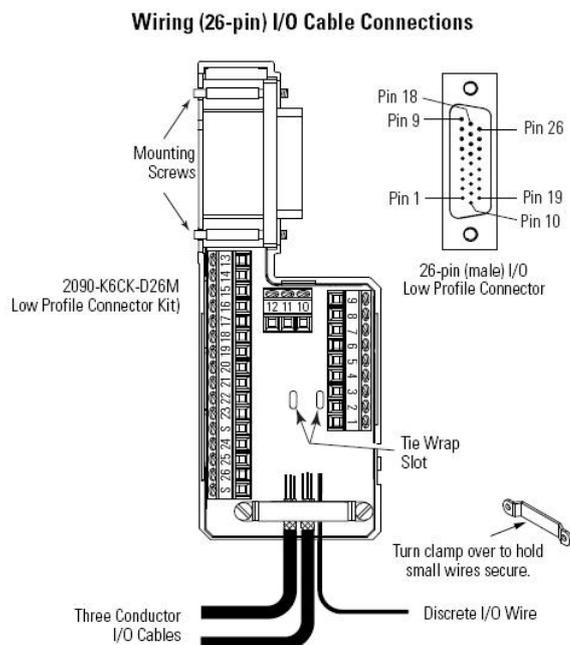


Figura 3.22: Conector da entrada I/O do Kinetix.

Os pinos dessa entrada possuem aplicação genérica. Pode ser usado, por exemplo, para o acionamento do eletroímã.

3.5 Programação

Os dados para a programação do controle do sistema vêm dos sensores e dos encoders.

O primeiro passo para a utilização dos sensores da rede DeviceNet é a atribuição de um endereço para cada um. O software que faz a manipulação dessa rede é o RSNetWorx. A figura 3.23 mostra esse software já com os sensores endereçados.

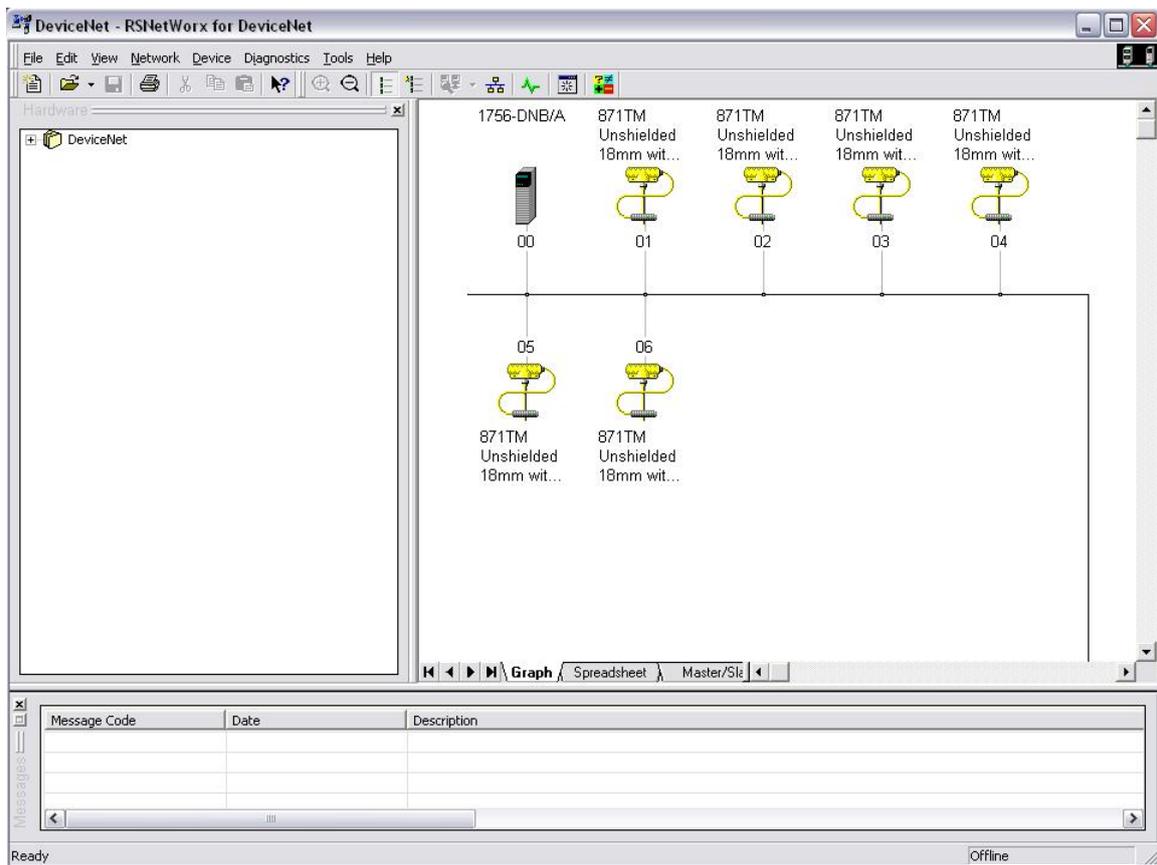


Figura 3.23: RSNetWorx - rede com sensores indutivos.

O endereço 00 é sempre definido como o endereço do módulo de comunicação da rede - que fica acoplado ao lado do controlador. Os seis sensores foram enumerados de 01 a 06. Para modificar o número de endereço de um componente, basta seguir este procedimento: clica-se no dispositivo desejado e a tela da figura 3.24 aparecerá.

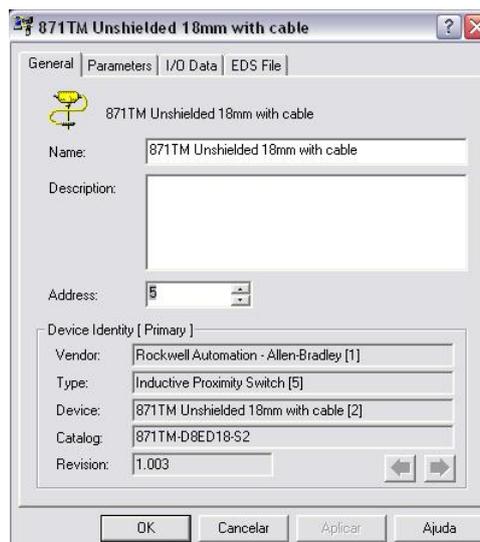


Figura 3.24: Modificando o endereço do sensor.

Coloca-se o número desejado, no campo 'Address' e clica-se em 'Aplicar'. Todo dispositivo DeviceNet possui endereço original 63.

Agora o sensor possui o endereço correto, mas somente no programa - RSNNetWorx. É necessário transferir essas novas informações a toda a rede, mas, antes disso, o módulo DeviceNet deve reconhecer esses sensores. Para isso, seleciona-se o módulo - no endereço 00. A aba 'Scanlist' possui todos os dispositivos conectados à rede. Como todos os sensores são elementos de entrada para a rede, todos eles devem estar listados na aba 'Input' (figura 3.25).

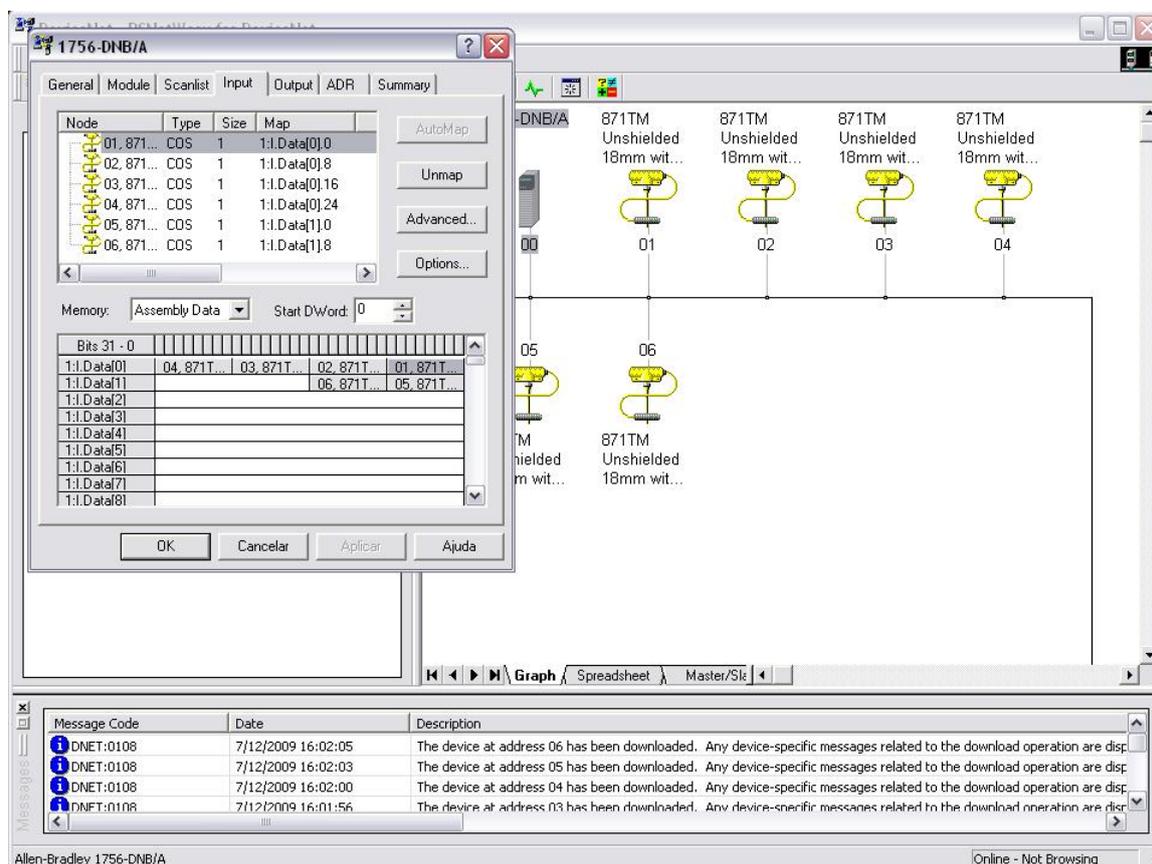


Figura 3.25: RSNNetWorx - Input.

Agora, as informações podem ser passadas do software para a rede. Para isso, clica-se em 'Device' e seleciona-se 'Download to Device' (figura 3.26).

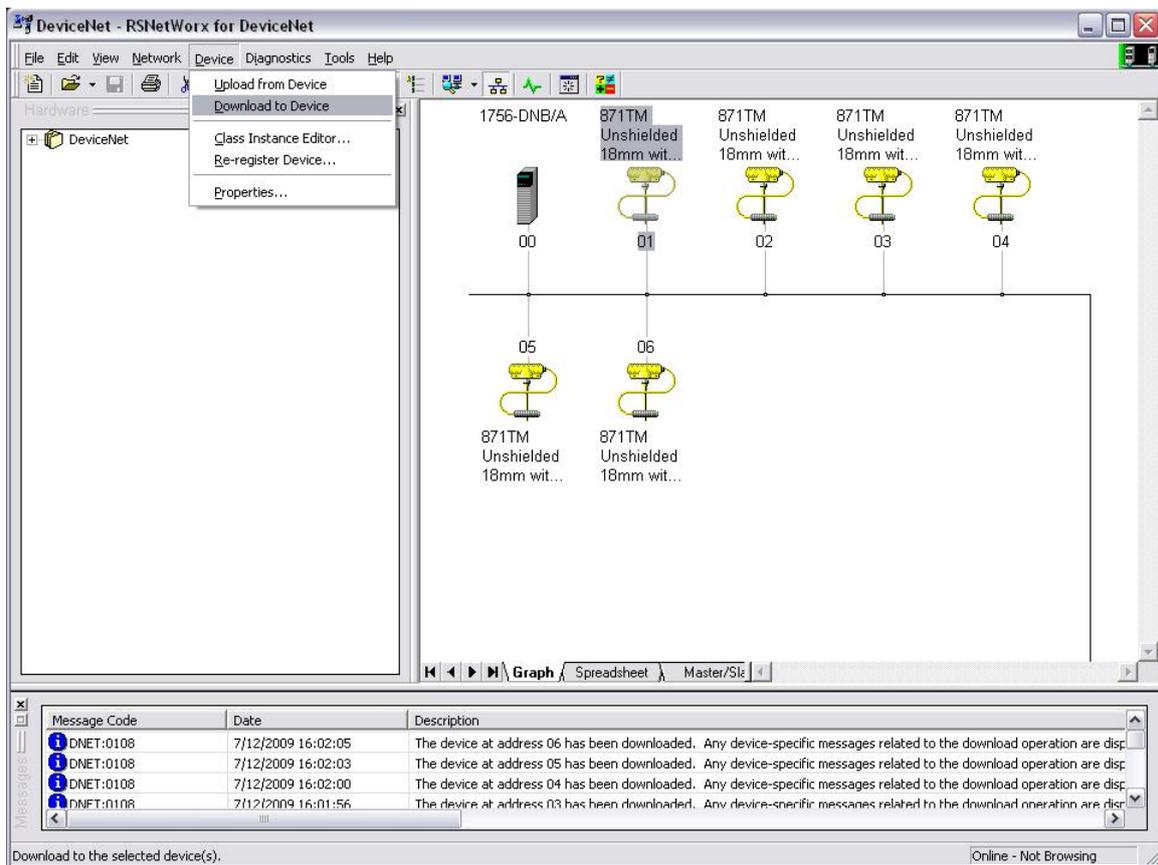


Figura 3.26: Download to Device.

Nesse ponto, a rede está pronta para ser utilizada. Basta clicar em 'Network' e selecionar 'Online'.

Os sensores podem, então, ser vistos pelo RSLinx (figura 3.27).

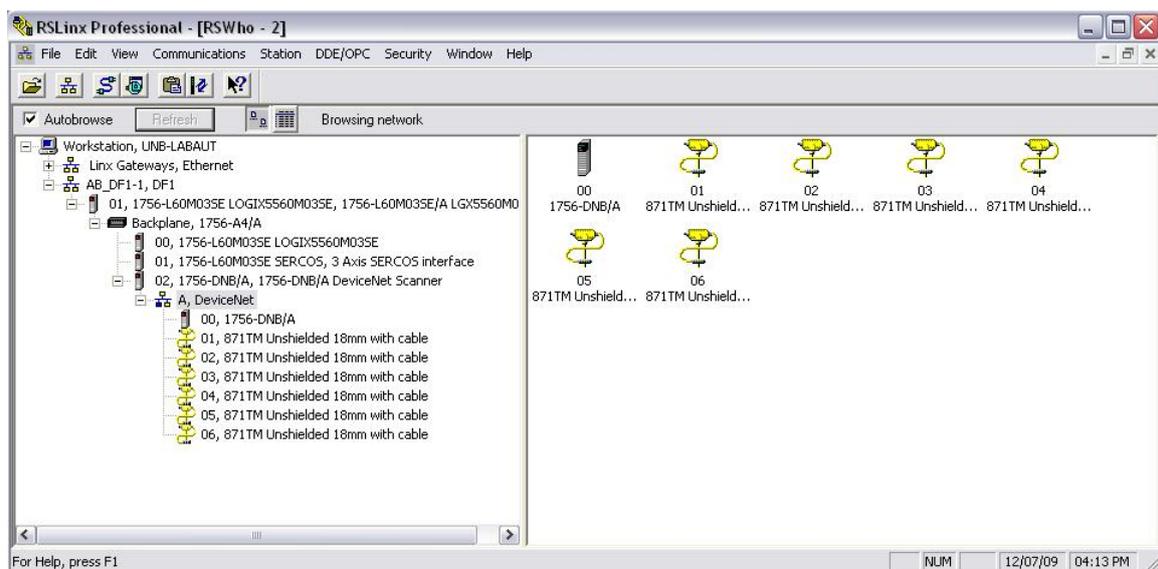


Figura 3.27: RSLinx com a rede DeviceNet configurada.

Para a utilização do software RSLogix, é necessária, novamente, a definição do caminho de comunicação. A figura 3.28 mostra o local onde se deve seleccionar 'AB-DF1-1*'.

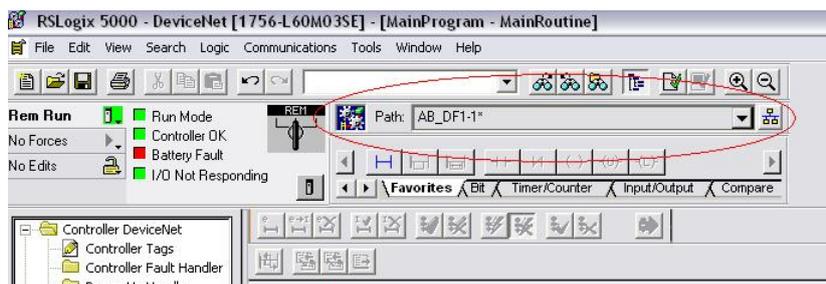


Figura 3.28: RSLogix - Path.

Já com a rede DeviceNet em funcionamento, pode-se ver, na scanlist do módulo, quais os endereços que contêm um dispositivo. Os seis sensores indutivos utilizados no projeto possuem endereços definidos de 01 a 06, como pode ser visto na figura 3.29.

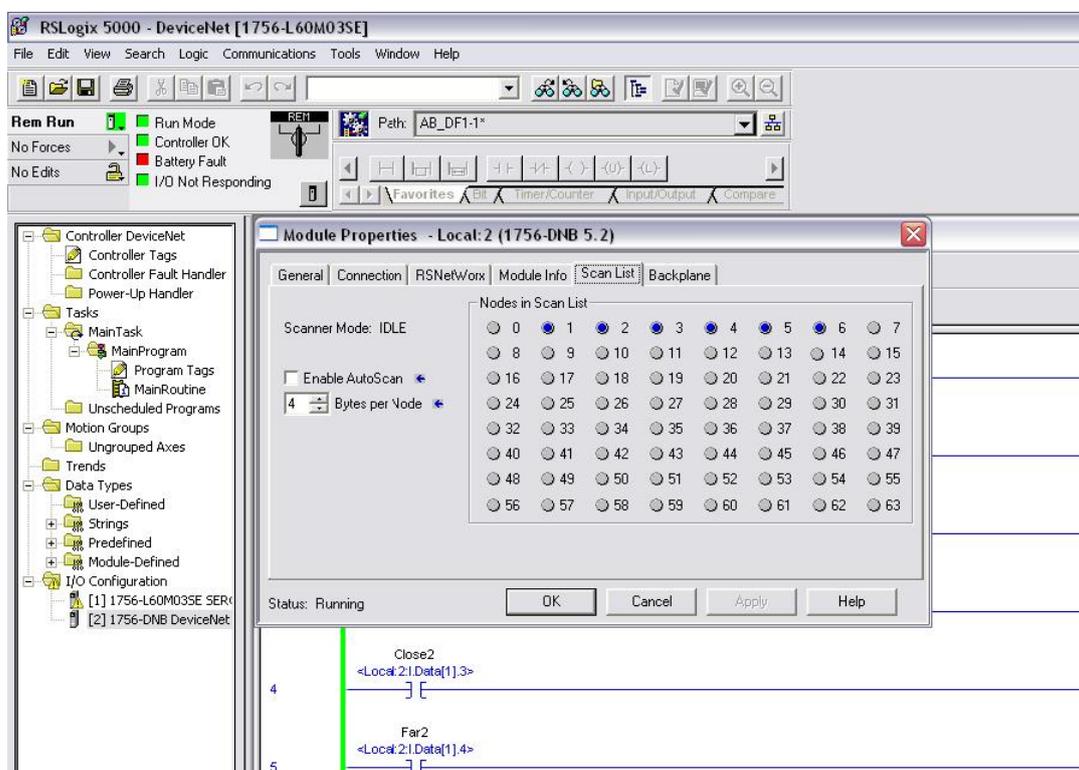


Figura 3.29: Scan List - DeviceNet.

Os sensores indutivos utilizados no projeto possuem oito bits de informações relativas à detecção de objetos ferromagnéticos. Utilizam-se três bits de maior relevância. O bit 0 mostra a presença ou ausência de material. O bit 3 detecta se um objeto esta muito próximo ao sensor. O bit 4 testa se a peça está no limiar de detecção do sensor.

O bit 0 - que detecta presença - é o bit que se deve utilizar na lógica do programa. Os bits 3 e 4 - 'To Close' e 'To Far', respectivamente - servem para detectar possíveis erros de colocação

das peças nos buffers ou de posicionamento dos sensores. Caso um desses dois bits - 'Too Close' ou 'Too Far' - tenha valor lógico 1, a movimentação deve ser parada, pois, devido a diferenças de posições esperadas para as peças, pode haver choques entre a haste e a peça, ou até mesmo com um sensor (no caso de ser este que esteja mal posicionado).

Como se pode observar na figura 3.1, a haste não encolhe. Sendo assim, ela somente pode carregar uma peça se não houver nenhuma outra em seu caminho. Caso contrário, haverá um choque entre as peças. A peça da haste pode cair, e a que está no buffer pode ficar deslocada ou cair também.

Como o depósito de saída fica no meio da estrutura, quanto mais perto do centro, maior a prioridade da peça. Além disso, arbitrou-se uma prioridade maior para as peças do buffer 1 (lado esquerdo). A numeração mostrada na figura 3.30 mostra que as peças de menores números possuem maiores prioridades.



Figura 3.30: Numeração para priorização.

O programa mostrado na figura 3.31 foi feito para se respeitar esse critério de preferência na ordem de coleta das peças.

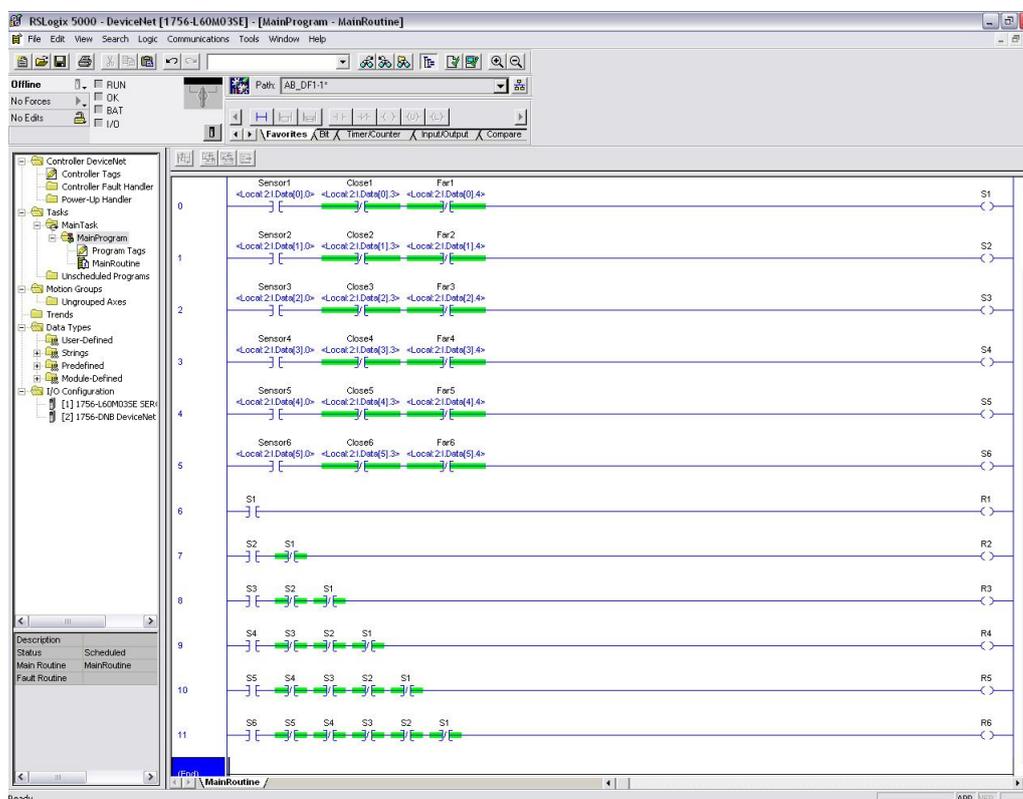


Figura 3.31: Ladder para os sensores indutivos.

As seis primeiras linhas fazem um AND lógico com os três bits citados para cada sensor. Se o sensor detecta (Sensor1 = 1) e a peça não está perto demais (Close1 = 0) nem longe demais (Far1 = 0), a respectiva saída (S1) é acionada. Essas seis saídas - S1, S2, S3, S4, S5 e S6 - são utilizadas para se dar prioridade para cada buffer de peça.

As seis linhas de baixo fazem a lógica dessa priorização. A peça 1 pode ser sempre alcançada; a 2, somente se não houver a peça 1; a 3, só sem as peças 1 e 2. E assim, até chegar à peça 6.

A figura 3.32 mostra todas as variáveis - tanto de entrada como de saída - que foram utilizadas nesse programa.

Tag Name	Value	Force Mask	Style	Type	Description
Close1	0	0	Decimal	BOOL	
Close2	0	0	Decimal	BOOL	
Close3	0	0	Decimal	BOOL	
Close4	0	0	Decimal	BOOL	
Close5	0	0	Decimal	BOOL	
Close6	0	0	Decimal	BOOL	
Far1	0	0	Decimal	BOOL	
Far2	0	0	Decimal	BOOL	
Far3	0	0	Decimal	BOOL	
Far4	0	0	Decimal	BOOL	
Far5	0	0	Decimal	BOOL	
Far6	0	0	Decimal	BOOL	
Liga	0	0	Decimal	BOOL	
R1	0	0	Decimal	BOOL	
R2	0	0	Decimal	BOOL	
R3	0	0	Decimal	BOOL	
R4	0	0	Decimal	BOOL	
R5	0	0	Decimal	BOOL	
R6	0	0	Decimal	BOOL	
S1	0	0	Decimal	BOOL	
S2	0	0	Decimal	BOOL	
S3	0	0	Decimal	BOOL	
S4	0	0	Decimal	BOOL	
S5	0	0	Decimal	BOOL	
S6	0	0	Decimal	BOOL	
Sensor1	0	0	Decimal	BOOL	
Sensor2	0	0	Decimal	BOOL	
Sensor3	0	0	Decimal	BOOL	
Sensor4	0	0	Decimal	BOOL	
Sensor5	0	0	Decimal	BOOL	
Sensor6	0	0	Decimal	BOOL	

Figura 3.32: Variáveis do ladder - 'Program Tags'.

Capítulo 4

Conclusões

Após uma motivação do projeto - simulação de uma planta de fundição -, definiu-se o problema e seus objetivos. Esse é um importante tema na área de engenharia, pois envolve a segurança no trabalho. Ambientes industriais podem oferecer riscos à saúde humana, e esse projeto visa um sistema que diminui a presença humana nesses lugares. Outro aspecto importante é que ele engloba vários conhecimentos de Engenharia Mecatrônica - controle, redes de comunicação, programação, sistemas discretos, construção mecânica.

Para esse projeto, vários equipamentos e tecnologias foram pesquisados: dispositivos de controle - CLP's, PAC's e seus softwares; sensores - indutivos, fotoelétricos e encoders ; equipamentos para acionamento - motor de indução e inversor de frequência; tecnologias de comunicação - rede DeviceNet e interface SERCOS.

Mostrou-se a interligação que esses equipamentos têm uns com os outros e a utilidade de cada um para o projeto da planta de controle.

Apresentou-se o projeto mecânico do sistema - estrutura de suporte, acoplador do motor com o guia-linear e a peça de suporte do motor. Fez-se a configuração dos sensores indutivos juntamente com a rede DeviceNet.

Os três softwares utilizados - RSLogix 5000, RSLinx e RSNetWorx - foram configurados para interagir com esse sistema. Iniciou-se a programação ladder no RSLogix 5000. A estrutura em ladder é bem simples e de fácil visualização, o que agiliza a programação.

A utilização de sensores prontos para uso com a rede DeviceNet facilitou bastante o trabalho, tanto na montagem elétrica do sistema, quanto em sua configuração. Além de possuírem a saída digital - para detecção da peça -, os sensores também possuem outras informações importantes, que podem ser passadas para o controlador.

Para a continuação desse trabalho, é necessária a montagem do inversor - Kinetix - e do motor. Depois disso, o controle de posição do carro deve ser feito, para que se possa posicionar corretamente a haste com o eletroímã sobre os sensores. O controle do motor pode ser feito com um bloco de ladder do RSLogix 5000 que representa um controlador PID.

Com esse controle feito, analisa-se o sistema todo para definir a movimentação do motor e o

acionamento do eletroímã. Um autômato ou uma rede de Petri pode ser feito para a modelagem do sistema.

Em princípio, essa planta de automação serve para a simulação de uma grua, mas também pode ser utilizada para experimentos do tipo "Pêndulo invertido", "Problema de Swing-up" e outros.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] SANTOS, M. *Fabricação Mecânica*. São Carlos, SP, 2009.
- [2] AUTOMATION, R. *Controladores Logix5000*. 1^a. ed. Milwaukee,WI, 2005.
- [3] WILSON, J. S. *Sensor Technology Handbook*. 1^a. ed. Burlington,MA: Elsevier, 2005.
- [4] FRADEN, J. *HANDBOOK OF MODERN SENSORS Physics, designs and Applications*. 3^a. ed. San Diego,CA: Springer, 2004.
- [5] ROCKWELL AUTOMATION - ALLEN BRADLEY. *PHOTOSWITCH Photoelectric Sensors*. 1^a. ed. Milwaukee,WI.
- [6] RIPKA, A. T. P. *Modern Sensors Handbook*. 1^a. ed. Newport Beach,CA: ISTE Ltd., 2007.
- [7] WEBSTER, J. G. *Measurement, Instrumentation and Sensors Handbook CRCnetBASE*. 1^a. ed. Boca Raton,FL: CRC Press, 1999.
- [8] CONTROLS, D. I. *Encoder Application Handbook*. 1^a. ed. Gurnee, IL, 2003.
- [9] INC., S. *Motor Feedback Systems*. 1^a. ed. Dayton, OH, 2009.
- [10] KOSOW, I. L. *Maquinas Eletricas e Transformadores*. 3^a. ed. Englewood Cliffs,NJ: Globo, 1982.
- [11] RASHID, M. H. *Power Electronics Handbook*. 1^a. ed. San Diego,CA: Academic Press, 2001.
- [12] AUTOMATION, R. *Kinetix 6000 Multi-axis Servo Drives*. 1^a. ed. Milwaukee,WI, 2008.
- [13] OGASHAWARA, O. *Tópicos em mecatrônica*. São Carlos, SP, 2009.
- [14] PRATICAL Fieldbus, DeviceNet & Ethernet for Industry. 2004.
- [15] DEVICENET Technical Overview. 2004.
- [16] ZURAWSKI, R. *Industrial Communications Technology Handbook*. 1^a. ed. Florida: CRC Press, 2000.
- [17] AUTOMATION, R. *Encoders*. 1^a. ed. Milwaukee,WI, 2008.
- [18] AUTOMATION, R. *Feedback Cables*. 1^a. ed. Milwaukee,WI, 2008.

ANEXOS

I. DIAGRAMAS ESQUEMÁTICOS

I.1 Materiais do Projeto

Tabela I.1: Materiais da remessa original

Nome do Equipamento	Número do Modelo	Qtd.	Função
ControlLogix	5560	1	Controlador e módulo de comunicação SERCOS
Módulo DeviceNet	1756 DNB	1	Módulo de comunicação DeviceNet
Sensor Indutivo	871TM-D8ED18-S2	6	Deteção da presença de peças a serem coletadas
Sensor Fotoelétrico	42CF-D1LNA1-A2	4	Sensores de fim de Curso
Encoder Externo	845H-SJDN14CSY2	1	Deteção da posição da haste
Motor de Indução	MPL-A310F-SJ22AA	1	Movimentação da Guia Linear
Kinetix	2094-AC05-MP5-S	1	Acionamento do Motor
Low Profile Conector (I/O)	2090-K6CK-D26M	1	Ligação dos fios na entrada I/O do Kinetix
Low Profile Conector (Motor Feedback)	2090-K6CK-D15M	1	Feedback do motor
Low Profile Conector (Auxiliar Feedback)	2090-K6CK-D15F	1	Feedback do encoder externo
Cabo de Feedback	2090-XXNFMP-S09	1	Feedback do motor
Cabo de Energia	2090-XXNTP-16Sxx	1	Alimentação do Motor
Contatora trifásica	100-C43	1	Servirá de botão de parada emergencial

I.1.1 Materiais Faltantes

Tabela I.2: Materiais que não foram enviados na remessa original. Encomendados

Nome do Equipamento	Número do Modelo	Qtd.	Função
Cabo para encoder 10 pinos	845-CA-C-10	1	Cabo para encoder externo
Power Rail para 1 IAM	2094 -PRS1	1	Trilho de suporte, alimentação e comunicação do Kinetix
Line Interface Module (LIM)	2094-AL09	1	Fonte para o Kinetix
Low Profile Conector(LIM)	2090-K6CK-D26M	1	Conector para entrada de 26 pinos do LIM
Acoplador para encoder	845-FC-B-B	1	Acoplador para ligar a haste ao encoder externo

I.2 Esquemáticos de Ligação

I.2.1 Ligações Kinetix

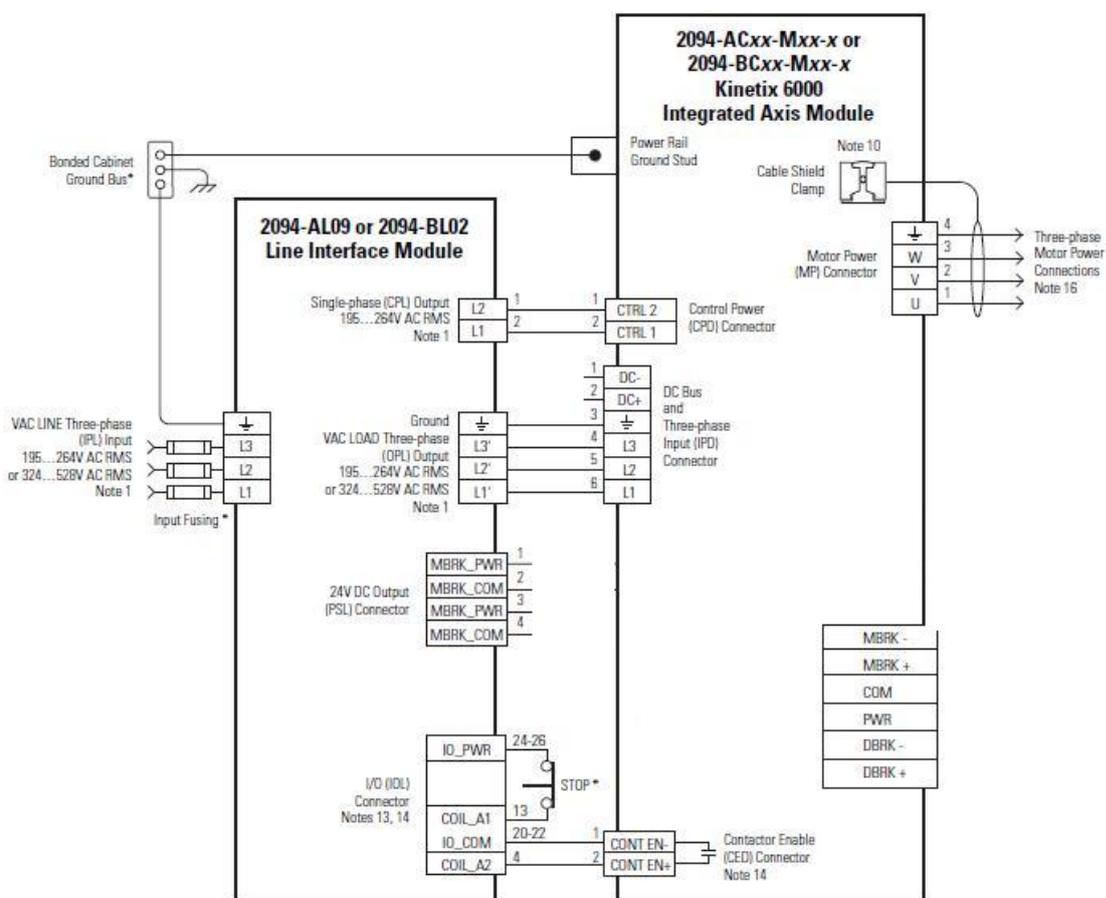


Figura I.1: Esquemático de ligação do Kinetix com o LIM

A figura I.1 é uma adaptação do manual [12], mostrando o esquema de ligação do Kinetix. O LIM recebe a alimentação trifásica diretamente da tomada (que se encontra no teto do laboratório). Uma vez alimentado, o LIM fornece toda as ligações para o Kinetix diretamente. O freio resistivo não será usado uma vez que o motor do experimento não possui essa funcionalidade. Os bornes DC bus do Kinetix não são usados, porque não será usado alimentação DC em motores. A chave STOP não é obrigatória, mas serve de botão de parada uma vez que a alimentação do Contactor Enable é obrigatória e a não alimentação do mesmo causa a parada de alimentação ao motor (mantendo o Kinetix ligado). A figura I.2[12] mostra como são ligadas as fibra óptias pra formar o anel descrito no capítulo 2.5.2.

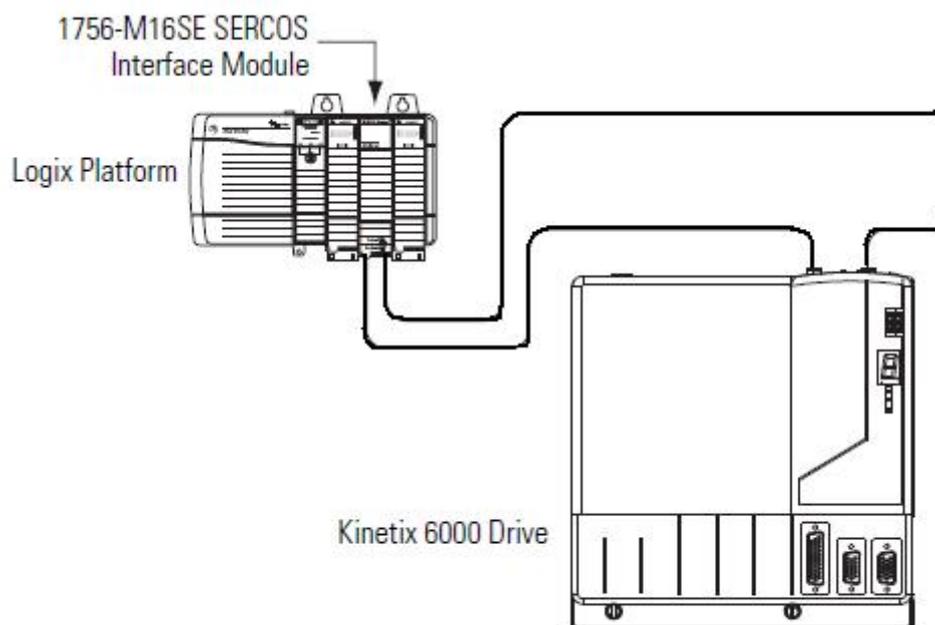


Figura I.2: Ligação das fibras ópticas entre o Kinetix e o controlador

I.2.2 Identificação da pinagem e dos cabos

Os cabos ligados no Kinetix se conectam com o uso do Low Profile Conector correspondente. Cada um desses conectores consiste num invólucro com um circuito que conecta diversas borneiras ao pino correspondente no conector. Os fios devem ser ligados no borne correspondente ao pino da função correspondente. A seguir uma descrição dos cabos e pinos do Kinetix e uma imagem (I.3 [12]) de um Low Profile Conector aberto. A coluna Liga a... mostra qual fio de qual cabo deve ser ligado no conector, sendo que a descrição vale para o cabo e pino. Apenas os pinos com previsão de ligação estão descritos, mais detalhes podem ser obtidos nos manuais correspondetes no CD.

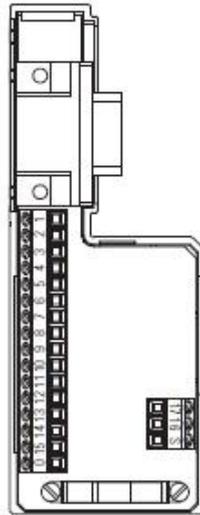


Figura I.3: Exemplo de um Low Profile Conector (Auxiliary Feedback)

Tabela I.3: Pinos da entrada I/O do Kinetix.

#	Identificação [12]	Liga a...
7	Fonte 24VDC	o fio marrom do sensor fotoelétrico
8	Fim de curso positivo	o fio preto do sensor fotoelétrico
9	Comum	o fio azul do sensor fotoelétrico
10	Fonte 24VDC	o fio marrom do sensor fotoelétrico
11	Fim de curso positivo	o fio preto do sensor fotoelétrico
12	Comum	o fio azul do sensor fotoelétrico

Cada entrada deve estar ligada ao receptor do sensor fotoelétrico em cada um dos lados (positivo e negativo). O fio branco deve ser ligado junto ao marrom para LO e junto ao preto para DO. Veja a seção sobre o sensor fotoelétrico (2.3.2) ou o manual no CD para mais informações.

Tabela I.4: Pinos da entrada Auxiliary Feedback do Kinetix. (todos os fios estão no cabo 845-CA-C-10 a chegar)

#	Identificação [12]	Liga a...[17]
1	a+/sin+	fio branco
2	a-/sin-	fio preto enrolado no fio branco
3	b+/cos+	fio azul
4	b-/cos-	fio preto enrolado no fio azul
5	I+	o fio verde
6	Comum	fio preto, enrolado no fio vermelho
10	I-	fio preto enrolado no fio verde
14	+5 VDC	fio vermelho

Para cada fio, há um preto correspondente, que vem enrolado com o fio com o qual ele faz

par. Uma vez desenrolados os fios, não será mais possível distinguir visualmente quais fios pretos correspondem a qual sinal, então desenrole os fios somente se eles forem ligados em seguida.

Tabela I.5: Pinos da entrada Motor Feedback do Kinetix (todos os fios estão no cabo 2090-XXNFMP-S09).

#	Identificação [12]	Liga a...[18]
1	a+/sin+	fio preto
2	a-/sin-	fio branco e preto
3	b+/cos+	fio vermelho
4	b-/cos-	fio branco e vermelho
5	Data+	o fio verde
6	Comum	fio branco e cinza
10	Data-	fio branco e verde
11	Thermal Switch	fio branco e laranja
14	+5 VDC	fio cinza

II. DESCRIÇÃO DO CONTEÚDO DO CD

O CD contém materiais úteis à continuação do projeto da grua Rockwell Automation. O conteúdo foi organizado em pastas de acordo com a descrição abaixo

- Manuais: Manuais dos equipamentos divididos de acordo com a função
 - ◇ Sensores: manuais sobre os sensores fotoelétricos, os sensores indutivos e o encoder externo
 - ◇ Kinetix e motor: manual de montagem do motor, cabos e Kinetix. Além disso, inclui manuais de acessórios como o LIM e Power Rail
 - ◇ Fonte, Chassis e Módulos: manuais da fonte e chassis do ControlLogix
 - ◇ Programação: Manual da descrição da programação, arquivo índice de manuais de assuntos específicos, manual de instruções de instruções genéricas, manual de instruções de movimentação e programação com DeviceNet
 - ◇ Comunicação: DeviceNet e SERCOS
- Montagem: desenhos em CAD da parte mecânica
 - ◇ arquivos do SolidWorks 2009 com o projeto da bancada e do suporte do motor
 - ◇ documentos sobre o acoplador motor-guia
 - ◇ fiação necessária para conclusão das ligações elétricas (.xls)
- Programas: programas de auxílio aos 3 programas (RSLogix, RSLinx e RSNetWorx) e rotina de leitura dos sensores DeviceNet
- Apresentação: Imagens, este trabalho e a apresentação em Power Point