

UNIVERSIDADE DE TAUBATÉ – UNITAU

Donizeti Bísaro

**PROPOSTA DE ARQUITETURA PARA RECONFIGURAR
TAREFAS EM CÉLULA FLEXÍVEL DE PRODUÇÃO**

Taubaté - SP

2011

Donizeti Bísvaro

**PROPOSTA DE ARQUITETURA PARA RECONFIGURAR
TAREFAS EM CÉLULA FLEXÍVEL DE PRODUÇÃO**

Dissertação apresentada para obtenção do Título de Mestre pelo Curso de Pós-Graduação do Departamento de Engenharia Mecânica da Universidade de Taubaté.

Área de Concentração: Automação.

Orientador: Prof. Dr. Francisco Carlos Parquet Bizarria.

Taubaté - SP

2011

Sistema Integrado de Bibliotecas – SIBi/UNITAU
Biblioteca Setorial de Engenharia Mecânica

B621p Bísvaro, Donizeti
Proposta de arquitetura para reconfigurar tarefas em célula flexível de produção / Donizeti Bísvaro. – 2011.
84f. : il.

Dissertação (Mestrado em Engenharia Mecânica) –
Programa de Pós-graduação em Engenharia
Mecânica/Produção, Departamento de Engenharia Mecânica,
Universidade de Taubaté, 2011.

Orientação: Prof. Dr. Francisco Carlos Parquet Bizarria,
Departamento de Engenharia Mecânica.

1. Interface gráfica. 2. Célula flexível de produção. 3.
Automatização de sistema. I. Título.

Dedico este trabalho à minha família por ter me ensinado a ser perseverante e por sempre apoiar e admirar meu esforço.

AGRADECIMENTOS

Primeiramente ao criador de todas as coisas, sem o qual sequer existiríamos. À minha esposa Agnes e minhas filhas Gabriela e Natália, pelo incentivo, paciência e tolerância às privações passadas em todos os sentidos durante a elaboração deste trabalho. A minha mãe, Maria pela dedicação e confiança e a meu pai, Osvaldo pelo exemplo. A meu irmão, Wilson pelas conversas informais e convivência.

Agradeço também ao SENAI, na pessoa do Professor José Rogério Chavier por ter apoiado meu trabalho e cedido as instalações onde foram realizados o desenvolvimento e implementação do protótipo da arquitetura proposta.

De maneira muito especial agradeço ao Professor Dr. Francisco Carlos Parquet Bizarria pela orientação e apoio durante a confecção deste trabalho e principalmente pela crença e pelo cumprimento de sua missão de educador.

- 1 Ouvi, filhos, a instrução do pai, e estai atentos para conhecerdes o entendimento.
- 2 Pois eu vos dou boa doutrina; não abandoneis o meu ensino.
- 3 Quando eu era filho aos pés de meu, pai, tenro e único em estima diante de minha mãe,
- 4 ele me ensinava, e me dizia: Retenha o teu coração as minhas palavras; guarda os meus mandamentos, e vive.
- 5 Adquire a sabedoria, adquire o entendimento; não te esqueças nem te desvies das palavras da minha boca.
- 6 Não a abandones, e ela te guardará; ama-a, e ela te preservará.
- 7 A sabedoria é a coisa principal; adquire, pois, a sabedoria; sim, com tudo o que possuis adquire o entendimento.
- 8 Estima-a, e ela te exaltará; se a abraçares, ela te honrará.
- 9 Ela dará à tua cabeça uma grinalda de graça; e uma coroa de glória te entregará.
- 10 Ouve, filho meu, e aceita as minhas palavras, para que se multipliquem os anos da tua vida.
- 11 Eu te ensinei o caminho da sabedoria; guiei-te pelas veredas da retidão.
- 12 Quando andares, não se embaraçarão os teus passos; e se correres, não tropeçarás.
- 13 Apega-te à instrução e não a largues; guarda-a, porque ela é a tua vida. (Salomão no livro de Provérbios, capítulo 4).

RESUMO

Este trabalho apresenta proposta de arquitetura orientada para um sistema que utiliza a interface gráfica como elemento principal para reconfigurar a seqüência de tarefas que podem ser executadas por uma célula flexível de produção. É dada especial atenção na elaboração dos componentes dessa interface, pois as corretas definições e operações desses componentes são condições determinantes para se alcançar êxito no uso do sistema. A validação da funcionalidade dessa interface é obtida por meio da realização de testes em um protótipo, que adota os elementos básicos previstos na mencionada arquitetura. Os resultados positivos observados nesses testes indicam que o sistema proposto é adequado para a finalidade a qual se destina.

Palavras-Chave: Interface Gráfica; Célula Flexível de Produção; Automatização de Sistema.

ABSTRACT

This work presents a proposal for a system-oriented architecture that uses the graphical interface as the main element to reconfigure the sequence of tasks that can be performed by a flexible production cell. Particular attention is given in the preparation of the components of this interface, because the correct settings and operations of these components are critical to achieving success in the use of the system. Validation of the functionality of this interface is obtained by means of conducting tests on a prototype, which adopts the basic elements contained in the mentioned architecture. The positive results observed in these tests indicate that the proposed system is suitable for the purpose for which it is intended.

Key words: Graphic Interface; Flexible Production Cell; System Automatization.

LISTA DE FIGURAS

FIGURA 1 - EXEMPLO DE JANELA PARA SUPERVISÃO DE PROCESSO	20
FIGURA 2 - DIAGRAMA EM BLOCOS DE UM CLP	23
FIGURA 3 - MODELOS DE CONTROLADORES	25
FIGURA 4 - FLUXOGRAMA DO MODO DE OPERAÇÃO DE UM CLP	25
FIGURA 5 - LINHAS DE UM PROGRAMA EM LINGUAGEM LADDER	27
FIGURA 6 - INSTRUÇÕES BÁSICAS DE ENTRADA EM LADDER.....	28
FIGURA 7 - INSTRUÇÃO BÁSICA DE SAÍDA EM LADDER.....	28
FIGURA 8 - DIFERENÇAS FÍSICAS ENTRE CLP E CUBLOC	30
FIGURA 9 - EXECUÇÃO EM MONOTAREFA E MULTITAREFA	31
FIGURA 10 - JANELA PRINCIPAL DO CUBLOC STUDIO.....	32
FIGURA 11 - ROBÔ IRB 1410	34
FIGURA 12 - CONTROLADOR DO ROBÔ IRB 1400	35
FIGURA 13 - OPTOACOPLADOR TÍPICO	40
FIGURA 14 - DIAGRAMA DE UM SENSOR INDUTIVO	41
FIGURA 15 - SENSORES CLASSIFICADOS CONFORME A SAÍDA	41
FIGURA 16 - SENSOR INDUTIVO COM SISTEMA DE FIXAÇÃO	41
FIGURA 17 – MAPA MENTAL DA ARQUITETURA LÓGICA	47
FIGURA 18 - ARQUITETURA PROPOSTA.....	48
FIGURA 19 - VISTA DO PROTÓTIPO	50
FIGURA 20 – ESQUEMA DE LIGAÇÕES DO PROTÓTIPO	51
FIGURA 21 - UNIDADE DE ACIONAMENTO E SUPERVISÃO	52
FIGURA 22 - INTERFACE COM ACOPLADORES ÓPTICOS (OPT).....	53
FIGURA 23 - CIRCUITO ELETRÔNICO UTILIZADO PARA O MÓDULO OPT.....	53
FIGURA 24 - FLUXOGRAMA MODO AUTOMÁTICO	57
FIGURA 25 - FLUXOGRAMA MODO ASSISTIDO	58
FIGURA 26 - JANELA PARA REGISTRO DE SENHA E USUÁRIO.....	59
FIGURA 27 - JANELA DE MODO ASSISTIDO DA INTERFACE GRÁFICA.....	60
FIGURA 28 – JANELA DE MODO ASSISTIDO DESLIGADO.....	61
FIGURA 29 – JANELA MODO AUTOMÁTICO DESATIVADO	62
FIGURA 30 - JANELA MODO AUTOMÁTICO ATIVADO	63
FIGURA 31 – SELEÇÃO DE EVENTOS EM MODO AUTOMÁTICO.....	63
FIGURA 32 – SELEÇÃO NA JANELA DE MODO AUTOMÁTICO	64
FIGURA 33 – OBJETO UTILIZADO PARA OS TESTES DO PROTÓTIPO	65
FIGURA 34 - INTERFACE COM A AÇÃO “REPOUSO” SELECIONADA.....	65
FIGURA 35 - ROBÔ NA POSIÇÃO DEFINIDA COMO “REPOUSO”	66
FIGURA 36 - INTERFACE COM A AÇÃO “GARRA ABERTA” SELECIONADA.....	66
FIGURA 37 - ROBÔ COM A “GARRA ABERTA”.....	67
FIGURA 38 - INTERFACE COM A AÇÃO “ALIMENTADOR” SELECIONADA.....	67
FIGURA 39 - ROBÔ EXECUTANDO A AÇÃO “ALIMENTADOR”	68
FIGURA 40 - INTERFACE COM A AÇÃO “SENSOR” SELECIONADA	68
FIGURA 41 - ROBÔ MOVENDO A PEÇA PARA A POSIÇÃO “SENSOR”	69
FIGURA 42 - INTERFACE COM O BOTÃO “BALANÇA” SELECIONADO.....	69
FIGURA 43 - ROBÔ COM A PEÇA NA POSIÇÃO “BALANÇA”	70
FIGURA 44 - INTERFACE COM O BOTÃO “ENTREGA” SELECIONADO	70
FIGURA 45 - ROBÔ COM A PEÇA NA POSIÇÃO “ENTREGA”	71
FIGURA 46 - INTERFACE COM O BOTÃO “POS 1A” SELECIONADO.....	71
FIGURA 47 - ROBÔ COM A PEÇA NA POSIÇÃO “POS 1A”	72

FIGURA 48 - INTERFACE COM O BOTÃO “POS 2A” SELECIONADO	72
FIGURA 49 - ROBÔ COM A PEÇA NA POSIÇÃO “POS 2A”	73
FIGURA 50 - INTERFACE COM O BOTÃO “POS 3A” SELECIONADO	73
FIGURA 51 - ROBÔ COM A PEÇA NA POSIÇÃO “POS 3A”	74
FIGURA 52 - INTERFACE COM O BOTÃO “POS 1B” SELECIONADO	74
FIGURA 53 - ROBÔ COM A PEÇA NA POSIÇÃO “POS 1B”	75
FIGURA 54 - INTERFACE COM O BOTÃO “POS 2B” SELECIONADO	75
FIGURA 55 - ROBÔ COM A PEÇA NA POSIÇÃO “POS 2B”	76
FIGURA 56 - INTERFACE COM O BOTÃO “POS 3B” SELECIONADO	76
FIGURA 57 - ROBÔ COM A PEÇA NA POSIÇÃO “POS 3B”	77
FIGURA 58 - INTERFACE COM SEQUÊNCIA PRÉ-SELECIONADA	77
FIGURA 59 – JANELA DE AJUDA.....	80

LISTA DE ABREVIATURAS E SIGLAS

BO	Bobina
CA	Corrente Alternada
CH	Computador Hospedeiro
CLP	Controlador Lógico Programável
EIA	<i>Electronic Industries Alliance</i>
IHM	Interface Homem-Máquina
LED	<i>Light Emitting Diode</i>
NA	Normalmente Aberto
NF	Normalmente Fechado
OSI	<i>Open Systems Interconnection</i>
PCB	<i>Printed Circuit Board</i>
RAM	<i>Random Access Memory</i>
ROM	<i>Read Only Memory</i>
RTU	<i>Remote Terminal Unit</i>
SCADA	<i>Supervisory, Control and Data Acquisition</i>
UCP	Unidade Central de Processamento
ULA	Unidade de Lógica e Aritmética
UC	Unidade de Controle

SUMÁRIO

CAPÍTULO 1 - INTRODUÇÃO	13
1.1 Descrição do Problema.....	13
1.2 Solução Proposta.....	14
1.3 Objetivos do Trabalho	15
1.4 Revisão Bibliográfica	15
1.5 Estrutura do Trabalho.....	17
CAPÍTULO 2 - CONCEITOS BÁSICOS	18
2.1 Sistema de Supervisão	18
2.2 Controlador Lógico Programável.....	21
2.3 Linguagem Ladder	27
2.4 Sistema CUBLOC	29
2.4.1 Características do Sistema CUBLOC	29
2.4.2 Ambiente de Desenvolvimento.....	31
2.5 Robô.....	32
2.6 Redes de Comunicação de Dados	36
2.6.1 Redes Determinísticas e Probabilísticas	36
2.6.2 Protocolos	38
2.7 Optoacopladores.....	39
2.8 Sensores Indutivos	40
2.9 Manufatura enxuta (<i>Lean Manufacturing</i>).....	42
CAPÍTULO 3 - DESENVOLVIMENTO.....	43
3.1 Requisitos Lógicos e Físicos.....	43
3.1.1 Mapa mental da arquitetura lógica	46
3.2 Arquitetura Física Proposta	47
3.3 Protótipo	49
3.3.1 Esquema de ligações.....	51
3.3.2 Programa de Gerenciamento	53
3.3.3 Interface Gráfica	59
3.3.3.1 Janela <i>Login</i>	59
3.3.3.2 Janela Modo Assistido	60
3.3.3.3 Janela Modo Automático	61
3.4 Testes Práticos.....	64
3.4.1 Testes com o Modo Assistido.....	65
3.4.2 Testes com o Modo Automático.....	77
3.4.3 Janela de ajuda.....	80
CAPÍTULO 4 - CONCLUSÕES	81
REFERÊNCIAS	82

CAPÍTULO 1 - INTRODUÇÃO

Este capítulo é composto pela descrição do problema, pela solução proposta, pelos objetivos do trabalho, pela revisão bibliográfica e pela estrutura do trabalho. Na descrição do problema é destacada a importância de se utilizar uma interface gráfica como meio para reconfigurar as tarefas executadas por uma célula de manufatura. Na solução proposta é sugerida a utilização de arquitetura baseada em sistema de supervisão e controlador para reconfigurar as tarefas da célula de manufatura. Nos objetivos do trabalho é proposta a utilização de uma arquitetura capaz de reconfigurar as tarefas de uma célula de produção por meio de controlador e interface gráfica. Na seção destinada à revisão bibliográfica são apresentados resumos de trabalhos técnicos relacionados com a presente dissertação. No item referente à estrutura do trabalho é mostrada a sequência adotada para a distribuição dos capítulos e os respectivos comentários sobre os seus conteúdos.

1.1 Descrição do Problema

Atualmente as exigências impostas pela economia globalizada configuram um cenário no qual os produtos comercializados devem possuir como requisitos básicos: custo competitivo, opções variadas de configuração e qualidade mínima para a sua comercialização. Para atender esse cenário, as indústrias estão investindo na automatização das suas linhas de produção e instalando células flexíveis em pontos estratégicos do processo produtivo, a fim de reduzir os custos de mão-de-obra, minimizar a quantidade de ciclos de produção e aumentar a qualidade e a consistência do produto (GROOVER, 2011). Decorrente das características operacionais estabelecidas para o funcionamento de cada uma dessas células se faz necessário que nos seus leiautes estejam basicamente presentes sistemas que: forneçam matéria-prima, executem o processo e transportem o produto acabado. O

robô manipulador é um exemplo típico de sistema que é aplicado na realização de diversificados tipos de tarefas em células flexíveis de produção (ROSÁRIO, 2005).

A tarefa de programação do robô manipulador para operar com outros sistemas presentes numa célula flexível de produção, a fim de atender as diferentes possibilidades de montagem no processo é atividade que exige profissional com perfil especializado, demanda tempo de treinamento e requer custos expressivos para a implementação. A manutenção dessa situação deve ser minimizada e/ou eliminada, pois a continuidade da mesma tem potencial suficiente para limitar a capacidade de crescimento e/ou permanência de uma empresa no mercado.

1.2 Solução Proposta

Utilizar uma arquitetura física e lógica para gerar as possíveis sequências de tarefas que podem ser executadas pela célula de produção. Essa arquitetura deverá possuir interface dedicada para que o usuário do sistema efetue a programação da sequência de tarefas, sem a necessidade do mesmo possuir conhecimentos especializados de programação e/ou parametrização dos componentes presentes na célula de produção.

Essa arquitetura deverá ser baseada em *software*¹ que permita a realização de modificações em ambiente integrado de desenvolvimento e *hardware*² dotado com recursos para comunicação por meio de rede de dados e módulo de controle, que possua dispositivos de entrada e saída configuráveis.

¹ *Software* – qualquer programa ou conjunto de programas de computador.

² *Hardware* – componente, ou conjunto de componentes físicos de um computador ou de seus periféricos.

1.3 Objetivos do Trabalho

Este trabalho tem por principal objetivo propor uma arquitetura baseada em controlador e sistema de supervisão que utiliza a interface gráfica como o meio para reconfigurar as tarefas que podem ser executadas por uma célula flexível de produção.

Comprovar a eficácia desse sistema por meio de testes em protótipo que adote os elementos básicos previstos na aludida arquitetura.

1.4 Revisão Bibliográfica

Em pesquisas realizadas nos anos de 2010 e 2011, foram observados vários artigos técnicos que guardam relação com o presente trabalho, sendo os mais relevantes apresentados nesta subseção, porém, não foram encontrados registros de trabalhos focados na sistematização de linhas de produção que oferecessem subsídios para reconfiguração de tarefas em células de manufatura.

A indústria de manufatura, nacional e internacional, busca produzir com maior precisão e agilidade. Um recurso de automação aplicado com o objetivo de melhorar a uniformidade, elevar a qualidade do produto e aumentar a taxa de produção é o arranjo em células de manufatura. Segundo Vieira (2007), as expressões “redução de custos”, “economia de energia” e “aumento de produtividade”, já há muito tempo, estão na pauta das reuniões das empresas de manufatura. Para atingir esses objetivos, o mercado oferece uma extensa gama de equipamentos e soluções automáticas, que atuam nos mais diferentes pontos de uma empresa. As empresas estão investindo no conceito de “Sistemas Integrados de Manufatura” com um objetivo comum: produzir mais, com mais qualidade, com flexibilidade de linha de produção, dentro de um espaço de tempo compatível com as necessidades de clientes. O que o mercado está implantando, de fato, como produção integrada,

vai muito além de máquinas orientadas por computador, daí a necessidade de se elaborar melhor as interfaces entre o operador e o sistema de produção.

De acordo com Pereira et al (2005), a utilização de células de montagem pode oferecer uma importante evolução para os ambientes de produção e linhas de montagem, pois essas privilegiam de modo mais significativo a gestão do conhecimento do que a gestão hierárquica, por meio de um enfoque moderno voltado mais para os valores do ser humano, sua intelectualidade e questões de aprendizagem inteligente. Isso mostra uma evolução do sistema chefe e empregado, onde a hierarquia é o que estabelece o padrão de conduta do operador ou funcionário e ainda apresenta uma disposição mais flexível, sem negligenciar a existência de controles e procedimentos. Nesse contexto, o foco nos processos de aprendizagem, no aprimoramento cultural e no nível de colaboração em grupo, tem potencial para proporcionar maiores possibilidades de aumento de qualidade e produtividade de forma inteligente e racional.

Segundo Rosário (2005), algumas das vantagens na utilização de robôs no processo produtivo estão relacionadas com a sua flexibilidade de programação e a habilidade para se adaptar a novas linhas de produção automatizadas, sem interromper o ciclo produtivo. Para que isso ocorra de forma eficiente, deve-se dispor de procedimentos sistemáticos para alterar a programação, permitindo assim uma reconfiguração de tarefas de forma rápida e precisa com o mínimo de tempo de parada entre as trocas de tarefas.

Como os diversos tipos de produtos têm seus ciclos de vida reduzidos, o custo do projeto, a construção e instalação de um novo sistema de manufatura, para produção de novas peças torna-se muito elevado (GROOVER, 2011). Os problemas vêm tanto em termos de tempo como de investimento financeiro. Na prática de engenharia moderna, mesmo os sistemas de produção de modelo único, estão sendo concebidos com características que permitam alterações com relativa facilidade. Essas características devem permitir que se reconfigure o sistema de produção para atender a novas demandas com agilidade e precisão.

O operador desempenha um papel importante na indústria moderna, sendo que na maioria das plantas industriais onde se utiliza sistemas *SCADA* (*Supervisory Control and Data Acquisition* – Supervisão, Controle e Aquisição de Dados) existem monitores de vídeo onde o operador pode interagir com o *software* em execução. Segundo Fonseca et al (2008), nos mencionados sistemas, os elementos reais de

processo como motores, válvulas e controladores são representados na programação pelo meio de objetos de virtuais relacionados com bibliotecas, os quais são testados e possuem interfaces de dados padronizadas.

1.5 Estrutura do Trabalho

O capítulo 1 é composto pela descrição do problema, pela solução proposta, pelos objetivos do trabalho, pela revisão bibliográfica e pela estrutura do trabalho. A descrição do problema apresenta, de maneira resumida, a importância de se proporcionar um meio para que se faça a reconfiguração de uma célula de produção pelo próprio operador, gerando assim uma menor dependência da célula com relação à equipe de manutenção. Na solução proposta, é enfatizado que a solução deve permitir modificações em ambiente integrado de desenvolvimento e que o hardware deve possuir elementos configuráveis. Nos objetivos do trabalho, são determinadas as principais metas que deverão ser alcançadas, ou seja, a elaboração de proposta de arquitetura e validação do sistema por meio de testes em protótipo e a comprovação do funcionamento do sistema de modo satisfatório. Na revisão bibliográfica, são apresentados artigos técnicos que guardam relação com o tema desta dissertação. A estrutura do trabalho apresenta a forma utilizada para a constituição dos capítulos deste trabalho.

No capítulo 2, são apresentados os conceitos básicos sobre os principais sistemas que o leitor deve conhecer para compreender a terminologia e os sistemas utilizados no desenvolvimento deste trabalho.

O capítulo 3 é dedicado para o desenvolvimento do trabalho, o qual apresenta a arquitetura proposta, a arquitetura do protótipo, a interface gráfica, o fluxograma analítico do sistema de controle, os testes práticos e os seus resultados.

No capítulo 4, são apresentadas as conclusões do trabalho.

CAPÍTULO 2 - CONCEITOS BÁSICOS

Este capítulo tem como objetivo principal apresentar de forma resumida, visando à aplicação apresentada neste trabalho, os conceitos sobre Sistemas de Supervisão, Controlador Lógico Programável (CLP), Robôs Manipuladores, Sistema CUBLOC, Linguagem LADDER, Redes de Computadores e Protocolos de redes, Optoacopladores, Sensores indutivos e Sistema de Manufatura Enxuta (*Lean Manufacturing*). Esses conceitos formarão a base mínima necessária para o leitor compreender as terminologias e sistemas utilizados nesta dissertação.

2.1 Sistema de Supervisão

Com a meta de facilitar a supervisão e o comando de variáveis do processo, em pontos estratégicos de uma planta automatizada e dessa forma auxiliar a tarefa de operação e manutenção do programa desenvolveu-se um sistema adequado, no âmbito de eficiência e ergonomia, o qual é denominado no mercado por Sistema de Supervisão ou Sistema Supervisório.

Os estados de um processo produtivo ou instalação física podem ser supervisionados por meio de interfaces gráficas que retratam as informações coletadas de equipamentos dedicados à aquisição de dados e, em seguida, essas são manipuladas, analisadas, armazenadas e apresentadas ao operador do sistema. Esses sistemas também são comumente designados por SCADA (*Supervisory Control and Data Acquisition* – Supervisão, Controle e Aquisição de Dados) (MORAES e CASTRUCCI, 2001).

Com o uso da tecnologia baseada em computadores e comunicação de dados, relacionadas com os sistemas de automação industrial, tornou-se possível efetuar a coleta de dados em ambientes complexos, eventualmente dispersos

geograficamente e a respectiva apresentação desses dados de modo expressivo para o operador do sistema.

Tipicamente os SCADAs utilizam *Tags*³ vinculadas com operações matemáticas, lógicas, vetores e *strings*⁴ ou podem representar pontos de entrada e saída de dados do processo, para que os dados sejam observados remotamente, formatados nos padrões determinados pelo operador e com controle facilitado.

O sistema SCADA também pode verificar condições de alarmes, sendo possível gravar registros em bancos de dados, ativar sinal sonoro e/ou visual, mudar cores de componentes gráficos e enviar mensagens. Esses sistemas são tipicamente constituídos por:

- **Banco de Dados:** armazenam informações, geralmente provenientes dos endereços de registradores pertencentes ao Controlador Lógico Programável (CLP), por meio de *Tags* onde estão armazenados esses conteúdos.
- **Gráfico de Tendência:** mostra o comportamento de determinadas variáveis, tipicamente relacionadas com o sistema supervisionado, ao longo do tempo e por meio de sua imagem gráfica. Essa imagem pode possuir diferentes cores para se obter uma melhor comparação entre os dados supervisionados e/ou analisados.
- **Alarmes:** são utilizados para monitorar estados de variáveis, principalmente, com a meta de sinalizar para o operador ou usuário do sistema a modificação relacionada com o estado do processo, objetivo específico atingido e indicação global sobre o estado do processo. Esses alarmes podem ser do tipo sonoro e/ou visual.
- **Módulo de Comunicação:** é responsável pela comunicação entre o sistema de supervisão, instalado na camada do aplicativo do computador hospedeiro, e o Controlador Lógico Programável (CLP). Essa

³ *Tag* – elemento que relaciona uma variável, digital ou analógica, do Controlador Lógico Programável (CLP) com a do sistema de supervisão.

⁴ *String* - quaisquer séries de caracteres alfanuméricos ou palavras consecutivas que são manipuladas e tratadas como uma unidade pelo computador.

comunicação pode ser realizada por meio de *drivers*⁵ de acordo com a plataforma utilizada.

- **Módulo de Edição de Janelas ou Janelas:** possui recursos para edição gráfica e biblioteca de figuras que permitem representar graficamente o ambiente industrial na janela do computador hospedeiro, permitindo o monitoramento do sistema representado ao longo do tempo. A Figura 1 mostra um exemplo de janela utilizada por um sistema para supervisão de processo industrial.

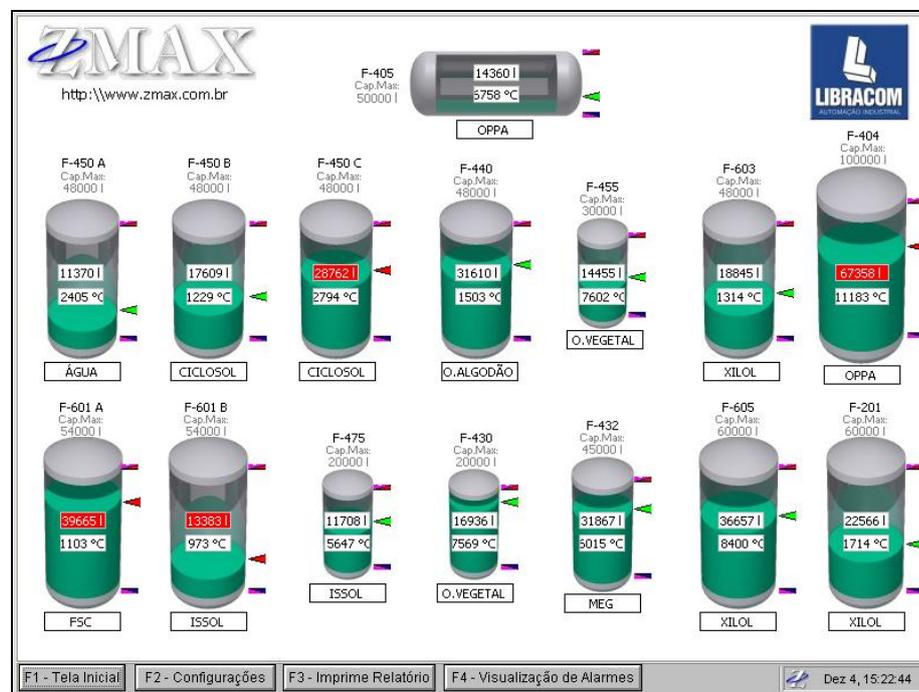


Figura 1 - Exemplo de janela para supervisão de processo (ELIPSE, 2011)

- **Acesso e Segurança:** o acesso ao sistema de supervisão pode ser restringido por meio de janela específica de usuário (*login*⁶) e senha. Essa restrição pode estar prevista na iniciação do aplicativo.

⁵ *Drivers* - software necessário para a adequada interação entre o *hardware* e o sistema operacional.

⁶ *Login* - conjunto de caracteres solicitados para os usuários que, por algum motivo, necessitam acessar algum sistema computacional.

- **Módulo de Programação:** permite, pela utilização de *scripts*⁷, realizar cálculos matemáticos, funções lógicas por meio de estruturas condicionais e conversões de unidades que têm diversas aplicações no gerenciamento de sistemas industriais.

2.2 Controlador Lógico Programável

O primeiro CLP (Controlador Lógico Programável) foi desenvolvido no final da década de 60. O dispositivo de computação chamado MODICON⁸, foi desenvolvido e construído a partir das necessidades de uma divisão da empresa GM (General Motors). Decorrente da grande dificuldade de mudar a lógica de controle e/ou comando de painéis elétricos que utilizavam relés⁹, a cada mudança nas linhas de montagem, a GM desenvolveu a seguinte especificação, que refletia as necessidades de muitos usuários desses tipos de painéis, não só para a indústria automobilística como para outras indústrias de manufatura (FRANCHI, 2011):

- Possibilidade de manutenção e reparo, com blocos de entrada e saída modulares.
- Confiabilidade, para que possa ser utilizado em um ambiente industrial.
- Redução de tamanho em comparação ao sistema tradicional que utilizava relés.
- Ser competitivo em custo com relação a painéis de relés e eletrônicos equivalentes.

⁷ *Script* - designa uma seqüência de comandos e tarefas a serem executados.

⁸ MODICON (*Modular Digital Controller* – Controlador Digital Modular) – foi primeiro CLP desenvolvido por uma companhia americana chamada Bedford Associated nos anos 60.

⁹ Relé - eletroímã que tem por função abrir ou fechar contatos elétricos, a fim de estabelecer ou interromper circuitos.

- Possibilitar entradas em tensão de 115 V e saídas com 115 V e com capacidade mínima de corrente de 2 A para operar com válvulas solenoides¹⁰ e relés.
- Possibilitar expansões sem grandes alterações no sistema.
- Memória programável com no mínimo 4 kbytes¹¹ e possibilidade de expansão.
- Estações de operação com interface expressiva e intuitiva.
- Possibilidade de integração dos dados de processo do CLP em bancos de dados gerenciais, para tornar disponíveis as informações sobre o chão de fábrica para os departamentos envolvidos com o planejamento da produção.

Um CLP (Controlador Lógico Programável) é um “aparelho eletrônico digital que utiliza memória programável para o armazenamento interno de instruções para implementações específicas, tais como lógica, seqüenciamento, temporização e contagem, para controlar através de módulos de entradas e saídas vários tipos de máquinas e processos” de acordo com a NEMA¹².

Com relação à nomenclatura, o termo CP (Controlador Programável) é mais genérico e engloba qualquer tipo de controlador programável, independente de sua configuração (FONSECA et al., 2008) e os termos PLC¹³ e CLP são mais empregados na indústria. Neste trabalho optou-se pelo uso do termo CLP por ser mais comum e ter correspondência direta em português.

A Figura 2 apresenta o diagrama em blocos simplificado de um típico CLP para uso industrial.

¹⁰ Solenóide - bobina cilíndrica, de comprimento maior que o diâmetro, a qual, quando por ela passa uma corrente, se comporta como um ímã em forma de barra, de modo que um núcleo móvel é puxado para dentro dela, geralmente empregada no controle de fluxo em sistemas pneumáticos e hidráulicos.

¹¹ Kbyte - unidade de medida de dispositivos de armazenamento, significando 1.024 bytes.

¹² NEMA - *National Electrical Manufactures Association*, em português Associação americana de fabricantes de equipamentos elétricos.

¹³ PLC – *Programable Logic Controller*, em português Controlador Lógico Programável.

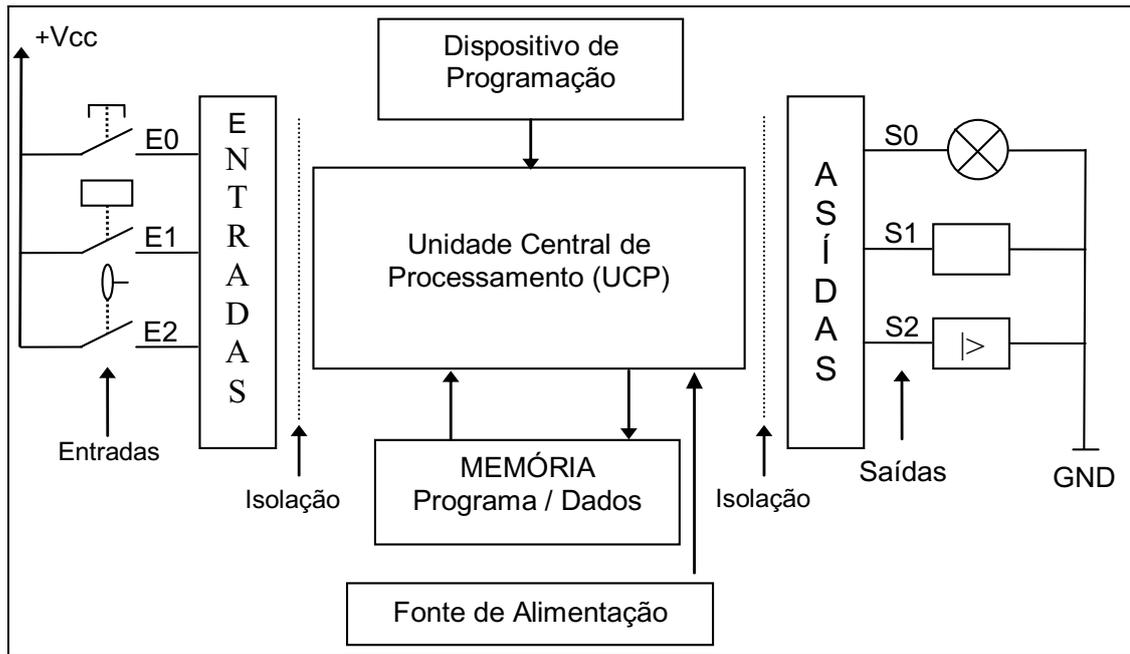


Figura 2 - Diagrama em blocos de um CLP (modificado de GEORGINI, 2007)

A fonte de alimentação, mostrada na Figura 2, é o componente responsável pelo fornecimento adequado de energia elétrica para a Unidade Central de Processamento (UCP), as memórias e os módulos de entrada e saída, eventualmente pode haver uma bateria para manter os dados internos do CLP (Controlador Lógico Programável), como por exemplo, o programa do usuário.

Os módulos de entrada e saída realizam a comunicação entre o CLP e o meio externo, garantindo a isolação galvânica e proteção da Unidade Central de Processamento (UCP). Os módulos de entrada recebem os sinais de dispositivos de entrada, como chaves e sensores e os condicionam para o processamento na UCP. Os módulos de saída são responsáveis por enviar os sinais aos dispositivos de saída, como relés e válvulas, tipicamente atuadores, em conformidade com o programa em execução no controlador.

Esses módulos podem operar com sinais digitais e/ou analógicos e, geralmente, são isolados dos sensores e atuadores por meio de separadores galvânicos, como acopladores ópticos, relés e outros relacionados.

A Unidade Central de Processamento (UCP) do Controlador Lógico Programável (CLP) tradicional é formada por um microprocessador, memórias e um sistema de comunicação. A função dessa unidade está concentrada em: realizar leitura das entradas, executar a lógica de controle de acordo com instruções do programa do usuário e atualizar as saídas do controlador.

O sistema de memórias é constituído tipicamente por memórias do tipo RAM (*Random Access Memory* - Memória de Acesso Aleatório) e ROM (*Read Only Memory* - Memória Somente de Leitura), podendo ser estruturado da seguinte forma (GEORGINI, 2007):

- Memória do Sistema de Operação: composta do programa desenvolvido pelo fabricante do CLP, que supervisionam e executam a seqüência de operações e as atividades de controle e comunicação com os dispositivos periféricos.
- Rascunho do sistema: é uma área de memória reservada para o armazenamento temporário de pequena quantidade de dados utilizada para cálculos do sistema, como contagens, calendário, entre outros relacionados.
- Memória de Aplicação ou do Usuário: armazena o programa desenvolvido pelo usuário para execução do controle desejado.
- Tabela de dados (memória imagem): armazena dados que são utilizados pelo programa de aplicação desenvolvido pelo usuário, como valores de pré-ajuste (*preset*), além dos estados (*status*) dos pontos de entrada e saída.

Os fabricantes oferecem várias alternativas de equipamentos, como os micro CLPs com estrutura fixa, para aplicações com pequena quantidade de pontos de entrada e saída, e para estruturas mais complexas, são oferecidas estruturas modulares e expansíveis com maior quantidade de pontos, como se pode observar na Figura 3.

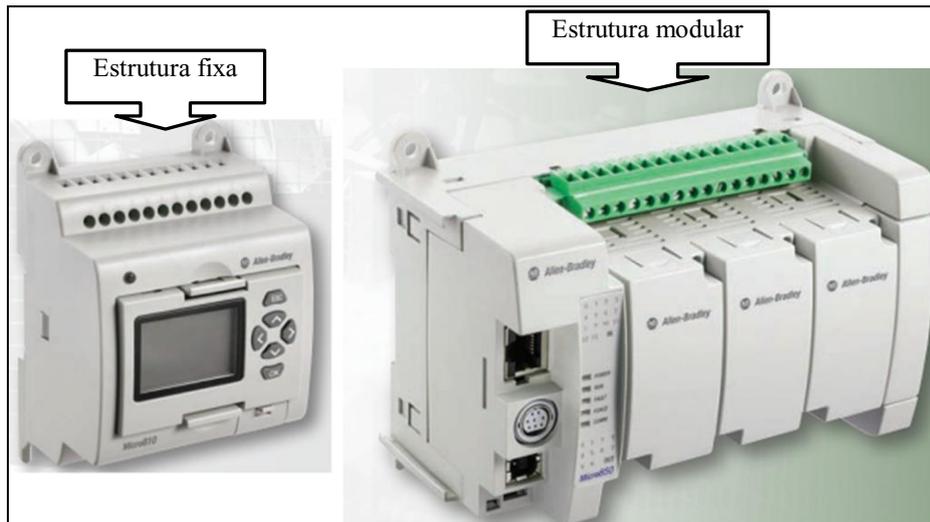


Figura 3 - Modelos de Controladores (ALLEN-BRADLEY, 2011)

O CLP atua segundo um ciclo de varredura que consiste em realizar operações de forma seqüencial e repetidas. A Figura 4 mostra, por meio de um fluxograma analítico, as fases do ciclo de varredura de um CLP, para operação em modo de execução (*Run*).

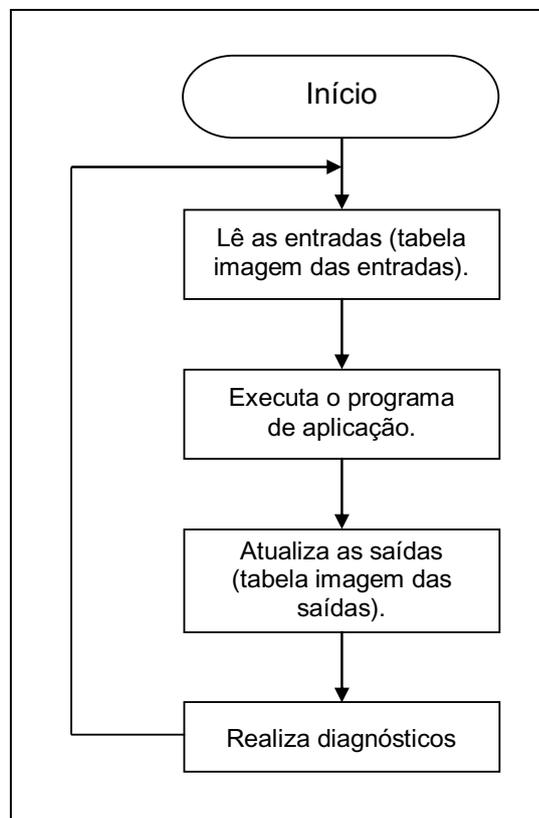


Figura 4 - Fluxograma do modo de operação de um CLP (FRANCHI, 2011)

A programação de um Controlador Lógico Programável (CLP) pode ser realizada em vários tipos de linguagens (FONSECA et al., 2008), como por exemplo:

- Linguagem Gráfica: é baseada na representação gráfica:
 - Diagrama Ladder (também designada por linguagem ladder) - utiliza diagrama de contatos para a representação das instruções de um programa.
 - Diagrama de Blocos de Funções (FBD - *Function Block Diagram*) - utiliza blocos funcionais para representação das instruções de um programa aplicativo.
- Linguagem Textual - consiste em um conjunto de caracteres bem definido e finito, regras para combinação desses caracteres para formação de palavras ou outras expressões e uma associação específica de significados para as palavras e expressões:
 - Lista de Instruções (IL - *Instruction List*): é uma linguagem textual que usa instruções para a representação do programa aplicativo.
 - Texto Estruturado: é uma linguagem textual que usa atribuições, controles de subprogramas, declarações de seleção e iteração para representação do programa aplicativo.

A linguagem habitualmente empregada na programação de CLP é o Diagrama Ladder, pois utiliza instruções com símbolos semelhantes aos contatos e bobinas de relés que são empregados na elaboração de esquemas elétricos.

A Figura 5 mostra o exemplo de uma parte do programa elaborado em linguagem Ladder. As instruções básicas usadas em um programa em linguagem Ladder são: Contato Normalmente Aberto (NA), Contato Normalmente Fechado (NF) e Bobina (BO).

Essas instruções são representadas na forma de símbolos gráficos inseridos nas linhas do programa, sendo por isso que a linguagem Ladder é também conhecida como “simbologia de contatos de relés”.

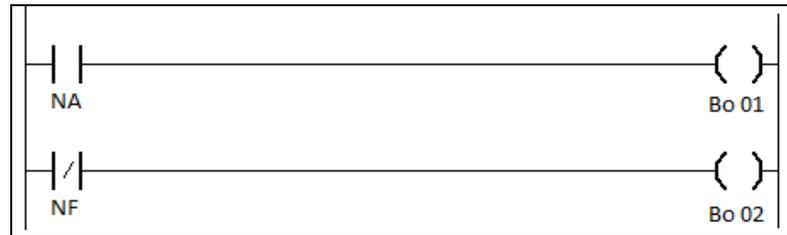


Figura 5 - Linhas de um programa em linguagem Ladder

2.3 Linguagem Ladder

A linguagem Ladder é do tipo gráfica, de interpretação expressiva, que representa ligações lógicas entre componentes elétricos virtuais (por exemplo: contatos e bobinas), por meio de símbolos semelhantes aqueles utilizados em diagramas elétricos de controle. A denominação Ladder deve-se à representação da linguagem ser semelhante a uma escada (*ladder*, em inglês), na qual duas barras verticais paralelas são interligadas pela lógica de controle, formando os degraus (*rungs*, em inglês) da escada.

Os diagramas Ladder são representados por linhas verticais (barras de alimentação), interligadas por meio de linhas horizontais que representam as casualidades formadas por um conjunto de elementos lógicos. A representação lógica dos diagramas Ladder considera apenas os símbolos e despreza a tensão elétrica envolvida na barra de alimentação e a intensidade da corrente pelo circuito (MORAES e CASTRUCCI, 2001).

Em um programa escrito em linguagem Ladder são encontrados, com frequência, os seguintes elementos:

- **CONTATO (*Contact*):** é o elemento capaz de representar os estados alcançados por um sensor, ou seja, a entrada de sinal no bloco de controle lógico. Pode ser uma chave, um sensor reflexivo, uma chave de fim de curso ou até mesmo o contato de algum relé auxiliar. Os contatos em Ladder podem assumir o estado aberto (NA – Normalmente Aberto) ou fechado (NF –

Normalmente Fechado), e sua representação é efetuada tipicamente por meio dos símbolos apresentados na Figura 6.



Figura 6 - Instruções básicas de entrada em Ladder

- **BOBINA (Coil):** é o elemento atuador, ou seja, o elemento acionado ou desligado pelo bloco de controle lógico. Pode representar os estados de operação de um motor, uma lâmpada, um atuador sonoro ou outro elemento que ao ser provocado execute uma ação. A bobina no diagrama Ladder pode assumir dois estados, ou seja, não energizada e energizada. O símbolo que representa esse elemento é apresentado na Figura 7, sendo “I1” um contato tipo NA e “O1” uma bobina.

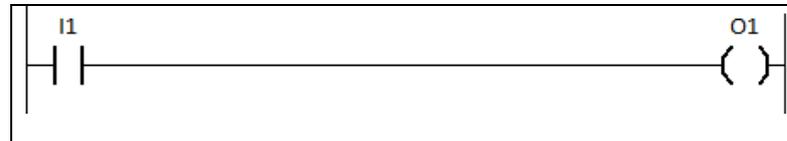


Figura 7 - Instrução básica de saída em LADDER

- **MEMÓRIA ou Relé Interno (Internal Relay):** é a representação do estado de um contato ou bobina em memória, sem conexão direta com elementos externos, permite que se utilizem contatos auxiliares no desenvolvimento do programa sem que esses estejam disponíveis na saída do CLP.

2.4 Sistema CUBLOC

O sistema CUBLOC integra em um mesmo invólucro, o micro controlador, dispositivo oscilador, circuito de iniciação e programa permanente local (*firmware*¹⁴). Essa configuração de *hardware* e *software* associados ao seu ambiente integrado de desenvolvimento (CUBLOC Studio), permite que o usuário elabore programas, em linguagem Ladder e/ou BASIC, em modo multitarefa, para ser executado diretamente pelo sistema no modo de operação (CUBLOC, 2006).

O sistema CUBLOC é construído e produzido usando uma metodologia inovadora denominada “*on-chip*” (em um *chip*), onde os principais componentes que auxiliam a operação do microprocessador encontram-se embutidos em um único dispositivo (*chip*). Isso proporciona maior liberdade e flexibilidade aos desenvolvedores, diminuindo o tamanho do desenho, do projeto e das aplicações, o que resulta em trabalhos menores, porém eficientes.

Os módulos CUBLOC possuem tamanho reduzido e elevada capacidade de processamento, o que permite aos desenvolvedores projetar Placas de Circuito Impresso (*Printed Circuit Board* - PCB) mais eficientes e personalizadas para suas aplicações.

2.4.1 Características do Sistema CUBLOC

Os equipamentos da linha CUBLOC apresentam diferenças importantes em relação aos Controladores Lógicos Programáveis (CLP) convencionais. O tamanho físico reduzido do CUBLOC é um grande diferencial, pois na atualidade, economizar espaço pode ser fator decisivo para sobrevivência de um projeto.

Outro recurso do sistema CUBLOC é a possibilidade de utilizar as instruções em linguagens Ladder e BASIC ao mesmo tempo, o que representa benefício

¹⁴ Firmware - conjunto de instruções operacionais programadas diretamente no hardware.

significativo ao se comparar com a tecnologia empregada nos Controladores Lógicos Programáveis (CLPs) disponíveis atualmente no mercado.

A lógica Ladder oferece condições para o desenvolvimento de aplicações, porém, como qualquer linguagem de programação, também apresenta limites. Ao integrar a linguagem BASIC no mesmo ambiente, têm-se um produto final que oferece mais opções ao desenvolvedor, pois as duas linguagens atuam sem comprometer o desempenho do equipamento.

A Figura 8 apresenta o exemplo de um Controlador Lógico Programável (CLP) e um controlador industrial baseado no sistema CUBLOC, sendo possível observar as diferenças físicas entre os mesmos.

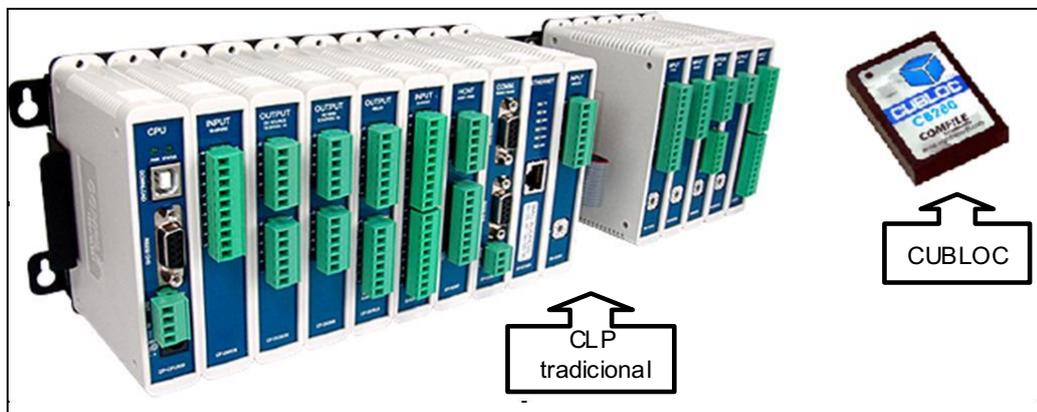


Figura 8 - Diferenças físicas entre CLP e CUBLOC (COMFILE, 2006)

No mercado de Automação Industrial não é comum encontrar Controladores Lógicos Programáveis (CLPs) que suportam as linguagens Ladder e BASIC que operam em modo multitarefa.

O sistema CUBLOC apresenta processamento otimizado e multitarefa, o que resulta em maior desempenho e capacidade de processamento, evitando perda de informação na entrada de dados, instabilidades e lentidão. A Figura 9 mostra o diagrama representativo do modo monotarefa (*SINGLE TASK*) e multitarefa (*MULTI TASK*), que são possíveis de serem executados no sistema CUBLOC para as linguagens Ladder e BASIC.

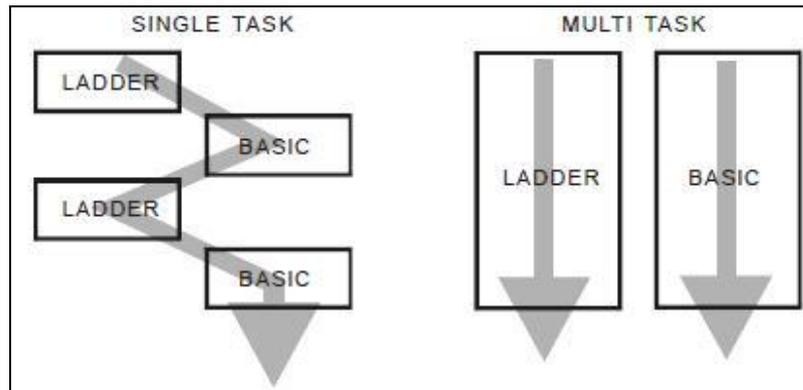


Figura 9 - Execução em monotarefa e multitarefa (CUBLOC, 2006)

A operação em modo 32-bits¹⁵ regulamentada pelo IEEE¹⁶ proporciona ao sistema CUBLOC suporte para operações matemáticas de ponto flutuante. Outro aspecto significativo é a utilização de protocolo *Modbus*¹⁷ ASCII¹⁸/RTU¹⁹ incorporado. Esse recurso proporciona aos equipamentos que utilizam o sistema CUBLOC destaque no mercado de automação industrial.

2.4.2 Ambiente de Desenvolvimento

Em conjunto com o *hardware* do sistema CUBLOC é oferecido o ambiente integrado de desenvolvimento denominado CUBLOC Studio, o qual utiliza o sistema operacional Microsoft® Windows.

O CUBLOC Studio permite: i) gravação e atualização do programa local de controle do *hardware (firmware)*, ii) teste e iniciação (*Reset*) de memória, iii) elaboração de programas, iv) gravação de programa do usuário, e v) supervisão dos estados alcançados pelas instruções durante a execução do programa do usuário. Posteriormente a realização da programação no sistema CUBLOC, o

¹⁵ Bit - a menor unidade de informação que pode ser armazenada ou transmitida.

¹⁶ IEEE - *Institute of Electrical and Electronic Engineers* - Instituto de Engenheiros Eletricistas e Eletrônicos.

¹⁷ Modbus – Protocolo de comunicação de dados criado pela empresa Modicon em 1979, muito utilizado em sistemas de automação industrial.

¹⁸ ASCII - *American Standard Code for Information Interchange* - Código Padrão Americano para o Intercâmbio de Informação.

¹⁹ RTU - *Remote Terminal Unit* – Dispositivo eletrônico multiprocessado, usado para monitorar campos digitais e sinais analógicos para depois os transmitir para uma estação central de monitoramento.

mesmo opera de modo independente do seu ambiente integrado de desenvolvimento, ou seja, modo isolado (*stand-alone*). Mesmo na hipótese de faltar energia elétrica para o *hardware* do sistema CUBLOC, o programa do usuário não é perdido, pois a memória do usuário é do tipo não volátil (ROM).

A Figura 10 apresenta a janela principal do ambiente de desenvolvimento CUBLOC *Studio* com parte de um programa elaborado em Ladder.

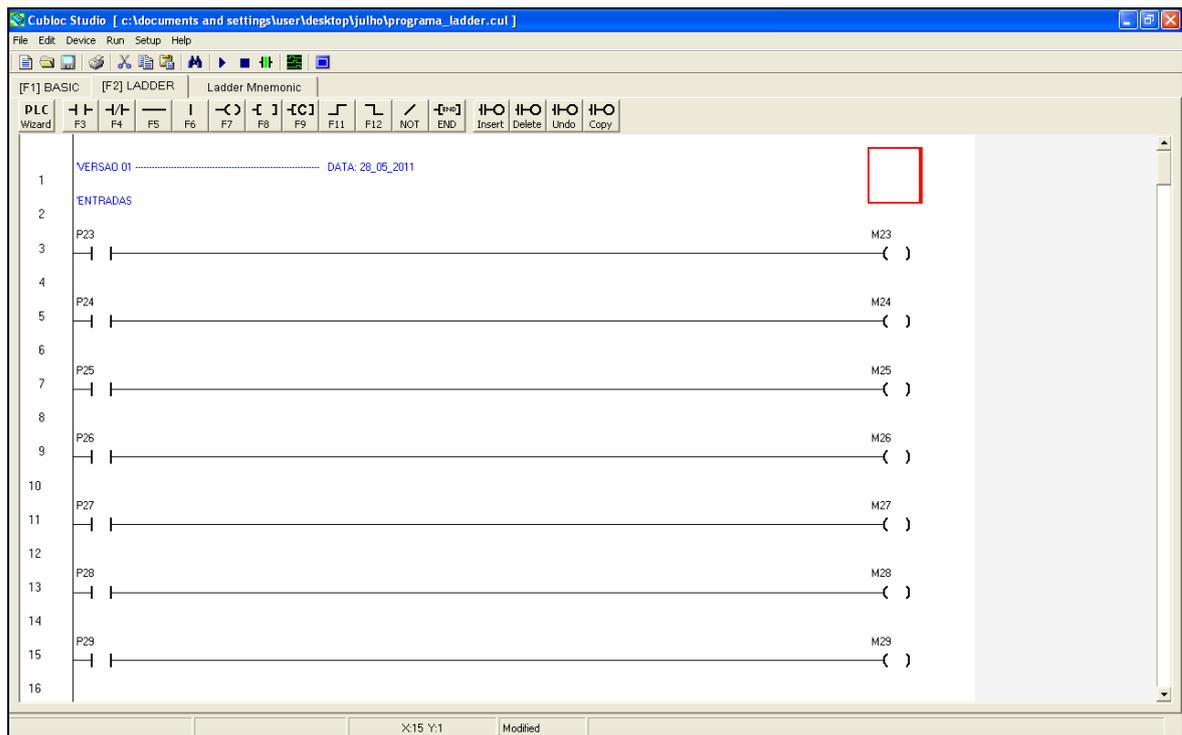


Figura 10 - Janela principal do CUBLOC Studio

2.5 Robô

De acordo com a norma ISO 8373²⁰ de 1994, robô é um manipulador controlado automaticamente, multifunção e programável em três ou mais eixos, podendo ser fixo ou móvel, para uso em aplicações industriais automatizadas. Deve

²⁰ ISO – *International Organization for Standardization* é um instituto internacional, com representantes em 162 países e sede em Genebra, que desenvolve e publica normas internacionais. A norma ISO 8373 define termos relevantes para robôs manipuladores industriais operados em ambiente industrial.

ser um equipamento cujos movimentos ou funções auxiliares podem ser modificados sem alterações físicas e capaz de ser adaptado a diferentes aplicações.

Um robô é composto basicamente por um atuador e um controlador. O atuador é um braço mecânico que executa trabalhos de manipulação de ferramentas e o controlador é um dispositivo eletrônico, programável que executa cálculos e define as tarefas que o atuador deve realizar.

Na década de 1960, quando foram lançados no mercado, os robôs eram caros e acessíveis a pouquíssimas empresas em países mais desenvolvidos, principalmente no Japão e nos Estados Unidos. A partir de 1976, seus preços começaram a baixar devido à redução de custos que ocorreu na informática e na eletrônica, que são responsáveis pelo controle dos eixos do robô.

O uso de robôs contribui para a automação em processos de média e pequena escala, onde basta alterar o programa e a ferramenta do robô para alterar o produto, consegue grandes incrementos na produtividade, possibilita uma melhora na qualidade do produto, uniformizando a produção e diminuindo as perdas e refugos. A capacidade do robô de reconfigurar seu ciclo de trabalho, unida à versatilidade e variedade de seus órgãos terminais (pinça, garras etc.) lhe permite adaptar-se facilmente a evolução e a troca dos processos de produção, facilitando sua reconversão para executar uma nova tarefa (GROOVER, 2011).

Na indústria os robôs são utilizados principalmente para executar tarefas onde seria inviável usar operadores humanos, são áreas de risco, insalubres ou para execução de tarefas onde haja a necessidade de alcançar alto grau de precisão mecânica e/ou repetição de movimentos, como por exemplo:

- Manipulação de objetos com alta temperatura na saída de forno.
- Processo de colagem com colas de alta viscosidade.
- Solda ponto em indústria automobilística.

Um sistema robótico pode ser dividido em três grandes componentes:

- i. Mecanismo: é a parte mecânica que tem liberdade de movimentos, por sua vez, se subdivide em três outras partes: Braço, Punho e Órgão terminal (efetuador, ferramenta etc.).
- ii. Acionamento: responsável por ligar e desligar os motores de acordo com as coordenadas do Sistema de Controle.

- iii. Sistema de controle: gera comandos, de acordo com um programa a ser feito pelo usuário ou por manipulação direta para, através do acionamento, efetuar a movimentação do robô.

A Figura 11 ilustra um robô, também denominado de “braço manipulador” com seis eixos ou graus de liberdade. Esse robô tem capacidade de movimentar peças de até 5 kg, comumente utilizado em processos de manipulação, pintura de peças e soldagem (ABB, 2011).



Figura 11 - Robô IRB 1410 (ABB, 2011)

A Figura 12 mostra uma unidade de controle, onde se pode destacar:

- Painel de operações – permite ao operador selecionar o modo de operação:
 - Posicionamento manual – utilizado para posicionar o robô utilizando a Unidade de Programação com velocidade reduzida.
 - Manual a 100% - utilizado para posicionar o robô utilizando Unidade de Programação com a velocidade programada.

- Automático – permite o robô executar o programa nele gravado e habilitado pelo operador.
- Unidade de Disco (*Disk drive*) – permite gravar e ler programas para alterações e execução do robô.
- Unidade de Programação (*Teach pendant*) – utilizado para a programação do robô, execução de tarefas de sincronismo e ajustes de posição.
- Chave principal – liga e desliga todo o conjunto.

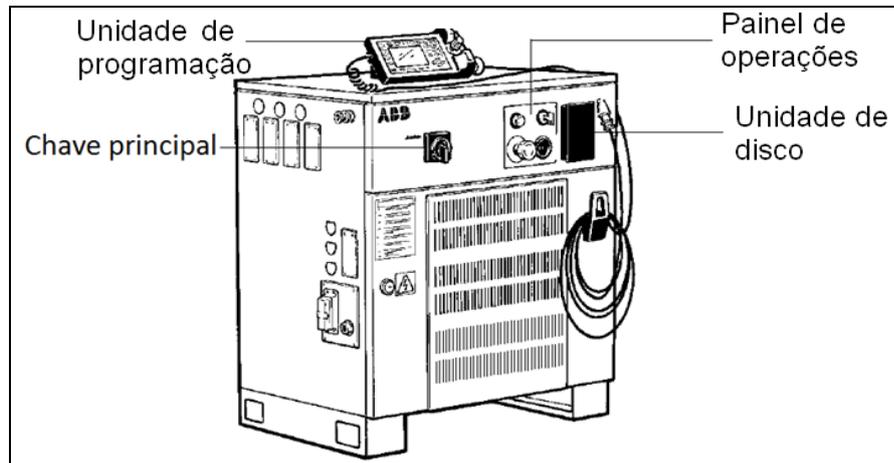


Figura 12 - Controlador do Robô IRB 1400 (ABB, 2003)

No Controlador do robô encontram-se, além dos elementos anteriormente mencionados, todo o *hardware* e *software* necessário para a elaboração, armazenamento e execução de programas responsáveis por colocar e manter o robô em operação.

2.6 Redes de Comunicação de Dados

Decorrente do expressivo avanço tecnológico ocorrido nas últimas décadas, as redes de comunicação de dados que são aplicadas na automação estão sendo maciçamente utilizadas por apresentarem as seguintes vantagens em relação a sistemas ponto a ponto de cabeamento: i) menor quantidade de condutores elétricos, ii) facilidade para realizar a manutenção, iii) flexibilidade na configuração dos componentes da rede e iv) capacidade para realizar diagnóstico de dispositivos. Além das aludidas vantagens, essas redes usam protocolos de comunicação padronizados, o que possibilita a integração de equipamentos de distintos fabricantes. Tais sistemas são classificados como abertos, sendo tendência em todas as áreas da tecnologia devido à sua flexibilidade e capacidade de expansão (MORAES e CASTRUCCI, 2001).

2.6.1 Redes Determinísticas e Probabilísticas

Redes determinísticas são aquelas cuja transmissão de dados ou de informações ocorre em instantes e intervalos de tempo determinados. Redes desse tipo permitem que o tempo de resposta seja conhecido, evitando problemas de iniciação, atrasos e perda de informações.

As redes probabilísticas permitem apenas calcular a probabilidade de transferência de informações que ocorrem em um determinado intervalo de tempo. De modo geral são considerados no projeto das redes de comunicação de dados os seguintes elementos (MORAES e CASTRUCCI, 2001):

- **Taxa de transmissão** - é a quantidade média de dados a serem transmitidos na rede em um período de tempo. O termo utilizado para especificação é capacidade de banda (*throughput*). A taxa de transferência de dados é medida em quilobits por segundo (kbps), que significa 1000 bits por segundo.

- **Topologia física da rede** - está relacionada com a disposição construtiva na qual os dispositivos estão conectados nas redes de barramentos distribuídos ou a sistemas de controle distribuídos.
- **Meio físico de transmissão** - estão relacionados ao cabeamento utilizado para a interconexão dos dispositivos. Existem muitos tipos de meios físicos de transmissão, e alguns exemplos são: cabos em par trançado, cabo coaxial e fibra ótica. Os meios físicos são selecionados de acordo com a aplicação. A seleção depende da distância entre os dispositivos, da taxa de transferência desejada, do protocolo a ser utilizado e outros pertinentes.
- **Algoritmo de acesso ao barramento** - é utilizado pelos nós para acessar ou disponibilizar informações na rede. Algoritmos típicos de acesso ao barramento são processos de varredura ou cíclica, como *Carrier Sense Multiple Acces Collision Detection*, o *CSMA/CD*²¹, *token, passing* etc.
- **Tecnologia de comunicação** – é a forma de gerenciamento entre os pontos de comunicação (nós) da rede com relação à comunicação de dados, sendo que essa pode ser conforme apresentado a seguir:
 - **Mestre-Escravo:**
 - Escravo (*slave*) - um escravo é um periférico (dispositivos inteligentes de Entrada/Saída, Drivers, Interfaces Homem-Máquina, Válvulas, Transdutores etc.), que recebe uma informação do processo e/ou utiliza informações de saída do mestre para atuar na planta. São dispositivos que respondem a requisições diretas vindas do mestre.
 - Monomestre - há somente um mestre no barramento durante a operação. Geralmente a Unidade Central de Processamento do Controlador Lógico Programável (CLP) é o componente de controle

²¹ CSMA/CD - *Carrier Sense Multiple Access with collision detection*. Protocolo de telecomunicações que organiza a forma como os computadores compartilham uma rede, tipicamente Ethernet.

central. Os escravos são acoplados de modo descentralizados no barramento através do meio de transmissão de dados.

- Multimestre - a imagem das entradas e saídas pode ser lida por todos os mestres, porém somente um mestre controla um dado escravo.
- **Ponto a Ponto (Origem – Destino):**
 - Nas redes ponto a ponto os dados devem ser enviados várias vezes para cada destino e a sincronização entre os nós ocorrem em instantes diferentes.

2.6.2 Protocolos

Os protocolos caracterizam os elementos de maior importância nas redes de automação industrial, tanto que as mesmas normalmente passam a ser denominadas pelos protocolos utilizados. São exemplos de protocolos: Profibus, Ethernet, Modbus.

O protocolo Modbus foi desenvolvido em 1979, pela empresa Modicon (Gould Modicon Company), para sistemas de controle de processos. Consideravelmente mais lento que outros protocolos, porém de mais fácil operação e manutenção, o que causou sua grande aceitação por parte de muitos fabricantes que produziram equipamentos compatíveis com esse protocolo na época (CLARK, 2004).

O Modbus opera com o princípio mestre-escravo e permite até 247 estações escravas e a forma de comunicação é realizada por meio de pergunta e resposta, ou seja, a estação principal faz uma solicitação (pergunta) para a estação secundária e depois de processado envia uma resposta de retorno. Nesse caso somente uma estação secundária é acessada por vez. A outra forma de comunicação utilizada

pelo Modbus é o envio de mensagens via “broadcast”²², assim todas as estações secundárias recebem o dado, mas não enviam respostas de volta (CLARK, 2004).

O protocolo Modbus padrão pode ser enquadrado na camada de aplicação do modelo OSI (*Open System Interconnection* - Interconexão de Sistemas Abertos), o qual padroniza o modelo em sete camadas: física, enlace, rede, transporte, sessão, aplicação e apresentação (MODBUS, 2011). Variações dos protocolos Modbus são usadas de modo associado com outros protocolos da camada de enlace e camada física (RS-232, RS-485 e Ethernet). Ele define uma estrutura de mensagens de comunicação empregadas para transferir dados discretos e analógicos entre dispositivos micro processados com detecção e informação de erros de transmissão.

2.7 Optoacopladores

Em muitos equipamentos são utilizados dispositivos ópticos para trocar informações entre diversas partes de seus circuitos, principalmente os que requerem um elevado grau de isolação galvânica entre o transmissor e o receptor. Exemplos desse uso são as conexões de circuitos lógicos e/ou micro processados com os circuitos de potência, onde a parte do sistema de controle é suprida por tensão elétrica contínua, tipicamente entre 5 a 24 Volts e a parte controlada, por tensões alternadas, com pontos de referência diferentes ou por tensões vindas de fontes diferentes. Decorrente da sua característica de funcionamento, os acopladores ópticos ou opto acopladores são utilizados principalmente em aplicações digitais (PINTO, 2011).

Um típico optoacoplador contém um LED²³ que emite radiação infravermelha, de modo alinhado, para um fotodetector instalado no mesmo bloco. Esses dispositivos se colocam de tal maneira que a energia irradiada pelo LED é transmitida eficientemente ao detector através de um meio dielétrico. O dielétrico

²² Broadcast – é o processo de transmissão ou difusão de determinada informação, tendo como principal característica que a mesma informação é enviada para muitos receptores ao mesmo tempo.

²³ LED – acrônimo de *light-emitting diode* - diodo emissor de luz, dispositivo ou componente eletrônico composto de cristal semiconductor capaz de emitir radiação desde a faixa ultravioleta até a infravermelha. Os mais comuns emitem luz vermelha, verde, amarela, azul ou branca.

está envolvido por um material opaco, o qual proporciona proteção contra as possíveis perturbações da luz ambiente. Nesse sistema não há conexão elétrica entre a entrada (LED) e a saída (elemento fotossensível). A Figura 13 apresenta o exemplo de um optoacoplador que utiliza LED e transistor fotossensível.

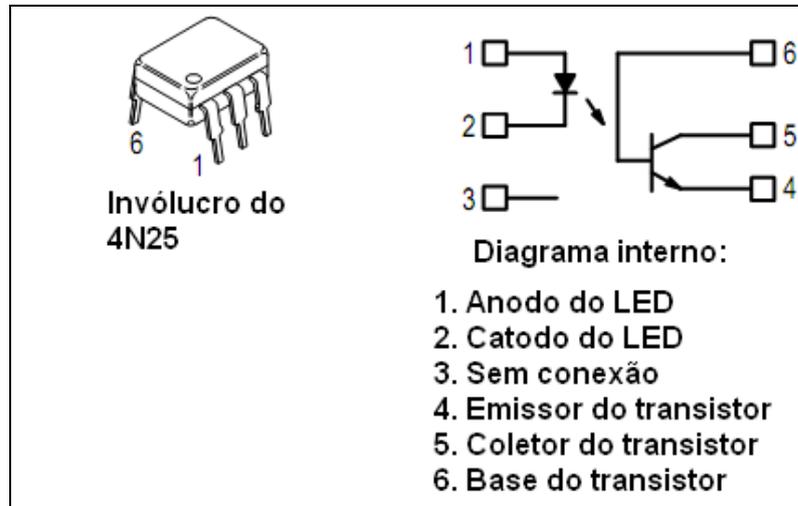


Figura 13 - Optoacoplador típico (MOTOROLA, 2011)

2.8 Sensores Indutivos

Os sensores indutivos foram desenvolvidos para atender as necessidades dos sistemas modernos de produção, nos quais é necessário conciliar elevadas velocidades e confiabilidade, substituindo com vantagem as chaves de fim de curso (COEL, 2011). São compostos por uma bobina que gera um campo eletromagnético de alta frequência, que é emitido pela face do sensor. Se um objeto metálico se aproxima dessa face, altera-se a frequência do oscilador, e essa alteração é amplificada e faz acionar a saída do sensor (THOMAZINI, 2005). A Figura 14 mostra o diagrama de um sensor indutivo típico.

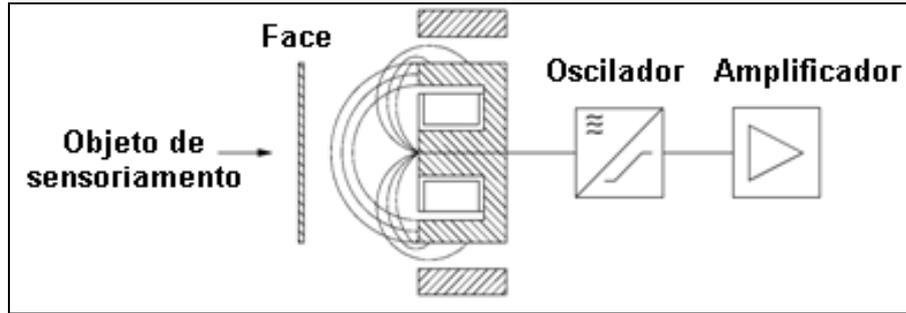


Figura 14 - Diagrama de um sensor indutivo (COEL, 2011)

Em Corrente Contínua (CC) a saída do sensor indutivo pode ser de dois tipos: PNP ou NPN. Esses tipos de saídas definem o modo como o sensor será conectado, como pode ser observado na Figura 15.

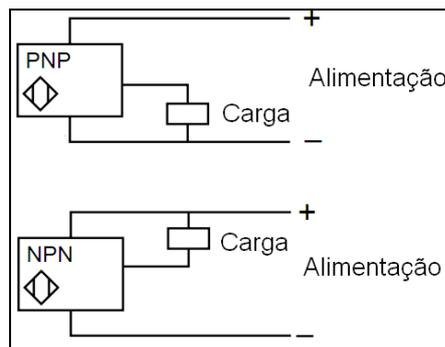


Figura 15 - Sensores classificados conforme a saída (SCHNEIDER, 2011)

Para facilitar a instalação dos sensores nas máquinas e/ou equipamentos, esses podem ser especificados com sistema de rosca e porcas, o que facilita o ajuste da distância de acionamento, como pode ser observado na Figura 16.

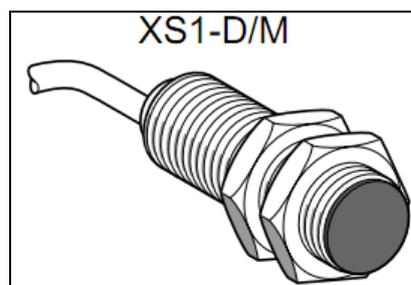


Figura 16 - Sensor indutivo com sistema de fixação (SCHNEIDER, 2011)

2.9 Manufatura enxuta (*Lean Manufacturing*)

A busca por melhorias no processo produtivo tem passado por diversos estágios nas últimas décadas e a Manufatura Enxuta, ou *Lean Manufacturing* tem conquistado cada vez mais espaço no ambiente industrial.

Esse sistema foi implantado pela empresa automotiva japonesa Toyota na busca de melhorias no processo produtivo. Essas melhorias vão desde a redução dos custos até uma maior flexibilidade em atender a demanda do mercado e seus conceitos são aplicáveis a toda a cadeia produtiva, da concepção do produto até o atendimento pós-venda (WOMACK et al, 2004).

Segundo Carneiro (2011), os pilares de sustentação da Produção Enxuta são compostos pela aplicação de diversas técnicas e ferramentas de apoio como a Manutenção Produtiva Total (MPT), Kanban, Gestão Visual, Círculo da Qualidade e outros. Tais ferramentas são utilizadas diretamente pelos trabalhadores do chão de fábrica, precisando por isso ser criados meios para compartilhar a responsabilidade dos objetivos organizacionais junto a esses trabalhadores, levando a um maior envolvimento dos mesmos com o resultado final do processo produtivo.

Nesse contexto, cabe citar o trabalho desenvolvido por Barros (2011), o qual trata da aplicação de recursos da manufatura enxuta na melhoria dos resultados da empresa GM Powertrain (Divisão de motores da *General Motors*) de São José dos Campos. De acordo com o autor, “as perdas mais comuns no processo produtivo são decorrentes de quebras de máquinas, tempo de parada das máquinas para realização de *setup*²⁴ ou ajustes, operação em vazio, pequenas paradas, redução da velocidade da máquina, fabricação de produtos defeituosos e perdas durante o início da produção”. Assim, um recurso que possibilite reduzir o tempo de parada de máquinas para a realização de *setup*, certamente impactará em melhorias no processo produtivo, em especial a redução do tempo de produção.

²⁴ *setup* – termo que designa o tempo gasto para a troca de ferramentas, programas e/ou equipamentos em um processo industrial.

CAPÍTULO 3 - DESENVOLVIMENTO

Neste capítulo, é apresentada uma proposta de arquitetura para reconfiguração de célula de produção. São apresentados, ainda, os requisitos previstos para o sistema, o mapa mental para a arquitetura lógica, o protótipo desenvolvido para validar os componentes previstos nessa proposta, o diagrama de ligações adotado, o programa de gerenciamento do sistema, a interface gráfica e os registros de testes efetuados.

3.1 Requisitos Lógicos e Físicos

Os principais requisitos lógicos e físicos que devem ser atendidos pela arquitetura baseada em controlador e sistema de supervisão, que utiliza a interface gráfica como o meio para reconfigurar as tarefas a serem executadas por uma célula flexível de produção, estão listados na Tabela 1 e Tabela 2 respectivamente.

Nessas tabelas estão previstas colunas específicas para: i) Item, ii) Descrição de requisito e iii) Tipo. O conteúdo da coluna denominada “Item” é utilizado como indexador para facilitar a localização de informações relacionadas com a descrição e tipo de requisito nas referidas tabelas. A coluna intitulada “Descrição de Requisito” contém as principais exigências físicas e lógicas que devem ser atendidas pelos componentes presentes na arquitetura proposta neste trabalho. Na coluna denominada “Tipo”, quando conter a indicação “Funcional” significa que o requisito afeta diretamente a operação do sistema em questão e quando conter a indicação “Não Funcional” essa operação não é afetada.

Tabela 1 - Requisitos lógicos da arquitetura.

Item	Descrição de Requisitos	Tipo
1	A arquitetura deve possuir interface gráfica dedicada para permitir o acesso do operador aos recursos do sistema, com usuário e senha.	Funcional
2	A arquitetura deve possuir interface gráfica dedicada e com recursos necessários para permitir ao operador: 2.1 Comandar, individualmente, o deslocamento do robô manipulador para as seguintes posições: Repouso, Alimentador, Sensor Indutivo, Balança, Mesa para Descarte de Produto ou Unidade de Armazenamento. 2.2 Comandar a operação de abrir ou fechar a garra do robô. 2.3 Parar a movimentação do robô. 2.4 Mudar para a janela com recursos de operação automática. 2.5 Mudar para a janela com informações auxiliares sobre a operação do sistema. 2.6 Sair do sistema	Funcional
3	A arquitetura deve possuir interface gráfica dedicada e com recursos necessários para permitir ao operador: 3.1 Definir uma entre as possíveis seqüências de movimentações que o robô pode executar na célula de produção flexível. 3.2 Executar a seqüência definida no item 3.1, anterior, uma única vez. 3.3 Executar a seqüência definida no item 3.1, anterior, constantemente. 3.4 Parar a execução da seqüência definida no item 3.1, anterior. 3.5 Mudar para a janela com recursos de operação assistida. 3.6 Mudar para a janela com informações auxiliares sobre a operação do sistema. 3.7 Sair do sistema	Funcional
4	A arquitetura deve possuir interface gráfica dedicada para permitir ao operador obter auxílio sobre a utilização dos recursos do sistema.	Não Funcional

Os principais requisitos que necessitam ser atendidos pela arquitetura física proposta neste trabalho são listados na Tabela 2.

Tabela 2 - Requisitos físicos da arquitetura.

Item	Descrição de Requisitos	Tipo
1	<p>A arquitetura deve possuir, no mínimo, três camadas para realizar a automação do sistema, da seguinte forma:</p> <p>1.1 A primeira camada deve atender os sensores e atuadores dos módulos presentes na célula flexível.</p> <p>1.2 A segunda camada deve realizar o controle desses módulos de acordo com os passos previstos no algoritmo de gerenciamento da arquitetura.</p> <p>1.3 A terceira camada deve ser dedicada para realizar a interface lógica com o operador do sistema.</p>	Funcional
2	<p>O transporte de sinais, digitais e analógicos, entre os sensores e atuadores da primeira camada e a interface física de entrada e saída do sistema de controle da segunda camada deve ser realizado por meio físico dedicado, sem utilizar protocolo de comunicação de dados, bidirecional e ponto a ponto.</p>	Funcional
3	<p>O transporte de sinais digitais entre o sistema de controle da segunda camada e a interface lógica da terceira camada deve ser realizado por meio físico dedicado, com a utilização de protocolo padronizado, bidirecional e ponto a ponto.</p>	Funcional
4	<p>O sistema de controle da segunda camada deverá possuir:</p> <p>4.1 Interface física com entradas e saídas, digitais e/ou analógicas, capaz de atenderem as amplitudes e frequências de sinais elétricos dos sensores e atuadores presentes nos módulos da arquitetura.</p> <p>4.2 Memória com capacidade de armazenamento suficiente para alojar o programa que efetua o gerenciamento de todas as atividades previstas para a operação da arquitetura de automação da célula flexível.</p> <p>4.3 Ciclo de processamento compatível com o modo de operação e as frequências de sinais elétricos presentes nos módulos da arquitetura.</p> <p>4.4 Interface física capaz de atender as amplitudes e frequências</p>	Funcional

	de sinais elétricos para realizar a comunicação de dados com o módulo que aloja a interface lógica com o operador do sistema.	
5	<p>O sistema de controle da segunda camada deverá possuir ambiente integrado desenvolvimento capaz de operar na camada do aplicativo de computador hospedeiro com arquitetura Intel® e recursos para:</p> <p>5.1 Realizar a programação, em linguagem gráfica e normalizada, dos passos previstos no algoritmo que gerencia as operações da arquitetura da célula flexível.</p> <p>5.2 Supervisionar os valores alcançados pelas variáveis contidas no programa de gerenciamento da arquitetura durante a sua operação.</p>	Não Funcional
6	O computador hospedeiro que atende a terceira camada deverá possuir interface para comunicação de dados compatível com o controlador da segunda camada, arquitetura Intel® e meios necessários para alojar o ambiente integrado desenvolvimento da interface lógica com o operador do sistema.	Não Funcional

3.1.1 Mapa mental da arquitetura lógica

O mapa mental elaborado para atender os requisitos previstos na Tabela 1 para a arquitetura lógica do sistema proposto neste trabalho é apresentado na Figura 17.

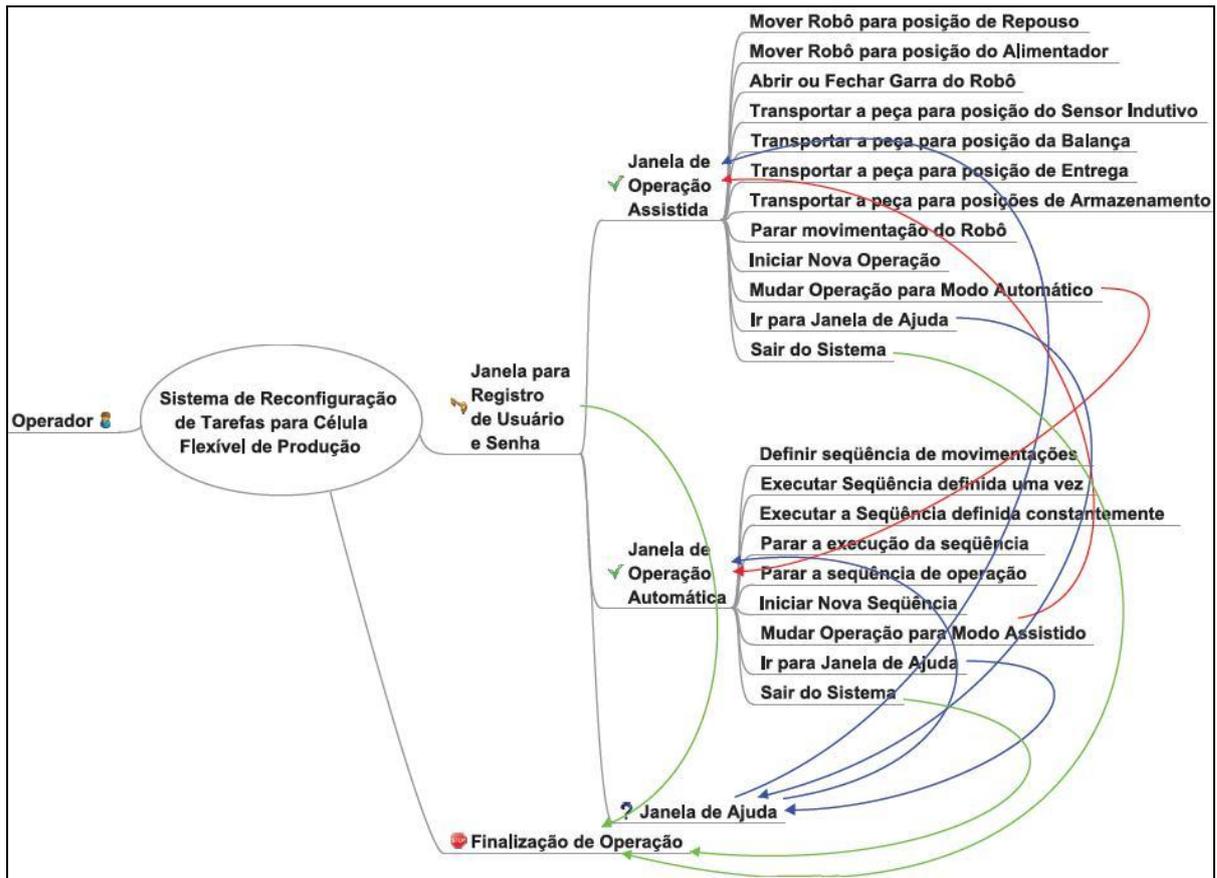


Figura 17 – Mapa mental da arquitetura lógica

No mapa apresentado na Figura 17 está previsto a utilização de quatro janelas para: i) Registro de Usuário e Senha, ii) Operação Assistida, iii) Operação Automática e iv) Ajuda. Somente nas janelas de Operação Assistida e Automática estão previstos os componentes que realizam a operação dinâmica dos módulos presentes na célula flexível. As setas contidas nesse mapa ilustram as possibilidades de saltos entre as janelas, os recursos e os estados de operação do sistema em questão.

3.2 Arquitetura Física Proposta

Os principais componentes representativos da arquitetura proposta neste trabalho para atender os requisitos físicos da Tabela 2, são mostrados na Figura 18.

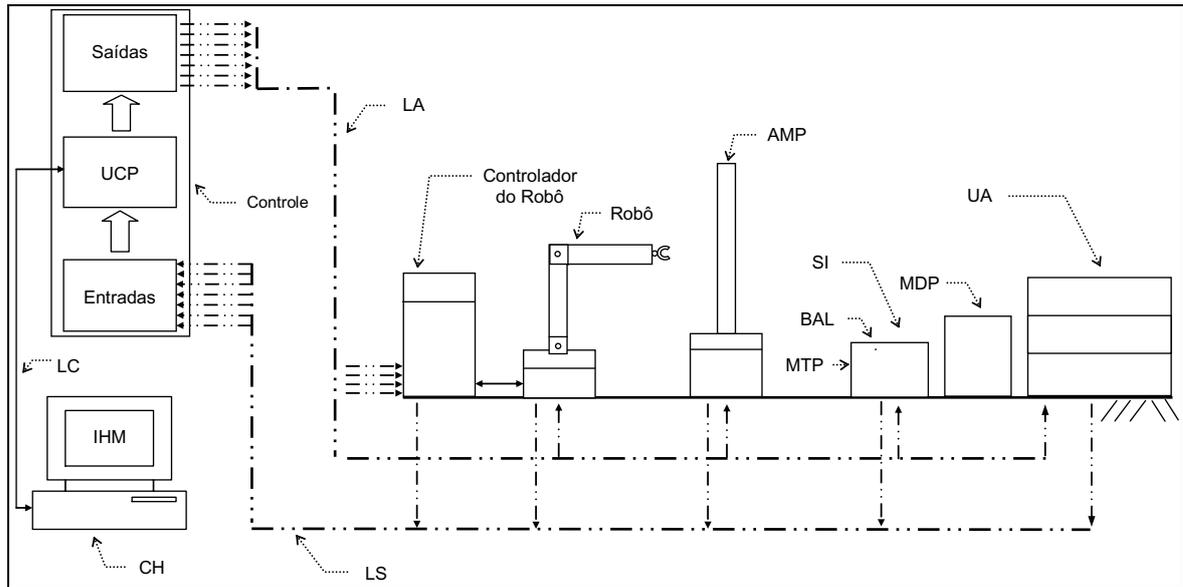


Figura 18 - Arquitetura proposta

A função do bloco intitulado por Controle, apresentado na Figura 18, pode ser executada por um Controlador Lógico Programável (CLP), pois esse sistema tem capacidade de operar com entradas e saídas, digitais e/ou analógicas, que atendem vasta faixa de amplitude de sinais elétricos (MIYAGI, 1996). Na memória destinada ao módulo da Unidade Central de Processamento (UCP) é instalado o programa que efetua o gerenciamento de todas as atividades previstas para a operação da arquitetura proposta (ROSÁRIO, 2005). Esse programa executará as tarefas de gerenciamento do sistema conforme a seleção efetuada pelo operador na Interface Homem-Máquina (IHM).

As principais funções do computador hospedeiro (CH) estão concentradas em abrigar a Interface Homem-Máquina (IHM) do sistema (SILVEIRA, 2002), e participar da transmissão e recepção de informações relacionadas com o programa instalado no bloco de Controle.

Os recursos previstos nessa interface permitem que o operador do sistema execute o acionamento e a supervisão dos seguintes elementos, previstos na célula flexível: i) robô manipulador (Robô), ii) alimentador de matéria-prima (AMP), iii) mesa para teste de produto (MTP), a qual é equipada com Balança (BAL) e Sensor Indutivo (SI) para seleção do tipo de material (metálico ou não metálico), iv) mesa para descarte de produto (MDP), v) unidade de armazenamento (UA). A interface

possui os recursos gráficos para estabelecer a operação da célula flexível em dois modos básicos: modo assistido e modo automático.

Na janela da interface que opera em modo assistido estão presentes recursos que permitem ao operador selecionar os movimentos específicos que o robô deve executar na célula flexível de produção. Esse recurso é destinado para atender a manutenção e definir a sequência otimizada de operações para utilização no modo automático.

Após a definição da sequência otimizada de operações obtida no modo assistido pode ser utilizado o modo automático. Nesse modo de operação a interface possui uma janela que permite ao operador estabelecer uma sequência de ações que o robô deverá executar automaticamente.

O meio físico que transporta sinais unidirecionais entre o módulo de Saídas do bloco de Controle e os atuadores presentes nos elementos previstos na célula flexível é denominado Linha de Atuadores (LA). Essas linhas são separadas galvanicamente e providas de proteção contra interferências eletromagnéticas de modo comum e de modo diferencial.

As Linhas de Sensores (LS) transportam sinais unidirecionais entre os sensores presentes nos elementos previstos na célula flexível e o módulo de Entradas do bloco de Controle. Essas linhas também são separadas galvanicamente e providas de proteção contra interferências eletromagnéticas de modo comum e de modo diferencial.

3.3 Protótipo

Uma vista dos módulos previstos no protótipo que foi montado para avaliar a proposta de arquitetura que utiliza uma interface gráfica para reconfigurar as tarefas executadas por uma célula flexível de produção é apresentada na Figura 19. Nesse protótipo foram realizados os testes para validar os principais blocos do sistema proposto neste trabalho, o qual está representado na Figura 18.

Como pode ser observado na Figura 19 os módulos previstos no protótipo são: i) Robô Manipulador (Robô), ii) Alimentador de Matéria-Prima (AMP), iii) Mesa para Teste de Produto (MTP), a qual é equipada com Balança (BAL) e Sensor

Indutivo (SI) para seleção do tipo de material (metálico ou não metálico), iv) Mesa para Descarte de Produto (MDP), v) Unidade de Armazenamento para as peças selecionadas (UA), vi) Computador Hospedeiro (C.H.), vii) Interface Homem-Máquina (IHM), viii) Controle e, ix) Controlador do robô.

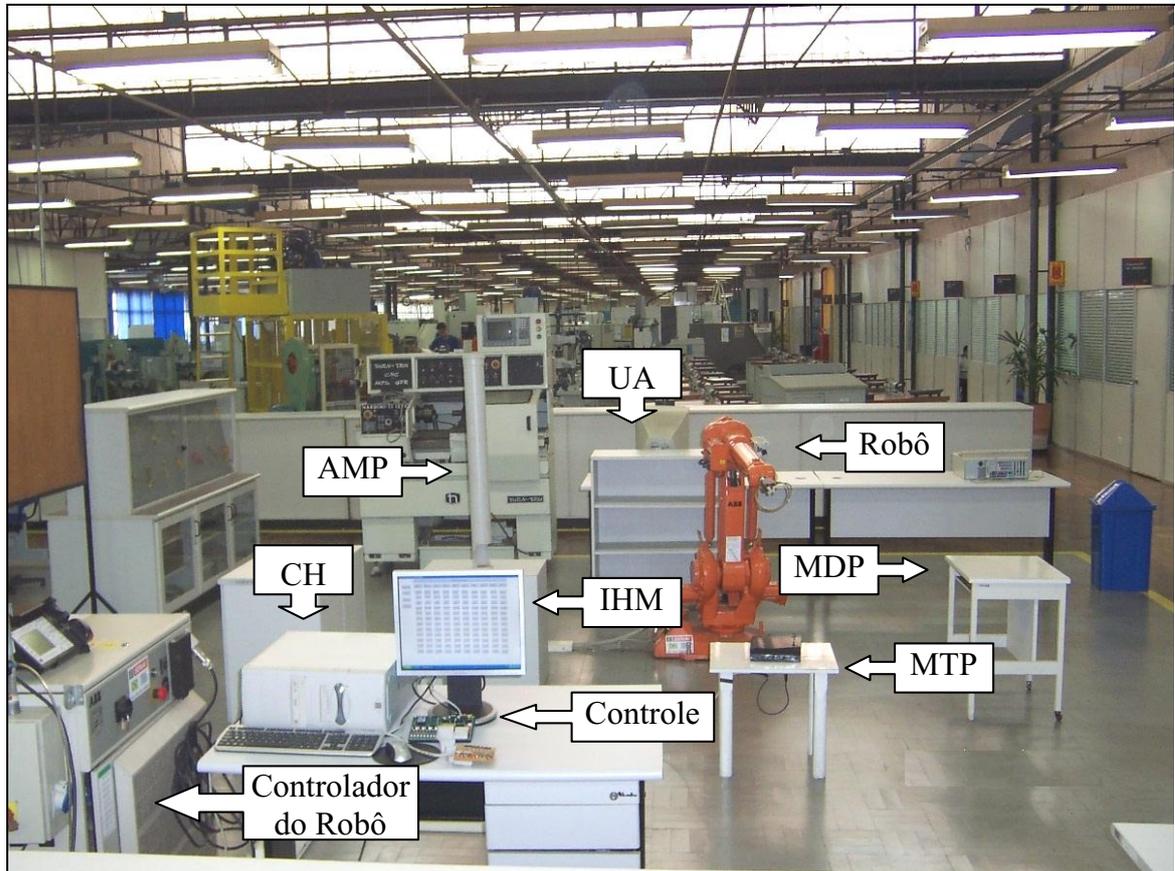


Figura 19 - Vista do protótipo

A interface Homem-Máquina (IHM) está alojada na camada do aplicativo do Computador Hospedeiro (CH), o qual foi utilizado nos testes práticos deste trabalho. Essa interface foi elaborada com a quantidade de janelas e recursos necessários para permitir que o operador do sistema efetue o acionamento remoto do robô em modo assistido e automático. Dentre os vários recursos disponíveis nessa interface o operador pode:

- Definir uma entre as possíveis sequências que deseja realizar.
- Iniciar a operação automática de sequência definida.
- Comandar a execução da sequência uma única vez ou repetidamente.

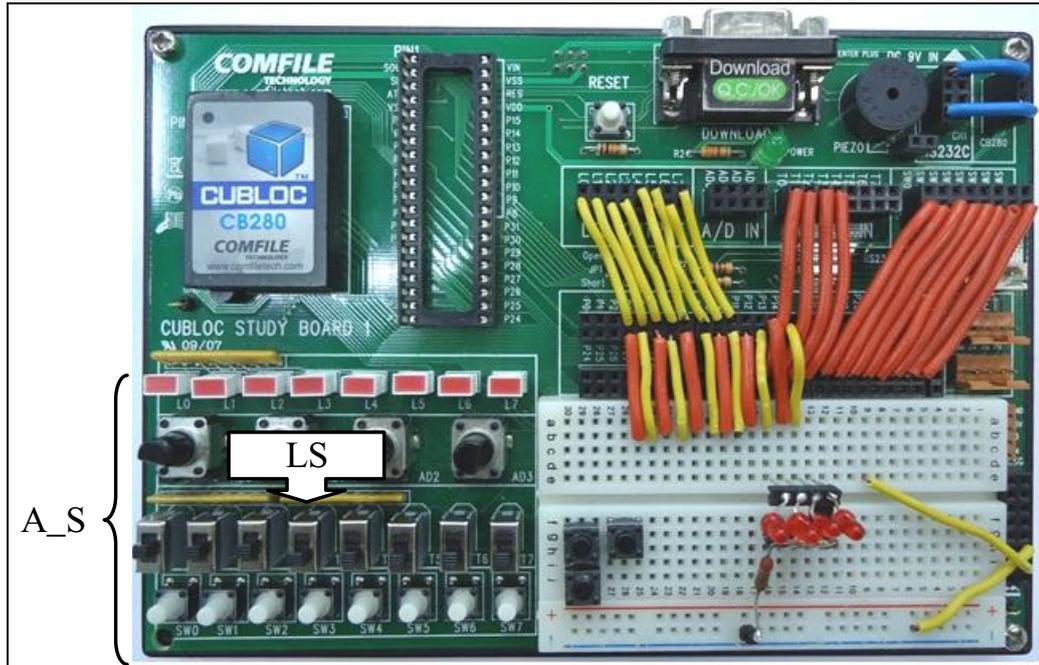


Figura 21 - Unidade de acionamento e supervisão

Os sinais que acionam as entradas do robô são aplicados por meio de uma placa de interface com acopladores ópticos (OPT), pois as saídas do sistema CUBLOC fornecem tensão de 5 Volts e as entradas do robô operam com 24 Volts. O módulo OPT é conectado ao robô por um cabo multivias mostrado na Figura 22. O referido circuito garante ainda uma isolamento galvânica entre os dois sistemas envolvidos.

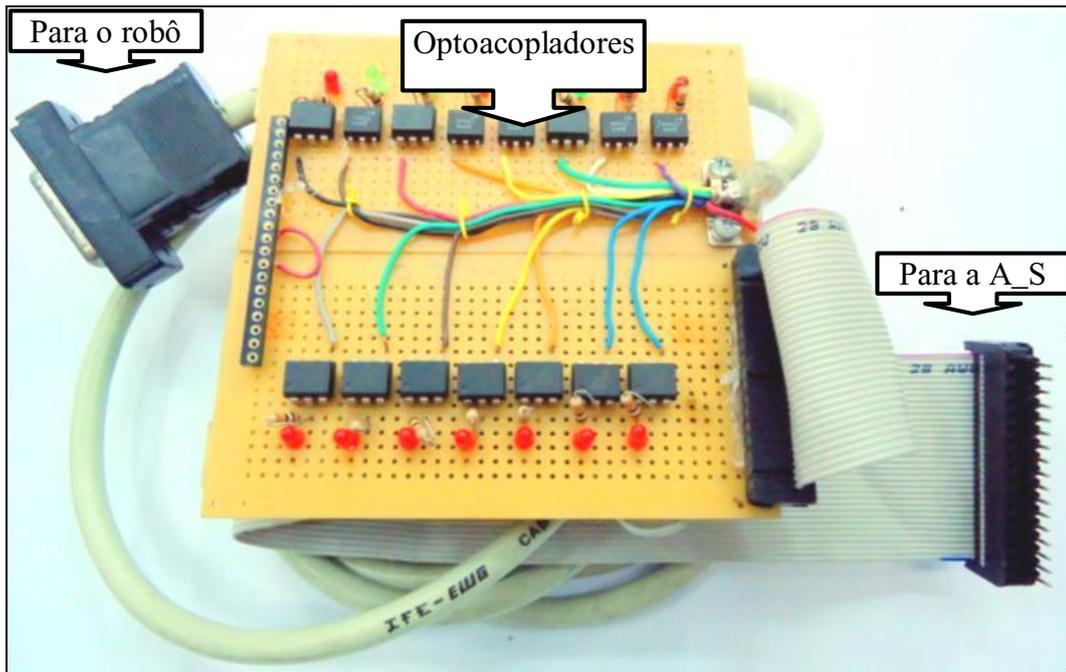


Figura 22 - Interface com acopladores ópticos (OPT)

A Figura 23 apresenta o circuito eletrônico utilizado para o módulo com acopladores ópticos (OPT), o qual possui optoacoplador (U1) 4N25 (MOTOROLA, 2011), diodo emissor de luz (D1) e resistor (1k Ω).

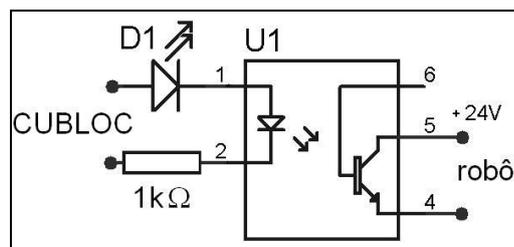


Figura 23 - Circuito eletrônico utilizado para o módulo OPT

3.3.2 Programa de Gerenciamento

O programa de gerenciamento de informações da arquitetura proposta neste trabalho foi elaborado para atender dois principais modos de operação, ou seja, Modo Assistido e Modo Automático.

Na operação em Modo Assistido o objetivo está concentrado em determinar a sequência otimizada de acionamentos desejada pelo operador de acordo com o processo que deverá ser executado. Nesse sentido, o programa de gerenciamento de informações para atender esse modo de operação possui rotinas que permitem ao operador comandar o Robô, deslocando-o para os pontos desejados por meio das seguintes opções:

- Mover o robô para a posição de Repouso.
- Mover o robô para a posição do Alimentador.
- Abrir ou Fechar a Garra do Robô.
- Transportar a peça para a posição do Sensor indutivo.
- Transportar a peça para a posição da Balança.
- Transportar a peça para a posição de Entrega.
- Transportar a peça para as posições 1A, 2A 3A, 1B, 2B ou 3B.
- Mudar a operação para o modo Automático.

Os recursos do Modo Automático permitem ao operador do sistema:

- Efetuar a programação de uma sequência de operações que a célula flexível deverá executar automaticamente.
- Executar a sequência programada apenas uma única vez.
- Executar repetidamente a sequência programada.
- Reiniciar o sistema para uma nova programação.
- Paralisar a operação do sistema.

Os principais passos previstos no algoritmo que foi elaborado para atender o programa de gerenciamento da arquitetura proposta neste trabalho são apresentados a seguir:

Passo 1 – Registrar Usuário e Senha para acesso ao sistema.

Passo 2 – Escolher o modo de operação.

Passo 3 – Ativar o modo de operação Assistido?

Sim, acionar o modo assistido e executar o passo 10.

Não, ir para o Passo 4.

Passo 4 – Ativar o modo de operação Automático?

Sim, executar o passo 6.

Não, executar o passo 5.

Passo 5 – Sair do programa?

Sim, ir para o passo 18.

Não, retornar para o passo 2.

Passo 6 – Colocar o robô na posição de repouso.

Passo 7 – Selecionar a sequência de operações.

Passo 8 – Executar a sequência selecionada

Passo 9 – Sair do modo Automático?

Sim, retornar para o passo 2.

Não, ir para o passo 6.

Passo 10 – Ir para a posição de repouso?

Sim, colocar o robô na posição de repouso e ir para o passo 11.

Não, ir para o passo 11.

Passo 11 – Abrir a garra?

Sim, abrir a garra do robô e retornar para o passo 10.

Não, ir para o passo 12.

Passo 12 – Fechar a garra?

Sim, fechar a garra do robô e retornar para o passo 10.

Não, ir para o passo 13.

Passo 13 - Verificar se a peça é de metal?

Sim, movimentar o robô até o sensor indutivo e retornar para o passo 10.

Não, ir para o passo 14.

Passo 14 - Ir para o alimentador?

Sim, movimentar o robô até o alimentador e retornar para o passo 10.

Não, ir para o passo 15.

Passo 15 - Entregar a peça?

Sim, movimentar o robô até posição de entrega e retornar para o passo 10.

Não, ir para o passo 16.

Passo 16 - Testar peso da peça?

Sim, movimentar o robô até posição da balança e retornar para o passo 10.

Não, ir para o passo 17.

Passo 17 – Sair do modo Assistido?

Sim, retornar para o passo 2.

Não, retornar para o passo 10.

Passo 18 – Finalizar o aplicativo.

O fluxograma analítico que representa uma sequência específica de ações previstas nesse programa de gerenciamento, que foi utilizada na realização dos ensaios práticos deste trabalho é apresentado na Figura 24 e Figura 25.

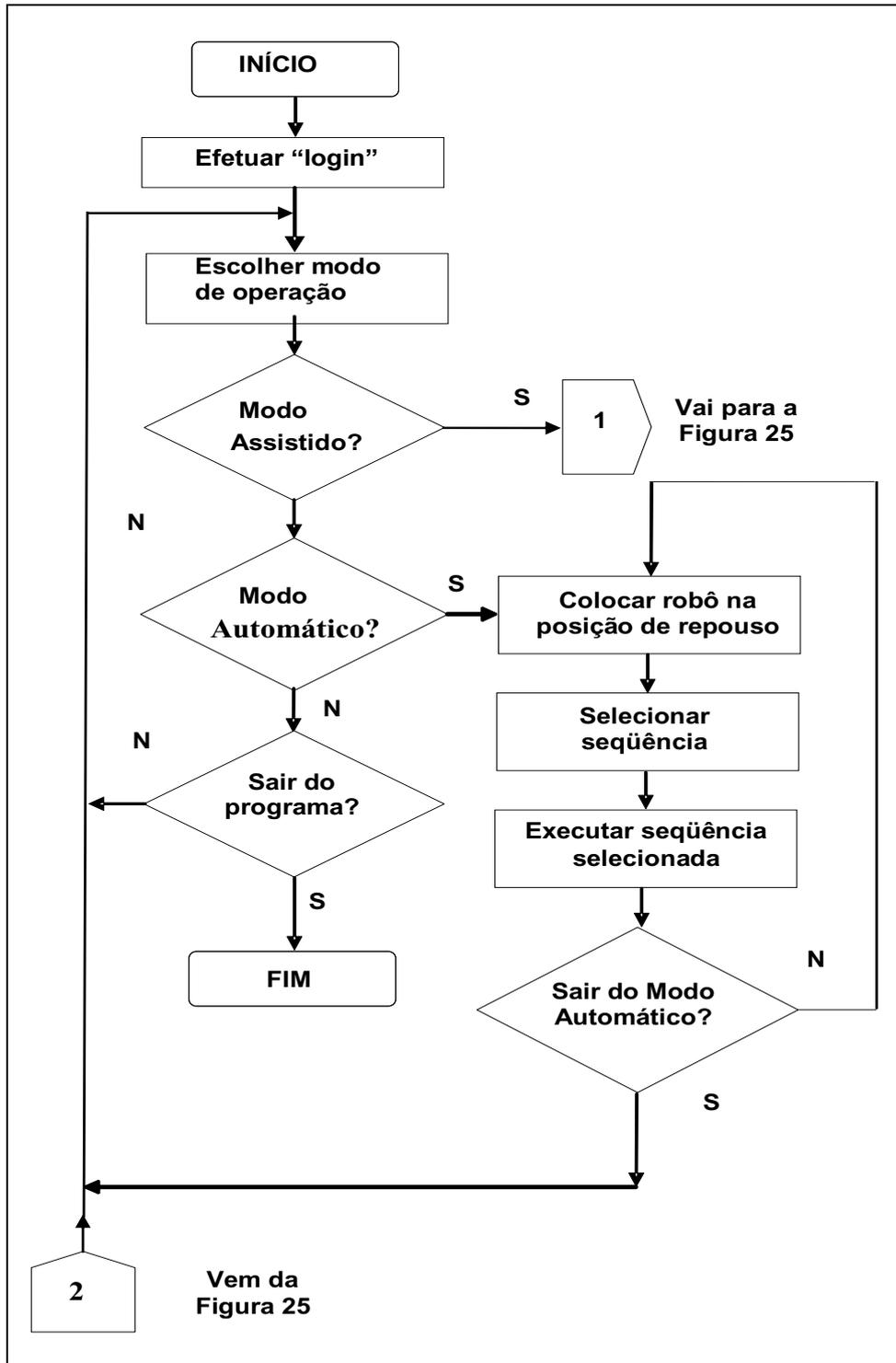


Figura 24 - Fluxograma modo automático

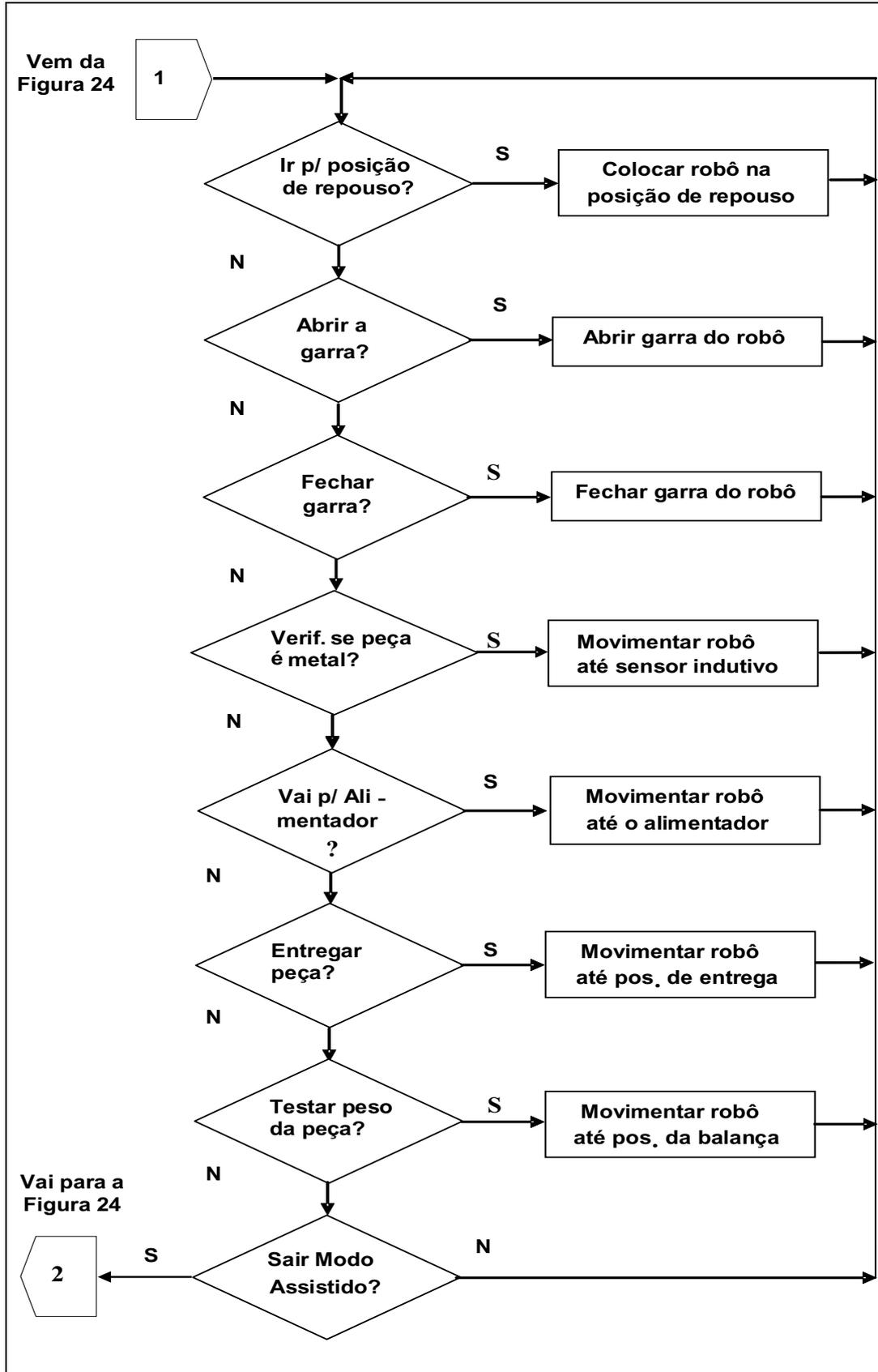


Figura 25 - Fluxograma modo assistido

3.3.3 Interface Gráfica

A principal função da Interface Homem-Máquina (IHM) está concentrada em servir de meio para o operador comandar, programar e supervisionar as operações relacionadas com a arquitetura proposta neste trabalho. Essa interface foi elaborada com a utilização de um sistema de Supervisão, Controle e Aquisição de Dados (*Supervisory Control And Data Acquisition - SCADA*) disponível no mercado brasileiro, (ELIPSE WINDOWS, 2006). Para esta fase foram elaboradas três janelas para atender essa interface, conforme apresentado a seguir:

- Janela que permite o acesso do operador aos recursos do sistema, com senha e usuário (*Login*).
- Janela para operação em Modo Assistido.
- Janela para operação em Modo Automático.

3.3.3.1 Janela *Login*

A janela para registro de senha e usuário (*Login*), mostrada na Figura 26 é um recurso que tem por meta restringir a utilização dos recursos do sistema aos usuários previamente cadastrados, para evitar operações indevidas com a célula de produção. Após a validação da senha e usuário, o operador pode selecionar entre o Modo Automático ou Assistido de operação do sistema.



Figura 26 - Janela para registro de senha e usuário

3.3.3.2 Janela Modo Assistido

A janela da interface gráfica para operar no Modo Assistido é apresentada na Figura 27 com o botão Modo Assistido LIGADO, acionado.



Figura 27 - Janela de Modo Assistido da interface gráfica

Nessa janela, após o acionamento do botão Modo Assistido DESLIGADO, a cor da mesma é alterada, como pode ser visto na Figura 27 e permite que o operador acesse um dos botões da interface, possibilitando, de acordo com o programa de gerenciamento, realizar as seguintes ações:

- Mover o robô para a posição de Repouso.
- Mover o robô para a posição do Alimentador.
- Abrir ou Fechar a Garra do Robô.
- Transportar a peça para a posição do Sensor indutivo.
- Transportar a peça para a posição da Balança.
- Transportar a peça para a posição de Entrega.
- Transportar a peça para as posições 1A, 2A 3A, 1B, 2B ou 3B.
- Mudar a operação para o modo Automático.
- Iniciar uma nova operação (Reset).
- Sair Programa.

Para que o operador tenha acesso a esse conjunto de recursos, o botão Modo Assistido DESLIGADO deve ser pressionado preliminarmente, caso contrário

os botões com as funções anteriormente descritas permanecem ocultos, como mostrado na Figura 28.



Figura 28 – Janela de Modo Assistido DESLIGADO

3.3.3.3 Janela Modo Automático

Na janela para o Modo Automático, o operador pode selecionar uma sequência de operações desejada para ser executada pelo robô. Para que isso seja possível, o botão de Modo Automático DESATIVADO deve ser pressionado, caso isso não ocorra, os botões relativos às opções de operações a serem realizadas pelo robô ficam ocultos, como mostra a Figura 29.

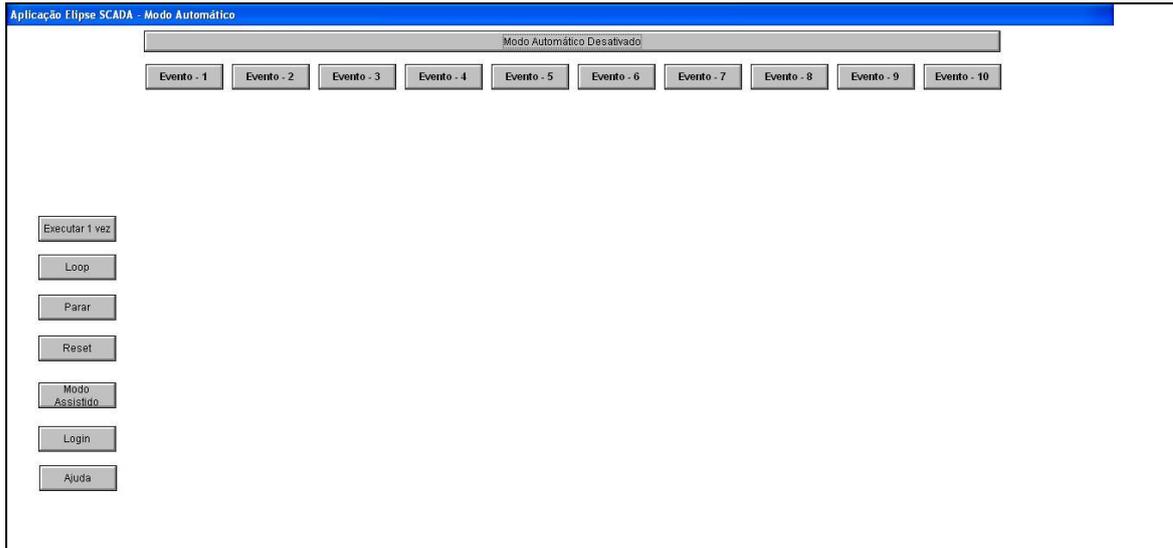


Figura 29 – Janela Modo Automático DESATIVADO

O acionamento do botão de operação “Modo Automático DESATIVADO”, permite o acesso aos recursos de programação de eventos. Ao ser ativado, o botão muda de cor, além de liberar o acesso ao Modo Automático.

Na Figura 30 (Coluna 1) são apresentados os componentes gráficos que permitem ao usuário selecionar os seguintes recursos:

- Habilitar a utilização dos botões do modo assistido (Modo Automático Desativado).
- Executar a sequência programada uma única vez.
- Executar a sequência programada constantemente (Loop).
- Parar a execução (Parar).
- Retornar a interface a sua condição original (Reset).
- Alterar para o Modo Assistido de operação.
- Sair do programa.



Figura 30 - Janela Modo Automático ATIVADO

As colunas subseqüentes dessa janela, na área denominada “Eventos”, destacadas na Figura 31 permitem que o operador programe uma seqüência de até dez operações que devem ser executadas pelo robô em modo automático.



Figura 31 – Seleção de eventos em Modo Automático

Ao selecionar uma sequência, o botão Evento, enumerado de 1 a 10, muda de cor e somente o botão escolhido por coluna pode ser visto. Os demais botões ficam ocultos como pode ser observado na Figura 32.

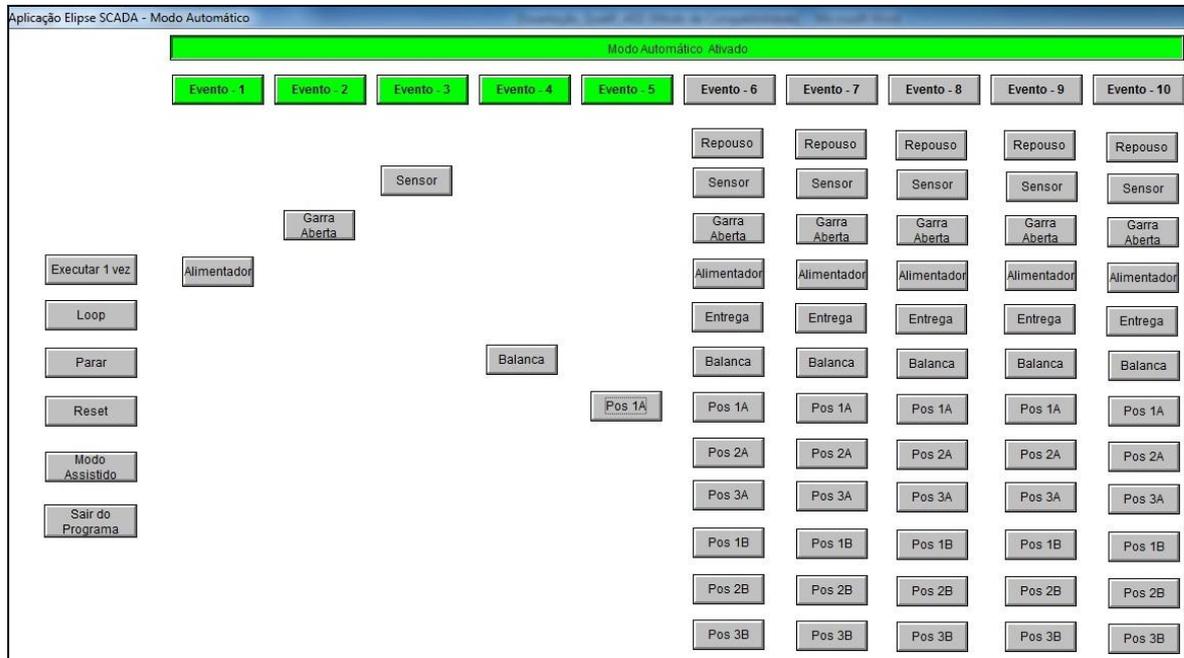


Figura 32 – Seleção na janela de Modo Automático

3.4 Testes Práticos

Para realizar os testes práticos de avaliação operacional do protótipo desenvolvido, foi implementada a arquitetura apresentada na Figura 18, e elaborado um programa em linguagem LADDER, para o sistema de desenvolvimento (S_D), o qual atende os passos contidos no fluxograma analítico mostrado na Figura 24 e Figura 25.

Os testes mostraram que a interface proposta é capaz de reconfigurar as tarefas que podem ser executadas por uma célula flexível de produção. Nos subitens seguintes são apresentados os principais resultados dos testes realizados com o protótipo.

3.4.1 Testes com o Modo Assistido

Inicialmente o sistema apresenta a janela de acesso (*login*), conforme pode ser observado na Figura 26. Efetuado o registro de usuário e a senha aceita pelo sistema, o operador tem acesso à janela do Modo Assistido, apresentada na Figura 27. Esse modo permite que o usuário avalie cada operação que a célula pode executar de modo individual. O primeiro teste, neste modo de operação, foi idealizado para que o robô retire uma peça do alimentador. A peça que deverá ser retirada é mostrada na Figura 33.



Figura 33 – Objeto utilizado para os testes do protótipo

A Figura 34 mostra a janela com a ação “Repouso” selecionada na interface gráfica.



Figura 34 - Interface com a ação “Repouso” selecionada

A consequência da seleção anteriormente efetuada é mostrada na Figura 35, na qual pode ser observado que o robô é movimentado para uma posição definida como “Repouso”, a partir da qual realizará os demais movimentos.



Figura 35 - Robô na posição definida como “Repouso”

A Figura 36 mostra a janela com a ação “Garra Aberta” selecionada na interface gráfica.



Figura 36 - Interface com a ação “Garra Aberta” selecionada

Essa seleção, mostrada na Figura 37, executa o movimento de abrir a garra do robô, que é automaticamente fechada após um tempo de 5 segundos.



Figura 37 - Robô com a “Garra Aberta”

A Figura 38 mostra a janela com a ação “Alimentador” selecionada na interface gráfica.



Figura 38 - Interface com a ação “Alimentador” selecionada

A consequência da seleção anteriormente efetuada é mostrada na Figura 39, na qual pode ser observado que o robô pega a peça no alimentador.



Figura 39 - Robô executando a ação “Alimentador”

O segundo teste tem por meta deslocar a peça da posição do alimentador para a posição do sensor indutivo, sendo que para tanto é necessário selecionar o botão “Sensor”, conforme mostrado na Figura 40.



Figura 40 - Interface com a ação “Sensor” selecionada

A consequência da seleção anteriormente efetuada é mostrada na Figura 41, sendo possível observar que o robô desloca a peça da posição do alimentador para a posição do sensor indutivo. Cabe mencionar que essa ação permite à célula testar

se a peça é metálica ou não metálica. A peça em questão possui arruelas metálicas o que a identifica como uma peça metálica.

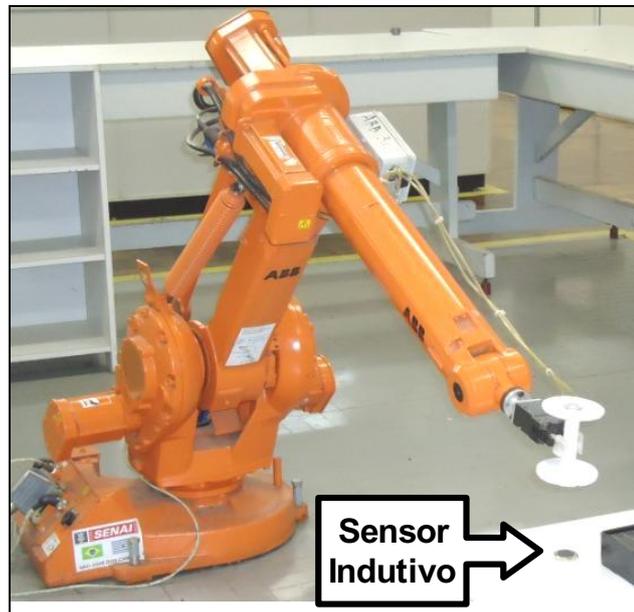


Figura 41 - Robô movendo a peça para a posição “Sensor”

Além de verificar a peça quanto a sua composição metálica, há possibilidade de testá-la com relação a sua massa. Para tanto está previsto o botão “Balança”, apresentado na Figura 42, que faz o robô deslocar a peça até a posição da balança.



Figura 42 - Interface com o botão “Balança” selecionado

A consequência da seleção anteriormente efetuada é mostrada na Figura 43, na qual pode ser observado que o robô desloca a peça para a posição da balança.



Figura 43 - Robô com a peça na posição “Balança”

Além dos deslocamentos já citados, as peças podem ser deslocadas para algumas posições que veremos a seguir. A primeira possibilidade, relativa ao botão “Entrega”, pode ser visualizada na Figura 44.



Figura 44 - Interface com o botão “Entrega” selecionado

A Figura 45 mostra o robô na posição “Entrega”, posição que pode ser utilizada, por exemplo, para o descarte de peças que não estejam em conformidade com algum dos parâmetros testados (peso ou metal).



Figura 45 - Robô com a peça na posição “Entrega”

O robô pode também ser deslocado para uma posição pré-determinada de armazenamento das peças, para isso o operador deve selecionar uma posição dentre 1A, 2A, 3A, 1B, 2B ou 3B. A Figura 46 ilustra a seleção do botão relativo à posição 1A.



Figura 46 - Interface com o botão “POS 1A” selecionado

A Figura 47 mostra o deslocamento do robô para a posição 1A no armazenador de peças.



Figura 47 - Robô com a peça na posição “POS 1A”

A Figura 48 ilustra a seleção do botão relativo à posição 2A.

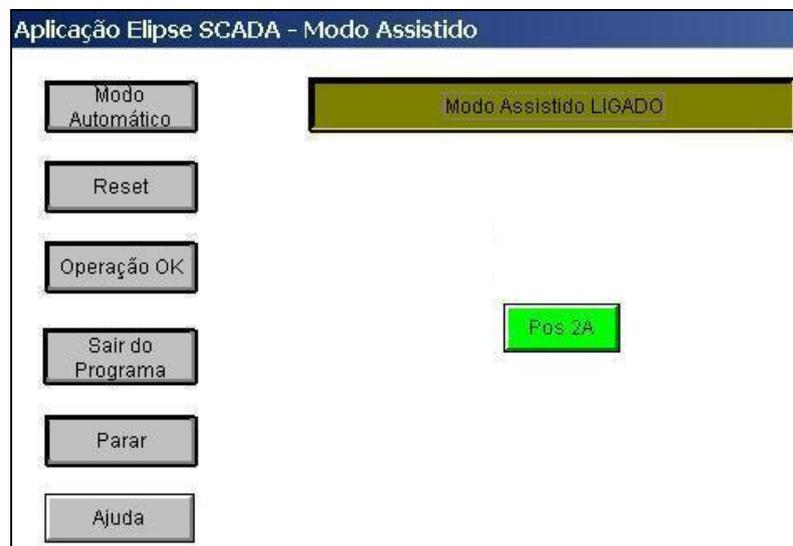


Figura 48 - Interface com o botão “POS 2A” selecionado

A Figura 49 mostra o deslocamento do robô para a posição 2A.



Figura 49 - Robô com a peça na posição “POS 2A”

A Figura 50 ilustra a seleção do botão relativo à posição 3A.



Figura 50 - Interface com o botão “POS 3A” selecionado

A Figura 51 mostra o deslocamento do robô para a posição 3A.



Figura 51 - Robô com a peça na posição “POS 3A”

A Figura 52 ilustra a seleção do botão relativo à posição 1B.

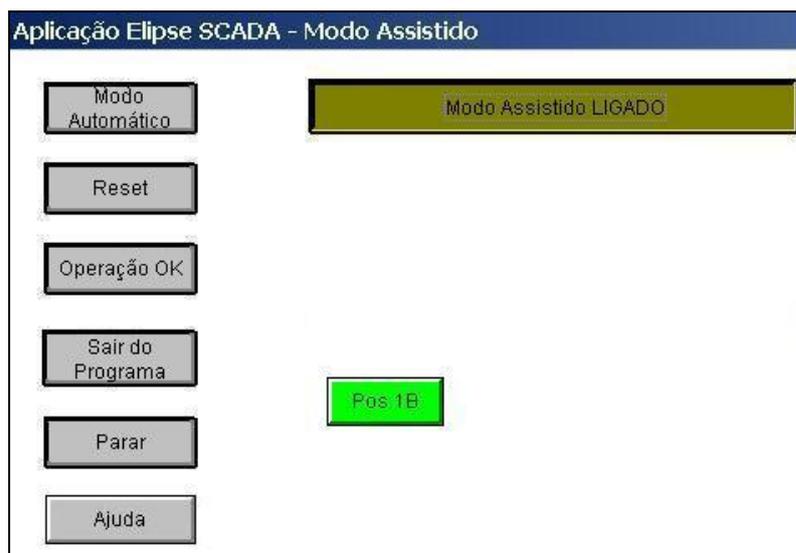


Figura 52 - Interface com o botão “POS 1B” selecionado

O deslocamento do robô para a posição 1B é ilustrado na Figura 53.



Figura 53 - Robô com a peça na posição “POS 1B”

A Figura 54 ilustra a seleção do botão relativo à posição 2B.



Figura 54 - Interface com o botão “POS 2B” selecionado

A Figura 55 mostra o deslocamento do robô para a posição 2B.



Figura 55 - Robô com a peça na posição “POS 2B”

A Figura 56 ilustra a seleção do botão relativo à posição 2A.



Figura 56 - Interface com o botão “POS 3B” selecionado

A Figura 57 mostra o deslocamento do robô para a posição 3B.



Figura 57 - Robô com a peça na posição “POS 3B”

3.4.2 Testes com o Modo Automático

No modo automático foram realizados testes com sequências pré-estabelecidas, simulando a seleção efetuada por um operador, como na Figura 58.

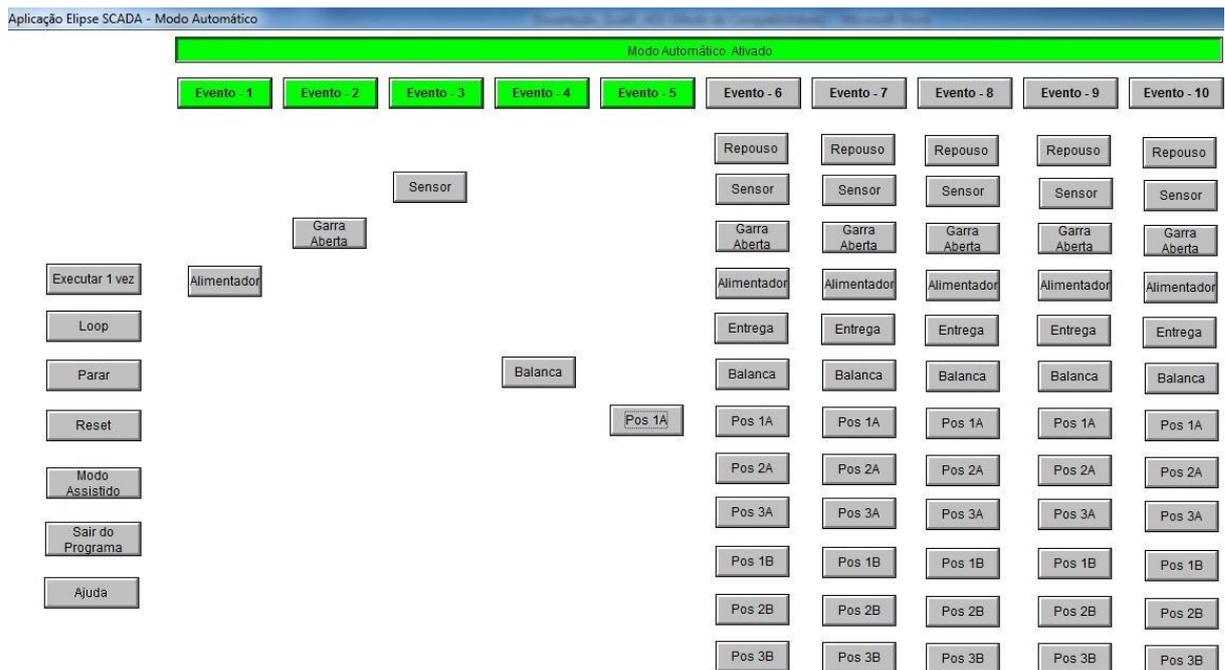
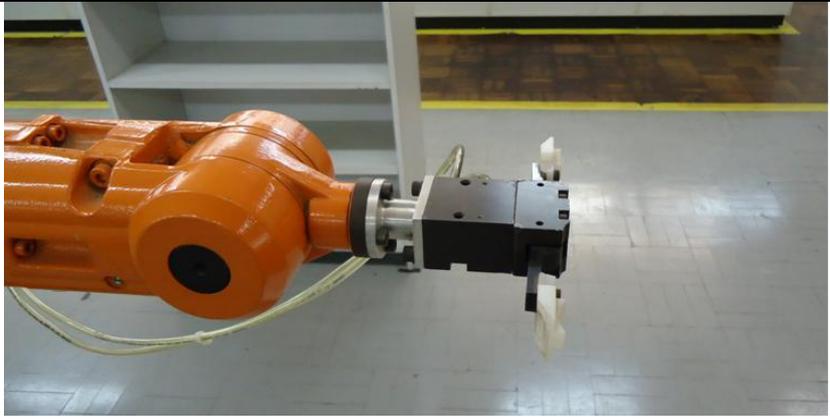
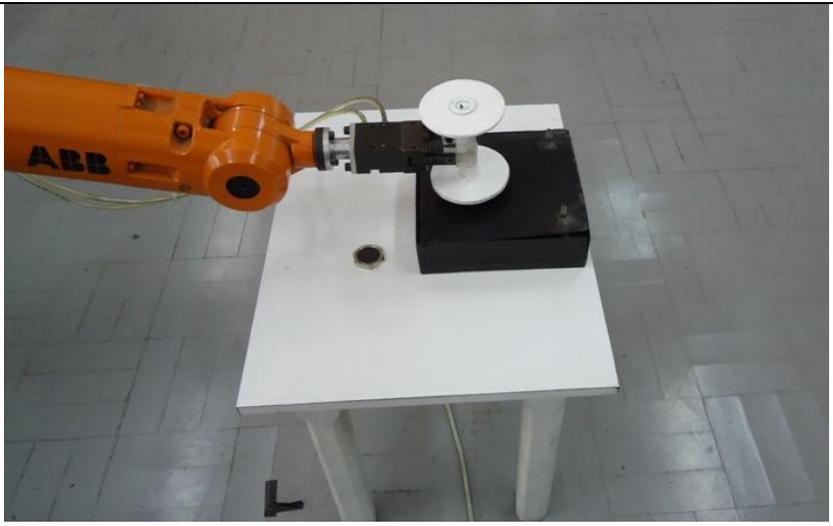


Figura 58 - Interface com sequência pré-selecionada

A Tabela 3 ilustra a execução da sequência pré-selecionada na interface da figura 58.

Tabela 3 - Execução da sequência pré-selecionada na interface no Modo Automático.

Evento	Ação	Execução
1	Alimentador	
2	Garra Aberta	

3	Sensor	 An orange ABB robotic arm is positioned over a white table. The end effector of the arm is holding a white, cylindrical sensor component. A black rectangular box is also visible on the table to the right of the sensor.
4	Balança	 An orange ABB robotic arm is positioned over a white table. The end effector is holding a black rectangular scale with a white circular weighing platform on top. The ABB logo is visible on the side of the arm.
5	POS. 1A	 An orange ABB robotic arm is shown in a specific position, labeled 'POS. 1A'. The arm is extended towards a white cabinet or structure. A small white label with 'ABB' and '1A' is attached to the arm. The background shows a factory floor with yellow safety lines.

3.4.3 Janela de ajuda

Para auxiliar o usuário na operação da interface, foi inserida uma janela para ajuda, a qual pode ser consultada a partir das janelas dos modos assistido e automático. A janela de ajuda apresenta informações sobre o modo de operação da arquitetura, conforme apresentado na Figura 59.

O programa de gerenciamento foi elaborado para atender dois principais modos de operação:

- Modo Assistido e
- Modo Automático.

A operação em **Modo Assistido** permite determinar a seqüência de acionamentos desejada pelo operador de acordo com o processo que deverá ser executado. O programa de gerenciamento possui rotinas que permitem ao operador comandar o Robô, deslocando-o para o pontos desejados por meio das seguintes opções:

- Mover o robô para a posição de Repouso.
- Mover o robô para a posição do Alimentador.
- Abrir ou Fechar a Garra do Robô.
- Transportar a peça para a posição do Sensor indutivo.
- Transportar a peça para a posição da Balança.
- Transportar a peça para a posição de Entrega.
- Transportar a peça para as posições 1A, 2A 3A, 1B, 2B ou 3B.
- Mudar a operação para o modo Automático.

Os recursos do **Modo Automático** permitem ao operador do sistema:

- Efetuar a programação de uma seqüência de operações que a célula flexível deverá executar automaticamente.
- Executar a seqüência programada apenas uma única vez.
- Executar repetidamente a seqüência programada.
- Reiniciar o sistema para uma nova programação.
- Paralisar a operação do sistema.

Em cada um destes modos é permitido ao operador optar por:

- Mudança de modo de operação.
- Ir para tela de ajuda.
- Finalizar o programa.

Figura 59 – Janela de AJUDA

CAPÍTULO 4 - CONCLUSÕES

Os resultados satisfatórios observados nos testes práticos mostram que a proposta apresentada neste trabalho é viável tecnicamente e pode ser levada a efeito para a aplicação a qual se destina.

Os objetivos previstos foram alcançados, principalmente no que concerne em propor uma arquitetura de sistema que utiliza a interface gráfica como elemento principal para reconfigurar tarefas em célula de manufatura flexível.

Os componentes previstos na interface gráfica são expressivos e de utilização intuitiva o que permite a um operador, não especializado na programação de sistemas robóticos, definir uma possível sequência de operações para ser executada na célula flexível.

A escolha do sistema CUBLOC para o módulo de controle da arquitetura mostrou-se adequada por: possuir custo moderado para aquisição, dispor de ambiente integrado de desenvolvimento gratuito e, principalmente, aumentar a flexibilidade da arquitetura proposta, uma vez que permite tratar sinais de outros dispositivos, como leitores de código de barras, dispositivos de identificação por radiofrequência, entre outros.

A arquitetura proposta se alinha com os princípios da Manufatura Enxuta, permitindo que o operador tenha uma maior participação no processo produtivo, além de possibilitar a diminuição do tempo de parada da célula.

REFERÊNCIAS

ABB, 2003. **Especificação do produto S4Cplus ABB robotics products AB**. Vasterås, Sweden, 2003. 43p.

ALLEN-BRADLEY. **Catálogo de controlador lógico programável, modelos Micro810 e Micro850**. Disponível em: <http://literature.rockwellautomation.com/idc/groups/literature/documents/br/2080-br001_-en-p.pdf>. Acesso em: 04 ago. 2011.

BARROS, H. S. **Utilização de ferramentas da manufatura enxuta na melhoria dos resultados da GM powertrain de São José dos Campos**. Disponível em: <http://www.automotiva-poliusp.org.br/mest/banc/pdf/barros_hugo.pdf>. Acesso em 02 nov. 2011.

CARNEIRO, F. L. **O sistema de produção enxuta e sua implantação na Volkswagen do Brasil**. Disponível em: <http://www.ogerente.com.br/novo/artigos_ler.php?canal=13&canallocal=45&canalsub2=144&id=177>. Acesso em: 04 nov. 2011.

CLARK, G; REYNDERS, D. **Modern SCADA protocols: DNP3, IEC 60870-5 and related systems**. Newnes, 2004.

COEL. **Controles elétricos Ltda**. Disponível em: <http://www.syar.com.uy/pdf/coel/inductivos_capacitivos.pdf>. Acesso em: 27 de ago. 2011.

COMFILE TECHNOLOGY INC. 2006. Disponível em: <<http://www.comfiletech.com>>. Acesso em 15 mai. 2010.

CUBLOC. **User manual version 2.0.0**. Comfile technology©, 2006.

ELIPSE. Disponível em: <<http://www.elipse.com.br/imagens/produtos/SCADA/scadawin1.png>>. Acesso em: 06 jul. 2011.

ELIPSE Windows. **Sistema de supervisão e controle - Elipse software manual do usuário**, 2006.

ELIPSE software. **ELIPSE Scada – supervisory control and data acquisition system**. Manual do Usuário, 2004.

FRANCHI, C. M.; CAMARGO, L. A. **Controladores lógicos programáveis – sistemas discretos**. São Paulo: Érica, 2011.

FONSECA, M. de Oliveira et al. **Aplicando a norma IEC 61131 na automação de processos**. São Paulo : ISA Distrito 4, 2008.

GEORGINI, M. **Automação aplicada: descrição e implementação de sistemas seqüenciais com PLCs**. São Paulo: Érica, 2007.

GROOVER, M. **Automação industrial e sistemas de manufatura**. São Paulo: Prentice Hall, 2011.

MIYAGI, P. E. **Controle programável – fundamentos do controle de sistemas a eventos discretos**. São Paulo: Edgard Blücher, 1996.

MODBUS. Disponível em: <http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf>. Acesso em: 27 ago. 2011.

MORAES, C. C. ; CASTRUCCI, P. L. **Engenharia de automação industrial**. Rio de Janeiro: LTC, 2001.

MOTOROLA. **Folha de dados do optoacoplador Motorola 4N25**. Disponível em: <<http://www.datasheetcatalog.org/datasheet/motorola/4N26.pdf>>. Acesso em: 27 ago. 2011.

PEREIRA, M. C.; BRITO, M. J.; SILVA. **O processo de trabalho nas células de montagem e aprendizagem organizacional: um estudo na indústria automobilística**, Bauru, SP, 2005. XII SIMPEP. Disponível em: < http://www.simpep.feb.unesp.br/anais/anais_12/copiar.php?arquivo=Pereira_MC_O%20processo%20 >. Acesso em: 17 mar. 2011.

PINTO, L.A.V. **Artigo sobre opto acopladores**. Disponível em: <<http://www.vargasp.net/download/livros/Automac4.pdf>. >. Acesso em: 27 ago. 2011.

ROBÔ IRB 1410. **ABB**, Vasterås, Sweden, 06/09/2011. Disponível em: <[http://www04.abb.com/global/gad/gad02007.nsf/0/ADCA01B5C8640AE8C1256EC30068FC4F/\\$File/irb1400_12_720.jpg](http://www04.abb.com/global/gad/gad02007.nsf/0/ADCA01B5C8640AE8C1256EC30068FC4F/$File/irb1400_12_720.jpg)>. Acesso em: 06 set. 2011.

ROSÁRIO, J. M. **Princípios de mecatrônica**. São Paulo: Person Prentice Hall, 2005.

SILVEIRA, P. R., & SANTOS, W. E. **Automação – controle discreto**. São Paulo: Érica, 2002.

SCHNEIDER-ELECTRIC. Disponível em: < [http://www.global-download.schneider-electric.com/85257578007E5C8A/all/384E41D3B6363C1E88257578006AF5B3/\\$File/144717401a55.pdf](http://www.global-download.schneider-electric.com/85257578007E5C8A/all/384E41D3B6363C1E88257578006AF5B3/$File/144717401a55.pdf) >. Acesso em: 06 ago. 2011.

THOMAZINI, D; ALBUQUERQUE, P. U. B. **Sensores industriais – fundamentos e aplicações**. São Paulo: Érica, 2005.

VIEIRA, Sérgio. **Sistema integrado de manufatura: a eletrônica é uma das responsáveis pela necessidade de alterar a forma de trabalho na manufatura**. *Mecatrônica Atual*, ano 6, n. 34, jun. jul. 2007.

WOMACK, J. P.; JONES, D. T.; ROOS, D. **A máquina que mudou o mundo**. Rio de Janeiro: Campus, 2004.