

ESTE EXEMPLAR CORRESPONDE A REDAÇÃO FINAL DA
TESE DEFENDIDA POR João Vilhete
Viegas D'Abreu E APROVADA PELA
COMISSÃO JULGADORA EM 17/10/02

João Maurício Rosário
ORIENTADOR

**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO**

Integração de Dispositivos Mecatrônicos para Ensino-Aprendizagem de Conceitos na Área de Automação

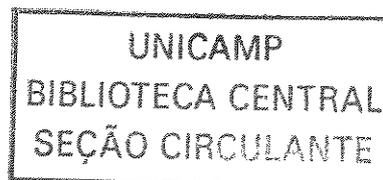
Autor:

João Vilhete Viegas D'Abreu

Orientador:

Prof. Dr. João Maurício Rosário

27/02



UNICAMP
BIBLIOTECA CENTRAL

**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO**

Integração de Dispositivos Mecatrônicos para Ensino-Aprendizagem de Conceitos na Área de Automação

Autor:

João Vilhete Viegas d'Abreu

Orientador:

Prof^o. Dr. João Maurício Rosário

Área de Concentração:

Mecânica dos Sólidos e Projeto Mecânico

Tese de doutorado apresentada à comissão de Pós-Graduação da Faculdade de Engenharia Mecânica, como requisito para obtenção do título de Doutor em Engenharia Mecânica.

Campinas, 2002

S.P. - Brasil

UNIDADE Be
Nº CHAMADA T/UNICAMP
D114i
V _____ EX _____
TOMBO BCI 51025
PROC 16.837102
C _____ D^x _____
PREÇO R\$ 11,00
DATA 27/09/02
Nº CPD _____

CM00174321-B

BIB ID 259189

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

D114i D'Abreu, João Vilhete Viegas
Integração de dispositivos mecatrônicos para ensino-
aprendizagem de conceitos na área de automação / João
Vilhete Viegas D'Abreu. --Campinas, SP: [s.n.], 2002.

Orientador: João Maurício Rosário.
Tese (doutorado) - Universidade Estadual de
Campinas, Faculdade de Engenharia Mecânica.

1. Automação. 2. Robótica. 3. Ensino. 4.
Aprendizagem. 5. Conceitos. I. Rosário, João Maurício.
II. Universidade Estadual de Campinas. Faculdade de
Engenharia Mecânica. III. Título.

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM
ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO

TESE DE DOUTORADO

**Integração de Dispositivos Mecatrônicos para
Ensino-Aprendizagem de Conceitos na
Área de Automação**

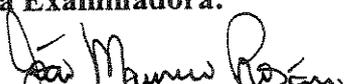
Autor:

João Vilhete Viegas d'Abreu

Orientador:

Prof. Dr. João Maurício Rosário

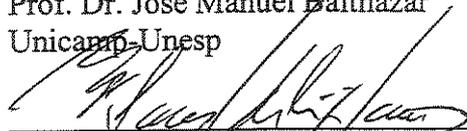
Banca Examinadora:



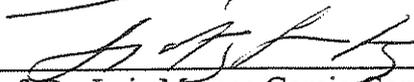
Prof. Dr. João Maurício Rosário
Unicamp



Prof. Dr. José Manuel Balthazar
Unicamp-Unesp



Prof. Dr. Klaus Schlünz Junior
Unesp



Prof. Dr. Luiz Marcos Garcia Gonçalves
UFRN



Prof. Dr. Vitor Ferreira Romano
UFRJ

Campinas, julho de 2002

Este trabalho é dedicado aos meus filhos
Frederico, Hugo, Daniel e Luiz Felipe por
serem a minha fonte de motivação.

Agradecimentos

É difícil listar todas as pessoas que de alguma forma colaboraram para que este trabalho tivesse resultado nesta Tese de Doutorado, pois de alguma forma em momentos específicos a ajuda de cada uma delas foi muito importante. Sendo assim, gostaria de agradecer:

A todos os professores e colegas do Departamento de Projeto Mecânico e do Laboratório de Ensino Pesquisa em Automação e Robótica – LAR.

À coordenação do Núcleo de Informática Aplicada à Educação – NIED e a todos os meus colegas de trabalho pelo apoio e incentivo.

Aos alunos das Engenharias Mecânica e Mecatrônica Unicamp, de Engenharia e Ciências da Computação da Unicamp e aos Técnicos Especializados da Delphi-Harrison pela colaboração na parte experimental deste trabalho.

Ao professor João Maurício Rosário, pela orientação, pelas suas sugestões diretas e objetivas em relação a este trabalho.

Em especial, à minha esposa Maria de Fátima pelo carinho, paciência, estímulo e colaboração.

Resumo

Este trabalho teve como objetivo o estudo e aprendizado de conceitos na área de mecatrônica a partir do processo de concepção, estruturação e programação de dispositivos mecânicos, utilizando kits educacionais de montagem LEGO, dispositivos robóticos Robix e outros materiais de padrão comercial ou não. Este trabalho foi desenvolvido no Núcleo de Informática Aplicada à Educação – NIED/UNICAMP e no Laboratório de Ensino e Pesquisa em Automação e Robótica – LAR, da Faculdade de Engenharia Mecânica - FEM, no Departamento de Projeto Mecânico – DPM/UNICAMP. No NIED, realizou-se oficinas de trabalho junto aos técnicos especializados de uma fábrica. No LAR/FEM, realizou-se atividades didáticas junto aos alunos de graduação nos cursos de Engenharia Mecânica, Engenharia de Controle e Automação (Mecatrônica) e Engenharia e Ciências da Computação. A partir de atividades práticas realizadas com os técnicos especializados na fábrica e com esses alunos, desenvolveu-se uma série de análises que buscam comprovar quanto promissor pode ser um ambiente de ensino-aprendizagem que utiliza implementação de dispositivos mecatrônicos, como forma de se aprimorar o aprendizado de conceitos em qualquer nível de ensino. O trabalho analisa a possibilidade de se transformar as atividades desenvolvidas, utilizando LEGO e outros materiais, em uma disciplina do currículo regular de graduação, pois, atualmente, no curso de Mecatrônica, os conceitos são apresentados de forma fragmentada em diferentes disciplinas. Essa disciplina deverá iniciar-se no primeiro ano do curso e ter a sua continuidade nos anos subseqüentes. Com base em trabalhos práticos experimentais a tese apresenta propostas que possibilitam integrar os diferentes conceitos que o aluno aprende durante o processo da sua formação. A tese analisa situações de aprendizagem que possibilitam com que esses conceitos sejam apresentados de forma contextualizada, assim contribuindo com o aprendizado acadêmico e também com a integração mais rápida do aluno nas diversas vertentes profissionais para as quais a formação em Mecatrônica se direciona.

Abstract

The aim of this work is to study and the learning concepts in mechatronic area from the process of conception, structuring and programming of mechanical devices, using educational assembling kits such as LEGO, robotics devices Robix and others, whether of commercial standards or not. This work was developed at NIED/UNICAMP, the Nucleus of Informatics Applied to Education, and at LAR, the Lab of Teaching and Research in Automation and Robotics from the Mechanical Engineering Faculty (FEM), at DPM/UNICAMP, the Department of Mechanical Project. At NIED, workshops were held with factory specialized technicians. At LAR/FEM, pedagogical activities were carried out with undergraduate students from Mechanical Engineering, Control and Automation Engineering (Mechatronic) and the Computing Science School. After the practical activities were conducted with factory specialized technicians and with these students, a number of analyses were developed in order to show that the teaching-learning environment using the implementation of mechatronics devices can improve the learning of concepts at any level of teaching. This work analyze the possibility of transforming the developed activities, using LEGO and the other materials, into a subject in the undergraduate curriculum, since, at present, the concepts are presented in a fragmented way in different subjects. This proposed new subject should begin in the first college year and continue in the following years. Based on practical and experimental work, this thesis presents suggestions on how to integrate the different concepts that the students acquire in their learning process. The thesis analyses learning situations that enable such concepts to be presented in a contextualized way, therefore, contributing to the academic learning as well as to a faster integration of the student with the various professional areas that the Mechatronics course, is involved with.

ÍNDICE

| | |
|--|----|
| RESUMO..... | IX |
| ABSTRACT..... | XI |
| INTRODUÇÃO..... | 3 |
| I.1. APRESENTAÇÃO DO PROBLEMA DE PESQUISA..... | 3 |
| I.2. QUESTÃO A INVESTIGAR..... | 7 |
| I.3. JUSTIFICATIVA..... | 7 |
| I.4. OBJETIVOS GERAIS..... | 9 |
| I.5. OBJETIVOS ESPECÍFICOS..... | 9 |
| I.6. DELIMITAÇÃO DA INVESTIGAÇÃO..... | 9 |
| I.7. ESTRUTURA DO TRABALHO..... | 10 |
| CAPÍTULO 1..... | 15 |
| CONCEITOS BÁSICOS DE AUTOMAÇÃO E ROBÓTICA..... | 15 |
| 1.1 AUTOMAÇÃO..... | 15 |
| 1.2 AUTOMAÇÃO ALTERANDO OS MEIOS DE TRABALHO..... | 17 |
| 1.3 AUTOMAÇÃO E EDUCAÇÃO DA CLASSE TRABALHADORA..... | 23 |
| 1.4 QUALIFICAÇÃO..... | 25 |
| 1.5 POLIVALÊNCIA E POLITECNIA..... | 27 |
| 1.6 ASPECTOS ECONÔMICOS EM ESTRUTURAS FLEXÍVEIS..... | 30 |
| 1.7 TECNOLOGIA E ORGANIZAÇÃO DO TRABALHO..... | 32 |
| 1.8 ESPECIFICAÇÃO DE PROJETOS EM AUTOMAÇÃO..... | 33 |
| 1.8.1 <i>Objetivo das especificações</i> | 35 |
| 1.8.2 <i>Características das especificações</i> | 35 |
| 1.8.3 <i>Especificações preliminares e finais</i> | 36 |
| 1.9 CONSIDERAÇÕES SOBRE O CAPÍTULO 1..... | 37 |
| CAPÍTULO 2..... | 41 |
| AUTOMAÇÃO NO CONTEXTO DE AMBIENTES DE APRENDIZAGEM..... | 41 |
| 2.1 O QUE É A INFORMÁTICA NA EDUCAÇÃO..... | 41 |
| 2.2 AMBIENTE DE ROBÓTICA PEDAGÓGICA..... | 42 |
| 2.3 AMBIENTE DE ROBÓTICA PEDAGÓGICA LEGO-LOGO..... | 44 |
| 2.3.1 <i>O que se ganha em termos de aprendizagem quando se cria o ambiente LEGO-Logo?</i> | 44 |
| 2.4 LEGO-LOGO NA ESCOLA..... | 45 |
| 2.5 LEGO-LOGO NA FÁBRICA..... | 46 |
| 2.6 IMPLEMENTAÇÃO DE AMBIENTES EDUCACIONAIS: UTILIZANDO SOFTWARES DIVERSOS..... | 47 |
| 2.6.1 <i>Ambiente Everest</i> | 47 |
| 2.6.2 <i>Ambiente Robolab</i> | 48 |
| 2.6.3 <i>Ambiente Coach Júnior</i> | 50 |
| AMBIENTE DE PROGRAMAÇÃO DO COACH JÚNIOR..... | 50 |
| 2.7 IMPLEMENTAÇÃO DE AMBIENTES EDUCACIONAIS: COM BASE NO LOGO..... | 51 |
| 2.7.1 <i>Ambiente LEGO-Logo para MSX</i> | 51 |
| 2.7.2 <i>Ambiente LEGO TcLogo</i> | 52 |
| 2.7.3 <i>Ambiente LEGO-Logo para Control Lab</i> | 54 |
| 2.8.4 <i>Ambiente LEGO Mindstorms Robotic Control X - RCX</i> | 55 |
| SOFTWARE..... | 56 |
| 2.8.5 <i>Ambiente SuperI</i> | 57 |
| 2.8.6 <i>Ambiente SIROS (Sistemas Robóticos com SuperLogo)</i> | 63 |
| 2.8.7 <i>Ambiente de Telerobótica</i> | 65 |
| 2.9 ESTADO DA ARTE..... | 67 |

| | |
|--|------------|
| 2.10 CONSIDERAÇÕES SOBRE O CAPÍTULO 2..... | 71 |
| CAPÍTULO 3 | 75 |
| SISTEMAS AUTOMATIZADOS COM ÊNFASE NO USO EDUCACIONAL..... | 75 |
| 3.1 SISTEMAS AUTOMATIZADOS..... | 75 |
| 3.2 LINGUAGENS DE PROGRAMAÇÃO PARA CONTROLADOR LÓGICO PROGRAMÁVEL – CLP | 76 |
| 3.3 MODELAGEM DE SISTEMAS AUTOMATIZADOS | 77 |
| 3.3.1 O QUE É GRAFCET..... | 78 |
| 3.3.2 <i>Grafcet de nível 1 – Descrição Funcional</i> | 79 |
| 3.3.3 <i>Grafcet de nível 2 – Descrição Tecnológica</i> | 79 |
| 3.3.4 <i>Grafcet de nível 3 – Descrição Operativa</i> | 79 |
| 3.4 CARACTERÍSTICAS DO GRAFCET..... | 79 |
| 3.5 ELEMENTOS DE UM GRAFCET | 80 |
| 3.6 DESCRIÇÃO DE ELEMENTOS CONSTITUINTES DE SISTEMAS AUTOMATIZADOS | 80 |
| 3.6.1 <i>Sistema em Malha Aberta</i> | 82 |
| 3.6.2 <i>Função de Transferência Malha Aberta</i> | 82 |
| 3.6.3 <i>Sistema em Malha Fechada</i> | 83 |
| 3.6.4 <i>Função de transferência em Malha Fechada</i> | 83 |
| 3.7 SISTEMAS DE AÇIONAMENTO | 85 |
| 3.8 REDUTOR..... | 87 |
| 3.9 UTILIZAÇÃO DE SENSORES E ATUADORES EM AUTOMAÇÃO | 88 |
| 3.9.1 <i>Sensores do ponto de vista tecnológico</i> | 89 |
| 3.10 SENSOR USADO COMO TRANSDUTOR | 90 |
| 3.10.1 <i>Foto transistor</i> | 90 |
| 3.10.2 <i>Interruptor de lâminas</i> | 91 |
| 3.10.3 <i>Sensor de reflexão</i> | 92 |
| 3.10.4 <i>Sensor de interrupção de luz</i> | 92 |
| 3.10.5 <i>Foto Diodo</i> | 92 |
| 3.11 CARACTERÍSTICAS DE SENSORES | 93 |
| 3.12 SENSORES DE USO EDUCACIONAL | 93 |
| 3.12.1 <i>Sensores LEGO</i> | 94 |
| 3.13 ADAPTADOR LEGO DCP | 96 |
| 3.14 ATUADORES | 98 |
| 3.14.1 <i>Atuadores do ponto de vista tecnológico</i> | 98 |
| 3.15 COMO OS ATUADORES SÃO UTILIZADOS..... | 99 |
| 3.15.1 <i>Atuador como motor</i> | 99 |
| 3.15.2 <i>Atuador como eletroímã</i> | 100 |
| 3.15.3 <i>Atuador como freio magnético</i> | 100 |
| 3.15.4 <i>Atuador como fechadura magnética</i> | 100 |
| 3.15.5 <i>Atuador como válvula solenóide</i> | 100 |
| 3.16 ATUADORES DE USO EDUCACIONAL..... | 101 |
| 3.17 CONSIDERAÇÕES SOBRE O CAPÍTULO 3..... | 101 |
| CAPÍTULO 4 | 107 |
| PROJETOS INTEGRADOS UTILIZADOS PARA VALIDAÇÃO DE CONCEITOS..... | 107 |
| 4.1 MONTAGEM E AUTOMAÇÃO DE DISPOSITIVOS MECATRÔNICOS..... | 107 |
| 4.2 MOTIVAÇÃO DO PROJETO PRÁTICO EXPERIMENTAL..... | 109 |
| 4.2.1 <i>Atividade de Pesquisa Prática Realizada</i> | 109 |
| 4.3 SISTEMA INTEGRADO PARA TRIAGEM DE PRODUTOS EM UMA LINHA AUTOMATIZADA DE PRODUÇÃO (DESENVOLVIDO NA UNIVERSIDADE) | 110 |
| 4.3.1 <i>Em que consistiu o projeto</i> | 111 |
| 4.3.2 <i>Divisão da Plataforma</i> | 111 |
| 4.3.3 <i>Sistema de Transferência de Peças</i> | 112 |
| 4.3.4 <i>Sistema de Reconhecimento de Cor</i> | 113 |

| | |
|--|------------|
| 4.3.5 Sistema de Separação de Peças | 114 |
| 4.3.6 Funcionamento do Sistema Integrado | 114 |
| 4.3.7 Software | 117 |
| 4.3.8 Conceitos abordados | 119 |
| 4.4 UTILIZANDO O SISTEMA LEGO-LOGO E ROBIX: SISTEMA AUTOMATIZADO DE PRODUÇÃO SAP (DESENVOLVIDO NA UNICAMP) | 120 |
| 4.4.1 Vantagem do ponto de vista educacional..... | 121 |
| 4.4.2 Linha de produção composta de três postos | 122 |
| 4.4.3 Aspectos do funcionamento de célula de produção | 123 |
| 4.4.4 Conceitos abordados | 124 |
| 4.5 MÁQUINA DE SOLDA AUTOMÁTICA (DESENVOLVIDO NA FÁBRICA)..... | 125 |
| 4.5.1 Máquina de Solda por Ponto | 125 |
| 4.5.2 Sistema mecânico da máquina de solda..... | 126 |
| 4.5.3 Automação da Máquina de Solda | 127 |
| 4.5.4 Programas elaborados em TcLogo..... | 128 |
| to controle..... | 129 |
| 4.5.5 Apresentação do Projeto pelos Técnicos Especializados | 129 |
| 4.5.6 Conceitos abordados e Considerações sobre o projeto..... | 130 |
| 4.6 CONSIDERAÇÕES SOBRE O CAPÍTULO 4..... | 131 |
| CAPÍTULO 5 | 135 |
| PROPOSTA DE DISCIPLINA DE FORMAÇÃO NA ÁREA DE AUTOMAÇÃO | 135 |
| 5.1 TRABALHO DESENVOLVIDO NAS ENGENHARIAS MECÂNICA E MECATRÔNICA | 136 |
| 5.1.1 Exemplo de projetos desenvolvidos com alunos da Engenharia Mecânica..... | 138 |
| 5.1.2 Exemplo de projeto desenvolvido com alunos da Engenharia Mecatrônica | 142 |
| 5.1.3 Reflexão sobre os projetos desenvolvidos nas Engenharias | 148 |
| 5.2 ANÁLISE DO TRABALHO DESENVOLVIDO EM CIÊNCIAS E ENGENHARIA DE COMPUTAÇÃO..... | 150 |
| 5.3 APRESENTAÇÃO DA DISCIPLINA | 151 |
| 5.3.1 Objetivos | 152 |
| 5.3.2 Pré-requisitos..... | 152 |
| 5.3.3 Avaliação do Aprendizado | 152 |
| 5.3.4 Trabalhos Práticos..... | 152 |
| 5.3.5 Caderno de Laboratório | 153 |
| 5.3.6 Seminário..... | 153 |
| 5.3.7 Competição | 153 |
| 5.3.8 Página WWW..... | 153 |
| 5.3.9 Aulas Teóricas | 154 |
| 5.3.10 Aulas Práticas..... | 154 |
| 5.4 LEVANTANDO DADOS MULTICATEGÓRICOS..... | 155 |
| 5.4.1 Categoria 1: Diferentes Heurísticas para Solucionar o Problema | 156 |
| 5.4.2 Categoria 2: Utilização de conceitos de outras disciplinas..... | 164 |
| 5.4.3 Categoria 3: Solução por software..... | 167 |
| 5.4.4 Categoria 4: Cooperação entre os colegas | 169 |
| 5.4.5 Categoria 5: Identificação dos Robôs..... | 174 |
| 5.4.6 Categoria 6: Aproveitando Idéias..... | 175 |
| 5.5 CONSIDERAÇÕES SOBRE O CAPÍTULO 5..... | 177 |
| CAPÍTULO 6 | 183 |
| CONSIDERAÇÕES FINAIS E PERSPECTIVAS FUTURAS | 183 |
| 6.1 CONSIDERAÇÕES | 183 |
| 6.2 PROPOSTAS FUTURAS | 191 |
| REFERÊNCIAS BIBLIOGRÁFICAS | 195 |

Introdução

A parte introdutória desta tese pretende apresentar um panorama geral do trabalho desenvolvido, fazendo um convite ao leitor para a leitura dos capítulos em que ela se constitui. Apresenta-se aqui, o problema pesquisado, justificativa para se desenvolver esta pesquisa, seus objetivos gerais e específicos, a questão que foi investigada, a forma como foi delimitada esta questão e a estrutura geral dos capítulos da tese¹.

I.1. Apresentação do problema de pesquisa

A proposta de integração de dispositivos mecatrônicos para ensino-aprendizagem de conceitos científicos, passa necessariamente pela utilização do computador como ferramenta educacional que, por sua vez, diz respeito ao enriquecimento do ambiente de ensino-aprendizagem onde se objetiva estudar conceitos científicos² como conceitos reais, passíveis de serem trabalhados/manuseados durante o processo de montagem automação e controle de dispositivos mecatrônicos. O enriquecimento do ambiente de ensino-aprendizagem pode ser visto, nesse contexto, como o grande desafio do qual este trabalho se originou. Nesse processo, a tentativa de se desenvolver atividades de pesquisa na área de hardware educacional, se direcionou para desenvolvimento de projetos e montagem de circuitos eletrônicos, concepção, estruturação e programação dos dispositivos, inicialmente chamados de robóticos. A implementação de dispositivos robóticos implicou na elaboração de software para permitir o controle destes

¹ Optou-se em redigir este trabalho de forma impessoal ou na terceira pessoa do singular, ou ainda na terceira pessoa do plural dependendo do contexto a ser descrito.

² Segundo Vygotski (1991), conceitos científicos são conceitos reais. Nos conceitos científicos que o sujeito adquire na escola, a relação com um objeto é mediada, desde o início, por algum outro conceito. (Vygotski, *in Pensamento e Linguagem*, pg. 80, 1991).

dispositivos por meio do computador. O propósito em utilizar esses dispositivos com finalidades educacionais, que é o grande objetivo a ser alcançado, levou à elaboração de metodologias que propiciam o uso pedagógico desses dispositivos. Todo esse processo aconteceu, não de forma seqüenciada, mas sim num movimento de "vai e vem" envolvendo um constante fazer, refazer, resignificar, reformular de idéias. À medida que essas idéias foram se consolidando o desafio, que a princípio se dava nos âmbitos de ensino fundamental e médio passou a se dar também no terceiro grau desenvolvendo atividades junto aos alunos de cursos de especialização e mais recentemente junto aos alunos de cursos regulares de graduação na Unicamp. Este trabalho foi desenvolvido num contexto de implementação e utilização de dispositivos mecatrônicos com propósitos educacionais junto aos alunos de graduação dos cursos de Engenharia Mecânica, Engenharia de Controle e Automação, Ciências de Computação, todos da Unicamp e, também técnicos especializados em uma fábrica. Em termos gerais as atividades de implementação e utilização de ambientes de ensino-aprendizagem foram desenvolvidos em dois contextos de aprendizagem: o formal e o não-formal. Contexto formal entendido para fins deste estudo como sendo aquele em que os ambientes foram desenvolvidos em locais caracterizados especificamente para o ensino: escolas de ensino fundamental e médio e universidades. Contexto não-formal entendeu-se como sendo os ambientes desenvolvidos em locais diferentes de escolas e universidades tais como: fábrica, museus, parques de diversões, feiras, dentre outros.

Um dos principais objetivos do curso de Engenharia de Controle e Automação (Mecatrônica) da Faculdade de Engenharia Mecânica da Unicamp - FEM é formar engenheiros com habilitação em controle e automação que atuarão na interface entre o sistema produtivo e o sistema gerencial de empresas. A formação multidisciplinar nas áreas de mecânica, eletrônica, instrumentação industrial, informática, controle e gestão da produção permitirá ao profissional elaborar estudos e projetos, bem como participar da direção e fiscalização de atividades relacionadas com o controle de processos e a automação de sistemas industriais. Nesse sentido, uma das metas na organização curricular do curso, é criar condições para que o aluno comece desde o primeiro ano a cumprir algumas horas de atividades de laboratório ligadas às aulas teóricas ministradas. Desse modo, os conceitos teóricos apresentados na sala de aula poderão ser aplicados concomitantemente no laboratório de uma maneira mais contextualizada. À medida que as disciplinas de cunho teórico progredirem e se tornarem mais complexas, ao longo do curso, as

atividades do laboratório deverão seguir o mesmo ritmo. Este é o objeto de investigação sob o qual a proposta de trabalho da tese foi desenvolvida buscando analisar ambientes em que os alunos foram incentivados a refletir sobre a possibilidade da certeza e a questionar a visão, muitas vezes exagerada, da infalibilidade da ciência (Poole, 1995). Nesse caso, buscando entender a ciência inerente ao processo de implementação de dispositivos mecatrônicos para se transformar esse processo em uma atividade produtora do conhecimento científico (Kuhn, 1989).

A disciplina Introdução à Engenharia de Controle e Automação – ES101 tem por objetivo ensinar aos alunos: Fundamentos Básicos da Engenharia de Controle e Automação; Atribuições Profissionais e Mercado de Trabalho; Conceitos de Integração de Sistemas; Introdução aos Sistemas Automatizados de Produção; Sistema Digitais e Analógicos; Circuitos Lógicos Básicos; Utilização do SFC – Sequential Flow Chart (Grafcet) para modelagem de Sistemas Seqüenciais e Introdução a Robótica. Em se tratando de Introdução à Engenharia, nesta disciplina tem sido desenvolvido atividades práticas envolvendo implementação de projetos de dispositivos mecatrônicos utilizando Kits de montar LEGO e também outros tipos de materiais educativos. Essas atividades práticas de implementação que envolvem concepção, estruturação e programação de dispositivos mecânicos/mecatrônicos têm sido desenvolvidas desde o segundo semestre de 1997 buscando criar um ambiente de ensino-aprendizagem que propicie ao professor e ao aluno condições para se trabalhar de forma concreta os conceitos teóricos apresentados na sala de aula.

As atividades práticas da disciplina Introdução à Engenharia de Controle e Automação são desenvolvidas, no primeiro semestre do curso, portanto, com os alunos iniciantes do curso de Mecatrônica. Isso se deve ao fato da proposta metodológica do curso pretender, desde o início, envolver os alunos em atividades que buscam sempre estabelecer uma ponte entre teoria e a prática, como forma de se criar um ambiente propício para se começar a familiarização com os diferentes conceitos das disciplinas que serão ministradas futuramente. Este ambiente propício é, na verdade, todo um contexto real que se deve criar para se desenvolver conteúdos, dos mais simples aos mais complexos, que não devem ser ensinados somente de forma teórica, mas também, com atividades práticas que pressupõem o desenvolvimento de funções intelectuais tais como: memória lógica, abstração, atenção deliberada, capacidade de comparar e diferenciar, capacidade de antecipar, dentre outras, (Vygostski, 1991). Tudo isso significando também

valorização do fazer como forma de aprender, criando situações problemas que sejam interessantes para os alunos contendo desafios capazes de serem resolvidos experimentalmente por meio de soluções fundamentadas no conhecimento científico (Carvalho & Lima, 1998).

Neste trabalho são apresentadas propostas de desenvolvimento de ambientes educacionais que podem ser incorporadas paulatinamente ao currículo do curso de Engenharia de Controle e Automação a fim de se alcançar um modelo de curso onde as disciplinas e os conteúdos sejam totalmente integrados para que exista um diálogo tanto entre os pares, colegas professores, quanto entre as disciplinas, os conteúdos destas, para que haja troca de idéias. Para Paulo Freire, o diálogo começa quando o educador pergunta em torno "de quê" ele vai dialogar com seus alunos (Freire, 1987). Ainda mais, o diálogo mantido no ambiente interdisciplinar deve ajudar a estabelecer a comunicação e a cooperação entre alunos e professores, tomando isso como uma prática social.

A tese apresenta e discute propostas para o desenvolvimento dessas atividades práticas como uma nova disciplina incluída no elenco das demais disciplinas que o aluno cursa, durante os dez semestres do curso de graduação, a fim de se obter o título de Engenheiro Mecatrônico. Esta nova disciplina deverá promover a integração das demais disciplinas do curso, em prol de uma formação menos fragmentada possível para os alunos. Portanto, por meio de atividades práticas no laboratório estariam sendo resgatados os conceitos teóricos trabalhados em classe por outras disciplinas, acompanhando o aluno do primeiro ao último ano da sua formação, propiciando condições para que surjam oportunidades de se realizar trabalhos científicos que possibilitam competência técnica, intercultural, pessoal, social, dentre outras.

Portanto a proposta de pesquisa da tese fundamenta-se nos trabalhos de implementação de dispositivos mecânicos/mecatrônicos para estudo de conceitos onde os sujeitos da pesquisa são alunos dos cursos de mecânica, mecatrônica e de computação da UNICAMP, e técnicos especializados em uma fábrica, (Harrison Delphi em Piracicaba - SP) que produz componentes automotivos. Essas atividades serão descritas no capítulo 4 tese.

I.2. Questão a Investigar

A grande questão que começou a ser investigada neste trabalho envolve a implementação de ambientes educacionais. Ao implementar o ambiente, que características especiais ele deve ter para auxiliar no processo de ensino-aprendizagem, portanto no *design* do ambiente, tanto no que diz respeito ao software quanto ao hardware? Que recursos são necessários para que o ambiente sirva para construção de conhecimento? Outra questão decorrente desta é a questão metodológica: uma vez implementado o ambiente, que tipo de metodologia de trabalho deve ser utilizada pelo grupo (professor e alunos) para que os projetos de dispositivos mecatrônicos a serem desenvolvidos contemplem os conceitos inerentes as diferentes disciplinas?

Na tentativa de solucionar essas questões, algumas universidades e centros de pesquisa no Brasil e, em outros países ao redor do mundo vêm se preocupando não somente com a implementação e introdução de novas tecnologias na educação, mas também com o desenvolvimento de metodologias para utilização dessas tecnologias.

I.3. Justificativa

A normatização dos cursos de graduação, quer seja na área das engenharias ou em outras áreas, por si sós, se vistas de forma descontextualizadas, são instrumentos burocráticos que mais se preocupam com o funcionamento destes cursos do que com a aprendizagem propriamente dita.

As ferramentas educacionais existentes, não servem para implementar situações de aprendizagem de nosso interesse, na medida em que, na sua grande maioria, elas são construídas para um contexto de uso de um ensino fragmentado onde não existe a integração/interação dos conteúdos, das disciplinas, dos professores e nem da direção como um todo.

No contexto do curso de Engenharia de Controle e Automação, construir ambientes educacionais diferenciados que propiciem uma situação de aprendizagem integradora é um desafio do qual deve participar os professores, os alunos, a coordenação de graduação e a direção

da Faculdade de Engenharia Mecânica da UNICAMP. A idéia de implementação de ambiente educacional implica em algo mais do que simplesmente aglutinar um grupo de pessoas interessadas em aprender e outra, interessada em ensinar. Significa criar uma comunidade educacional composta por pessoas com linguagens, afinidades e objetivos educacionais comuns (Garcia, 2002). No entanto, para se construir esse ambiente, é preciso implementar as ferramentas de hardware e de software para fins educacionais, permeadas por essa filosofia de trabalho, e desenvolver uma metodologia de trabalho que, além de possibilitar o uso destes recursos, permita também a integração das disciplinas, uma inter-relação entre os professores e os conteúdos a serem aprendidos durante o curso e, finalmente, uma concatenação entre professores alunos e os órgãos diretivos visando sempre promover a integração da formação acadêmica com a prática profissional (Vallim, 2000).

Isso objetiva, dentre outros aspectos acadêmicos, o envolvimento dos alunos em projetos de pesquisas que podem produzir desde projetos de iniciação científica até teses de doutorado. No caso do Laboratório de Ensino Pesquisa em Automação e Robótica – LAR pode-se citar um projeto de iniciação científica que nasceu com base nesse tipo de trabalho (Takiguthi, 2001).

Sob esse enfoque, o grande desafio consiste em criar ambientes de aprendizagem em que os alunos devem ser incentivados a pesquisar, a refletir e a questionar os resultados obtidos. Para se ter esse ambiente é preciso que as escolas de engenharia incorporem nos seus currículos, conteúdos que ajudem os alunos, no processo de aprendizagem a adquirir fluências em diversas áreas do conhecimento. Ou propondo problemas/desafios que os sejam interessantes e que possam ser resolvidos experimentalmente levando-os a obter soluções que tenham como base o conhecimento científico (Carvalho & Lima, 1998). Para Mendelsohn, as escolhas pedagógicas dos ambientes de ensino e aprendizagem (AEA) podem ser descritas sobre dois eixos ortogonais: o primeiro eixo representa a abertura do sistema relativamente às ações que o sujeito pode realizar no ambiente, onde um software aberto permite ao aluno uma grande combinação de comandos e funções no ambiente. Este autor aponta o Logo como exemplo de um software com esta característica. O segundo eixo representa o tipo de conhecimento sobre o qual o software visa fornecer uma ajuda eficaz do ponto de vista de aprendizagem (Mendelsohn, 1989). Nesta

pesquisa a implementação de ambientes educacionais tem com base as abordagens desses autores.

I.4. Objetivos Gerais

Apresentar e discutir propostas para se desenvolver ambientes de aprendizagem baseados na construção de dispositivos mecatrônicos automatizados e utilizá-los para estudo de conceitos. Num primeiro momento os professores das disciplinas consideradas profissionalizantes serão os primeiros a se envolverem neste processo e posteriormente os professores das disciplinas básicas. Isso porque o processo todo se constitui em uma grande mudança estrutural que dificilmente acontecerá em um ou dois semestres. Certamente serão necessários, pelo menos meia década de um trabalho continuado, para que as mudanças se façam sentir de forma mais efetiva e abrangente.

I.5. Objetivos Específicos

Construir sistemas mecânicos similares aos existentes nas indústrias, baseados em atuadores, sensores e demais elementos elétricos/eletrônicos. Identificar nesses sistemas princípios de automação e controle. Utilizar esses sistemas numa abordagem multi e interdisciplinar. Desenvolver metodologias que permitam trabalhar de forma concreta o aprendizado de conceitos nas diferentes disciplinas do curso de Automação e Controle.

I.6. Delimitação da Investigação

A atividade de pesquisa da qual este trabalho teve início encontra-se fortemente relacionada com a atividade profissional que o autor vem realizando, há aproximadamente quinze anos, no Núcleo de Informática Aplicada à Educação - NIED/UNICAMP, envolvendo escolas de ensino fundamental e médio, cursos de especialização em universidades, oficinas de trabalho em instituições formais e não formais de ensino e, mais recentemente, as atividades didáticas no

Laboratório de Automação e Robótica - LAR/FEM/UNICAMP e no Instituto de Computação – IC também da UNICAMP. Este estudo envolveu pesquisas em três locais distintos mas que se inter-relacionavam.

No Núcleo de Informática Aplicada à Educação – NIED, na investigação do processo de construção do conhecimento, mediado pelo uso do computador, passando pelo desenvolvimento de equipamentos e de software com finalidades educacionais; na implementação de ambientes para utilização desses equipamentos e softwares por alunos e professores e, na criação de uma metodologia de ensino-aprendizagem com base na implementação desses ambientes.

No LAR, na implementação de ambientes de ensino-aprendizagem com base na utilização de kits LEGO e outros kits educacionais, do tipo Robix, na montagem, automação e controle de dispositivos mecânicos/mecatrônicos envolvendo alunos de engenharia mecânica e alunos do primeiro ano do curso de controle e automação (mecatrônica) e um aluno de iniciação científica. A implementação de ambientes educacionais que retratavam aspectos reais existentes nas diversas atividades inerentes ao processo de automação, controle industrial, gestão de produção. Isso num período de 4 anos.

No Instituto de Computação - IC, na implementação de um ambiente de ensino-aprendizagem utilizando Kits LEGO com alunos do curso de Engenharia e Ciências da Computação durante o segundo semestre de 2001. As atividades desenvolvidas foram similares às desenvolvidas com os alunos de mecânica e mecatrônica, porém, com ênfase na percepção robótica, sistema sensorial e motor.

I.7. Estrutura do Trabalho

Ao longo dos capítulos serão discutidos aspectos teóricos e práticos envolvendo a implementação de ambientes de aprendizagem baseado no uso de kits LEGO bem como outros materiais. Os capítulos estão assim constituídos:

Capítulo 1 - discute a automação do ponto de vista histórico e também tecnológico no que consiste a sua utilização como sistemas mecânicos, eletrônicos à base de computadores. Será discutida a questão da educação e da tecnologia, sobre o papel importante que o trabalho artesanal exerceu em determinado momento da história da humanidade e sobre o descompasso do processo de ensino-aprendizagem quando comparadas com as exigências do mercado de trabalho. Por meio da utilização de ambientes de aprendizagem onde o computador associado a outras ferramentas permite implementar modelos que sejam o mais próximo possível do modelo real, para que durante o processo de formação o sujeito (técnicos na fábrica ou aluno na sala de aula) adquira competência pessoal, técnica, intercultural, social, dentre outras. Esta é uma das questões a ser apresentada neste capítulo.

Capítulo 2 - discorre sobre a informática na educação, utilização de ambientes de robótica pedagógica como forma de se fazer automação no contexto educacional. O capítulo também apresenta descrições de alguns tipos de ambientes educacionais baseados no uso de dispositivos mecatrônicos com propósitos educacionais e o Estado da Arte no tocante à implementação desses ambientes nos âmbitos nacional e internacional. Além disso, o capítulo discute um ambiente de telerobótica em alguns dos seus diferentes enfoques de utilização na rede Internet.

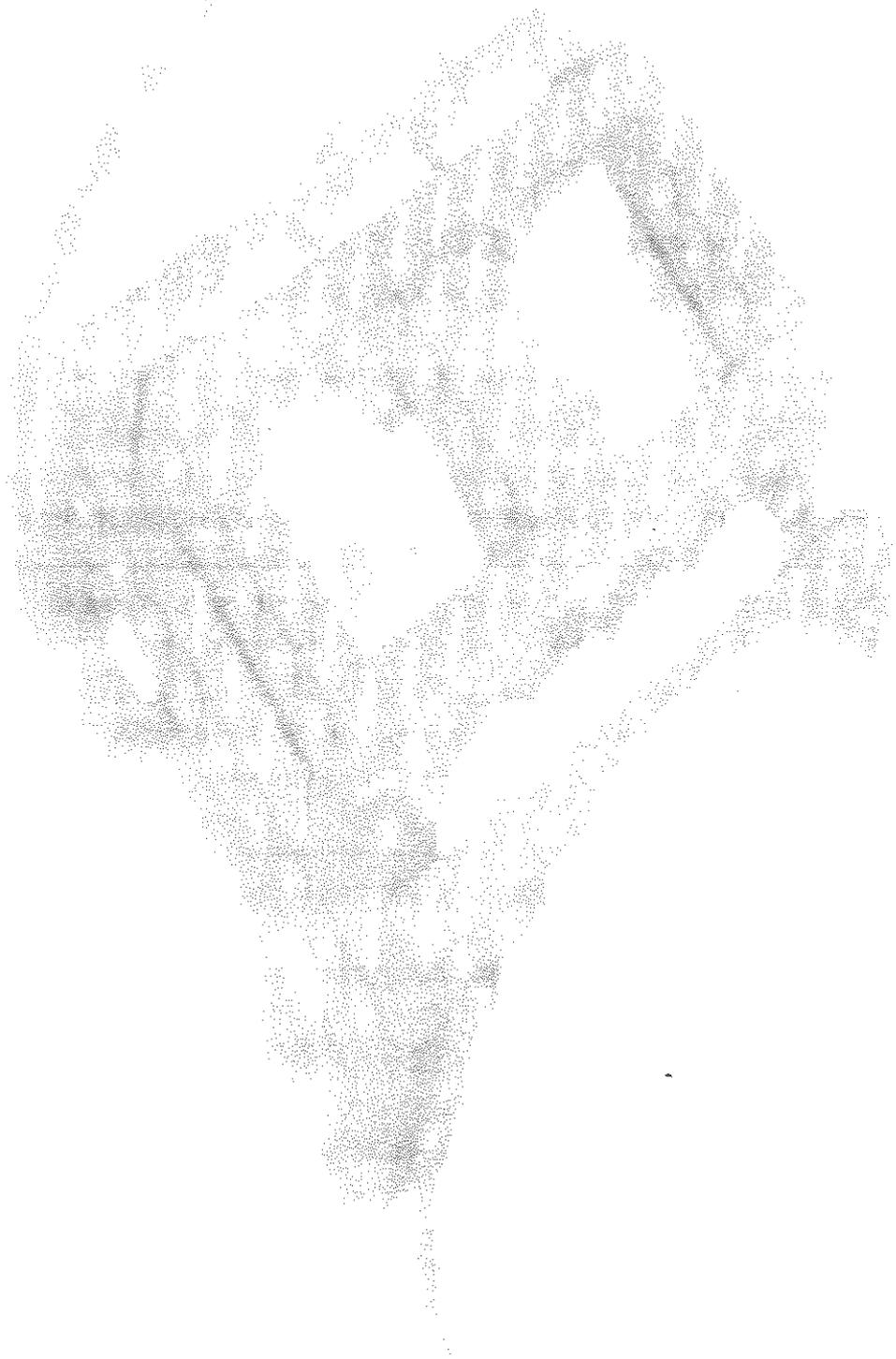
Capítulo 3 – discute sistemas automatizados discretos com ênfase na utilização de Grafset e sistemas contínuos com base em função de transferência de sistemas em malha aberta e malha fechada. O capítulo aborda também sensores e atuadores como os principais elementos operativos que fazem parte da implementação de um dispositivo robótico/mecatrônico. Além disso, o capítulo apresenta alguns sensores e atuadores de uso comercial e também os de uso educacional, destacando os comercializados pela LEGO.

Capítulo 4 - apresenta a parte experimental desta tese, descrevendo de três projetos práticos experimentais desenvolvidos junto aos alunos da Engenharia de Controle e Automação (Mecatrônica), Engenharia Mecânica e Técnicos Especializados numa fábrica. Os projetos objetivavam o aprendizado de conceitos de automação por meio de concepção, estruturação e programação de dispositivos mecatrônicos utilizando motores e sensores.

Capítulo 5 – apresenta duas abordagens que foram realizadas ao longo da pesquisa de campo desenvolvida junto aos alunos dos cursos de Engenharia Mecânica, Engenharia de Controle e Automação (Mecatrônica) e Engenharia e Ciências da Computação. Além disso, o capítulo apresenta um levantamento e análise de dados multicategóricos, discorrendo sobre a descrição desses dados demonstrando a riqueza do caráter interdisciplinar de um ambiente educacional onde se privilegia o desenvolvimento de projetos mecatrônicos como forma de se introduzir conceitos de automação e controle.

Capítulo 6 - trata da conclusão apontando para as perspectivas futuras e pesquisas que poderão vir a ser desenvolvidas nessa área.

A seção reservada para bibliografias apresenta todas as bibliografias referenciadas ao longo da tese. Na parte final estão apresentados os anexos relativos a listagens de programas elaborados durante o trabalho, e outras informações complementares consideradas pertinentes inclusive algumas páginas Internet visitadas.



Capítulo 1

Conceitos Básicos de Automação e Robótica

Este capítulo discute a automação do ponto de vista histórico e também tecnológico no que consiste a sua utilização como sistemas mecânicos e eletrônicos à base de computadores. Será discutida a questão da educação e da tecnologia, sobre o papel importante que o trabalho artesanal exerceu em determinado momento da história da humanidade e sobre o descompasso do processo de ensino-aprendizagem quando comparadas com as exigências do mercado de trabalho. Por meio da utilização de ambientes de aprendizagem onde o computador associado a outras ferramentas permite implementar modelos que sejam o mais próximo possível do modelo real, para que durante o processo de formação o sujeito (técnico na fábrica ou aluno na sala de aula) adquira competência pessoal, técnica, intercultural, social, dentre outras. Esta é uma das questões a ser apresentada neste capítulo.

1.1 Automação

A automação é um termo técnico que, a cada dia, de forma consciente ou não, é utilizado por inúmeras pessoas e que, vem fazendo parte da vida de todos nós em todas as instituições e organizações em que participamos, inclusive a doméstica. Do ponto de vista da indústria, automação pode ser definida como o processo que abrange as máquinas e os procedimentos de execução das operações e a informatização entendida, como a gestão das informações necessárias à execução das operações. A automação vem alterando os meios de trabalho introduzindo novas formas e conceitos de organização da produção (Coriat, 1988). O grande desdobramento disso tem sido as mudanças de ordem social, econômica, educacional, política, comportamental, dentre

outras que vêm ocorrendo em virtude do nível de automação incorporados aos equipamentos, quer seja os de uso individual quer seja os de uso coletivo.

A automação também pode ser definida como sendo o controle eletrônico ou mecânico que substitui a função de controle do cérebro humano (O'Brien, 1964), ou ainda, uma tecnologia que se ocupa do uso de sistemas mecânicos, eletrônicos e à base de computadores na operação e controle da produção (Groover at alli, 1988). Embora essas definições evoquem o uso do computador e da eletrônica que são tecnologias e inventos relativamente modernos, a automação por si não é uma inovação recente. No final do século XVIII, por exemplo, foi introduzido na França os teares automatizados que eram controlados por cartões perfurados semelhantes aos que eram usados nas antigas máquinas IBM até a década de 70 no século XX (Ullrich, 1987). Três décadas, após a invenção dos teares, por volta de 1822, Charles Babbage construiu a máquina de calcular, um precursor do computador. Podemos afirmar que a automação existe, pelo menos, desde a Revolução Industrial em formas e graus variados (Taulie, 1988). Todavia, foi no início do século XX que a mesma foi incorporada pelas fábricas.

Do ponto de vista histórico, a automação nos remete à técnica, como sendo “todo método racional em vista de uma finalidade” (Nélis, 1965); procedimentos e métodos de criação de bens materiais de consumo, de equipamentos ou de comunicação, cuja livre disposição determina a evolução da existência do homem (Daumas, 1983). Pois, é no aperfeiçoamento da técnica para a realização de um trabalho que o homem vem caminhando em direção a automação ao longo de séculos. Da técnica do homem primitivo que afiava um machado de pedra utilizando pedras mais duras iniciou-se o processo de transformação da matéria. Desse processo cronologicamente se distingue:

- **Utilidades**, produtos que não servem senão ao homem.
- **Instrumentos**, produtos que se prestam a um trabalho ulterior (posterior) sobre a matéria.
- **Utensílios**, são complexos da matéria movidos pelo homem e agindo sob seu controle (ex: martelo).

- **Máquinas-Ferramentas**, complexos de utensílios movidos pelo homem (ex: máquina de escrever).
- **Máquinas**, complexos de utensílios movidos por uma força diferente do homem, mas, agindo sob seu controle (ex: britadeira).
- **Máquinas-Autômatos**, complexos de máquinas que agem sem o controle do homem, mas, com controle pré-determinado (ex: rádio).
- **Autômatos**, máquinas que agem de modo condicionado. Estrutura movida por uma força diferente do homem e age sem controle humano (ex: radar).
- **Automação**, sistema automático de controle pelo qual os mecanismos verificam o seu próprio funcionamento efetuando medições e introduzindo correções sem a interferência humana.

Da última definição pode se acrescentar uma outra, mais abrangente, que é a de automação como sendo qualquer dispositivo que reduz o esforço humano, físico e/ou mental, necessário para produzir trabalho.

Com base no enfoque cronológico, pode se concluir que a técnica vem produzindo progresso ao longo dos séculos, progressos esses que trouxeram e têm trazido, cada vez mais, avanços à sociedade, embora nem toda a humanidade participe deles.

1.2 Automação Alterando os Meios de Trabalho

O mundo empresarial dos anos 80 outorgou, ao mercado a responsabilidade de destinar eficazmente os recursos e determinar os preços, as indústrias a competição em mercados de outros países, a população, maior voracidade e exigência no consumo (Santomé, 1998). Isso tem

levado à uma nova concepção da organização do trabalho, que segundo o Taichi Ohno, (engenheiro-chefe da empresa Toyota) distingue-se pelos seguintes fatores:

Eliminação de recursos redundantes "produção enxuta"³;

Fabricação ou produção de algo com maior rapidez possível "*just-in-time*"⁴;

Defeito zero "Qualidade Total".

Formação Contínua "Qualificação em Serviço"

Estes fatores associados ao desenvolvimento de novas tecnologias têm propiciado ao mundo empresarial e instituições realizar processos, de forma descentralizada e, aumentando a flexibilidade dos modelos organizativos, num contexto onde o processamento e o gerenciamento da informação é o núcleo básico sobre o qual articula-se uma nova capacidade da humanidade de controlar o que está a sua volta (Castells at all, 1986).

A flexibilidade e a polivalência da classe trabalhadora tem permitido que um mesmo operário seja responsável pelo manejo e controle de várias máquinas, ao contrário do modelo fordista, onde cada trabalhador atendia apenas a uma tarefa específica. Além disso, a flexibilidade pode permitir um aumento de salários por intermédio do aumento da produtividade e formação contínua de técnicos.

A inovação tecnológica pela qual as empresas têm passado pode ser caracterizada pelos seguintes fatores: a informatização dos meios de produção, automação flexível, novas formas de gestão da empresa e novas formas de gestão da produção e organização do trabalho. Nenhum destes fatores isolados é responsável pela inovação tecnológica, mas sim o conjunto de todos eles utilizados sob a óptica de uma mudança de paradigmas que busca a integração e a flexibilidade

³ A idéia principal da produção enxuta está centrada em produzir sem desperdício de trabalho, energia, tempo, dinheiro e materiais. Ou seja, produzir reduzindo os custos ao mínimo necessário (Mazzone, 1993)

⁴ A idéia de Just-in-Time significa produzir somente o que é necessário, quando necessário na quantidade necessária. Ou seja, reduzindo o estoque ao mínimo (Klaus, 2000).

da linha de produção (Coriat, 1988). A partir dos anos 60/70, com o surgimento da microeletrônica novos meios de trabalho passaram a ser programáveis, as inovações tecnológicas passaram a impor às empresas novas necessidades de integração para que elas pudessem dar saltos qualitativos de produtividade e vencer, tanto o acirramento da concorrência, quanto os entraves sociais inerentes às formas tradicionais de produção e organização do trabalho. Neste contexto, surge a necessidade de mudança de paradigma de uma empresa tradicional "taylorista-fordista" para uma empresa flexível (Salerno, 1992).

Uma estrutura flexível de manufatura tende a:

- Trabalhar com estoque baixo de produto em processo.
- Ter um giro elevado da produção.
- Trabalhar com produtos que apresentam baixo ciclo de vida.

Os fatores acima citados aumentam a variabilidade de produção e diminuem o tempo entre a encomenda e a entrega dos produtos. Neste sentido, a função de técnicos passa a ser de gerir a variabilidade visando manter o fluxo de produção dentro das especificações de tempo e qualidade de conformação. Os projetos dos processos de produção, dos meios de trabalho e de formas automatizados de controle definem modelos de como a produção deve se comportar.

Uma estrutura flexível de manufatura pode ser descrita como sendo um sistema de controle que pressupõe coleta de dados de pontos de processo que podem ser lidos por um conjunto de sensores (termômetros, medidores de pressão, de fluxo, de temperatura, etc.). Esses dados são enviados para computadores, que utilizando um determinado software de controle de processo, analisa se os mesmos são coerentes com os valores previstos teoricamente. Havendo discrepância, os computadores devem emitir ordens para tentar adequar o processo real ao previsto teoricamente.

A descrição acima idealiza as condições concretas da produção automatizada onde todas as variáveis são controladas automaticamente e as devidas correções, efetuadas pelos computadores. Porém, na prática, nem sempre é isso que ocorre, pois podem existir falhas e o modelo, às vezes, não funciona como previsto. A matéria prima apresenta variações, os sensores não funcionam ou dão informações incorretas, os atuadores falham, os computadores sofrem interferências de raios etc. etc.. Todas estas situações são imprevistos os quais o operário de uma estrutura flexível tem que administrar. E, imprevistos quando acontecem, significam, para os técnicos mais qualificados, lidar como uma série de variáveis desconhecidas até então. Ou para os menos qualificados, seguir fielmente alguma receita, o que normalmente acaba não funcionando. Portanto, para que haja uma boa produtividade, é preciso que o operário de processos integrados e flexíveis tenha como função básica a gestão da variabilidade. Ou seja, saiba lidar com o imprevisto.

É fundamental que o operário, seja capaz de abstrair as etapas do processo e tenha capacidade de elaborar uma representação do que está acontecendo. Esta representação seria um modelo mental do estado do processo e de suas possibilidades. Ou seja, ações operatórias que o operário teria que fazer (Salerno, 1992). A elaboração de uma representação depende de uma formação teórica acerca de um dado processo, da prática de controle deste mesmo processo, do tipo da organização do trabalho, de um bom conhecimento espacial das instalações, etc. Isso significa dizer que o trabalho humano de elaboração é fundamental em qualquer tipo de produção por mais automatizada e informatizada que ela seja. É preciso que, diante de situações problema, o sujeito tenha um modelo mental estruturado do problema a ser resolvido para que ele seja capaz de criar representações mentais que o ajudem a solucioná-lo. Ser capaz de buscar diferentes estratégias de solução. No capítulo cinco este assunto será discutido tendo como base o trabalho de campo desenvolvido junto aos alunos de Ciência da Computação.

Com maior ou menor intensidade as mudanças tecnológicas hoje se fazem presentes em praticamente todas as sociedades. Conforme já foi citado neste capítulo, elas se devem fundamentalmente aos avanços na área da microeletrônica e seus desdobramentos em termos de informatização, automação, e robotização do sistema produtivo; mudanças genéticas na área da microbiologia; biotecnologia; uso de novas fontes de energia; etc.. Estas mudanças têm sido

caracterizadas por alguns autores, como a terceira revolução industrial. Revolução esta que vem causando profundos impactos nos seguintes campos da sociedade:

- Econômico, por intermédio das relações técnicas e sociais da produção;
- Político, por intermédio das relações de poder;
- Cultural, por intermédio das relações de valores e éticos.

Por sua vez, diferentemente da primeira e da segunda revolução industrial, na visão taylorista da organização da produção, e fordista da produção em massa, cujas características se galgam na potencialização do trabalho manual, a terceira revolução⁵ e os seus impactos se configuram como sendo uma nova base científica para o processo de trabalho, potencializando as capacidades intelectuais do ser humano (Frigotto, 1992). Capacidades estas que se expressam em termos da qualificação que se dá por dois mecanismos: uma flexibilização por agregação de novas funções ao trabalhador e pela demanda de um novo perfil, com elevada capacidade de abstração.

Entretanto, é preciso entender dois aspectos importantes na introdução de novas tecnologias em um dado setor produtivo. Primeiro, que ela não representa a superação das formas anteriores de relações sociais de produção e de trabalho. Segundo, ela e seus impactos sobre as relações de produção devem ser entendidas, como um processo de trabalho, como uma relação de força. Acreditamos que, a despeito da sociedade globalizada em que vivemos onde, em função da rapidez e eficiência dos meios de comunicação, não existe praticamente distâncias geográficas, quando se trata do tráfego de informação, a tecnologia ainda deve ser tratada como um processo dependente, um produto das relações sociais. Por isso, as suas aplicações devem ser variadas dependendo dos regimes políticos dos países, das tradições culturais, da capacidade de organização e luta dos trabalhadores, promovendo assim impacto social diferenciado.

⁵ Esta terceira revolução poder ser considerada uma revolução dos três I's Idéias, Inteligência e Informação. Quem possuir os três I's tem grande chances de sucesso (Handy, & Kanter, 2000).

Ao fazermos uma análise sobre a mudança tecnológica no contexto da fábrica, constatamos que ela é um processo que vem ocorrendo a aproximadamente 40 anos. Esta mudança tem se expressado em termos de toda a revolução tecnológica pela qual as indústrias têm passado ao longo destes anos, revolução esta que tem marcado/definido o rumo econômico das corporações e determinando o comportamento destas. Ou seja, nas economias de mercado a grande meta dos dirigentes das empresas consiste em produzir benefícios que, em última análise, se transforme na condição fundamental para a sobrevivência da empresa (Ozaki, 1992). Os dirigentes investem em novas tecnologias para alcançar objetivos tais como:

- Diminuição da contribuição de mão-de-obra a fim de reduzir os custos de produção. Maior eficácia nas operações por meio de um controle mais rígido do processo de produção;
- Maior qualidade dos produtos ou serviços obtendo uma maior precisão das operações e rapidez no processamento da informação via computador;
- Maior capacidade de produzir produtos adaptados aos clientes em pequenos lotes.

Entretanto, do ponto de vista social a resposta dos trabalhadores a estes propósitos, ao longo dos anos, tem sido um tanto relutante. Quando da introdução da tecnologia de microeletrônica nas indústrias as políticas sindicais tinham um caráter fundamentalmente defensivo. Os sindicatos protegiam os seus trabalhadores contra os efeitos nocivos da nova tecnologia, em particular contra as possibilidades da perda do emprego (Ozaki, 1992). Contudo, a partir dos anos 70, a classe trabalhadora tem, de uma forma ou de outra, se esforçado para conviver com as mudanças tecnológicas, pois elas vieram para ficar e estão aí. Percebido isso, a classe trabalhadora teve que introduzir mudanças na política sindical para se adaptar a esta realidade. Ao nosso ver, este esforço precisa ser intensificado, sobretudo, no que diz respeito a formação educacional. A classe trabalhadora tem que lutar para que, utilizando-se de recursos como computador, *Kits* de montar do tipo LEGO e outros materiais, sejam implantados, em locais de trabalho, ambientes de ensino-aprendizagem onde, de forma contextualizada, as pessoas possam, não só aprimorar seus conhecimentos sobre a atividade que realizam, como também

adquirir novos conhecimentos ou mesmo suprir algumas falhas advindas da sua formação anterior.

Nesse contexto, na medida em que a competência passa a ser cada vez mais exigida, um número cada vez maior de trabalhadores e sindicatos começa a perceber que a inovação tecnológica é inevitável. Portanto, vale a pena tirar proveito dela, para poder melhorar a competitividade da empresa, tanto da pequena quanto da grande e, conseqüentemente, propiciar a garantia e condições do emprego (Tertre, 1989). Cabe aos sindicatos, participar cada vez mais do processo de inovação tecnológica, para possibilitar que a nova tecnologia seja introduzida em benefício da classe trabalhadora reduzindo ao mínimo seus efeitos desfavoráveis, pois, é bom lembrar que ao mesmo tempo em que ela tira um tipo de emprego outro é criado no seu lugar.

1.3 Automação e Educação da Classe Trabalhadora

Uma forma de se permitir que a nova tecnologia seja introduzida em benefício da classe trabalhadora é criar condições para que o nível educacional dos trabalhadores melhore cada vez mais. Acredita-se ser esta a única forma de se conseguir uma coexistência pacífica entre a classe empresarial e os trabalhadores. Todavia, isso parece ser um processo que ao longo dos anos, de forma gradativa, tende a se aprimorar, desde que as partes envolvidas, o Estado, os empresários e a classe trabalhadora, cumpram, cada uma, o seu papel.

O Estado deve ter como papel a definição e a gestão de uma política educacional que propicie o acesso a todos os cidadãos dos benefícios decorrentes das inovações tecnológicas nas diversas áreas econômicas, sociais e culturais. Ou seja, criar condições para que haja, pelo menos, uma educação básica de qualidade para todos, sobretudo nas escolas públicas. Para as classes populares, o acesso à escola básica deve ser a condição necessária, ainda que não suficiente, para uma qualificação humana que as capacite a lutar pelos seus direitos fundamentais. Essa qualificação básica deve ser de qualidade tal que propicie a oportunidade de uma formação profissional mais específica de nível mais elevado. Esse deve ser o papel do Estado, pois sem a formação básica, a formação profissional simplesmente se transformará em puro adestramento (Frigotto, 1992).

Nesse contexto, cabe aos empresários ao investirem nas novas tecnologias, não praticarem uma política de exclusão, ou de substituição do homem pela máquina, e sim de requalificação e reaproveitamento de mão-de-obra. Tarefa difícil, porém necessária para se amenizar o quadro social desfavorável que a introdução de novas tecnologias acarreta.

Ao trabalhador, cabe o papel de estar se aprimorando, buscando sempre adquirir mais conhecimento para evitar a possibilidade de perda de emprego. Problema que só uma boa formação escolar pode ajudá-lo a resolver. Certamente, o trabalhador que detiver um bom nível escolar facilmente se adaptará às inovações tecnológicas que o mercado exigir. Mesmo porquê, a qualificação para o trabalho diante das inovações tecnológicas tem um caráter dinâmico que faz com que a competência especializada para um dado conjunto tecnológico pode tornar-se obsoleta e inadequada para outro aparato tecnológico. Entretanto, existe algo confortante, que é o fato de que à medida que avançamos na sociedade de conhecimento, adaptamos as nossas vidas à realidade tecnológica que se fizer presente e que nos cercam.

Ao surgir o Sistema Flexível de Manufatura – FMS, por exemplo, admitia-se a hipótese de que ele seria capaz de propiciar uma dispensa total da intervenção humana. Entretanto, com o tempo, com a evolução da tecnologia, o que se tem verificado é que ele possibilita na verdade o deslocamento do homem das operações rotineiras para as inteligentes, deixando para os equipamentos computadorizados a realização das tarefas repetitivas com tomadas de decisão simples (Lepikson, 1995). Isso deixa claro a necessidade de se exigir trabalhadores com maior e melhor qualificação para tomada de decisões mais complexas. Isto é, decisões flexíveis e ágeis, que são inerentes aos problemas de qualidade numa fábrica.

A intervenção humana na tomada de decisões mais complexas está associada a capacidade de julgamento, destreza para execução de determinadas tarefas, experiência para reagir frente a situações imprevistas e capacidade de reconhecimento de características. Portanto, um trabalhador com esse perfil tem que necessariamente ter um bom nível intelectual e estar bem capacitado para adquirir bom conhecimento do processo, do relacionamento com o sistema produtivo, dos equipamentos da supervisão e do software de controle do processo.

Nesse sentido, acredita-se ser a educação a única alternativa para se alcançar este objetivo. Portanto, o que se deve esperar da escola é que ela seja um excelente ambiente de aprendizagem capaz de propiciar condições aos indivíduos, para que cheguem ao mercado de trabalho aptos para adquirir o perfil que vem sendo descrito ao longo do texto.

1.4 Qualificação

Com o uso da tecnologia microeletrônica, os equipamentos tornaram-se flexíveis, com comandos que são programáveis. Isso tem permitido que eles sejam utilizados para atender diferentes finalidades, assim propiciando um atendimento mais diversificado, de um mercado que vem crescendo a cada dia. Os trabalhadores que utilizam estes equipamentos precisam de maior qualificação, na medida em que a automação flexível necessita de processos de trabalho flexíveis e de flexibilização das funções (Machado, 1992).

Um processo de trabalho rígido é um processo onde ocorre intensa divisão e fragmentação do trabalho, com um grande controle da supervisão, para que exista um adequado funcionamento da linha de produção. Este tipo de processo pode ser considerado de desqualificação do trabalhador, na medida em que as tarefas são fixas, específicas, repetitivas, e monótonas.

Contrariamente, um processo de trabalho flexível, é um processo que tende a reduzir a possibilidade dos níveis de fragmentação e divisão do trabalho. Este tipo de processo de trabalho pode ser considerado de qualificação na medida em que ele oportuniza a intercambiabilidade de funções e a polivalência do trabalhador, conforme este é alocado em diferentes tarefas.

Segundo Machado (1992), a flexibilização que oportuniza a intercambiabilidade de funções e a polivalência do trabalhador é definida como a flexibilização funcional. Esta pode ser dividida em dois tipos: primeiro, que agrega funções para cada trabalhador, exemplo das ilhas de fabricação onde um único homem controla um conjunto articulado de máquinas. Segundo, a rotação por diferentes tarefas, com trabalho por equipes, que se responsabilizam pela seqüência inteira de uma dada etapa produtiva, arcando com todas as funções coletivamente.

A flexibilização funcional vem exigindo um novo perfil de qualificação da força de trabalho, fazendo com que os trabalhadores sejam detentores de: uma escolaridade básica, capacidade de adaptação a novas situações, compreensão global das tarefas e capacidade de abstração para trato e interpretação de informações. Neste contexto, acreditamos que, tem sido proposto uma abertura para novas situações de aprendizagem, com um certo estímulo à criatividade para enfrentamento de imprevistos. Isso vem ao encontro aos grandes desafios que vem sendo colocados atualmente aos trabalhadores como: ter visão crítica do problema, ter capacidade de tomar decisões, o que demanda finalmente, um constante aprender a aprender.

Diferentes autores discutem a qualificação usando abordagens que a definem tanto como uma questão social onde ela pode ser vista como uma maneira de se mobilizar, reproduzir, adicionar as diversas formas de trabalho, quanto como uma questão técnica da forma de gestão e organização do trabalho, que pressupõe conhecimento, experiência, autoridade e possibilidades materiais. O importante a ressaltar é que qualquer uma das abordagens apresenta como pano de fundo os efeitos que a introdução de novas tecnologias causa no processo de qualificação ou desqualificação dos trabalhadores.

Neste contexto, para Castro (1992), os trabalhadores podem ser considerados qualificados em função da existência de regras deliberadas de restrição à ocupação, coletivamente produzidas, partilhadas e barganhadas, e não necessariamente em decorrência da natureza da ocupação que desempenham. O que concebe a qualificação como sendo algo socialmente construído, um "status social". Neste caso, a qualificação é dependente de fatores como: o costume e a tradição socialmente construída, ou a organização coletiva que protege corporações ou coletivos profissionais (Castro, 1992).

A nova concepção dos métodos modernos de produção tem exigido dos trabalhadores o uso de múltiplas habilidades, a utilização do conhecimento tático, criatividade, autonomia e prática de autocontrole. Isso faz com que a qualificação dos trabalhadores não possa ser pensada isoladamente e sim, levando em consideração a organização do processo de trabalho (Leite, 1994).

A organização do processo de trabalho aí tem um papel fundamental. Na medida em que quando ela não acontece, a introdução de novas tecnologias pode levar à desqualificação do trabalhador ao invés de qualificá-lo. Isso ocorre porque, uma vez introduzida a nova tecnologia sem se efetuar a organização do trabalho, passa-se a exigir uma qualificação a mais para realizar uma mesma atividade. Ou seja, faz-se necessário a utilização de conhecimentos anteriores e de novos conhecimentos, inerentes à nova tecnologia introduzida, para se realizar uma velha tarefa. Este modo de produção, com certeza, não aumenta a produtividade.

Finalmente se, por um lado, o processo de adoção de novas tecnologias e das novas formas de trabalho tende à elevação da qualificação média da força de trabalho, por outro lado, ela é um processo seletivo discriminatório que cria um pequeno grupo com perfil qualificado e um grande grupo com perfil desqualificado e descartável. Além disso, em termos globais, existe a perspectiva de um grande crescimento de desemprego e adoção de formas contratuais precárias, temporárias e fragmentadas.

1.5 Polivalência e Politecnia

A nova tecnologia tem sido responsável pela alteração do modo de organização do trabalho. Associado a este fato, vem a questão do uso de equipamentos a base de microeletrônica/informatizados que normalmente tendem a propiciar em equipe, o que fomenta a polivalência (Ozaki, 1992).

A polivalência significa um trabalho mais variado com aberturas para a possibilidade de administração do tempo pelo trabalhador, não importando necessariamente na mudança qualitativa das tarefas. Para ser um trabalhador polivalente, basta possuir os conhecimentos empíricos existentes. Contudo a politecnia representa o domínio da técnica no âmbito intelectual e a possibilidade de um trabalho flexível com reposição de tarefas no âmbito crítico. O conhecimento deixa de ser empírico e passa a se exigir do trabalhador recursos e formas de pensamentos mais abstratos.

Um trabalhador polivalente se apóia nas regras e processos de trabalho definidos nos parágrafos anteriores. Além disso, dele se exige: abertura, adaptação às mudanças, possibilidades de lidar com regras e normas em situações diferenciadas, curiosidade, vontade de aprender, motivação, iniciativa, atenção, responsabilidade, dentre outros atributos imprescindíveis.

Já para um trabalhador politécnico, é necessário: discernimento e julgamento crítico, compreensão dos determinantes sociais, econômicos e políticos das ações a serem empreendidas, independência na avaliação das implicações das intervenções humanas frente a outras alternativas e finalidades e criatividade no enfrentamento das contradições. Além disso, é preciso a compreensão teórica e prática das bases das ciências contemporâneas, princípios tecnológicos que expressam o uso da ciência no emprego de materiais, métodos e meios de trabalho e princípios da organização do trabalho e de gestão social nas diversas esferas da vida humana. Para um trabalho politécnico recomenda-se: saber transferir e usar de forma versátil conhecimentos e experiências em diferentes oportunidades e situações, saber manipular instrumentos básicos úteis a um leque amplo de tarefas, saber trabalhar em equipe. Ou seja, saber criar a rede de conhecimentos, onde seja possível unir a destreza do fazer com a inteligência e o pensar num nível superior.

A construção de um saber polivalente depende de educação básica, mas sem que seja necessária uma grande revolução na escola. O trabalhador polivalente se faz no trabalho com algumas inovações na organização do processo de trabalho, com atualização do treinamento em serviço.

A construção de um saber politécnico pressupõe uma completa reestruturação do ensino básico e da formação básica nos cursos tanto profissionalizantes quanto de nível superior. Sem isso, não será viável a possibilidade da autonomia necessária à educação continuada e qualificação constante. O saber politécnico se insere num contexto de um projeto de desenvolvimento social de ampliação dos processos de socialização, sem se restringir aos imediatismos do mercado de trabalho (Machado, 1992).

Numa escala crescente de valor, podemos afirmar que o trabalhador formado segundo o paradigma fordista-taylorista encontra-se em um pólo. O trabalhador polivalente encontra-se em uma situação intermediária, e o trabalhador politécnico encontra-se em um pólo totalmente oposto.

Finalmente, a medida que as inovações tecnológicas vem sendo impostas à sociedade, desde que haja abertura e, condições propícias para a mudança de paradigma, no sistema de produção fordista-taylorista para um sistema polivalente e deste para politécnico, elas se tornam imprescindível para uma indústria moderna.

Ainda do ponto de vista social, no que diz respeito ao trabalho, podemos afirmar que dois tipos de profissionais acabam se definindo. O primeiro, é o profissional tradicional, fruto do processo de industrialização caracterizado por: baixo salário, baixo nível educacional, instabilidade no emprego e desempenho de funções desqualificadas. Estes são os profissionais que pertencem ao setor social que vem sofrendo os vários processos de desenraizamento, quer pela alternância do trabalho ou pelas instabilidades sociais. O segundo, é o novo profissional que desfruta de um nível salarial mais elevado, que tem um nível educacional mais alto e goza de relativa estabilidade no emprego, o que se constitui em uma certa aristocracia técnica (Machado, 1992).

Entretanto, mesmo para este segundo tipo, tem ocorrido outras formas de desenraizamento, decorrentes da aplicação de novas tecnologias desvinculadas de uma política sócio-econômica e cultural de caráter mais global para a sociedade. Pois, caso não haja uma visão mais ampla da realidade em suas várias dimensões, o acesso a informações técnicas, por si só, não garante uma efetiva compreensão do trabalho que se executa nem do mundo a qual a atividade humana está inserida. Todavia, a automação flexível possui elementos novos capazes de permitirem o controle do saber e da produção pelos trabalhadores, na medida em que o uso da nova tecnologia implica no interesse, na motivação, responsabilidade, atenção, capacitação, participação, enfim do conhecimento que esse trabalhador detém. A falha em algum destes elementos pode inviabilizar o funcionamento do restante (Machado, 1992).

1.6 Aspectos Econômicos em Estruturas Flexíveis

As inovações tecnológicas que têm sido processadas nas economias industrializadas vêm respondendo a dois objetivos fundamentais: tornar as empresas mais aptas a disputarem o mercado e aumentar o controle sobre a produção e os trabalhadores (Leite, 1994).

Este processo de inovação tecnológica, no que diz respeito a competitividade, tem propiciado possibilidades de diminuição dos custos de produção, elevação de produtividade e melhoria de qualidade. Esta melhoria dos equipamentos automatizados tem possibilitado uma adequação mais rápida da produção às exigências do mercado, na medida em que a flexibilidade das máquinas programáveis garantem melhor produção.

Por outro lado, no que diz respeito ao controle as máquinas programáveis garantem uma certa independência das empresas, em relação aos trabalhadores. Na medida em que, as máquinas passam a impor um ritmo de trabalho mais regular, isso assegura que a velocidade e a qualidade do trabalho passam a ser garantidas pela máquina e não mais pela habilidade e nem pela vontade humana.

Em um Sistema Flexível de Manufatura (SFM), definido como sendo um sistema de produção automatizada totalmente controlada por computador, pode-se oferecer vantagens substanciais quando comparada com o sistema de produção convencional (Kalkunte, Sarin & Wilhelm, 1986). Neste sistema, a única preocupação com máquina é se ela está parada, sendo essa a única situação em que a mesma não está produzindo. A partir do momento em que a mesma entra em funcionamento, a princípio, a produção está garantida sem perdas dentro da programação que foi especificada. A máquina programável é automática e permite à empresa uma disciplina maior do processo de fabricação, eliminando a figura humana "capataz" encarregado pelo controle do fluxo produtivo. Isso permite um clima melhor de trabalho porque com a figura de encarregado, da época convencional, era sempre difícil o relacionamento por parte dos trabalhadores.

Nesse contexto, em vista do que existe hoje do SFM, e da organização do trabalho, advinda desta, podemos afirmar, com bastante segurança, que o processo de produção antes da automação era um processo artesanal dependente do humor do trabalhador. Ou seja, o conhecimento para

execução de determinada tarefa estava com o trabalhador. Atualmente isso não ocorre. O fato deste tipo de conhecimento estar agora na máquina e fazer parte do patrimônio da empresa, faz com que o desempenho da empresa seja previsível, com garantia de uma determinada qualidade do produto num espaço de tempo bem definido e, conseqüentemente, com uma previsão de maior lucratividade.

Nesse sentido, ao assegurar que a produção tenha uma certa qualidade e seja feito em um determinado ritmo que garanta a produtividade, novas tecnologias permitem em parte que a administração das empresas aumente o controle sobre o processo produtivo. Além disso, como facilitam a adoção de rígidos planejamentos de produção a ser seguidos pelos técnicos, novas tecnologias podem criar condições para a gerência aumentar o controle sobre a mão-de-obra. O que não parece ser muito agradável para o operário.

Uma questão que tem sido colocada como uma exigência, cada vez maior, do próprio mercado consumidor, é o rápido atendimento no que se refere à diminuição dos prazos de entrega, como forma de ser mais competitiva, vem exigindo das empresas uma reorganização da produção, objetivando agilizar os tempos, efetivando assim a entrega dos produtos o mais rápido possível.

O fato interessante a se colocar aqui é que a automação flexível permite que alguns setores da empresa, sobre tudo o da manutenção, tenha alguma ociosidade. Inteligentemente algumas empresas têm integrado os trabalhadores de manutenção na produção, como forma de, por um lado, viabilizar mais rapidamente o atendimento ao mercado. E, por outro lado, eliminar o departamento de manutenção, na medida em que as máquinas automáticas exigirão, cada vez mais, menos manutenção. Atualmente, isso tem sido feito em algumas empresas, com o objetivo de que o manutentor se transforme em operador e este entenda de manutenção. Entretanto, a visão das empresas parece ser a de reorganização do processo produtivo dentro do princípio de *just-in-time*, diminuindo o pessoal não ligado diretamente à produção, por meio da integração das atividades de manutenção e controle de qualidade à produção. Ou seja, o operador está deixando de ser só operador e, fazendo um pouco de manutenção, enquanto as máquinas ainda exigirem manutenção. No futuro ele certamente será capaz de fazer tudo.

1.7 Tecnologia e Organização do Trabalho

Segundo Ozaki, a introdução da tecnologia de microeletrônica consiste em colocar no local de trabalho equipamentos informatizados. Isso exige habitualmente uma certa reorganização do trabalho. Os procedimentos para esta reorganização são bastante amplos. Alguns aspectos destes procedimentos são:

- A composição e qualificação de pessoal para serviço de produção;
- A amplitude com a qual o funcionamento direto das máquinas está separado do planejamento do trabalho;
- Grau de fragmentação das funções trabalhistas e o grau de rigidez da divisão do trabalho;
- A amplitude com a qual o processo de trabalho está controlado pela diretoria.

A relação que existe entre a tecnologia de microeletrônica e organização do trabalho não é direta. Mesmo porquê, quando as empresas introduzem nova tecnologia, nesta área, elas podem se organizar de diferentes maneiras. Vários aspectos influenciam a organização do trabalho e a nova tecnologia. Dentre eles podemos destacar: a influência dos trabalhadores e dos sindicatos; a influência dos sistemas das relações de trabalho e os tipos de organização do trabalho existentes no momento da introdução da mudança tecnológica.

Alguns fatores que intervêm na relação entre tecnologia e a organização do trabalho, como por exemplo, as dimensões do produto e o tamanho da fábrica, não têm uma relação direta com os trabalhos. Nas indústrias que fabricam produtos em grandes lotes a organização do trabalho tende a ser mais segmentada, com mais rigidez, e a distribuição das qualificações estão mais polarizadas, do que nas indústrias que fabricam em pequenos lotes. Nesse sentido, as pequenas empresas apresentam uma tendência maior que as grandes, em combinar as funções de fabricação e programação.

A tecnologia microeletrônica tem influenciado o nível de qualificação dos trabalhadores em diferentes setores. No setor da imprensa tem se verificado que o tempo necessário para se formar um tipógrafo pleno era de sete anos. O tempo necessário para formar um operador pleno para utilizar equipamentos de tipografia computadorizados é em torno de seis meses.

No setor metalúrgico, vem acontecendo algo pior, a informatização tem proporcionado aos empregadores a possibilidade de desqualificar o trabalho dos técnicos e de reduzir os custos de mão-de-obra. Nesse contexto, um dos fatores determinantes da qualificação do operário, tem sido a amplitude com que se tem delegado ao trabalhador qualificado a programação das máquinas. Quanto mais se delega aos engenheiros e programadores esta função, mais desqualificado se torna o operário, criando com isso uma polarização de qualificações entre os que programam a máquina e os que a operam. Entretanto, existe um grande risco de degradação das aptidões do trabalhador qualificado. Primeiro, porque a nova tecnologia pode ser introduzida conservando-se a qualificação dos técnicos. Segundo porque as máquinas informatizadas estão se tornando, cada vez mais fácil de serem operadas. Nesse sentido, acredita-se que o caminho mais viável e seguro para toda a classe trabalhadora é investir no conhecimento a fim de se adquirir a habilidade mental para atuar/participar nas fábricas informatizadas, que daqui para frente se farão presentes, cada vez mais em todos os setores produtivos. A seguir será discutida a especificação de projetos como sendo uma das tarefas mais importantes na área das engenharias e conseqüentemente da automação.

1.8 Especificação de projetos em Automação

Do ponto de vista semântico a especificação é definida como o ato ou efeito de especificar, descrição rigorosa e minuciosa das características que um material, uma obra ou um serviço deverá apresentar. Especificar significa indicar a espécie, explicar com pormenores, apontar individualmente, especializar, explicar mudamente, determinar o modelo preciso e específico. E, o específico se refere a espécie particular, especial, exclusivo.

Na medida em que as ciências se tornaram autônomas, dando origem às especializações, que por sua vez, possibilitaram a decomposição de um problema em partes, tornou-se necessário

a síntese, a integração das soluções parciais em uma solução global. Entretanto, para se chegar a uma solução global é preciso formar uma equipe de especialistas interdisciplinar de forma que essa equipe possa se comunicar de forma mais eficiente na solução de um problema. É preciso que haja uma linguagem comum. Essa é uma abordagem que se aplica a sistemas: Sistemas políticos, administrativos, econômicos, sociais, dentre outros. (Mendonça, 1972). Todavia, esta mesma abordagem pode ser utilizada para administração de projetos, entendendo-se aqui, a elaboração de projetos mecânicos de dispositivos mecatrônicos que é o escopo deste trabalho. Neste caso, ao se iniciar um projeto precisamos definir: os objetivos do projeto, os seus requisitos, o desenvolvimento da árvore de especificação e a produção das especificações. Sendo a especificação um documento completo com todos os requisitos que o projeto deve atender. No contexto do ambiente de aprendizagem baseado no uso de dispositivos robóticos, dependendo do nível do ensino onde esse ambiente é usado e da complexidade do dispositivo a ser implementado a especificação pode ser discutida como uma ferramenta que se aplica com maior ou menor profundidade e rigor.

Nos âmbitos das engenharias mecânica, de computação, elétrica, dentre outras, para se criar um novo projeto ou alterar/modificar outro já existente é preciso especificar o que se deseja. A especificação serve para eliminar dúvidas, incertezas ou opções, dando informação clara, concisa e inequívoca sobre o projeto. Portanto, nela não podem existir ambigüidades, nem falhas, de maneira que as diferentes etapas do projeto sejam de fácil compreensão e interpretação.

Na área de Engenharia de software, por exemplo, a especificação é a descrição sistemática daquilo que o software deve fazer a partir do que foi analisado. Neste contexto, a análise e especificação de requisitos de software envolvem as atividades de determinar os objetivos de um sistema de software e as restrições a eles associados (Leite, 1999).

Na área de engenharia mecânica geralmente a complexidade dos projetos é grande advindo do fato de que, normalmente, não existe um diálogo entre as diversas equipes que contribuem, em diferentes momentos, no projeto, implementação, montagem, automação e controle de um determinado dispositivo. Portanto, é importante que as diversas partes em que se constituirá o dispositivo seja especificada da melhor maneira possível. Assim, ao se montar às partes, pode-se

garantir o funcionamento de todo o dispositivo. O fato de poder existir diversos grupos trabalhando no mesmo projeto, em momentos diferentes, evidencia a existência de diferentes níveis de especificação, a serem levados em consideração, começando pela idéia da concepção do dispositivo em si, passando pela elaboração do seu projeto mecânico, pelo processo de fabricação das peças que o compõem, pela montagem e pelo funcionamento dele como um todo.

Na área de Engenharia de Sistemas, ela é um documento que fornece ao realizador toda a informação necessária para obter um sistema, subsistema ou componente conforme a concepção do idealizador (Mendonça, 1972). Esse documento é composto de objetivo e características das especificações e, especificações preliminares e finais.

Nesse sentido, na elaboração de projeto de automação, a especificação é uma etapa imprescindível do processo. Ela tem que ser clara, sem ambigüidades, sem falhas, para que, as várias etapas do processo de fabricação, por exemplo, seja de fácil compreensão e interpretação. No caso do dispositivo robótico isso é tido como garantia de que na operação, na automação e no controle deste dispositivo não ocorrerá falhas. Pois, durante a operação não é possível realizar a experimentação. Espera-se que o dispositivo funcione, para que seja realizado uma determinada operação com sucesso.

1.8.1 Objetivo das especificações

Diz respeito à necessidade que temos de especificar o que desejamos. Especificar para eliminar dúvidas, para caracterizar com mais clareza o que queremos.

1.8.2 Características das especificações

Diz respeito às características que as especificações devem ter. São elas: objetividade, suficiência, exaustividade, ausência de exigências desnecessárias, compatibilidade e formato.

Objetividade:

Uma especificação deve descrever apenas o necessário à realização do seu objetivo. Ela deve estabelecer os requisitos a serem satisfeitos pelos componentes do sistema em linguagem concisa e clara.

Suficiência:

Deve permitir que o realizador, guiado apenas pelo documento, consiga realizar o especificado.

Exaustividade:

Explicitação, de forma minuciosa, de tudo relacionado ao projeto.

Ausência de exigências desnecessárias:

Uma especificação não deve conter exigências desnecessárias. É contraproducente introduzir na especificação exigências que não irão contribuir para melhorar aquilo que realmente desejamos.

Compatibilidade:

As especificações devem ser escritas de forma estruturadas de modo a facilitar a visualização da compatibilidade de todo o projeto.

Formato:

As especificações devem ser escritas dentro de certos padrões de forma, com indexação que favoreça a referência como, por exemplo, as normas do Grafset, que serão apresentadas no capítulo 3.

1.8.3 Especificações preliminares e finais

Especificações preliminares são aquelas elaboradas no estágio inicial de desenvolvimento do projeto. Especificações finais são aquelas elaboradas no estágio final de desenvolvimento do projeto. Da fase preliminar à fase final de elaboração as especificações sofrem alterações objetivando otimizações em diversos aspectos.

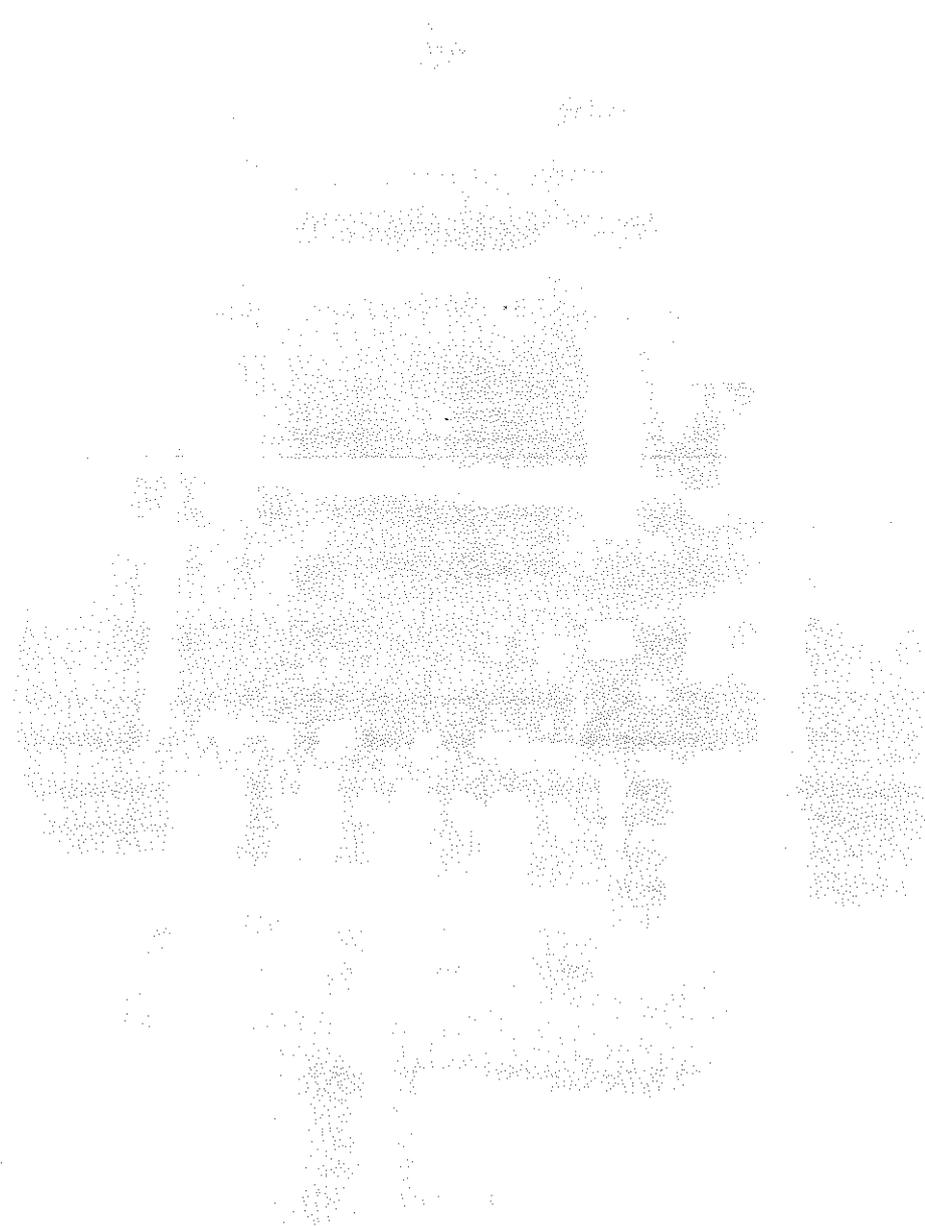
1.9 Considerações sobre o Capítulo 1

Este capítulo discutiu a automação do ponto de vista histórico e também tecnológico no que consiste a sua utilização como sistemas mecânicos, eletrônicos à base de computadores. Enfocou-se a forma como inovações tecnológicas que têm como base à informatização dos meios de produção, automação flexível e novas formas de gestão da empresa, vem alterando os meios de trabalho ao longo dos séculos implicando em desenvolvimento de técnicas e de ferramentas de especificação como formas de se sintetizar a integração das soluções parciais em uma solução global.

Ao longo do capítulo, apresentou-se a questão da educação e da tecnologia. O papel importante que o trabalho artesanal exerceu em determinado momento da história da humanidade e sobre o descompasso do processo de ensino-aprendizagem quando comparadas com as exigências do mercado de trabalho. Para o cidadão do terceiro milênio, esse descompasso tem que ser, pelo menos, minimizado. Esse cidadão terá que ser capaz de interferir e se envolver na gestão e no funcionamento da empresa onde ele atua. Portanto, é necessário que a escola, a fábrica ou mesmo a sociedade forme cidadãos com perfis diferentes do que vem sendo formado até então. É necessário que as inovações tecnológicas sejam incorporadas ao processo de formação desses cidadãos. Isso significa e justifica a necessidade de se criar condições para melhorar o nível educacional da classe trabalhadora para que ela adquira conhecimentos que permitam-na absorver as constantes atualizações tecnológicas exigidas pelo mercado de trabalho. Por meio da utilização de ambientes de aprendizagem onde o computador, associado a outras ferramentas, permite implementar modelos que sejam o mais próximo possível do modelo real, para que durante o processo de formação o sujeito (trabalhador na fábrica ou aluno na sala de aula) adquira competência pessoal, técnica, intercultural, social, dentre outras. São essas competências que certamente o ajudará a trabalhar em equipe, a liderar projetos e implementar/especificar rotinas de trabalho que poderão ser utilizadas por outros grupos em locais geograficamente distantes. Nesse contexto, é importante que os ambientes de aprendizagem, sobretudo as instituições de ensino da área tecnológica sejam dotados de *kits*

laboratoriais⁶ e de outros recursos que permitem introduzir o aprendizado da automação, não só voltada para os seus aspectos técnicos mas também e, principalmente, na possibilidade de se integrar de forma interdisciplinar diferentes áreas de conhecimento. Uma discussão sobre automação no contexto de ambientes de aprendizagem será apresentada no capítulo a seguir.

⁶ São kits constituídos de componentes eletro-eletrônicos que permitem a implementação de ambientes de robótica pedagógica. Fazem parte desses kits, dentre outros materiais, atuadores, sensores, interfaces para conversão de sinais, interfaces para acionamento de transdutores em geral via computador. Os kits LEGO possuem alguns desses componentes.



Capítulo 2

Automação no contexto de ambientes de aprendizagem

A automação é utilizada no contexto educacional, numa abordagem denominada robótica pedagógica que envolve a utilização de materiais: softwares, *kits* educacionais do tipo LEGO, manipuladores robóticos do tipo Robix, atuadores e sensores de diversos tipos em diferentes níveis de ensino. Esta abordagem, que é fortemente utilizada por profissionais da área de informática aplicada à educação, consiste em utilizar o computador integrado com esses materiais para auxiliar o processo de aquisição de conceitos em ambientes formais de aprendizagem, como salas de aulas, e também, em ambientes não formais como museus, fábricas, parque de diversões, dentre outros. Nesse processo, o que se pretende é ampliar a forma de utilização do computador fazendo com que ele, atuando sobre objetos externos, enriqueça e diversifique a forma de se utilizá-lo como ferramenta educacional, num contexto chamado de informática na educação.

2.1 O que é a Informática na Educação

Segundo a literatura, existem diferentes maneiras de se classificar o uso do computador na educação. Taylor (citado em Kenzek, et al, 1988) categoriza o uso do computador em três modos: **o computador como tutor** - o programa dirige o estudante; **o computador como ferramenta** - o estudante usa o computador para realizar tarefas e **o computador como tutelado** - o estudante ensina o computador.

Um outro enfoque é dado por Berger e Carlson, que dividem o uso do computador em duas categorias, conforme a ênfase que é dada ao ensino ou à aprendizagem: **ensino por exposição** - ocorre nos programas tipo exercício e prática, que progridem em pequenos incrementos, usados para basicamente ensinar memorização, o professor controlando a atividade do aluno; **aprendizagem por descoberta** - ocorre com as simulações e jogos educativos, que possibilitam aos alunos usar o computador para explorar habilidades de resolução de problemas (Stahl, 1990).

Pode-se afirmar que existe uma diferença entre ensino de informática e ensino por intermédio da informática. O ensino de informática caracteriza-se no uso do computador como objeto de estudo para se adquirir conhecimentos sobre o próprio computador e princípios computacionais de programação. O ensino por intermédio do computador caracteriza-se no uso do computador como ferramenta de ensino aprendizagem de propósito geral. Na modalidade de ensino por intermédio da informática pode-se incluir o ambiente de robótica pedagógica que será descrito a seguir.

2.2 Ambiente de Robótica Pedagógica

Um ambiente de robótica pedagógica pressupõe a existência de professor, aluno e ferramentas que propiciam a montagem, automação e controle de dispositivos mecânicos. Alunos e professores interagindo entre si e com essas ferramentas produzem novos conhecimentos caracterizando esse ambiente como um ambiente pedagógico que não existe *a priori*.

Desse cenário podem fazer parte dispositivos como montagens LEGO, manipulador robótico, robô móvel, tartaruga mecânica de solo, traçador gráfico educacional, dentre outros. O ambiente que mais se utiliza, em diversas instituições educacionais, nesta área tanto no Brasil quanto no exterior tem sido: LEGO-Logo, LEGO *Mindstorms*, *Controlab*, *Robotic Command Explorer* – RCX e *Robolab*. LEGO-Logo significa montar dispositivos LEGO e controlá-los via

computador utilizando a linguagem de programação Logo⁷. Uma discussão sobre esses ambientes será apresentada nos próximos itens deste capítulo.

Em um ambiente de robótica pedagógica, na **montagem** de dispositivos pode-se utilizar **peças mecânicas** tais como rosca sem-fim, engrenagens, eixos, cremalheiras, correias dentadas etc., para montar estruturas mecânicas. Essas peças são devidamente acopladas respeitando alguns princípios da Mecânica, da Física, da Matemática, buscando trabalhar os conceitos de uma forma interdisciplinar. Além disso, utilizam **componentes elétricos** como motores, sensores (de luz, toque, temperatura, som, posição), lâmpadas que possibilitam o acionamento dos dispositivos.

Em um ambiente de robótica pedagógica, na **automação e controle** dos dispositivos elabora-se programas utilizando software do tipo SuperLogo⁸, por exemplo, ou *Robolab* que são softwares elaborados com propósitos educacionais. Além disso, podemos utilizar o tijolo programável, *Robotic Command Explorer (RCX)* da LEGO.

Em um ambiente de robótica pedagógica, **automatizar** um dispositivo mecânico que representa uma máquina significa criar uma interação por intermédio do programa elaborado entre as peças mecânicas e os componentes elétricos. Ou seja, precisamos descrever para o computador, em termos de uma seqüência lógica devidamente estruturada, os comandos que acionam luzes, motores e sensores conectados a uma determinada máquina, para que ao colocá-la em funcionamento essa seqüência lógica represente, mais próximo possível, a máquina projetada/idealizada. Caso isso não ocorra, abre-se uma gama de possibilidades de depuração propiciando a interação com diversas áreas de conhecimento numa abordagem interdisciplinar (D'Abreu, 1995).

⁷ A linguagem de programação Logo foi desenvolvida no Massachusetts Institute of Technology (MIT), Cambridge, Estados Unidos da América por volta de 1968-1969, por um grupo de pesquisadores liderados por Seymour Papert. O Logo era uma linguagem de programação que processava listas e permitia criar novos procedimentos. Mais informações sobre o Logo podem ser obtidas no site: http://el.www.media.mit.edu/groups/logo-foundation/resources/links_org.html

⁸ SuperLogo é uma versão da Linguagem Logo desenvolvida pela Universidade de Berkley (EUA), traduzida para o português pelo Núcleo de Informática Aplicada à Educação - NIED da Universidade Estadual de Campinas - UNICAMP.

2.3 Ambiente de robótica pedagógica LEGO-Logo

LEGO-Logo é um ambiente de robótica pedagógica que originariamente utilizava kits de montar educacional da LEGO Dacta e a linguagem de programação Logo. Os ambientes LEGO-Logo desenvolvidos no Núcleo de Informática Aplicada à Educação NIED/UNICAMP utilizaram diferentes versões do Logo como Logo MSX, Logo Writer, Logo Controlab e SuperLogo. Entretanto a denominação LEGO-Logo é mantida pelo NIED nos âmbitos da proposta educacional Logo desenvolvida por Seymour Papert, (Papert, 1985). A versão da linguagem Logo que atualmente vem sendo utilizada no NIED é o SuperLogo acrescido de comandos que permitem o controle de dispositivos mecânicos automatizados no modo local e no modo remoto utilizando recursos da rede Internet. Isso tem permitido desenvolver atividades de telerobótica onde qualquer usuário da rede Internet, a partir de um endereço de *site*⁹ pode em qualquer lugar do mundo acessar o laboratório de robótica educacional, controlar e visualizar na tela do seu computador os movimentos dos dispositivos sob o seu comando mantendo contato síncrono (*on-line*), via *chat* com o laboratório.

2.3.1 O que se ganha em termos de aprendizagem quando se cria o ambiente LEGO-Logo?

Esse ambiente tem fornecido inúmeras oportunidades de aprendizado na medida em que a atividade de montagem e controle de dispositivos propicie a construção de conhecimento em diversas áreas. A atividade LEGO-Logo envolve aprender sobre:

- a) cada componente LEGO e sua utilização como elemento mecânico ou eletrônico de um dispositivo;
- b) montar e controlar dispositivos baseados nos mesmos princípios mecânicos e físicos utilizados na construção das máquinas que estão no nosso dia-a-dia;
- c) conceitos específicos do dispositivo que está sendo construído. Por exemplo, na construção de um veículo a pessoa tem a oportunidade de manusear idéias de planejamento e montagem, e obter como produto um dispositivo com características de um veículo. Além disso, a

aplicação de conceitos de velocidade, peso, atrito, força, deslocamento, dentre outros no controle do carro;

- d) conceitos de controle de processos. No caso da construção do veículo, este pode ser controlado pelo computador e, assim, pode ser inserido num contexto onde existe um semáforo, outros veículos, etc.;
- e) montagem de um ambiente de simulação de uma linha de produção de uma fábrica que utiliza sistema automatizado de produção composta de esteira rolante, braço mecânico, elevadores, etc. que integram células que fazem parte da linha.

2.4 LEGO-Logo na Escola

Em uma sala de aula o ambiente LEGO-Logo permite criar uma representação do real criando-se a possibilidade de se apre(e)nder a realidade por meio da brincadeira pois, segundo Geus, “o sujeito que brinca está simplesmente fazendo experiências com um objeto que de alguma forma representa a realidade. Brincar é experimentar um brinquedo que o sujeito que brinca aceita como representativo de sua realidade. Isso torna o brinquedo uma representação do mundo real com o qual o aprendiz pode fazer experiências sem temer as conseqüências. Sob toda aura de diversão existe uma finalidade muito séria: brincar com a própria realidade permite que o sujeito entenda mais o mundo em que vive. Brincar é aprender”, (Geus, 1998 p. 53). Desenvolver atividades no ambiente LEGO-Logo nos propicia esse brincar como forma de aprender.

No ambiente LEGO-Logo podem ser trabalhados conceitos de:

Educação artística: Na observação do objeto real e a construção do protótipo.

Geometria: Na análise das formas geométricas.

Ciências: Na determinação de massa, peso, velocidade, espaço e tempo.

⁹ <http://www.nied.unicamp.br/~siros/>

Física: No cálculo do movimento uniformemente variado, força resultante, normal, aceleração gravitacional.

Matemática: No cálculo do intervalo de tempo, deslocamento e velocidade.

Português: Na elaboração do relatório e discussão do projeto.

Programação: Na elaboração de procedimentos que controlam as máquinas.

Informática: No uso das ferramentas do *Windows*.

Esses conceitos são trabalhados buscando resgatar as relações interdisciplinares que normalmente permeiam o ambiente LEGO-Logo (D'Abreu, 1995). Sob esse enfoque, a atividade de robótica pedagógica no ambiente LEGO-Logo é interdisciplinar porque ela envolve um rigor científico, que implica na mudança de atitude do professor, que por sua vez refletirá na mudança de postura do aluno frente ao conhecimento.

2.5 LEGO-Logo na Fábrica

O ambiente LEGO-Logo foi introduzido na Delphi-Harrison¹⁰, por intermédio de uma oficina de trabalho, buscando ajudar os técnicos especializados a trabalhar, de forma concreta, conceitos de automação que eles usavam empiricamente (Valente, 1997). Dessa oficina participaram engenheiros de controle de qualidade, técnicos da linha de montagem e técnicos de manutenção. Esses trabalhadores construíram máquinas similares às existentes na fábrica, tornando assim mais significativa a sua aprendizagem.

Os dispositivos implementados podiam ser divididos em duas categorias: primeiro, os que se preocupavam com o aprimoramento da segurança das operações executadas na célula de trabalho. Segundo, aqueles que ajudavam na compreensão de um mecanismo específico. Nesse contexto, definiu-se mecanismo como sendo uma associação/montagem de elementos mecânicos que levam em conta alguns princípios científicos. Por sua vez, a montagem pode se constituir em

¹⁰ Delphi-Harrison era uma indústria automotiva, localizada em Piracicaba – SP, que produzia componentes para a indústria automobilística e praticava idéias de produção enxuta.

uma estrutura mecânica que pode ser um dispositivo mecatrônico que tem funcionalidade no sentido de girar, andar, elevar, transportar, enfim, gerar movimento produzindo trabalho.

Da primeira categoria, foi construído um sistema de segurança que impedia que a máquina entrasse em operação quando o operário estivesse com as mãos em uma determinada área definida como área de segurança. A segunda categoria, envolveu a construção da máquina de solda automática que utiliza recursos tecnológicos que a máquina de solda real da fábrica não tinha, pois, esta não era automática. A máquina de solda automática será descrita no capítulo 4.

Um outro exemplo de projeto desenvolvido nessa fábrica, que envolvia o mesmo contexto do projeto LEGO-Logo, foi o de criação ambientes para aprendizagem de conceitos de Controle Estatísticos de Processos - CEP (Schlünzen, 2000).

2.6 Implementação de Ambientes Educacionais: utilizando softwares diversos

Dentro dessa categoria de ambientes podemos incluir os implementados usando softwares tais como: Everest, Robolab e Coach Junior. A seguir, será descrito de forma sucinta cada um desses ambientes.

2.6.1 Ambiente Everest

O Everest¹¹ é um software de autoria, uma espécie de "oficina de criação", equipado com diversas ferramentas que permitem o desenvolvimento de projetos multimídia. Como software de autoria ele é um software aberto que permite que alunos e professores possam criar seus próprios conteúdos, não precisando mais se adaptar aos produtos fechados. Este software permite criar aplicações de forma fácil, sem necessitar de conhecimentos de programação, agregando elementos como sons, imagens, vídeo, textos, animações e bancos de dados. A Figura 2.1 mostra uma interface eletrônica denominada de Super Robby implementada para automação e controle de dispositivos mecânicos automatizados utilizando o Everest.

¹¹ <http://www.complex.com.br/>

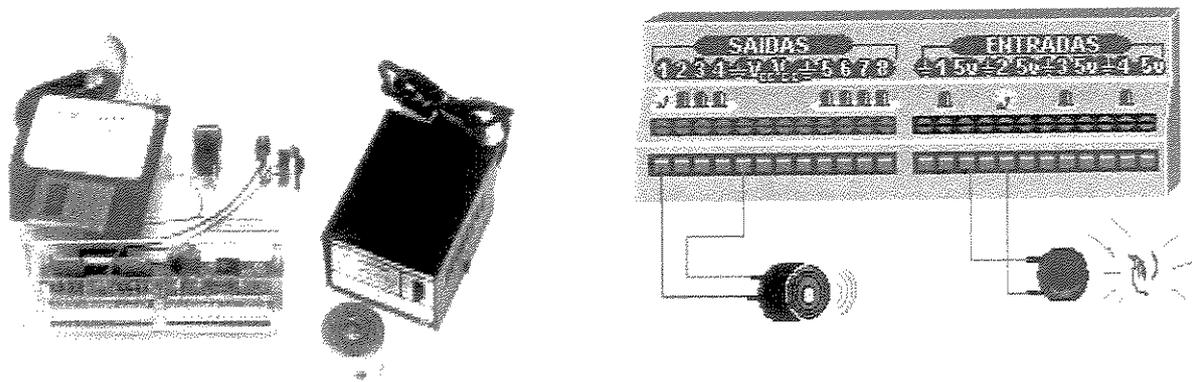


Figura 2.1 – Interface Super Robby

2.6.2 Ambiente Robolab

Robolab é um software de programação, desenvolvido especificamente para ser utilizado na programação do Robotic Control X - RCX e, que utiliza uma linguagem totalmente gráfica, baseada em ícones.

O Robolab foi desenvolvido em parceria entre a National Instruments¹² e a Tufts University - College of Engineering¹³. Este software que se baseia no LabVIEW™ da National Instruments, possui duas áreas principais, uma chamada Administrador e a outra chamada Programador.

A programação do RCX no Robolab (Figura 2.2) apresenta duas opções chamadas de **Pilot** e **Inventor** cada uma com 4 níveis.

A fase Pilot compreende uma série de modelos com um formato fixo associado a eles. A programação Pilot usa uma interface fácil, do tipo Clique & Escolha, num modelo que pode ser alterado conforme a necessidade.

¹² <http://sine.ni.com/apps/we/nioc.vp?cid=1381&lang=US>

¹³ <http://www.cceo.tufts.edu/graphics/fl/>

Programação no modo Inventor

A programação no modo Inventor é uma seqüência da fase Pilot e também apresenta 4 níveis, sendo o Inventor 1 o mais simples e o Inventor 4 o que apresenta total flexibilidade e possibilidades ilimitadas.

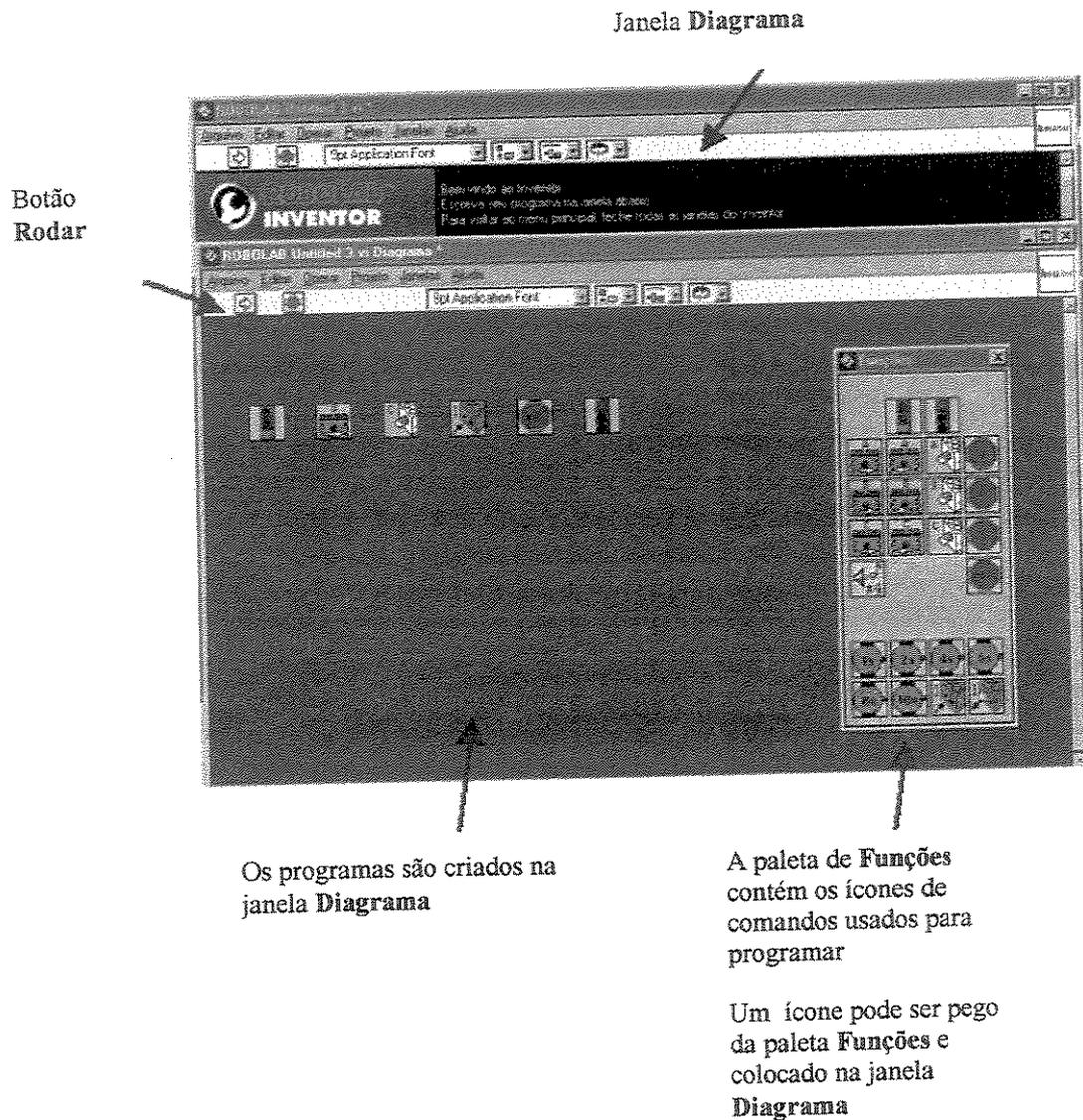


Figura 2.2 - Tela do Robolab

A fase Inventor usa os mesmos ícones de comandos da fase Pilot. Além disso, são acrescentadas várias opções de comando conforme o usuário avança pelos níveis. O nível

Inventor é uma modalidade de uso que evidencia o potencial do RCX. A interface de programação do Inventor é diferente da utilizada no Pilot. A interface é do tipo Pegar & Colocar. A elaboração de programas neste ambiente significa selecionar uma certa quantidade de objetos e interliga-los formando uma seqüência lógica. Robolab é o primeiro ambiente que os alunos aprendem a trabalhar quando iniciam as atividades de montagem automação e controle de dispositivos mecatrônicos. No anexo 1 (**Software Robolab**) encontra-se descrito o Robolab em detalhes.

2.6.3 Ambiente Coach Júnior

Coach Júnior é um pacote de software de autoria desenvolvido pela Foundation CMA - Centre for *Microcomputer Applications* que possibilita controlar as interfaces LEGO Dacta Control Lab, Lego RCX, CMA Coachlab entre outros.

Ambiente de programação do Coach Júnior

O ambiente de programação permite o acionamento das saídas para controle de atuadores e ativar as entradas para leitura dos sensores conectados a interface.

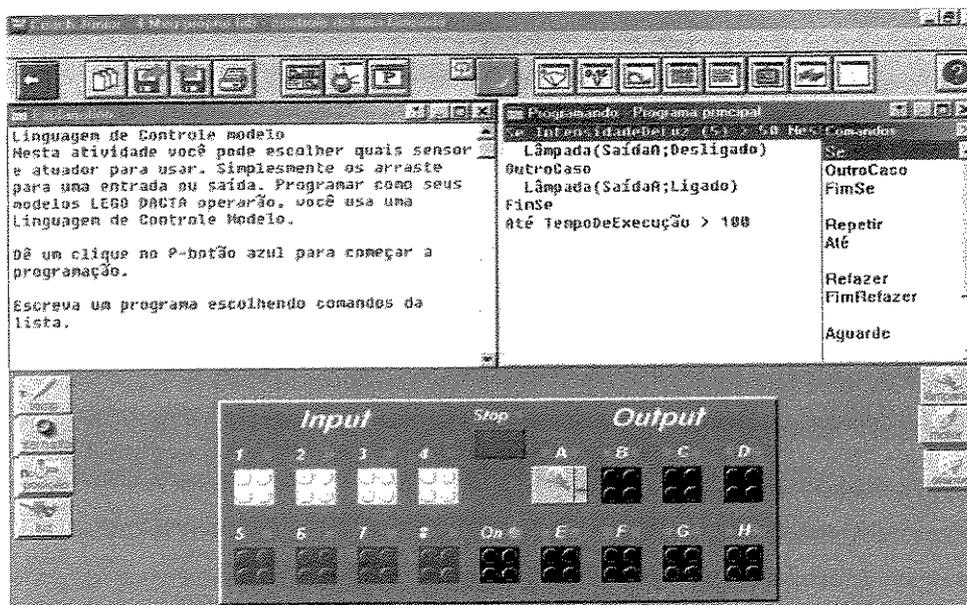


Figura 2.3 - Tela de programação e controle do Coach Júnior

Utilizando o Coach Júnior podemos controlar dispositivos mecânicos automatizados construídos com LEGO ou outro tipo de material.

2.7 Implementação de Ambientes Educacionais: com base no Logo

Dentro desta categoria de ambientes educacionais podem ser incluídos os seguintes ambientes: LEGO-Logo para MSX, LEGO TcLogo, LEGO-Logo para Control Lab, LEGO Mindstorms Robotic Control X – RCX, SuperI e SIROS (Sistemas Robóticos com SuperLogo). A seguir, será descrito de forma sucinta cada um destes ambientes.

2.7.1 Ambiente LEGO-Logo para MSX

Do ponto de vista de implementação de ambientes de aprendizagem baseados na utilização de dispositivos mecânicos automatizados o ambiente LEGO-Logo para MSX (computador de 8 bites) foi o primeiro ambiente por nós implementado. Este ambiente consistia de um conjunto de peças LEGO que permitia a montagem de dispositivos (máquinas e/ou animais) e de um conjunto de comandos Logo que possibilitava a elaboração de programas para controle destes dispositivos. Um dos modelos deste ambiente foi o implementado em 1989 no NIED/UNICAMP baseado na utilização do Software Hot-Logo da Sharp ou do MSX Logo da Gradiente, conforme as Figuras 2.4 e 2.5.

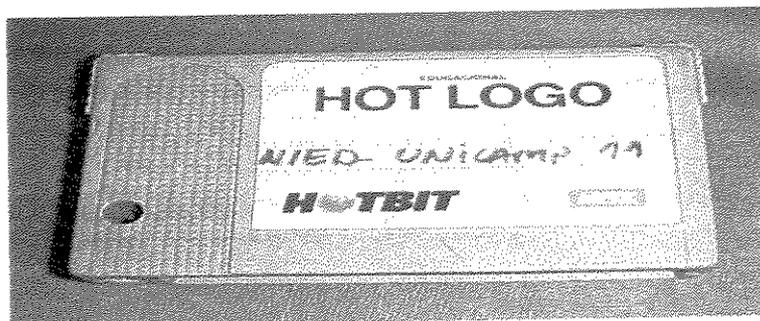


Figura 2.4 Cartucho Logo para MSX



Figura 2.5 - Microcomputador MSX e interface para controle de componentes elétricos LEGO

Na implementação deste ambiente, ao software Logo original foram acrescentados comandos para controle de uma interface eletrônica contendo, portas de entrada e saídas. A interface eletrônica se conectava a saída paralela do microcomputador 8 bits MSX.

2.7.2 Ambiente LEGO TcLogo

Este ambiente é uma versão do software Logo Writer, em inglês, acrescentado de comandos que permitem o controle da interface LEGO 70288, (figura 2.7), que possibilita a implementação de programas computacionais. O TcLogo foi desenvolvido para DOS em 1988 utilizando-se uma metáfora de páginas na definição e execução de procedimentos e na manipulação de arquivos.

O TcLogo possui comandos que acionam uma interface eletrônica conectada ao microcomputador por meio de uma placa inserida no barramento deste.

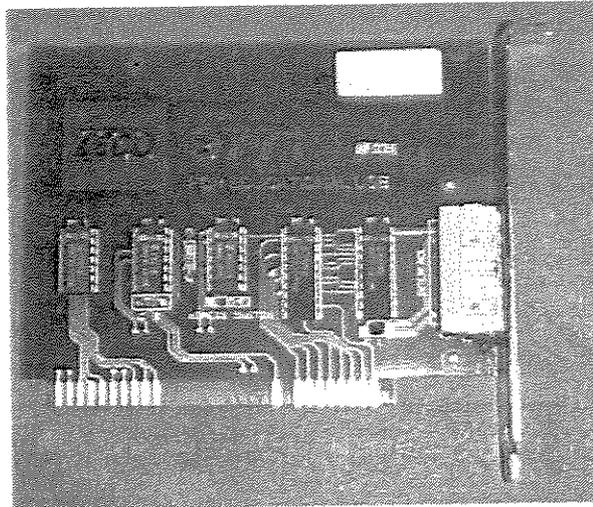


Figura 2.6 - Placa de comunicação microcomputador interface eletrônica

Na interface eletrônica, lâmpadas, motores e sensores, que fazem parte do dispositivo mecânico, são acionados.

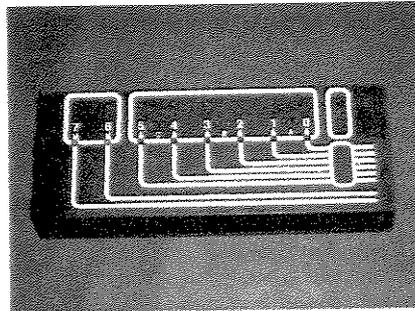


Figura 2.7 - Interface eletrônica LEGO 70288

```
to controle
  lto 2 on
  lto 6
  reseto
  ver
end

to ver
  if sensor? [lto "a setodd onfor 30 wait
  10 rd onfor 30]
  ver
end
```

Figura 2.8 - Tela com um programa para controle da porta automática em TcLogo

2.7.3 Ambiente LEGO-Logo para Control Lab

O Software LEGO Dacta Control Lab versão 1.2, desenvolvido pela LCSI - Logo Corporation Systems Inc. em 1993, para o ambiente DOS, é também um software baseado na idéia do Logo Writer com primitivas que permitem o controle de portas de entradas e saídas na interface eletrônica 70909.

Diferentemente dos outros Logo's, o Control Lab não possui a Tartaruga gráfica do Logo. Entretanto ele possui uma caixa de ferramentas que representa a idéia de instrumentos de um laboratório, (Figura 2.11).

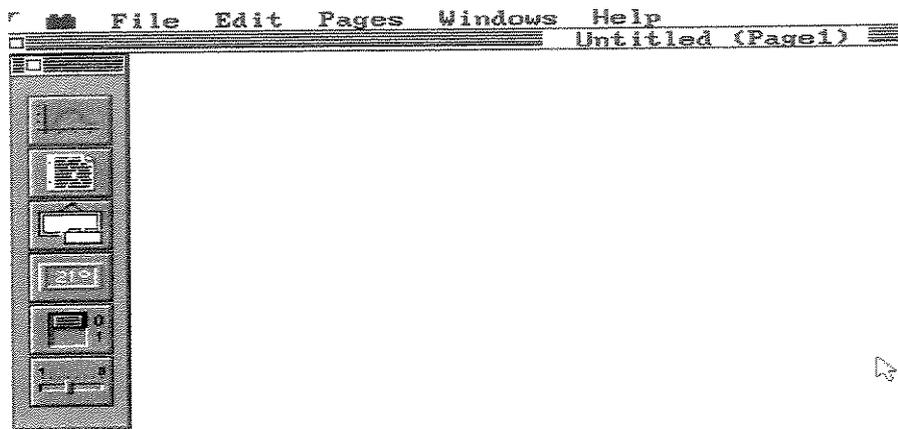


Figura 2.9 - Tela com as caixas de ferramentas

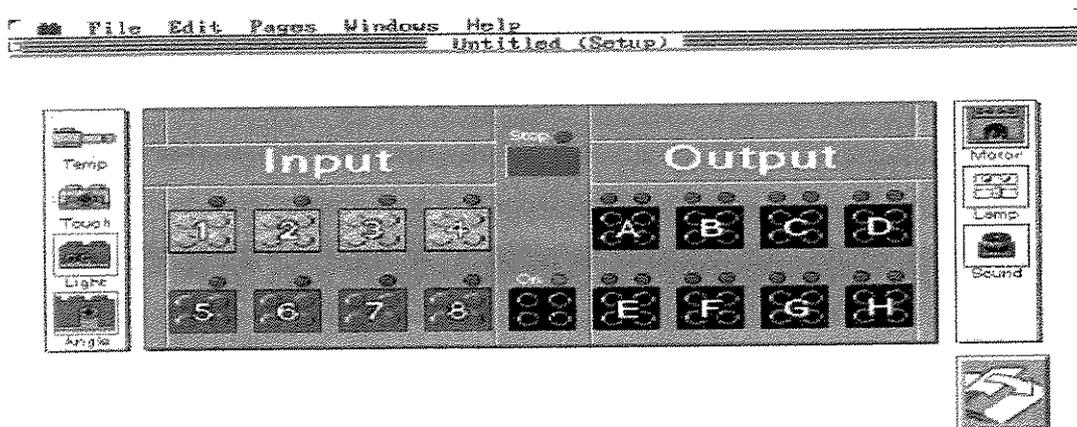


Figura 2.10 - Tela de trabalho da interface

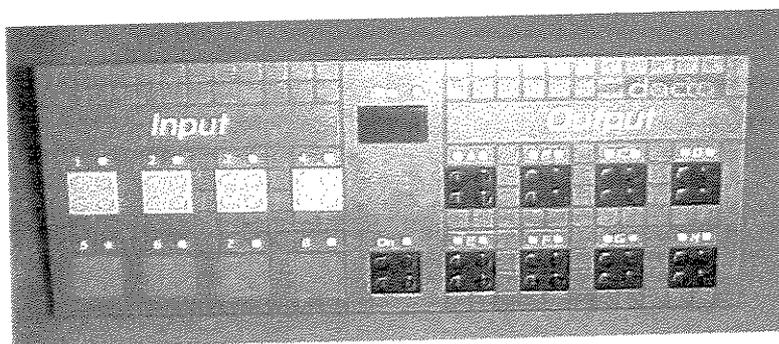


Figura 2.11 - Interface eletrônica LEGO 70909

A interface LEGO da Figura 2.11 possui 6 saídas para controle de motores, lâmpadas, buzinas e 6 entradas para controle de sensores de toque, luz, temperatura e posição. A comunicação desta interface com o computador é feita via porta serial.

2.8.4 Ambiente LEGO Mindstorms¹⁴ Robotic Control X - RCX¹⁵

Mindstorms é o nome de uma das linhas de produtos comercializados pela LEGO. O “carro chefe” dessa linha de produtos é o Robotic Control X - RCX.

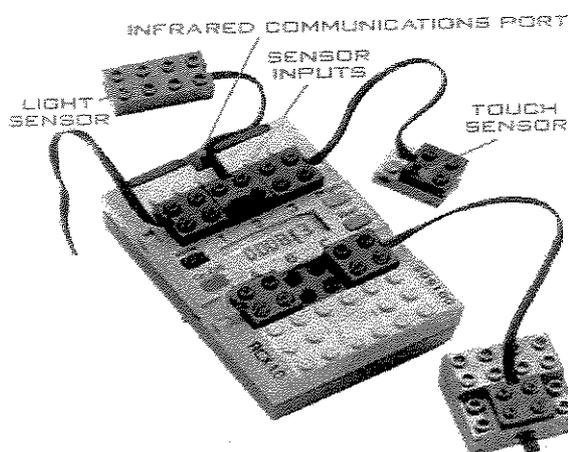


Figura 2.12 – Tijolo programável RCX com alguns componentes elétricos conectados

¹⁴ Mindstorm, (RCX) é um tijolo programável da LEGO que possui 3 entradas para sensores, 3 saídas para motor, luz e buzina, e um transmissor infra-vermelho, que permite a comunicação com o microcomputador.

¹⁵ O LEGO RCX é um tijolo programável constituído de 3 entradas para sensores e três saídas para controle de motores e lâmpadas.

O RCX se comunica com o microcomputador por intermédio de uma conexão, sem fio, feita a partir de uma torre que emite sinais na faixa de infravermelho. Esta torre, que se conecta à porta serial do microcomputador, permite também a comunicação entre dois RCX.

Software

Uma vez escrito um programa para controle de um determinado dispositivo robótico, este programa tem que ser carregado no RCX, através da torre, para ser executado.

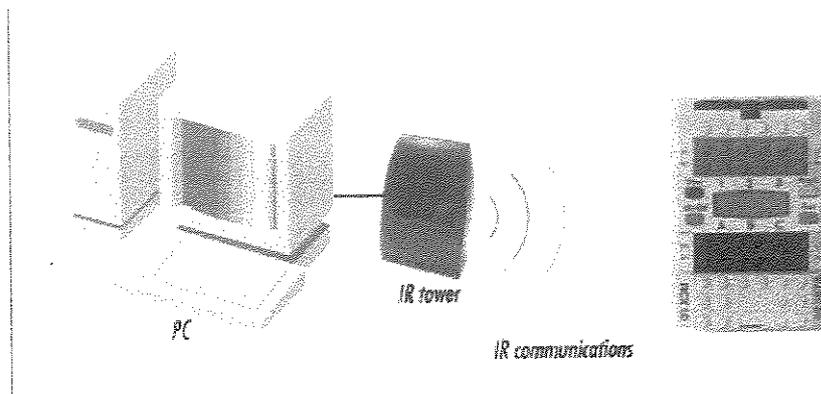


Figura 2.13 - Comunicação entre microcomputador e RCX

Trabalhar no ambiente LEGO RCX consiste basicamente de quatro passos:

- a) construir o dispositivo robótico/mecatrônico propriamente dito;
- b) escrever um programa utilizando as ferramentas de software existentes no microcomputador. Aqui, escrever programas para o controle de dispositivos robóticos significa, agrupar segundo uma certa lógica, comandos que permitem acionar motores sensores e lâmpadas;
- c) carregar o programa no RCX;
- d) executar o programa.

Exemplo de um programa no ambiente LEGO Mindstorms/RCX

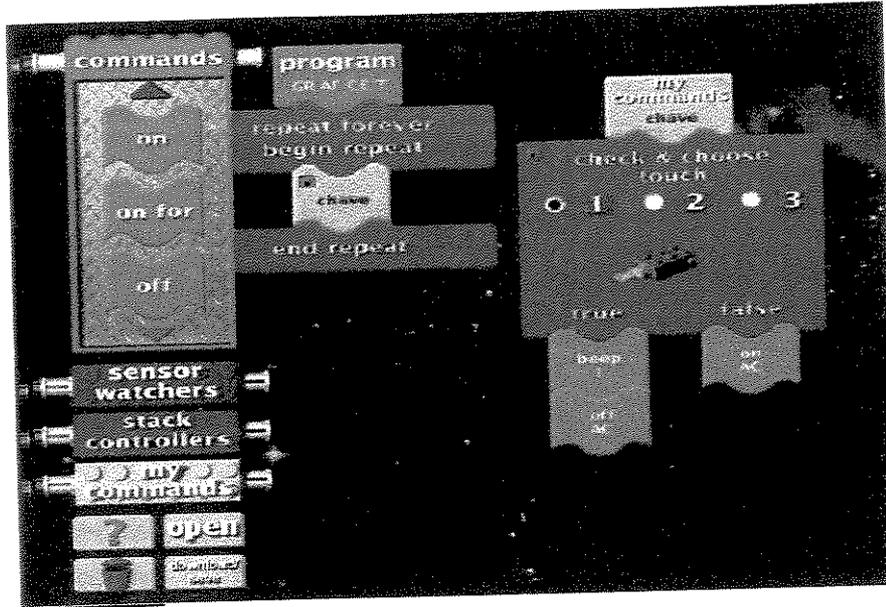


Figura 2.14 - Tela de programação do Mindstorms

No anexo 2 (Introdução ao RCX) Mindstorms/RCX está descrito com maiores detalhes.

2.8.5 Ambiente SuperI

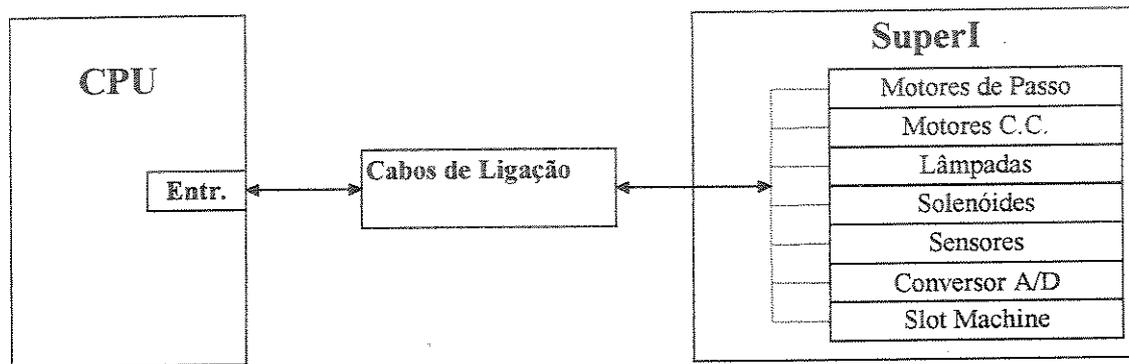
Esse ambiente é um projeto de pesquisa desenvolvido no NIED/UNICAMP que consistiu em criar uma interface eletrônica que possuía diversos módulos controlados pelo software educacional SLogoW¹⁶ também desenvolvido no NIED. SuperI possibilitava o acionamento de diversos tipos de atuadores e sensores. Este ambiente será descrito de forma sucinta a seguir.

¹⁶ SLogoW é um Interpretador Logo para ambiente windows compatível com microcomputadores da linha PC/AT desenvolvido pelo NIED-UNICAMP.

Descrição dos módulos da SuperI

Os diferentes módulos da SuperI se comunicavam com o microcomputador através da porta de saída serial, sendo que o SLogoW permitia a comunicação, em tempo real, com um ou mais módulos de forma concomitante. Isso possibilitava, por exemplo, o acionamento de motores de passo e em seguida o de corrente contínua num mesmo dispositivo. As primitivas desenvolvidas para a comunicação com a SuperI foram implementadas de tal forma que, ao elaborar procedimentos em SLogoW, elas eram utilizadas juntamente com as primitivas Logo já existentes, guardando todas as características da linguagem Logo original.

Diagrama de blocos da SuperI



Os recursos de hardware implementados na placa dessa interface são descritos de maneira mais detalhada a seguir:

Módulo controlador de motores de passo:

Esse módulo consistia de circuitos que permitiam o acionamento de dois motores de passo via computador ou manualmente.

Acionamento via computador

No acionamento via computador ao ser digitado um comando que aciona o(s) motor(es) o microcomputador gera sinais de pulso (*clock*) e de direção para o circuito de controle permitindo o giro do(s) mesmo(s) no sentido horário ou anti-horário.

Do ponto de vista pedagógico, o uso do módulo controlador de motor de passo permitia, controle de dispositivos eletromecânicos via microcomputador, possibilitando a reprodução de forma concomitante no chão ou no papel dos desenhos produzidos na tela no ambiente Logo gráfico.

Modo manual

Esse modo permitia o posicionamento da caneta no chão ou no papel, independentemente do microcomputador, no local a partir do qual o desenho seria reproduzido.

Do ponto de vista pedagógico, esse modo propiciava mais liberdade para o usuário na escolha do local onde o desenho deveria ser realizado, tanto no papel quanto no chão.

Os dois modos eram utilizados para controlar uma Tartaruga Mecânica ou Traçador Gráfico (Plotter).

A **Tartaruga Mecânica** como ilustra a Figura 2.15, constituía-se de dois motores de passo e um solenóide, montados sobre um carro com o formato de uma pequena tartaruga. Com a Tartaruga Mecânica a atividade de programação era a reprodução simultânea, no solo, dos desenhos produzidos pela Tartaruga de tela no ambiente gráfico do SLogoW.

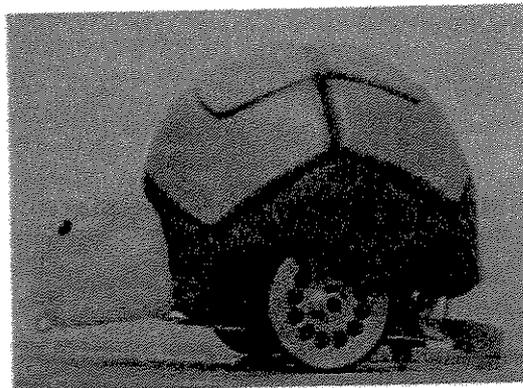


Figura 2.15 – Tartaruga Mecânica de Solo

O **Traçador Gráfico** como ilustra a Figura 2.16, constituía-se de dois motores de passo e de um solenóide, montados sobre uma estrutura mecânica baseada em um sistema de correias dentadas ou roldanas e cabos de aço ou nylon. Com o Traçador Gráfico a atividade de programação dos alunos era a reprodução simultânea, no papel dos desenhos produzidos pela Tartaruga de tela no ambiente gráfico do SLogoW. Isso permitia desenvolver habilidades na atividade de programar utilizando o computador para interagir com dispositivos eletromecânicos em tempo real (D'Abreu, 1994).

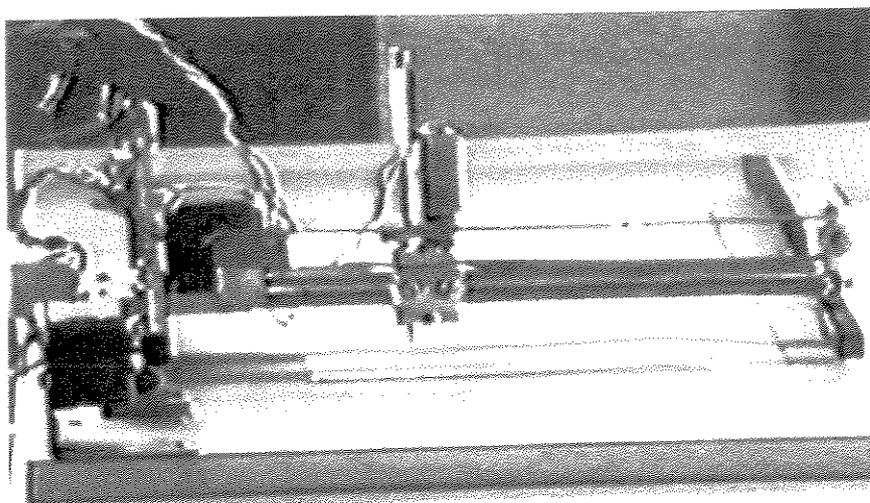


Figura 2.16 - Traçador Gráfico Educacional

Módulo controlador dos motores de corrente contínua:

Esse módulo constituía-se de circuitos que permitiam o acionamento de três motores de corrente contínua, com variação da potência entregue a estes de maneira graduada em oito valores (potência variando de nível “0” a nível “7”). Cada motor podia ser acionado nos dois sentidos (horário e anti-horário), ou dois motores girando em um sentido único. Esse módulo era utilizado para controlar motores LEGO ou qualquer outro tipo de motores de corrente contínua.

Do ponto de vista pedagógico, na utilização desse módulo podia-se criar atividades que propiciavam o manuseio de forma concreta de diferentes conceitos científicos presentes no processo de montagem automação e controle de dispositivos mecatrônicos utilizando motores de corrente contínua.

Módulo acionador de lâmpadas e solenóides

Esse módulo possibilitava o acionamento de quatro conjuntos de contatos, dois de 12 Volts e dois de 5 Volts, empregados para a ativação de lâmpadas e solenóides que eram utilizados na Tartaruga Mecânica e/ou no Traçador Gráfico.

Módulo controlador de sensores

Esse módulo permitia o controle de sensores de toque e de sensores de luz. O módulo possibilitava a leitura do estado de até quatro sensores de luz (iluminados ou não) e de dois sensores de toque (pressionados ou não). O módulo proporcionava a utilização desses dois tipos de sensores nos dispositivos montados utilizando tijolos.

Módulo controlador do conversor analógico-digital:

O conversor analógico-digital consistia basicamente de um chip ADC0809 (um conversor A/D), que possibilitava a leitura de até oito canais de entrada de sinal analógico independentes.

A conversão analógica-digital, permitia o monitoramento de fenômenos físicos empregando diversos tipos de transdutores. Do ponto de vista pedagógico, este módulo propiciava a utilização de um laboratório de ciências baseado em sensores. Os sensores captavam diferentes fenômenos físicos no mundo real e o aluno criava representações no computador para esses fenômenos.

Alguns transdutores utilizados eram:

- Células Fotoelétricas;
- Fotodiodos;
- Sensores de Pressão;
- Sensores de Temperatura.

Módulo controlador da leitora de cartões perfurados:

Esse módulo foi denominado de Slot Machine e dividido em duas partes: um circuito de controle, montado na SuperI, e um cartão decodificador externo responsável pelo acionamento de até 64 slots de leitura ótica.

O circuito montado na SuperI era responsável pelo controle do fluxo de informações entre os cartões óticos e o microcomputador.

O cartão decodificador externo desempenhava a função de um banco multiplexador/demultiplexador de oito bits bidirecional. Cada slot lia um cartão perfurado, utilizando para isso, oito bits leitura.

O Slot Machine tinha como finalidade substituir o teclado do computador por um bloco, onde eram colocados cartões correspondentes a comandos a serem executados. Do ponto de vista pedagógico, isso permitia que crianças não alfabetizadas ou pessoas impossibilitadas de utilizar o teclado pudessem programar. O conjunto de cartões inseridos num bloco contendo vários slot tinha a seqüência lógica de comandos que o computador executava passo a passo. À medida que cada cartão era lido a Tartaruga de tela e/ou Mecânica se movimentava em função da informação contida no cartão. A cada slot era associada também uma luz (um *Led*) que indicava que o mesmo estava sendo executado. Estava previsto também a utilização de vários conjuntos de blocos, de cores diferentes. Isso permitia, por exemplo, que a leitura de um dos cartões num bloco (azul) mudasse o fluxo de leitura para o bloco (amarelo) e os cartões do bloco amarelo passariam a ser lidos e executados. A proposta era “simular” a idéia de subprocedimentos do Logo, ou de execução de programas contendo várias macros onde uma macro principal chamaria as demais.

A utilização da SuperI e demais dispositivos, sobretudo para os ambientes implementados com base no Logo no NIED, apresentou/apresenta um grande empecilho que é a impossibilidade de produzi-los em grande quantidade (em escala) no padrão comercial. Esse empecilho tem sido contornado, utilizando os Kits LEGO que é um material de padrão comercial disponível em

diversos países, inclusive no Brasil. Embora os Kits educacionais LEGO não permitam controlar um conversor AD ou um motor de passo, por exemplo, esses Kits permitem montar uma infinidade de dispositivos mecatrônicos com condições de funcionamento muito próximo de dispositivos reais.

2.8.6 Ambiente SIROS (Sistemas Robóticos com SuperLogo)

Esse ambiente é um projeto de pesquisa que está sendo desenvolvido no NIED/UNICAMP objetivando implementar no software SuperLogo recursos que possibilitam ao usuário controlar dispositivos mecânicos automatizados, no modo presencial e a distância utilizando a rede Internet. O ambiente permite controlar dispositivos mecânicos automatizados numa faixa de abrangência que inclui desde kits disponíveis comercialmente até dispositivos desenvolvidos com materiais alternativos. No anexo 3 (**SIROS e RCX**) encontra-se descrito de forma técnica o ambiente siros, e programas elaborados no SuperLogo para RCX.

Utilizando o ambiente no Modo Presencial

Nesse modo o SIROS é utilizado acrescido das demais funções do SuperLogo e interfaces de software/hardware para controlar dispositivos mecânicos automatizados LEGO e/ou outros dispositivos também.

Utilizando o ambiente no Modo remoto (à distância)

Nesse modo o SIROS é utilizado para operação e monitoramento remoto de dispositivos mecânicos automatizados, por intermédio da rede Internet. Nesse modo de utilização, o cliente usuário pode digitar comandos que acionam o dispositivo instalado fisicamente no nosso laboratório no NIED/UNICAMP. Ou ainda digitar uma seqüência de comandos que permite ao dispositivo executar uma determinada tarefa. O usuário acompanha os movimentos do dispositivo na tela do seu monitor de vídeo.

Procedimentos SuperLogo para a interface de Software – Para controlar os recursos disponíveis no RCX de forma transparente e mantendo a estética Logo onde não existe nada pronto, são fornecidas sim ferramentas para o aluno construir conhecimentos a partir do processo de manipulação/utilização dessas ferramentas. Foram desenvolvidos procedimentos que enviam e recebem instruções da Interface de Software. Esses procedimentos são conjuntos de instruções para acionamento de motores, seleção e aquisição de dados por sensores.



Figura 2.17- Tela inicial do ambiente SIROS

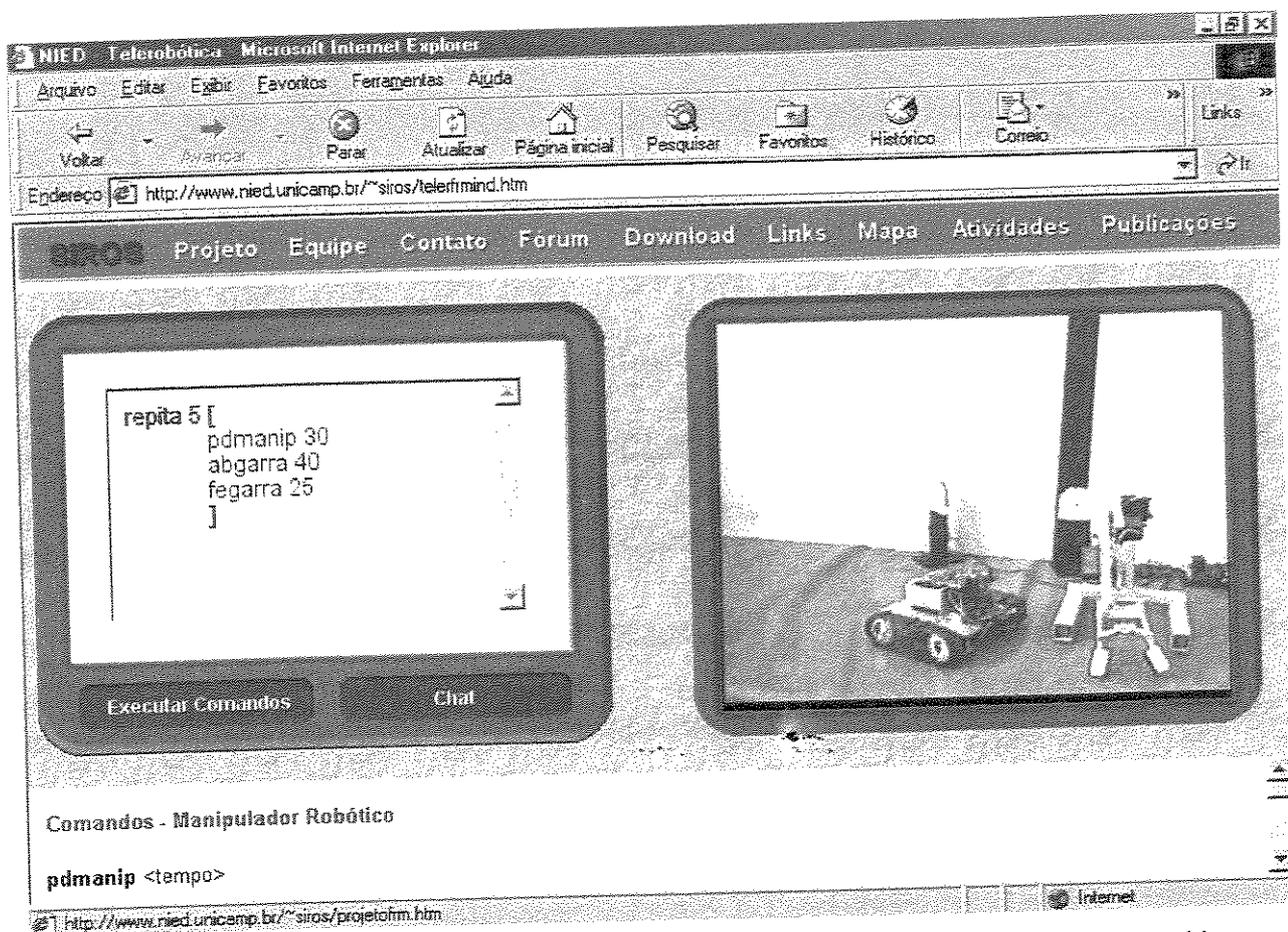


Figura 2.18 - Telas de programação visualização de movimentos e de “Chat” do ambiente

Com o SIROS está se tornando possível aos pesquisadores do NIED implementar a Telerobótica cuja descrição será apresentada de forma sucinta a seguir.

2.8.7 Ambiente de Telerobótica

O Ambiente de Telerobótica permite que um programa em SuperLogo controle dispositivos mecânicos automatizados que podem ser classificados em duas categorias, os de morfologia fixa¹⁷ e os de morfologia variável¹⁸ (D'Abreu & Chella, 2001).

¹⁷ Do ponto de vista de automação e controle definiu-se dispositivos de morfologia fixa como sendo aqueles dispositivos que, uma vez projetados e construídos, a sua Parte Operativa não se altera, porém a Parte Comando, varia em função da tarefa que executam em um determinado momento. Alguns exemplos destes dispositivos são: Manipulador Robótico, Tartaruga Mecânica, Robô Móvel, Traçador Gráfico, etc..

¹⁸ Do ponto de vista de automação e controle definiu-se dispositivos de morfologia variável como sendo dispositivos cuja Parte Operativa é criada e alterada rapidamente e a Parte Comando varia em função da tarefa que estes executam. Entre a execução de uma tarefa e outra o mecanismo do dispositivo pode ser modificado. Fazem parte de dispositivos com essa características: montagens feitas utilizando tijolos LEGO em geral.

Esse ambiente utiliza-se dos recursos disponíveis pela rede Internet implementados em torno de uma interface gráfica que segue o padrão WEB. Também faz parte desse ambiente um conjunto de interfaces de hardware e software. As interfaces de hardware atuando entre o computador e os dispositivos robóticos são responsáveis por receber as instruções oriundas do computador e converte-las em sinais elétricos capazes de atuar sobre motores e sensores. Os recursos de software permitem a comunicação entre o usuário remoto e o usuário que está no local onde o dispositivo está instalado fisicamente permitindo controle do dispositivo, a partir de qualquer microcomputador conectado a Internet, sem que seja necessária instalação de software adicional ou *plug-in* e, finalmente, a apresentação da interface gráfica. Constituem esses recursos os seguintes itens:

- **Servidor WEB** - responsável por disponibilizar as páginas com a interface gráfica e tratar os comandos enviados pelo usuário para o controle dos dispositivos robóticos.
- **Software de Vídeo** – responsável pela captura, por meio de uma câmera, das imagens dos dispositivos robóticos que estão sendo controlados transmitindo-as pela rede permitindo a visualização remota.
- **Servidor de Chat** - propicia a comunicação por meio de texto entre o(s) usuário(s) remoto(s) e o(s) situados no local onde estão instalados os dispositivos robóticos.
- **Software SuperLogo** – executa os programas e comandos enviados pelo usuário remoto. Para isso foi acrescido a este software primitivas para controle do LEGO RCX, Servomotores e Detectores de Infravermelho.
- **Interface de Software** - O controle do LEGO RCX pelo micro PC ocorre por meio do envio de instruções à porta serial onde a torre de infravermelho esta conectada seguindo as especificações do protocolo de comunicação. Para que esta tarefa seja possível a LEGO disponibiliza uma biblioteca para programação que segue a tecnologia

Activex¹⁹. Como o SuperLogo não oferece suporte a esta tecnologia foi utilizado o ambiente de programação Visual Basic da Microsoft para o desenvolvimento de um software que atuasse recebendo e enviando mensagens dos procedimentos SuperLogo com instruções que serão executadas no LEGO RCX.

2.9 Estado da Arte

No que diz respeito ao estado da arte na implementação/utilização de dispositivos mecânicos automatizados no contexto educacional verifica-se que existem algumas experiências similares as que vêm sendo desenvolvidas no Laboratório de Automação e Robótica – LAR/UNICAMP e no Núcleo de Informática Aplicada à Educação – NIED/UNICAMP em atividades nas áreas de ensino, pesquisa e extensão. A seguir, será apresentado alguns *sites* que descrevem esses ambientes.

Projeto do departamento da Ciências de Computação da Universidade de Aarhus²⁰

É uma dissertação de mestrado, escrita por (Bjerre *et alli*, 1999) alunos do departamento de Ciências de os estudantes na Universidade de Aarhus. O trabalho desses pesquisadores se consistiu em desenvolver um cachorro-robô, usando como componente principal o LEGO *Mindstorms*.

Universidade de Lancaster na Inglaterra²¹

O COM120 é uma disciplina do curso de ciências da Universidade de Lancaster. Esta disciplina é dividida em 2 módulos: *Internet e Application Design*, sendo que a ênfase da última é a programação. Neste módulo, eles utilizam 3 recursos diferentes, ROBO LAB (Kit de programação para o LEGO), QBasic e Visual Basic. O LEGO e ROBO LAB são utilizados para

¹⁹ A tecnologia Activex é um recurso de programação que permite ao programador criar uma janela com o seu código associado definindo um comportamento (Cantù, 1998). Windows Activex permite criar programas que funcionam como *plug-in* em outros tipos de aplicações. Por exemplo, criar uma figura em um outro aplicativo e usá-la como *plug-in* num processador de texto.

²⁰ <http://www.daimi.au.dk>

²¹ <http://info.comp.lancs.ac.uk/com120/introduction.htm>

ministrar aulas de programação. As atividades são desenvolvidas, por sessões, com aulas práticas no laboratório. Cada sessão tem algumas tarefas a serem cumpridas e os alunos trabalham em grupos de 5. São 30 alunos divididos em 6 grupos de 5 e têm 6 robôs.

Laboratório de Estudos Cognitivos da Universidade Federal do Rio Grande do Sul (LEC-UFRGS)²²

As atividades do LEC na área de robótica educacionais têm como enfoque a possibilidade de se criar ambientes de aprendizagem onde se possa refletir sobre a própria aprendizagem (Petry, 1996). Neste sentido, um dos objetivos do ambiente LOGO, por exemplo, é o de favorecer situações onde ocorram tomadas de consciência sobre os próprios processos cognitivos. O trabalho do LEC aborda outras áreas de conhecimento tais como: Artes Cibernética e Tecnologia de Controle, *Design*, Física, Matemática, Motricidade, Problemas Sistêmicos e Vida Artificial.

Sistema LEGO para Aquisição de Dados e Geração de Protótipos²³

É um projeto da universidade de Tufts que tem por objetivo introduzir o estudo da engenharia para crianças de jardim de infância até o segundo ano da faculdade. Eles desenvolveram uma série de *drives* (software) que permitem a comunicação com o sistema LEGO Dacta usando o programa LabVIEW, desenvolvido pela *National Instruments*. LabVIEW é uma linguagem de programação gráfica que pode ser usado por estudantes de jardim de infância até a universidade. Uma das idéias do projeto é mudar a forma como se ensinam ciências na escola primária.

²²<http://www.psico.ufrgs.br/lec/repositorio/robot/>

²³ <http://ldaps.ivv.nasa.gov>

*Universal robots: the history and Workings of robotics*²⁴

É um *site* criado com a idéia de introduzir ao visitante da página a ciência inerente ao projeto de teleoperação dos robôs. Neste *site*, existem robôs que após a interação com a exibição, dos seus movimentos o visitante poderá:

- 1) definir um robô como uma máquina que recolhe a informação sobre o meio ambiente;
- 2) reconhecer vantagens e limitações dos robôs quando comparados com seres humanos.
- 3) explorar o uso de robôs como máquinas que pensam, e atuam em tarefas de manufatura, na pesquisa e exploração de solo, em ajustes de equipamentos nas viagens espaciais, dentre outras;
- 4) entender como a conexão com a tecnologia pode criar um contexto para as pessoas se interessarem em Ciências e na Matemática num espaço de trabalho em que o computador, dispositivos robóticos, e a automação são comuns e essenciais.

*RemoteBot.net*²⁵

RemoteBot.net é da propriedade de *K-Team* e de *DreamLab*, que são projetos de pesquisa desenvolvidos no instituto de tecnologia federal suíço em Lausana. O *site* disponibiliza robôs do tipo Khepera, que permitem ao visitante criar rápida e facilmente seu próprio mundo "non-virtual" "on line". O *site* hospeda um museu de robótica que fica no ar 24 horas.

²⁴ <http://www.thetech.org/robotics/index.html>

²⁵ <http://remotebot.k-team.com/>

*Austrália Telerobot on the Web*²⁶

É um *site* que permite que diversas pessoas em locais diferentes do planeta possam trabalhar de forma cooperativa no controle de um robô. Quando alguma pessoa assume o controle do robô a outra pessoa pode auxiliá-la fazendo medições que poderão ser utilizadas pela primeira pessoa no controle do robô. A interação entre as pessoas é feita via *Chat* no bate-papo do *Windows*.

*TEC de Monterrey Campus Cuernavaca*²⁷

É um *site* do departamento de Engenharia em Mecatrônica do Instituto Tecnológico de Monterrey Campus Cuernavaca em México. O *site* apresenta as vantagens que um curso de Engenharia Mecatrônica oferece como sendo a fusão entre a eletrônica, computação, mecânica para processos de manufatura industrial e robótica.

Universidade Federal de Santa Catarina. Laboratório de Experimentação Remota²⁸

O Laboratório de Experimentação Remota é uma aplicação educacional que permite a estudantes buscar informações no mundo real a partir de um computador remoto, introduzindo o conceito, Experimentação Remota. O Laboratório de Experimentação Remota é um sistema combinado de um computador (Servidor) e dispositivos externos.

O Laboratório tem como uma das suas metas proporcionar ao usuário externo contato com experimentação remota, permitindo rodar um programa para o microcontrolador 8051, complementando assim o ensino de microcontroladores. Com isso, estudantes de microcontroladores podem fazer experiências práticas mesmo sem dispor do componente.

²⁶ <http://telerobot.mech.uwa.edu.au/>

²⁷ <http://www.mor.iteesm.mx/~dptolec/mecatronica.html>

²⁸ <http://www.inf.ufsc.br/~jbosco/frame3.htm>

2.10 Considerações sobre o Capítulo 2

Este capítulo discutiu a automação no contexto educacional nos âmbitos de ambientes de aprendizagem envolvendo o uso de dispositivos robóticos como ferramenta auxiliar de ensino num cenário cujo enfoque foi a informática na educação numa abordagem fortemente utilizada por professores de ensino fundamental e médio denominada de robótica pedagógica.

O capítulo descreveu o ambiente LEGO-Logo como sendo um dos ambientes de robótica pedagógica mais utilizado e também outros tipos de ambientes. Foram descritos dois contextos de utilização do ambiente LEGO-Logo. Um contexto que se pode caracterizar como situação formal de utilização que é o uso na escola, e outro que é a situação não formal que é o uso numa fábrica. Discutiu-se os ganhos em termos de aprendizagem que qualquer uma dessas situações propicia. Apresentou-se neste capítulo o SuperLogo como sendo a versão do Logo acrescido de comandos que permitem o controle de dispositivos mecânicos automatizados no modo local e no modo remoto utilizando recursos da rede Internet, o que tem permitido aos pesquisadores do NIED desenvolver atividades de telerobótica onde qualquer usuário da rede Internet, a partir de um endereço de *site* pode, em qualquer lugar do mundo, controlar e visualizar na tela do seu computador os movimentos dos dispositivos sob o seu comando mantendo contato síncrono via *chat* com os pesquisadores no NIED. Ainda do ponto de vista de softwares utilizados para controle de dispositivos mecatrônicos, o capítulo discutiu também uma série de outros softwares de uso comercial que têm a finalidade de se implementar ambientes de aprendizagem e também a implementação/construção de dispositivos mecatrônicos baseados em outros tipos de materiais diferentes de LEGO.

No que diz respeito ao estado da arte da automação no contexto educacional, o capítulo apresentou alguns ambientes de robótica e telerobótica que foram e/ou estão sendo implementados em diversas instituições de ensino e pesquisa nos âmbitos nacional e internacional, dando destaque ao ambiente SIROS que é um projeto de pesquisa do NIED/UNICAMP. A implementação dos ambientes acima mencionados envolve, dentre outras tarefas, criar condições que permitam que os alunos desenvolvam atividades de montagem automação e controle de dispositivos mecatrônicos que possuem, como componentes principais sensores e atuadores.

No próximo capítulo será discutido sistemas automatizados em termos de sua Parte Operativa e Parte Comando abordando a utilização do Grafcet como sendo uma ferramenta para especificação de projetos na área de automação. Projetos esses que podem ser implementados em ambientes do tipo LEGO-Logo.

Capítulo 3

Sistemas Automatizados com Ênfase no Uso Educacional

Este capítulo discute sistemas automatizados sob dois enfoques. Um trata de sistemas automatizados discretos com ênfase na utilização de Grafset como ferramenta utilizada no processo de especificação desses sistemas e outro que aborda a utilização de sistemas contínuos com base em função de transferência de sistemas em malha aberta e malha fechada. O capítulo aborda também sensores e atuadores como os principais elementos operativos que fazem parte da implementação de um dispositivo robótico/mecatrônico. Estes elementos operativos serão discutidos como transdutores que mudam o seu comportamento sob a ação de uma grandeza física ou que fazem atuar a energia mecânica sobre uma máquina, para realizar um determinado trabalho. Serão apresentados alguns sensores e atuadores de uso comercial e também os de uso educacional, destacando os comercializados pela LEGO.

3.1 Sistemas Automatizados

Os sistemas automatizados são em geral compostos de duas partes: Parte Operativa (PO) e Parte Comando (PC). A parte operativa corresponde ao processo físico a automatizar, que opera sobre a matéria prima e o produto. Ela é constituída pelos atuadores que realizam operações, agindo sobre os materiais para os transformar, transportar ou armazenar. São componentes constituintes da parte operativa: válvulas, pistões, bombas, motores, soldadores, dentre outros (Inazumi, 2001). A parte comando coordena as ações da parte operativa. Para cada máquina ou processo é necessário escolher, dentre as diferentes tecnologias de comando disponíveis, as mais

adequadas e as que melhor se adaptam ao processo e a matéria prima em questão. Dentre as tecnologias que podem ser usadas para este fim pode-se citar comandos pneumáticos, hidráulicos, relés controladores lógicos programáveis, etc..A Figura 3.1 apresenta a parte operativa e a parte comando de um sistema automatizado destacando o uso de atuadores e sensores que serão discutidos a seguir.

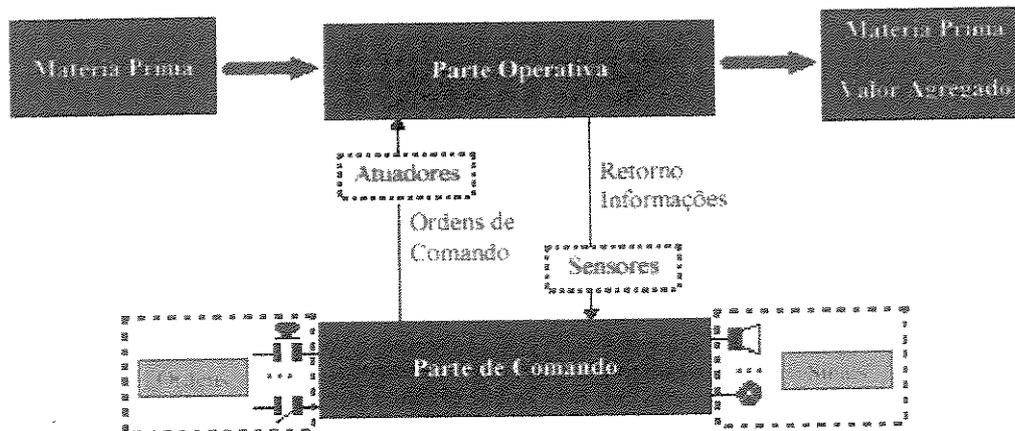


Figura 3.1 – Parte Operativa e Parte Comando de um sistema automatizado.

Do ponto de vista da automação o sistema da Figura 3.1, constituído de Parte Operativa e Parte Comando é considerado um sistema estruturado cuja estrutura permite um diálogo proveitoso entre o futuro utilizador deste sistema e o técnico de automatismo responsável pela parte de comando. Portanto é importante que exista uma descrição precisa do funcionamento da parte comando, por uma aproximação progressiva das funções a realizar, até à sua materialização (Rosário at all, 1997). A ferramenta utilizada para se fazer essa descrição é o Grafcet.

3.2 Linguagens de Programação para Controlador Lógico Programável – CLP

Numa indústria as atividades realizadas podem ser muito diversificadas implicando também em utilização de diferentes tipos de dispositivos, os que trabalham com sinais analógicos, os que trabalham com sinais digitais, os que realizam cálculos e operações complexas, dentre outros. Neste contexto, para resolver um problema de automação, a equipe de especialista em automação deve fazer uma análise do problema e, mediante uns símbolos

próprios e experiência na área, apresentar esquemas correspondentes à tecnologia que será utilizada. É nesse tipo de situação que um CLP pode ser utilizado e o Grafcet é empregado como sendo a ferramenta útil e eficaz para facilitar na representação e ajudar na solução de um sistema automatizado composto de parte operativa e parte comando, conforme descrito no item 3.1.

A parte comando de todo sistema automatizado deve ser bem especificada e documentada, de modo a evitar ambigüidades e dúvidas durante a sua respectiva implementação e/ou manutenção. No caso de utilização, por exemplo, de um Controlador Lógico Programável – CLP a norma IEC 61131-3 padronizou métodos de especificação para os diferentes fabricantes deste dispositivo definindo 5 linguagens de programação:

- Lista de Instruções (IL : "Instruction List")
- Texto Estruturado (ST : "Structured Text")
- Diagramas Ladder (LD : "Ladder Diagram")
- Diagrama de Blocos de Função (FDB : "Function Block Diagram")

Estas linguagens de programação são utilizadas na programação de CLP's existentes atualmente no mercado (Inazumi, 2001).

A seguir será descrito, de forma sucinta, o Grafcet e no capítulo 4 apresentado um exemplo de projeto onde se utilizou o Grafcet funcional que foi o aspecto desta ferramenta que mais se adequou aos projetos desenvolvidos, pelos alunos, durante a tese. Entretanto, uma descrição mais detalhada sobre esta ferramenta encontra-se no anexo 8 (GRAF CET).

3.3 Modelagem de Sistemas Automatizados

Na área de Engenharia Mecânica a modelagem de sistemas automatizados faz parte do projeto de automação. Conforme já se descreveu no capítulo 1, para a elaboração de projeto de automação, a especificação é uma etapa imprescindível do processo. Devido a constante complexidade das aplicações de controle industrial, faz-se necessário o uso de metodologias que permitam especificar de forma clara, concisa e não ambígua todos os requisitos funcionais dessas

aplicações. Se essa aplicação for à concepção, estruturação e a programação de um dispositivo mecatrônico isso é tido como garantia de que na operação desse dispositivo não ocorrerão falhas. Pois, durante a operação não é possível realizar a experimentação. Espera-se que o dispositivo funcione, para que seja realizada uma determinada operação com sucesso. É nesse contexto que o Grafcet é utilizado.

3.3.1 O Que é Grafcet

Na atualidade, as constantes revoluções tecnológicas pelas as quais as indústrias têm passado está fazendo com que elas sejam obrigadas a adotar de forma constante novas técnicas de automatização de máquinas e processos. Grafcet é baseada na norma da *International Electrotechnical Commission* – IEC 60848. Ele surgiu da necessidade de se dispor de um método de descrição de processos seqüenciais, de forma eficaz, simples e de fácil interpretação por técnicos de diferentes áreas (Martínez & Estragués, 1999). Ele é o gráfico funcional de controle de etapas e transições. É um sistema gráfico de representação de controle seqüencial mediante a sucessão alternada de etapas e transições. É uma variante do diagrama de transições de estados. É uma simplificação sistematizada de redes de Petri. Ou ainda, uma linguagem que se baseia na descrição funcional das partes em que se constituem um sistema automatizado, envolvendo a especificação de todas as seqüências de operações dos elementos que forma este sistema (Rosário at all., 1997).

O Grafcet, em complemento com os princípios de álgebra de Boole, permite algo mais do que a descrição e interpretação gráfica de processos. Ele é uma ferramenta de desenho flexível que, incorporada a linguagens de programação, facilita o desenvolvimento de muitos automatismos.

Grafcet foi criado na França em 1977 pela AFCET (Associação Francesa para a Cibernética Econômica e Técnica) e pela ADEPA (Associação para o Desenvolvimento de Produção Automatizada). Um dos editores de Grafcet utilizado nos meios acadêmicos é o SAGITAL para o ambiente windows no contexto da norma IEC 60848 (Toussaint, 2000).

3.3.2 Grafcet de nível 1 – Descrição Funcional

Trabalha com as especificações funcionais de automação, independente da tecnologia a ser empregada. Descreve as ações que devem ser realizadas e os elementos de controle que intervirão sem especificar os elementos concretos que serão utilizados. Uma vez o sistema terminado e em funcionamento, uma revisão em grupo a partir do Grafcet de nível 1, permitirá analisar as especificações para a melhoria do sistema (Castillo, 2001).

3.3.3 Grafcet de nível 2 – Descrição Tecnológica

- Indica todas as especificações dos órgãos operativos;
- Detalha os elementos tecnológicos que intervirão.

3.3.4 Grafcet de nível 3 – Descrição Operativa

Especifica todos os elementos em termos de entradas e saídas, assim como as marcas e o relês internos que serão utilizados.

3.4 Características do Grafcet

- Possuir normas de decisão bastante simples;
- Ser adequado para montagem de estruturas seqüenciais;
- Ter uma disciplina de programação;
- Ser muito estruturado;
- Existir CLP's que incorporam pacotes de programação em Grafcet;
- Ser prático para sistemas puramente combinacionais.

O Grafcet em complemento com os princípios de álgebra de Boole permite algo mais do que a descrição e interpretação gráfica de processos. Ele é uma ferramenta de desenho potente e flexível que, incorporada com linguagens e programação, facilita o desenvolvimento de muitos automatismos.

3.5 Elementos de um Grafcet

Etapas e ações associadas, Transições receptividades e ligações.

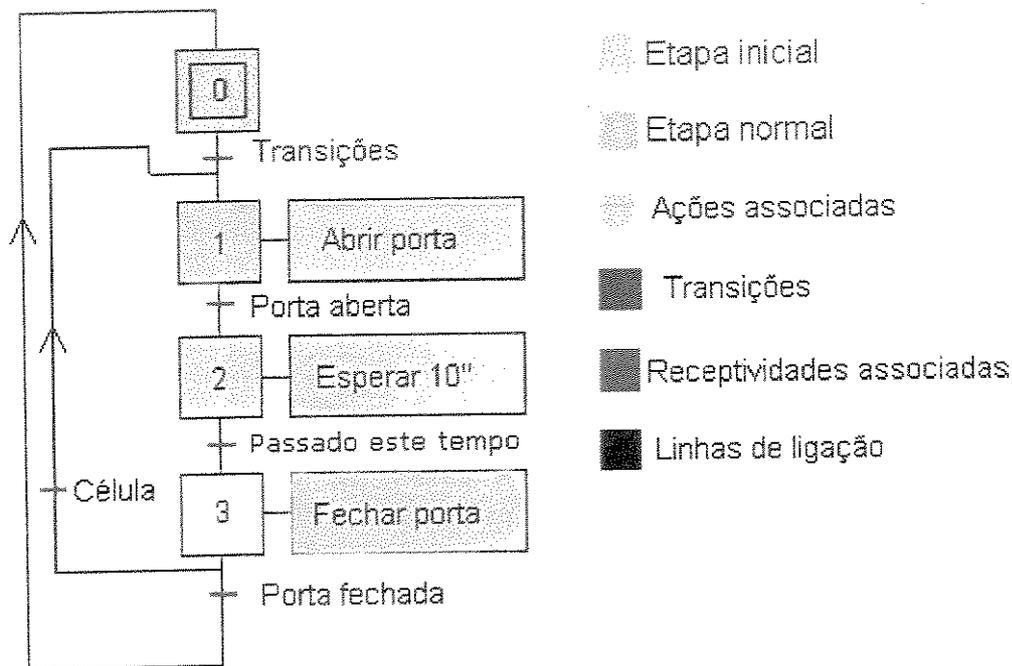


Figura 3.2 – Representação esquemática dos elementos de um Grafcet

3.6 Descrição de Elementos Constituintes de Sistemas Automatizados

Um sistema automatizado pode ser dividido em subsistemas integrados através de modelagem por Grafcet. Nesse contexto, um subsistema típico é o de um dispositivo robótico,

que numa analogia com o corpo humano, a teoria de controle estuda o funcionamento do “coração” desse dispositivo. Controle clássico, também conhecido como “*feedback*”²⁹ control”, consiste de um conjunto de técnicas que permitem a um sistema alcançar um determinado estado desejado, e permanecer neste estado mesmo que haja variações nas condições externas, (Martin, 2001). A Figura 3.3 mostra um diagrama “canônico” da representação de um sistema clássico de realimentação “*feedback*”. Esse diagrama representa um sistema com retroação negativa (Dorf & Bishop, 2001).

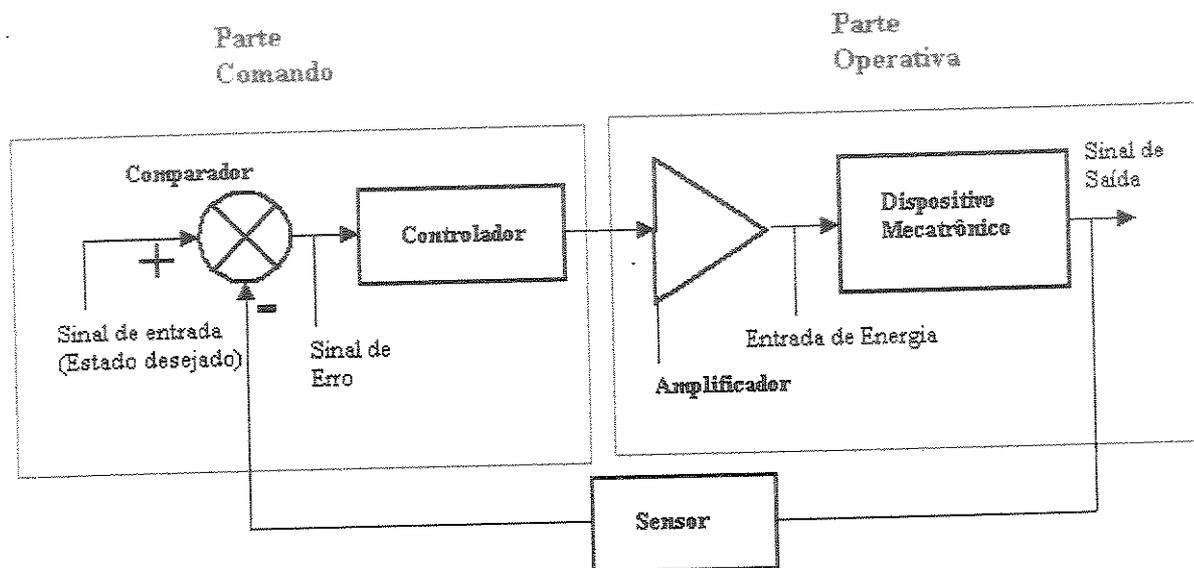


Figura 3.3 - Sistema clássico de realimentação

Na Figura 3.3 um sinal de entrada que representa o estado desejado, é combinado com o sinal de *feedback* representando o estado atual do sistema. O sinal de erro corresponde a variações no estado atual do sistema em relação ao sinal de entrada desejado. Este sinal de erro é uma informação que alimenta um controlador, que por sua vez, alimenta um amplificador. A saída do amplificador, por sua vez, transforma a informação em energia de entrada do dispositivo mecatrônico a ser controlado, o que poderá corresponder a uma ação física como giro de um

²⁹ Feedback é um conceito da cibernética que tem a ver com o estado de auto-organização de um sistema. Existe feedback positivo sempre que um sistema produz efeitos que aumentam este estado.

Existe feedback negativo sempre que um sistema produz efeitos que diminuem este estado.

Um exemplo de feedback negativo é o de um sistema que possui um termostato que controla o aquecimento e a refrigeração do ar em uma sala. Ajustando a temperatura para 70 graus se a sala ficar quente (acima de 70 graus) o termostato liga o ar condicionado, fazendo com que a temperatura volte aos 70 graus (Papert, 1994).

motor, aumento de temperatura, acionamento de uma buzina, etc. Finalmente, o sinal de realimentação, por meio de um sensor, fecha o ciclo jogando o sinal de saída novamente para a entrada. Todos os sistemas realimentados são baseados nestes elementos: sensor, comparador, controlador e sensor novamente.

3.6.1 Sistema em Malha Aberta

Existem alguns sistemas de controle que não têm a realimentação “feedback”. Estes sistemas são denominados de sistemas em malha aberta. Uma das vantagens de sistemas de malha aberta é a simplicidade de controle de velocidade. Isso torna este sistema mais viável economicamente (Kuo, 1982). Uma máquina elétrica de lavar roupa, por exemplo, é um sistema de malha aberta porque tipicamente a duração do tempo de lavagem da máquina é inteiramente determinada pelo julgamento e estimativa do operador humano.

No ambiente LEGO alguns de sistemas de controle que se realiza são de malha aberta (Martin, 1992). Por exemplo, um robô que possui um motor, que gira no sentido horário, e um sensor de toque e se desloca ao longo de uma parede de modo que o sensor permita com que o mesmo perceba o choque contra a parede, é um tipo de dispositivo que usa o controle em malha aberta. Ao chocar contra a parede, o sistema de controle permite que o motor gire no sentido anti-horário, alguns graus, para em seguida continuar seguindo a parede.

3.6.2 Função de Transferência Malha Aberta

Sistemas para os quais a grandeza de saída não produz efeitos sobre a grandeza de excitação são chamados de sistemas de controle a malha aberta conforme apresentado na Figura 3.4 (D’Azzo & Houpis, 1975). Nesses sistemas a ação de controle é independente da saída.

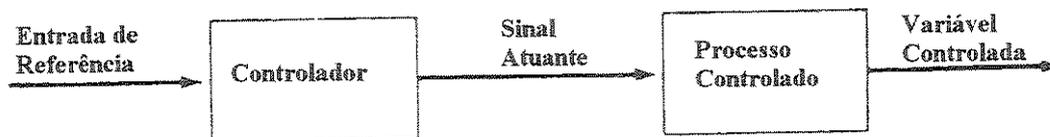


Figura 3.4 - Sistema em malha aberta

3.6.3 Sistema em Malha Fechada

Um sistema de controle em malha fechada, representado na Figura 3.5, é um sistema onde existe uma realimentação da saída à entrada do sistema. Isso permite obter um controle mais preciso na medida em que o sinal controlado é realimentado e comparado com uma entrada de referência e um sinal atuante proporcional à diferença entre a entrada e a saída deve ser enviada através do sistema para corrigir o erro.

3.6.4 Função de transferência em Malha Fechada

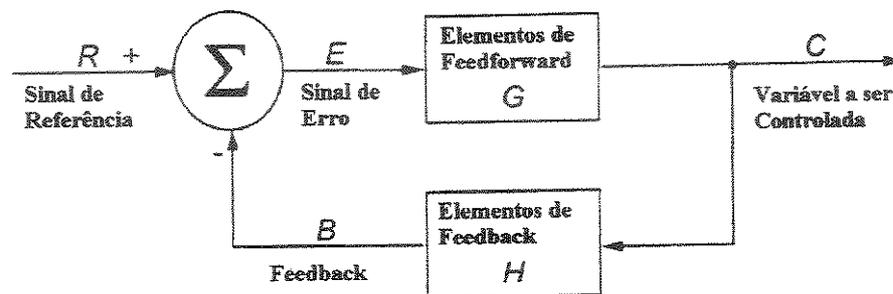


Figura 3.5 - Sistema em malha fechada

Feedforward - significa que o sinal comandado é uma função de alguns parâmetros que são medidos na realimentação.

De forma genérica, conforme Figura 3.5, a referência $R(s)$ é comparada com a realimentação $B(s)$, gerando um sinal de erro $E(s)$. Este sinal de erro é remetido para o elemento de feedforward do controlador, $G(s)$ que geralmente consiste de alguns estágios de controle compostos do dispositivo físico a ser regulado. O resultado é a variável $C(s)$ que é o estado do sistema cujo controle se pretende realizar. Os elementos de feedback do controlador transformam a variável a ser controlada em uma forma consistente com a referência de entrada. Os elementos $G(s)$ e $H(s)$ são muitas vezes referidos como função de transferência de avanço de fase “forward” e de realimentação “feedback”. Estes elementos transformam sinais de entrada, em outras formas de sinais, via equação diferencial, que expressa a dinâmica do sistema.

Para um sinal de referência de entrada na Figura 3.5 tem-se:

$$\begin{aligned} C(s) &= G(s)E(s) \\ B(s) &= H(s)C(s), \text{ e} \\ E(s) &= R(s) - B(s) \end{aligned}$$

A função de transferência em malha fechada pode ser dada como:

$$\begin{aligned} \frac{C(s)}{R(s)} &= \frac{G(s)E(s)}{R(s)} = \frac{G(s)(R(s) - B(s))}{R(s)} = \frac{G(s)}{\frac{R(s) - B(s)}{R(s)}} = \frac{G(s)}{\frac{B(s) + E(s)}{E(s)}} = \frac{G(s)}{1 + \frac{B(s)}{E(s)}} = \frac{G(s)}{1 + \frac{H(s)C(s)}{E(s)}} \\ \frac{C(s)}{R(s)} &= \frac{G(s)}{1 + G(s)H(s)} \end{aligned}$$

O resultado acima sugere que a malha de realimentação pode sempre ser expressada em termos da alimentação. Esta forma da função de transferência, conforme a Figura 3.6 relaciona a variável a ser controlada, $C(s)$, diretamente com a referência de entrada, $R(s)$.

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)} \Rightarrow (1 + G(s)H(s))C(s) = R(s)G(s)$$

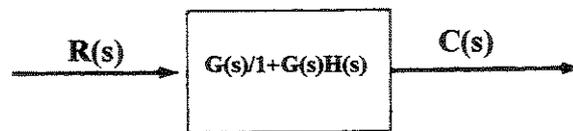


Figura 3.6 – Diagrama de bloco relacionado à entrada com a saída

Caso a equação diferencial acima represente uma equação de movimento com $R(s) = 0$ vem:

$$(1 + G(s)H(s))C(s) = 0$$

Isto é, a forma homogênea da equação do movimento. Isso implica dizer que a equação característica deste sistema é expressada em termos do denominador da função de transferência, $(1 + G(s)H(s))$ e que a estabilidade e a resposta em malha fechada são completamente determinadas por esta equação (D'Azzo & Houpis, 1975).

3.7 Sistemas de Acionamento

Sistemas de acionamento podem ser do tipo hidráulico, elétrico ou pneumático. A título de exemplo descreve-se a seguir um sistema de acionamento elétrico bastante utilizado em automação que consiste de um motor de corrente contínua. No caso, um motor LEGO de 9 Volts, conforme a Figura 3.7.

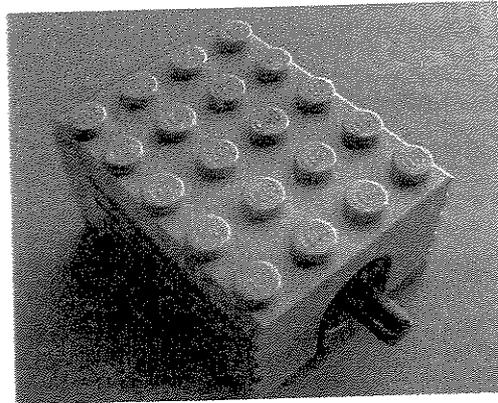


Figura 3.7 – Motor LEGO de 9 Volts

A função de transferência entre a tensão de entrada V e a posição angular θ do eixo de saída de um motor de corrente contínua conforme a Figura 3.8 (Distefano at all, 1972).

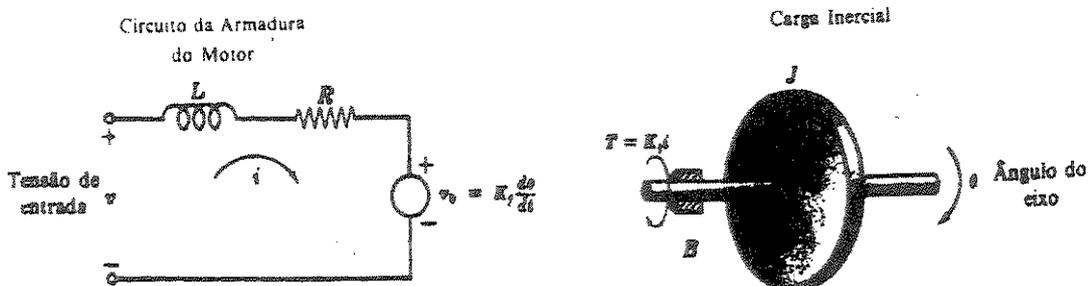


Figura 3.8 – Circuito da armadura e a carga inercial de um motor de corrente contínua.

$$T = k_i i$$

$$v_b = K_f \frac{d\theta}{dt}$$

Onde:

L e R - representam a indutância e a resistência do circuito da armadura

v_b - representa a força contra eletromotriz gerada, que é proporcional à velocidade e dada por

$$\frac{d\theta}{dt}$$

T - representa o torque gerado pelo motor e é proporcional à corrente de armadura.

i - representa a corrente de armadura

J - representa a inércia combinada da carga e da armadura do motor

B - representa o atrito viscoso total atuando sobre o eixo de saída.

K_f - representa a constante de proporcionalidade entre a tensão v_b e a velocidade do eixo $\frac{d\theta}{dt}$.

K_t - representa a constante de proporcionalidade entre a corrente de armadura i e o torque gerado pelo motor.

As equações diferenciais do circuito da armadura e da carga inercial são:

$$Ri + L \frac{di}{dt} = v - k_f \frac{d\theta}{dt}$$

e

$$K_t i = J \frac{d^2\theta}{dt^2} + B \frac{d\theta}{dt}$$

Fazendo a transformada de Laplace de cada equação, ignorando as condições iniciais vem:

$$(R + sL)I = V - K_f s\Theta$$

e

$$K_t I = (Js^2 + Bs)\Theta$$

Resolvendo essas equações simultaneamente para a função de transferência V e θ vem:

$$\frac{\Theta}{V} = \frac{K_t}{(Js^2 + Bs)(Ls + R) + K_t K_f s} = \frac{K_t / JL}{s[s^2 + (B/J + R/L)s + BR/JL + K_t K_f / JL]}$$

3.8 Redutor

Na modelagem do sistema de acionamento de uma junta robótica costuma-se levar em conta diversos blocos dentre eles o bloco redutor. O redutor em uma máquina é utilizado para que, com um certo valor de torque, se obtenha uma melhor performance do motor. Quase todos os mecanismos de controle de juntas robóticas necessitam de baixa velocidade e grande torque. Para estes mecanismos, normalmente, tem que se usar redutores.

a) Transmissões por engrenagens

Genericamente um sistema de redução de velocidade em uma junta robótica pode ser implementado utilizando engrenagens conforme representado na figura 3.9.

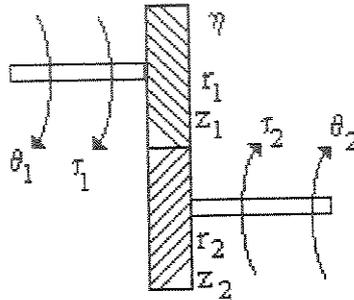


Figura 3.9 – Representação esquemática de um redutor

Do ponto de vista da cinemática, um sistema de redução, constituído de duas engrenagens apresenta as seguinte equações cinemáticas:

$$\eta = \frac{Z_2}{Z_1} \text{ e } \theta_2 = \frac{r_1}{r_2} \theta_1$$

Sendo:

Z - número de dentes das engrenagens

η - razão de transmissão

r - raio da engrenagem.

b) Uso de engrenagens como redutor no trabalho com LEGO

Do ponto de vista prático, no uso de componentes LEGO, na montagem de quase todos os dispositivos mecânicos automatizados que possuem movimento, engrenagens se fazem presentes. Engrenagens são utilizadas como redutor. Existem engrenagens de diferentes números de dentes que podem ser utilizadas com esta finalidade. A Figura 3.10 representa uma montagem combinando uma engrenagem de 8 dentes com uma de 24 dentes num sistema composto de motor redutor e uma carga.

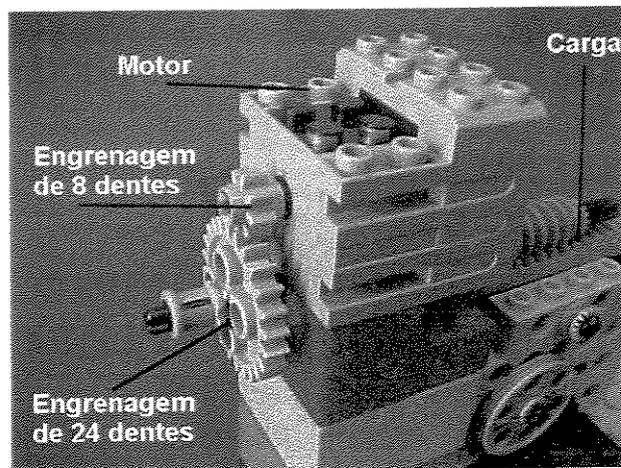


Figura 3.10 - Representação de duas engrenagens LEGO

A engrenagem de 8 dentes tem três vezes menos dentes que a engrenagem de 24 dentes. Portanto, cada três voltas na engrenagem de 8 dentes corresponde a uma volta na engrenagem de 24 dentes. Isto significa dizer que, em termos de movimento angular, que a engrenagem de 8 dentes precisa girar três vezes mais, 1080 graus, para realizar o mesmo percurso que a engrenagem de 24 dentes realiza somente com um giro de 360 graus.

3.9 Utilização de Sensores e Atuadores em Automação

Do ponto de vista de robótica podemos afirmar que um robô que não possui um sensor é simplesmente uma máquina. Isto é, o robô precisa ter sensores para que ele deixe de ser uma

máquina qualquer e passe a ser um dispositivo que percebe o mundo a sua volta e reaja em função de mudança de situações (Martin, 1992).

3.9.1 Sensores do ponto de vista tecnológico

Um sensor pode ser definido como sendo um transdutor que altera a sua característica física interna devido a um fenômeno físico externo pela presença ou não de luz, som, gás, campo elétrico, campo magnético, etc.. O sensor muda o seu comportamento sob a ação de uma grandeza física, podendo fornecer direta ou indiretamente um sinal que indica essa grandeza, convertendo uma quantidade física em um sinal elétrico (Derenzo, 1990). Quando o sensor opera diretamente, convertendo uma forma de energia neutra, é chamado transdutor. O de operação indireta, altera suas propriedades, como a resistência, a capacitância ou a indutância, sob ação de uma grandeza, de forma mais ou menos proporcional. Um sensor analógico produz continuamente uma grande faixa de variação. No caso de uma fotocélula essa faixa vai de $1k\Omega$ para situação de luz brilhante até $300 k\Omega$ para situação de escuro total. Um sensor digital, por sua vez, apresenta somente dois estado 0 ou 1. No caso do sensor de toque isso representa chave não pressionada (resistência de valor infinito) e chave pressionada (resistência zero).

No caso do ambiente LEGO-Logo, na montagem de um dispositivo do tipo uma esteira e um caminhão que transporta peças, a utilização de sensor de luz (como um fototransistor) possibilita a automatização deste processo de transporte. Ou seja, o sensor no formato de um dispositivo emissor e um receptor de luz poderão ser acoplados ao suporte da esteira, numa de suas extremidades, onde está o caminhão para receber as peças. Desse modo, cada peça que passa em frente ao sensor faz com que a luz emitida (pelo emissor) seja recebida de volta (pelo receptor). Um procedimento poderá ser elaborado para contar estas mudanças de estado do sensor, que na verdade, correspondem ao número de peças que passaram pela esteira e foram depositadas no caminhão. Após uma certa contagem, a esteira poderá ser parada automaticamente e o caminhão, no mesmo instante, sair carregando as peças. O caminhão poderá viajar até uma certa distância e retornar para receber novo carregamento.

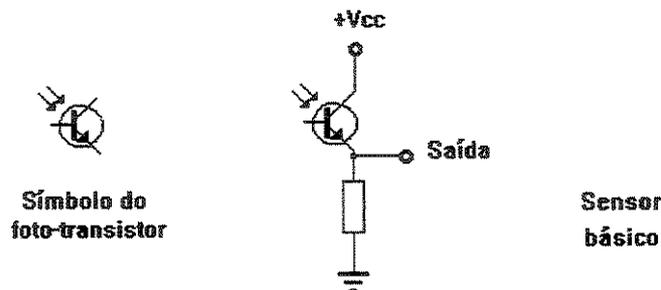
3.10 Sensor usado como transdutor

A principal propriedade de um sensor eletrônico usado, como transdutor, é medir algumas características de fenômenos como som, luz ou pressão convertendo-as em quantidades representáveis eletricamente. Por exemplo:

- Uma fotocélula é um sensor que responde ao estímulo elétrico mudando a sua resistência.
- Um Fototransistor é um sensor que responde ao estímulo elétrico mudando o fluxo de corrente.
- Um sensor Infra-Vermelho responde ao estímulo elétrico mudando a sua voltagem de saída.

3.10.1 Foto transistor

É um sensor baseado no transistor cuja junção coletor-base fica exposta à luz e atua como um foto-diodo. O transistor amplifica a corrente e fornece alguns miliamperes (mA) com alta luminosidade. A velocidade de resposta desse tipo de sensor é menor do que a do foto-diodo. Suas aplicações são as mesmas do foto-diodo, exceto em sistemas de fibra-óptica, que operam em alta frequência.



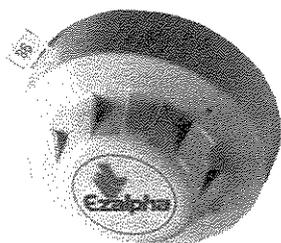
3.10.3 Sensor de reflexão

Consiste em um feixe luminoso que incide sobre um disco com um furo ou marca de cor contrastante, que gira. O sensor recebe o feixe refletido, mas na passagem do furo a reflexão é interrompida (ou no caso de marca de cor clara a reflexão é maior), e é gerado um pulso pelo sensor.

3.10.4 Sensor de interrupção de luz

Consiste também em um disco com um furo. A fonte de luz e o sensor ficam em lados opostos. Na passagem pelo furo, o feixe atinge o sensor, gerando um pulso. A frequência destes pulsos é igual à velocidade, em rotações por segundo (rps), nos dois tipos de sensores. As vantagens destes sensores estão no tamanho (pequeno), custo, durabilidade e a possibilidade de leitura à distância. Estes sensores são usados em sistemas de controle e tacômetros portáteis. Sensor de luz é o sensor mais comumente utilizado na área de robótica. Esse tipo de sensor é usado também em fotometria para controlar a luminosidade, como relés fotoelétricos de iluminação pública e, como sensores indireto de velocidade e posição. Além disso, no este tipo de sensores são muito utilizados como sensores de presença.

3.10.5 Foto Diodo³⁰



Este sensor detecta fumaça. Em condições de ar claro, o foto-diodo não recebe luz do *led* e produz um sinal analógico correspondente. Quando a fumaça entra na câmara, dispersa luz no foto-diodo, aumentando o sinal transmitido ao equipamento de controle.

Figura 3.12 – Foto-diodo comercial

É um sensor baseado num diodo semicondutor cuja junção está exposta à luz. A energia luminosa desloca elétrons para a banda de condução, reduzindo a barreira de potencial pelo

³⁰ http://www.ezalpha.com.br/equipamentos/otico_xp_95.htm

aumento do número de elétrons, que podem circular se aplicada polarização reversa. A corrente nos foto-diodos é da ordem de dezenas de miliampere (mA) com alta luminosidade e resposta rápida. Existem foto-diodos para todas as faixas de comprimentos de onda, do infravermelho ao ultravioleta, dependendo do material. Este tipo de sensor é usado em controle remoto, em sistemas de fibra óptica, detector óptico de fumaça, em leitoras de código de barras, *scanner* (digitalizador de imagens, para computador), canetas ópticas (que permitem escrever na tela do computador), toca-discos CD, fotômetros e como sensor indireto de posição e velocidade, dentre outros.

3.11 Características de sensores

1) Linearidade

É o grau de proporcionalidade entre o sinal gerado e a grandeza física. Quanto maior a linearidade, mais fiel é a resposta do sensor ao estímulo.

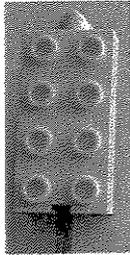
2) Faixa de atuação

É o intervalo de valores da grandeza em que pode ser usado o sensor, sem causar a destruição do mesmo ou imprecisão na leitura.

3.12 Sensores de uso educacional

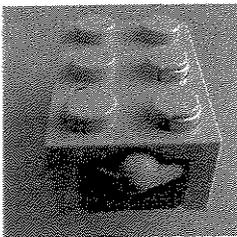
No contexto educacional o ambiente de automação em que trabalhamos com o material LEGO se caracteriza pela utilização basicamente da família de sensores que fazem parte dos kits LEGO conectáveis às entradas do RCX ou a outras interfaces de comunicação com o computador. Fazem parte dessa família sensor de temperatura, toque, luz, rotação e sensor infravermelho. A seguir apresentaremos esses sensores.

3.12.1 Sensores LEGO



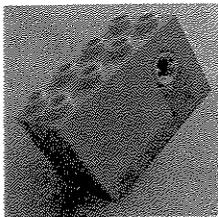
Este é um sensor LEGO, que pode ser conectado a uma das entradas (A, B ou C) do RCX.

Figura 3.13 – Sensor de toque com adaptador para conectar um eixo alongador.



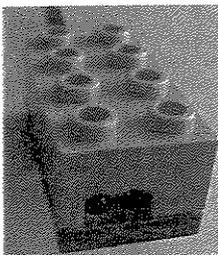
Este é outro modelo de sensor LEGO, que pode ser conectado a uma das entradas (A, B ou C) do RCX.

Figura 3.14 – Sensor de toque sem adaptador para conectar um eixo alongador.



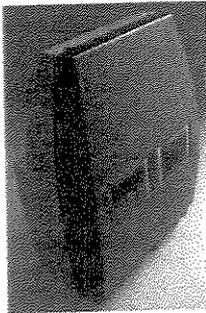
Este é um sensor LEGO, que pode ser conectado a uma das entradas (A, B ou C) do RCX.

Figura 3.15 – Sensor de rotação



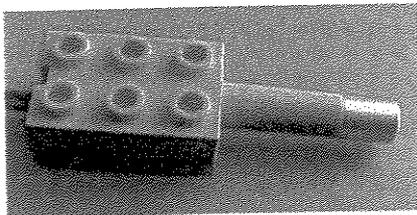
Este é um sensor LEGO, que pode ser conectado a uma das entradas (A, B ou C) do RCX.

Figura 3.16 – Sensor de Luz



Este é um sensor utilizado para se estabelecer à comunicação entre o RCX e o computador. Um sensor infravermelho responde ao estímulo elétrico mudando a sua voltagem de saída.

Figura 3.17 – Sensor Infravermelho



Este é um sensor LEGO, que pode ser conectado a uma das entradas (A, B ou C) do RCX

Figura 3.18 – Sensor de temperatura

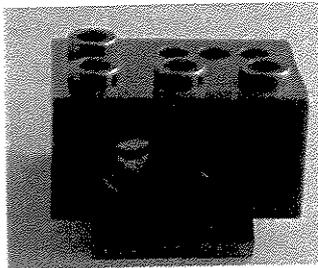


Figura 3.19 – Sensor de luz modelo antigo

Este é um sensor de luz LEGO, que pode ser conectado a uma das entradas (6 ou 7 na interface da Figura 2.7). Este sensor fazia parte dos kits mais antigos da LEGO Dacta que já deixaram de ser comercializados.

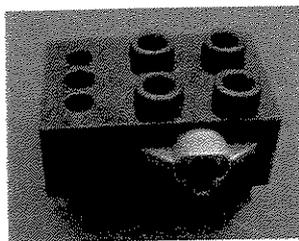


Figura 3.20 – Sensor de toque modelo antigo

Este é um sensor de toque LEGO, que pode ser conectado a uma das entradas (6 ou 7 da interface da Figura 2.7). Este sensor fazia parte dos kits mais antigos da LEGO Dacta que já deixaram de ser comercializados.

Além dos sensores mostrados nas figuras anteriores, existem outros tipos de sensores fabricados pela empresa DCP *microdevelopments* que podem ser utilizados com o RCX e o software Robolab. A marcante característica desse tipo de sensores é que eles são sensores industriais que, acoplados a um adaptador, permite aos alunos a utilização do ambiente LEGO-Robolab para realização de experimentos em níveis de confiabilidade e precisão que se realiza em uma indústria. A seguir apresentamos o adaptador e a alguns desses sensores.

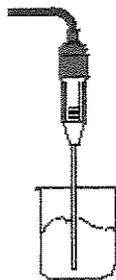
3.13 Adaptador LEGO DCP

Este adaptador permite utilizar micros sensores produzidos pela empresa DCP com o RCX.



Figura 3.21 – Adaptador DCP

As figuras a seguir mostram os tipos de micro-sensores DCp controlados por RCX.

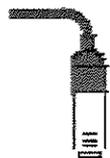


Faixa de operação: -30°C a + 130°C

Resolução: 0,1°C

Precisão com RCX: 2%

Figura 3.22 - Sensor de Temperatura DCP



Faixa de operação: 50 dB a 100dB

Tempo de Resposta: 0,5 segundos

Precisão com RCX: 5%



Figura 3.23 - Sensor de Som DCP

Sensor Rotação



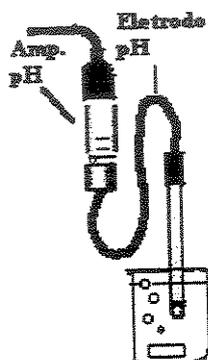
Faixa de operação: 0° a 360°

Torque: 0,25

Carga máxima: 50 gramas

Precisão com RCX: 2%

Figura 3.24 - Sensor de Rotação DCP



Faixa de operação: 0 a 14 pH

Resolução: 0,1°pH

Precisão com RCX: 2%

Figura 3.25 - Sensor de PH DCP



Faixa de operação: 1 a 100% RH

Resolução: 0,1°RH

Precisão com RCX: 2%

Figura 3.26 - Sensor de Umidade Relativa DCP

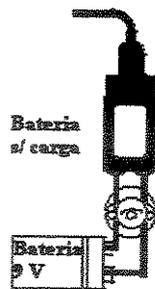
Faixa de Temperatura: -20°C a 80°C



Faixa de operação: 0 a 30 psi

Precisão com RCX: 5%

Figura 3.27 - Sensor de Pressão



Faixa de operação: -24 a +24 Volts DC

Precisão com RCX: 4%

Figura 3.28 - Sensor para medida de Tensão DC

3.14 Atuadores

Do ponto de vista tecnológico os dispositivos mecânicos que aplicam ou fazem atuar energia mecânica sobre uma máquina, para realizar um determinado trabalho são chamados de atuadores. Atuadores são transdutores de saída, que alteram a grandeza controlada. Ou ainda, transdutores eletrônicos que convertem energia elétrica em uma quantidade física (Derenzo, 1990).

3.14.1 Atuadores do ponto de vista tecnológico

Atuadores adquirem diferentes significados em função da área tecnológica onde eles são usados.

Na área da aeronáutica, por exemplo, os atuadores são definidos como sendo elementos de potência na saída de qualquer sistema automático de controle de voo. Na robótica eles são os dispositivos de transmissão de potência mecânica que fornecem o “músculo” para mover o braço do robô (Groover, Weiss et al, 1988). Ainda nessa área, o atuador é definido como um efetuator, todo um sistema montado na extremidade do vínculo mais distante da base do robô, cuja tarefa é agarrar objetos, ferramentas e/ou transferi-las de um lugar para outro.

3.15 Como os atuadores são utilizados

Na prática os atuadores podem ser utilizados na forma de motores, eletroímã, freio magnético, fechadura magnética, válvula solenóide, calefadores, etc..

3.15.1 Atuador como motor

Nos sistemas de controle de posição e velocidade, nos robôs e máquinas industriais, impressoras, plotter's, toca-discos CD, videocassetes e unidades de disco de computadores são usados motores, freio magnético em geral junto a engrenagens ou correias, como atuadores, conforme a Figura 3.29.

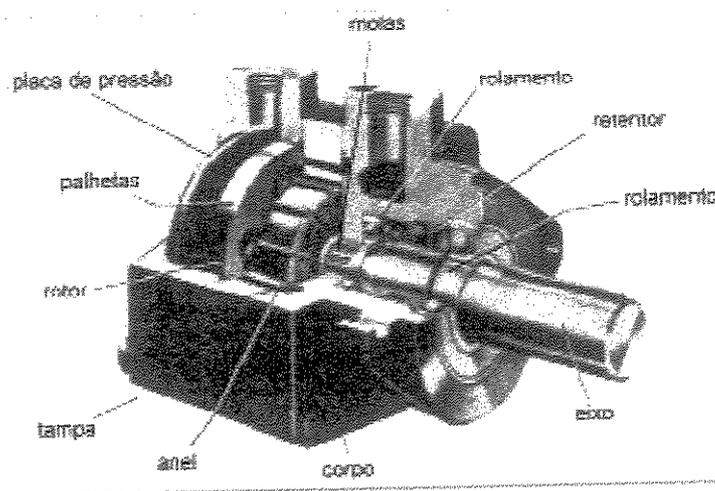


Figura 3.29 – Exemplo de um atuador como motor

3.15.2 Atuador como eletroímã

Em aplicações como o transporte de peças de ferro ou níquel, em guindastes, ou garras de robôs e travas magnéticas, são empregados eletroímãs, apesar do alto consumo de energia.

3.15.3 Atuador como freio magnético

Um tipo de eletroímã que paralisa o movimento rotativo de um motor. É comum em sistemas com reversão de sentido de rotação, reduzindo o golpe mecânico e o pico de corrente na reversão.

3.15.4 Atuador como fechadura magnética

Em sistemas de segurança, cofres e porteiros eletrônicos se utiliza uma fechadura cuja trava é liberada através de um eletroímã, com um breve pulso. Em certos casos o fechamento não é manual, mas através de outro eletroímã.

3.15.5 Atuador como válvula solenóide

Atuador é usado como válvula, conforme a Figura 3.30, em controle de fluxo de líquidos, em indústrias. Consiste de um ou mais caminhos que são interceptados por pistões, acoplados a eletroímãs, que liberam ou não o fluxo.

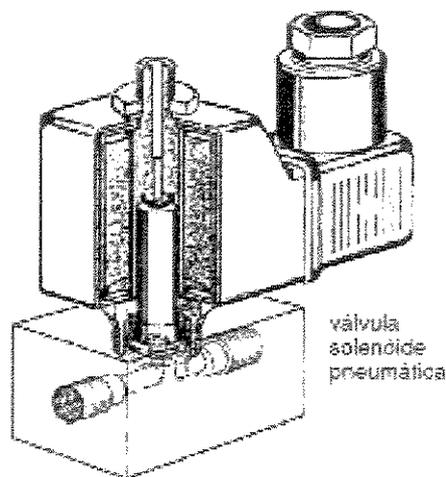


Figura 3.30 – Exemplo de um atuador como válvula

3.16 Atuadores de uso educacional

No que diz respeito a atividade experimental desta tese, os atuadores utilizados na implementação de dispositivos mecânicos automatizados foram basicamente motores de corrente contínua, e servo-motores e lâmpadas. As figuras a seguir mostram estes atuadores.

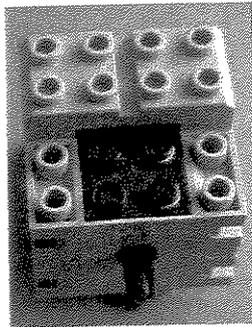


Figura 3.31 - Motor LEGO 9 volts com redução interna

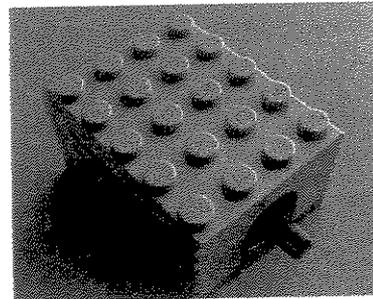


Figura 3.32 - Motor LEGO 9 volts sem redução interna

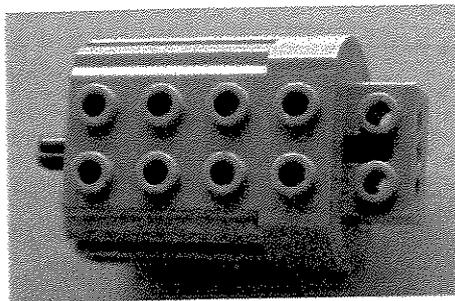


Figura 3.33 - Motor LEGO 4,5 volts

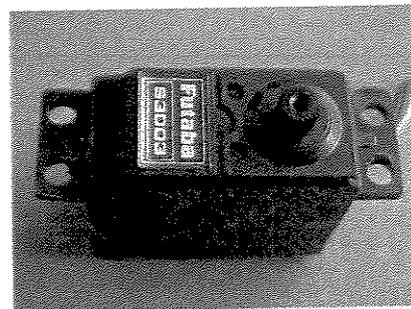


Figura 3.34 - Servo-motor da Futaba

3.17 Considerações sobre o Capítulo 3

Este capítulo discutiu uma das ferramentas utilizadas para especificação de projetos no campo da automação que é o Grafcet. Discutiu-se ao longo do capítulo sistemas automatizados tendo como enfoque a parte operativa e a parte comando desses sistemas. Foram apresentados os elementos de um Grafcet, etapas e ações associadas, transições receptividades e ligações. Além disso, foram apresentadas algumas soluções tecnológicas envolvendo a utilização do Grafcet e dois exemplos de projetos que abordam aplicação prática desta ferramenta no que diz respeito ao seu uso funcional. Para finalizar o capítulo, foi apresentada uma síntese enfocando diferentes

aspectos em que se pode utilizar Grafset na especificação do processo de automação de dispositivos em diferentes fases de trabalho. No próximo capítulo sensores e atuadores serão estudados como elementos operativos utilizados no contexto educacional.

Neste capítulo, estudou-se alguns tipos de sensores e atuadores aplicados à automação. No campo de automação quer seja para fins industriais, ou fins pedagógicos sensores e atuadores são utilizados como transdutores eletrônicos no controle de processos. Características técnicas de alguns tipos de sensores foram estudados, sobretudo aqueles baseados em foto-transistores, foto-diodos e infravermelhos por serem semelhantes tecnologicamente aos sensores LEGO utilizados nos projetos apresentados neste trabalho.

No que diz respeito a atuadores o capítulo apresentou, do ponto de vista tecnológico, alguns tipos desse dispositivo discutindo-se a sua funcionalidade em diferentes áreas das engenharias.

O capítulo apresentou também discussão sobre a importância de utilização de transdutores (atuadores/sensores) como elementos que permitem a transformação de uma forma de energia em outra, possibilitando assim com que eles sejam utilizados nas malhas de controle. Foram apresentados sistemas de controle em malha aberta e malha fechada enfocando suas vantagens e desvantagens.

Foi focado neste capítulo, de forma genérica sistemas de redução de velocidade em uma junta robótica que podem ser implementadas utilizando-se engrenagens. Buscou-se, no capítulo, quantificar matematicamente o efeito de se associar duas engrenagens com o objetivo de demonstrar, no contexto de montagens de princípios mecânicos com LEGO quão precisas essas montagens podem ser. Ou seja, demonstrar que os formalismos matemáticos utilizados nos projetos e implementação de máquinas industriais de grande complexidade podem, em certo modo, ser também utilizados em pequenos projetos desenvolvidos nos laboratórios da universidade onde atuamos.

Direta ou indiretamente os conceitos apresentados neste capítulo puderam ser trabalhados pelos alunos durante a elaboração dos projetos experimentais que serão discutidos a partir do próximo capítulo, tanto no que diz respeito à implementação de sistemas automatizados discretos

com ênfase na utilização de Grafocet quanto de sistemas contínuos com base em função de transferência em malha aberta e malha fechada.



Capítulo 4

Projetos Integrados utilizados para Validação de Conceitos

Com base em atividades, sobretudo oficinas de trabalho, desenvolvidas para alunos e professores de ensino fundamental, ensino médio, ensino superior e cursos de especialização, onde o enfoque era a utilização do ambiente LEGO para introduzir conceitos de ciências nas mais variadas áreas do conhecimento idealizou-se os projetos práticos experimentais que serão apresentados neste capítulo. Este capítulo trata de três projetos práticos experimentais desenvolvidos junto aos alunos da Engenharia de Controle e Automação (Mecatrônica), Engenharia Mecânica e Técnicos Especializados numa fábrica. Os projetos objetivam o aprendizado de conceitos de automação por meio de concepção, estruturação e programação de dispositivos mecatrônicos utilizando motores e sensores.

4.1 Montagem e Automação de Dispositivos Mecatrônicos

Ao se realizar a montagem, automação e controle de dispositivos mecatrônicos utiliza-se peças como: engrenagens, eixos, roldanas, polias, rodas, motores, lâmpadas, sensores, dentre outras. O controle/automação desses dispositivos é feito elaborando programas que possibilitam movimentar máquinas do tipo esteiras, elevadores, carrinhos, mesas giratórias, etc. ou que simulam movimentos de animais ou de um braço manipulador.

No processo de automação de um dispositivo LEGO, por exemplo, é descrito uma seqüência de comandos em linguagem de programação, do tipo Logo, Robolab, NQC³¹, RCX Code, etc., onde, um primeiro procedimento pode iniciar o funcionamento de uma máquina e transferir o fluxo de controle para um segundo procedimento. O segundo procedimento, por sua vez, poderá monitorar a mudança de estado de um sensor possibilitando, com isso, a parada de um motor e a transferência do fluxo para um terceiro procedimento e assim sucessivamente. Este processo é praticamente o mesmo, independentemente de se estar tratando da automação de uma máquina montada em LEGO ou de uma máquina real que está em uma linha de produção de uma determinada fábrica. Isso significa dizer que a seqüência de fluxo que é realizada, para automatizar uma máquina no ambiente LEGO-Logo, por exemplo, é similar à seqüência que é usada na indústria para a especificação de um sistema automatizado. Nas duas situações é necessário especificar, descrever em termos de uma seqüência lógica as operações/etapas que devem ser seguidas para executar uma determinada tarefa. No contexto educacional esta especificação é realizada implementando a montagem, automação e controle do dispositivo via computador. No contexto da fábrica ela é realizada utilizando ferramentas como o Grafset já abordado no capítulo 3.

Sob este ponto de vista, a abordagem adotada nesta tese enfocando a implementação de dispositivos mecânicos automatizados como forma de se criar ambientes de ensino-aprendizagem de conceitos, em diferentes níveis de ensino, tem se mostrado coerente. Com base nas diversas oficinas de trabalho realizadas, com esse enfoque, e também por meio das atividades experimentais desenvolvidas durante a elaboração desta pesquisa foi possível observar inúmeras situações de aprendizagem onde a montagem automação e controle de um dispositivo mecatrônico auxilia a compreensão de um determinado conceito.

A seguir, serão descritas algumas atividades experimentais desenvolvidas durante a tese. Estas atividades foram essencialmente de implementação, em LEGO e/ou outros materiais, de máquinas e sistemas similares aos existentes na fábrica buscando criar um ambiente propício para o aprendizado de conceitos de automação.

³¹ NQC é a sigla de Not Que C, é uma linguagem para programação dos produtos LEGO Mindstorms.

4.2 Motivação do Projeto Prático Experimental

Desenvolvimento e diversificação de ambientes de aprendizagem onde, por intermédio de atividades práticas o aluno pode trabalhar os conceitos teóricos, sobretudo de automação e controle, de forma contextualizada. É essa a motivação da parte experimental desta tese.

4.2.1 Atividade de Pesquisa Prática Realizada

A princípio a pesquisa a ser desenvolvida aborda a questão de desenvolvimento de dispositivos robóticos simples envolvendo a montagem de diversos modelos mecânicos primitivos centrados nos elementos de máquina. Para este fim foram utilizados, inicialmente, diversos componentes mecânicos LEGO para montagem e validação destes modelos. Posteriormente, os modelos poderão ser construídos, na forma mais definitiva, utilizando-se materiais mecanicamente mais resistentes.

Os modelos construídos de forma mais definitiva deverão ser implementados respeitando alguns aspectos, tais como, modularidade e facilidade de integração. Esses aspectos permitirão com que iniciemos uma montagem de forma simples e, paulatinamente, sejam agregados diversas partes até se chegar em modelos mais complexos. Isso apresenta uma grande vantagem, pois, à medida que surgirem avanços tecnológicos, tanto da área de software quanto de hardware, esses avanços poderão ser empregados para aprimorar e atualizar o nosso modelo. A nossa intenção é “não reinventar a roda” a todo instante mas sim utilizar-se de recursos e de trabalhos já existentes objetivando melhorar e inovar/aprimorar o nosso trabalho. Para isso, a implementação e integração de diversos modelos deverão propiciar a construção de diferentes sistemas robóticos.

Neste sentido, a implementação do projeto de pesquisa envolveu primeiro, uma discussão do protótipo do ambiente a ser construído. Segundo, a elaboração do projeto mecânico incluindo especificações técnicas de software, hardware e dos dispositivos eletrônicos e eletromecânicos a serem utilizados. Terceiro, obtenção de infra-estrutura básica para se iniciar a construção do primeiro protótipo. Quarto, a montagem do ambiente propriamente dito. Quinto, o desenvolvimento de uma metodologia para a sua utilização com propósitos educacionais,

ultrapassando a sala de aula formal e alcançando outros ambientes de aprendizagem como fábricas que, diferentemente da universidade, apresentam uma situação mais realista de aplicação dos conceitos.

Não podemos deixar de ressaltar que, por se tratar da implementação de um ambiente educacional, o desenvolvimento de metodologia, embora seja colocada por último, meramente por uma questão cronológica, ele permeia toda as outras etapas do projeto de pesquisa. Portanto o uso pedagógico do ambiente deverá ser levado em consideração durante todo o processo de implementação. A seguir serão apresentados três exemplos de projetos práticos desenvolvidos: **Sistema Integrado para Triagem de Produtos em uma Linha Automatizada de Produção, Utilizando o Sistema LEGO-Logo e Robix; Sistema Automatizado de Produção SAP e Máquina de Solda por Ponto.**

4.3 Sistema Integrado para Triagem de Produtos em uma Linha Automatizada de Produção (desenvolvido na Universidade)

Em uma linha automatizada de produção, é freqüente a utilização de um sistema de reconhecimento de cores. Pode-se citar, como exemplo, a separação de frascos de diferentes tipos de *shampoo* de acordo com a cor da tampa. Este projeto, que foi um trabalho de iniciação científica desenvolvido por um aluno do curso de mecatrônica que havia desde o primeiro semestre desse curso aprendido a trabalhar com LEGO, aproxima-se muito de um processo industrial real, pois permite a separação de peças em uma linha automatizada de produção a partir do reconhecimento de cor de tijolos LEGO, nas cores amarelo, azul, cinza, preto, verde e vermelho (Takiguthi, 2001). Este projeto foi apresentado e obteve aprovação da FAPESP e desenvolvido durante o ano de 2001.

O objetivo fundamental do projeto de pesquisa consistiu na integração de dispositivos de automação industrial propiciando por meio destes a montagem de uma plataforma que tinha por finalidade a triagem de produtos em uma linha de produção a partir do reconhecimento de cores. A Parte Operativa desse sistema automatizado foi implementada utilizando componentes LEGOTM, disponível no Laboratório de Automação Integrada e Robótica – LAR / UNICAMP. A

interface de supervisão e controle foi implementada na linguagem Delphi™. O desenvolvimento desse sistema automatizado permitiu a integração de ferramentas industriais de automação numa plataforma direcionada ao ensino e pesquisa em mecatrônica.

4.3.1 Em que consistiu o projeto

Este projeto, que tem bastante similaridade com o processo industrial real de separação de peças em uma linha automatizada, consistiu na integração de dispositivos de automação industrial por meio da montagem de uma plataforma para triagem de produtos, a partir do reconhecimento de cores básicas, utilizando-se componentes LEGO (Kits LEGO Mindstorms) e um braço mecânico Robix disponíveis no Laboratório de Automação Integrada e Robótico - LAR.

4.3.2 Divisão da Plataforma

A plataforma pôde ser dividida em três partes: um Sistema de Transferência de Peças, um Sistema de Reconhecimento de Cor e um Sistema de Separação de Peças. Cada sistema foi implementado e testado separadamente e, uma vez em funcionamento de maneira eficaz, eles foram integrados.

Para a montagem do Sistema de Transferência e do Sistema de Reconhecimento de Cor, foram utilizados componentes do kit LEGO™ Mindstorms, tais como: motor, engrenagens, esteira, eixos, sensor de luz. Foi utilizado também o manipulador robótico ROBIX™ RS-6 (Figura 4.1), que compõe o Sistema de Separação de Peças.

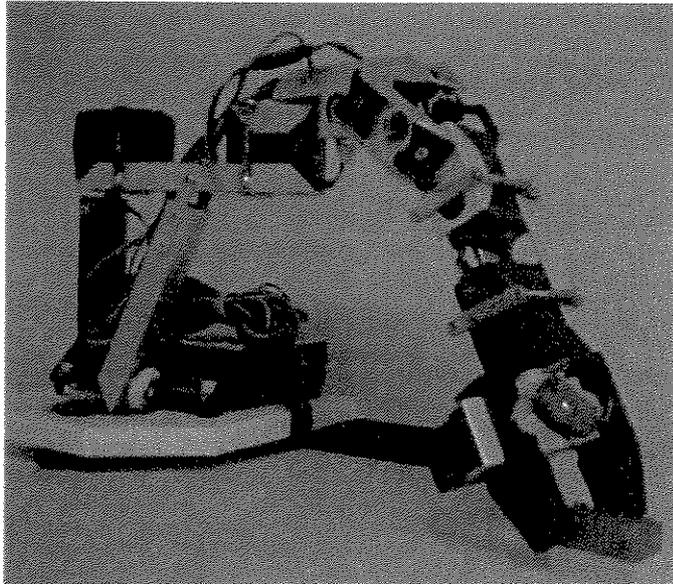


Figura 4.1 – Manipulador Robix

Para realizar o controle do processo e a integração entre estes três sistemas, foi implementado um software em linguagem Delphi™. Esse software obtém informações importantes sobre o processo de reconhecimento de cor, tais como o horário de identificação e a quantidade de peças cuja cor foi reconhecida. A seguir será descrito sucintamente cada um desses sistemas.

4.3.3 Sistema de Transferência de Peças

Este sistema é responsável pelo transporte das peças a partir de um ponto de alimentação de peças até seu posto de retirada, passando pelo posto de leitura de cor (Figura 4.2). O Sistema de Transferência consiste em uma esteira composta por tijolos comuns, engrenagens, eixos, esteira e motor, todos componentes LEGO™.

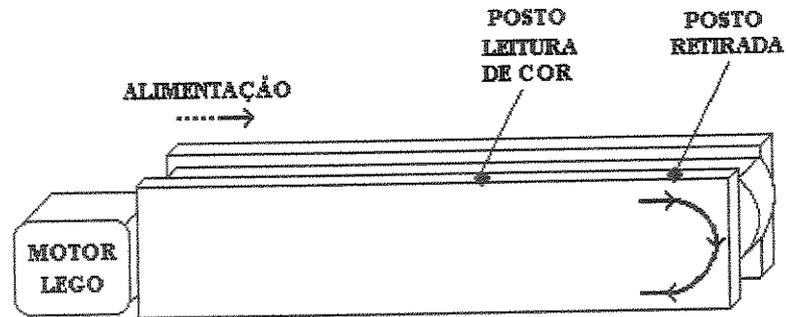


Figura 4.2 – Sistema de transferência de peças

4.3.4 Sistema de Reconhecimento de Cor

Este sistema tem por finalidade identificar as cores dos tijolos LEGO que passam pelo Sistema de Transferência. O Sistema de Reconhecimento de cor é composto por um sensor de luz, um sensor magnético e a esteira, conforme as Figuras 4.3 e 4.4. O sensor de luz tem a função de reconhecer a cor do tijolo LEGO posicionado pela esteira à sua frente, e o sensor magnético, a de detectar a peça durante a fase de reconhecimento de cor.

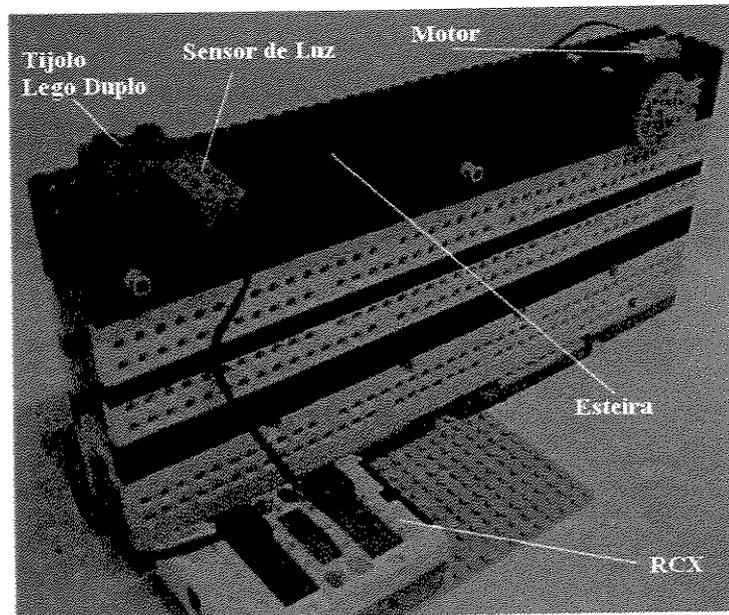


Figura 4.3 – Sistema de reconhecimento montado em LEGO

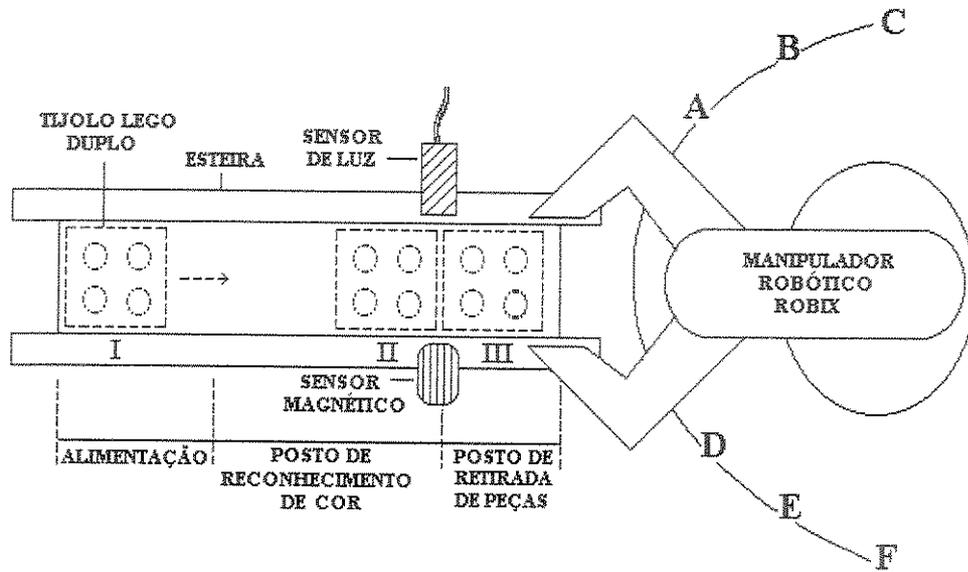


Figura 4.4. – Sistema de reconhecimento com sensor magnético

4.3.5 Sistema de Separação de Peças

Este sistema é responsável pela retirada dos tijolos LEGO no final do Sistema de Transferência e posicioná-los, de acordo com a cor, nas seis plataformas A, B, C, D, E e F, como ilustra a figura 4.4. O Sistema de Separação de Peças é composto por um Manipulador Robótico Robix™, com cinco graus de liberdade e seis plataformas para disposição das peças.

4.3.6 Funcionamento do Sistema Integrado

O processo se inicia com o recebimento de um tijolo inserido no sistema de transferência (Posição I na Figura 4.4), cuja cor ainda não foi identificada pelo sistema de reconhecimento de cor. O tijolo é transportado ao longo da esteira. Ao atingir o Posto de Reconhecimento de Cor (Posição II), ele é detectado pelo sensor magnético e terá a sua cor identificada por meio do sensor de luz.

A fase seguinte se inicia com o tijolo deixando o Posto de Reconhecimento e chegando no Posto de Retirada (Posição III) para a separação. Posicionado o tijolo em III, o manipulador robótico Robix™ retira-o do Sistema de Transferência e, de acordo com a cor, coloca-o em uma das seis plataformas (A, B, C, D, E ou F) ao redor do manipulador Robix™. O controle do motor da esteira e do sensor de luz é feito pelo tijolo programável RCX. O manipulador robótico Robix™ possui uma interface de comunicação entre ele e o computador e um software para a programação das trajetórias por ele realizado.

As tabelas 4.1 e 4.2 a seguir mostram as variáveis de entrada e saída e, os respectivos significados. Estas variáveis foram definidas para se fazer um Grafset da plataforma integrada Figura 4.5.

Tabela 4.1 - Variáveis de Entrada para Grafcet da Plataforma

| Variáveis de Entrada | |
|----------------------|---|
| Variável | Significado |
| SM | Sensor Magnético |
| Sluz | Sensor de Luz |
| Rob0 | Resposta do Robix quando na posição inicial |
| Rob1 | Resposta do Robix quando executada trajetória 1 |
| Rob2 | Resposta do Robix quando executada trajetória 2 |
| Rob3 | Resposta do Robix quando executada trajetória 3 |
| Rob4 | Resposta do Robix quando executada trajetória 4 |
| Rob5 | Resposta do Robix quando executada trajetória 5 |
| Rob6 | Resposta do Robix quando executada trajetória 6 |
| Rob7 | Resposta do Robix quando na posição para desligamento |

Tabela 4.2 - Variáveis de Saída para Grafcet da Plataforma

| Variáveis de Saída | |
|--------------------|---|
| Variável | Significado |
| Motor | Motor da esteira |
| 1 Traj0 | Execução da trajetória 01 pelo Robix – Plataforma A |
| 2 Traj0 | Execução da trajetória 02 pelo Robix – Plataforma B |
| 3 Traj0 | Execução da trajetória 03 pelo Robix – Plataforma C |
| 4 Traj0 | Execução da trajetória 04 pelo Robix – Plataforma D |
| 5 Traj0 | Execução da trajetória 05 pelo Robix – Plataforma E |
| 6 Traj0 | Execução da trajetória 06 pelo Robix – Plataforma F |
| 7 Traj0 | Execução da trajetória 06 pelo Robix - Posição Desligamento |

Grafcet

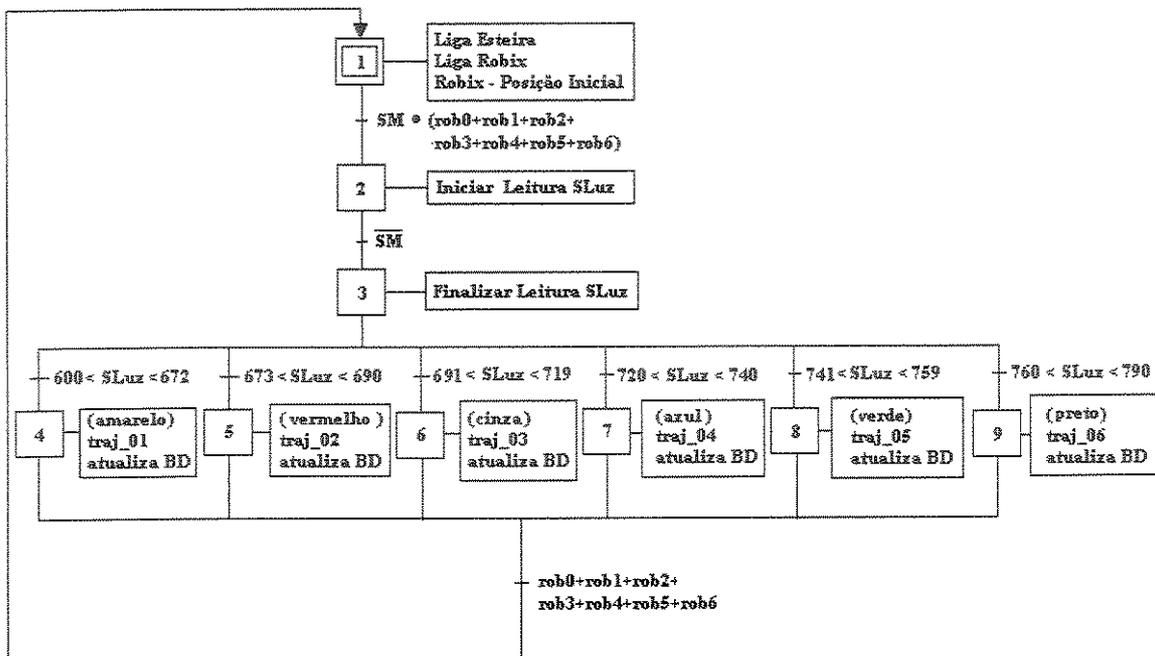


Figura 4.5 – Grafcet da plataforma integrada

4.3.7 Software

De forma sucinta, o software implementado neste projeto continha os seguintes módulos principais:

- Módulo de configuração do RCX (Figura 4.6), em que se pode obter as informações da comunicação entre o RCX e o computador além de selecionar a porta serial a ser utilizada, fazer o carregamento do sistema operacional do RCX e iniciar as configurações do motor e do sensor de luz. Mais informações sobre programas elaborados neste projeto encontram-se no anexo 4 (**Programas do Sistema de Triagem de Produtos**).

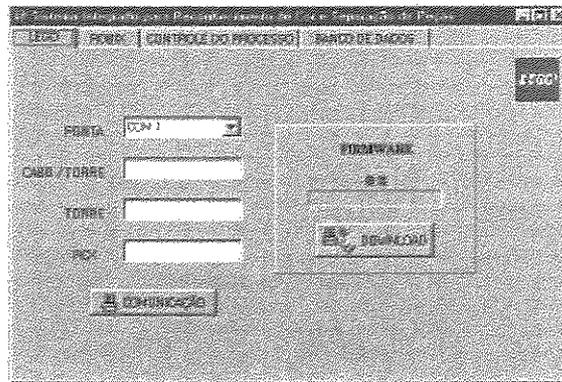


Figura 4.6 - Módulo de configuração do RCX

- Módulo para o controle e supervisão do processo de identificação e separação de tijolos (Figura 4.7), que é responsável pela integração dos sistemas.

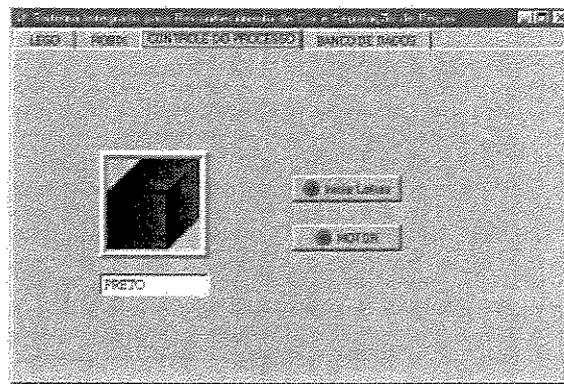


Figura 4.7 - Módulo para o controle e supervisão do processo

- Módulo banco de dados, onde serão armazenadas informações a respeito da peça como cor, horário e data de identificação (Figura 4.8) e gráfico da produção (figura 4.9).

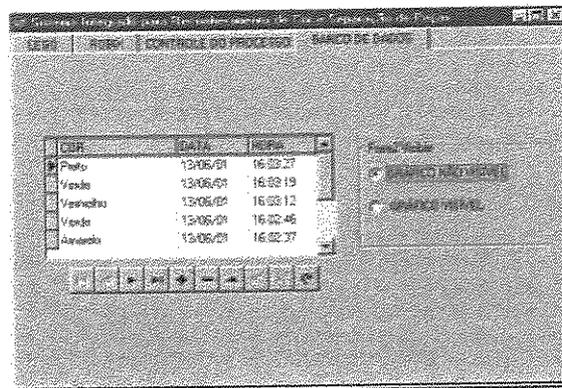


Figura 4.8 – Módulo Banco de Dados contendo cor, horário e data da identificação.

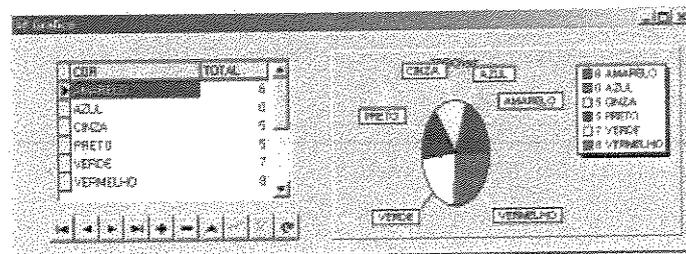


Figura 4.9 – Módulo Banco de Dados contendo Quantidade de peças e Gráfico

4.3.8 Conceitos abordados

Acompanhando a implementação deste projeto pode-se observar algumas abordagens conceituais relacionadas com a forma como o desenvolvimento dos trabalhos se deu. Dentre estas abordagens destaca-se a documentação/descrição das partes do projeto, que foi o momento onde mais se questionou os aspectos conceituais de conteúdos de Engenharia de Controle e Automação trabalhados durante a montagem da plataforma. Dos conceitos abordados podem ser mencionados, a programação tanto do Robix quanto do RCX na linguagem Delphi, elaboração de banco de dados, controle de processos em tempo real, especificação com a implementação do

Graficet dos movimentos do Robix, montagem de dispositivos mecatrônicos com LEGO, dentre outros.

4.4 Utilizando O Sistema LEGO-Logo e Robix: Sistema Automatizado de Produção³² SAP (desenvolvido na Unicamp)

Neste projeto foi implementado um sistema que permitiu a interação LEGO-Robix criando assim um ambiente montagem/simulação de linha de produção de uma fábrica que utiliza sistema automatizado de produção, onde a Parte Operativa foi feita usando montagens LEGO a Parte Comando foi feita usando a linguagem de programação Logo e programa de controle do Robix. No que diz respeito ao sistema automatizado de produção conceitos relativos à Parte Operativa e Parte Comando também poderiam ser trabalhados como forma de se sistematizar as operações que as máquinas executavam (Passeto, 2000).

Com o Sistema LEGO, na versão que possui componentes elétricos e peças mecânicas que possibilitam montagem de dispositivos que se movimentam, os alunos de graduação do curso de engenharia mecânica da Unicamp construíram uma esteira rolante e uma mesa rotativa, similares as utilizadas nas fábricas.

Com o Sistema Robix, um braço mecânico, que combina as características de robótica de nível industrial com facilidades de ser usado com propósito educacional, foi possível fazer a integração entre a esteira e a mesa, permitido que o conjunto, esteira, braço mecânico Robix e a mesa rotativa constituíssem uma linha de produção composta respectivamente de três células.

A automação dos dispositivos montados com LEGO foi feita por meio de microcomputadores utilizando a linguagem Logo, num ambiente chamado de LEGO-Logo. Neste

³² Este projeto deu origem a um artigo científico intitulado Desenvolvimento de Ambientes de Aprendizagem Baseados no Uso de Dispositivos Robóticos, que foi publicado nos anais do X Simpósio Brasileiro de Informática na Educação – SBIE99 “As Novas Linguagens da Tecnologia na Aprendizagem”, Universidade Federal de Paraná – UFPR, Curitiba – PR de 23 a 25/11/99.

caso, programas elaborados – cuja descrição reflete o comportamento do funcionamento dos dispositivos – controlavam os componentes elétricos (motores, sensores e lâmpadas) LEGO.

A automação do Robix foi feita utilizando programas específicos desenvolvidos na linguagem C para comando de servo-motores que controlam os movimentos do Robix.

4.4.1 Vantagem do ponto de vista educacional

Do ponto de vista pedagógico educacional, a grande vantagem existente na simulação de uma linha de produção utilizando esses materiais está em podermos propiciar aos alunos uma vivência de ambiente de uma fábrica dentro de um laboratório da universidade. No que diz respeito ao sistema automatizado de produção, a Parte Operativa de uma fábrica é específica, inerente somente a ela, entretanto, a Parte Comando é genérica, à medida que os conceitos envolvidos na lógica de programação independem da linguagem a ser utilizada. E, neste caso, tanto LEGO quanto Robix permitem montar células de produção com diferentes configurações "simulando" diferentes linhas de produção. Para uma determinada fábrica, embora a Parte Operativa seja específica, utilizando componentes LEGO, existem condições de diversificar a montagem e a disposição nas diferentes células de produção. Isso se constitui em uma vantagem pedagógico-educacional, na medida em que diferentes abordagens e hipóteses podem ser apresentadas e estudadas à medida que as configurações se alteram. Isso tem propiciado o desenvolvimento de estratégias para resolução de problemas.

Ainda no que diz respeito ao sistema automatizado de produção, a utilização tanto do LOGO quanto da linguagem de programação do Robix tem a vantagem de que os conceitos aplicados são genéricos, ou seja, não deixam nada a desejar com relação aos conceitos utilizados numa indústria, bastando, para tanto, adaptá-los à realidade da indústria em questão. Do ponto de vista pedagógico educacional isso representa um grande ganho.

Na montagem das células de produção com o LEGO, o ambiente tem permitido a utilização de peças mecânicas e conseqüentemente dos princípios mecânicos necessários para construção de algumas máquinas. Na automação, o ambiente tem permitido a descrição, a elaboração e a

estruturação de programas para o computador. No controle da operação da linha de produção como um todo, o ambiente tem permitido que conceitos importantes sobre sistemas automatizados de produção sejam manuseados.

A interação LEGO Robix permitiu a montagem de um ambiente de simulação de uma linha de produção de uma fábrica, utilizando um sistema automatizado de produção, como esta que vem descrita a seguir.

4.4.2 Linha de produção composta de três postos

O primeiro posto (Figura 4.10), é uma esteira rolante alimentadora de peças cuja montagem utilizou sensores de luz, lâmpadas, motores e outros elementos mecânicos. O segundo posto (Figura 4.11), é um braço mecânico, o Robix que retira peças. O terceiro posto (Figura 4.12), é uma mesa rotativa para trabalho de peças.

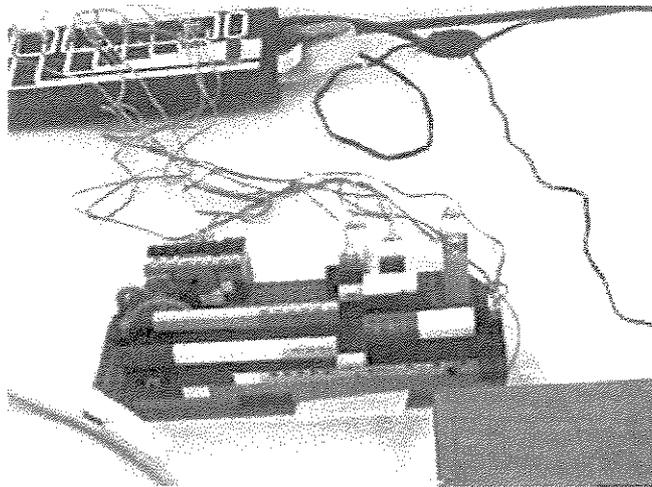


Figura 4.10 - Esteira rolante alimentadora de peças

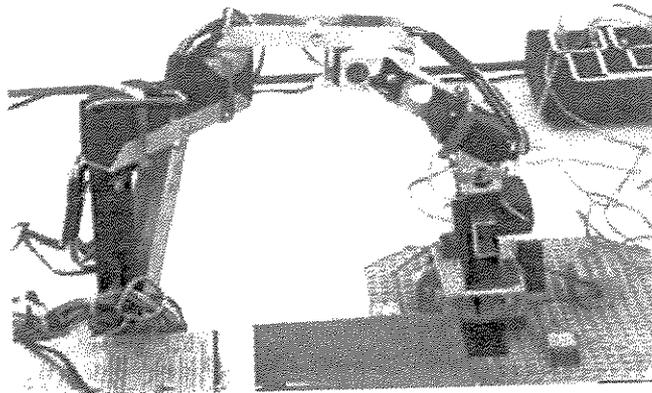


Figura 4.11 - Braço mecânico, Robix

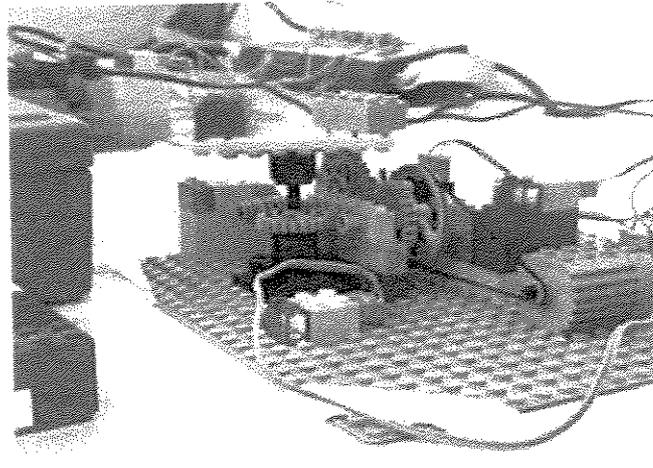


Figura 4.12 - Mesa giratória para trabalho de peças

4.4.3 Aspectos do funcionamento de célula de produção

Quando a linha de produção entra em operação depois de retirada uma peça da esteira para a mesa, pelo Robix, automaticamente a esteira se movimenta colocando uma nova peça na posição para ser retirada. Enquanto a primeira peça colocada na mesa está sendo trabalhada a segunda é colocada na esteira. Ao término do trabalho na primeira peça a mesa faz um giro de 180°, a peça trabalhada é retirada e colocada numa caixa. Todo o processo de colocar e retirar peças é feito de forma automática pelo braço mecânico Robix. A operação é feita seqüencialmente de tal forma que sobre a mesa sempre existam duas peças, uma trabalhada e

outra em trabalho. Além disso, a linha pode ser programada para processar lotes diferentes de peças. Por exemplo, para processar cinco lotes de duas peças, depois de retiradas as dez peças, pelo braço mecânico, a décima primeira peça será rejeitada automaticamente pela esteira.

4.4.4 Conceitos abordados

No controle da operação da linha de produção, como um todo, conceitos teóricos importantes sobre sistemas automatizados de produção, apresentados na sala de aula pelo professor, são manuseados pelos alunos no laboratório. Com este tipo de atividade, onde se integra diferentes dispositivos para se criar a célula de produção, os alunos aprendem/manuseiam conceitos importantes de Engenharia de Integração e de Sistema Automatizado de Produção-SAP. Essas atividades têm apresentado algumas características interdisciplinares na medida em que possibilitam a integração sinérgica da engenharia mecânica com a eletrônica e o controle por computador, do projeto e da manufatura de produtos. Isso certamente contribuirá para a formação de profissionais habilitados a projetar e a produzir produtos inteligentes baseados em microprocessadores e sistemas de controle. O engenheiro formado nesta área terá como campo de trabalho essencialmente a indústria de processos de manufatura, segmento em que se encontram as metalúrgicas e as fábricas de autopeças, por exemplo. Portanto, é importante na formação, deste profissional que ele aprenda a elaborar estudos e projetos, bem como participar da direção, organização, fabricação e fiscalização de atividades relacionadas com o controle de processos e a automação de sistemas produtivos. Isto é, que ele seja capaz de harmonizar soluções e equipamentos que integram as áreas de: mecânica, elétrica, eletrônica, computação, automação, *design* de produtos, gerenciamento da produção, dentre outras áreas afins.

A utilização do ambiente LEGO-Logo em integração com o Robix tem possibilitando, no curso de graduação em Engenharia de Controle e Automação (Mecatrônica) desenvolver atividades, num laboratório da universidade projetos muito próximo à realidade da fábrica. Com isso, está-se iniciando a construção de um ambiente de ensino e aprendizagem que nos propiciará, de forma concreta, realizar a Engenharia de Integração, onde diferentes tipos de dispositivos robóticos podem ser integrados e utilizados com finalidades educacionais.

4.5 Máquina de Solda Automática (desenvolvido na fábrica)

O projeto teve como objetivo criar um ambiente de aprendizagem na fábrica onde trabalhadores de diferentes níveis puderam compreender os conceitos de automação inerentes às máquinas reais que eles operavam no dia-a-dia, buscando ajudar os técnicos especializados da Delphi-Harrison, a entender de forma concreta, a partir de uma oficina de trabalho, conceitos de automação que eles usavam empiricamente. Da oficina participaram engenheiros de controle de qualidade, técnicos da linha de montagem e técnicos de manutenção. Esses trabalhadores construíram máquinas similares às existentes na fábrica, tornando assim mais significativo a sua aprendizagem. De modo geral, as máquinas possuíam motores e sensores o que permitiu criar situações que possibilitaram discutir, de forma contextualizada, sistemas de segurança, de manutenção e melhorias no sistema de produção. Os dispositivos implementados podiam ser divididos em duas categorias: primeiro, os que se preocupavam com o aprimoramento da segurança das operações executadas na célula de trabalho e segundo, aqueles que ajudavam na compreensão de um mecanismo específico. Da primeira categoria, foi construído um sistema de segurança que impedia com que a máquina entrasse em operação quando o operário estivesse com as mãos em uma determinada área definida como área de segurança. A segunda categoria, envolveu a construção da máquina de solda automática que utilizava recursos tecnológicos que a máquina de solda real da fábrica não tinha. Essa máquina de solda automática será descrita a seguir.

4.5.1 Máquina de Solda por Ponto³³

O projeto consistiu em construir uma máquina de solda automática, Figura 4.13, utilizando **peças mecânicas** LEGO tais como: rosca sem-fim, engrenagens, eixos, cremalheiras, correias dentadas etc., montando uma estrutura mecânica parecida com uma máquina de solda real. As peças foram devidamente acopladas respeitando, alguns princípios da Mecânica, da Física, da Matemática, dentre outros. Também foram utilizados como **componentes elétricos** LEGO, motores e sensor de luz, para possibilitarem a geração de movimento e medida do ângulo de giro.

³³ Este nome foi definido para um soldador automático que fazia solda em vários pontos numa peça colocada sobre uma base giratória.

O desenvolvimento desse projeto envolveu a montagem do sistema mecânico da máquina de solda, em si, e a automação desta.

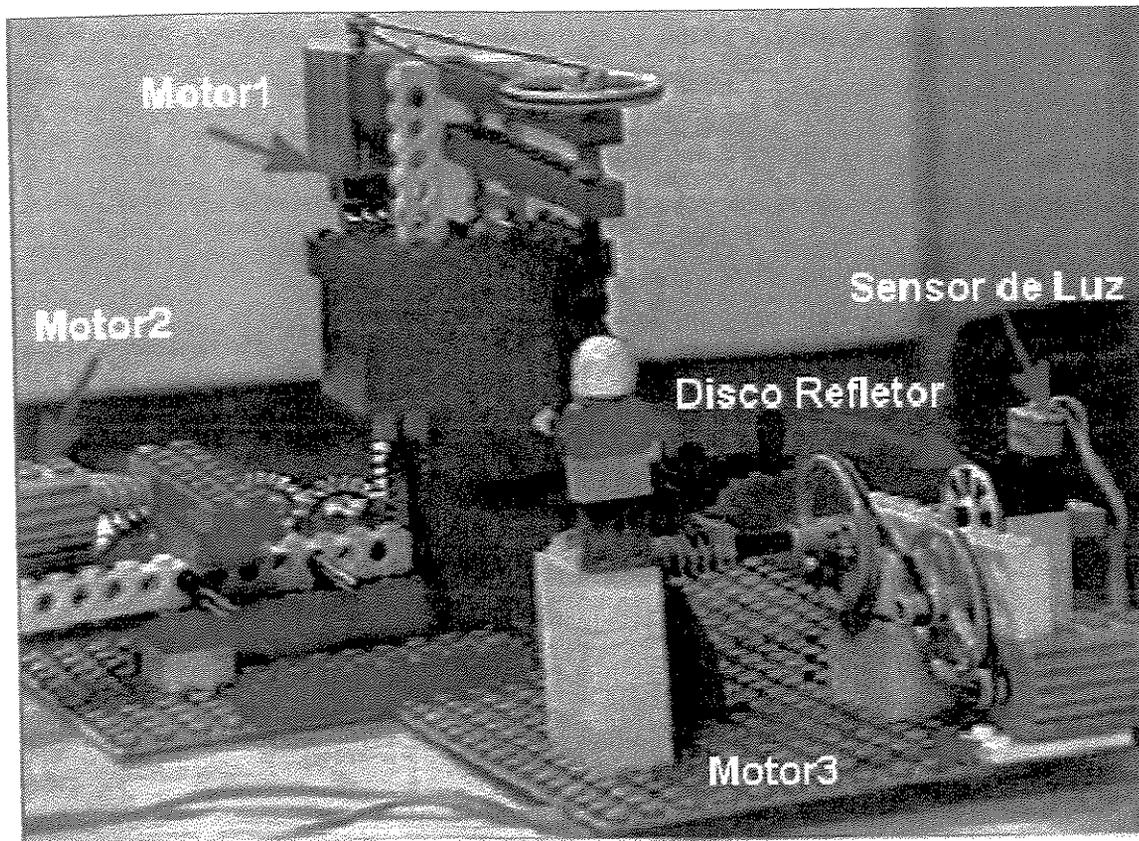


Figura 4.13 - Máquina de Solda por ponto

4.5.2 Sistema mecânico da máquina de solda

O sistema mecânico da máquina de solda é composto de um soldador e de uma mesa giratória onde a peça a ser soldada é fixada. Esse sistema pode ser dividido em três subsistemas das quais fazem parte, conforme Figura 4.13, **motor1**, **motor2** e **motor 3** junto com o **sensor de luz** e **disco refletor** “*encoder*”:

- Mecanismo composto de roldanas, polias, eixos, **motor1** e demais peças é o soldador em si. Esse mecanismo realiza a solda em um determinado ponto na peça que está sobre a mesa giratória.

- Mecanismo composto de jogos de engrenagens, eixos, rosca sem fim, **motor2** e demais peças, posiciona o soldador no ponto onde deve ser feita a solda. Esse mecanismo realiza movimento na vertical para abaixar ou levantar o soldador.
- Mecanismo composto de roldanas, polias, **disco refletor**, sensor de luz, **motor3** e demais peças controla o giro da mesa. Esse mecanismo faz com que, a cada solda realizada num determinado ponto, a mesa faça um giro de 10 graus para que uma nova solda seja realizada em um outro ponto. Ao todo a mesa realiza 4 giros correspondendo aos 4 pontos de solda.

O **sensor de luz** e o **disco refletor**, acoplado a um eixo, que por sua vez está conectado ao eixo do **motor3**, permite que esse eixo gire e o sensor mude de estado (**verdadeiro/falso; 0/1**), enquanto a mesa giratória está em movimento. A cada 10 mudanças de estado do sensor o **motor3** é desligado e a mesa pára de girar. Isso serve para posicionar a peça, de forma precisa, na posição certa para que a solda seja realizada.

4.5.3 Automação da Máquina de Solda

No contexto de LEGO-Logo automatizar uma máquina significa criar, uma interação por intermédio do programa elaborado, entre as peças mecânicas e os componentes elétricos, isto é, descrever para o computador, em termos de uma seqüência lógica, os comandos que acionam luzes, motores e sensores conectados às estruturas mecânicas. Quando colocamos a máquina em funcionamento esta seqüência lógica deve permitir que o movimento da estrutura mecânica seja o mais próximo possível da máquina real idealizada.

Automatizar a máquina de solda significou elaborar programas em TcLogo que permitissem com que o acionamento dos três motores e do sensor de luz representasse os movimentos e operações que um soldador automático e uma mesa giratória realizam durante o processo de solda por ponto.

4.5.4 Programas elaborados em TcLogo

Para programar a máquina de solda foram utilizados basicamente os seguintes comandos:

| | |
|---|---|
| Counter | Retorna o nº de vezes que o estado do(s) sensor(es) foi mudado de 0 para 1(verdadeiro/falso) a partir do último comando resetc |
| If true/false list-to-run | Se (verdadeiro/falso) for verdadeiro, a lista é executada caso contrário ela é ignorada (comando para execução de uma lista) |
| Listento (lto) port/portlist lto port/portlist | Indica que porta (s) será(ão) usada(s) para receber(em) mensagem(ns) do(s) sensor(es). As portas de entrada são 6 e 7 |
| Off | Desliga o(s) motor(es) ou a(s) luz(es) das portas A, B e C ou 0, 1, 2, 3, 4 e 5 |
| On | Liga o(s) motor(es) ou a(s) luz(es) das portas A, B e C ou 0, 1, 2, 3, 4 e 5 |
| Onfor time | Liga por um certo tempo o(s) motor(es) ou a(s) luz(es) das portas A, B e C ou 0, 1, 2, 3, 4 e 5 |
| Rd | Inverte o sentido de rotação do(s) motor(es) das portas A, B e C |
| Resetc | Reinicializa o(s) contador(es) a partir de zero referentes às entradas 6 e 7 ativadas pelo listento |
| Sensor? | Reporta o estado do sensor (verdadeiro/falso) |
| Stop | Pára o procedimento que estiver sendo executado |
| Talkto port/portlist Tto port/portlist | Indica que porta será usada |

De posse desses comandos foram elaborados os procedimentos **solda**, **controle** e **robot**, da seguinte maneira:

| | |
|---|---|
| <pre> to solda tto 3 on lto 6 resetc tto 4 on controle recycle solda end </pre> | <p>Esse procedimento inicializa o processo de solda zerando o contador ativando motor2, motor3 e passando a execução para o procedimento controle.</p> |
|---|---|

| | |
|--|--|
| <pre> to controle if counter = 40 [tto 4 off robot stop] controle end </pre> | <p>Esse procedimento controla o motor3 e também o estado do sensor verificando se o contador chegou em 40. Se verdadeiro então os 4 pontos de solda foram realizados e pode-se iniciar nova solda chamando o procedimento solda. Caso contrário, o procedimento robot continuará sendo executado.</p> |
| <pre> to robot tto [a] onfor 16 tto 2 onfor 3 tto [a] rd onfor 16 rd end </pre> | <p>Esse procedimento controla o motor2 para abaixar e levantar o soldador e o motor1 para realizar a solda. Após realizada a solda em um ponto o soldador é levantado enquanto a mesa gira para posicionar um novo ponto.</p> |

Com esses três procedimentos foi possível controlar a máquina de solda por pontos. Uma vez terminada a montagem e a automação da máquina, o projeto estava concluído passando então para a fase de apresentação.

4.5.5 Apresentação do Projeto pelos Técnicos Especializados

Embora durante a oficina tenham ocorrido momentos de pausa e reflexão, acerca do projeto que estava sendo desenvolvido, existiu um momento final bastante importante que foi o de socialização do projeto desenvolvido. Essa socialização se deu por meio da apresentação do processo e do produto implementado. Ela continha a síntese, entendida como a totalização dos conhecimentos produzidos. Foi o momento em que cada participante teve para falar, “narrar” aos colegas como foi a sua participação, o seu envolvimento, as dificuldades que surgiram, as tentativas, os fracassos e as soluções encontradas. Enfim, foi o momento de reflexão que propiciou às pessoas envolvidas desenvolverem o poder da argumentação.

O exemplo da máquina de solda por ponto construída pelos técnicos especializados da Delphi-Harrison, descrito anteriormente, é típico de uma atividade de montagem de um dispositivo automatizado passível de ser desenvolvido na fábrica. A Parte Operativa da máquina de solda consistiu na montagem/integração de diferentes componentes LEGO respeitando-se alguns princípios mecânicos para se obter um dispositivo parecido com uma máquina de solda. A

Parte Comando envolveu descrição, funcionamento, definição de entradas e saídas, programa de controle, enfim toda a especificação da automação dessa máquina.

4.5.6 Conceitos abordados e Considerações sobre o projeto

O processo de montagem e automação dessa máquina propiciou o estudo de conceitos nas áreas de matemática, física, *design*, automação e controle, programação, dentre outras áreas de conhecimento, partindo-se de um problema real. Isso mostra que o ambiente LEGO-Logo é um ambiente de ensino-aprendizagem que pode ser introduzido numa fábrica como forma de aprimorar o nível de conhecimento dos trabalhadores do mais baixo ao mais alto nível hierárquico.

Do ponto de vista de montagem, as construções mecânicas eram perfeitas uma vez que os trabalhadores tinham boa familiaridade com montagem de mecanismos. Entretanto, a programação dos dispositivos em Logo apresentava problemas. Era difícil para eles entenderem, por exemplo, que a mudança de intensidade de luz captada por um sensor pode ser lida pelo computador e isso podia ser usado para alterar o fluxo de execução de um programa. Esse fenômeno físico podia ser usado para controlar um motor conectado a um computador, fazendo com que este ligue/desligue, de forma automática, sempre que o sensor de luz mude de estado (0 para 1). Na verdade, o que verificamos é que não sabiam descrever para o computador como a máquina que eles construíram deveria funcionar, em função da falta de conhecimento científico que fundamentasse a experiência prática que eles tinham em operar a máquina real. Naquele momento era necessária a busca de uma teoria que fundamentasse a prática (Fazenda, 1993). No decorrer da oficina realizávamos várias vezes pausas para reflexão. Essas pausas eram os momentos de interação entre os participantes que envolviam, não somente a discussão da máquina que estava sendo montada como também as questões relativas à formalização de alguns conceitos, tanto de mecânica quanto de outras áreas de conhecimento. A partir dessas discussões, muitos técnicos puderam entender o grau de automação existente em algumas máquinas com as quais trabalhavam, e propor mais automação para outras delas. No processo de se construir essas máquinas as pessoas puderam aprender sobre o comportamento delas. A elaboração de programas que controlavam as máquinas utilizava comandos que acionavam luzes, motores e sensores. O

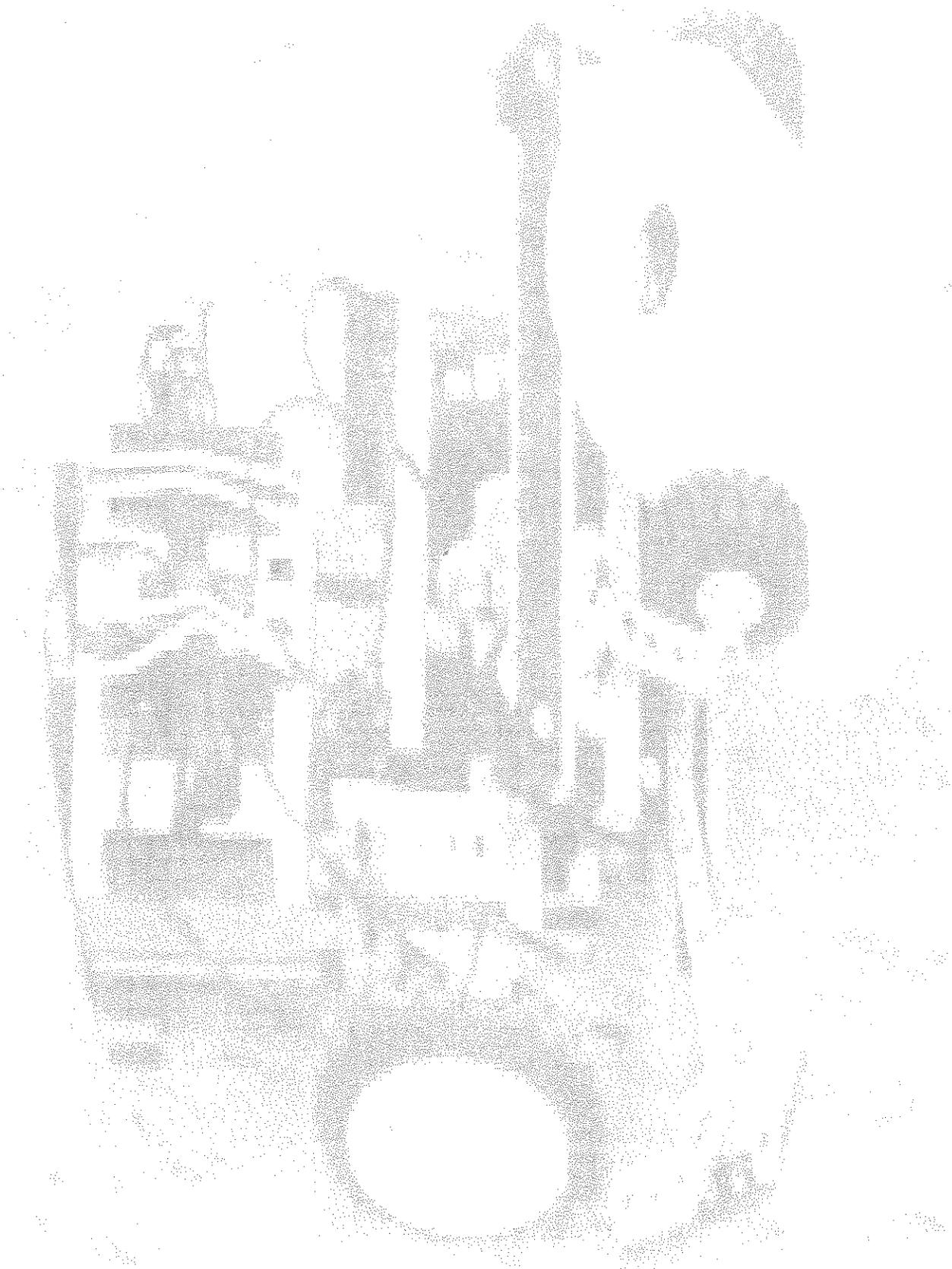
projeto e o estudo realizado sobre a máquina de solda por ponto pode perfeitamente ser realizado para uma outra máquina de uma célula de produção.

4.6 Considerações sobre o Capítulo 4

Este capítulo apresentou a parte experimental da tese discutindo às montagens e implementações práticas realizadas ao longo do trabalho envolvendo descrição de projetos desenvolvidos dentro e fora da Unicamp. Os exemplos de projetos apontaram para possibilidades de se usar Kits educacionais LEGO e também outros tipos de materiais. Os projetos desenvolvidos procuraram abordar situações concretas similares àquelas que são vividas numa indústria. Tanto na proposta quanto na implementação de pelo menos um dos projetos, procurou-se questionar a pertinência e a análise do mesmo enfocando: motivação para desenvolvê-lo, tópicos interessantes de pesquisa que poderiam ser trabalhados durante o desenvolvimento do mesmo, sua consistência, vantagem do ponto de vista educacional, aspectos de funcionamento prático e conceitos científicos abordados.

Dentre os projetos apresentados ao longo deste capítulo, o desenvolvido pelos técnicos especializados na fábrica apontou para algumas peculiaridades, do ponto de vista educacional que valem a pena ser ressaltadas como, por exemplo, o fato dos técnicos não entenderem, porque a mudança de intensidade de luz captada por um sensor de luz pode ser lida pelo computador e isso podia ser usado para alterar o fluxo de execução de um programa. Constatou-se no entanto, que, a não compreensão advinha da falta de conhecimento científico que fundamentasse a experiência prática e que naquele momento fazia-se necessário realizar uma pausa, estudar os conceitos de outras áreas de conhecimentos para depois solucionar um problema específico de automação com o qual eles estavam se deparando. Isso denotou, mais uma vez, o contexto interdisciplinar que se pode criar nessa forma de trabalho com projetos. Este capítulo apontou para evidências de que a proposta de investigação adotada na tese, enfocando a implementação de dispositivos mecânicos automatizados como forma de se criar ambientes de ensino-aprendizagem de conceitos, em diferentes níveis de ensino, tem se mostrado coerente. Além disso, para reforçar/confirmar a viabilidade desta proposta de investigação, no próximo capítulo analisa-se reflexões realizadas

em trabalhos de campo desenvolvidos com alunos das engenharias mecânica e mecatrônica e engenharia e ciências da computação.



Capítulo 5

Proposta de disciplina de formação na área de Automação

Neste capítulo serão apresentadas duas abordagens que foram realizadas ao longo da pesquisa de campo desenvolvida junto aos alunos dos cursos de Engenharia Mecânica, Engenharia de Controle e Automação (Mecatrônica) e Engenharia e Ciências da Computação. As atividades desenvolvidas com os alunos das Engenharias eram atividades práticas relativas as disciplinas de Introdução à Robótica e Introdução à Engenharia de Controle e Automação, respectivamente. A proposta de trabalho prático em qualquer uma das duas disciplinas enfocava a utilização de kits educacionais LEGO, manipulador robótico Robix associado a outros materiais de padrão comercial ou não. Esses materiais eram utilizados para desenvolvimento de projetos básicos que possibilitavam, de forma prática, que os alunos trabalhassem conceitos como Sequential Flow Chart – SFC (Grafcet), projetos robóticos integrados, sistemas automatizados de produção, dentre outros.

Com relação aos alunos de Engenharia e Ciências da Computação as atividades desenvolvidas podem ser consideradas de um nível mais avançado, envolvendo alunos de Graduação e de Pós-graduação. As atividades práticas desenvolvidas, somente com LEGO, objetivaram a utilização desse material para auxiliar no aprendizado de conceitos de Percepção, Visão Robótica e Inteligência.

5.1 Trabalho desenvolvido nas Engenharias Mecânica e Mecatrônica

Diferentemente dos projetos desenvolvidos pela disciplina da computação, os projetos dos alunos das Engenharias que serão apresentados foram projetos que nasciam de trabalhos em grupo realizados nas aulas práticas e que acabavam sendo concluídos individualmente em função da dinâmica adotada para as aulas práticas.

Essa dinâmica consistia basicamente na divisão dos alunos em duas turmas de aproximadamente 20 alunos sendo que um grupo trabalhava com LEGO e o outro com Robix e se revezavam ao final de quatro encontros de aproximadamente 3:00 horas cada um num total de 12 horas-aula. Concomitantemente, a esses alunos eram ministradas aulas teóricas totalizando em um curso semestral de 130 horas-aula. Uma síntese de como isso era realizado pode-se ver na tabela 5.1 a seguir. Essa etapa inicial era fundamental para o aprendizado de utilização das possibilidades de cada uma dessas ferramentas.

Esboço do cronograma de aulas para os alunos de Mecatrônica proposta para um curso semestral de 130 horas-aula.

Divisão dos alunos em duas turmas A e B.

Divisão das duas turmas em quatro grupos: A1, A2, B1 e B2.

Tabela 5.1 - Quadro contendo divisão das Turmas

| Data | Turma | Grupo | Atividade |
|------|-------|--------------------|---|
| 1. | A | A1 e A2 | Apresentação do LEGO/Robolab e Mindstorms |
| 2. | A | A1 | Apresentação do ambiente LEGO/Robolab Familiarização com componentes mecânicos e elétricos (motores, sensores e atuadores LEGO). |
| 3. | A | A1 | Desenvolvimento de montagem de dispositivos mecânicos no ambiente LEGO/Robolab. Proposta de pequeno projeto para ser desenvolvido. |
| 4. | A | A2 | Apresentação do ambiente LEGO/Robolab Familiarização com componentes mecânicos e elétricos (motores, sensores e atuadores LEGO). |
| 5. | A | A2 | Desenvolvimento de montagem de dispositivos mecânicos no ambiente LEGO/Robolab. Proposta de pequeno projeto para ser desenvolvido. |
| 6. | B | B1 e B2 | Apresentação do LEGO/Robolab e Mindstorms |
| 7. | B | B1 | Apresentação do ambiente LEGO/Robolab Familiarização com componentes mecânicos e elétricos (motores, sensores e atuadores LEGO). |
| 8. | B | B1 | Desenvolvimento de montagem de dispositivos mecânicos no ambiente LEGO/Robolab. Proposta de pequeno projeto para ser desenvolvido. |
| 9. | B | B2 | Apresentação do ambiente LEGO/Robolab Familiarização com componentes mecânicos e elétricos (motores, sensores e atuadores LEGO). |
| 10. | B | B1 | Desenvolvimento de montagem de dispositivos mecânicos no ambiente LEGO/Robolab. Proposta de pequeno projeto para ser desenvolvido. |
| 11. | A e B | A1, A2, B1 e B2 | Desenvolvimento de projeto |
| 12. | A e B | A1, A2, B1 e B2 | Desenvolvimento de projeto |
| 13. | A e B | A1, A2, B1 e B2 | Desenvolvimento de projeto |
| 14. | A e B | A1, A2, B1 e B2 | Apresentação do projeto (encerramento do semestre) |
| 15. | A e B | A1, A2, B1 e B2 | Apresentação do projeto (encerramento do semestre) |

Ao final dessa etapa inicial, conforme o quadro, as turmas se agrupavam a fim de desenvolver projetos mais complexos utilizando-se dos conceitos aprendidos nas aulas teóricas e práticas. É importante ressaltar que mesmo após o final do semestre, quando as atividades práticas não mais contribuíam para a avaliação final da disciplina, alguns alunos optavam em continuar trabalhando nos projetos com o objetivo de aprofundar seus conhecimentos acabando muitas vezes por se constituir em projetos individuais. A seguir serão apresentados alguns projetos desenvolvidos pelos alunos.

5.1.1 Exemplo de projetos desenvolvidos com alunos da Engenharia Mecânica

Os braços robóticos educacionais Robix são construídos a partir de servos usados normalmente em modelos radiocontrolados de carros ou aviões. São vendidos na forma de kits, e sua construção é modular, de modo que sua configuração pode ser modificada conforme a necessidade.

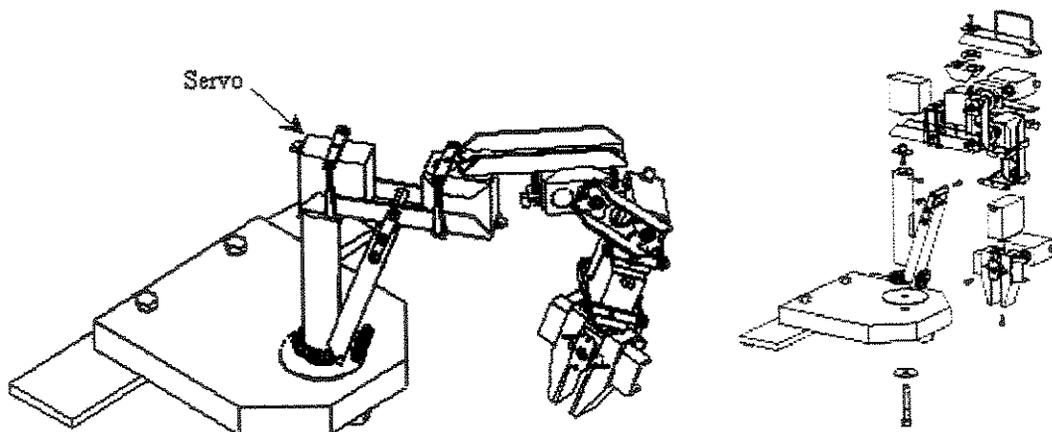


Figura 5.1 - Parte operativa do Robix, montada na configuração de braço robótico(esq.) e o esquema de montagem desta configuração, mostrando as partes do sistema (à direita).

Em diversos projetos desenvolvidos com alunos, Robix foi utilizado e, em algumas situações, este dispositivo foi representado em diferentes configurações objetivando com que os alunos entendessem as várias formas de utilizá-lo.

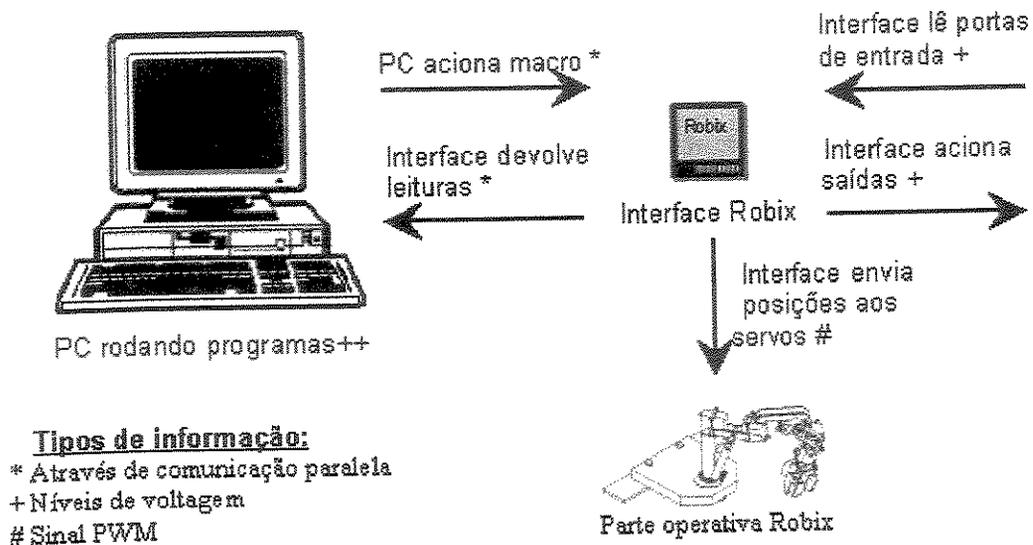


Figura 5.2 - Desenho esquemático da comunicação envolvendo o Robix e o PC.

A figura 5.2 - representa os tipos de sinais, níveis de voltagem e o percurso da comunicação estabelecida entre o microcomputador e o Robix. Essa comunicação pode ser dividida em três partes:

- Parte Operativa do dispositivo mecatrônico Robix;
- Parte Comando que aciona as macros que ativam os servo-motores do Robix;
- Parte Comando que executa os programas em C++.

As três partes são implementadas/construídas e programadas separadamente e finalmente integradas.

Na figura 5.3 o Robix está retirando uma peça da esteira de alimentação para colocá-la na mesa giratória (mais à direita) e em seguida no depósito de peças prontas. Junto à base do Robix, está sua interface, ligada à porta paralela do microcomputador e ao fundo, está a interface LEGO-Dacta, ligada a uma placa no barramento do microcomputador.

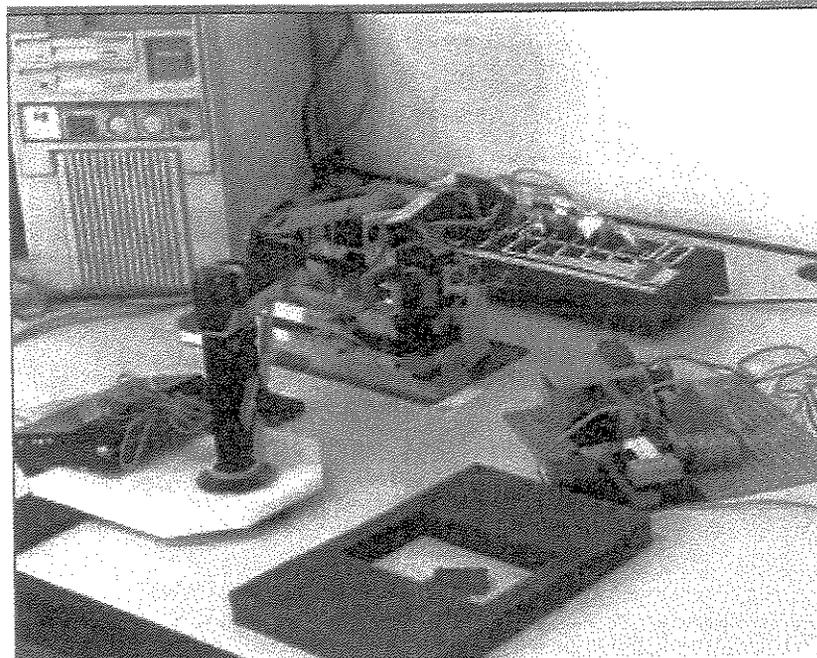


Figura 5.3 - Simulação de um sistema automatizado de produção

Projeto Tratamento de Peças

Na figura 5.4 tem-se à esquerda, a montagem LEGO simulando uma esteira transportadora e um forno de tratamento térmico.

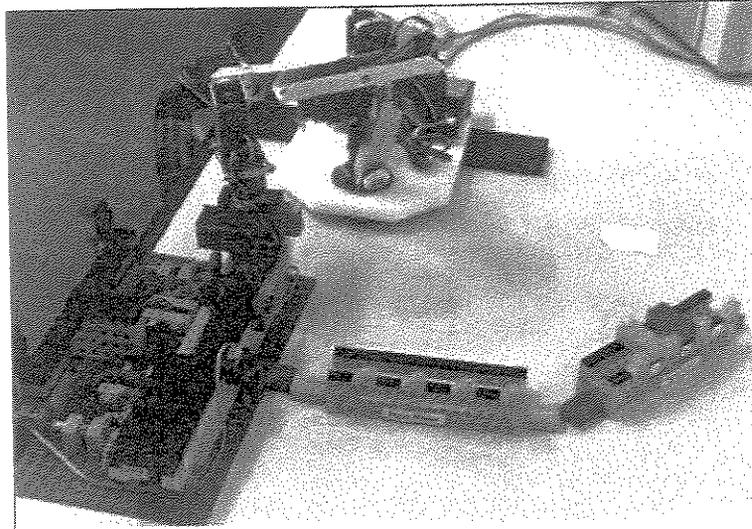


Figura 5.4 - Colocação de Peças em posições previamente definidas

Funcionamento

A operação se inicia com um operador colocando peças não trabalhadas, em um suporte contendo dois compartimentos (um para a peça a ser trabalhada e outro para peça trabalhada). A peça não trabalhada é retirada pelo manipulador robótico Robix e colocada sobre a esteira que a transporta para o forno. Uma vez no forno, a esteira é desligada e a peça permanece por 5 segundos no forno. Passados este tempo, o motor da esteira é acionado no sentido contrário permitindo com que a peça seja transportada até uma determinada posição especificada. Nesta posição, a peça é retirada da esteira pelo Robix e colocada no suporte na posição reservada para peças trabalhadas.

O suporte para as peças é fixado de modo que, alterando a posição do servo 0 (que faz a rotação do braço em relação à base), é possível usar a mesma programação para pegar ou depositar peças em um suporte ou no outro. Isso aponta para a idéia de modularidade que é uma prática na elaboração de programas em automação que permite elaborar programas estruturados de fácil depuração.

Este projeto foi desenvolvido para a participação na feira de informática e tecnologia Educar'98 que se realizou durante o Congresso de Informática na Educação em São Paulo. Este projeto demonstra a possibilidade de se usar kits educacionais para implementar um projeto de caráter industrial em um laboratório na universidade, propiciando aos alunos e professores vivenciarem situações reais de uma indústria.

As Figuras 5.3 e 5.4 mostram exemplos de projetos integrando LEGO e Robix desenvolvido pelos alunos. No capítulo 4 item 4.3 e 4.4 foram descritos outros projetos desenvolvidos com a mesma finalidade.

5.1.2 Exemplo de projeto desenvolvido com alunos da Engenharia Mecatrônica

Traçador Gráfico

Introdução

Para determinadas situações pode ser interessante utilizar um outro meio para reproduzir o desenho feito na tela do computador utilizando um dispositivo robótico. Um traçador X Y pode ser usado com esta finalidade.

Objetivo

Criar um traçador gráfico que reproduz no papel os traços produzidos na tela pela Tartaruga do SuperLogo. O desenvolvimento deste projeto tinha também como objetivo reproduzir em LEGO o traçador gráfico educacional descrito no capítulo 2, ambiente SuperI item 2.8.5.

Descrição

O Traçador X Y é um dispositivo robótico cujo sistema mecânico é constituído por componentes LEGO™ (jogos de engrenagens, rosca sem-fim, cremalheiras, motores e tijolos comuns) conforme ilustra a figura 5.5. Além disso, uma caneta é utilizada para a impressão.

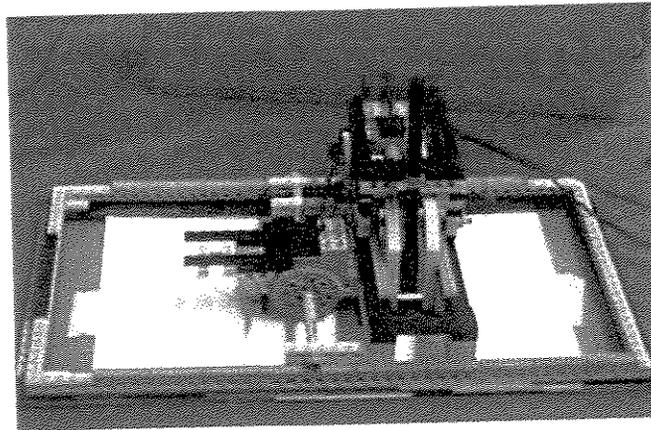
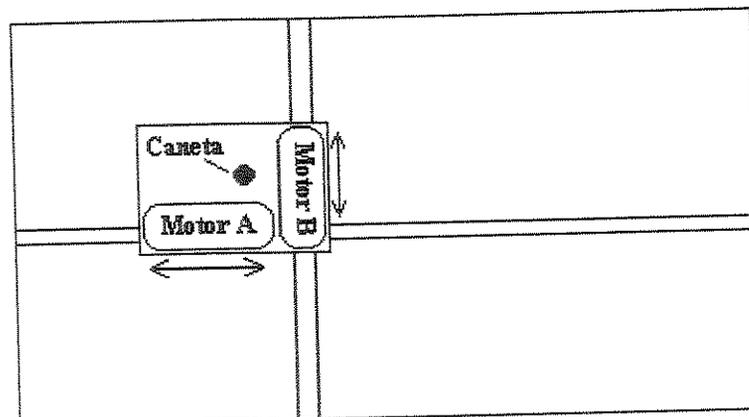


Figura 5.5 – Traçador Gráfico

Montagem

Para a montagem deste dispositivo, além do RCX propriamente dito, foram utilizados como componentes elétricos 2 motores de 9 volts.

Esquema da montagem



A conexão dos componentes elétricos as entradas e saídas do RCX são a seguinte:

ENTRADAS

| |
|-------------------------------|
| Não foram utilizados sensores |
|-------------------------------|

SAÍDAS

| | Saída do RCX | Função |
|---------|--------------|---|
| Motor A | Saída A | Movimentar a caneta no sentido horizontal no plano do papel |
| Motor B | Saída B | Movimentar a caneta no sentido vertical no plano do papel |

Funcionamento

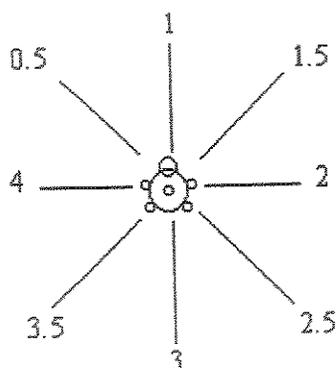
O traçador gráfico, por meio do comando da Tartaruga no SuperLogo faz a impressão, no papel, da trajetória realizada por ela. Foi criada uma variável denominada direção que relaciona a cada direção e sentido da Tartaruga um número. Como foram utilizados motores LEGO, que são de corrente contínua (cc), foi possível apenas obter trajetórias retas e perpendiculares entre si. Com este dispositivo é possível também obter trajetórias com ângulo de 45°, uma vez que a velocidade de deslocamento horizontal e o vertical são praticamente iguais.

Para relacionar a direção e o sentido da Tartaruga com a linha que o traçador deveria riscar, foi utilizado do seguinte recurso:

- a cada giro de 45° para direita da Tartaruga, era somado 0.5 a uma variável do programa escrito em SuperLogo.
- a cada giro de 45° para esquerda da Tartaruga, era subtraído 0.5 a essa mesma variável.
- a variável poderia no mínimo assumir o valor 0.5 e no máximo 4. Caso a Tartaruga estivesse no sentido 4 e girasse 45° para direita, a variável é atualizada com o valor 0.5.

Analogamente, se a Tartaruga estivesse no sentido 0.5 e girasse 45° para esquerda, a variável seria atualizada com o valor 4.

A relação entre a direção e sentido da Tartaruga pode ser interpretada conforme a figura a seguir.



Desta forma, foi possível diferenciar os movimentos nas "diagonais", em que os motores são ligados e desligados simultaneamente, dos movimentos horizontais e verticais, em que somente um motor é acionado.

Os movimentos nas direções dos números inteiros estão relacionados aos movimentos gerados por apenas um motor, enquanto que os dos números não inteiros aos movimentos gerados por dois motores.

Controle do dispositivo

Um possível programa para controle deste dispositivo foi escrito em SuperLogo. Os procedimentos utilizados serão apresentados a seguir.

| Comando | Função |
|-----------------|---|
| Iniciar | Inicia a variável de controle da direção da tartaruga |
| Pdireita90 | Tartaruga gira em torno do próprio eixo 90° para direita |
| Pesquerda90 | Tartaruga gira em torno do próprio eixo 90° para esquerda |
| Pdireita45 | Tartaruga gira em torno do próprio eixo 45° para direita |
| Pesquerda45 | Tartaruga gira em torno do próprio eixo 45° para esquerda |
| Pfrente [valor] | Tartaruga se movimenta na tela do SuperLogo e o plotter imprime no papel a trajetória equivalente à realizada pela tartaruga. |

Procedimentos em SuperLogo

```
aprenda iniciar  
atr "direcao 1  
fim
```

```
aprenda pdireita45  
Se :direcao<5 [atr "direcao :direcao+0.5]  
Se :direcao=4.5 [atr "direcao 0.5]  
pd 45  
fim
```

```
aprenda pdireita90  
Se :direcao<5 [atr "direcao :direcao+1]  
Se :direcao=5 [atr "direcao 1]  
pd 90  
fim
```

```
aprenda pesquerda45  
Se :direcao>0 [atr "direcao :direcao-0.5]  
Se :direcao=0 [atr "direcao 4]  
pe 45  
fim
```

```
aprenda pesquerda90  
Se :direcao>0 [atr "direcao :direcao-1]  
Se :direcao=0 [atr "direcao 4]  
pe 90  
fim
```

```
aprenda pfrente :unid
```

```
atr "tempo 21
pf:unid
Se :direcao=0.5 [
  rcxmotorfrente [a]
  rcxrevertemotor [b]
  rcxligamotor [a b]
  Espere :tempo*:unid
  rcxdesligamotor [a b]
]
Se :direcao=1 [
  rcxrevertemotor [b]
  rcxligamotor [b]
  Espere :tempo*:unid
  rcxdesligamotor [b]
]
Se :direcao=1.5 [
  rcxrevertemotor [a b]
  rcxligamotor [a b]
  Espere :tempo*:unid
  rcxdesligamotor [a b]
]
Se :direcao=2 [
  rcxrevertemotor [a]
  rcxligamotor [a]
  Espere :tempo*:unid
  rcxdesligamotor [a]
]
Se :direcao=2.5 [
  rcxmotorfrente [b]
  rcxrevertemotor [a]
  rcxligamotor [a b]
```

```

    Espere :tempo*:unid
    rcxdesligamotor [a b]
]
Se :direcao=3 [
    rcxmotorfrente [b]
    rcxligamotor [b]
    Espere :tempo*:unid
    rcxdesligamotor [b]
]
Se :direcao=3.5 [
    rcxmotorfrente [a b]
    rcxligamotor [a b]
    Espere :tempo*:unid
    rcxdesligamotor [a b]
]
Se :direcao=4 [
    rcxmotorfrente [a]
    rcxligamotor [a]
    Espere :tempo*:unid
    rcxdesligamotor [a]
]
fim

```

Um outro projeto desenvolvido pelo aluno encontra-se descrito no anexo 3 (**SIROS e RCX**).

5.1.3 Reflexão sobre os projetos desenvolvidos nas Engenharias

Uma reflexão sobre os projetos desenvolvidos nas engenharias, com base nos relatórios elaborados pelos alunos e nas atividades desenvolvidas ao longo de 4 anos, pode-se afirmar que

para alguns alunos esse trabalho propiciou mudanças com relação à forma como eles aprendem. Nesse contexto, alguns aspectos podem ser citados tais como:

1. Possibilidade de desenvolvimento de projetos muito próximos a situações reais da fábrica suscitando no aluno o interesse pelo aspecto profissional futuro;
2. Condições para se desenvolver projetos que primam pela qualidade. Laboratório equipado que permite o acesso do aluno conforme a sua disponibilidade de tempo, necessidade e interesse propiciam desenvolvimento de bons projetos;
3. Aquisição de postura de pesquisador. O aluno começa a adquirir postura de pesquisador na medida em que se envolve com um projeto real que aborda de forma abrangente conceitos de engenharia de forma interdisciplinar (ver exemplo do projeto de iniciação científica no capítulo 4 item 4.4).
4. Autonomia em desenvolver projetos individuais ou em grupo independentemente da avaliação do professor;
5. Continuidade dos projetos. O trabalho não se esgota em apenas 1 semestre. Os alunos que sentem a necessidade de aprofundamento no conhecimento que ele estão adquirindo, continuam a trabalhar sozinhos no projeto no semestre seguinte, fora da carga horária da disciplina, do controle do professor, ou seja independente de aulas;
6. Trabalho em equipe: integração aluno-aluno, professor-aluno, aluno-direção, enfim formação de uma verdadeira comunidade educacional.

Os aspectos descritos anteriormente não se referem somente aos exemplos de projetos apresentados nos itens 5.1.1 e 5.1.2 mas, também aos outros projetos desenvolvidos pelos alunos e que foram descritos no capítulo 4, item 4.3 e 4.4. Entretanto, há que se aprimorar os trabalhos já iniciados para que tudo isso beneficie e integre, ano após ano, um número maior de alunos.

5.2 Análise do Trabalho desenvolvido em Ciências e Engenharia de Computação

Acompanhar de forma sistematizada o trabalho desenvolvido por uma turma durante um semestre significa negociar com os alunos, logo no início do semestre, alguns critérios sobre como esse trabalho deveria ser realizado. Ou seja, definir metas para serem atingidas e compromissos para as partes tanto dos professores quanto dos alunos. Para se estudar as estratégias que esses alunos utilizam para resolver problemas e como acontece o aprendizado de conceitos, exige-se dentre outras tarefas, ministrar aulas e interagir com os alunos de forma individual, em pequenos grupos e também coletivamente. A partir daí, no processo, coletar, analisar e tratar informações utilizando principalmente apontamentos de cadernos de alunos. Essas informações uma vez organizadas e tratadas, evidenciam um conjunto de atributos e características que ajudam a explicitar fenômenos presentes no processo de aprendizagem desses alunos. Esta parte da tese apresenta a análise do trabalho de pesquisa de campo realizado juntos aos alunos do Instituto de Computação da Unicamp, durante o segundo semestre letivo de 2001, ministrando aulas na disciplina MO-810/MC-959 (Robótica: Sistemas Sensorial e Motor). Esta análise, qualitativa de dados multicategóricos, descreve o ponto de vista do pesquisador sobre as atividades desenvolvidas pelos alunos, baseado na leitura dos apontamentos que cada aluno fez em seu caderno individual utilizado nas aulas práticas de montagem de dispositivos mecânicos. Além disso, foram também utilizadas as páginas de Internet criadas por cada um dos grupos. A análise consiste da verificação de 25 cadernos de uma turma de 25 alunos onde 100% dos cadernos foram analisados num contexto em que a turma foi dividida em 6 grupos em função das afinidades e relacionamento entre os alunos. O caderno era o material de bancada que funcionava como um portfólio individual que o aluno levava para todas as aulas práticas a fim de anotar as etapas, as dúvidas, as sugestões/hipóteses suas e/ou de outros colegas, as implementações e experimentações, enfim todos os momentos que considerassem importante durante o processo de desenvolvimento dos projetos no seu grupo.

Os projetos de robôs desenvolvidos pelos alunos nas aulas práticas da disciplina MO-810/MC-959 consistia de um trabalho de grupo onde o que se pretendia era a construção de robôs. Todos os alunos receberam mesmo tipo de material (01 Kit LEGO 9790) para desenvolver projetos onde os robôs deveriam executar tarefas específicas. Embora trabalhando em grupo,

observou-se que cada aluno tinha a sua opinião pessoal, que nem sempre convergia com a do colega do mesmo grupo, sobre as implementações em um determinado projeto. A partir daí, será descrita uma análise sobre a forma como esses alunos trabalharam, tipos de conceitos científicos utilizados, estratégias para resolução de problemas empregadas, dentre outros aspectos levantados nos cadernos de cada um deles. Os aspectos identificados, evidenciados nos cadernos, são dados empíricos cuja organização e interpretação originou a construção de categorias de análise (Molina Pagnez, 2001) que serão mostradas na tabela 5.2 e descritas posteriormente a partir do item 5.3. A seguir, será apresentada a disciplina MO-810/MC-959.

5.3 Apresentação da disciplina

A proposta de se desenvolver as atividades práticas da disciplina teve, desde o início a preocupação de se criar um ambiente mais propício possível para que os alunos desenvolvessem os seus projetos. Uma das condições criadas para isso foi a distribuição logo no início do curso de um Kit LEGO 9790 que continham 01 tijolo programável RCX, 02 motores, 02 sensores de toque, 01 sensor de luz e entrono de 700 peças diversas que possibilitam montar mecanismos. Cada grupo de alunos recebeu um kit que ficou sob a responsabilidade do grupo até o final do semestre. Isso possibilitou com que as montagens não acontecessem somente durante os horários estipulados para as aulas, mas sim, de forma flexível, em dias e horas mais convenientes para cada grupo. Todavia, existiram dias e horários específicos em que aconteceram as aulas tanto teóricas quanto práticas com a presença dos professores. Cada grupo implementou três diferentes projetos de robôs. No final do primeiro e do segundo projetos foi realizado uma apresentação na forma de seminário na sala de aula. Já no final do terceiro e último, projeto foi realizada uma apresentação na forma de competição, onde estiveram presentes os professores e alunos da turma, outros professores e outros alunos do Instituto de Computação, e também repórteres do Jornal da Unicamp que registraram o evento e, que foi posteriormente publicado sob o título “*Inteligência Artificial: Alunos fazem exercícios com Brinquedos LEGO para Aplicar Conceitos Sofisticados de Robótica*” (Jornal da Unicamp nº 170 fevereiro de 2002).

A fim de contextualizar como foi este processo, apresenta-se a seguir, de forma sucinta, alguns tópicos da estruturação do curso (retiradas da página Internet “<http://www.dcc.unicamp.br/~lmarcos/courses/mo810/>”), que fizeram parte de um elenco de informações transmitidas aos alunos no início do semestre e que estavam disponibilizadas no *site* do curso. No anexo 5 (**Informações sobre o curso MO-810/MC-959**) apresenta-se essas informações de forma mais detalhada.

5.3.1 Objetivos

A disciplina (Robótica: Sistemas Sensorial e Motor) tinha como objetivo abordar os princípios fundamentais da robótica incluindo percepção.

5.3.2 Pré-requisitos

Para se matricular na disciplina o aluno deveria ter conhecimentos de álgebra linear, cálculo vetorial e física básica (especialmente mecânica clássica). Além disso, conhecimentos da linguagem de programação C (ou C++).

5.3.3 Avaliação do Aprendizado

A avaliação do aprendizado foi feita por meio de listas de exercícios, trabalhos práticos, seminários de apresentação de projetos, uma prova e um projeto final.

5.3.4 Trabalhos Práticos

A finalidade dos trabalhos práticos da disciplina era de se construir conhecimentos referentes a cada um dos aspectos de um projeto maior, que seria o desenvolvimento de um robô móvel capaz de executar com sucesso tarefas previamente especificadas. Cada detalhe, idéia, desde a concepção até a implementação deveria constar no caderno de laboratório de cada aluno. Os trabalhos práticos foram projetados objetivando a preparação para a competição final, que consistia da integração dos diversos subsistemas de hardware (estrutura mecânica, sensores,

atuadores) e software (controle do robô). Os trabalhos práticos foram realizados e apresentados ao grupo conforme instruções constantes na página do curso.

5.3.5 Caderno de Laboratório

Era recomendado que cada aluno tivesse o seu próprio caderno. Nele deveriam ser anotadas idéias de mecanismos, estratégias, algoritmos, etc.. No caderno poderiam ter também figuras, fotos, gráficos e tabelas obtidas de calibração de sensores, etc..

5.3.6 Seminário

Ao longo da disciplina foram preparados e apresentados, pelos alunos, seminários onde cada grupo discorreu sobre seu projeto, dificuldades, resultados esperados, resultados alcançados, etc.. No final do terceiro e último projeto foi realizado uma apresentação na forma de competição.

5.3.7 Competição

Desde o início do semestre foi definido que haveria uma competição onde seria apresentado o projeto final. O projeto consistiria na construção de um robô móvel autônomo. A estrutura mecânica dos robôs seria implementada com componentes da LEGO.

5.3.8 Página WWW

Fazia parte da atividade da disciplina a construção por cada grupo de alunos de uma página Internet. Esta página deveria conter a documentação do projeto do grupo. As atividades desenvolvidas pelo grupo durante a semana deveriam estar disponíveis a cada segunda-feira pela manhã. Cada grupo deveria ter um nome, pelo qual o seu robô também seria conhecido. Toda a evolução do projeto deveria ser documentada com fotos digitais para serem incorporadas na página do grupo.

5.3.9 Aulas Teóricas

As aulas teóricas consistiam de aulas expositivas dialogadas onde o aspecto fundamental era a abordagem dos conceitos teóricos de Percepção e Visão Robótica, Inteligência Artificial (IA), Teoria de Controle. O objetivo dessas aulas era apresentar os conceitos inerentes a esses campos do conhecimento interrelacionando-os com o desenvolvimento dos projetos propostos.

5.3.10 Aulas Práticas

As aulas práticas consistiam de montagem, automação e controle de dispositivos robóticos. Nessas aulas os conceitos teóricos apresentados em classe, de forma expositiva, eram trabalhados/utilizados na prática pelos alunos.

Montagem de dispositivos significava utilizar peças mecânicas tais como rosca sem-fim, engrenagens, eixos, cremalheiras, correias dentadas etc., para montar estruturas mecânicas. Estas peças devidamente acopladas, respeitando alguns princípios mecânicos, e utilizando-se de componentes elétricos como motores e sensores possibilitavam com que os robôs se movimentassem.

Automação e Controle dos dispositivos significava elaborar programas utilizando software *Robolab* e *Not Quite C – NQC*. Estes softwares foram desenvolvidos por pessoas e/ou empresas para controlar o tijolo programável, *Robotic Command Explorer (RCX)* da LEGO. O que se esperava era que, ao funcionar, o dispositivo se comportasse conforme a descrição elaborada no programa e realizasse a tarefa desejada, tais como, não se chocar contra uma parede, evitar a luz, não cair da mesa, entrar e sair de um labirinto, dentre outras. No anexo 6 (**Programas em NQC**) encontram-se alguns algoritmos de programas elaborados pelos alunos em NQC.

Automatizar um dispositivo robótico utilizando o RCX significa descrever para o computador, em termos de uma seqüência lógica, os comandos que acionam motores e sensores conectados ao dispositivo. Os programas são elaborados no computador e transferidos para o RCX, conectado àquele dispositivo robótico, para serem executados. Toda execução do programa é realizada somente no RCX.

5.4 Levantando Dados Multicategóricos

A análise de dados, feita a partir dos cadernos dos alunos permitiu comprovar que os dados eram de natureza multicategórica, que se inter-relacionavam entre si em maior ou menor graus. O estudo fundamentou-se na definição de categoria, com base em Molina Pagnez, como sendo *“um grupo com um conjunto de atributos ou características comuns, que tem poder de explicitar uma grande quantidade de fenômenos presentes no processo (...)”* (Molina Pagnez, 2001, p 75). Para analisar os cadernos foi feita uma leitura prévia dos mesmos e, em função desta leitura, foram agrupados os aspectos considerados mais relevantes.

O critério que foi utilizado para se agrupar os aspectos observados em categorias baseou-se em eleger como aspectos representativos aqueles que foram descritos/citados por no mínimo 80% dos alunos. Foram definidas e analisadas seis categorias tais como: Diferentes heurísticas para solucionar o problema; Utilização de conceitos de outras disciplinas; Solução por software, Cooperação entre os colegas; Denominação dos robôs; Aproveitando idéias. Essas categorias foram tabuladas e explicitadas na Tabela 5.2, e serão analisadas a seguir.

Nesta tabela a marca com “x” na horizontal representa o aspecto presente na categoria. A seta na vertical representa o aspecto contido na categoria. As setas demonstram que uma categoria pode ser comum a vários aspectos. As marcas com “x” demonstram que aspecto pode estar presente em mais de uma categoria.

Tabela 5.2 – Representação de Dados Multicategóricos

| Aspectos Levantados / Categorias | 1 Diferentes Heurísticas para solucionar o problema | 2 Utilização de conceitos de outras disciplinas | 3 Solução por software | 4 Cooperação entre os colegas | 5 Identificação dos robôs | 6 Aproveitando idéias |
|---|--|--|-------------------------------|------------------------------------|-------------------------------|-------------------------------|
| Elaboração de desenhos/esquemas | X | | X | | | |
| Utilização de idéias do projeto anterior em projetos subsequentes | | | | X | | X |
| Elaboração de protótipo | X | X | | | | |
| Formulando matematicamente o problema | | X | X | | | X |
| Uso de Idéias Computacionais; Sistema de tateamento | X | X | | | | |
| Conceitos de Física | | X | | | | |
| Denominando o grupo e os projetos | | | | X | X | |
| Implementação de algoritmos | | | X | | | |
| Uso de conceitos de estatística: probabilidade | | X | X | | | |
| Montagem de diversas estruturas | X | X | | | | |
| Construção de Diferentes versões de robôs | X | X | X | | X | |
| Definição de estratégias | X | X | | | | X |
| Sugestão e ética no trabalho em equipe | | | | X | | |
| | Grupos: 1, 2, 3, 4, 5, e 7 100% | Grupos: 1, 2, 3, 5, e 7 80% | Grupos: 1, 2, 3, 4 e 7 80% | Grupos: 1, 2, 3, 4, 5, e 7 100% | Grupos: 1, 2, 3, 4 e 5 80% | Grupos: 1, 2, 3, 4 e 5 80% |

A seguir, será analisado e discutido cada categoria e a sua inter-relação com os aspectos levantados.

5.4.1 Categoria 1: Diferentes Heurísticas para Solucionar o Problema

Como pôde ser visualizado na tabela 5.2, ao se defrontarem com os projetos de montagem automação e controle dos robôs os alunos desenvolveram ao longo do curso diferentes estratégias/heurísticas para solucionar os problemas. O nível de complexidade dos problemas

crescia à medida que se passava de um projeto mais simples para outro mais sofisticado. Algumas estratégias utilizadas pelos alunos foram: elaboração de desenhos/esquemas, elaboração de protótipo, montagem de diversas estruturas, construção de diferentes versões de robôs, definição de estratégias. Esses foram os aspectos levantados nos cadernos que produziram a categoria1. A seguir será discutido e analisado cada um deles.

a) Elaboração de desenhos/esquemas

Esta era a estratégia utilizada, muitas vezes pelos alunos, para explicar por meio de desenhos esquemáticos (feitos a mão livre), para o professor e/ou outro colega a idéia da montagem que o grupo pretendia desenvolver como mostram as Figuras 5.6, 5.7, 5.8, e 5.9. Além dos desenhos feitos a mão-livre, alguns grupos utilizaram o software Ldraw (uma ferramenta de modelagem) para especificar e demonstrar a seqüência de montagem dos seus protótipos, como mostrado na Figura 5.10.

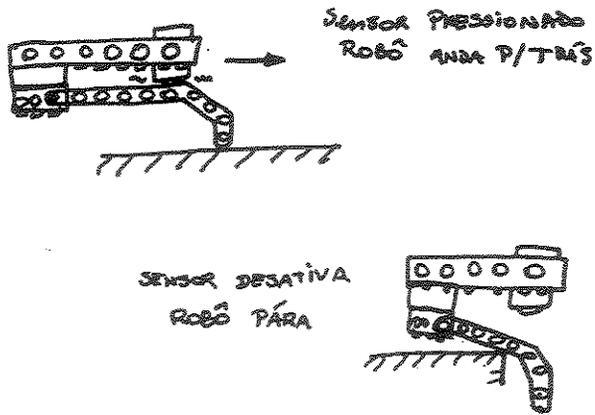


Figura 5.6 – Desenho a mão livre do sistema de garra com sensor de toque

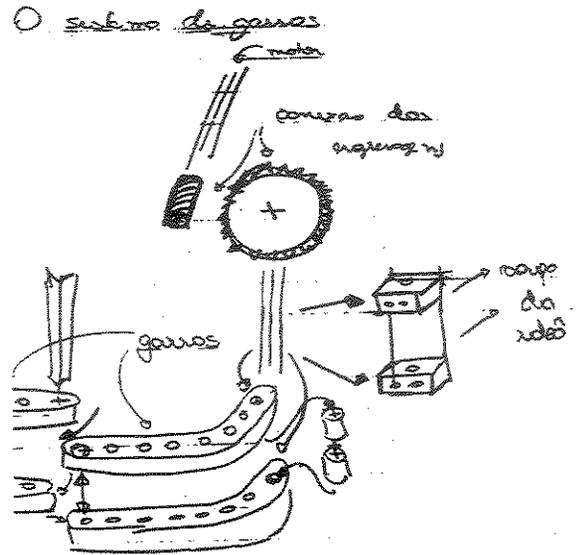


Figura 5.7 – Desenho a mão livre do sistema de garra com engrenagem e rosca sem fim

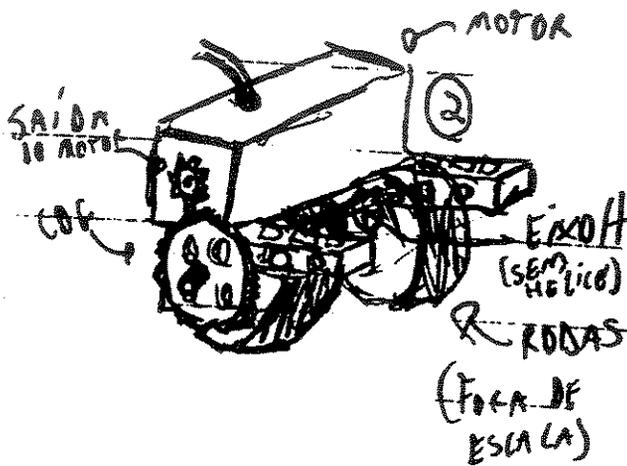


Figura 5.8 – Desenho a mão livre do sistema de direção

Um motor aciona 2 rodas num mesmo sentido?

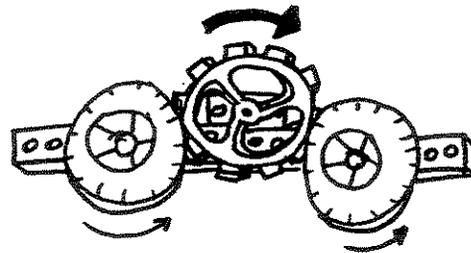


Figura 5.9 – Desenho a mão livre do sistema de engrenagens

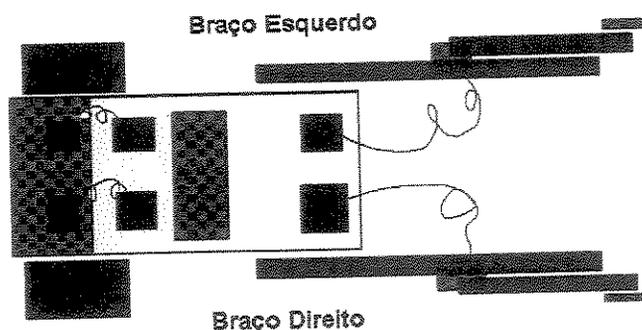


Figura 5.10 – Protótipo de um dispositivo robótico em Ldraw

Os desenhos demonstram, de alguma forma, a antecipação das idéias, dos conceitos que o grupo tinha sobre o dispositivo mecatrônico que pretendiam implementar. Analisando os desenhos dos grupos, sob a ótica de aprendizagem de conceitos, pode-se de certo modo considerar que, num primeiro momento os desenhos dos grupos representavam um desenvolvimento de conceitos espontâneos³⁴ que precisavam alcançar um certo nível de compreensão para que posteriormente os alunos fossem capazes de absorver o conceito científico correlato (Vygotski, 1991).

b) Elaboração de protótipo

Estratégia utilizada pelos alunos consistia na construção de pequenos protótipos de alguns mecanismos que deveriam fazer parte dos robôs. Neste contexto, define-se o mecanismo como sendo uma associação de elementos mecânicos levando em conta algum princípio da Física, da Mecânica, da Eletrônica, etc.. Muitos protótipos eram construídos e testados isoladamente, uma vez funcionando, eles eram incorporados aos robôs. Alguns desses protótipos são: sistemas de redução de velocidade usando jogos de engrenagens, sistemas de acoplamentos de eixos, sistema diferencial/cardin, sistema de

³⁴ Conceito espontâneo é um conceito que é aplicado de forma não-consciente. Ao operar com conceitos espontâneos, a sujeito não está consciente deles, pois a sua atenção está sempre centrada no objeto a qual o conceito se refere, nunca no próprio ato de pensamento (Vygotski, in *Pensamento e Linguagem*, p. 79, 1991)

acionamento de sensores, sistema de garra para o braço robótico, sistema de direção, etc.. As figuras 5.11 e 5.12 mostram dois desses protótipos.

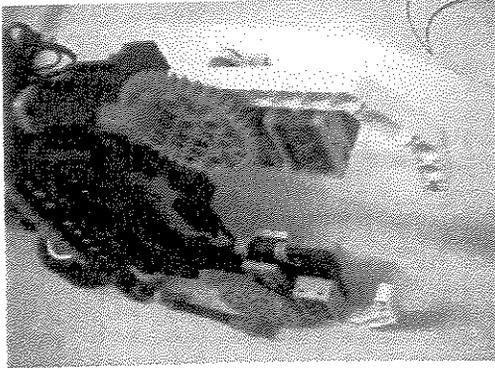


Figura 5.11 – Sistema de Garra

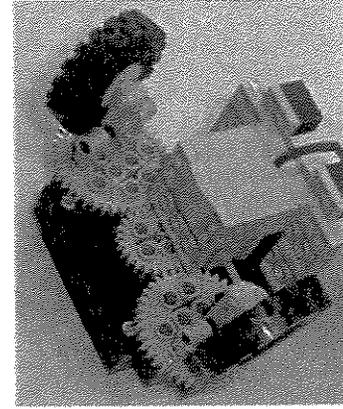


Figura 5.12 – Sistema de direção

c) Montagem de diversas estruturas

Assim como na elaboração de protótipos de mecanismos, diferentes estruturas mecânicas foram montadas pelos alunos, utilizando diferentes tipos de tijolos LEGO. Nesse contexto, considerou-se como estruturas mecânicas montagens que têm alguma funcionalidade e que geram, por exemplo, movimentos tais como: andar, girar, levantar abaixar. Inicialmente, as estruturas montadas se desmoronavam facilmente, ao primeiro choque do robô contra qualquer obstáculo, inclusive outro robô. À medida que as estruturas começaram a ser construídas utilizando técnicas de entrelaçamento de tijolos e montagem em forma de treliça, as estruturas tornaram-se mais resistentes e já não se desmoronavam facilmente. Foram construídas e testadas diversas configurações de estruturas abordando questões conceituais de Física, Engenharia Mecânica, Engenharia Civil, Matemática, dentre outras. As Figuras as 5.13 e 5.14 mostram algumas dessas estruturas.

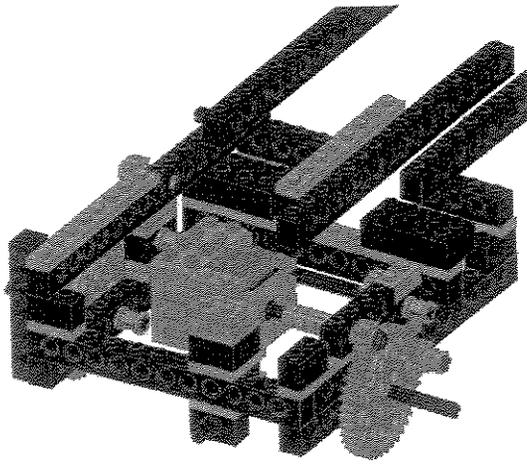


Figura 5.13 – Estrutura resistente contendo um motor e sistema de redução de velocidade por engrenagens.

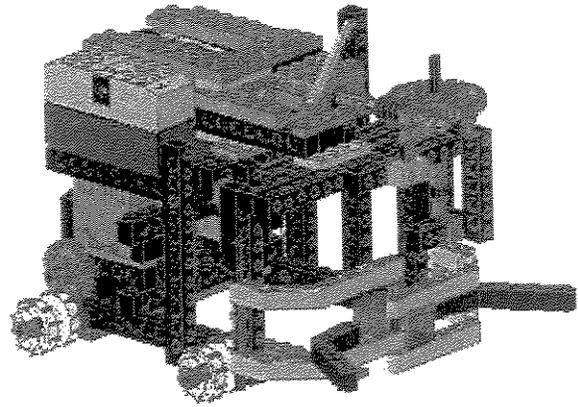


Figura 5.14. - Estrutura resistente contendo um RCX dois motores e sistema de redução de velocidade por engrenagens.

d) Construção de diferentes versões de robôs

Na maioria das vezes os alunos construíam diversas versões do robô, com diferentes denominações, partindo de alguma configuração que já conheciam para depois chegarem no formato definitivo. Este processo envolvia a utilização de conceitos de Física ao determinar a velocidade, Matemática no cálculo da relação de engrenagens, computação na programação dos robôs. A Figura 5.15, por exemplo, mostra a primeira versão do robô Afrodite, utilizando sistema de deslocamento baseado em esteiras. A Figura 5.16 mostra a última versão deste robô utilizando sistema de deslocamento baseado em rodas.

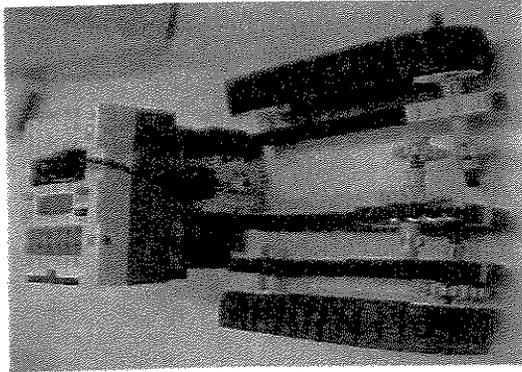


Figura 5.15 – Robô Afrodite com sistema de deslocamento baseado em esteiras

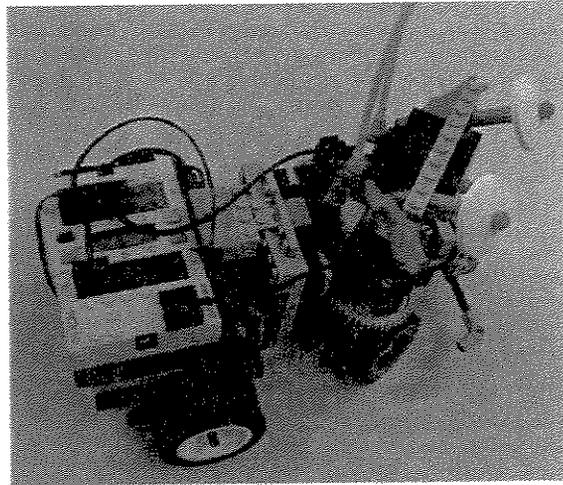


Figura 5.16 – Robô Afrodite com sistema de deslocamento baseado em rodas

e) Definição de estratégias

A grande maioria dos alunos descrevia as estratégias que utilizaram para resolver os problemas encontrados nos projetos e também as propostas de soluções para serem discutidas com os professores. Eram anotadas as diversas idéias que surgiam durante a aula prática. Em seguida essas idéias eram implementadas. Durante o processo de implementação surgiam novas idéias, que muitas vezes, já eram uma depuração/aprimoramento da idéia anterior. Em alguns momentos a definição de estratégias envolvia a utilização de conceitos de outras disciplinas para auxiliar na explicitação de uma determinada idéia a ser implementada. Além disso, a contribuição de outros colegas revelou que existia a socialização de idéias.

Ao analisar a constituição desta categoria (**Diferentes Heurísticas para solucionar o problema**) observou-se que existe uma relação horizontal e outra vertical entre aspectos observados no caderno dos alunos e as categorias constituídas a partir deles, como podemos ver na Tabela 5.2. Entretanto, isso não aconteceu somente com esta categoria, aconteceu com as outras também.

Um exemplo da relação vertical pode ser visto na Tabela 5.2 tomando-se os aspectos **elaboração de protótipo e montagem de diversas estruturas**. Isso se deve ao fato de que para se montar o protótipo necessariamente há que se construir uma estrutura.

Um exemplo da relação horizontal pode ser vista na tabela 5.2 observando-se o aspecto **Construção de diferentes versões de robôs** que contribuiu para a constituição de quatro categorias: **diferentes heurísticas para solucionar o problema, utilização de conceitos de outras disciplinas, solução por software e identificação dos robôs**. Esta relação se deve ao fato de que no processo de construção de diferentes versões de robôs foram utilizados conceitos científicos presentes nestas categorias.

Um exemplo de estratégia que se pode descrever foi na elaboração do programa para o robô que anda sobre uma mesa e não cai, em que o grupo lançou mãos de idéias de Circuitos Lógicos criando uma Tabela de Estados (veja a Tabela 5.3) contendo oito estados possíveis que o robô poderia assumir sobre a mesa. A Tabela de Estados, por sua vez, evoca o uso de conceitos de Eletrônica. A partir desta tabela é possível implementar uma função lógica que possibilita projetar o circuito eletrônico combinacional que em última análise pode ser um Circuito Integrado (CI) passível de ser implantado em um robô.

Tabela 5.3 – Representação de Estados

| | Sensor de Luz | Sensor de Toque 2 | Sensor de Toque 2 | Estado |
|----|---------------|-------------------|-------------------|--------------|
| 1. | 0 | 0 | 0 | Não acontece |
| 2. | 0 | 0 | 1 | Fora da mesa |
| 3. | 0 | 1 | 0 | Fora da mesa |
| 4. | 0 | 1 | 1 | Fora da mesa |
| 5. | 1 | 0 | 0 | Não acontece |
| 6. | 1 | 0 | 1 | Fora da mesa |
| 7. | 1 | 1 | 0 | Fora da mesa |
| 8. | 1 | 1 | 1 | Não acontece |

Nos estados 1, 5 e 8 o robô não tem nenhuma ação.

Nos estados 2, 3 e 4 o robô vira para esquerda até que o sensor de luz fique sobre a mesa.

Nos estados 6 e 7 o robô vira para esquerda até que o sensor de luz saia da mesa.

Outro exemplo de estratégia de um dos grupos para o projeto do robô que não cai da mesa foi o de construir um robô que tinha dois detectores com formato de braços que estão inicialmente acionados por dois sensores de toque (um para cada braço).

Explicação do grupo:

"Quando o robô chega perto da borda da mesa um dos braços ou os dois braços caem, desativando o/os sensor(es) e o robô vira para um determinado lado. A escolha do lado para o qual o robô iria virar foi feita da seguinte forma:

Caso caísse primeiramente o braço direito o robô vira para o lado esquerdo. Caso caísse primeiro o esquerdo ele vira para o lado direito. Porém, se caísse os dois braços ao mesmo tempo ele vira para o último lado que ele havia virado. O último movimento do lado era guardado em uma variável de estado, cujo valor "default" inicial era para direita. (alunos do grupo 1)

5.4.2 Categoria 2: Utilização de conceitos de outras disciplinas

Utilização de conceitos de outras disciplinas neste contexto significa lançar mãos de conteúdos e conceitos de outros campos do conhecimento, que não apenas a Robótica, para solucionar um problema específico de robótica como, por exemplo, utilização de software LEGO DRAW para elaborar e especificar a seqüência de montagem de protótipos. Essa categoria foi constituída pelos aspectos, retirados dos cadernos dos alunos, **Formulando Matematicamente o Problema e Conceitos de Física**. Além destes aspectos, também colaboraram com esta categoria os aspectos: elaboração de desenhos/esquemas, elaboração de protótipo, uso de idéias

computacionais sistemas de tateamento, uso de conceitos de Estatística e Probabilidade, montagem de diversas estruturas, construção de diferentes versões de robôs, definição de estratégias. Ao analisar esta categoria verificou-se que ela praticamente se inter-relaciona com todos os aspectos levantados. O que mais uma vez demonstra a natureza interdisciplinar/multidisciplinar dos projetos desenvolvidos.

a) Formulando Matematicamente o Problema

Em muitas situações os alunos utilizaram recursos matemáticos na busca de solução para um determinado problema, por exemplo, fazer o robô, em movimento, detectar a luz. Neste caso, foi definido um período de tempo (tempo de calibração) e considerado uma velocidade inicial. Sabendo este valor de tempo (t) e o número de voltas que a roda efetua, pode-se calcular a distância percorrida. O número de voltas da roda é contado utilizando-se sensor de rotação. Ou seja, para uma roda de diâmetro (d) e um giro de 90° tem-se à distância percorrida (S) como sendo: $S = \pi d/2$. Uma vez obtida a distância S , pôde-se calcular a velocidade (v) do robô uma vez que $v = S/t$.

Neste exemplo, conceitos de Física e de Matemática foram utilizados para se formalizar um problema, que era calcular a velocidade, numa situação em que o valor da velocidade era um fator determinante para fazer com que o robô detectasse a luz. Em muitos projetos a velocidade foi determinada empiricamente, mas este grupo teve a preocupação de determina-la cientificamente.

Ainda sobre a formulação matemática de um problema, para reduzir a velocidade em um eixo utilizando polias, o aluno propôs:

*"conectando duas polias, mantém-se a velocidade escalar (v) de cada uma, mas não a velocidade angular (α). Como $v = \alpha * R$*

Temos:

$$V1 = V2 \Leftrightarrow \alpha1 * R1 = \alpha2 * R2$$

$$\Leftrightarrow \alpha2 = R1/R2 * \alpha1 \text{ assim, se } R2 > R1, R1/R2 < 1$$

e $\alpha_2 < \alpha_1$ (para velocidade angular α_2 e α_1)

Por outro lado se:

$$\alpha_1 = \alpha_2 \Leftrightarrow V_1/R_1 = V_2/R_2 \Rightarrow$$

$$V_2 = R_2/R_1 * V_1 \text{ e se,}$$

$$R_2 < R_1, R_2/R_1 < 1 \text{ e } V_2 < V_1''$$

(alunos do grupo 4/6)

Assim, combinando um conjunto adequado de polias, teremos a redução necessária de velocidade.

b) Conceitos de Física

A utilização de conceitos de Física foi um dos aspectos que permeou todos os cadernos. Foi muito marcante a utilização de conceitos de Física em praticamente 100% dos cadernos analisados. Em muitas situações os alunos para explicarem as idéias a serem implementadas usavam conceitos como: momento de inércia, peso, gravidade, centro de massa, velocidade, dentre outros. Esses conceitos se inter-relacionam tanto verticalmente, com os aspectos levantados, quanto horizontalmente com as categorias criadas a partir desses aspectos.

c) Uso de Conceitos de Estatística

A utilização de conceitos de Estatística também foi um dos aspectos bastante evidenciados no caderno dos alunos, em situações como, captar 50 amostras de luz para determinar a calibração do sensor de luz. Cálculo tolerância de uma margem de erro a partir dos valores lidos no *display* do RCX. A utilização de todos esses conceitos estava fortemente concatenada com a utilização da programação, mais uma vez, pelo menos duas disciplinas se inter-relacionado. Esses conceitos também se inter-relacionam tanto verticalmente, com os aspectos levantados, quanto horizontalmente com as categorias criadas a partir desses aspectos.

5.4.3 Categoria 3: Solução por software

Um usuário com conhecimento básico de Informática, de Eletrônica e alguma noção de mecânica que era o caso de estudantes de computação, consegue, de forma rápida, sair da situação de usuário comum (aquele que utiliza o ambiente sem se aprofundar/preocupar com os aspectos técnicos de hardware/software) e passar a ser um usuário da área técnica com condições de desenvolver aplicações específicas, em diferentes áreas da Robótica. Isso significa se interessar em usar conhecimentos de hardware e de softwares capazes de aprimorarem as ferramentas originariamente disponibilizadas, para um certo uso limitado, em ferramentas mais diversificadas, sobretudo para uso em atividades de pesquisa acadêmica. Entretanto, para os alunos de computação, observou-se uma grande tendência na busca de solução por software. Por exemplo, numa situação de utilização do sensor de luz, antes de se propor um melhor posicionamento do mesmo, para detectar a luz, foi proposto à elaboração de um algoritmo para teste deste sensor comparando centenas de leituras captadas, ou algo semelhante. Outro exemplo descrito por um aluno:

"...tentativa de redução de velocidade do robô por software (sem perder potência) para que o robô não continue andando pelo efeito do momento de inércia..." A redução de velocidade do robô poderia ser obtida inclusive com aumento de torque se fosse pensada uma solução mecânica utilizando jogos de engrenagens (alunos grupo 7).

Além disso, observou-se a utilização pelos alunos de outros softwares diferentes do software Robolab originariamente apresentado, o que foi uma atitude bastante positiva. Alguns dos outros softwares utilizados foram:

LDRAW - É uma classe de ferramentas para modelagem de robôs Lego (<http://www.ldraw.org/>). Substituem, com vantagem, o uso de câmeras digitais para digitalização das imagens.

NQC - Not Quite C: linguagem de programação para RCX (<http://www.cs.uu.nl/people/markov/lego/>). Muito parecida com linguagem C.

PERL - Uma versão da linguagem Perl que controla o RCX LEGO Mindstorms.

LeJOS - Uma versão da linguagem Java para Lego Mindstorms.

legOS - Uma linguagem fonte desenvolvida para controle RCX Mindstorms usando plataforma Linux, (<http://www.noga.de/legOS/>).

Mais informações sobre os *sites* nesta área e também sobre outros tópicos abordados na tese e encontram-se no anexo 7 (**Alguns *sites* visitados**).

Dentre os vários aspectos em que se constitui esta categoria (veja tabela 5.2) a implementação de algoritmos será descrita por ser o que se considerou de maior relevância.

a) Implementação de Algoritmos

Uma das tarefas que o robô teria que realizar no terceiro projeto era entrar em um determinado espaço reconhecer um amigo e retirá-lo para fora desse espaço. Para isso, uma vez reconhecido e capturado o amigo, o robô deveria ser capaz de saber se ele estava defronte de uma parede e caso estivesse, ele deveria seguir contornando a parede até chegar na porta de saída. Para contornar a parede, um dos grupos desenvolveu o "sistema de tateamento" onde se elaborou um programa que possibilitava ao robô seguir as arestas conferindo periodicamente, por meio de sensor de luz, se ele estava no caminho certo. Para o robô encontrar a saída deste espaço foi sugerido também, além da utilização do sistema de tateamento, a utilização de um algoritmo baseado na memória visual onde a idéia era guardar o caminho percorrido. Como por exemplo, para solucionar o problema de deslocamento do robô para sair de um espaço fechado que só tinha uma porta, o aluno propôs:

"Tentar também fazer um mapeamento do caminho de acordo com o movimento e o tempo dos motores para fazer o caminho reverso para voltar e sair do cercado. Mas

se ele andar muito talvez não haja memória. Pode-se pensar usar uma pilha na memória para guardar o caminho, assim é fácil desfazê-lo."

Muitos dos algoritmos desenvolvidos pelos alunos, neste projeto, se baseavam em fazer do robô um Seguidor de Arestas. A idéia do Seguidor de Arestas consiste em fazer o robô, por intermédio de sensor de luz, procurar buraco enquanto ele estiver "enxergando" a mesa e procurar a mesa enquanto ele estiver "enxergando" o buraco. Com isso, o robô é capaz de se deslocar sem cair da mesa. Um pseudo - Código elaborado para o seguidor de arestas é o seguinte

- Gira 360° e pega a maior e a menor intensidade de luz
- Enquanto não há buraco na frente
- Anda o suficiente para as rodas traseiras ficarem na posição das dianteiras
- Pára
- Verifica se há buraco
- Gira para direita até achar a mesa (*)
- Anda
- Pára
- Gira para esquerda até achar buraco
- Volta para (*)

Estas descrições acima mostram alguns conceitos computacionais associados às estratégias de resolução de problemas utilizadas pelos alunos.

5.4.4 Categoria 4: Cooperação entre os colegas

A cooperação entre os alunos foi uma das categorias que se evidenciou durante a análise dos cadernos. Pôde-se observar que existia por parte dos alunos, desde o início, uma preocupação em formar grupos com os colegas com os quais tinham mais afinidade. Houve situações em que algumas idéias utilizadas, sem sucesso por um determinado grupo não eram utilizadas por outro grupo, pois devido à cooperação que existiu entre os

grupos uns informavam aos outros o que funcionou bem e o que não funcionou também. A partir das análises nos cadernos dos alunos foi possível identificar que o ambiente de trabalho propiciado pela disciplina (Robótica: Sistemas Sensorial e Motor) era um ambiente onde sugestão e ética no trabalho em equipe foi uma das qualidades que os alunos puderam experienciar.

Sugestão e ética no trabalho em equipe

Este aspecto pode ser discutido sob dois enfoques. Um que trata da sugestão, ética e cooperação no trabalho em equipe e outro que trata do estabelecimento de regras para a competição que ocorreu no final do semestre como parte de atividade do projeto 3.

a) Ética e cooperação no trabalho em equipe

Opinar sobre o projeto do colega ou discutir as regras sob as quais a competição deveria ser submetida e também definir critérios para formação dos grupos foram algumas características que se observou ao longo do semestre na disciplina (Robótica: Sistemas Sensorial e Motor). Um exemplo do critério de formação de um dos grupos colocado no *site* de um dos grupos foi o seguinte:

"Os integrantes foram selecionados por um critério rigoroso de seleção na busca de pessoas capazes de trabalhar em equipe e com ideais semelhantes, para assegurar um nível ótimo nos trabalhos realizados" Fica evidente que para esse grupo era importante a afinidade entre as pessoas que o constituíssem e, isso estava condicionado ao compromisso com o trabalho em grupo e com a qualidade do produto final.

Sugestões de colegas do mesmo grupo ou de outros grupos eram muito bem aceitas. Em muitas situações houve pedido de ajuda entre os alunos. Houve também socialização de informação como aconteceu na seguinte mensagem disponibilizada na página dos alunos de um dos grupos:

"Atenção, antes de começar a editar a página do seu grupo de robótica, leia estes avisos do Clyde:

Gente, antes de editarem qualquer arquivo dentro desta Home page, lembrem-se de fazer algumas coisas:

- *Sempre alterem o arquivo que vocês acabaram de fazer para o grupo GOLEM, para as permissões de leitura para todos, e escrita para user e group. Os comandos para fazer isso são: `chgrp golem "nome do arquivo"`, `chmod a+r "nome do arquivo"`, `chmod g+w "nome do arquivo"`.*
- *Lembrem-se sempre de avisar o resto do grupo por e-mail quando vocês alterarem um arquivo, de maneira que todos possam olhar e dar sugestões.*
- *Quando alguém avisar por e-mail que alterou um arquivo, olhem e dêem sugestões".*

Outro tipo de cooperação, desta vez uma sugestão, disponibilizada pelo professor na página do curso:

"use o sensor de luz para diferenciar entre objetos diferentes. Façam os testes sempre considerando uma calibragem dos sensores, usando ambientes com diferentes quantidades de iluminação. No dia da competição, a luz será a mesma para todos os grupos e os robôs terão um tempo de um minuto para a calibragem. Procure fazer um programa modular, ou seja, cada módulo é responsável por uma ação. Por favor, comuniquem qualquer problema encontrado na especificação... "

b) Regras para a competição

O encerramento do projeto 3 e também da disciplina se deu com realização de uma competição onde dois robôs, conforme a Figura 5.17, UVs (*Umnanned Vehicles*), um de cada

equipe, deveria entrar no ambiente (arena) composto por 4 paredes com uma porta de entrada apenas, e retirar humanos amigos deste ambiente.

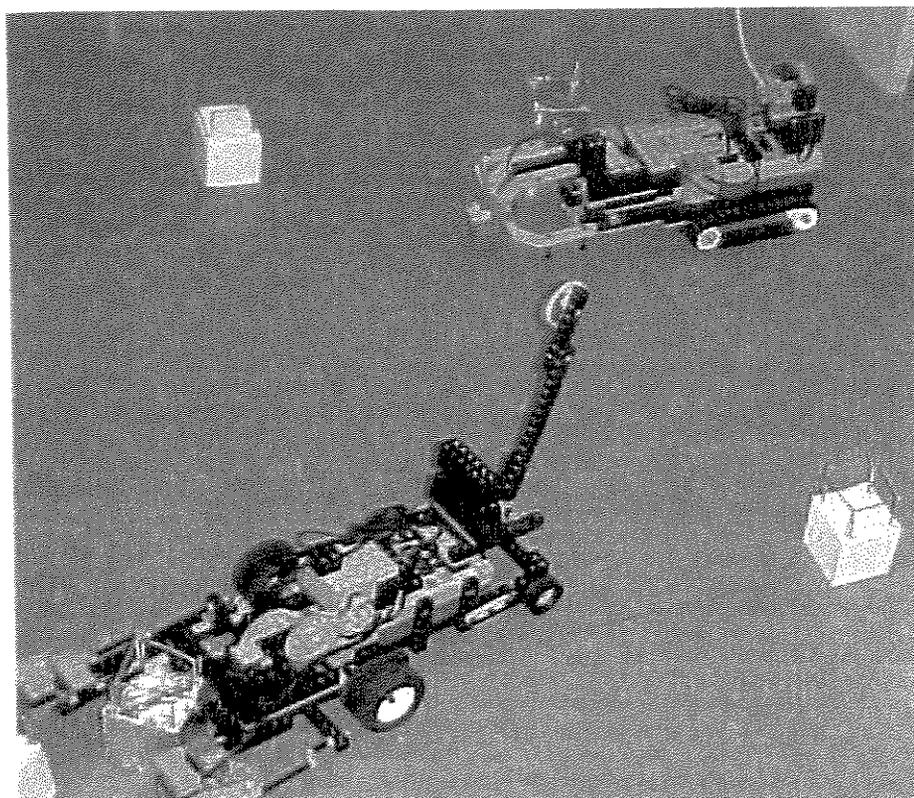


Figura 5.17 – Dois robôs competindo na arena

Uma situação interessante de se ressaltar neste aspecto foi o momento de definição de regras para a competição em um das aulas. Durante esta aula, os alunos discutiram e definiram várias normas que deveriam ser obedecidas no momento da competição como, por exemplo, a retirada do robô da área reservada para competição caso ele se entale num dos vértices e não consiga sair. Este robô poderia ser retirado, porém, o grupo a qual ele pertencia perderia pontos. Um fato curioso de se descrever foi à atitude de um dos alunos que, logo após a definição das regras, perguntou aos colegas se não havia mais nenhuma regra a ser definida. Mediante a resposta afirmativa dos colegas, de que todas as regras estavam definidas, este aluno disse "já que as regras estão definidas tudo que não foi definido está valendo".

Esta atitude causou estranheza em todos os seus colegas, pois muitos deles não entenderam de imediato a intenção daquele colega que, na verdade, estava demonstrando um gesto antiético de se aproveitar das brechas nas regras definidas para tirar vantagens, provavelmente para o seu grupo. Segundo este aluno, "se não está na regra então não é proibido". Ao final desta aula foram definidas as regras que depois de discutidas com o professor ficaram assim estabelecidas:

- 1 Arena da competição: espaço com dimensões de 1,5 por 2,0 metros. Humanos a serem resgatados, (na realidade, caixinhas feitas de isopor com 5x5 cm e 3 cm de altura) espalhados aleatoriamente pela arena.
- 2 Podiam ser usadas caixinhas de duas cores, e estas cores deveriam ser diferentes da cor das paredes que limitavam a arena. Cor da parede da arena é cinza.
- 3 Cor das caixinhas (humanos) prateado e branco.

Obs: Em função da cor da parede os grupos se reuniram para determinar a cor dos humanos. As cores prateado e branco foram escolhidas porque estas duas cores não predominam em nenhuma parte do kit LEGO. Assim, os humanos podiam ser diferenciados dos robôs na arena.

- 4 As caixinhas tinham que ter quatro ganchos (um em cada lateral) de forma que uma garra posicionada na frente do robô UV pudesse engatar neste gancho.
- 5 Gancho tinha que ter o formato de trave de gol com 3 cm de altura de forma que o robô só podia agarrar a caixinha humano, pelo gancho, se a sua garra estivesse a uma altura que variava entre 5 e 8 cm acima do chão.
- 6 Após pegar a caixinha o robô podia levantá-la.
- 7 Podia ser usado outro tipo de resgate.

- 8 Ao invés do robô fazer o engate do humano o robô pode envolve-lo com uma garra e arrastá-lo para fora da arena. O robô pode também "engolir" o humano, colocando-o embaixo dele arrastá-lo para fora da arena e em seguida soltá-lo.
- 9 Existia apenas uma porta de entrada na arena.
- 10 As caixinhas (humanos) deveriam ser agarradas uma a uma e depositadas em área externa a arena.
- 11 Os robôs podiam entrar e sair simultaneamente da arena.
- 12 Portanto, os alunos deveriam prever o caso de um robô encontrar outro no caminho.
- 13 Não era permitido que o robô agarrasse e atirasse os humanos para fora da arena, o robô deveria sair da arena para depois soltar o humano.

5.4. 5 Categoria 5: Identificação dos Robôs

Conforme pode ser visualizado na tabela 5.2, dar nome aos robôs desenvolvidos nos projetos e as páginas Internet construídas foi uma das características que se evidenciaram e que nos demonstra não só a necessidade que os alunos tinham em se identificarem e mas, que também realçou o lado lúdico que o ambiente de aprendizado da disciplina Sistemas Sensorial e Motor pôde proporcionar. Com base na análise dos cadernos dos alunos pôde-se identificar e criar o aspecto **Denominando o grupo e os projetos** que será descrito a seguir.

Denominando o Grupo e os Projetos

Alguns robôs desenvolvidos foram denominados de KGG - Kaixinha de Glub Glub Lagosta, Aanil, Annie, Afrodite. Estes nomes tinham a ver com as aparências que os robôs adquiriram ao longo do projeto ou com a tarefa que eles deveriam desempenhar, ou alguma homenagem a algum integrante do grupo. A Lagosta era um robô com a aparência de uma

lagosta. O KGG - (Kaixinha de Glub Glub) era um robô que capturava caixas de isopor. A Annie, por exemplo, é uma homenagem à filhinha de um dos integrantes do grupo. A Aanil (Assassino Narcisista de Inocentes Legos) foi o nome atribuído ao primeiro robô construído pelo grupo. Afrodite foi o nome da dado a um robô que não tinha uma forma bem definida, pois, segundo o grupo Afrodite é uma deusa e, portanto, podia assumir a forma que ela quisesse.

As páginas construídas foram batizadas como, Lobo solitário, Adoradores do Tchuuu, Equipe 3MIC, Estaleiro Imperial e Afrodite. Pelo fato dessas páginas já não existirem, deixou-se de colocar o endereço delas.

5.4.6 Categoria 6: Aproveitando Idéias

Em muitas situações a tentativa de solução de um problema, pelos alunos, envolveu a utilização de conceitos de outras disciplinas e também o aproveitamento de alguma idéia que já havia sido testada em outra situação. Por exemplo, com base no chassis montado para o projeto 1, uma vez depurado e solucionado os problemas ocorridos, utilizou-se o mesmo chassis já com algumas melhorias no projeto 2. Isso foi demonstrado nos cadernos da maioria dos alunos e partir desta constatação foi constituída esta categoria composta pelos aspectos **Utilização de Idéias do projeto anterior em projetos subseqüentes** e outros dois aspectos (Formulando matematicamente o problema e Definição de estratégias), conforme a tabela 5.2, já descritos, anteriormente.

Utilização de Idéias do projeto anterior em projetos subseqüentes

A opção, de alguns grupos de alunos, de utilizar às idéias de um projeto em projetos subseqüentes ficou evidenciada em algumas descrições colocadas nos *sites* dos grupos. A seguir apresenta-se descrições do *site* do grupo 1 e do grupo 2.

Descrição do Grupo 1

"Projeto 0: Aanil - Aanil Assassino Narcisista de Inocentes LEGOS, foi a primeira experiência do grupo com o kit mindstorms da LEGO, não realmente um projeto.

Projeto 1.0: Choogaboom v1.0 - Choogabom é o nome do carro à prova de balas da quadrilha de morte, e do nosso robô do curso também. Nesta versão, ele deve ser capaz de desviar de paredes utilizando sensores de luz. O Choogaboom 1.0 utiliza um motor para a roda direita e um para a esquerda, e vira utilizando força invertida nas rodas.

Projeto 1.1: Choogaboom v1.1 - Bugs Corrigidos: Foram feitas uma série de modificações, que levaram à nova versão. A principal delas é o uso de uma roda solta atrás do robô.

Projeto 1.2: Choogaboom v1.2 - A versão 1.1 apresentava problemas sérios de desvio da rota original. Duas mudanças no robô consertaram isso: A roda solta fica na frente, e o uso de "rodas estereo" :P"

Descrição do Grupo 2

1ª forma de Afrodite

Implementação (não muito funcional :) de Afrodite com um mecanismo parecido com o funcionamento de um carro (vira rodas como um carro).

2ª forma de Afrodite

Implementação de um mecanismo (apelidado de pêndulo) que permite que o robô utilize somente um motor para movimentação (com capacidade de curvas, essa é a parte interessante! :).

3ª forma de Afrodite e primeiro projeto

Implementação utilizada para solucionar o problema do primeiro projeto (detectar obstáculo). A mais bonitinha, mas com poucas fotos.

4ª forma de Afrodite e segundo projeto

Forma que foi utilizada na resolução do segundo projeto. Utiliza "braços" que caem ao "encontrar" uma borda desativando o sensor de toque e evitando que o robô caia. As fotos dela precisam ser escaneadas... logo as colocaremos aqui. :)

5ª forma de Afrodite e terceiro projeto (campeonato!)

Implementação para o campeonato (terceiro projeto). Ainda em fase de testes...

5.5 Considerações sobre o Capítulo 5

Estas considerações se referem aos trabalhos desenvolvidos com os alunos das engenharias mecânica e mecatrônica e engenharia e ciências da computação, portanto todas elas se aplicam aos três grupos com os quais se realizou o trabalho de pesquisa de campo, incluindo-se aqui os alunos de mestrado da computação.

No que diz respeito às atividades desenvolvidas junto aos alunos das engenharias vale ressaltar a importância em se desenvolver projetos muito próximos a situações reais da fábrica suscitando no aluno o interesse pelo aspecto profissional futuro. Isso foi em muitos momentos, durante a pesquisa, utilizado pelos alunos como uma forma de desenvolvimento pessoal, de demonstrarem o quanto eram capazes. Tanto primar pela qualidade, quanto à aquisição da postura de pesquisador são atitudes que devem continuar sendo incentivados nesses alunos. Equipar cada vez mais os laboratório para atender um número maior de alunos deve ser um desafio constante da coordenação dos cursos.

A análise qualitativa de dados multicategóricos, do trabalho realizado junto aos alunos do Instituto de Computação da Unicamp consistiu da verificação dos cadernos desses alunos. O

trabalho foi realizado durante um semestre pelo autor ministrando aulas de montagem, automação e controle de dispositivos robóticos a partir de leitura e interpretação de comentários e anotações dos cadernos dos alunos e também das páginas de Internet desenvolvidas por eles. O caderno era o material de bancada individual que o aluno levava para todas as aulas práticas a fim de anotar todos os momentos que ele considerava importantes durante o processo de desenvolvimento dos projetos no seu grupo. A fim de contextualizar esse processo, o capítulo apresentou a forma como foi estruturada este curso, que parece ter sido uma das condições que tornaram possível, ao final do semestre a realização dessa análise.

Ainda no que diz respeito ao trabalho desenvolvido com os alunos da Computação, ao analisar a constituição de diversas categorias, observou-se que existe uma relação horizontal e outra vertical entre aspectos observados no caderno dos alunos e as categorias constituídas a partir deles. Por exemplo, a tentativa de se calcular o tempo de giro do eixo do motor num robô com o formato de um carro, levou à manipulação de conceitos de Peso do chassi que por sua vez levou ao cálculo do Atrito das rodas, e deste para a alteração do Centro de Massa e finalmente a diminuição do Torque. Isso demonstra um entre vários encadeamentos de raciocínios descritos pelos alunos envolvendo diferentes conceitos na mesma disciplina e, às vezes entre diferentes disciplinas.

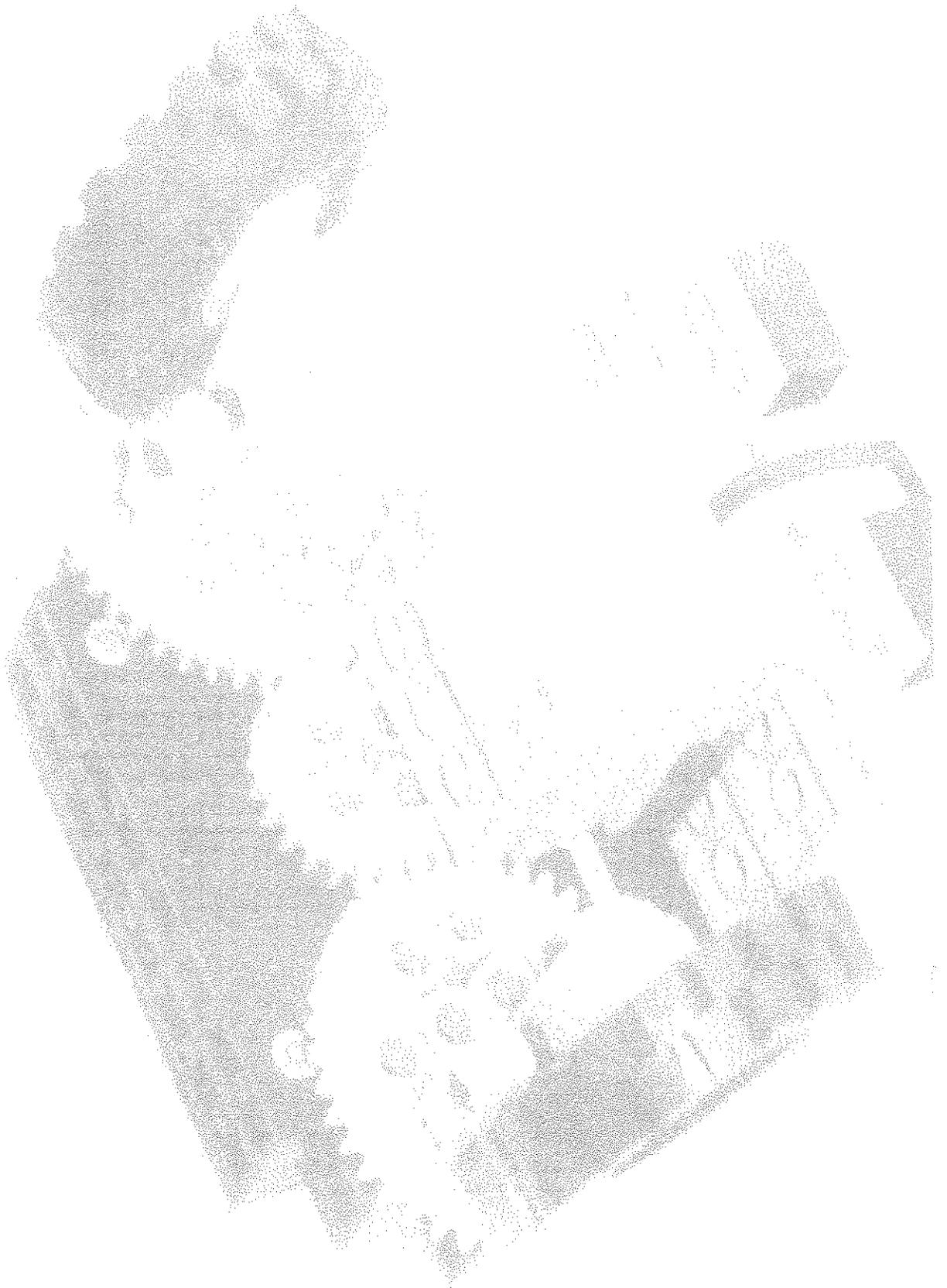
Embora tivesse sido elaborado um planejamento onde, por exemplo, os seminários seriam todos apresentados ao final da disciplina, a dinâmica do curso mostrou uma realidade diferente, em que seria mais proveitoso apresentar o seminário de cada projeto logo após o término deste.

Neste capítulo descreveu-se a forma como os alunos, tanto das engenharias mecânica e mecatrônica quanto de engenharia e ciências da computação trabalharam conceitos científicos, estratégias utilizadas para resolução de problemas dentre outros aspectos levantados nos cadernos ou observados no laboratório. Discutiu-se também como esses aspectos foram organizados em categorias e as evidências para as quais eles apontavam. O critério utilizado para se agrupar os aspectos observados em categorias também foi discutido pois ele era um dos fatores importantes a ser definido durante o processo de leitura dos cadernos a fim de se selecionar os pontos mais relevantes apontados pelo aluno. Os aspectos e categorias representados numa tabela, quando

foram analisados demonstrou-se que existe uma relação horizontal e outra vertical entre eles, o quê, na realidade, denota a relação interdisciplinar que permeia toda a atividade relacionada com implementação e desenvolvimento de projetos mecatrônicos.

Um dos tópicos também discutido neste capítulo foi à possibilidade que se criou para desenvolver projetos em grupo, respeitando a individualidade de cada um, perpassando pelas questões da ética e cooperação inerentes ao trabalho em equipe.

Ao longo deste capítulo foram desenvolvidas uma série de idéias que apontam para estudos futuros passíveis de serem realizados sobre como a atividade de montagem automação e controle de dispositivos mecatrônicos podem permitir no desenvolvimento de projetos que levam a um aprofundamento do estudo de conceitos científicos de forma abrangente. Ou seja, partindo de um projeto simples da montagem de um carro que evita obstáculos é possível desenvolver conceitos de Mecânica, Eletrônica e Computação, sem contar a contribuição da Matemática, Física, Estatística, etc.. Estudar o nível de aprofundamento e de contribuição de cada um desses conceitos ao desenvolver um projeto mecatrônico pode ser uma atividade de pesquisa interessante. Além disso, identificar conceitos de outras áreas de conhecimento que podem ser inter-relacionados a um determinado projeto pode ser uma proposta de trabalho a ser desenvolvida objetivando identificar relações interdisciplinar possíveis de se estabelecer ao desenvolver projetos mecatrônicos. O próximo capítulo discute as perspectivas futuras desse trabalho com base no que foi descrito até aqui, e também as suas considerações finais.



Capítulo 6

Considerações Finais e Perspectivas Futuras

Ao longo da tese, foi apresentado um panorama geral acerca da importância de se introduzir novas tecnologias no processo de ensino-aprendizagem implementando ambientes que propiciam o aprendizado de conceitos científicos tanto em situações formais de aprendizagem (escolas e universidades) quanto em situações não formais (parque de diversões, fábricas e museus, dentre outros). O contexto da investigação, no entanto, limitou-se a analisar ambientes de aprendizagem na universidade e na fábrica, buscando demonstrar a necessidade de a universidade visualizar o ensino nos cursos, de graduação sob um enfoque interdisciplinar/multidisciplinar que propicie ao aluno, futuro profissional de Engenharia Mecatrônica, a vivência de situações reais com as quais irá se defrontar quando estiver no mercado de trabalho.

6.1 Considerações

O estudo teve como sujeitos da pesquisa alunos do curso de engenharias mecânica, mecatrônica, de engenharia e ciências da computação (de graduação e pós-graduação) da Unicamp, e técnicos especializados, em ambiente de trabalho, de uma fábrica, (Delphi - Harrison) que produzia componentes automotivos. As atividades desenvolvidas num período de quatro semestres com os alunos de mecânica e mecatrônica e um semestre com os alunos do curso de

computação, demonstraram que um ambiente de aprendizagem que possui materiais que propiciam ao aluno a chance de criar e de expressar as suas idéias por meio dos dispositivos que ele próprio constrói é um ambiente rico em situações de aprendizagem. Os dados empíricos evidenciaram que, a oportunidade que os alunos tiveram para implementar nos laboratórios da universidade sistemas similares aos sistemas reais existentes nas indústrias, permitiu a esses estudantes vivenciarem problemas concretos de linhas de produção num contexto onde eles puderam avaliar seus erros e validar suas hipóteses de forma simples, econômica e segura, o que já não seria viável em uma indústria, pois é praticamente impossível, interromper a linha de produção em uma fábrica para se explicar a um técnico detalhes de uma determinada máquina em operação nessa linha. Parar a linha de produção pode implicar em aumento de custos (e até prejuízos) para a empresa.

A pesquisa de campo desenvolvida com os técnicos especializados na fábrica demonstrou que somente com uma formação básica de boa qualidade é que o técnico terá condições de se aprimorar na sua área profissional num contexto onde a atualização tecnológica é uma necessidade constante imposta pelo mercado. Isso foi apontado no capítulo 1 discutindo-se a questão da flexibilização funcional que vem exigindo um novo perfil de qualificação da força de trabalho, apontando para a necessidade de que os trabalhadores sejam detentores de uma escolaridade básica, que lhes permita a capacidade de adaptação a novas situações, compreensão global das tarefas e capacidade de abstração no trato e interpretação de informações, implicando em uma abertura para novas situações de aprendizagem e estímulo à criatividade para enfrentamento de imprevistos.

Os dados de pesquisa levantados na fábrica de componentes automotivos e nos cursos da UNICAMP permitiram estabelecer um paralelo entre a aprendizagem na universidade e na indústria sob os pontos de vista da automação, aspectos de desenvolvimento de projetos e aprendizado de conceitos.

Do ponto de vista da automação, dentro do processo de inovações tecnológicas presentes nas indústrias onde os alunos de mecatrônica irão atuar, analisou-se em detalhes uma das ferramentas para especificação de projetos que é o Grafcet. Em alguns projetos desenvolvidos

pelos alunos, essa ferramenta foi utilizada com objetivo de se trabalhar a idéia de documentação de projetos numa linguagem mais apropriada para a automação. Sensores e atuadores foram estudados como elementos transdutores principais que fazem parte da implementação de um dispositivo robótico/mecatrônico mostrando a utilidade do primeiro como dispositivo que muda o seu comportamento sob a ação de uma grandeza física e o segundo como dispositivo que aplica ou faz atuar energia mecânica sobre uma máquina, para realizar um determinado trabalho. Diferentes tipos de sensores e atuadores foram apresentados, destacando-se os do tipo educacional comercializados pela LEGO por serem os que foram usados em todos os projetos desenvolvidos nesse trabalho. Isso levou à discussão a respeito do estado da arte na implementação/utilização de dispositivos mecânicos automatizados, no contexto educacional, abordando-se alguns projetos desenvolvidos em instituições e universidades tendo como ênfase à robótica e a telerobótica em alguns dos seus enfoques de uso na rede Internet.

A parte experimental da tese se constituiu na observação, atuação como profissional do NIED, como professor ministrando aulas práticas nas disciplinas Robótica: Sistema Sensorial Motor e Introdução à Engenharia de Controle e Automação no Instituto de Computação e na Faculdade de Engenharia Mecânica-DPM/LAR respectivamente e, no acompanhamento e análise de diversos projetos prático-experimentais envolvendo trabalho na fábrica e na sala de aula.

Na atividade desenvolvida com os alunos do Instituto de Computação, a metodologia utilizada para a análise dos dados foi a análise qualitativa com base na leitura dos apontamentos e comentários que cada aluno fez em seu caderno individual utilizado nas aulas práticas de montagem de dispositivos mecânicos. Incluiu-se também nesta análise as páginas de *site* que cada grupo desenvolveu sobre os projetos. A análise do material empírico da pesquisa permitiu a construção de categorias que evidenciaram um conjunto de atributos e características comuns, que ajudaram a explicitar fenômenos tais como: diferentes heurísticas para solucionar um problema, solução por software, identificação, etc., presentes no processo dos alunos implementarem os seus projetos. As inter-relações entre os aspectos e as categorias demonstraram a relação interdisciplinar entre os conceitos trabalhados e isso evidenciou a importância que a atividade de desenvolvimento de projetos robóticos/mecatrônicos pode ter no sentido de integrar conhecimentos de diferentes disciplinas. O papel aglutinador na relação aluno-

aluno e na relação professor-aluno foi bastante destacado neste trabalho que envolveu discussão de questões relevantes que terão reflexos não somente na vida acadêmica, mas também na vida profissional dos alunos como capacidade de trabalho em equipe, respeito às regras definidas no grupo, e ética.

O acompanhamento e a análise desses projetos desenvolvidos por alunos e técnicos demonstrou alguns aspectos comuns relevantes sob o ponto de vista da aprendizagem de ambos tais como: motivação pelo projeto prático-experimental, realização de atividade de pesquisa (utilização de diversos softwares da área de robótica), consistência nos projetos desenvolvidos, vantagens do ponto de vista educacional, aprendizagem de aspectos específicos de funcionamento das máquinas desenvolvidas, aprendizagem de conceitos científicos em geral, dentre outros. Esses aspectos foram tratados numa abordagem que apontava para a preocupação com o material a ser utilizado, especificação do projeto mecânico a ser desenvolvido, software a ser utilizado na programação dos dispositivos em função da sua performance, similaridade do sistema implementado com o sistema real existente em uma fábrica, dentre outros.

Em ambos os contextos de aprendizagem formal e não-formal, foram apresentadas abordagens para aprendizado de conceitos a partir do processo de montagem e programação de dispositivos mecânicos, utilizando-se kits LEGO e outros materiais, num ambiente onde se valoriza o fazer como forma de aprender criando situações problemas, para os alunos, oferecendo-se desafios possíveis de serem resolvidos experimentalmente por meio de soluções fundamentadas no conhecimento científico. Foram demonstrados nos capítulos 4 e 5 exemplos de projetos e realização de trabalhos em salas de aulas e em laboratórios que apontam para uma proposta de integração de dispositivos mecatrônicos na atividade didática do professor.

No caso específico da disciplina Introdução à Engenharia de Controle e Automação, para que a atividade de montagem, automação de dispositivos mecatrônicos não aconteça somente no primeiro semestre do curso de Controle e Automação, momento em que esta disciplina é oferecida, os exemplos de projetos descritos nos capítulos 4 e 5 realçam a importância dos alunos desenvolverem ao longo do curso atividades que propiciam uma integração dos diferentes conteúdos que compõem a grade curricular. Esta integração pode ser viabilizada transformando a

atividade de montagem e automação de dispositivos mecatrônicos em uma nova disciplina que seria ministrada durante vários semestres. Tomando como base o objetivo principal do curso de Controle e Automação, de propiciar ao aluno uma formação multidisciplinar nas áreas de mecânica, eletrônica, instrumentação industrial, informática, controle e gestão da produção. Essa nova disciplina integraria de forma paulatina o conteúdo das demais disciplinas. Ela integraria e caminharia na progressão com as demais disciplinas possibilitando sempre um contexto prático para se aplicar nos laboratórios os conceitos teóricos abordados no curso. A coordenação do curso de Mecatrônica deve criar condições para que o professor responsável por essa nova disciplina se mantenha em constante interação com os demais professores para possibilitar com que os exemplos de projetos desenvolvidos no laboratório de montagem e automação de dispositivos mecatrônicos sejam os mais abrangentes possíveis buscando cobrir, principalmente, os conteúdos das disciplinas consideradas da área profissionalizante.

A nova disciplina deverá propiciar condições que: favoreçam a realização de projetos de iniciação científica dos alunos do curso de Mecatrônica; contemplem a implementação de situações práticas existentes nas indústrias; propiciem a interação com os outros professores do curso de Controle e Automação; possibilitem uma rede de comunicação envolvendo professor aluno e coordenação do curso. Enfim, que crie um clima de diálogo e cooperação entre todas as instâncias envolvidas nesse processo para que o gerenciamento dos recursos e materiais disponibilizados para esta disciplina se faça de forma eficiente e beneficie a todos os interessados.

Sob esta ótica, acredita-se ser possível adotar uma proposta metodológica que não só permite estabelecer no curso de Controle e Automação uma ponte entre a teoria e a prática como também construir um currículo permeado de propostas de projetos que o aluno desenvolva paulatinamente ao longo de sua vida acadêmica chegando ao término do seu curso com um Projeto de Fim de Curso consubstanciado em conhecimentos adquiridos nas diferentes disciplinas que ele cursou e também nos diferentes projetos já realizados. Isso difere da situação atual em que a maioria dos alunos só começa a pensar no Projeto de Fim de Curso nos últimos semestres de graduação, buscando empregar/utilizar num período de seis meses ou um ano e, num único projeto, quase todo conhecimento adquirido desde o primeiro ano.

Os exemplos de projetos desenvolvidos durante a implementação da tese, apresentados/analísados no capítulo 5 abordam uma série de conceitos que podem ser trabalhados envolvendo diferentes disciplinas do currículo de graduação utilizando projetos de sistemas mecânicos similares aos existentes nas indústrias, baseados em atuadores, sensores e demais elementos elétricos/eletrônicos. Identificar nesses sistemas princípios de automação e controle, utilizá-los dentro de uma abordagem multi e interdisciplinar, desenvolver metodologias que permitam trabalhar de forma concreta o aprendizado de conceitos nas diferentes disciplinas do curso de Engenharia de Controle e Automação (Mecatrônica), são posturas que devem ser adotadas pelos professores responsáveis por esta nova disciplina.

Ainda do capítulo 5, a reflexão sobre os projetos desenvolvidos nas engenharias e a análise dos cadernos tornou-se possível porque o pesquisador acompanhou todo o processo. A familiarização e a convivência com os grupos dos alunos facilitaram em muitos momentos a compreensão de algo que, muitas vezes, o grupo queria expressar, mas que efetivamente não estava explícito. Como pôde ser constatado com as diferentes populações de alunos com os quais o pesquisador trabalhou, existe em todo esse contexto uma relação de afinidades que podem ser estabelecidas entre aluno-aluno, professor-aluno, aluno-direção do curso e professor-direção, que permite com que todos se conheçam mais, se identifiquem com algo em comum e trabalhem em prol do desenvolvimento da coletividade e também do desenvolvimento pessoal. Todavia isso não é uma tarefa que se deve fazer em um semestre. Em função de tudo isso, com base neste estudo, pode-se propor que as atividades desenvolvidas em laboratório com os alunos de mecatrônica sejam transformadas em uma disciplina, visando substituir uma situação de trabalho que hoje é desenvolvida de forma totalmente fragmentada por uma situação que assegure a continuidade do primeiro ao último ano de Mecatrônica. Na fragmentação, as ações realizadas se perdem e caem no esquecimento. Na continuidade as ações tendem a ser produtivas e seus resultados são mais expressivos, tendem a perdurar, e acabam se tornando mais visíveis. Isso pode ser afirmado com base nas evidências demonstradas no trabalho desenvolvido pelo aluno de iniciação científica, do qual, em partes, o pesquisador teve participação, cujo percurso foi e tem sido: primeiro, contato com LEGO por um semestre na disciplina Introdução a Engenharia de Controle e Automação. Segundo, desenvolvimento de projetos com LEGO como estagiário no NIED por um ano. Terceiro, escrita e submissão de um projeto de iniciação científica para

FAPESP para obtenção da bolsa e desenvolvimento deste projeto, integrando LEGO e Robix nos laboratórios do LAR/FEM. Quarto, atuação deste aluno como monitor, no primeiro semestre de 2002 na disciplina Introdução à Engenharia de Controle e Automação, ministrando aulas para alunos iniciantes do primeiro ano do curso de Controle e Automação. Quinto, desenvolvimento de diversas atividades empreendedoras no seu curso como a criação de uma entidade (Grupo de Estudos e Projetos em Automação Industrial - GEPAI) que implementa e viabiliza uma série de atividades integradoras de alunos e professores do seu curso e demais professores da FEM com a comunidade externa (empresas da área de automação em geral). Essas atividades envolvem: palestras, cursos de treinamento e uso de equipamentos, etc.. Isso tudo acontecendo em seqüência, sem nenhuma interrupção de 1999 até o presente momento. O fato de este aluno ter tido a oportunidade de continuar a trabalhar com LEGO após o curso de Introdução à Engenharia de Controle Automação certamente contribuiu para que ele realizasse o percurso acadêmico acima descrito. O crescimento desse aluno foi e é bastante significativo na medida em que ele participa, de forma bastante ativa e produtiva, em diversas atividades do curso de mecatrônica contribuindo até com o processo de formação de seus colegas. O aluno tem ajudado no processo de integração das disciplinas do seu curso, por intermédio do uso de LEGO, promovendo atualmente um curso de gestão produção que consiste na montagem de um sistema de redução de velocidade obedecendo todos os critérios de um sistema real de produção.

Pelo exposto, ao longo da tese conclui-se que o contexto atual denota uma mudança. A mudança de paradigma que tem que acontecer no processo de ensino e aprendizagem envolvendo, as instituições de ensino, pesquisa e extensão. Uma das formas de contribuirmos com esta mudança tem sido as implementações de ambientes de aprendizagem que propiciam ao aluno, cada vez mais, a chance de ser autônomo, criativo e responsável pelo seu próprio processo de aprendizado. Todavia, estes ambientes na forma aqui apresentados não existem *a priori*, portanto, há que ser construído. Para serem ambientes de caráter interdisciplinar, essa construção requer o envolvimento de pessoas (profissionais) e instituições.

Ambientes de aprendizagem baseado no uso de dispositivos robóticos tem possibilitado, de forma simples, econômica, rápida e segura, disponibilizar recursos tecnológicos para a aprendizagem, não só de robótica, mas de ciências de uma maneira geral. Isso tem propiciado

criar situações de aprendizagem onde os alunos e seus professores possam avaliar resultados, experimentar idéias e testar hipóteses simulando situações reais em contextos criados para a finalidade de ensino-aprendizagem.

A utilização de dispositivos robóticos/mecatrônicos tem criado uma interação homem-máquina diferente, na medida em que o ambiente de programação se torna mais rico e mais desafiante. Desenvolver atividades de robótica requer um processo de depuração mais amplo, envolvendo projeto, construção/montagem e elaboração de programas que controlam o dispositivo. Isso amplia a gama de desafios que o aluno tem que vencer. Ao desenvolver atividades em grupo, geralmente existe um bom relacionamento e cooperação entre os parceiros, o que propicia sucessos na aprendizagem.

Os resultados deste estudo envolvendo a aprendizagem de alunos dos cursos de engenharia mecatrônica e de computação e ambiente de trabalho na fábrica demonstrou um aspecto importante a se considerar: a questão da **interdisciplinaridade no processo de formação dos futuros engenheiros**, inerente à atividade de montar e controlar dispositivos via computador. Isso demonstrou a necessidade de que o Laboratório de Automação Integrada e Robótica - LAR, possa contar com um laboratório didático para desenvolvimento e pesquisas aplicadas à área de automação de manufatura e robótica que sirva não só aos interesses do LAR mas, também aos outros departamentos da Faculdade de Engenharia Mecânica, outras faculdades e Institutos da UNICAMP e à comunidade científica em geral, nos âmbitos nacional e internacional.

Acredita-se que a implementação de um Sistema Integrado para Separação de Produtos numa Linha Automatizada de Produção a Partir do Reconhecimento de Cores, será um ambiente que permitirá realizar a integração do sistema de gestão da produção, dentro das instalações da universidade, proporcionando aos alunos um ambiente de simulação muito próximo ao ambiente real existente nas fábricas. O sistema será um ambiente, para a aprendizagem, onde os alunos poderão avaliar resultados, experimentar idéias e testar hipóteses, o que seria um pouco mais difícil de se fazer na fábrica.

Ao longo destes anos, o desenvolvimento de atividades de montagem automação e controle de dispositivos mecatrônicos na disciplina Introdução a Engenharia de Controle e Automação

tem produzido resultados bastante satisfatórios. Todavia, o fato das atividades acontecerem somente no primeiro semestre do curso e não pertencerem a uma disciplina específica tem se constituído em um empecilho, pois no momento em que os alunos começam a entender e a participar de forma mais efetiva das atividades, as mesmas são interrompidas porque se chegou ao fim do semestre e não houve tempo para se fazer à integração de conceitos de diferentes disciplinas. Na realidade, as atividades práticas, no formato como são realizadas atualmente, não estão propiciando condições para que, de forma continuada, sejam integradas e trabalhadas interdisciplinarmente conceitos das diferentes disciplinas do curso de Controle e Automação. Este curso é um produto de pelo menos duas grandes áreas de conhecimento (Mecânica e Eletrônica), portanto as disciplinas e os conceitos inerentes a elas têm origem nessas duas áreas que devem estar, de fato, integradas. Nesse contexto, desenvolver projetos de dispositivos mecatrônicos deve se constituir em uma atividade interdisciplinar de maior abrangência possível para que o aluno, auxiliado pelos seus professores, consiga fazer conexões entre conteúdos de diversas disciplinas do curso.

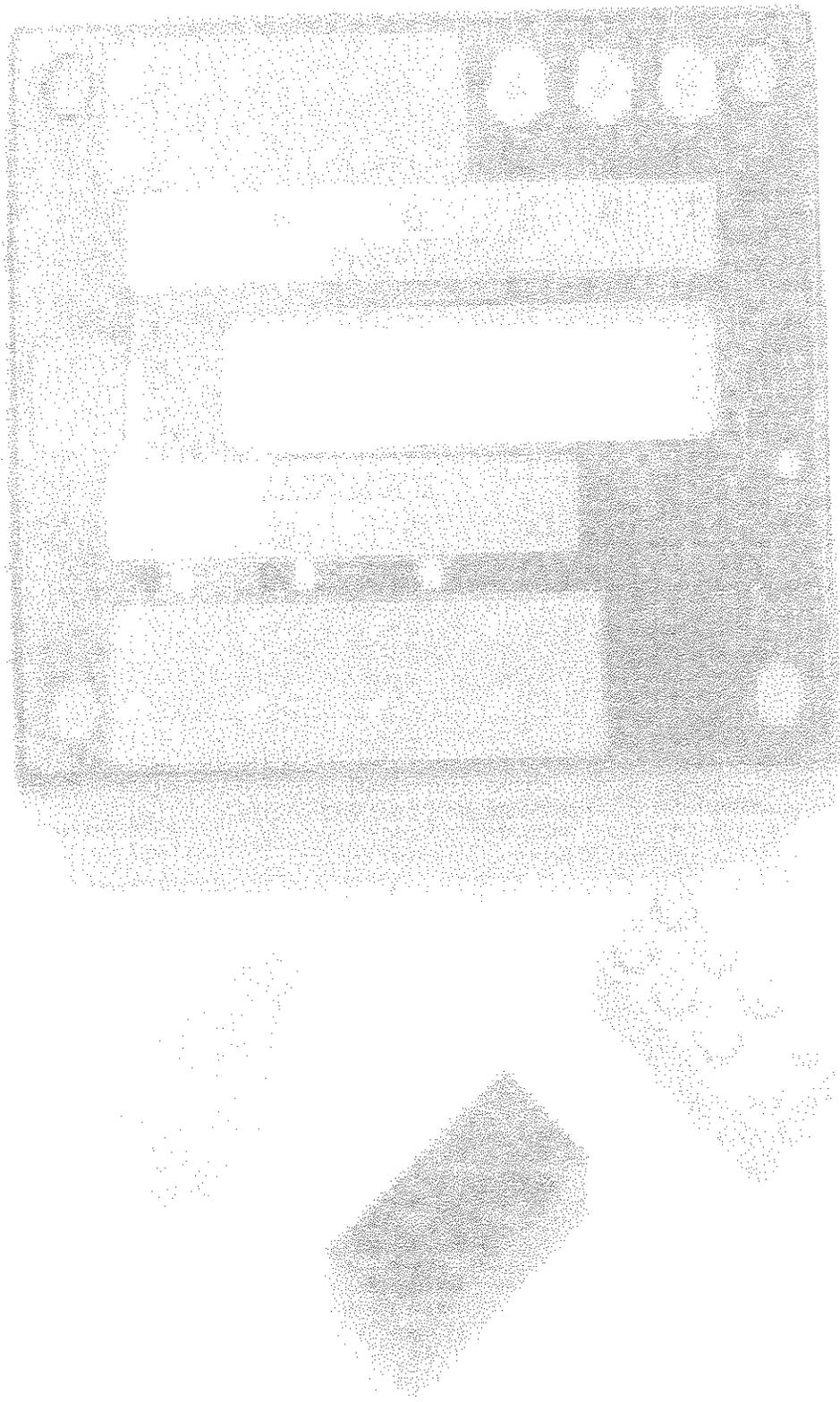
6.2 Propostas Futuras

Como uma das propostas futuras mais importantes, o pesquisador aponta para a realização de esforços objetivando a transformação da atividade com LEGO do curso de Controle e Automação (mecatrônica) em uma disciplina de graduação que se inicia no primeiro semestre e que tenha continuidade, nos semestres subsequentes, como forma de se integrar os diferentes conceitos que o aluno aprende durante a sua formação. Isso com certeza propiciará com que esses conceitos sejam apresentados de forma contextualizada e inter-relacionada de modo a quebrar as fragmentações existentes entre as disciplinas do curso de mecatrônica, o que contribuirá para melhores condições do aprendizado acadêmico e também para a integração, mais rápida possível, do aluno com as diversas vertentes profissionais para o qual o seu curso pode ser direcionado.

A rede mundial de computadores, Internet, é hoje o meio ideal para a construção de laboratórios acessíveis à distância, uma vez que o acesso a ela é quase universal e relativamente

barato, mesmo para habitantes de países em desenvolvimento. Portanto, uma das metas futuras de pesquisa pode ser o desenvolvimento de recursos que permitirão aos alunos e pesquisadores do LAR acessarem remotamente os dispositivos robóticos ali instalados, permitindo a utilização, disponibilização, compartilhamento e cooperação dos recursos e dos projetos com outras instituições, tanto as de ensino e pesquisa, quanto as de produção de bens e serviços. Isso, no campo educacional, possibilitará a construção de ambientes de aprendizagem que apresentam três grandes possibilidades: primeira, a de atendimento de uma quantidade maior de alunos ao mesmo tempo, em grande escala, à distância, permitindo multiplicar o número de usuários de um único conjunto de equipamentos. Segunda, a disponibilização e disseminação de ambientes de aprendizagem na área de mecatrônica a baixo custo, o que propicia o acesso à tecnologia de ponta a um grande número de estudantes de outras instituições no âmbito universal. Terceira, fazer do LAR um centro de referência de pesquisa na área de automação e controle de visibilidade nacional e internacional. Algumas pesquisas em desenvolvimento atualmente no LAR já apontam para isso.

Finalmente acredita-se que a tese deverá contribuir com o desenvolvimento de pesquisas tanto no NIED quanto no LAR como uma atividade contínua no processo de formação de alunos desde o nível fundamental até a pós-graduação.



Referências Bibliográficas

BJERRE C. & HVOLDAL J. at all, *Adaptive Pet Robots*, Master Tese, University of Aarhus, Department of Computer Science, Aarhus N, Denmark.,1999.

CANTÙ, M. *Dominando o Delphi 4 "A Bíblia"*. Makron Books do Brasil Ltda, São Paulo,1998.

CARVALHO, A. M. P. & LIMA, M. C. B. *O Falar, O Escrever e O Desenhar Na Construção de Conceitos Científicos, in Linguagens, Leituras e Ensino da Ciência*. Maria José P. M. de Almeida & Henrique César da Silva Orgs., Mercado de Letras, Campinas São Paulo, 1998.

CASTELLS, M., BARRERA, A., CASAL, P., et all. *El Desafio Tecnológico. Espanha y las Nuevas Tecnologías*. Alianza Editorial S. A., Madrid, 1986.

CASTILLO, J. C. M., *GRAF CET ENLACES*, Departamento de Electricidad, Béjar Salamanca, 2001.

CASTRO, N A., *Organização do Trabalho, Qualificação e Controle na Indústria Moderna*. In *Trabalho e Educação*. p. 69-86, Papirus, Campinas - SP, 1992.

CORIAT, B., *Automação Microeletrônica e Competitividade: Tendências no Cenário Internacional*. In, *Automação, Competitividade e Trabalho: A Experiência Internacional*. SCHMITZ, H., & CARVALHO, R. Q., *Humanismo, Ciência e Tecnologia "Hucitec*, São Paulo, 1988.

- D'ABREU, J. V. V. *Design de Dispositivos: uma Abordagem Interdisciplinar* In: VII Congresso Internacional Logo / I Congresso de Informática Educativa do Mercosul, Porto Alegre, RS, p. 48-55, 1995.
- D'ABREU, J. V. V., CHELLA, M. T. *Desenvolvimento de Ambientes Colaborativos de Aprendizagem à Distância Baseado no Controle de Dispositivos Robóticos* In: XII Simpósio Brasileiro de Informática na Educação - SBIE2001, Vitória, ES, p. 393 - 400, 2001.
- D'ABREU, J. V. V., *Construção de Um Traçador Gráfico para Fins educacionais*, dissertação de mestrado, FEEC/UNICAMP, Campinas, SP, 1994.
- D'ABREU, J. V. V., *Desenvolvimento de Ambientes de Aprendizagem Baseados no Uso de Dispositivos Robóticos* In: X Simpósio Brasileiro de Informática na Educação – SBIE99, Universidade Federal de Paraná – UFPR, PR, p. 9 - 16, 1999.
- D'AZZO, J. J., & HOUPIS, H., *Análise e Projeto de Sistemas de Controle Lineares*. Guanabara Dois, Rio de Janeiro, 1975.
- DAUMAS, M., *Las Grandes Etapas Del Progreso Tecnico*. Fondo de Cultura Económica, p. 63-123, México, 1983.
- DERENZO, S. E., *Interfacing a Laboratory Approach Using the Microcomputer for Instrumentation, Data Analysis and Control*. Prentice Hall, Englewood Cliffs, University of California, Berkeley, California, 1990.
- DISTEFANO, J. J., at all. *Sistema de Retroação e Controle* Coleção Schaum, McGraw-Hill do Brasil, LTDA, São Paulo, 1972.
- DORF, C. R., & BISHOP, H. R., *Sistemas de Controle Modernos*, Livros Técnicos e Científicos S. A, Rio de Janeiro, 2001.

- FAZENDA, I. A., *Interdisciplinaridade: Um Projeto em Parceria*. Coleção Educar, Edições Loyola, São Paulo, 1993.
- FREIRE, P. A *Pedagogia do Oprimido*. Paz e Terra, Rio de Janeiro, 1987.
- FRIGOTTO, G., *As Mudanças Tecnológicas e Educação da Classe Trabalhadora: Politecnia, Polivalência ou Qualificação Profissional?*, In *Trabalho e Educação*. p. 45-52, Papirus, Campinas SP, 1992.
- GARCIA, M. F., *A Produção do Conhecimento na escola Pública por Meio da Pesquisa; O Projeto Ciência na Escola*. Tese de doutorado, Faculdade da Educação FE/UNICAMP, Campinas - São Paulo, 2002.
- GEUS, A., *A Empresa Viva: Como as Organizações Podem a Prosperar e se Preperuar*, Campus Rio de Janeiro, 1998.
- GROOVER, M. P. & WEISS, M. at all. *Robótica Tecnologia e Programação*. McGraw-Hill, São Paulo, 1988.
- HANDY, C., & KANTER, R. M., *The Best of the Rest*. In Evans, D. (Org). *Management Gurus*. Pearson Education Limited, England, 2000.
- INAZUMI, P. C. M., *Especificação e Implementação de um Ambiente de Desenvolvimento em GRAFCET*, Tese de Doutorado (em andamento), FEM/UNICAMP, Campinas, SP, 2001.
- JORNAL DA UNICAMP, *Inteligência Artificial: Alunos fazem exercícios com Brinquedos LEGO para Aplicar Conceitos Sofisticados de Robótica*” *Jornal da Unicamp* nº 170 fevereiro de 2002).

- KALKUNTE, M. V. & SARIN, S. C. & WILHELM, W. E. A. in, *Flexible Manufacturing Systems: Methods and Studies*. Edited by Andrew kusiak, Elsevier Science Publishing Company, Inc. U. S. A., 1986.
- KENZEK, G. A., RACHLIN, S. L. & SCANNELL, P. *A Taxionomy for Educational Computing. Educational Technology*, p. 15-19, 1988.
- KUHN, T. S., *Qué son las revoluciones científicas? Y otros ensayos*, Instituto de Ciencias de la Educacion de la Universidad Autonoma de Barcelona, Bellaterra, Espanha , 1989.
- KUO, B. C., *Sistemas de Controle Automático*. Prentice/Hall do Brasil, Rio de Janeiro, 1982.
- LEITE, J. C., Análise e Especificação de Requisitos, notas de aula sobre Engenharia de Software, página Internet, <http://www.dimap.ufrn.br/~jair/ES/es99Ic3.html>, 1999.
- LEITE, M. P., *O Futuro do Trabalho Novas Tecnologias e Subjetividade Operária*, Página Aberta Ltda, São Paulo, 1994.
- LEPIKSON, H. A., *Qualidade da Manufatura Assegurada por Computador*, in *Manufatura Integrada por Computador: Contexto, Tendências Técnicas*, Org. Marília Markus & Pyrano P. da Costa Júnior, Fundação CEFETMINAS, BH, 1995.
- MACHADO, S. R. L., Mudanças tecnológicas da classe trabalhadora, in *Trabalho e Educação*. p. 9-25. Papyrus. Campinas, São – Paulo, 1992.
- MARTIN, F. G., *Robotic Explorations - "A Hands-On Introduction to Engineering"*. Prentice Hall, Inc. Upper Saddle River, New Jersey, 2001.
- MARTIN, F. G., *The 6.270 Robot Builder's Guide, MIT, LEGO Robot Design Competition*. The Media Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 1992.

- MARTÍNEZ, G., ESTRAGUÉS, F., *Sistema de Representació d'Automatimes Seqüencials GRAFCET*, Centre de Documentació I Experiència de Didàctiques Tecnològiques - CDEDT, Malgrat, Barcelona, 1999.
- MAZZONE, J. S., O Sistema "Enxuto" e a Educação no Brasil. In Valente, J, A. (Org). *Computadores e Conhecimento: Repensando a Educação*. Unicamp, Campinas, 1993.
- MENDELSON, P., *L'Ordinateur dans L'Enseignement. Colloque "Enseignement et Apprentissage avec L'Ordinateur"*. p. 6-90. Martigny, Université de Geneve. TECFA Document. 1989.
- MENDONÇA, F., *Engenharia de Sistemas: Planejamento e Controle de Projetos*, Editora Vozes, Petrópolis, Rio de Janeiro, 1972.
- MOLINA PAGNEZ, K. S. M., Projeto Eureka: uma trajetória. Dissertação de mestrado, Universidade Estadual de Campinas, Faculdade de Educação, FE/UNICAMP, Campinas São Paulo, 2001.
- NÉLIS, J., & LALOUP, J., *Homens e Máquinas*, Herder, São Paulo, 1965.
- O'BRIEN, *As Máquinas*, Livraria José Olympio Editora, Rio de Janeiro, 1964.
- OZAKI, M., at Alli, *Cambios Tecnológicos y Relaciones Laborales*, Fareso S. A., Madrid, 1992.
- PAPERT, S., *A Máquina das Crianças: Repensando a Escola na Era da Informática*, Artes Médicas, Porto Alegre, 1994.
- PAPERT, S., *Logo: Computadores e Educação*. Brasiliense, São Paulo, 1985.
- PASSETO, F. *Desenvolvimento de um Aplicativo para Simulação e Controle de Manipuladores Robóticos com Ênfase em Aplicações Didáticas*, dissertação de mestrado, FEM/UNICAMP, Campinas, SP, 2000.

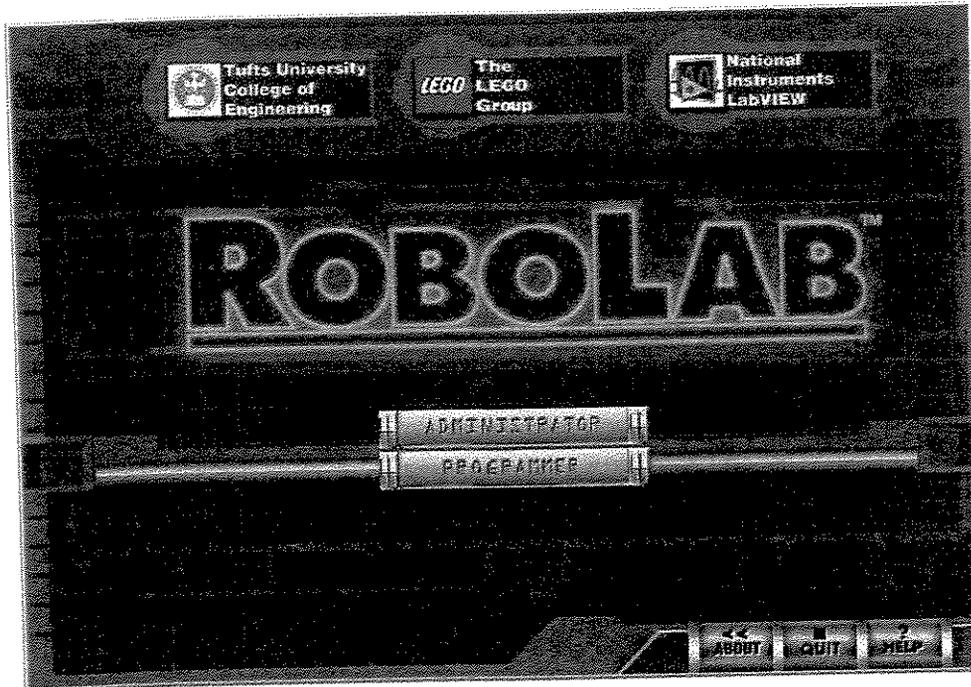
- PETRY, P. P. *Processos Cognitivos de Professores Num Ambiente Construtivista de Robótica Educacional*. Dissertação de Mestrado, UFRGS, Instituto de Filosofia e Ciências Humanas, Rio Grande do Sul, 1996.
- POOLE M., *Princípios e Valores na Educação Científica*, Instituto Piaget, Lote 544 2º, Lisboa, 1995.
- ROSÁRIO, J. M., et al., *Curso de Extensão em Engenharia de Integração de Sistemas (Parte-3)*, Faculdade de Engenharia Mecânica – FEM, Departamento de Projeto Mecânico, Laboratório de Automação Integrada e Robótica – LAR/UNICAMP, Campinas, 1997.
- SALERNO, M. S., *Produção Integrada e Flexível e Processo Operatório: Notas Sobre Sindicatos e a Formação Profissional*, in *Trabalho e Educação*. p. 87-93, Papirus, Campinas – São Paulo, 1992.
- SANTOMÉ, J. T., *Globalização e Interdisciplinaridade O Currículo Integrado*, Artes Médicas, Porto Alegre, 1998.
- SCHLÜNZEN, K. Jr., *A Criação de um Ambiente de Aprendizagem Contextualizado, Baseado no Computador para a Formação de Recursos Humanos em Empresas Enxutas*. Tese de Doutorado, FEEC/UNICAMP, Campinas, SP, 2000.
- STAHL, M. M., *Caracterização do software educacional: Subsídios para seu desenvolvimento*. Rio de Janeiro Publicações Técnicas. COPPE/UFRJ, 1990.
- TAKIGUTHI, G., *Implementação de Um Sistema Integrado para separação de produtos Numa Linha Automatizada de Produção a Partir do reconhecimento de Cores*. Relatório de Iniciação Científica, Universidade Estadual de Campinas – UNICAMP, Faculdade de Engenharia Mecânica – FEM, Departamento de Projeto Mecânico, Laboratório de Automação Integrada e Robótica – LAR/UNICAMP, Campinas, 2001.

- TAULIE, J. R., *Automação Microeletrônica e Competitividade: Tendências no Cenário Internacional* in Hubert Schmitz & Ruy Quadros C. Org, *Automação Competitividade: A Experiência Internacional*, Hucitee, São Paulo, 1988.
- TERTRE, C., *Technologie, Flexibilité, Emploi: Une approche sectorielle du post-taylorisme*. L'Harmattan, Paris, 1989.
- TOUSSAINT, C., *Synthèse et Analyse de Graficets In Tégrées et Assistées par Logiciel - SAGITAL*, Manuel d'Utilisation, Version 2.00, Centre d'Etudes Supérieures des Techniques Industrielles - CESTI, Laboratoire d'Ingénierie Intégrée des Systèmes Industriels - LIISI, Paris, 2000.
- ULLRICH, R. A., *Robótica Uma Introdução, O Porquê dos Robôs e seu Papel no Trabalho*. Editora Campos, Rio de Janeiro, 1987.
- VALENTE, J. A., LEGO Logo in a Lean Factory, *Logo Update*. The Logo Fundatiom Newsletter, volume 5, number 2, 1997.
- VALLIM, M. B. R., *Em Direção À Melhoria Do Ensino Na Área Tecnológica: A Experiência de Uma Disciplina de Introdução à Engenharia de Controle e Automação*, dissertação de mestrado. Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Santa Catarina, Florianópolis, 2000.
- VYGOTSKI L. S., *Pensamento e Linguagem*, Martins Fontes, São Paulo, 1991.

TELA PRINCIPAL OU TELA DE INTRODUÇÃO DO ROBOLAB

Ao iniciar o Robolab, uma tela de introdução é aberta, apresentando duas opções:

Administrador e Programador.



A opção **Administrador** serve para fazer as configurações do RCX e do software Robolab

A opção **Programador** serve para abrir a tela do Menu Principal, a partir da qual pode-se optar em programar na fase Piloto ou na fase Inventor.

Também faz parte da tela principal os seguintes botões: **Sobre** (*About*), **Sair** (*Quit*) e **Ajuda** (*Help*).

Sobre mostra os créditos para o software ROBOLAB.

Sair fecha o ROBOLAB.

Ajuda fornece ajuda, descrevendo os itens para a qual o cursor estará apontando.

1. ÁREA ADMINISTRADOR (ÁREA DE CONFIGURAÇÕES)

A opção **Administrador** possui três sub-seções: Manutenção básica e informações, Configurações do Robolab e Configurações do RCX.

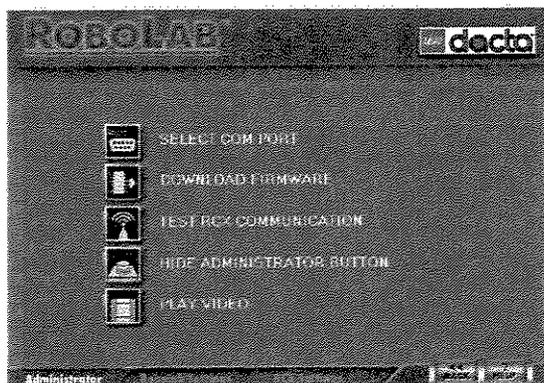
1.1 - Manutenção básica e Informações, composta de:

a) *Selecionar a Porta COM* – Seleciona a porta onde a comunicação entre o computador e o bloco RCX™ é realizada.

b) *Realizar a transferência do Firmware* - o *firmware* é o sistema operacional que controla o RCX™. Para transferir os programas elaborados no computador para o RCX, é necessário transferir primeiro o *firmware*. O *firmware* fica na memória do RCX™ até que as pilhas deste RCX™ sejam trocadas. Uma vez feita a transferência do *firmware*, o RCX™ fica pronto para receber, interpretar e executar os programas.

c) *Testar a Comunicação com o RCX™* - Realiza um teste de comunicação para verificar se o RCX™ responde aos sinais enviados pelo computador por intermédio da torre Infravermelho. Para realizar o teste é preciso que a torre de infravermelho esteja conectada ao computador e que o RCX™ esteja ligado e próximo a torre (10 a 20 cm).

d) *Esconder Botão administrador botão*- retira o botão **Administrador** da tela principal. Ao ocultar o botão, somente as pessoas que conhecem o software poderão verificar ou alterar as configurações. Estando na tela Principal, a tecla de função 'F5' do teclado realiza esta mesma operação, ou seja, também oculta ou mostra o botão **Administrador**.



e) *Rodar Vídeo* – Exibe um vídeo sobre Robolab, após a instalação do mesmo. Para assistir o vídeo, é preciso que o CD Robolab esteja no drive de CD e seguir as instruções sobre como rodar o vídeo. Para sair do vídeo, enquanto ele está sendo apresentado, clique no botão **Parar** e pressione a tecla **ESC**.

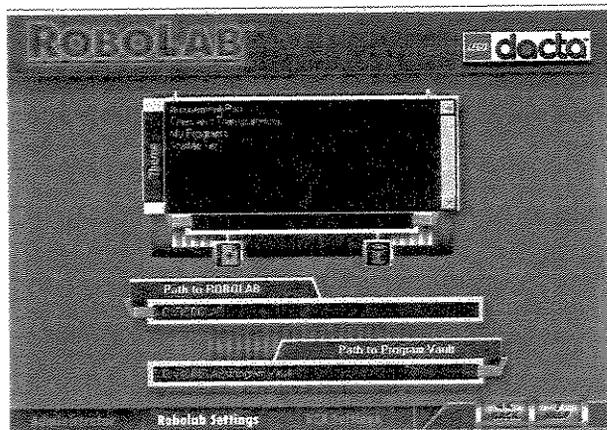
1.2 - Configurar Robolab → permite configurar os locais dos arquivos de programa e a criação ou exclusão de coleções. Nesta tela existe três objetos:

a) *Quadro de temas denominado “Tema”*:

Tema são coleções de programas gravados. Estes Temas são pastas que podem ser criadas para organizar arquivos. Abaixo do quadro existe uma caixa com um sinal de mais e uma outra caixa com uma lixeira.

Ao clicar sobre o sinal de mais você estará criando um novo Theme, o ROBOLAB irá pedir o nome deste novo Theme e ao clicar no botão Save, o nome que você digitou aparecerá dentro do quadro abaixo dos nomes que já existem. Existem dois themes que são padrão no ROBOLAB e são criados na instalação do software, são eles: “Amusement Park” e “My Programs”. Por padrão os programas são gravados em “My Programs”.

Ao clicar sobre a lixeira você estará excluindo um theme. Para realizar esta exclusão você precisa primeiro selecionar o nome do theme e depois clicar na lixeira, o ROBOLAB pede confirmação da exclusão. Caso não tenha nenhum nome selecionado, o Robolab avisa dizendo que o theme não existe. Os dois themes padrão não podem ser excluídos.



- b) *“Caminho para Robolab”*:
Mostra o “caminho” para o ROBOLAB no disco rígido. Este não pode ser mudado.

- c) *“Caminho para Galeria de Programas”*:

Mostra o “caminho” para os programas do usuário no disco rígido. Clicar sobre ele permite que você mude o local onde ficam os programas do usuário.

1.3 - Configurar RCX → Ao selecionar esta tela, o Robolab automaticamente estabelece a comunicação com o RCX. Caso este esteja desligado, ou fora do alcance do sinal emitido pela torre, aparece a mensagem de erro de comunicação. Se a comunicação for possível, o Robolab automaticamente configura os dados presentes nesta tela. Ao sair desta tela, o Robolab ajusta o RCX para a nova configuração.

Esta tela apresenta os seguintes dados:

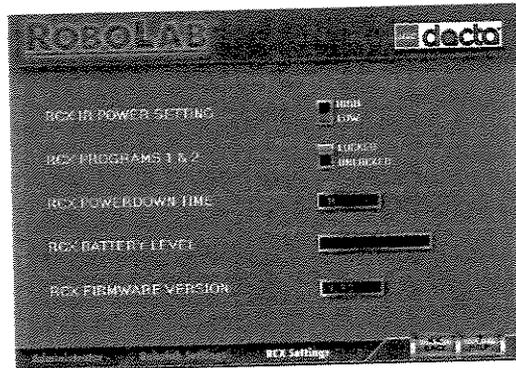
- a) *Ajuste de potência do Infravermelho (IV)*:

Regula a potência do IV do RCX em alto ou baixo. O valor alto só deve ser usada quando o RCX e o transmissor IV estiverem a mais de 30 centímetros. O valor baixo deve ser usado normalmente. A potência alta consome mais energia das pilhas.

O transmissor IV (torre de transmissão) também tem um ajuste de potência do IV; há um interruptor localizado na frente do transmissor para este fim.

- b) *RCX Programs 1 & 2*:

Bloqueie ou não as posições 1 & 2 da memória do RCX. Se as posições estiverem bloqueadas, ao tentar descarregar novos programas para essas posições, o ROBOLAB automaticamente descarrega para a posição 3 da memória.



c) *RCX Tempo de Desligamento:*

Define o tempo que o RCX ficará inativo antes de desligar. A variação é de 0 a 255 minutos. Por definição, o valor 0 significa que o RCX ficará sempre ligado, neste caso ele só se desligará ao pressionar o botão On-Off do RCX.

d) *RCX Carga das Pilhas:*

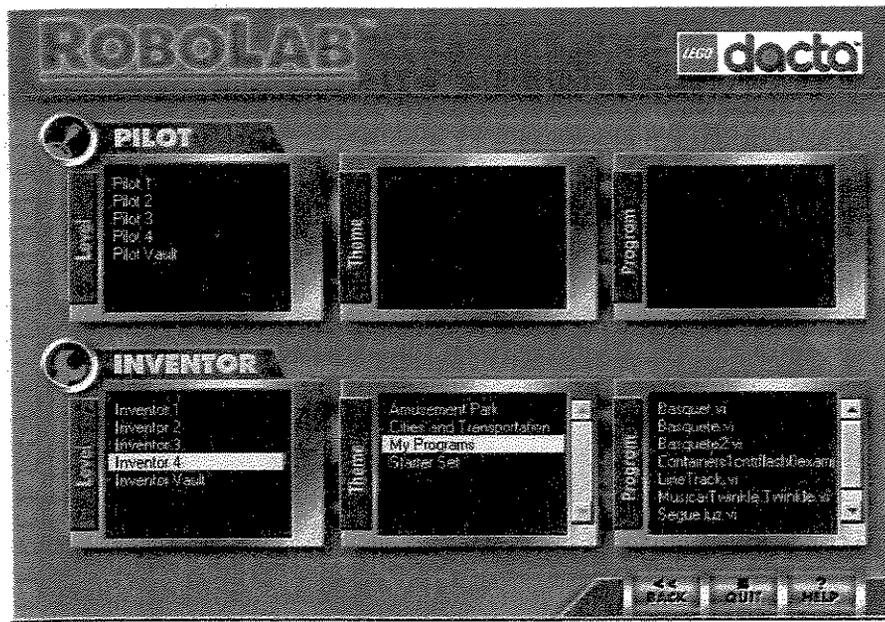
Mostra o nível atual da bateria do RCX.

e) *RCX Versão Firmware:*

Mostra versão do *firmware* em uso no RCX.

2. ÁREA DE PROGRAMAÇÃO

O software de programação do RCX no Robolab apresenta duas opções chamadas de **Pilot** e **Inventor** cada uma com 4 níveis.

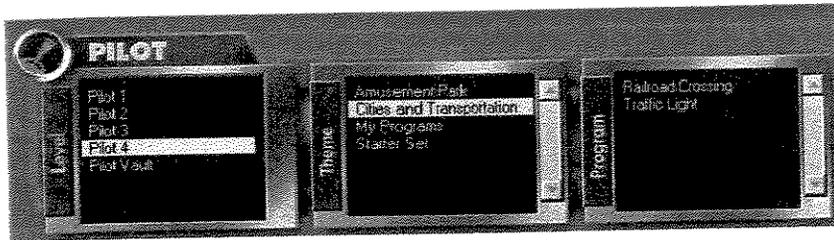


Cada seção apresenta 3 janelas: Level, Theme e Program. Em Level você seleciona o nível de trabalho. Um clique em Level faz a janela Theme mostrar os temas disponíveis. Um clique num tema faz a janela Program mostrar o conjunto de programas disponíveis para o tema e nível selecionados.

Um duplo-clique num programa, abre-o diretamente, enquanto que um duplo-clique num nível abre um programa padrão para aquele nível.

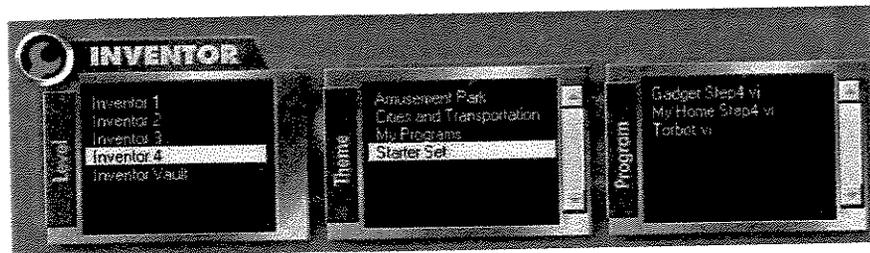
Pilot

A programação Pilot é a fase introdutória e apresenta 4 níveis, sendo o Pilot 1 o mais simples e o Pilot 4 o mais flexível.



Inventor

A programação Inventor é uma seqüência da fase Pilot e também apresenta 4 níveis, sendo o Inventor 1 o mais simples e o Inventor 4 o que apresenta total flexibilidade e possibilidades ilimitadas.



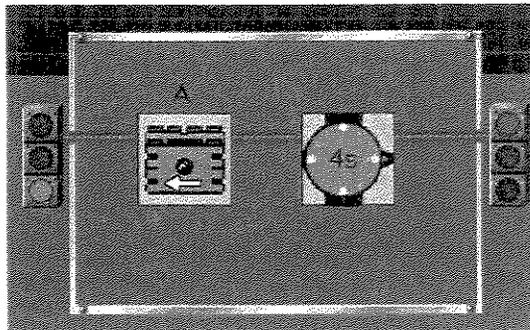
2.1 - PROGRAMAÇÃO PILOT

A fase Pilot compreende uma série de modelos com um formato fixo associado a eles. Trata-se de um modo eficiente de apresentar a seqüência lógica, pois é impossível modificar qualquer dos modelos de modo a criar um programa que falhe. O programa pode não fazer o que se espera, mas funcionará sempre e realizará exatamente a seqüência de comandos listados.

A programação Pilot usa uma interface fácil, do tipo Clique & Escolha, num modelo que pode ser alterado conforme suas necessidades de aprendizado.

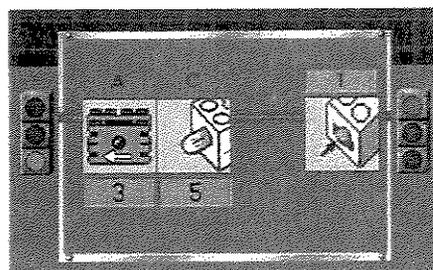
Abaixo segue uma descrição do programa padrão (modelo) encontrado em cada um dos níveis.

- O programa Pilot Nível 1 é uma tarefa simples que fornece potência máxima à Porta de Saída A do RCX durante um determinado tempo.

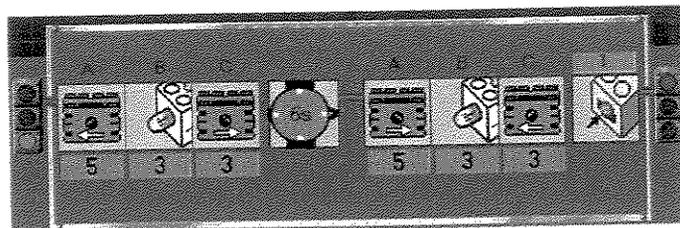


- O programa Pilot Nível 2 usa as Portas de Saída A e C do RCX. Os níveis de potência para cada porta pode ser ajustado. A potência é fornecida, ou por um tempo específico, ou até que o sensor de toque na Porta de

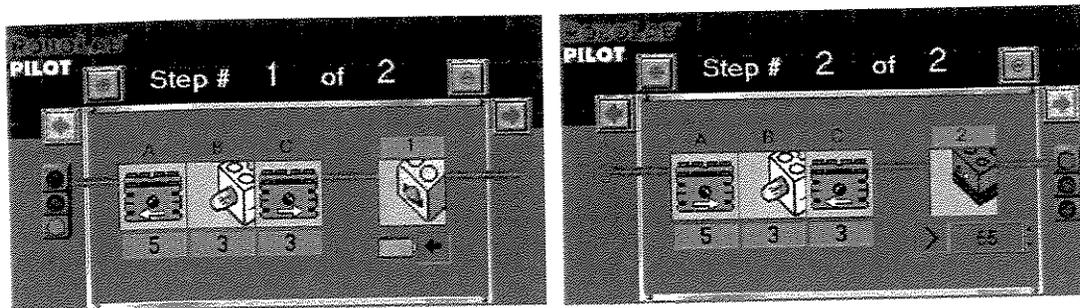
- Entrada 1 seja empurrada ou solta.



- O programa Pilot Nível 3 usa as Portas de Saída A, B e C. As portas podem receber potência de diferentes níveis. É introduzida a idéia de uma tarefa com dois passos. Todos os itens de um passo devem ser completados antes de iniciar o próximo. O primeiro passo liga o Motor A, a Lâmpada B e o Motor C por 6 segundos. Após esse tempo, começa o segundo passo. A Lâmpada B permanece ligada e os motores A e C invertem o sentido de rotação e funcionam até que o sensor de toque conectado na Porta 1 seja pressionado.



- O programa Pilot Nível 4 permite realizar um número ilimitado de passos seqüenciais. Entretanto, embora o programa possa conter vários passos, somente um passo fica visível na tela do computador de cada vez. O programa começa com dois passos: o primeiro liga o Motor A, a Lâmpada B e o Motor C até que o sensor de toque seja pressionado; o segundo, mantém a Lâmpada B ligada e inverte a direção dos Motores A e C, e funcionam até que o sensor de luz (luminosidade) da Porta 3 leia um valor maior que 55.



Nota:

- Cada ícone é um comando para o RCX fazer alguma coisa. Como exemplos de comandos temos: ligar o motor, esperar por um tempo, esperar que o sensor de toque seja pressionado.
- As informações que entram (input) nos comandos (como a quantidade de tempo ou onde o dispositivo está conectado) são chamadas de modificadores.

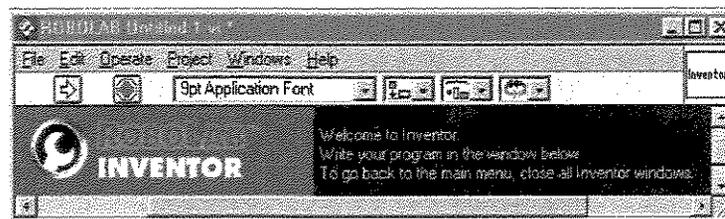
2.2 - PROGRAMAÇÃO INVENTOR

A fase Inventor usa os mesmos ícones de comandos da fase Pilot. Além disso, são acrescentadas várias opções de comando conforme o usuário avança pelos níveis. O nível Inventor é uma modalidade de uso que evidencia o potencial do RCX.

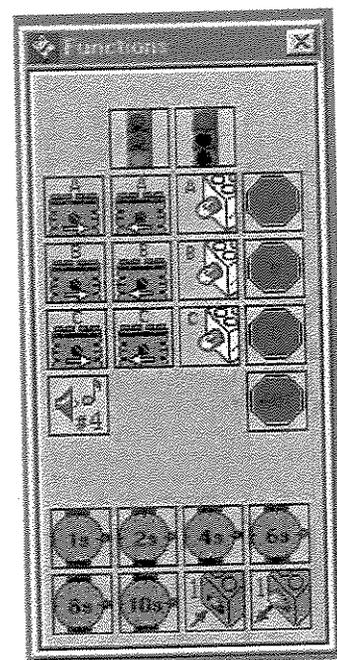
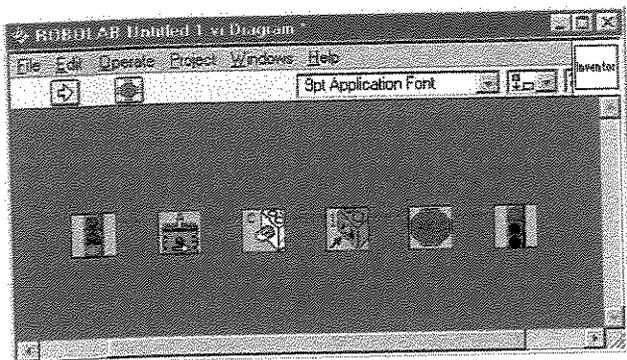
A interface de programação do Inventor é diferente da utilizada no Pilot. A interface é do tipo Pegar & Colocar.

Nesta interface, pode-se trabalhar com pelo menos 3 janelas.

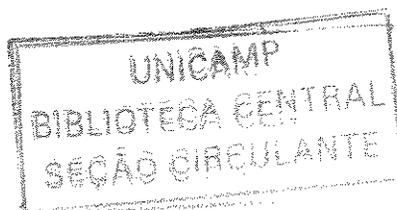
A janela Painel é usada na programação do Inventor. Ela precisa estar aberta para que o programa possa rodar.



A janela utilizada para programar é chamada de Diagrama e é nela que se cria as tarefas a serem executadas pelo RCX.

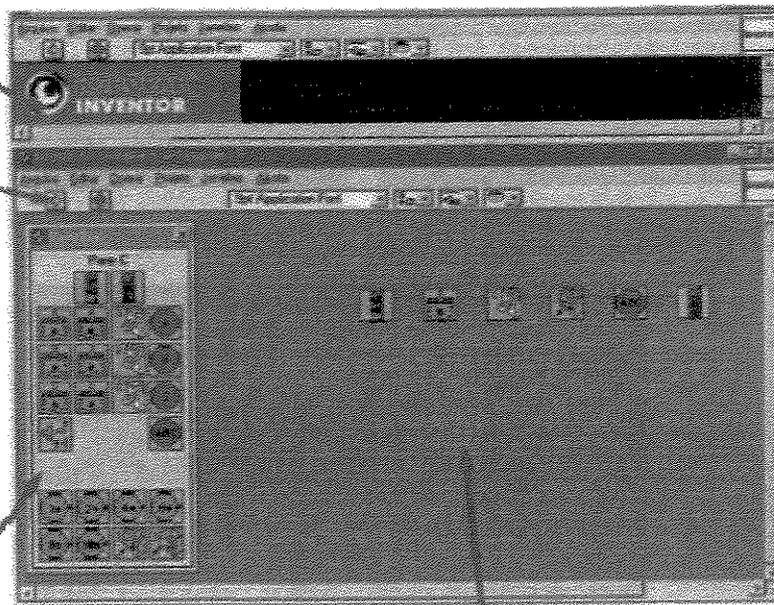


A outra janela é chamada de Paleta Functions (paleta de funções), a qual apresenta todos os comandos ou funções disponíveis para a montagem do seu programa. Esta paleta também contém os comandos de Espera, os de Estrutura, os Modificadores e outros mais.



Janela Diagrama

Botão Rodar



A Paleta **Funções** contém os ícones de comando usados para programar. Os ícones são pegos da paleta **Funções** e colocadas na janela **Diagrama**.

Os programas são criados na janela **Diagrama**.

3. - FERRAMENTAS

Durante a programação na fase Inventor, existe a necessidade de se trabalhar com algumas ferramentas:



Ferramenta Selecionar - Usada para pegar os ícones & colocar na janela Diagrama ou para mover os ícones.



Ferramenta Conectar
Usada para colocar ícones de comando na janela Diagrama



Ferramenta Texto - Usada para mudar valores ou acrescentar textos ou legendas ao seu programa.



Ferramenta Conectar - Usada para conectar os ícones na sequência lógica a ser executada.



Ferramenta Mudar Valores - Usada para mudar os valores numéricos.

Para trocar de ferramenta utilizar a barra de espaço ou a tecla Tab.

Barra de Espaço

Pressionar a barra de espaço faz o cursor alternar entre as ferramentas *Selecionar* e *Conectar*. Estas duas ferramentas são, provavelmente, as mais utilizadas.

Tecla Tab

Ao pressionar esta tecla o cursor circula entre as ferramentas *Selecionar*, *Colocar*, *Texto* e *Conectar*.

Tecla Delete

Apaga o item selecionado (ícone ou conexão) da janela Diagrama.

4. AJUDA

Caso seja necessário alguma, durante a programação, pode-se ativar a função ajuda a partir da barra de menu da janela de programação. A função ajuda fornece informações sobre o ícone de comando sobre o qual o cursor está. A ajuda faz uma descrição do comando, assim como os pontos de conexão da seqüência.

5. ÍCONES - LISTA DE CONSULTA

Início & Fim do Programa

| | | |
|---|------------|--|
|  | Início | No início do programa, exigido como o primeiro comando em qualquer programa Inventor. |
|  | Fim | No fim do programa, exigido como o último comando de cada tarefa no programa Inventor. |
|  | Parar A | Interrompe a potência da Porta A do RCX. |
|  | Parar tudo | Interrompe a potência das portas A, B e C do RCX. |

Saídas Simples

| | | |
|---|--------------------------------|---|
|  | Motor A (sentido horário) | Liga a Porta A do RCX no sentido horário com potência total. |
|  | Motor A (sentido anti-horário) | Liga a Porta A do RCX no sentido anti-horário com potência total. |
|  | Lâmpada A | Acende a Lâmpada conectada na Porta A do RCX com potência total. |

Saídas Gerais



Lâmpada

Acende a lâmpada.

Padrão: Potência total nas Portas A, B e C.



Motor (sentido horário)

Liga o motor no sentido horário.

Padrão: Potência total nas Portas A, B e C.



Motor (sentido anti-horário)

Liga o motor no sentido anti-horário.

Padrão: Potência total nas Portas A, B e C.



Direcionar

Direciona a potência para as portas especificadas, invertendo a direção anterior.

Padrão: Portas A, B e C.



Tocar Som

Toca um som no RCX. Os sons disponíveis são:

1-Clique de chave

4-Silvo ascendente (Padrão)

2-BeepBeep

5-Bzzzz

3-Silvo descendente

6-Silvo ascendente rápido

Esperar por? (Wait For)



Espera 1 segundo

Espera 1 segundo antes de continuar o programa.



Espera por tempo

Espera por um tempo especificado.

Padrão: 1 segundo.



Espera tempo aleatório

Espera por um tempo aleatório. Tempo entre 1 e 8 segundos.



Espera empurrar

Espera até que o sensor de toque seja pressionado.

Padrão: Porta de Entrada 1.



Espera não pressionado

Espera até que o sensor de toque não seja pressionado.

Padrão: Porta 1.



Espera por luz

Espera até que o sensor de luz leia um valor mais brilhante que o número especificado.

Padrão = 55.



Espera por escuro

Espera até que o sensor de luz leia um valor mais escuro que o número especificado.

Padrão = 55.



Mais claro

Espera até que o sensor de luz leia um valor maior que o atual.

Padrão: Porta 1, diferença de valor = 5.



Mais escuro

Espera até que o sensor de luz leia um valor menor que o atual.

Padrão: Porta 1, diferença de valor = 5.

Modificadores



Entrada 1

Anexe este modificador a um comando para selecionar a Porta de Entrada 1.



Saída A

Anexe este modificador a um comando para selecionar a Porta de Saída A.



Potência Nível 3

Anexe este modificador a um motor ou lâmpada para ajustar a potência no nível 3.



Constante Numérica

Anexe este modificador a um sensor ou temporizador para ajustar o valor da constante.



Valor do *Container* vermelho

Retorna o valor do *container* vermelho

| | | |
|---|--------------------------------|--|
|  | Container vermelho | Anexe-o a um comando <i>container</i> , para selecionar o <i>container</i> vermelho. |
|  | Número ? | Um número aleatório entre 1 e 8. |
|  | Valor da Porta 1 | Retorna o valor da Porta de Entrada 1. |
|  | Temporizador vermelho | Anexe-o a um comando de temporizador para selecionar o temporizador vermelho. |
|  | Valor do Temporizador vermelho | Retorna o valor do temporizador vermelho. |
|  | Valor da Carta | Retorna o valor da carta (carta). |

Música

| | | |
|---|------------------|--|
|  | Nota musical | Toca notas musicais no RCX. Padrão: semínimas na escala normal. |
|  | Pausa musical | Insere uma pausa na música. |
|  | Duração musical | Especifica quanto tempo uma nota deve tocar. |
|  | Uma oitava acima | Anexado a um comando musical, eleva o tom em uma oitava ou mais se mais de um for anexado. |

Estruturas

| | | |
|---|-----------------------------|---|
|  | Condição do Sensor de Toque | Faz o programa escolher entre dois caminhos, dependendo dos dados do sensor de toque. Padrão: Porta 1. |
|  | Junção | Usada sempre com uma condição para unir os dois caminhos da condição. |
|  | Começar Tarefa | Começa uma nova tarefa. Permite realizar tarefas múltiplas simultaneamente. O RCX permite um máximo de 10 tarefas. Começa um estrutura de "loop". |
|  | Começar Loop | Padrão: 1 vez. |
|  | Fim do Loop | Pula de volta ao início do "loop". |
|  | Jump (pulo) | Faz o programa pular para um lugar específico da seqüência. |
|  | Land (pouso) | Indica o lugar da seqüência onde o comando <i>Jump</i> vai pular. |

Container

| | | |
|---|--------------------------|--|
|  | Acrescentar ao Container | Adiciona um número ao <i>container</i> . Padrão: adiciona 1 ao <i>container</i> vermelho. |
|  | Retirar do Container | Subtrai um número do <i>container</i> . Padrão: subtrai 1 do <i>container</i> vermelho. |
|  | Encher Container | Ajusta o <i>container</i> para um certo valor. Padrão: Container vermelho com valor 1. |
|  | Container Toque | Ajusta o <i>container</i> para o valor do sensor de toque. |
|  | Container Temporizador | Ajusta o valor do <i>container</i> do temporizador. Padrão: <i>container</i> vermelho com o valor do temporizador vermelho. |

RCX para RCX



RCX envia carta

Envia carta (mensagem) para outro RCX. A mensagem deve ser um número entre 1 e 255.
Padrão: envia o número 1.

Reset



Zera *Container*

Zera o valor do *container*.
Padrão: *container* vermelho para zero.



Zera Relógio

Zera o valor do despertador.
Padrão: despertador vermelho para zero.



Zera Sensor de Rotação

Zera o sensor de rotação.
Padrão: Porta 1.



Zera Caixa postal

Zera o valor da caixa postal do RCX.

Espera Avançada



Espera Aumento de Temperatura

Espera até que a temperatura esteja mais alta que o número especificado.
Padrão: 30 ° Celsius na Porta 1.



Espera Rotação

Espera até que o valor do sensor de rotação seja maior que o número de rotações especificadas.
Padrão: 16 (uma rotação) na Porta 1.



Espera por Posição

Espera até que o valor do sensor de rotação seja maior que o ângulo especificado em qualquer direção.
Padrão: 180 graus na Porta 1.



Espera Container

Espera até que o *container* esteja igual ao número especificado.
Padrão: *container* vermelho igual a 1.



Espera Despertador

Espera até que o temporizador seja igual ao número especificado.
Padrão: despertador vermelho igual a 1. Precisa zerar o despertador primeiro!

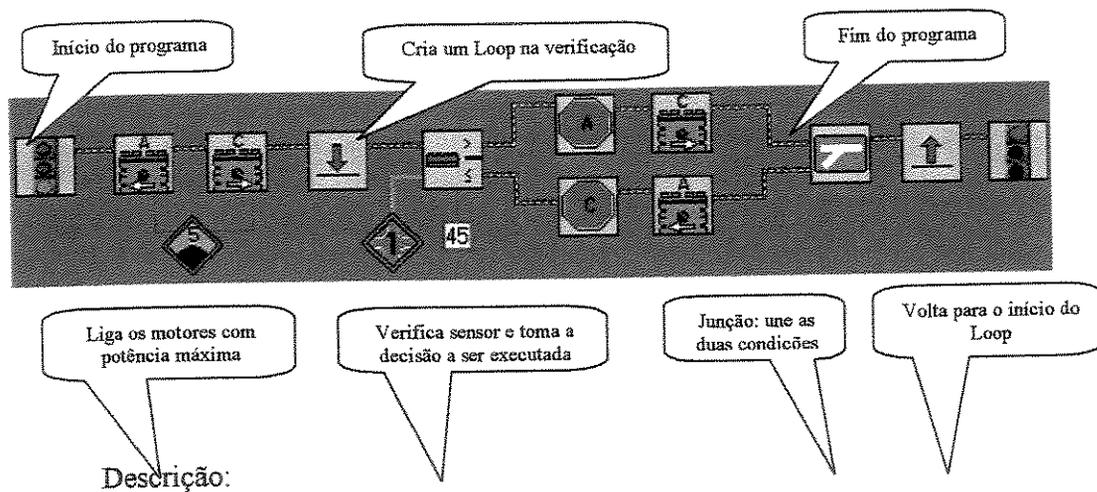


Espera Carta

Espera até que a mensagem do outro RCX seja recebida.

6. - PROGRAMA EXEMPLO

Abaixo, utilizando a mesma idéia do LineTrack, apresentamos a sua versão construída no Robolab. Verifique que esta montagem é mais simples e seu entendimento também.



1. Ao ser acionado, liga os motores com potência máxima e nos sentidos indicados nos próprios ícones;

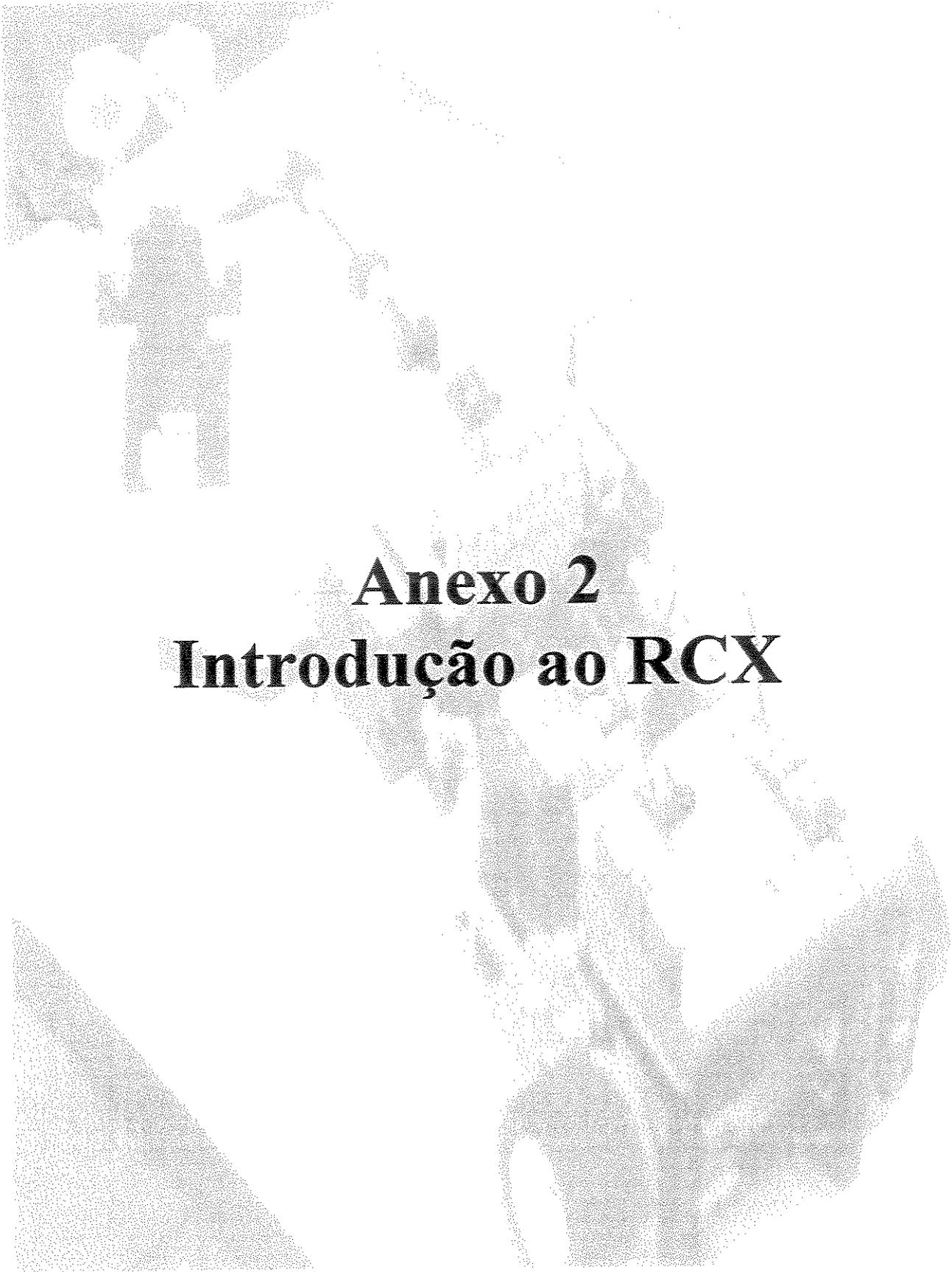
2. Inicia um Loop que faz a verificação constante do sensor de luz conectado na porta 1 (entrada). Caso a leitura seja maior que 45, aciona apenas o motor C, caso contrário, aciona apenas o motor A;
3. Volta ao item anterior fechando o Loop e começa novamente a realizar a verificação.

Discussão:

Há alguma forma de fazer o carro seguir uma trajetória demarcada no chão, usando o sensor de luz, sem que ele faça zig-zag? Como?

No processo de construir robôs é possível aprender um pouco mais sobre o comportamento destes objetos. A elaboração de programas que os controlam utiliza comandos que acionam, por intermédio do RCX, luzes, motores e sensores conectados nos robôs que se deslocam no espaço.

A atividade de robótica pedagógica permite professores e alunos descrever, refletir e depurar o que está sendo montado e explorar conceitos científicos, envolvidos na montagem dos objetos como conceitos de física (atrito, força, velocidade), conceitos de matemática (fração, proporção), conceitos de engenharia (construção do objeto, mecanismos), conceitos de programação (elaboração de procedimentos, comandos da linguagem) e de *design* (estética, funcionalidade). Este tipo de atividade propicia ao professor o resgate das relações interdisciplinares que normalmente permeiam o ambiente LEGO-Logo (D'Abreu, 1995).



Anexo 2

Introdução ao RCX

Introdução ao RCX

Esta apostila tem como objetivo básico apresentar ao aluno os comandos do tijolo programável RCX, um dos principais dispositivos utilizados para se trabalhar no ambiente de Robótica Pedagógica.

1. Robôs

Uma máquina pode ser considerada um robô caso seu comportamento possa ser programado. Esta definição é bastante ampla, não ficando restrita apenas aos conceitos de robô do tipo humanoíde. Dispositivos tais como videocassete, microondas, equipamentos de áudio, etc. são máquinas programáveis, portanto, podem ser considerados robôs. Os robôs possuem 5 componentes fundamentais:

1. **Cérebro:** que controla as ações do robô e responde a estímulo de sensores. Geralmente o cérebro é uma CPU.
2. **Corpo:** um chassi físico que simplesmente envolve as demais peças do robô e as mantém juntas.
3. **Atuadores:** que permitem com que o robô se movimente. Geralmente são motores, garras, juntas, pistões, lâmpadas, etc..
4. **Sensores:** que fornecem informação ao robô sobre o meio em que este se encontra. Podem ser de toque, temperatura, luminosidade, posição, etc..
5. **Fonte de Energia:** Supre a energia necessária para o funcionamento do robô.

2. LEGO-MINDSTORMS

MINDSTORMS é o nome de uma das linhas de produtos comercializados pela LEGO. O “carro chefe” dessa linha de produtos é o Robotic Invention System (RIS), que é um kit contendo peças que permitem a construção de robôs. Com este kit é possível construir robôs de maneira simples e rápida.

O kit RIS possui um “tijolo”, como mostra a figura 1, que pode ser considerado o cérebro do robô, o qual é chamado de Robotic Command Explorer (RCX). O RCX é na verdade um pequeno computador com o formato de um tijolo LEGO. Além disso, o kit inclui também sensores de toque, sensores de luz e motores. Enfim o RIS possui uma série de outras peças que, em conjunto com o RCX, podem ser utilizadas para construir um robô, que se locomove e, por meio de sensores, percebe o ambiente ao seu redor.



Figura 1: Tijolo Programável RCX

2.1 Constituição do RCX

O RCX (veja figura 1) possui as seguintes características:

Saídas

Três portas de saída, identificadas pelas letras **A**, **B** e **C**. Os atuadores do robô, motores ou luzes, podem ser conectadas a estas portas.

Entradas

Três portas de entrada, identificadas pelos números de **1**, **2** e **3**. Sensores de toque ou de luz podem ser conectados a estas portas, permitindo com que o RCX perceba o ambiente onde ele se encontra.

Tela

Faz parte do RCX uma pequena **tela** de cristal líquido. Esta tela exibe/mostra informações úteis ao usuário do RCX, tais como o valor da leitura dos sensores e estado das portas de saída.

Som

O RCX é capaz de produzir alguns *bips* em diferentes frequências através de um pequeno alto-falante nele embutido. Existem alguns sons já pré-programados na memória do RCX que são emitidos antes dele executar algumas tarefas, como por exemplo, carregar um programa.

Botões do Painel

Existem 4 botões no RCX: **On-Off**, **View**, **Prgm**, **Run**. Estes botões possibilitam respectivamente, ligar e desligar o RCX, visualizar os estados atuais dos sensores, selecionar o programa que desejamos executar, iniciar e interromper a execução de um programa.

Comunicação Infravermelho

O RCX se comunica com o microcomputador por intermédio de uma conexão, sem fio, feita a partir de uma torre que emite sinais na faixa de infravermelho (veja figura 2). Esta torre, que se conecta a porta serial do microcomputador, permite também a comunicação entre dois RCX.

2.2. Software

Um CD-ROM acompanha o kit RIS. Ele contém o software destinado à programação dos robôs. Uma vez escrito um programa, para controle de um determinado dispositivo robótico, este programa tem que ser carregado no RCX, através da torre, para ser executado.

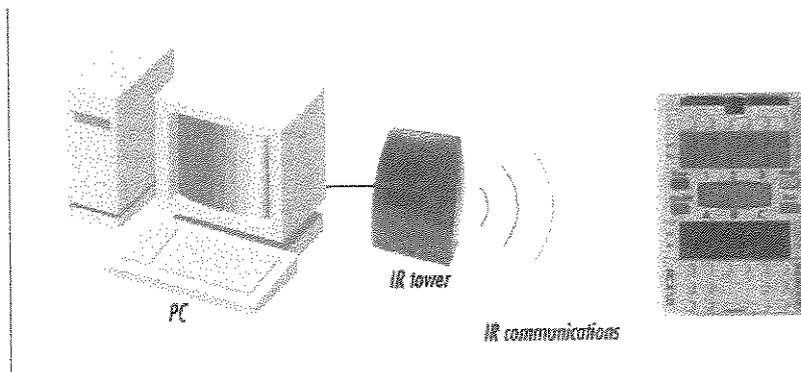


Figura 2: Comunicação entre microcomputador e RCX

A figura 2 apresenta o esquema básico de comunicação entre um microcomputador e o RCX. A implementação de um robô utilizando RCX consiste basicamente de quatro passos.

1. **Construir robô propriamente dito.**
2. **Escrever um programa utilizando as ferramentas de software existentes no microcomputador.** Aqui, escrever programas para o controle de dispositivos robóticos significa, agrupar segundo uma certa lógica, comandos que permitem acionar motores sensores e lâmpadas.
3. **Carregar o programa no robô.**
4. **Executar o programa.**

2.2.1 Programando o RCX



O bloco *program* inicia todos os programas elaborados para o RCX. Os blocos de comandos, que farão parte de um determinado programa, deverão ser anexados a este bloco. A seqüência de execução de um programa se dará de cima para baixo.

IMPORTANTE: Clicando sobre o *program* bloco com o botão direito do mouse, é possível salvar, apagar ou fazer o *download* de programas.

Os blocos de comando são aqueles blocos que informam ao RCX o que fazer. Eles possuem a coloração esverdeada. Alguns desses comandos, como por exemplo *on*, *off*, *set power*, *set direction* e *reverse direction* controlam as portas de saída A, B e C, onde os atuadores são conectados no RCX.

Nas portas de entrada 1, 2 e 3 poderão ser conectados sensores de toque, de rotação, de temperatura e de luz.



ON: aciona as portas A, B e/ou C do RCX.



ON FOR: aciona a porta A, B e/ou C por um determinado intervalo de tempo.



OFF: desliga as portas A, B e/ou C.



SET POWER: permite a mudança da potência do motor, sendo o valor mínimo 1 e o máximo 8. A porta é sempre ativada na potência máxima (8), ao menos que seja definido, anteriormente, um outro valor.



SET DIRECTION: define o sentido em que o motor deve girar. Sentido horário ou anti-horário.



REVERSE DIRECTION: permite a inversão de rotação do motor.



WAIT: faz uma pausa no RCX por um certo intervalo de tempo antes que ele inicie o próximo comando. O tempo é dado em décimos de segundos.



BEEP: permite a emissão sonora de um dos seis tipos de sons existentes.



TONE: permite ao RCX para emitir um tom, com uma determinada frequência, por um certo intervalo de tempo.



RESET COUNTER: reinicia a contagem do contador.



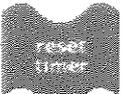
ADD TO COUNTER: incrementa de 1 a contagem feita pelo contador interno do RCX.



RESET ROTATION: redefinir a posição do ponto 0 do sensor de rotação. Este sensor sempre define como sendo o ponto 0 a posição na qual o RCX iniciou a execução do programa.



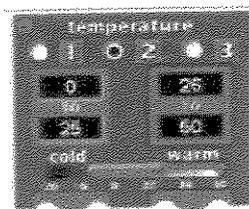
RESET MESSAGE: atualiza a mensagem no RCX receptor, permitindo a execução de comandos referentes à mensagem mais atual.



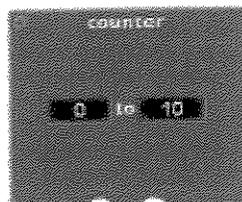
RESET TIMER: inicia a contagem a partir do 0 toda vez que o RCX começa a executar o programa. Usando-se este comando, é possível reiniciar a contagem em qualquer outra parte do seu programa.



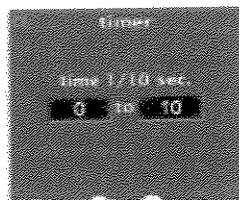
SEND TO RCX: envia mensagem de um RCX para outro RCX para a execução de um determinado comando ou uma sequência deles.



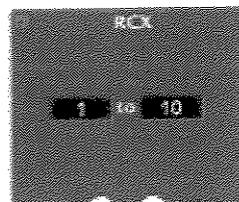
SENSOR DE TEMPERATURA: possui duas faixas de temperatura definidas pelo programador: “frio” e “quente”.



CONTADOR: executa os comandos anexados ao bloco “counter” (azul) quando o contador registra um número dentro do intervalo definido pelo programador. Para se utilizar o contador adequadamente, deve-se utilizar o bloco de comando “add to counter” (verde), que incrementará ao contador de uma unidade de contagem.



TIMER: inicia a contagem assim que o programa carregado no RCX, começa a ser executado. Os comandos anexados ao bloco “timer” serão iniciados dentro do intervalo estabelecido pelo programador.

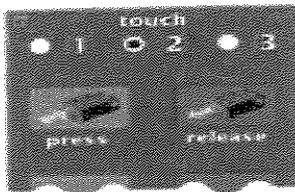


RCX: estabelece a comunicação entre dois ou mais RCX. Com isso, um RCX(transmissor) pode enviar mensagem para outro RCX (receptor) para que este inicie um certo comando ou seqüência deles. Os comandos anexados a este bloco serão executados assim que o transmissor enviar uma mensagem (“send to RCX”) para o receptor.

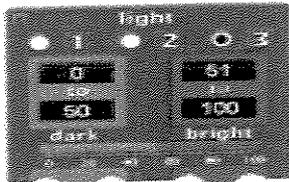


Monitores de Sensores

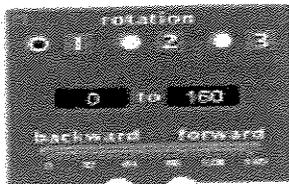
Por intermédio de sensores, o RCX consegue “perceber” o ambiente onde se encontra. Por exemplo, através do sensor de luz, o RCX “percebe” a variação de luminosidade e, assim, permite ao programador que associe os fenômenos *escuro* e *claro* com acionamentos de motores, beeps ou acendimento de lâmpadas. Os blocos referentes aos sensores possuem coloração azulada. Usa-se sensores de toque, de temperatura, de ângulo ou de rotação quando o programador deseja que, devido à mudança de estado do sensor, o sistema execute determinados comandos.



SENSOR DE TOQUE: possui dois estados: pressionado (press) e Não-pressionado (release).

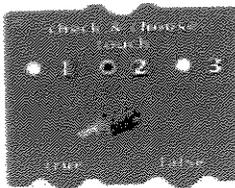


SENSOR DE LUZ: possui dois estados, definidos pelo programador, “escuro” e “claro”. Por exemplo, luminosidade 30% (dark) e 70% (bright).

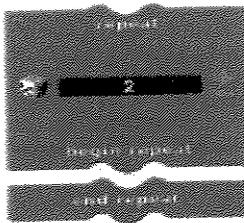


SENSOR DE ROTAÇÃO: permite que o eixo acoplado ao sensor mantenha uma certa rotação dentro de um intervalo, com incremento de 16 unidades de contagem por rotação.

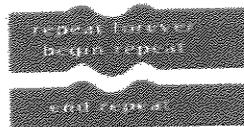
Estes blocos permitem que o programador escolha diferentes caminhos, que devem ocorrer durante a execução, para as partes do programa, usando como auxílio sensores, contadores, *timers* e mensagens para o controle do RCX. Estes blocos possuem coloração avermelhada.



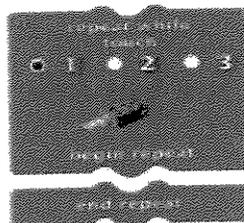
CHECK & CHOOSE: segue um dos caminhos determinados de acordo com a condição estabelecida pelo programador.



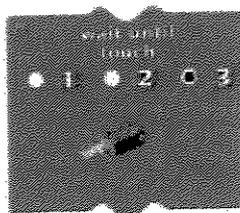
REPEAT: executa um determinado número de vezes os comandos anexados entre “*begin repeat*” e “*end repeat*” serão. Este número pode ser estabelecido pelo programador ou configurado no modo randômico “*random*”.



REPEAT FOREVER: executa dentro de um loop infinito os comandos anexados entre “*begin repeat*” e “*end repeat*”.



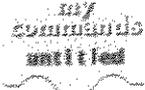
REPEAT WHILE: executa até que a condição não seja verdadeira os comandos anexados entre “*begin repeat*” e “*end repeat*”.



WAIT UNTIL: faz uma pausa na execução do programa até que a condição estabelecida pelo programador seja satisfeita. Quando a condição for verdadeira, os comandos anexados serão executados.

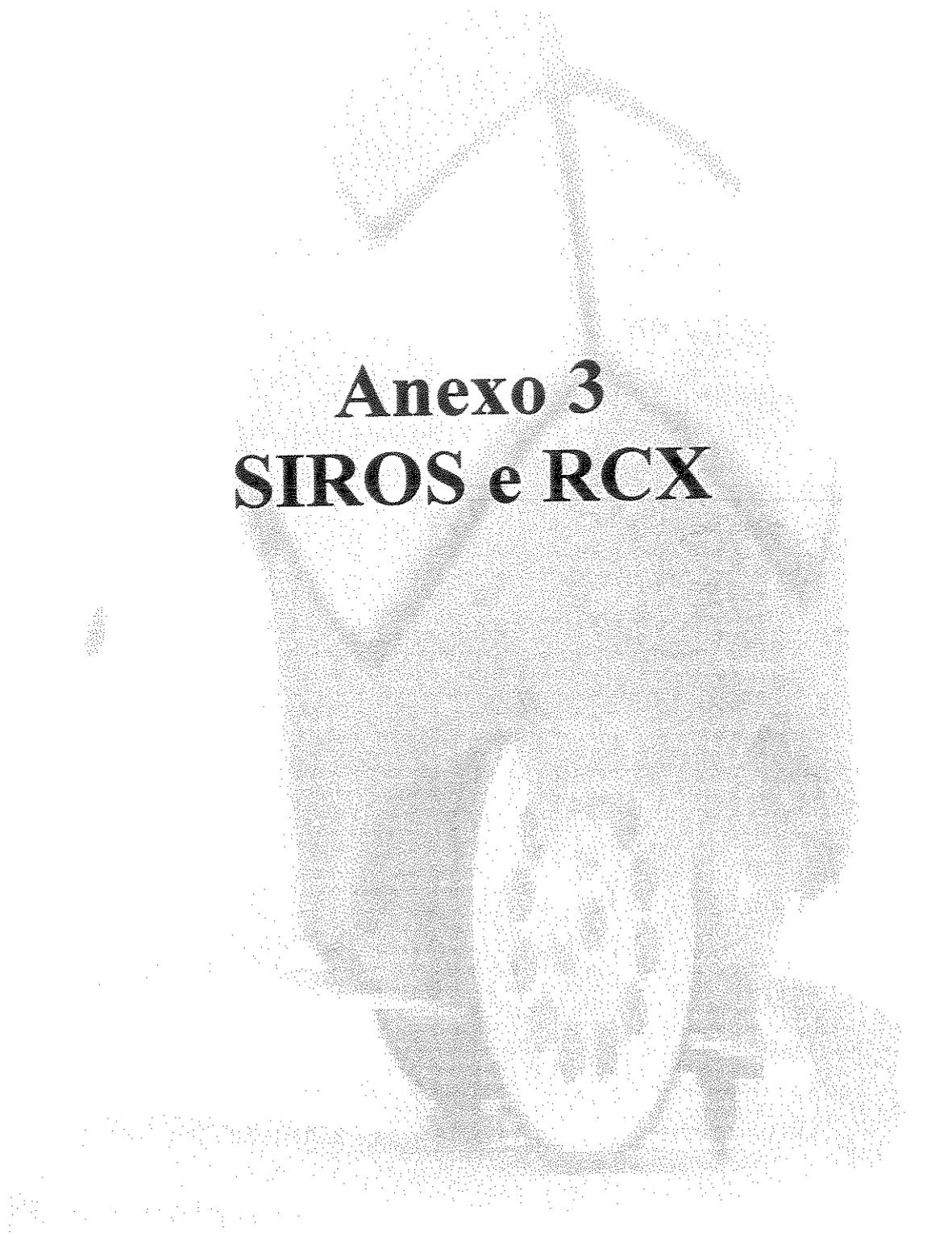


Criando Comandos (SubProcedimentos)
Possuem coloração amarelada.



MY COMMANDS: permite ao programador criar seus próprios comandos. Estes comandos são subprocedimentos que, uma vez criados passarão a ser mais um comando do RCX.

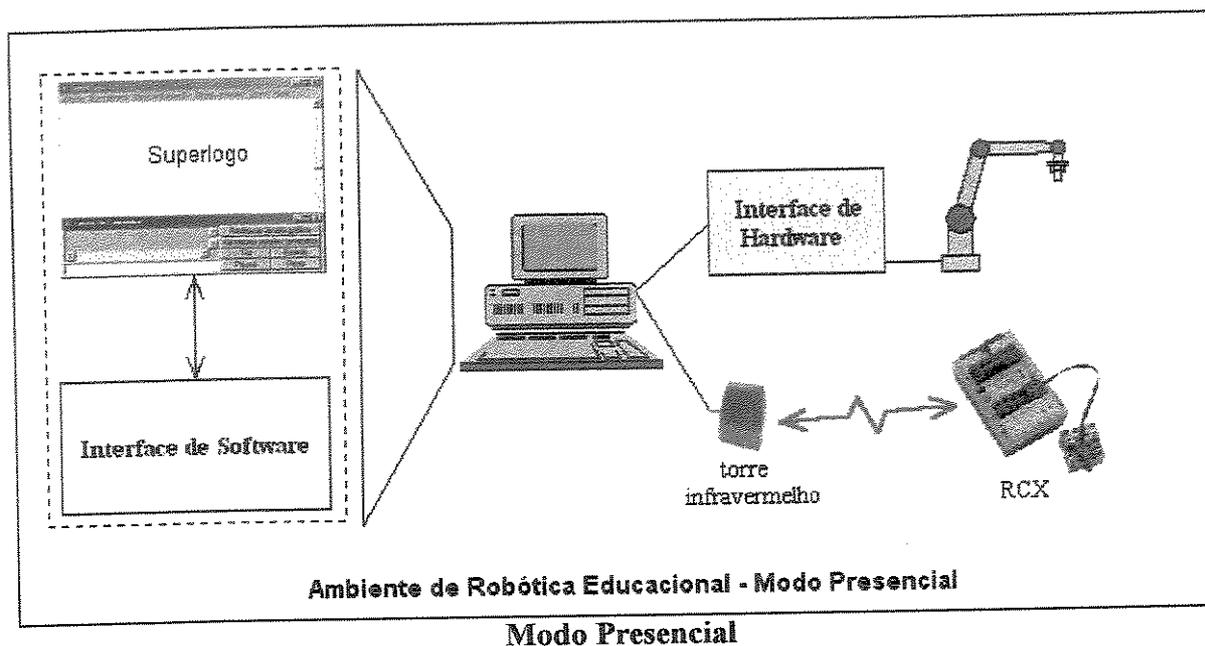




Anexo 3
SIROS e RCX

Ambiente de Robótica Educacional - SIROS Sistemas Robóticos com SuperLogo

No desenvolvimento do Ambiente de Robótica Educacional SIROS foram estudadas as diferentes tecnologias e ambientes existentes e baseado nelas foi desenvolvido um ambiente de robótica de baixo custo que propicia condições para manusear de forma mais concreta conceitos nas mais variadas áreas do conhecimento.



Para controlar os dispositivos no modo presencial foi necessário integrar diversos módulos de software e hardware que são descritos a seguir:

Interface de Software – O SuperLogo não dispõe dos recursos de software necessários para permitir a comunicação com o LEGO RCX. Assim para que seja possível controlar o LEGO RCX e acessórios conectados, como motores e sensores é necessária a criação de uma interface de software que atuará entre o SuperLogo e o LEGO RCX.

A interface de Software além de prover a comunicação entre o SuperLogo e o LEGO RCX provê meios para diagnóstico, configuração, e download do firmware. Para o desenvolvimento da Interface de Software foi utilizada a linguagem de programação Microsoft Visual Basic versão 6.0.

Procedimentos SuperLogo - O SuperLogo irá se comunicar com o LEGO RCX por meio da Interface de Software. Para que isto ocorra de forma transparente e preservando a estética LOGO um conjunto de procedimentos foram implementados.

Interface de Hardware –Foi desenvolvido com materiais genéricos e facilmente encontrados no comércio de material eletrônico o projeto de uma Interface de Hardware, esta interface pode ser uma opção de baixo custo para a implementação de projetos de dispositivos robóticos que envolvam o acionamento de motores, lâmpadas e leitura de sensores.

Procedimentos SuperLogo para a interface de hardware - Foram implementados procedimentos com o SuperLogo para controlar o acionamento dos dispositivos conectados a Interface de hardware de forma transparente mantendo a estética LOGO.

Ambiente de Telerobótica

Para possibilitar o controle e monitoramento de robôs remotamente utilizando a estrutura de rede da internet foi necessário implementar e integrar um conjunto de módulos de software e hardware .

O modelo utilizado é o Cliente-Servidor (*figura 2*) e os seguintes recursos são utilizados:

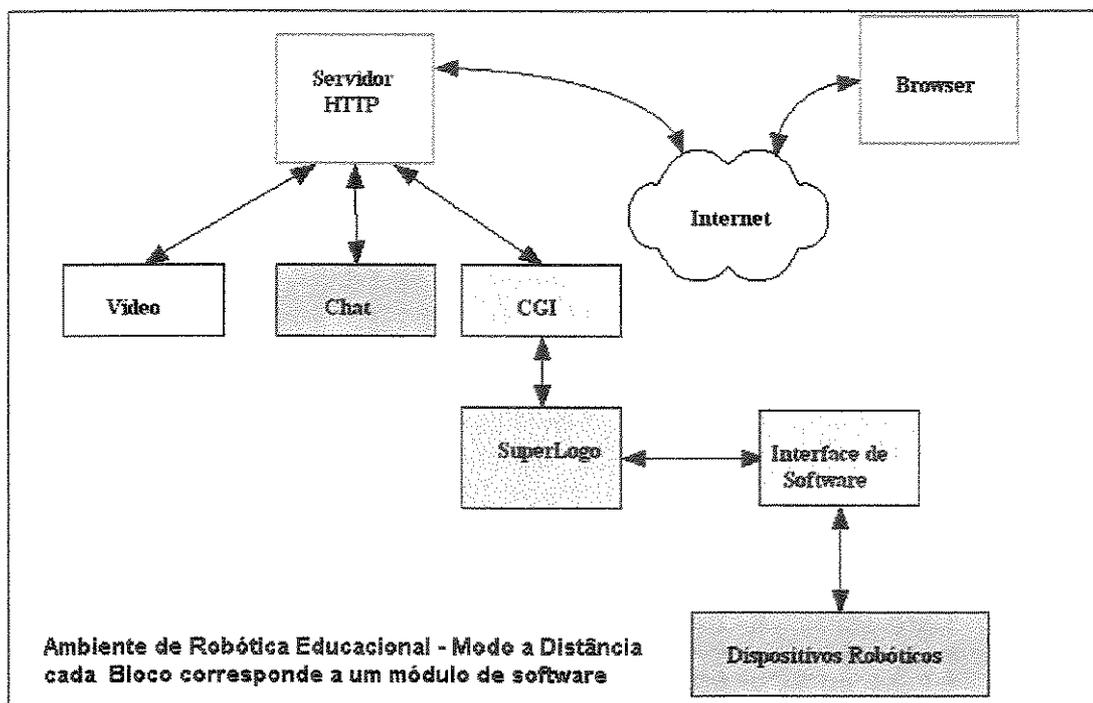


figura 2

Software com interface CGI (Common Gateway Interface) – É responsável por receber os comandos inseridos em um campo de formulário no browser do cliente, decodificá-lo e disponibilizar para utilização pelo SuperLogo, Interface de Software, LEGO RCX e Interface de Hardware.

Software de Chat - É utilizado para permitir a comunicação entre o usuário e o monitor que está no laboratório onde se encontram os dispositivos robóticos sob controle.

Software para apresentação de Vídeo - Este módulo é responsável por apresentar ao usuário que controla o dispositivo robótico, imagens que mostrem o resultado da sua ação. O software utilizado é o WEBCAM 2.25, que captura o sinal de uma câmera de vídeo conectada ao PC e apresenta ao usuário na WEB por meio de uma pagina HTML que contém um applet Java.

Servidor WEB –Os serviços disponibilizados no ambiente de telerobótica são suportados pelo servidor HTTP Xitami, que provê recursos para utilização dos módulos CGI, páginas WEB e apresentação de vídeo.

Site na WEB - A página web é a interface com o usuário remoto e tem a função de integrar os dados de controle do robô ao software CGI, apresentar o vídeo, chat e conter a documentação e ajuda ao usuário.

Procedimentos SuperLogo para RCX

```
aprenda contadorsensorluz :vsensor :vfaixa
atr "faixa :vfaixa
atr "sensorl :vsensor
rcxsetasensorluz :sensorl
atr "contador 0
atr "vsl 0
atr "passou 1
ativetemporizador 10 500 [
    rcxlrsensorluz :sensorl
    atr "vsl :valorsensorluz
    atr "vsld diferença :vsl :vsl1
    atr "vsld abs :vsld
    se :vsld < :faixa [] [
        se :passou = 1 [
            atr "contador :contador +1
            atr "passou 0
        ] [atr "passou 1]
    ]
    atr "vsl1 :vsl
    ;mo :contador
]
```

fim

```

aprenda contadorsensortoque :vsensor
atr "sensor1 :vsensor
rcxsetasensortoque :sensor1
atr "contador 0
atr "vsl1 0
atr "passo 1
ativetemporizador 10 100 [
    atr "vsl rcxlensensor :sensor1
    se :vsl = :vsl1 [] [
        se :passo =1
            [
                atr "contador :contador +1
                atr "passo 0
            ] [atr "passo 1]
        ]
    atr "vsl1 :vsl
    mo :contador
]
fim

```

```

aprenda execgi
atribua "x 0
façaenquanto [
    atribua "i "
    façaenquanto [
        abraparaler "c:\\logo\\corcgi.dat
        mudeleitura "c:\\logo\\corcgi.dat
        atr "i leiacaracteres 2
        ;mostre :i
        atr "vl sãoiguais "ok :i
        ;mostre :vl
        mudeleitura []
        fechearq "c:\\logo\\corcgi.dat
    ][:vl ="falso]
    abraparaler "c:\\logo\\formcgi.dat
    mudeleitura "c:\\logo\\formcgi.dat
    atr "coexec leialista
    mudeleitura []
    fechearq "c:\\logo\\formcgi.dat
    mostre :coexec
    pegue "erro[faça :coexec]
    abraparaescrever "c:\\logo\\corcgi.dat
    mudeescrita "c:\\logo\\corcgi.dat
    esc "falso
    mudeescrita []
    fechearq "c:\\logo\\corcgi.dat

```

```
][:x =0]
fim
```

```
aprenda rcxdesligamotor :lista
atribua "i "
façaenquanto [
    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i
    atr "vl sãoiguais "ok :i
    ;mostre :vl
    mudeleitura []
    fecheaq "c:\\logo\\cor.dat
    ][:vl ="falso]
```

```
abraparaescrever "c:\\logo\\cor.dat
mudeescrita "c:\\logo\\cor.dat
esc "motoroff
esc :lista
mudeescrita []
fecheaq "c:\\logo\\cor.dat
fim
```

```
aprenda rcxlrsensor :num
atribua "i "
façaenquanto [
    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i
    atr "vl sãoiguais "ok :i
    ;mostre :vl
    mudeleitura []
    fecheaq "c:\\logo\\cor.dat
    ][:vl ="falso]
```

```
abraparaescrever "c:\\logo\\cor.dat
mudeescrita "c:\\logo\\cor.dat
esc "lsensor
esc "9
esc :num
mudeescrita []
fecheaq "c:\\logo\\cor.dat
```

```
atribua "i "
façaenquanto [
```

```

    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i
    atr "vl sãoiguais "ok :i
    ;mostre :vl
    mudeleitura []
    fecheaq "c:\\logo\\cor.dat
    ][:vl ="falso]
abraparaler "c:\\logo\\cors.dat
mudeleitura "c:\\logo\\cors.dat
atr "l leiapalavra
;mostre :l
mudeleitura[]
fecheaq "c:\\logo\\cors.dat
envie :l
atr "valorsensorluz :l
fim

```

```

aprenda rxlensorluz :num
atribua "i "
façaenquanto [
    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i
    atr "vl sãoiguais "ok :i
    ;mostre :vl
    mudeleitura []
    fecheaq "c:\\logo\\cor.dat
    ][:vl ="falso]

```

```

abraparaescrever "c:\\logo\\cor.dat
mudeescrita "c:\\logo\\cor.dat
esc "lsensor
esc "9
esc :num
mudeescrita []
fecheaq "c:\\logo\\cor.dat

```

```

atribua "i "
façaenquanto [
    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i

```

```
atr "vl sãoiguais "ok :i
;mostre :vl
mudeleitura []
fechearq "c:\\logo\\cor.dat
][:vl ="falso]
```

```
abraparaler "c:\\logo\\cors.dat
mudeleitura "c:\\logo\\cors.dat
atr "l leiapalavra
;mostre :l
mudeleitura[]
fechearq "c:\\logo\\cors.dat
atr "valorsensornuz :l
fim
```

```
aprenda rxmlersensortoque :num
atr "tsensor rxmlersensor :num
senão :tsensor =1 [envie "verd ] [envie "falso ]
fim
```

```
aprenda rxligamotor :lista
atribua "i "
façaenquanto [
  abraparaler "c:\\logo\\cor.dat
  mudeleitura "c:\\logo\\cor.dat
  atr "i leiacaracteres 2
  ;mostre :i
  atr "vl sãoiguais "ok :i
  ;mostre :vl
  mudeleitura []
  fechearq "c:\\logo\\cor.dat
  ][:vl ="falso]
```

```
abraparaescrever "c:\\logo\\cor.dat
mudeescrita "c:\\logo\\cor.dat
esc "motoron
esc :lista
mudeescrita []
fechearq "c:\\logo\\cor.dat
```

fim

```
aprenda rxligamotorpor :listam :tempol
rxligamotor :listam
espere :tempol
```

```
rcxdesligamotor :listam
fim
```

```
aprenda rcxlimparsensor :lista
atribua "i "
façaenquanto [
    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i
    atr "vl sãoiguais "ok :i
    ;mostre :vl
    mudeleitura []
    fecheaq "c:\\logo\\cor.dat
][:vl ="falso]
```

```
abraparaescrever "c:\\logo\\cor.dat
mudeescrita "c:\\logo\\cor.dat
esc "rcx_clsens
esc :lista
mudeescrita []
fecheaq "c:\\logo\\cor.dat
```

```
fim
```

```
aprenda rcxmotorfrente :lista
atribua "i "
façaenquanto [
    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i
    atr "vl sãoiguais "ok :i
    ;mostre :vl
    mudeleitura []
    fecheaq "c:\\logo\\cor.dat
][:vl ="falso]
```

```
abraparaescrever "c:\\logo\\cor.dat
mudeescrita "c:\\logo\\cor.dat
esc "motorfwd
esc :lista
mudeescrita []
fecheaq "c:\\logo\\cor.dat
fim
```

```

aprenda rcxrevertemotor :lista
atribua "i "
façaenquanto [
    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i
    atr "vl sãoiguais "ok :i
    ;mostre :vl
    mudeleitura []
    fecharq "c:\\logo\\cor.dat
    ][:vl ="falso]
abraparaescrever "c:\\logo\\cor.dat
mudeescrita "c:\\logo\\cor.dat
esc "motor_reverte
esc :lista
mudeescrita []
fecharq "c:\\logo\\cor.dat
fim

```

```

aprenda rcxsetamodosensor :num :mode
atribua "i "
façaenquanto [
    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i
    atr "vl sãoiguais "ok :i
    ;mostre :vl
    mudeleitura []
    fecharq "c:\\logo\\cor.dat
    ][:vl ="falso]
abraparaescrever "c:\\logo\\cor.dat
mudeescrita "c:\\logo\\cor.dat
esc "setmsensor
esc :num
esc :mode
esc "31
mudeescrita []
fecharq "c:\\logo\\cor.dat
fim

```

```

aprenda rcxsetasensorluz :nsensor
rcxsetatiposensor :nsensor 3
rcxsetamodosensor :nsensor 0
fim

```

```
aprenda rcxsetasensortoque :nsensor
rcxsetatiposensor :nsensor 1
rcxsetamodosensor :nsensor 1
fim
```

```
aprenda rcxsetatiposensor :num :tipo
atribua "i "
façaenquanto [
    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i
    atr "vl sãoiguais "ok :i
    ;mostre :vl
    mudeleitura []
    fecharq "c:\\logo\\cor.dat
    ][:vl ="falso]
abraparaescrever "c:\\logo\\cor.dat
mudeescrita "c:\\logo\\cor.dat
esc "setsensor
esc :num
esc :tipo
mudeescrita []
fecharq "c:\\logo\\cor.dat
fim
```

```
aprenda rcxsom :freq :tempo
atribua "i "
façaenquanto [
    abraparaler "c:\\logo\\cor.dat
    mudeleitura "c:\\logo\\cor.dat
    atr "i leiacaracteres 2
    ;mostre :i
    atr "vl sãoiguais "ok :i
    ;mostre :vl
    mudeleitura []
    fecharq "c:\\logo\\cor.dat
    ][:vl ="falso]
abraparaescrever "c:\\logo\\cor.dat
mudeescrita "c:\\logo\\cor.dat
esc "rcx_som
esc :freq
esc :tempo
mudeescrita []
fecharq "c:\\logo\\cor.dat
fim
```

Atribua "coexec []
Atribua "i "ok
Atribua "l 786
Atribua "tsensor 0
Atribua "valorsensornuz 786
Atribua "vl "verd
Atribua "x 0

Funções SuperLogo para controle do RCX

Funções SuperLogo para comandar o RCX

Controle de Motores

rcxligamotor <lista de motores>
rcxdesligamotor <lista de motores>
rcxligapor <lista de motores> <tempo>
rcxmotorfrente <lista de motores>
rcxrevertemotor <lista de motores>

Controle de Sensores

rcxsetatiposensor <lista de sensores > <tipo>
rcxsetamodosensor <lista de sensores > <modo>
rcxlersensor < sensor>

Funções simplificadas para sensores

rcxsetasensornuz <lista de sensores>
rcxsetasensortoque <lista de sensores>
rcxlersensornuz_ <lista de sensores>
rcxlersensortoque <lista de sensores>

Som

rcxsom <frequência> < tempo>

Rcxligamotor <lista de motores>

lg <lista de motores>

Liga motores conectados às saídas do RCX .

Parâmetros:

lista de motores : lista com os motores que devem ser ligados.

Exemplo:

Os motores conectados as saídas A e B são ligados

rcxligamotor [A B]

Rcxdesligamotor <lista de motores>

dlg <lista de motores>

Desliga motores conectados às saídas do RCX .

Parâmetros:

lista de motores : lista com os motores que devem ser desligados.

Exemplo:

Os motores conectados as saídas A e B são desligados

rcxdesligamotor [A B]

Rcxligapor <lista de motores> <tempo>

lgpor <lista de motores>

Liga motores conectados às saídas do RCX por um período determinado.

Parâmetros:

lista de motores : lista com os motores que devem ser desligados.

tempo : tempo em que os motores permanecerão ligados em sendo 1/60 segundos por unidade.

Exemplo:

Os motores conectados as saídas A e B são ligados por 1 segundo

rcxligapor [A B] 60

Rcxmotorfrente <lista de motores>

mf <lista de motores>

Seta os motores conectados às saídas do RCX para que a rotação ocorra no sentido horário.

Parâmetros:

lista de motores : lista com os motores que devem ter o sentido determinado.

Exemplo:

Os motores conectados as saídas A e B tem o sentido de rotação determinado como horário.

rcxmotorfrente [A B]

Rcxrevertemotor <lista de motores>

rm <lista de motores>

Seta os motores conectados às saídas do RCX para que a rotação ocorra no sentido anti-horário.

Parâmetros:

lista de motores : lista com os motores que devem ter o sentido determinado.

Exemplo:

Os motores conectados as saídas A e B tem o sentido de rotação determinado como anti-horário.

rcxrevertemotor [A B]

Rcxsetatiposensor <lista de sensores > <tipo>

Seta e determina o tipo dos sensores conectados às saídas do RCX .

Parâmetros:

lista de sensores : lista com os sensores .

tipo do sensor:

0: Nenhum sensor

1: Chave

2: Temperatura

3: Reflexão (sensor de luz)

4: Ângulo

Exemplo:

Seta o sensor ligado as entradas 1 e 2 como chave.

```
rcxsetatiposensor [1 2] 1
```

Rcxsetamodosensor <lista de sensores > <modo>

Determina o modo como os sinais fornecidos pelos sensores serão representados .

Parâmetros:

lista de sensores : lista com os sensores .

modo do sensor:

0: Analógico

1: Booleano (verdadeiro / falso)

2: Contador de transições (positivas e negativas são contadas)

3: Contador de período

4: Percentagem

5: Celsius

6: Fahrenheit

7: Ângulo

Exemplo:

Seta o sensor ligado as entradas 1 e 2 como analógico .

```
rcxsetamodosensor [1 2 ] 0
```

Rcxlersensor <sensor>

ls <sensor>

retorna o valor do sensor conectado a entrada do RCX indicado em <sensor> .

Parâmetros:

sensor : lista com os sensor .

Exemplo:

Lê o valor do sensor ligado na entrada 1 .

rcxlersensor 1

Rcxsetasensornoz <lista de sensores>

ssl <lista de sensores>

Seta os sensores definidos em lista de sensores como sendo sensores de luz.

Parâmetros:

lista de sensores : lista com os sensores .

Exemplo:

Seta o sensor ligado em 1 e 3.

rcxsetasensornoz [1 3]

Rcxsetasensortoque <lista de sensores>

sst <lista de sensores>

Seta os sensores definidos em lista de sensores como sendo sensores de toque.

Parâmetros:

lista de sensores : lista com os sensores .

Exemplo:

Seta o sensor ligado em 1 e 3.

rcxsetasensortoque [1 3]

Rcxlensorluz <sensor>

lsl <lista de motores>

Retorna o valor do sensor indicado na lista sensor

Parâmetros:

sensor : lista com o sensor .

Exemplo 1 :

mo rcxlensorluz 1

Exemplo 2 :

Com comandos condicionais (se , enquanto) utilizar o valor retornado em valorsensorluz

se :valorsensor > 100 [mo [maior que 100]] [mo [menor que 100]]

Rcxlensortoque <lista de sensores>

lst <lista de motores>

Retorna o valor do sensor indicado na lista sensor.

Parâmetros:

sensor : lista com o sensor .

Exemplo:

rcxlensortoque 1

Rcxsom <frequência> < tempo>

tsom <frequência> < tempo>

Toca um tom no RCX.

Parâmetros:

frequência : frequência do tom emitido pode variar de 1 a 20000 Hz

tempo : determina a duração do tom emitido em passos de 10 ms , pode variar de 1 a 255

Exemplo:

Emite um tom com a frequência de 2000 Hz durante 1 segundo

rcxsom 2000 100

Exemplo de um projeto desenvolvido usando RCX SuperLogo

2.1. Carro de Combate

2.1.1. Introdução

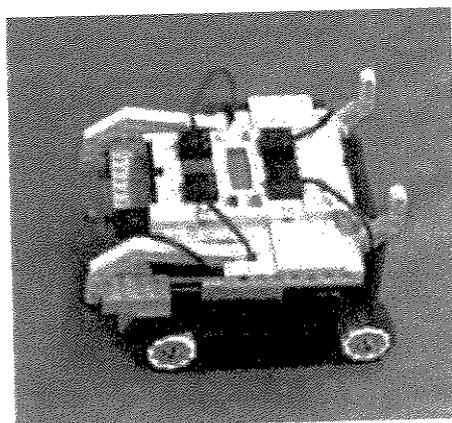
Um dispositivo robótico, que consegue desviar-se de um obstáculo, pode ser uma tarefa interessante quando se pretende construir um robô capaz entrar e sair de um labirinto. Um dos modelos deste robô pode ter o formato de um Carro de Combate.

2.1.2. Objetivo

Construção de um veículo capaz de desviar de obstáculos utilizando sensores de toque.

2.1.3. Descrição

O Carro de Combate é um dispositivo robótico cujo sistema mecânico é constituído por componentes Lego™ (jogos de engrenagens, rosca sem-fim, motor, sensor de toque e tijolos comuns), como mostra a figura abaixo.



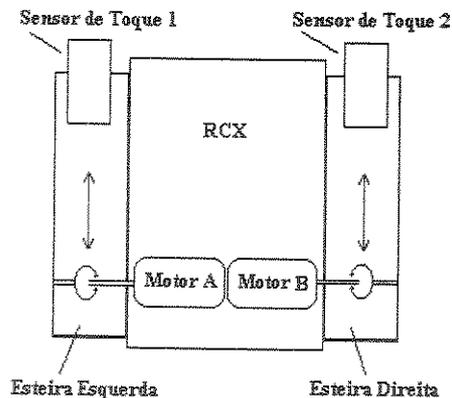
Carro de Combate

Montagem

Para a montagem deste dispositivo, além do RCX propriamente dito, foram utilizados os seguintes componentes elétricos:

- 2 Sensores de Toque
- 2 Motores

Esquema de montagem do Carro de Combate:



As conexões dos componentes elétricos as entradas e saídas do RCX são as seguintes:

ENTRADAS

| | Entrada do RCX | Função |
|-------------------|----------------|---|
| Sensor de Toque 1 | Entrada 1 | Detectar o contato com algum obstáculo no lado esquerdo do Carro de Combate |
| Sensor de Toque 2 | Entrada 3 | Detectar o contato com algum obstáculo no lado direito do Carro de Combate |

SAÍDAS

| | Saída do RCX | Função |
|---------|--------------|-------------------------------|
| Motor A | Saída A | Movimentar a esteira esquerda |
| Motor C | Saída C | Movimentar a esteira direita |

2.1.4. Funcionamento

O Carro de Combate possui dois motores que, individualmente, movimentam cada uma das esteiras relativas ao lado esquerdo e direito do veículo.

O Carro de Combate inicia seu movimento para frente, ligando os dois motores no sentido horário. Ao encontrar um obstáculo (verificado através do contato do sensor de toque com o objeto), como uma parede, o veículo recua, invertendo o sentido dos motores, e, posteriormente, gira em torno do próprio eixo no sentido oposto ao lado em que se encontra o sensor pressionado. Este movimento de rotação ocorre devido ao fato dos motores se movimentarem em sentidos opostos.

Automação e Controle do Dispositivo

Um possível programa para controle deste dispositivo foi escrito em linguagem SuperLogo. O código utilizado encontra-se abaixo.

PROCEDIMENTO EM SUPERLOGO

```
aprenda carro
  rcxsetasensortoque [1 3]
  rcxligamotor "ac
  atr "lcarro 0
  façaenquanto [
    atr "s0 rcxlensensor 1
    atr "s1 rcxlensensor 3
    mostre :s0
    se e :s0=0 :s1=0[
      rcxmotorfrente "ac
    ]
    se e :s0=1 :s1=1[
      rcxrevertemotor "ac
      espere 30
      rcxmotorfrente "a
    ]
    se e :s0=1 :s1=0 [
      rcxrevertemotor "ac
      espere 30
      rcxmotorfrente "a
    ]
    se e :s0=0 :s1=1 [
      rcxrevertemotor "ac
      espere 30
      rcxmotorfrente "c
    ]
  ]
  [:lcarro=0]
fim
```

Anexo 4
Programas do Sistema de
Triagem de produtos

CÓDIGO FONTE DO PROGRAMA PRINCIPAL

```
unit Unit1;

interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  OleCtrls, SPIRITLib_TLB, ComCtrls, StdCtrls, ExtCtrls, DBCtrls, Grids,
  DBGrids, Db, DBTables, Buttons;

type
  tipovetor = array [1..5] of integer; {armazenar valores do sensor de luz}
  TFormPrincipal = class(TForm)
    PageControl1: TPageControl;
    LegoSheet: TTabSheet;
    RobixSheet: TTabSheet;
    LRSheet: TTabSheet;
    Lego: T Spirit;
    PortaComboBox: TComboBox;
    DBSheet: TTabSheet;
    RCXEdit: TEdit;
    RCXLabel: TLabel;
    TorreLabel: TLabel;
    TorreEdit: TEdit;
    CaboTorreLabel: TLabel;
    CaboEdit: TEdit;
    PortaLabel: TLabel;
    FirmPanel: TPanel;
    CoresDataSource: TDataSource;
    CorDBGrid: TDBGrid;
    DBNavigator1: TDBNavigator;
    Form2VisibleRadioGroup: TRadioGroup;
    VisivelRadioButton: TRadioButton;
    NVISIVELRadioButton: TRadioButton;
    DownloadProgressBar: TProgressBar;
    DownloadTimer: TTimer;
    DownloadPercentualLabel: TLabel;
    ReconheceTimer: TTimer;
    CorEdit: TEdit;
    Panell: TPanel;
    TijoloImage: TImage;
    MotorButton: TBitBtn;
    ComButton: TBitBtn;
    DownloadButton: TBitBtn;
    Labell: TLabel;
    LeituraButton: TBitBtn;
```

```

CoresTable: TTable;
procedure FormCreate(Sender: TObject);
procedure ComButtonClick(Sender: TObject);
procedure PortaComboBoxChange(Sender: TObject);
procedure DownloadButtonClick(Sender: TObject);
procedure VisivelRadioButtonClick(Sender: TObject);
procedure NVISIVELRadioButtonClick(Sender: TObject);
procedure LegoDownloadDone(Sender: TObject; ErrorCode,
  DownloadNo: Smallint);
procedure LegodownloadStatus(Sender: TObject; timeInMS,
  sizeInBytes: Integer; taskNo: Smallint);
procedure DownloadTimerTimer(Sender: TObject);
procedure ReconheceTimerTimer(Sender: TObject);
procedure MotorButtonClick(Sender: TObject);
procedure LeituraButtonClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
  {integer}
  am,az,ci,vd,vm,pt, {Contagem do total de pecas}
  DownloadTime, {tempo de download}
  auxdownloadpercentual, {ProgressBar Percentual do Download}
  cont {contador reconhecetimer}
  : integer;
  {Real}
  DownloadPercentual: real;
  {TipoVetor}
  v:TipoVetor;
  {Boolean}
  cond:boolean;
end;

var
  FormPrincipal: TFormPrincipal;

implementation

uses Unit2;

{$R *.DFM}

```

```
{***** LEGOSHEET *****}
```

```
procedure TFormPrincipal.FormCreate(Sender: TObject);  
begin  
  {Ao criar o formulario principal}  
end;
```

```
procedure TFormPrincipal.ComButtonClick(Sender: TObject);  
begin  
  {Início da Configuração da Porta Serial LEGO}  
  If PortaComboBox.Items.Text='COM 1' then  
    Lego.ComPortNo:=1 else  
  If PortaComboBox.Items.Text='COM 2' then  
    Lego.ComPortNo:=2 else  
  If PortaComboBox.Items.Text='COM 3' then  
    Lego.ComPortNo:=3 else  
  If PortaComboBox.Items.Text='COM 4' then  
    Lego.ComPortNo:=4;  
  {Comunicação entre Torre IR e RCX}  
  Lego.InitComm;  
  {Verificação da Comunicação}  
  If Lego.TowerAndCableConnected then begin {Funcionamento da Torre e Cabo}  
    CaboEdit.Text:=' OK ';  
    If Lego.TowerAlive then begin {Funcionamento da Torre}  
      TorreEdit.Text:=' OK ';  
      If Lego.PBAliveOrNot then RcxEdit.Text:=' OK ' {Funcionamento do RCX}  
        else RcxEdit.Text:=' FALHA ';  
    end else begin  
      TorreEdit.Text:=' FALHA ';  
      RcxEdit.Text:='FALHA NA TORRE';  
    end end else begin  
      CaboEdit.Text:=' FALHA ';  
      TorreEdit.Text:='FALHA NA CONEXÃO';  
      RCXEdit.Text:='FALHA NA CONEXÃO';  
    end;  
end;
```

```
Try  
  Lego.SetSensorType(0,3);  
  Lego.SetSensorMode(0,0,0);  
finally  
end;  
cond:=false;  
{Atualiza variaveis - Banco de Dados CorTotalTable }  
  {**** amarelo **** }  
GraficoForm.TotalCorTable.FindField('AMARELO');  
am:=strtoint(GraficoForm.TotalCorTable.Fields[1].AsString);  
  {**** azul **** }
```

```

GraficoForm.TotalCorTable.FindField('AZUL');
az:=strtoint(GraficoForm.TotalCorTable.Fields[1].AsString);
    {**** cinza **** }
GraficoForm.TotalCorTable.FindField('CINZA');
ci:=strtoint(GraficoForm.TotalCorTable.Fields[1].AsString);
    {**** verde **** }
GraficoForm.TotalCorTable.FindField('VERDE');
vd:=strtoint(GraficoForm.TotalCorTable.Fields[1].AsString);
    {**** vermelho **** }
GraficoForm.TotalCorTable.FindField('VERMELHO');
vm:=strtoint(GraficoForm.TotalCorTable.Fields[1].AsString);
    {**** preto **** }
GraficoForm.TotalCorTable.FindField('PRETO');
pt:=strtoint(GraficoForm.TotalCorTable.Fields[1].AsString);
end;

```

```

procedure TFormPrincipal.PortaComboBoxChange(Sender: TObject);
begin
    {Mudança da Porta Serial}
    If PortaComboBox.Items.Text='COM 1' then
        Lego.ComPortNo:=1 else
    If PortaComboBox.Items.Text='COM 2' then
        Lego.ComPortNo:=2 else
    If PortaComboBox.Items.Text='COM 3' then
        Lego.ComPortNo:=3 else
    If PortaComboBox.Items.Text='COM 4' then
        Lego.ComPortNo:=4;
end;

```

{Download do Firmware}

```

procedure TFormPrincipal.DownloadButtonClick(Sender: TObject);
var versaofirm: widestring;
begin
    {Início da Configuração da Porta Serial LEGO}
    If PortaComboBox.Items.Text='COM 1' then
        Lego.ComPortNo:=1 else
    If PortaComboBox.Items.Text='COM 2' then
        Lego.ComPortNo:=2 else
    If PortaComboBox.Items.Text='COM 3' then
        Lego.ComPortNo:=3 else
    If PortaComboBox.Items.Text='COM 4' then
        Lego.ComPortNo:=4;
    {Comunicação entre Torre IR e RCX}
    Lego.InitComm;

```

```

{Verificação da Comunicação}
If Lego.TowerAndCableConnected then begin {Funcionamento da Torre e Cabo}
  CaboEdit.Text:=' OK ';
  If Lego.TowerAlive then begin {Funcionamento da Torre}
    TorreEdit.Text:=' OK ';
    If Lego.PBAliveOrNot then RcxEdit.Text:=' OK ' {Funcionamento do RCX}
    else RcxEdit.Text:=' FALHA ';
  end else begin
    TorreEdit.Text:=' FALHA ';
    RcxEdit.Text:='FALHA NA TORRE';
  end end else begin
    CaboEdit.Text:=' FALHA ';
    TorreEdit.Text:='FALHA NA CONEXÃO';
    RCXEdit.Text:='FALHA NA CONEXÃO';
  end;
  Try
    Lego.SetSensorType(0,3);
    Lego.SetSensorMode(0,0,0);
  finally
    end;
  versaofirm:='firm0309.lgo';
  If PortaComboBox.Items.Text='COM 1' then
    Lego.ComPortNo:=1 else
  If PortaComboBox.Items.Text='COM 2' then
    Lego.ComPortNo:=2 else
  If PortaComboBox.Items.Text='COM 3' then
    Lego.ComPortNo:=3 else
  If PortaComboBox.Items.Text='COM 4' then
    Lego.ComPortNo:=4;
  Lego.InitComm;
  If Lego.PBAliveOrNot then begin
    DownloadButton.Enabled:=false;
    Lego.DownloadFirmware(versaofirm); {Inicia Download do Firmware}
    DownloadTimer.Enabled:=true; {Inicia DownloadTimer}
    DownloadPercentual:=0;
    auxdownloadpercentual:=0;
  end;
end;

      {Verificação do Download do Firmware}
procedure TFormPrincipal.LegoDownloadDone(Sender: TObject; ErrorCode,
  DownloadNo: Smallint);
begin
  If (ErrorCode < 0) then begin
    DownloadTimer.Enabled:=false;
    ShowMessage('Erro no Download');
    DownloadButton.Enabled:=true;
    DownloadProgressBar.StepBy(-Trunc((DownloadTime-9960)/10));
  end;
end;

```

```

DownloadPercentual:=0;
DownloadPercentualLabel.Caption:=IntToStr(trunc(DownloadPercentual))+' %';
end else begin
DownloadTimer.Enabled:=false;
ShowMessage('Download Completo');
Lego.UnlockFirmware('Do you byte, when I knock?');
Try
Lego.InitComm;
Lego.SetSensorType(0,3);
Lego.SetSensorMode(0,0,0);
finally
end;
DownloadProgressBar.StepBy(-Trunc(DownloadTime-9960));
DownloadPercentual:=0;
DownloadPercentualLabel.Caption:=IntToStr(trunc(DownloadPercentual))+' %';
DownloadButton.Enabled:=true;
end;
end;
{Estimar o Tempo de Download}
procedure TFormPrincipal.LegodownloadStatus(Sender: TObject; timeInMS,
sizeInBytes: Integer; taskNo: Smallint);
begin
DownloadTime:=TimeInMs;
end;
{Atualizar ProgressBar do Download do Firmware}
procedure TFormPrincipal.DownloadTimerTimer(Sender: TObject);
begin
DownloadTimer.Interval:=1;
DownloadProgressBar.StepBy(10);
DownloadProgressBar.Max:=(DownloadTime-9960);
AuxDownloadPercentual:=AuxDownloadPercentual+10;
DownloadPercentual:=(100*AuxDownloadPercentual)/(DownloadTime-9960);
DownloadPercentualLabel.Caption:=IntToStr(trunc(DownloadPercentual))+' %';
end;
{***** BANCO DE DADOS SHEET *****}
procedure TFormPrincipal.VisivelRadioButtonClick(Sender: TObject);
begin
graficoform.visible:=true;
formprincipal.Align:=altop;
end;

procedure TFormPrincipal.NVISIVELRadioButtonClick(Sender: TObject);
begin
graficoform.Visible:=false;
formprincipal.Align:=alnone;
formprincipal.ClientHeight:=348;
formprincipal.ClientWidth:=536;

```

```
formprincipal.Left:=200;
formprincipal.Top:=50;
end;
```

```
{***** CONTROLE DO PROCESSO SHEET *****}
```

```
{Iniciar Reconhecimento de Cor}
procedure TFormPrincipal.LeituraButtonClick(Sender: TObject);
begin
  cont:=0;
  ReconheceTimer.Enabled:=true;
  LeituraButton.Glyph.LoadFromFile('ledon.bmp');
end;
```

```
{Calculo da Média dos valores lidos pelo sensor}
function calculamedia(var vetor:tipovetor):integer;
var a,total:integer;
begin
  total:=0;
  For a:=2 to 4 do begin
    If a=2 then total:=vetor[a]
    else total:=total+vetor[a];
  end;
  calculamedia:=trunc(total/3)
end;
```

```
{Ordenar vetor com valores dos sensores}
procedure ordena(var vetor:tipovetor);
var vaux,i,j:integer;
begin
  For j:=1 to 5 do
    For i:=2 to 5 do
      If vetor[i-1]>vetor[i] then begin
        vaux:=vetor[i-1];
        vetor[i-1]:=vetor[i];
        vetor[i]:=vaux;
      end;
    end;
end;
```

```
{Timer Reconhece Cor}
procedure TFormPrincipal.ReconheceTimerTimer(Sender: TObject);
begin
  If cont<=4 then begin
    cont:=cont+1;
    v[cont]:=Lego.Poll(12,0);
  end;
end;
```

```

end else begin
  ReconheceTimer.Enabled:=false;
  ordena(v);      {ordena o vetor}
  Case calculamedia(v) of      {calcula a média do vetor para
                                o reconhecimento de cor}
    600..672: begin
      CorEdit.Text:= 'AMARELO';
      TijoloImage.Picture.LoadFromFile('Amarelo.bmp');
      {Atualiza Banco de Dados Cores.Db - Amarelo}
      Corestable.insert;
      Corestable.Fields[0].AsString:='Amarelo';
      Corestable.Fields[1].AsDateTime:=date;
      Corestable.Fields[2].AsDateTime:=time;
      Corestable.Post;
      {Atualiza Banco de Dados TotalCores.Db - Amarelo}
      GraficoForm.TotalCorTable.First;
      GraficoForm.TotalCorTable.Edit;
      am:=am+1;
      GraficoForm.TotalCorTable.Fields[1].AsString:=IntToStr(am);
      GraficoForm.TotalCorTable.Post;
    end;
    673..690: begin
      CorEdit.Text:= 'VERMELHO';
      TijoloImage.Picture.LoadFromFile('Vermelho.bmp');
      {Atualiza Banco de Dados Cores.Db - Vermelho}
      Corestable.insert;
      Corestable.Fields[0].AsString:='Vermelho';
      Corestable.Fields[1].AsDateTime:=date;
      Corestable.Fields[2].AsDateTime:=time;
      Corestable.Post;
      {Atualiza Banco de Dados TotalCores.Db - VERMELHO}
      GraficoForm.TotalCorTable.First;
      GraficoForm.TotalCorTable.MoveBy(5);
      GraficoForm.TotalCorTable.Edit;
      vm:=vm+1;
      GraficoForm.TotalCorTable.Fields[1].AsString:=IntToStr(vm);
      GraficoForm.TotalCorTable.Post;
    end;
    691..719: begin
      CorEdit.Text:= 'CINZA';
      TijoloImage.Picture.LoadFromFile('Cinza.bmp');
      {Atualiza Banco de Dados Cores.Db- Cinza}
      Corestable.insert;
      Corestable.Fields[0].AsString:='Cinza';
      Corestable.Fields[1].AsDateTime:=date;
      Corestable.Fields[2].AsDateTime:=time;
      Corestable.Post;
    end;
  end;
end;

```

```

{Atualiza Banco de Dados TotalCores.Db - VERMELHO}
GraficoForm.TotalCorTable.First;
GraficoForm.TotalCorTable.MoveBy(2);
GraficoForm.TotalCorTable.Edit;
ci:=ci+1;
GraficoForm.TotalCorTable.Fields[1].AsString:=IntToStr(ci);
GraficoForm.TotalCorTable.Post;
end;

```

720..740: begin

```

CorEdit.Text:= 'AZUL';
TijoloImage.Picture.LoadFromFile('Azul.bmp');
{Atualiza Banco de Dados - Azul}
Corestable.insert;
Corestable.Fields[0].AsString:='Azul';
Corestable.Fields[1].AsDateTime:=date;
Corestable.Fields[2].AsDateTime:=time;
Corestable.Post;
{Atualiza Banco de Dados TotalCores.Db - Azul}
GraficoForm.TotalCorTable.First;
GraficoForm.TotalCorTable.MoveBy(1);
GraficoForm.TotalCorTable.Edit;
az:=az+1;
GraficoForm.TotalCorTable.Fields[1].AsString:=IntToStr(az);
GraficoForm.TotalCorTable.Post;
end;

```

741..759: begin

```

CorEdit.Text:= 'VERDE';
TijoloImage.Picture.LoadFromFile('verde.bmp');
{Atualiza Banco de Dados - Verde}
Corestable.insert;
Corestable.Fields[0].AsString:='Verde';
Corestable.Fields[1].AsDateTime:=date;
Corestable.Fields[2].AsDateTime:=time;
Corestable.Post;
{Atualiza Banco de Dados TotalCores.Db - Verde}
GraficoForm.TotalCorTable.First;
GraficoForm.TotalCorTable.MoveBy(4);
GraficoForm.TotalCorTable.Edit;
vd:=vd+1;
GraficoForm.TotalCorTable.Fields[1].AsString:=IntToStr(vd);
GraficoForm.TotalCorTable.Post;
end;

```

760..900: begin

```

CorEdit.Text:= 'PRETO';
TijoloImage.Picture.LoadFromFile('Preto.bmp');
{Atualiza Banco de Dados - Amarelo}

```

```

    Corestable.insert;
    Corestable.Fields[0].AsString:='Preto';
    Corestable.Fields[1].AsDateTime:=date;
    Corestable.Fields[2].AsDateTime:=time;
    Corestable.Post;
    {Atualiza Banco de Dados TotalCores.Db - Preto}
    GraficoForm.TotalCorTable.First;
    GraficoForm.TotalCorTable.MoveBy(3);
    GraficoForm.TotalCorTable.Edit;
    pt:=pt+1;
    GraficoForm.TotalCorTable.Fields[1].AsString:=IntToStr(pt);
    GraficoForm.TotalCorTable.Post;
end;
end;

    LeituraButton.Glyph.LoadFromFile('ledoff.bmp');
end;
end;

{Motor Esteira}
procedure TFormPrincipal.MotorButtonClick(Sender: TObject);
begin
    Lego.SetRwd('motor0');
    If cond then begin
        Lego.Off('motor0');
        cond:=false;
        MotorButton.Glyph.LoadFromFile('ledoff.bmp');
    end else begin
        Lego.On_('motor0');
        cond:=true;
        MotorButton.Glyph.LoadFromFile('ledon.bmp');
    end;
end;
end;
end.

```

CÓDIGO FONTE DO FORMULÁRIO REFERENTE AO GRÁFICO

```

unit Unit2;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,

```

ExtCtrls, DBCtrls, Grids, DBGrids, Db, DBTables, TeEngine, Series,
TeeProcs, Chart, DBChart;

type

```
TGraficoForm = class(TForm)
  TotalCorDataSource: TDataSource;
  TotalCorTable: TTable;
  CorTotalDBGrid: TDBGrid;
  DBNavigator1: TDBNavigator;
  DBChart1: TDBChart;
  Series1: TPieSeries;
  procedure FormClose(Sender: TObject; var Action: TCloseAction);
private
  { Private declarations }
public
  { Public declarations }
end;
```

var

```
GraficoForm: TGraficoForm;
```

implementation

uses Unit1;

```
{SR *.DFM}
```

```
procedure TGraficoForm.FormClose(Sender: TObject;
  var Action: TCloseAction);
begin
  formprincipal.nvisivelradiobutton.Checked:=true;
end;

end.
```

PROGRAMAÇÃO DAS TRAJETÓRIAS DO ROBIX™

TRAJETÓRIA 01

a) MACROS ENVOLVIDAS

| Nº | Nome da Macro | Descrição |
|----|---------------|------------------------------------|
| 1 | Home | Posição Inicial |
| 2 | Abre_Garra | Abertura da garra |
| 3 | Fecha_Garra | Fechamento da garra |
| 4 | Traj01 | Define a posição da plataforma A |
| 5 | Pega_Cubo | Pegar o cubo (movimento relativo) |
| 6 | Solta_Cubo | Soltar o cubo (movimento relativo) |

b) POSIÇÕES ENCONTRADAS

| Posição | Absol. / Relat. | 1 | 2 | 3 | 4 | 5 | 6 |
|----------------------|-----------------|-----|-----|-------|------|-----|-------|
| Inicial | Absoluta | 200 | 0 | -1000 | -600 | --- | --- |
| Abertura de Garra | Absoluta | --- | --- | --- | --- | --- | -1400 |
| Fechamento de Garra | Absoluta | --- | --- | --- | --- | --- | -1050 |
| Traj01 | Relativa | 400 | --- | --- | --- | --- | --- |
| Pega_Cubo (desce) | Relativa | --- | --- | --- | -425 | --- | --- |
| Pega_Cubo (retorno) | Relativa | --- | --- | --- | 425 | --- | --- |
| Solta_Cubo (desce) | Relativa | --- | --- | --- | -425 | --- | --- |
| Solta_Cubo (retorno) | Relativa | --- | --- | --- | 425 | --- | --- |

PROGRAMAÇÃO

```
# RESET DE MEMORIA  
FORGET ALL
```

```
#POSICAO INICIAL  
MACRO HOME  
  MOVE 1 TO 200, 2 TO 0, 3 TO -1000, 4 TO -600  
END
```

```
#ABERTURA DE GARRA  
MACRO ABRE_GARRA  
  MOVE 6 TO -1400  
END
```

```
#FECHAMENTO DE GARRA  
MACRO FECHA_GARRA  
  MOVE 6 TO -1050  
END
```

```
#DEFINE TRAJETORIA 1  
MACRO TRAJ01
```

```
MOVE 1 BY 400
END
```

```
#PEGAR CUBO COM MOVIMENTO RELATIVO
MACRO PEGA_CUBO
ABRE_GARRA
PAUSE 3
MOVE 4 BY -425 #DESCIDA
PAUSE 3
FECHA_GARRA
PAUSE 3
MOVE 4 BY 425 #SUBIDA
END
```

```
#SOLTAR CUBO COM MOVIMENTO RELATIVO
MACRO SOLTA_CUBO
MOVE 4 BY -425 #DESCIDA
PAUSE 3
ABRE_GARRA
PAUSE 3
MOVE 4 BY 425
PAUSE 3
FECHA_GARRA
END
```

```
#TRAJETÓRIA 1
HOME;ABRE_GARRA;PEGA_CUBO;TRAJ01;SOLTA_CUBO;HOME
```

TRAJETÓRIA 02, 03, 04, 05, 06 e 07

a) MACROS ENVOLVIDAS

As macros para as trajetórias 02, 03, 04, 05, 06 e 07 são as mesmas macros descritas anteriormente (Trajetória 01), com exceção da macro Traj01, que se refere ao movimento do manipulador até a plataforma A. Desta forma, a macro 4 de cada programa das seis trajetórias restantes serão, respectivamente, Traj02, Traj03, Traj04, Traj05, Traj06 e Traj07.

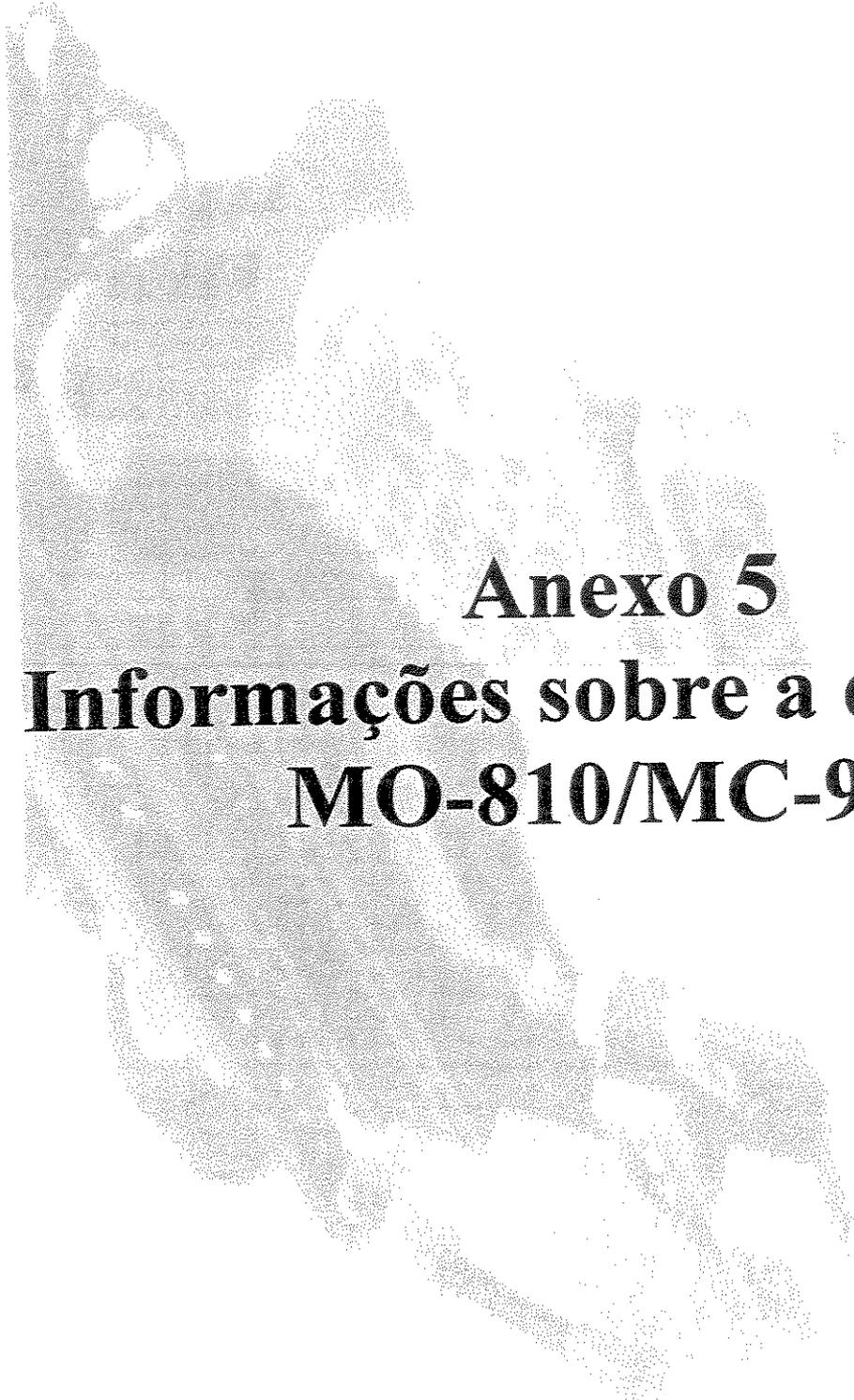
b) POSIÇÕES ENCONTRADAS

As posições também serão as mesmas, com exceção da macro Traj01. Abaixo estão descritas as posições das macros Traj02, Traj03, Traj04, Traj05, Traj06 e Traj07.

| Posição | Absol. / Relat. | 1 | 2 | 3 | 4 | 5 | 6 |
|---------|-----------------|-------|-----|-----|-----|-----|-----|
| Traj02 | Relativa | 800 | --- | --- | --- | --- | --- |
| Traj03 | Relativa | 1200 | --- | --- | --- | --- | --- |
| Traj04 | Relativa | -400 | --- | --- | --- | --- | --- |
| Traj05 | Relativa | -800 | --- | --- | --- | --- | --- |
| Traj06 | Relativa | -1200 | --- | --- | --- | --- | --- |
| Traj07 | Relativa | -200 | --- | --- | --- | --- | --- |

PROGRAMAÇÃO

Análogo ao código da trajetória 1, com as devidas modificações referente às macros restantes.



Anexo 5
Informações sobre a disciplina
MO-810/MC-959

Universidade Estadual de Campinas

Instituto de Computação

MO-810/MC-959

Cursos: Pós-graduação em Ciência da Computação, Bacharelado em Ciência da Computação e Engenharia de Computação

Disciplina: Robótica: Sistemas Sensorial e Motor

Informação Administrativa para segundo semestre 2001

Local e data-hora: Qua-Sex, 16:00 às 18:00, Sala CC-85

Instrutor:

Luiz M. G. Goncalves

- Email: lmarcos@ic.unicamp.br
- Sala:: IC-012, Fone: +8 5875

Professor colaborador (PED-II): João Vilhete

Recursos Online

A página Web da disciplina está em:

~lmarcos/courses/mc810/index.html

Esta página é a melhor fonte para voce obter informações a respeito da disciplina, incluindo trabalhos práticos, notas de aula e detalhes administrativos. Se houver material disponível, recomendo imprimi-los e usá-lo em sala, melhorando-o. Muito material (ou quase tudo) estará em Inglês. É suposto que voce já tenham tido inglês técnico durante o curso no IC.

Para questões verdadeiramente urgentes, use email ou telefone:-).

Objetivos

- O objetivo deste curso é abordar os princípios fundamentais da robótica incluindo percepção. A robótica tem tido um papel cada vez mais relevante para o homem, na

medida em que ela passa a colaborar em diversas tarefas essenciais, em particular no setor produtivo. Trata-se de uma área de pesquisa que é inerentemente multidisciplinar, e onde os conhecimentos de Computação além de fundamentais, são aplicados extensivamente.

- Não menos importante, o segundo objetivo do curso é prover ao aluno conceitos básicos de projeto completo de um "produto" que está em contato com o mundo real. Isso implica em uma melhor compreensão da distância que existe entre abstração e a implementação prática. Nesse sentido, durante o curso, os alunos estarão projetando e implementando robôs móveis para realizarem tarefas em um ambiente semi-estruturado, porém dinâmico. Esses robôs serão desenvolvidos em grupos de 6 alunos. O material do curso será coberto por meio de aulas expositivas e de laboratório.

Pré-requisitos

- Conhecimento de algebra linear, cálculo vetorial e física básica (especialmente mecânica clássica). A linguagem de programação a ser utilizada no curso será C (ou C++) e também uma linguagem visual provida pelo fabricante dos produtos LEGO.

Livros Texto

1. Fred G. Martin. Robotic Explorations: A Hands-on Introduction to Engineering. Prentice Hall CO. 2001.
2. Craig, J.J., Introduction to Robotics - Mechanics and Control - 2nd. Ed., Addison-Wesley, Reading, MA, 1989, ISBN 0-201-09528-9.
3. Jones, J. and Flynn, A.M., Mobile Robots - Inspiration to Implementation -A. K. Peters, Wellesley, MA, 1993, ISBN 1-56881-011-3.

Avaliação do Aprendizado

- A avaliação do aprendizado se dará através de listas de exercícios, trabalhos práticos, uma prova e um projeto final. A tabela a seguir é uma primeira aproximação da distribuição de valores para cada etapa como cada aluno será avaliado de valores se encontra na tabela a seguir.
 1. Trabalhos práticos (robô): total de 50%
 1. Robôs: 20%
 2. Página: 10%
 3. Caderno: 10%
 4. Software: 10%
 2. Parte teórica: total de 50%
 1. Prova: 25%
 2. Seminário: 10%
 3. Listas de exercícios: 15%
- Assume-se que o robô deve participar funcionalmente da competição final.

Listas de Exercício

- As listas de exercícios deverão ser entregues pelo aluno na data marcada, no início da aula. Listas fora do prazo não serão consideradas. Embora a discussão dos exercícios seja encorajada, a resolução e redação é estritamente individual.

Trabalhos Práticos

- A finalidade dos trabalhos práticos é se construir o conhecimento referente a cada um dos aspectos de um projeto maior, que será o desenvolvimento de um robô móvel capaz de executar com sucesso tarefas previamente especificadas. Cada detalhe, idéia, desde a concepção até a implementação deverá constar no caderno de laboratório (veja a seguir). É indispensável que o projeto seja acompanhado do planejamento dos testes. Testes deverão ser realizados de forma exaustiva para se garantir que o projeto está correto. Dessa maneira, ao se implementar as soluções planejadas, os testes correspondentes (previamente planejados) serão aplicados para verificação. Implemente e teste pequenas porções do projeto maior. Lembre-se que embora tedioso, esse processo consistente e sistemático é a chave de uma implementação bem sucedida. Esses ``pequenos detalhes" é que fazem a diferença!
- Os trabalhos práticos foram projetados de tal maneira a preparar para a competição final, que consistirá da integração dos diversos subsistemas de hardware (estrutura mecânica, sensores, atuadores) e software (controle do robô).
- Os trabalhos práticos serão realizados em grupo e deverão ser apresentados conforme instruções constantes na página do curso. Todos os membros do grupo deverão estar presentes na apresentação de cada trabalho prático. A participação individual de cada membro será avaliada.

Caderno de Laboratório

- Esse será o seu companheiro de todas as horas. Cada aluno deverá ter o seu próprio caderno. Recomenda-se fortemente que o caderno seja pequeno, tipo brochura (não pode ser espiral), de capa dura e com cerca de 80 páginas. Nele serão anotadas idéias de mecanismos, estratégias, algoritmos, etc. Deve ser bem cuidado e de fácil leitura (se não entendermos o que está escrito, não tem como avaliar). Figuras, fotos, gráficos e tabelas obtidos de calibração de sensores, podem ser colados diretamente nas páginas do caderno.

Seminário

- Os seminários serão preparados e apresentados por cada grupo no final do curso, discorrendo sobre cada projeto, dificuldades, resultados esperados, resultados alcançados, etc. A avaliação será composta de dois itens: avaliação realizada pelos alunos do curso e do conjunto de transparências feitas em PowerPoint, cujo arquivo será disponibilizado na

página do grupo (ver abaixo). Neste seminário final, cada grupo fará uma apresentação do seu projeto final (incluindo o robô), que deverá conter:

1. Descrição do hardware, detalhando acionamento, transmissão, etc., justificando as opções de projeto;
2. Descrição dos sensores - justificando as opções de projeto;
3. Descrição do software de controle, justificando as opções de projeto;
4. Dificuldades encontradas.

Competição

- A competição será divulgada em data estabelecida posteriormente (a princípio em dezembro). O projeto consistirá na construção de um robô móvel autônomo. A estrutura mecânica dos robôs será implementada com componetes da LEGO, bem como a parte eletrônica. Esses robôs serão desenvolvidos em grupos de 6 alunos.

Página WWW

- A documentação, que faz parte integrante do projeto e também será avaliada, estará contida na Home Page que cada grupo irá manter ao longo do curso. As atividades de cada semana deverão estar disponíveis a cada segunda-feira pela manhã. Um dos links da página deverá apontar para um arquivo compactado (.zip) contendo o "back-up" completo e atualizado da própria página. Esta página deverá ser disponibilizada em algum lugar publico.
- Cada grupo deverá ter um nome, pelo qual o seu robô também será conhecido. Toda a evolução do seu projeto deverá ser documentada com fotos digitais a serem incorporadas na página do grupo. Essas imagens deverão mostrar detalhes construtivos em "close-up". É interessante que a página contenha uma foto dos membros do grupo. Haverá um link na página do curso para página de cada grupo. Assim, preciso apenas do endereço desta página para endereçar cada grupo.

Política com relação à cola

A cola não será tolerada, tanto em programas quanto na parte teórica (trabalhos escritos e provas). Se alguém for pego colando, será enquadrado no regulamento da UNICAMP (lembro que este regulamento prevê no mínimo Zero na disciplina, ou seja, reprovação, e, dependendo da gravidade, o aluno poderá ser convidado a deixar a UNICAMP).

O que é considerado cola?

- Copiar o código de alguém para usar em seu programa.
- Copiar o trabalho de alguém nas tarefas escritas.
- Olhar/copiar da prova de outro.
- Deixar outro aluno olhar seu código ou deixar o código livre num local público.

O que não é considerado cola?

- Discutir algoritmos ou idéias com qualquer um para implementar um programa ou para os trabalhos de casa.
- Trocar código com seus colegas de grupo (em caso de trabalhos cooperativos).
- Perguntar ao instrutor sobre qualquer coisa.

Se voce encontrar algo que não esteja claro, por favor, me consulte para esclarecimentos.

Lista de Exercícios #1

Obs: Esta lista será uma das base para a prova teorica. Exercicios obrigatorios: 7, 8, 9, 11, 12, 14, 15, 17, 19, 20, 21, 24, 28, Escolha mais 5 exercicios da lista (18 no total). Entregar até dia 02 de outubro de 2001 (turma da UFMS, entregar ate 29 de setembro 2001). Por favor, escrevam à mão e procurem fazer gráficos à mão. Não precisa copiar o enunciado para as folhas de respostas. Não se esqueçam de identificar a folha resposta (ra e nome). Tempo estimado para responder todas as perguntas: 6 horas e 50 minutos.

1. Descreva com suas próprias palavras as diderenças básicas entre Robótica e Automação.
2. Quais são as três leis da Robótica?
3. O que são os "manipuladores"?
4. Quais são os tipos básicos de manipuladores que interessam ao nosso curso?
5. O que voce entende por "referencial"?
6. O que voce entende por mapeamento e transformação?
7. Dado o ponto $P1=(2,1,1)$, calcule o ponto $P2$, rotacionado de 30 graus em torno de X, 45 graus em torno de Y e 60 graus em torno de Z, tudo em relação ao mesmo referencial (calcule as novas coordenadas do ponto $P2$ no espaço). Obs: descreva cada matriz de rotação separadamente e aplique-as também separadamente e depois faça o mesmo, juntando as 3 rotações numa só matriz de rotação.
8. Aplique uma translação de $(+2, -1, +1.5)$ no resultado da questão anterior.
9. Repita os exercícios anteriores, combinando as matrizes e vetores usados em uma transformação homogênea única.
10. Defina cinemática direta e cinemática inversa.
11. Suponha que um manipulador possua dois links ($L1$ e $L2$) de comprimentos 0.5 e 0.3 metros respectivamente. Faça um desenho em escala do espaço de manipulação do mesmo (reachable space). Quantas soluções existem para o problema de cinemática inversa? Dica: explique separadamente cada situação e o seu número de soluções.

12. Faça o mesmo para um manipulador com 3 links (0.5, 0.3, e 0.2 metros respectivamente para L1, L2 e L3). Quantas soluções existem para o problema inverso neste caso?
13. Dado o manipulador mostrado na figura abaixo, dê o que se pede: Graus de liberdade: _____ Número de juntas: _____ Número de links: _____
14. Na figura, sendo os ângulos 1 e 2 iguais a 60 e -45 graus respectivamente e $(L1, L2) = (0.7, 0.5)$ metros respectivamente, descreva o estado atual de uma matriz para cada junta (2 ao todo) que permita manipulação através de cinemática direta. Considere juntar rotação e translação em uma única matriz homogênea em cada link. Considere a origem do sistema de coordenadas de mundo no frame 0 e apenas rotações em torno de Z. Assim, cinemática direta pode ser obtida pela multiplicação das matrizes obtidas.
15. Na figura acima, considere os mesmos comprimentos de links (0.7,0.5). Sendo as coordenadas do atuador $(x, y) = (-0.63, 0.69)$ em metros, ache a solução para o problema de cinemática inversa.
16. Determine se os seguintes pontos são alcançáveis pelo atuador do manipulador acima: $(x,y)=(1.2, 0.9)$, $(x,y) = (0.17, 0.30)$, $(x,y) =(-0.7, 0.9)$
17. Usando o Jacobiano, se a velocidade angular for dada por (0.2, 0.3) em radianos por segundo, para os ângulos 1 e 2 respectivamente, e considerando os mesmos ângulos atuais (60 e -45 graus - transforme para radianos) e mesmos comprimentos de links (0.7,0.5), qual a velocidade linear do atuador do manipulador (end-effector, no sistema cartesiano (x,y)) ?
18. Sendo a velocidade linear (em (x,y)) do end-efector igual a (0.9, 0.6) m/s, calcule o vetor velocidade angular (em radianos por segundo) para os ângulos 1 e 2.
19. Considere na figura acima, com o manipulador se movendo, que os ângulos 1 e 2, referentes às juntas (motores) 1 e 2, são iguais a 60 e -45 graus, respectivamente, o comprimento dos links $(L1, L2) = (0.7, 0.5)$ metros, a massa de cada um é igual a 1.2 e 0.8 kg, e a velocidade atual é (1.3,1.5) radianos por segundo respectivamente. Defina o estado corrente da matriz de inércia M (que relaciona torque e aceleração), do vetor V de torques dependentes da velocidade e do vetor G (que representa todas as forças gravitacionais). Este primeiro passo é denominado de "determinar a dinâmica do braço". Obs: $g = 9.8$ m/sec/seg. Use as equações dadas em aula para isso (ver final do CHAPTER4.pdf).
20. Com os dados do exercício anterior, qual o vetor de torque necessário para obter uma aceleração de (2.3,2.4) rad/seg/seg nas juntas 1 e 2 respectivamente?
21. Dado um vetor de deslocamento angular diferencial (d_teta1, d_teta2) a ser realizado num pequeno intervalo de tempo d_t , um controlador proporcional derivativo para o manipulador acima deve primeiramente calcular (em cada passo) o vetor de torque necessário pela equação: $(torque1, torque2) = Kp*(d_teta1, d_teta2) - Kd*(teta_dot1, teta_dot2)$ onde Kp e Kd são ganhos (ver parte de controle), (d_teta1, d_teta2) o deslocamento angular diferencial, e $(teta_dot1, teta_dot2)$ a velocidade angular corrente. Suponha que no estado atual o manipulador acima possui os mesmos dados de posição (60,-45 graus) e de velocidade (1.3,1.5 rad/seg) que o problema acima. Um vetor de deslocamento diferencial angular é dado por um planejador de movimento como sendo (+0.05,-0.08) radianos (transforme os graus para radianos, ou vice-versa). Na prática, um deslocamento maior foi "quebrado" em vários sub-deslocamentos como este, de acordo com um intervalo de tempo d_t dado pela "clock-rate" do sistema de tempo real. Supondo $Kp=10000$ e $Kd=200$, calcule para próximo instante de tempo o vetor de aceleração resultante (ver final de CHAPTER4.pdf e use a equação dada aqui para

calcular vetor torque), a velocidade resultante e, finalmente, a posição angular resultante, sendo $d_t = 30$ milissegundos. Este exercício usa os três passos do integrador de Euler (aceleração, velocidade e posição). O resultado prático deste sistema de controle é que o manipulador descreve um movimento onde no início ele acelera e no final ele desacelera. Lembre-se que após cada submovimento, as matrizes homogêneas de cinemática em cada junta podem ser atualizadas usando a posição angular corrente. Assim, um simulador pode usar isso para visualizar o manipulador. (ver código do Roger-the-crab na página - arm.c e control.c).

22. Explique o que é um sistema de controle em malha aberta e malha fechada. De exemplos de cada um com os respectivos diagramas e cite aplicações. Quais interessam ao curso?
23. Pesquise e descreva sucintamente o funcionamento de um sistema de controle atual que seja muito utilizado (pode ser qualquer um como por exemplo um controlador de temperatura em geladeira).
24. Explique sucintamente cada tipo de controle: liga-desliga, proporcional, derivativo, integral, proporcional-derivativo.
25. Descreva a diferença entre controle reativo e controle deliberativo. O que controle híbrido?
26. O que voce entende por um filtro (ver sensores)?
27. Explique com suas palavras, sucintamente, cada item abaixo(sensor, princípio de funcionamento, aplicações)
 1. interruptor de lâminas
 2. sensor ótico por reflexão
 3. sensor ótico por interrupção
 4. sensor ótico relativo
 5. sensor ótico absoluto
 6. encoder
 7. potenciômetro
 8. bussola
 9. GPS
 10. tacogerador
 11. giroscópio
 12. acelerômetro
 13. sonar
 14. LDR
 15. foto-diodo
 16. fototransistor
28. O que voce está achando do curso a esta altura? Por favor, seu retorno a esta questão é individual e não será reportado.

Trabalho prático #1

- Fazer um programa usando o Robolab que comande um UV (Umnanned Vehicle) a se deslocar em um ambiente e a parar se próximo a uma parede (ou algum obstáculo). Qualquer melhoria introduzida será leva em consideração. Por exemplo, após o UV parar, alguma decisão, como "virar à direita" (ou "esquerda") poderá ser tomada, e o UV iniciar deslocamento novamente. Uma melhoria, por exemplo, eh fazer o sensor de Luz ser indiferente `a Luz ambiente. Ou seja, seu projeto deve permitir ao UAV funcionar em qualquer lugar...
 - Este trabalho envolve o planejamento e construção da parte mecânica e eletrônica do UV, bem como da implementação do programa de controle no Robolab.
 - O objetivo deste primeiro trabalho prático é introduzir os alunos às técnicas LEGO, bem como entender o funcionamento na prática de um pequeno robô autônomo.
-

Trabalho prático #2

Fazer um programa usando o NQC que comande um UV (Umnanned Vehicle) a se deslocar sobre uma mesa e a parar se próximo à uma de suas arestas. Qualquer melhoria introduzida será leva em consideração. Por exemplo, após o UV parar, alguma decisão, como "virar à direita" (ou "esquerda") poderá ser tomada, e o UV iniciar deslocamento novamente.

Este trabalho envolve o planejamento e construção da parte mecânica e eletrônica do UV, bem como da implementação do programa de controle no NQC.

Trabalho prático #3

Fazer um programa usando o NQC que comande um UV (Umnanned Vehicle) a entrar num ambiente composto por 4 paredes e retirar ?pessoas? deste ambiente (ver ?Coppmpeção?), abaixo. Este trabalho envolve o planejamento e construção

da parte mecânica e eletrônica do UV, bem como da implementação do programa de controle no NQC.

Competição

O trabalho #3 será a base da competição. Dois UVs (Umanned Vehicle), um de cada equipe, deverão entrar no ambiente (arena) composto por 4 paredes com uma porta de entrada apenas, e retirar *humanos amigos* deste ambiente. O ambiente deverá ter dimensões de 1,5 por 2,0 metros e os humanos (na realidade, caixinhas feitas de isopor) estarão espalhados aleatoriamente pelo ambiente. Haverão caixinhas de duas cores, bem como estas cores serão diferentes da cor da parede. As paredes são da cor cinza. Os grupos se reuniram para determinar as cores dos humanos e ficou decidido as cores prateado e branco. Estas duas cores não predominam em nenhuma parte do kit lego, para que os humanos possam diferenciar os robôs. Esses humanos terão quatro ganchos encima, um em cada lateral da caixinha de isopor, de forma que uma garra posicionada na frente do UV possa engatar neste gancho (isto seria pegar o humano). Uma outra forma experimentada pelos alunos, a ser utilizada, o robo "agarra" o humano com uma garra. Outra maneira ainda, o robo "engole" o humano, colocando-o embaixo dele e depois o solta quando sair do ambiente. As especificações para o trabalho deverão ser:

Caixa: 5 x 5 cm, com quatro ganchos em forma de trave de gol com 3 cm de altura, acima dos 5 cm, ou seja, uma haste para agarrar o humano poderá pegá-lo se estiver a uma altura entre 5 e 8 cm acima do chão, na frente do robô. Após pega-la, a haste pode levantar a caixinha.

Haverá apenas uma porta de entrada no ambiente. Os humanos deverão ser ?agarrados?, um a um e depositados em área externa ao ambiente. Os robôs entrarão e sairão simultaneamente do ambiente (os alunos devem prever o caso de um robô encontrar outro no caminho).

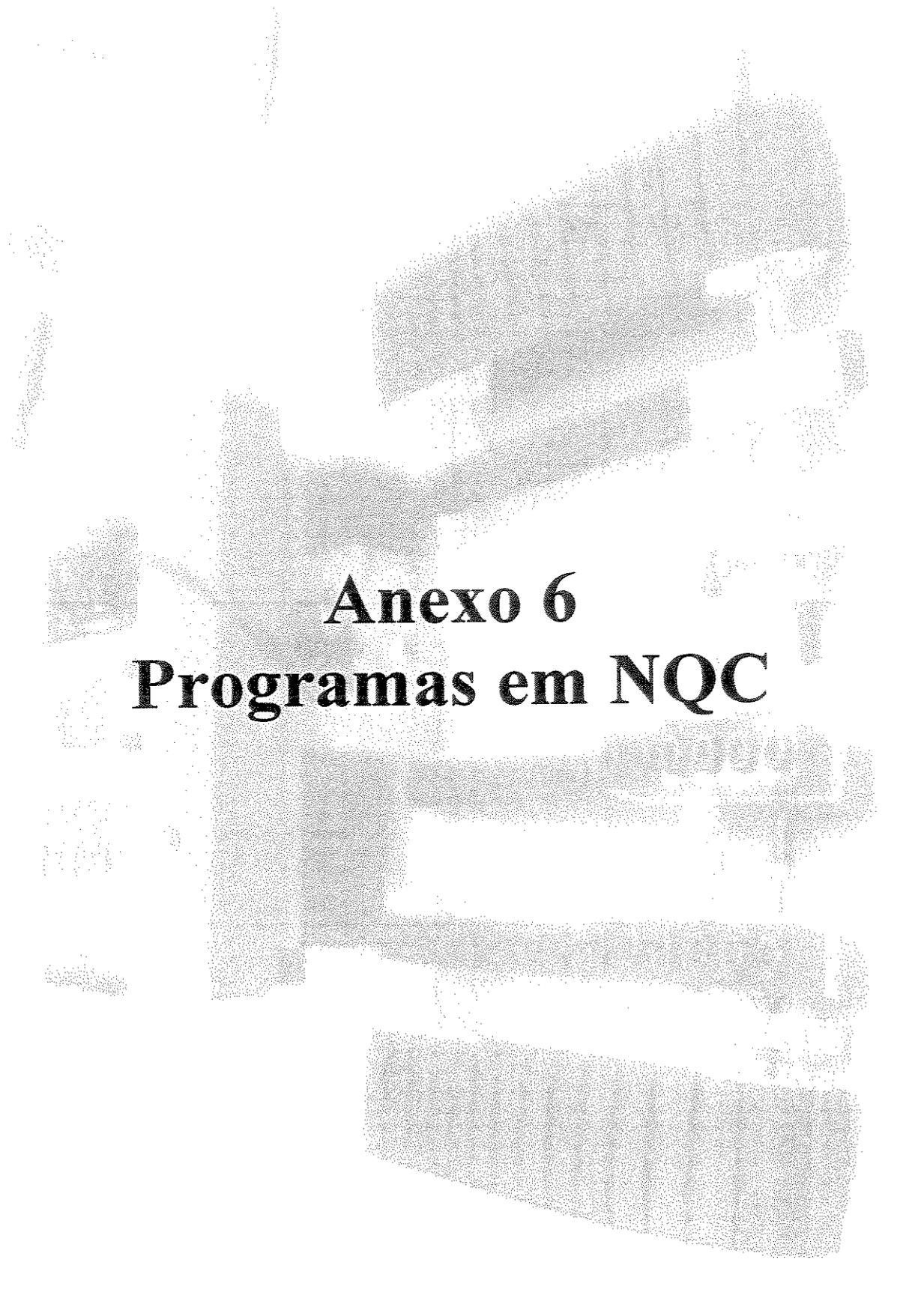
Diversos:

Não será permitido que o robô agarre e atire os humanos para fora do ambiente (o UV deverá sair do ambiente para soltar os humanos)

Sugestões: use o sensor de luz para diferenciar entre objetos diferentes. Façam os testes sempre considerando uma calibragem dos sensores, usando ambientes com diferentes quantidades de iluminação. No dia da competição, a luz será a mesma para todos os grupos e os robos terao um tempo de um minuto para a calibragem. Procure fazer um programa modular, ou seja, cada

módulo é responsável por uma ação (entrar no ambiente, encontrar humano, sair do ambiente, soltar humano).

Por favor, comuniquem qualquer problema encontrado na especificação acima (com sugestões para melhoria:-).



Anexo 6

Programas em NQC

Algoritmos

Aqui estão os algoritmos dos projetos do Lobo Solitário. Eles estão em NQC (Not Quite C). São facilmente inteligíveis.

Não cai de mesas

```
task main() {
    int sensormin;
    int f;
    int x;

    PlayTone(440, 50); // som de início
    // Inicialização dos sensores
    SetSensorType (SENSOR_1, SENSOR_TYPE_LIGHT);
    SetSensorType (SENSOR_2, SENSOR_TYPE_TOUCH);
    SetSensorType (SENSOR_3, SENSOR_TYPE_TOUCH);

    // O Robô vai girar ao calibrar o sensor de luz, para uma calibragem
    mais eficiente.
    SetOutput (OUT_A + OUT_C, OUT_OFF);
    SetDirection (OUT_A, OUT_REV);
    SetDirection (OUT_C, OUT_FWD);
    SetOutput (OUT_A + OUT_C, OUT_ON);

    //Calibrar sensor de luz.

    sensormin = 100;
    ClearTimer(0);
    while(Timer(0) < 52) {
        Wait(1);
        x = SensorValue(0);
        if (sensormin > x) sensormin = x;
    }
    SetOutput(OUT_A + OUT_C, OUT_OFF);

    ClearTimer(1);
    //timer da execução do loop da execução

    do {
```

```

SetDirection(OUT_A + OUT_C, OUT_FWD);
SetOutput(OUT_A + OUT_C, OUT_ON);
Wait(1);

while ((SensorValue(0) > sensormin) && !(SensorValue(1))) &&
(!(SensorValue(2)))) { }
    // Não faz nada de mais se não sentir nada de anormal

    // Achou uma região sem luz frontal - Frente está à beira do abismo
if (SensorValue(0) < sensormin) {
    ClearTimer(0);
    SetOutput(OUT_A + OUT_C, OUT_OFF);
    SetDirection(OUT_A + OUT_C, OUT_REV);
    SetOutput(OUT_A + OUT_C, OUT_ON);
    while(Timer(0) < 5) {}
    do {
        SetOutput(OUT_A + OUT_C, OUT_OFF);
        SetDirection(OUT_A, OUT_REV);
        SetDirection(OUT_C, OUT_FWD);
        SetOutput(OUT_A + OUT_C, OUT_ON);
    } while(SensorValue(0) < sensormin);
    ClearTimer(0);
    while (Timer(0) < 4) { }
}

else if (SensorValue(1)) { // Espada está caída - Lateral está à beira
de um abismo.

    SetOutput(OUT_A + OUT_C, OUT_OFF);
    SetDirection(OUT_A,OUT_REV);
    SetDirection(OUT_C,OUT_FWD);
    SetOutput(OUT_A + OUT_C, OUT_ON);
    while(SensorValue(1)) { }
}

else if (SensorValue(2)) { // Berço bateu - encostou em algo
    SetOutput(OUT_A + OUT_C, OUT_OFF);
    SetDirection(OUT_A, OUT_REV);
    SetDirection(OUT_C, OUT_REV);
    SetOutput(OUT_A + OUT_C, OUT_ON);
    ClearTimer(0);
    while (Timer(0) < 7) { }
}

```

```
SetOutput(OUT_A + OUT_C, OUT_OFF);
SetDirection(OUT_A, OUT_REV);
SetDirection(OUT_C,OUT_FWD);
SetOutput(OUT_A + OUT_C, OUT_ON);
ClearTimer(0);
while (Timer(0) < 4) {}
}

} while (Timer(1) < 1200); // Opera por 2 minutos.
SetOutput(OUT_A + OUT_C, OUT_OFF); // Desliga os motores no final
da operação
}
```

Código do Programa de controle do Afrodite

Versão4 em NQC

```
define SENSOR_ESQ SENSOR_1
#define SENSOR_DIR SENSOR_3
#define SENSOR_LUZ SENSOR_2
#define MOTOR_ESQ OUT_A
#define MOTOR_DIR OUT_C
#define REPETE 500
#define INFINITO 1000
#define TEMPO_GIRO 30
#define TEMPO_VOLTA 7
#define TOQUE 0
#define INCREMENTO 10

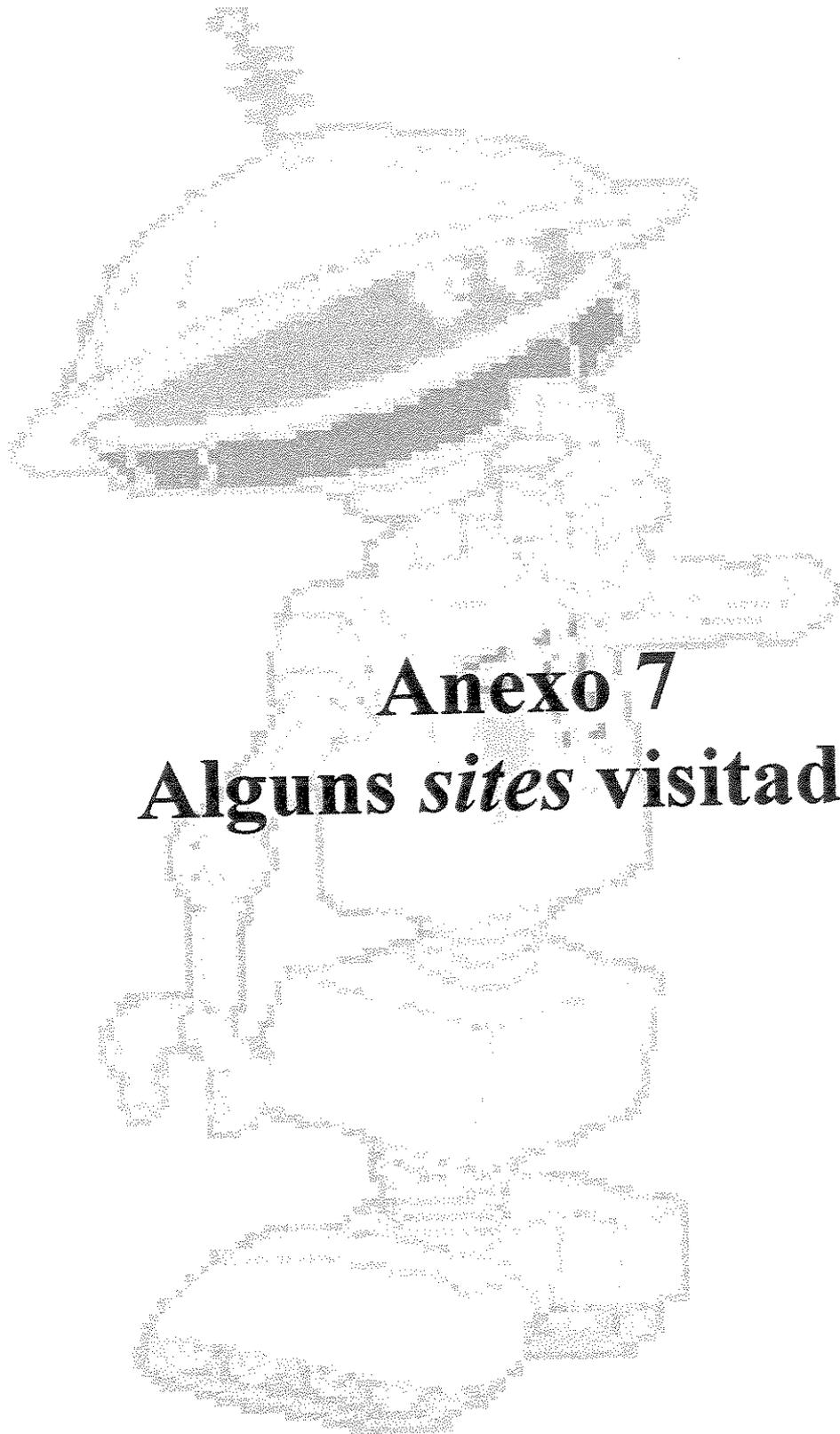
task main()
{
#ifdef __RCX
// RCX needs to be told what kind of sensor is used
SetSensor(SENSOR_ESQ, SENSOR_TOUCH);
SetSensor(SENSOR_DIR, SENSOR_TOUCH);
SetSensor(SENSOR_LUZ, SENSOR_LIGHT);
#endif
int i;
int leitura;
int maximo;
maximo = 0;
for (i=0; i<REPETE; i++) {

    leitura = SENSOR_LUZ;
    if (leitura > maximo){
        maximo = SENSOR_LUZ;
    }
}

    OnFwd(MOTOR_ESQ);
    OnFwd(MOTOR_DIR);

    while(1==1) {
        if (SENSOR_ESQ == TOQUE || SENSOR_DIR == TOQUE || (SENSOR_LUZ >=
maximo+INCREMENTO)) {
            Off(MOTOR_DIR + MOTOR_ESQ);
            OnRev(MOTOR_ESQ + MOTOR_DIR);
            ClearTimer(0);
            while (Timer(0) <= TEMPO_VOLTA) {
```

```
}  
Off(MOTOR_DIR + MOTOR_ESQ);  
OnFwd(MOTOR_ESQ);  
OnRev(MOTOR_DIR);  
ClearTimer(0);  
while (Timer(0) <= TEMPO_GIRO) {  
}  
Off(MOTOR_DIR + MOTOR_ESQ);  
OnFwd(MOTOR_DIR+MOTOR_ESQ);  
}  
}  
}
```



Anexo 7
Alguns sites visitados

LINKS INTERESSANTES

- Página Oficial do Curso de Robótica:

<http://www.ic.unicamp.br/~lmarcos/courses/mo810>

- Página com os links para os outros grupos a disciplina:

<http://www.ic.unicamp.br/~lmarcos/courses/mo810/exercicios/index.html>

Mini How-To para utilizar o Mindstorms (RCX, mais genericamente) em ambiente GNU/Linux:

<http://www.linux.com/howto/mini/Lego/index.html>

Ferramenta para montar virtualmente (algo como um CAD :) coisas com peças LEGO®:

<http://www.ldraw.org/>

- Links que foram utilizados para a resolução das listas de exercícios:

<http://www.cbpf.br/~fmello/curso2.htm>

<http://www.dcc.unicamp.br/~cpg/material-didatico/mo815/9802/curso/index.html>

<http://www.inf.furb.rct-sc.br/~dalton/DiscipCG/ModuloA/moAtop05.htm>

<http://200.211.212.141/aula6ccp.doc>

<http://imagcom.ffclrp.usp.br/~evandro/pim/Aulas/A5/aula5/aula5.html>

<http://envy.cs.umass.edu/~rich/book/the-book.html>

<http://www.emsl.pnl.gov:2080/proj/neuron/neural/what.html>

<http://www.din.uem.br/ia/neurais/>

- Lego Robot Pages (Muita coisa sobre kits que utilizam o RCX, e bastante coisa sobre NQC):

<http://www.cs.ruu.nl/people/markov/lego/>

- Página com um catálogo bem extenso de robôs bípedes feitos utilizando RCX.

<http://www.geocities.com/technicpuppy/lblinks.html>

<http://www.enteract.com/~dbaum/nqc/index.html>

Alguns Links consultados em diferentes oportunidades

<http://pan.nied.unicamp.br/~siros/>

<http://www.enteract.com/~dbaum/nqc/index.html>

http://graphics.stanford.edu/~kekoa/rcx/opcodes.html#set_motor_direction

<http://www.dcpmicro.com/lego/>

<http://www.thetech.org/robotics/activities/index.html>

<http://www.ai.mit.edu/projects/ants/ants.html>

<http://remotebot.k-team.com/index.html>

<http://www.mobotinc.com/main.html>

<http://mindstorms.lego.com/>

<http://www.lego.com/default.asp?x=x>

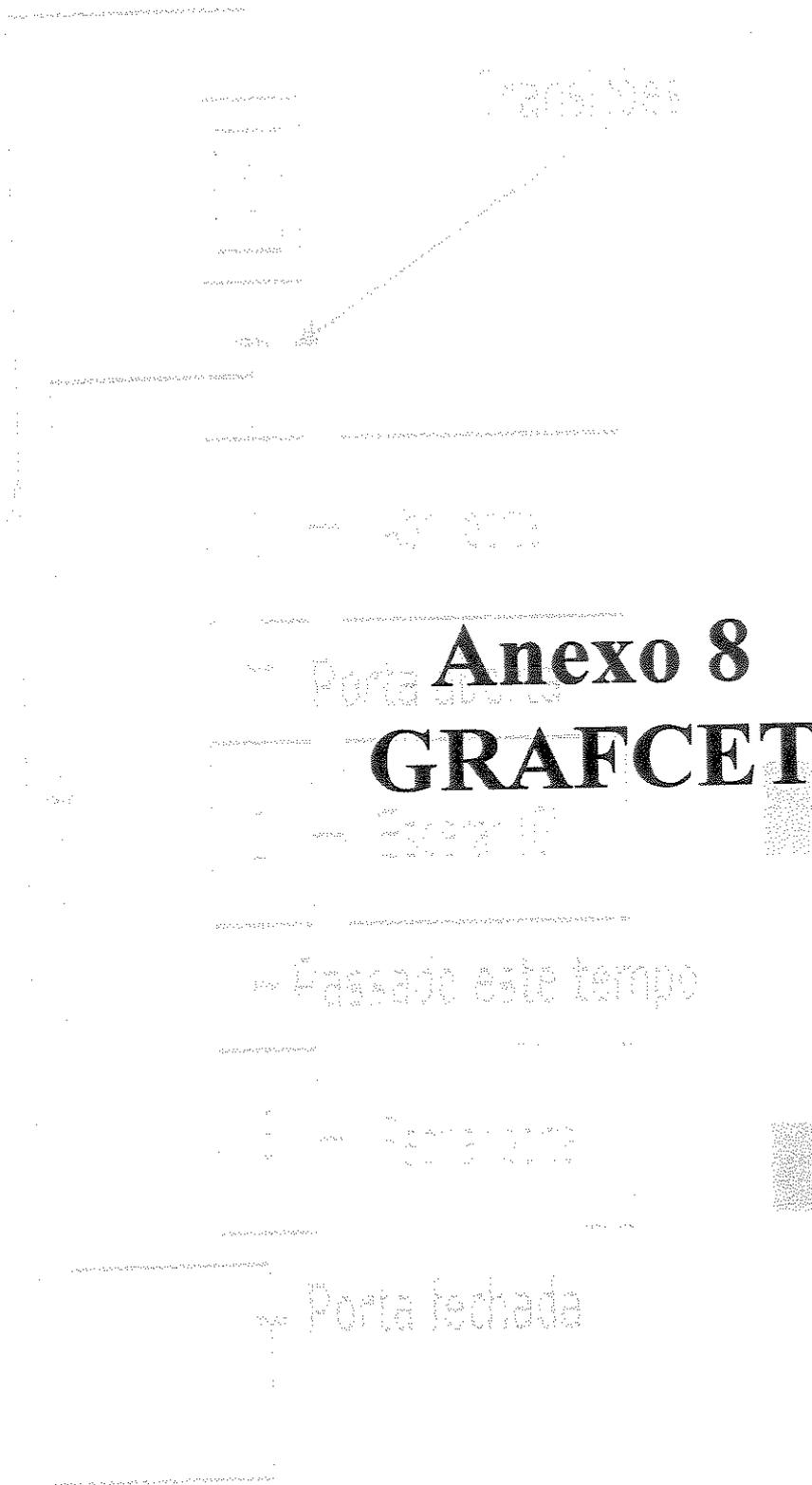
<http://www.lego.com/dacta/robolab/defaultjava.htm>

<http://www.oreilly.com/catalog/lmstorms/chapter/ch04.html>

<http://www.cs.uu.nl/people/markov/lego/index.html>

<http://www.oreilly.com/catalog/lmstorms/building/index.html>

<http://fredm.www.media.mit.edu/people/fredm/mindstorms/index.html>



Anexo 8

GRAFCET

- Transições
- - - Etapa inicial
- Etapa normal
- ▭ Ações associadas
- ▧ Transições
- ▨ Receptividades associadas
- ▩ Linhas de ligação

GRAF CET

A complexidade crescente dos sistemas automatizados industriais implica numa grande dificuldade por parte do usuário, na definição de uma maneira clara, concisa e não ambígua das especificações funcionais associadas a esses sistemas. Portanto, para se descrever o sistema de uma forma adequada nos âmbitos da automação utiliza-se a ferramenta Grafcet. A seguir, será apresentado, de forma detalhada, alguns aspectos desta ferramenta em complemento as informações constantes no capítulo 3. O Grafcet em complemento com os princípios de álgebra de Boole permite algo mais do que a descrição e interpretação gráfica de processos. Ele é uma ferramenta de desenho potente e flexível que, incorporada com linguagens e programação, facilita o desenvolvimento de muitos automatismos.

8.1 Elementos de um Grafcet

Etapas e ações associadas, Transições receptividades e ligações.

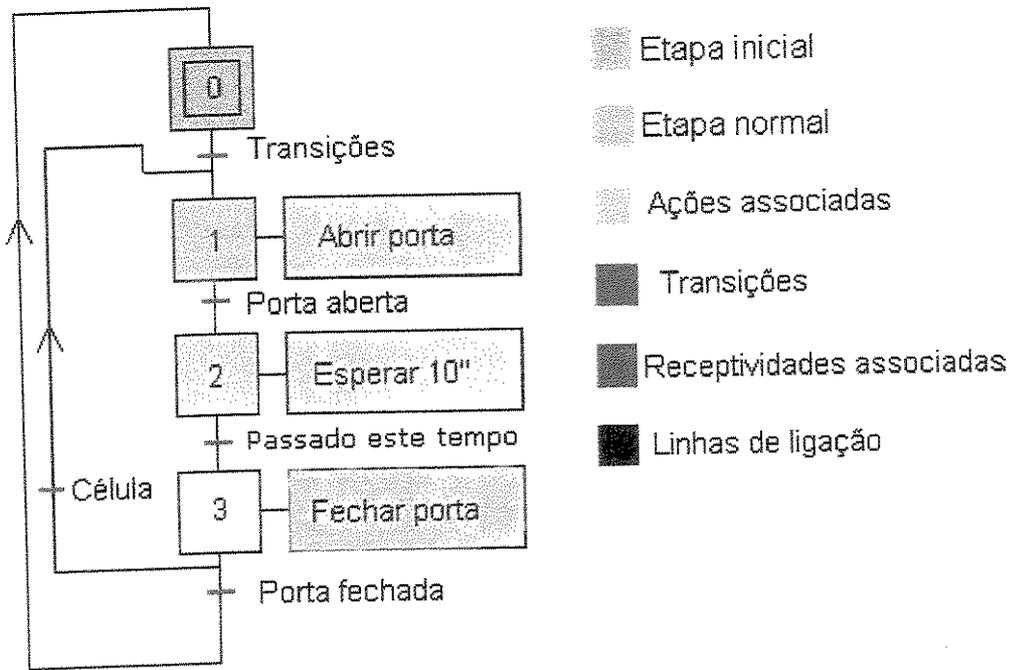


Figura a8.1 – Representação esquemática dos elementos de um Grafcet

8.1.1 Etapas Iniciais

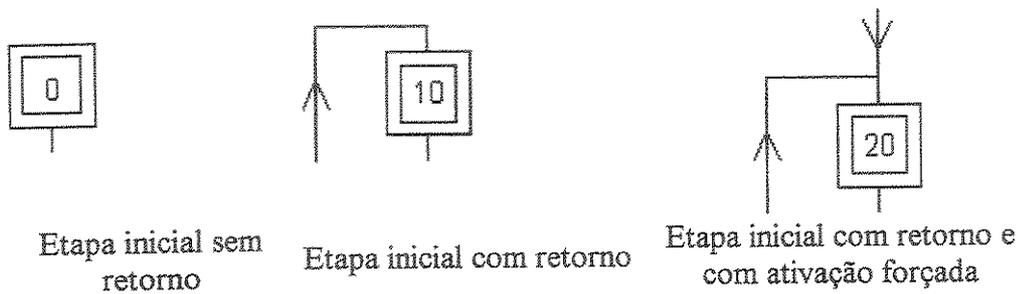


Figura a8.2 - Representação esquemática das etapas de um Grafcet

- Uma etapa inicial é representada por um quadrado duplo.
- As etapas iniciais de um sistema são ativadas ao iniciar o GRAFCET.
- Uma vez iniciadas, as etapas iniciais têm o mesmo tratamento que as outras etapas.
- Um sistema deve ter no mínimo uma etapa inicial.

8.1.2 Etapas Normais



Figura a8.3 - Representação esquemática das etapas normais de um Grafcet

- As etapas representam os estados estáveis do sistema.
- As etapas do GRAFCET são representadas por um quadrado enumerado.
- As etapas podem estar enumeradas; não necessariamente de forma correlatas. Não pode haver duas etapas com o mesmo número.

- As etapas podem estar ativas ou inativas. Ao representar o estado do Grafcet em um dado momento, pode-se indicar que uma etapa está ativa, com um ponto colorido (etapa 4 Figura a8.3).
- Nas etapas, podem ou não haver ações associadas.

8.1.3 Ações Associadas

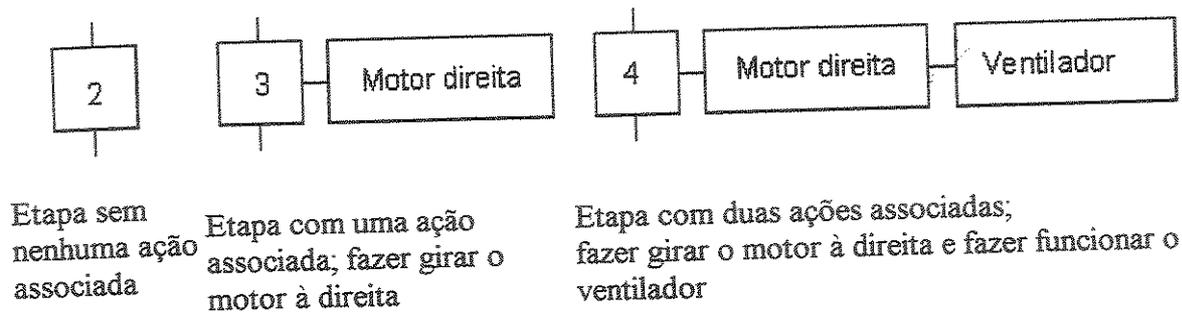
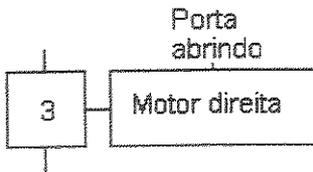


Figura a8.4 - Representação esquemática das etapas e ações associadas de um Grafcet

- Uma etapa sem nenhuma ação associada (etapa 2 da Figura a8.4) pode servir para deter uma ação mono estável que se começou a ser realizada na etapa anterior, ou como etapa de espera.
- Uma ação associada (etapa 3 da Figura a8.4) indica que ao se tornar ativa a etapa, o motor girará para direita
- Em uma etapa pode haver múltiplas ações associadas (etapa 4 da Figura a8.4). Em estando a etapa 4 ativa, o motor girará para direita, e ao mesmo tempo o ventilador estará funcionando.
- Se em um sistema em um dado momento há somente uma etapa ativa, então, só estará funcionando os elementos ativados pelas ações associadas a esta etapa.

1.1.4 Ações Associadas Condicionadas



A ação se realizará em uma ou mais ações associadas a uma etapa, que pode estar condicionada a uma função booleana adicional. Neste caso, o motor girará para direita enquanto estiver ativa a etapa 3. Além disso, a porta ainda não está totalmente aberta à direita.

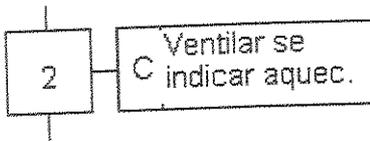
Figura a8.5 - Representação esquemática de ações associadas condicionadas de um Grafset

- No retângulo onde se representa a ação associada, existe uma entrada para as condições.
- A norma IEC-848 propõe as representações para as ações associadas condicionadas conforme a Figura a8.6.

| | |
|----------|-------------------------------|
| C | Ação condicionada |
| D | Ação retardada |
| L | Ação limitada no tempo |
| P | Impulso |
| S | Ação memorizada |

Figura a8.6 - Representação para as ações associadas

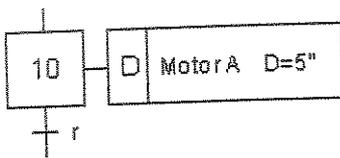
8.1.5 Ação Condicionada



Supondo um sistema em que temos um painel eletrônico, para a regulação de umas máquinas. Estando ativa a etapa de espera 2, e o termostato indicar um sobre aquecimento, o ventilador poderá entrar em funcionamento.

Figura a8.7 - Representação para ações Condicionadas

8.1.6 Ação Retardada

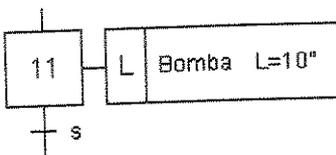


Retardo

O motor A entrará em funcionamento 5 segundos depois de se ativar a etapa 10; se a transição r tornar ativa antes desse tempo o motor não entrará em funcionamento.

Figura a8.8 - Representação para um retardo

1.1.7 Ação Limitada

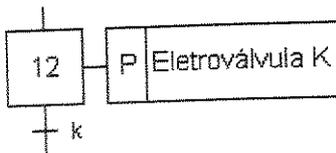


Limitante

A bomba entrará em funcionamento durante 10" depois de ativada a etapa 11, passado este tempo caso não ative a transição s, a bomba deixará de funcionar.

Figura a8.9 - Representação para ação Limitada

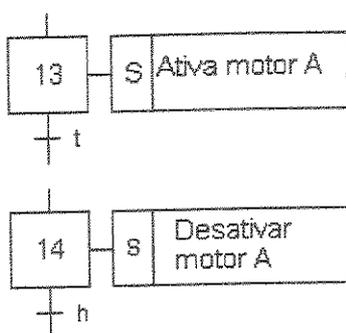
8.1.8 Ação Impulso



Ao ativar-se a etapa 12, se ativará também a eletroválvula K com um impulso de sinal.

Figura a8.10 - Representação para ação Impulso

8.1.9 Ação Memorizada



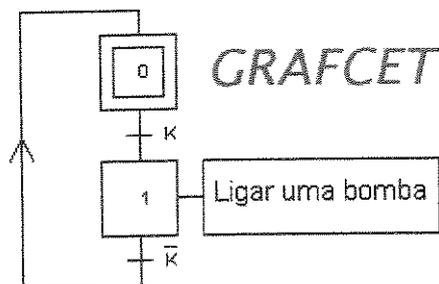
Quando a etapa 13 é ativada, o motor A entrará em funcionamento de forma biestável (set), e ao sair desta etapa, continuará funcionando até que ocorra um reset da ação.

Ao ativar-se a etapa 14, o motor A pára de funcionar uma vez que nesta etapa, a ação faz um reset no motor.

Figura a8.11 - Representação para ação Memorizada

8.1.10 Transições

As transições representam as condições que o sistema deve superar para poder passar de uma etapa para a etapa seguinte. Ao passar uma transição, o sistema deixa de estar em uma etapa e imediatamente vai para a etapa seguinte. Validar a transição implica em trocar de etapas ativas do Grafcet.



As transições são representadas com uma pequena linha horizontal que corta a linha de ligação entre duas etapas. São etapas de entrada a uma transição, todas as etapas que conduzem a uma transição. São etapas de saída a uma transição, as etapas que saiem de uma transição.

Figura a8.12 - Representação das transições de um Grafcet

Receptividades associadas às transições

- A condição ou condições que se devem superar para poder passar de uma transição, recebem o nome de receptividades. Em uma transição podemos ter:

- Uma condição simples [Pm];
- Uma função booleana [(Pm+Pk)*Pp'];
- Sinal de um temporizador o contador [T03]
Neste caso, é habitual que o temporizador seja ativado para contar a ação associada a etapa de entrada;
- A ativação de outra etapa do Grafcet [X12], onde X indica que a receptividade está condicionada ao ponto que a etapa (neste caso a 12) está ativa.

8.1.11 Linhas de Ligação

São linhas verticais ou horizontais que unem com uma direção significativa (a não ser que se indique o contrário de cima para baixo), as distintas etapas com as transições, e as transições com as etapas.

8.1.12 Desenho e Estruturas

- Desenvolvimento de Sistemas
- Estruturas Básicas
- Seqüência única
- Bifurcação em OU Seleção de seqüência
- Bifurcação em Y
- Trabalhos em paralelo
- Saltos de etapas
- Estruturas receptivas
- Subrotinas
- Macro-etapas
- Diagramas paralelos

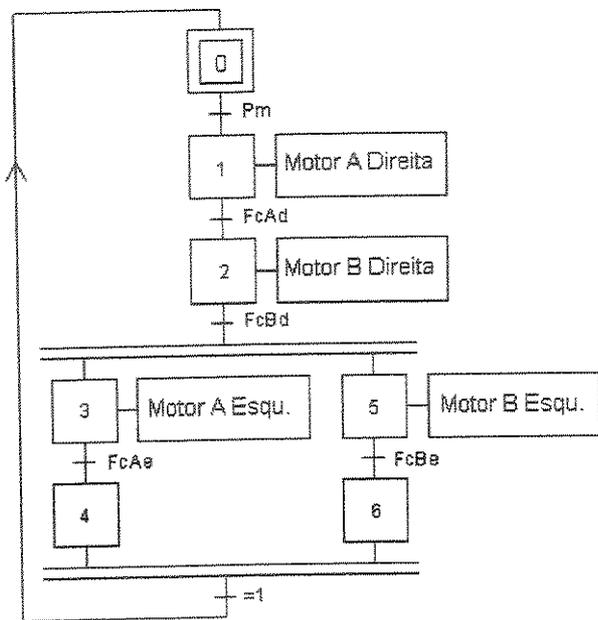
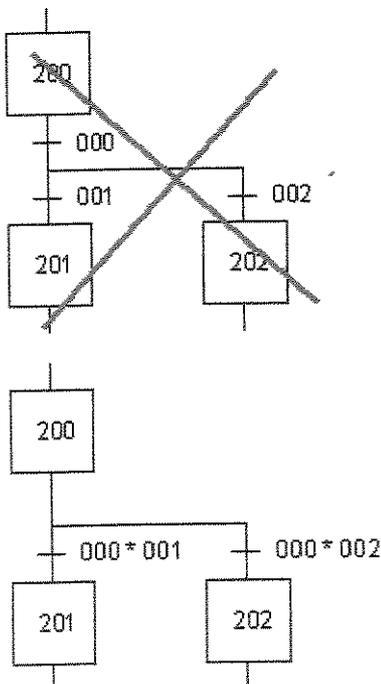


Figura a8.13 - Representação de diagramas paralelos em um Grafcet

8.1.13. Desenvolvimento do sistema

- Diagrama é desenhado com uma sucessão alternada de etapas e transições.
- Não pode haver duas etapas seguidas, nem tampouco duas transições seguidas.

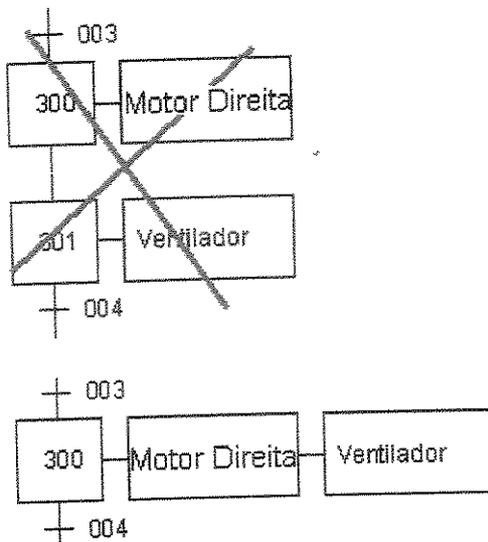


Entre as etapas 200 e 201 ou entre as etapas 200 e 202 existe duas condições para a transição (000 e 0001 ou 000 e 002).

Neste caso isto pode ser resolvido fazendo com que a receptividade da transição se for válida o resultado da função booleana AND.

$(000 * 001)$ ou $(000 * 002)$

Figura a8.14 - Representação de diagramas paralelos em um Grafcet



Ao se satisfazer à condição 003 da transição, o motor deve girar para a direita e também o ventilador deve ser acionado.

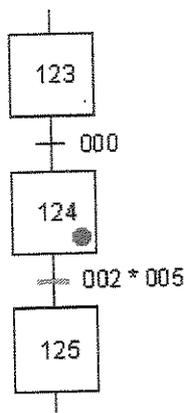
Para realizar isso é necessário colocar todas as ações associadas na mesma etapa.

Não pode haver duas etapas seguidas, nem duas transições também.

Figura a8.15 - Representação de ligação de etapas em um Grafcet

Para que o sistema possa evoluir é necessário:

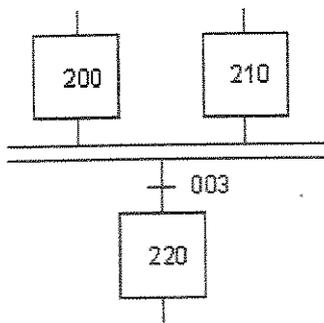
- a) Validar a transição. Todas as etapas de entrada da transição devem estar ativas.
- b) As receptividades associadas devem estar corretas. Devem estar corretas também as condições da transição.



A primeira transição só poderá ser validada se a etapa 123 estiver ativa, e, além disso, se a condição 000 for satisfeita. Neste instante, deixa de estar ativa 123, e o 124 entra em evidencia.

O Grafcet evolucionará para a etapa 125, se estiver ativa a etapa 124 e também se a condição 002* 005.

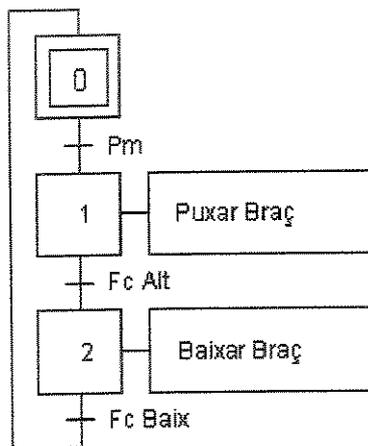
Figura a8.16 - Representação de ligação de etapas e transições em um Grafcet



As etapas 200 e 210 são etapas de entrada para a transição. Para validar a transição, devem estar ativas as duas etapas. Para poder entrar a etapa 220, a transição tem que estar validada e deve estar satisfeita a receptividade (003) associada a transição.

Figura a8.17 - Representação de ligação de etapas para transições em um Grafcet

8.1.14 Seqüência Única

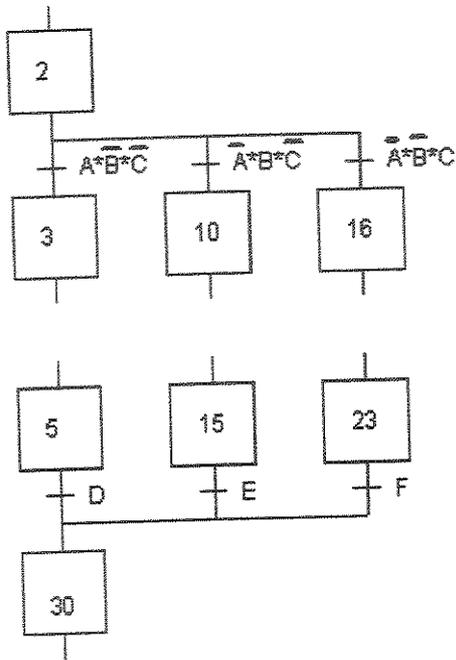


Um Grafcet será de seqüência única quando no seu diagrama há uma só ramificação: o conjunto de etapas se ativará uma após a outra depois de se validar as recepções associadas às transições.

Figura a8.18 - Representação de uma seqüência única em um Grafcet

8.1.15 Bifurcação em OU Seleção de seqüência

Haverá uma seleção de seqüências, quando ao chegar a um ponto encontramos uma bifurcação em OU.

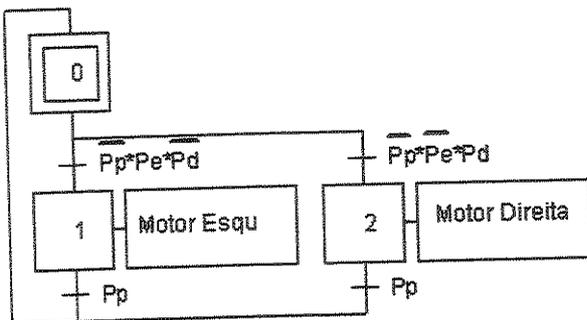


Na bifurcação será necessário escolher qual das distintas sucessões de etapas e transições se deve seguir

Não é necessário que os diferentes caminhos tenham o mesmo número de etapas; mas é conveniente que as receptividades associadas às transições sejam excludentes entre si.

Figura a8.19 - Representação de uma bifurcação em um Grafcet

8.1.16 Giro para direita e esquerda de um motor

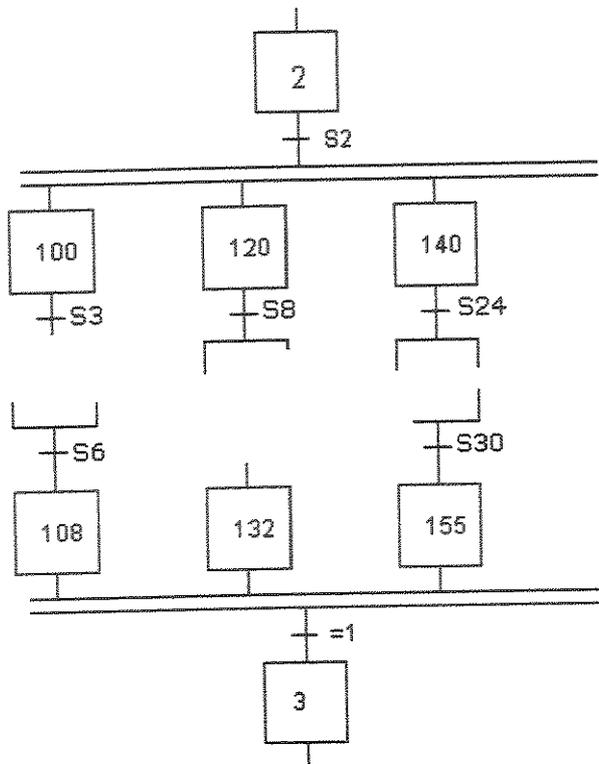


Para seleccionar o sentido de giro de um motor, utilizamos a bifurcação em OU.

Um motor pode girar:
Para direita OU para a esquerda.

Figura a8.20 - Representação de giros em um Grafcet

8.1.17 Bifurcação em Y



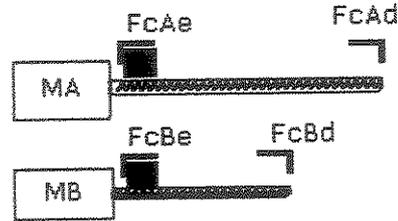
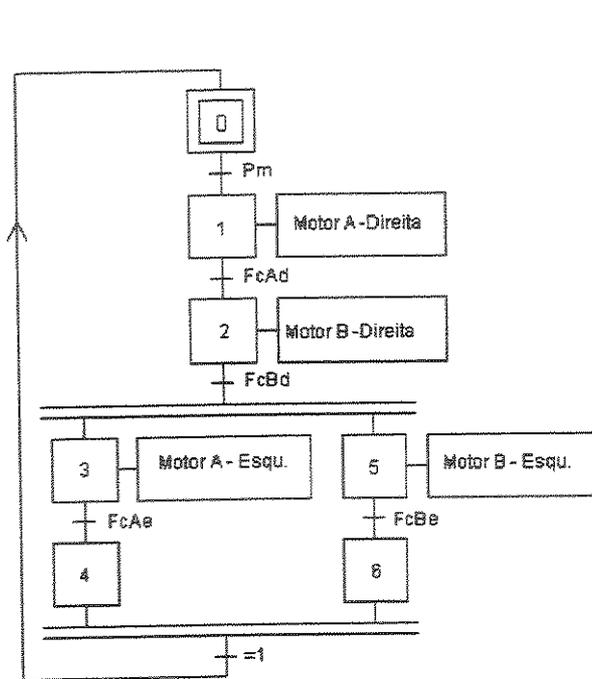
Em automatismo haverá uma bifurcação em Y ou “Trabalhos paralelos”, quando a partir de um ponto, ela evoluir de forma simultânea em todas as ramificações. No fim destas encontraremos umas etapas de espera. (108, 132, 155).

O sistema continuara sua evolução, quando cada uma das ramificações chegarem a sua etapa de espera.

Os nomes das etapas das diferentes ramificações podem ser distintos um do outro.

Figura a8.21 - Representação de uma bifurcação em Y em um Grafcet

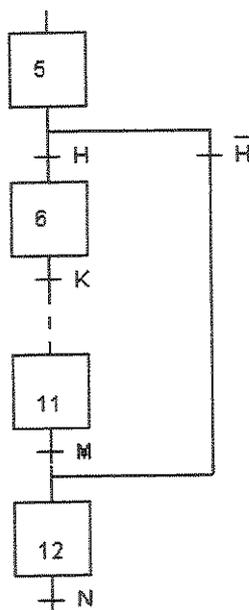
8.1.18 Motores com trabalhos simultâneos



Dois motores MA e MB deslocam uma peça. Primeiro o motor MA vai de FcAe para FcAd, então é o MB quem vai de FcBe até FcBd. Depois os dois voltam para a posição inicial FcAe e FcBe. O ciclo se reinicia quando os dois motores estão de novo nas posições iniciais.

Figura a8.22 - Representação de motores com movimentos simultâneos em um Grafset

8.1.19 Saltos de Etapas

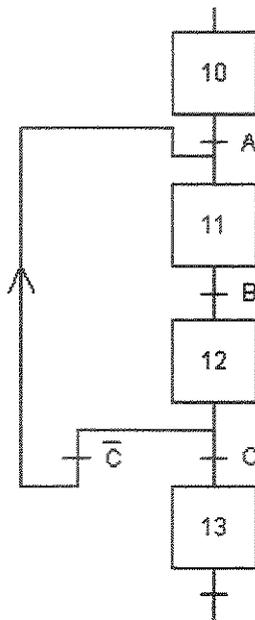


Em um ponto pode haver uma bifurcação que provoque um salto sobre um conjunto de etapas. Que se siga ou não a seqüência completa ou ocorre o salto, determinado pela estado da condição da transição (H).

Tem-se que ter em mente que as condições de entrada não devem ser excludentes (H e H). Pode ser realizado também salto em sentido ascendente (neste caso indicado por linhas de ligação).

Figura a8.23 - Representação de salto de etapas em um Grafset

8.1.20 Estruturas Repetitivas



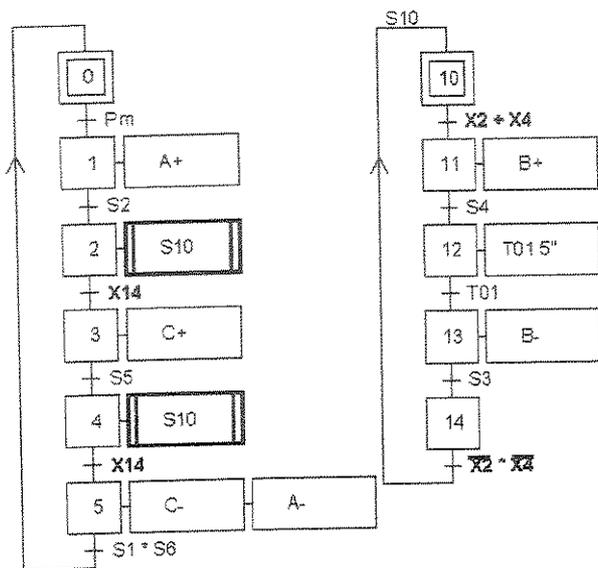
Haverá uma estrutura repetitiva, quando uma ou um conjunto de etapas se repetem, várias vezes, (controladas por um temporizador, um contador ou até que seja satisfeita uma determinada condição).

O ciclo de lavagem de uma lavadora de roupa repete várias vezes esta estrutura (giro para direita, espera, giro para esquerda, espera).

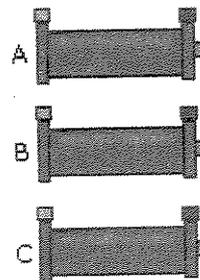
Figura a8.24 - Representação de estruturas repetitivas em um Grafset

8.1.21 Sub-rotinas

Uma sub-rotina é uma parte de um programa que realiza uma tarefa concreta, a qual pode ser solicitada uma ou várias vezes ao programa principal. Uma vez realizada as ações da sub-rotina o programa continua do ponto onde estava.



Os trabalhos desenvolvidos em um automatismo podem se dividir entre diferentes diagramas. Pode haver um diagrama principal (0-5) e outros secundários (10-14) que fazem determinadas funções que uma vez realizadas devolvem o controle ao diagrama principal.



Ao chegar a etapa 2 ou 4 do primeiro diagrama se for válida a transição $X2+X4$ começa a sub-rotina. Ao chegar a etapa 14 se válida a transição $X14$ e continua a evolução do diagrama principal para as etapas 3 ou 5, respectivamente.

Figura a8.25 - Representação de uma sub-rotina em um Grafcet

8.1.22 Macro-Etapas

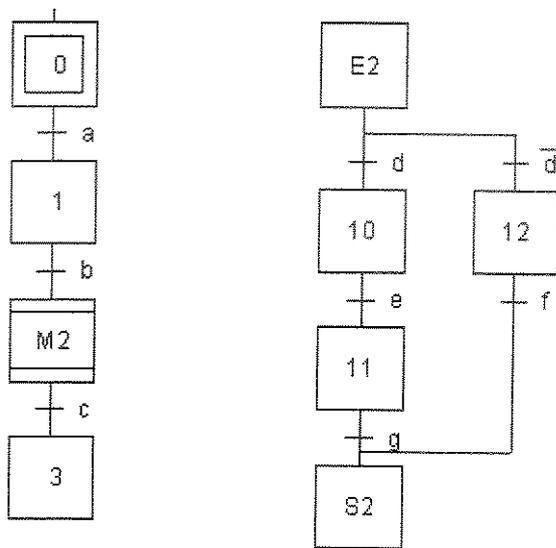
Ao fazer a descrição do automatismo, o Grafcet permite começar a partir de um ponto de vista muito geral e a partir dele fazer descrições cada vez mais concretas do processo de controle.

O esquema se realiza de forma descendente, em grandes blocos que se vão resolvendo de forma modular.

Uma macro-etapa é a representação mediante uma única etapa, de um conjunto de etapas, transições e ações associadas, as quais chamamos de expansão da macro-etapa.

A expansão da macro-etapa é na realidade uma parte do diagrama de Grafcet, com suas etapas, transições e normas de evolução, mas que em um esquema descendente englobamos em uma macro-etapa.

Podemos dizer que ao fazer a expansão da macro-etapa, na realidade o que fazemos é uma espécie de “zoom”, que nos ensina em detalhe, etapas, transições e ações concretas, que antes havia se referido de forma geral.



O diagrama principal evolui a partir da etapa 0 e a Transição a, uma vez ativa a etapa 1, a transição b estará receptiva, e ao validar-se, entraremos na macro etapa M2, a etapa E2 estará ativa, e segundo o estado da transição d, evoluirá até a etapa 10 ou a 12, e ao chegar a etapa S2 voltará ao diagrama principal.

A etapa E2 é a etapa de entrada da macro 2, a etapa S2, é a etapa de saída da macro 2.

Figura a8.26 - Representação de uma Macro Etapa um Grafcet

8.1.23 Diagramas Paralelos

Para resolver um automatismo, podem ser descritos diferentes diagramas paralelos, que evoluem cada um deles em separado e ao seu ritmo. Estes podem em vários pontos, ter ou não relação entre si.

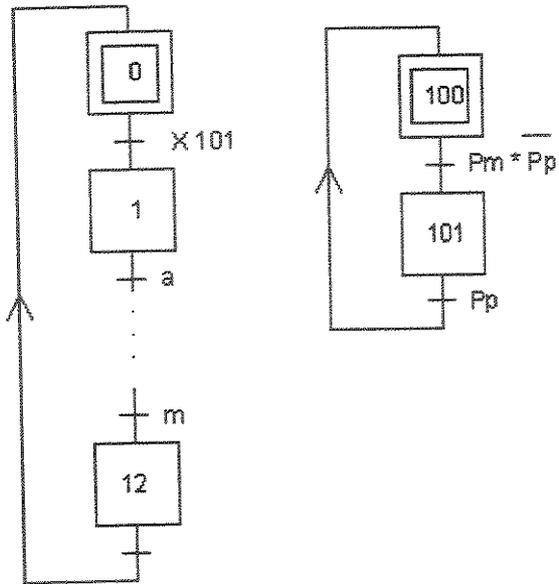


Figura a8.27 - Representação de um diagrama paralelo em um Grafcet

8.2 Implementação Tecnológica

Implementação

Com elementos monoestáveis

- Portas lógicas
- Diagrama de relês

Com elementos biestáveis

- Flip-Flop
- Funções biestáveis

8.2.1 Normas de ativação de Grafcet

Norma 1: Inicialização

Na inicialização do sistema deve-se ativar as etapas iniciais, as outras etapas devem permanecer inativas.

Norma 2: Evolução das transições

Para poder validar uma transição, é necessário que todas as suas etapas de entrada estejam ativas. Para poder supera-la tem-se que garantir que a receptividade associada à transição esteja correta.

Norma 3: Evolução das etapas ativas

No momento de superar uma transição deve-se ativar todas as suas etapas de saída e ao mesmo tempo desativar todas as suas etapas de entrada.

Norma 4: Simultaneidade na validação das transições

Se duas transições são simultaneamente válidas elas devem ocorrer de forma simultânea

Norma 5: Prioridade de ativação

Se uma etapa do Grafcet é ativada e desativada ao mesmo tempo, ela deve ficar ativa.

8.2.2 Soluções tecnológicas para a implementação do Grafcet

As especificações tecnológicas precisam o modo como um automatismo deverá inserir-se fisicamente no conjunto que constitui o sistema automatizado que o rodeia (Rosário et al, 1997). As soluções tecnológicas dizem respeito ao Grafcet nível 2.

No Grafcet nível 2 podemos por exemplo considerar, conforme a Figura a8.28, que se uma etapa ET2 está ativa e a transição b é correta, o sistema fará com que a etapa ET3 fique ativa, e o ET2 se desative e ET3 permaneça ativa até que em outro momento se ative o ET4.

$$ET3 = (ET2 * b + ET3) * ET4$$

A representação desta equação com portas lógicas é:

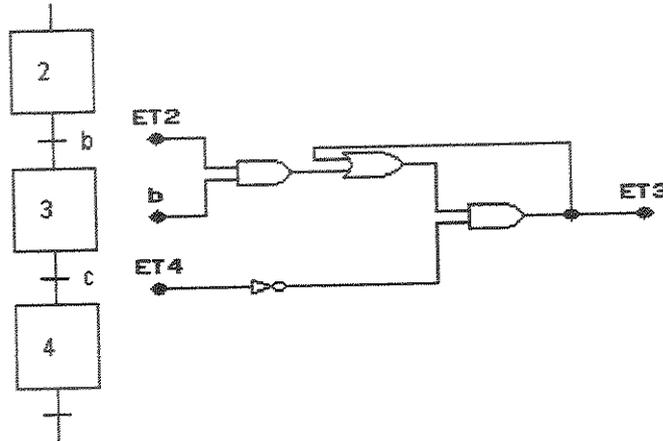


Figura a8.28 - Representação de uma solução tecnológica utilizando portas lógicas

Com esquemas de relês

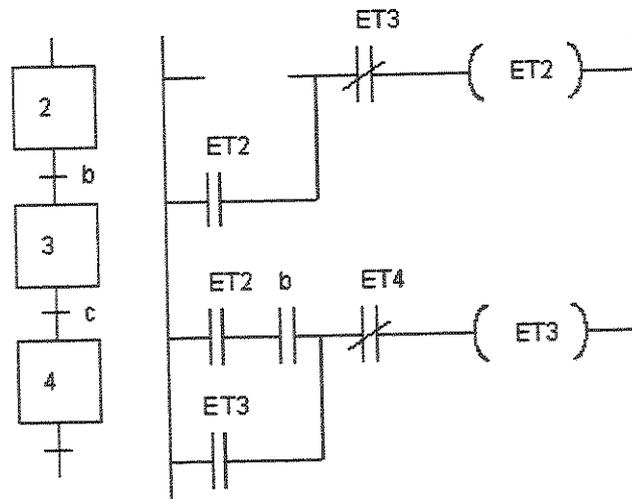


Figura a8.29 - Representação de uma solução tecnológica utilizando relês

Utilizando Flip-Flops biestáveis

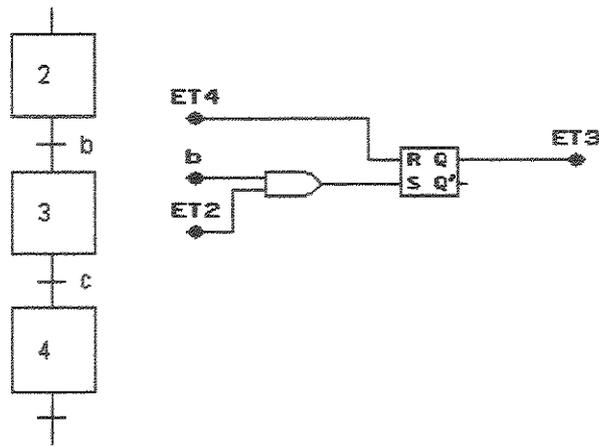


Figura a8.30 - Representação de uma solução tecnológica utilizando Flip-Flops

Com funções biestáveis

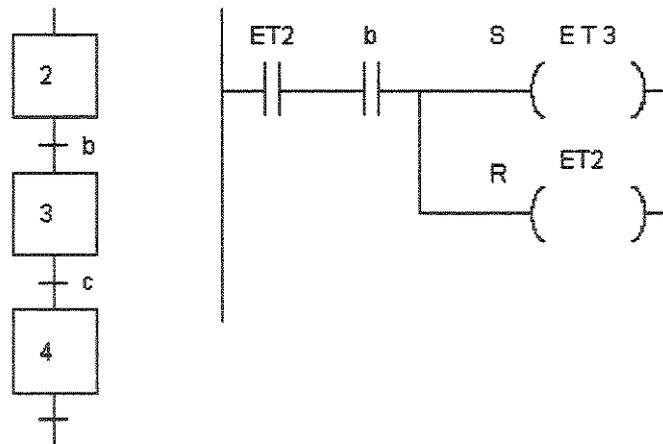


Figura a8.31 - Representação de uma solução tecnológica utilizando funções biestáveis

As funções biestáveis são úteis para tornar mais fácil a implementação. Estando ativa a etapa anterior se satisfeita as condições, a transição deve ativar a etapa seguinte e desativar a etapa atual (anterior).

8.3. Exemplos de especificação de projetos utilizando Grafcet

Exemplo 1

Especificação para uma porta automática. Uma vez acionado o motor, a porta se abre, permanece aberta por 10 segundos e se fecha. Automatizar a abertura e o fechamento da porta a partir de um motor monofásico contendo jogos de engrenagens que permitem o deslocamento da mesma.

1. Para controlar o sentido de giro do motor utilizou-se contatores K1 e K2 (ver Figura a8.32).
2. Para saber se a porta está aberta ou fechada existe duas chaves fim-de-curso S1 e S2.
3. Para abrir a porta utilizamos um sistema de comando à distância com sinal de rádio.
4. Para evitar acidentes, ao fechar a porta, utilizou-se uma célula fotoelétrica.

Grafcet Nível 1

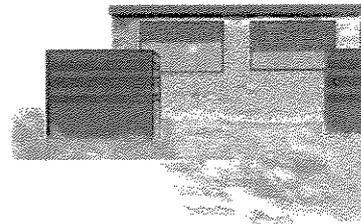
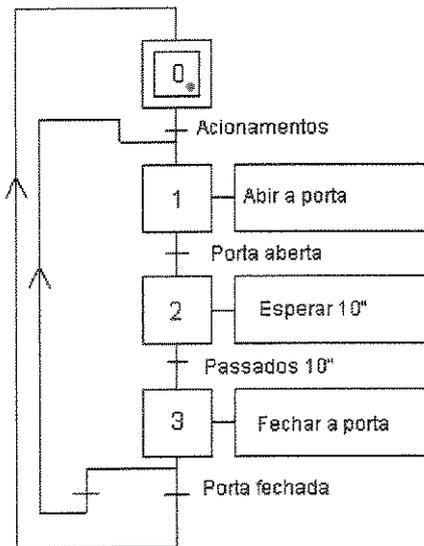


Figura a8.32 – Representação em Grafcet e uma imagem da porta automática.

Exemplo 2

Especificação para um carro com temporizador contendo um motor cujo acionamento é feito em 10 segundos no sentido horário e em seguida é invertido o sentido para anti-horário como pode-se ver na Figura a8.33.

X0 - acionamento do motor
Y0 - motor SH
Y1 - motor SAH

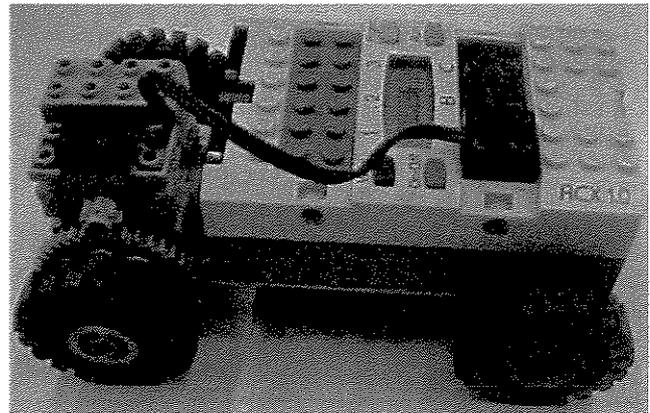
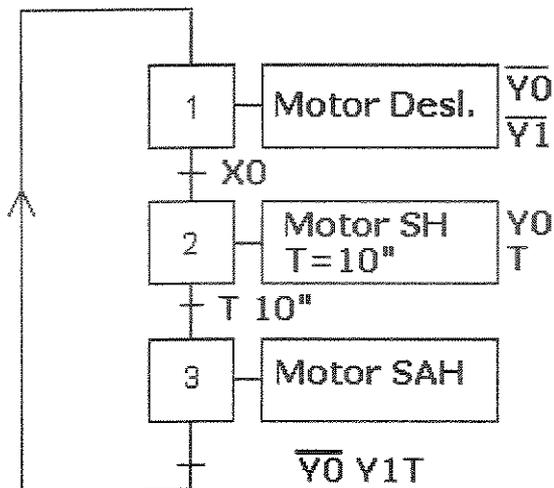


Figura a8.33 – Representação em Grafcet e a imagem do carro com temporizador

Este projeto foi desenvolvido pelos alunos da disciplina INTRODUÇÃO À ENGENHARIA DE CONTROLE E AUTOMAÇÃO no primeiro semestre de 2001, durante uma das atividades de montagem e automação de dispositivos mecatrônicos com LEGO.

8.4 Sintetizando GRAFCET

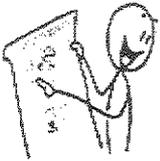


O Grafcet é um método para resolver problemas de automação de forma sistematizada. Uma vez definido o problema e montado/desenhado o Grafcet de nível 1, a adaptação para a solução tecnológica concreta desejada, se realiza mecanicamente.

Pode-se implementar o Grafcet de qualquer tecnologia. Por isso, ele é um método muito indicado para circuitos integrados programáveis e autômatos programáveis.



O Grafcet descreve os processos a automatizar, remarcando as ações e os eventos que as provocam. É uma representação diretamente relacionada à evolução do processo, facilitando o diálogo entre pessoas de especialidades distintas e de diferentes níveis de formação, a partir do momento da concepção do automatismo, e depois das fases de manutenção. Na reparação de avarias em máquinas, uma vez que estas entram em funcionamento, os problemas ficam delimitados sobre uma etapa e entre umas transições.



O Grafcet ajuda nas diferentes fases de desenvolvimento do automatismo:

Planejamento, o Grafcet é um método gráfico de representação. É independente da tecnologia empregada na resolução final. Isso permite planejar o problema, discuti-lo e resolve-lo entre todos (na prancheta), pelo menos em seus componentes funcionais.

Resolução tecnológica, nesta fase, a discussão sobre a tecnologia a ser empregada acontece valorizando as vantagens e os inconvenientes de cada tecnologia ou sistema. Pode-se também planejar o mesmo problema sobre diferentes tecnologias e uma vez realizados os projetos, discutir e avaliar os custos vantagens.

Realização do automatismo, a metodologia do Grafcet ajuda a fazer a implementação de forma mecânica. É importante documentar o projeto de forma completa.

Comprovação do funcionamento, nesta fase, o Grafcet permite fazer



o acompanhamento etapa por etapa do automatismo. Se aparecer alguma disfunção, ela estará localizada em uma etapa ou em uma transição concreta, facilitando assim a solução. Com o Grafcet, a localização de avarias é feita de forma sistematizada.

Aumento da melhoria, uma vez o sistema terminado e em funcionamento, uma revisão em grupo a partir do Grafcet de nível 1 permitirá analisar as especificações para a melhoria do sistema como: elementos de segurança e sistemas de detecção de avarias.

