

ESTE EXEMPLAR CORRESPONDE A REDAÇÃO FINAL  
TESE DEFENDIDA POR Antônio Carlo  
Baffi E APROVADA PE  
COMISSÃO JULGADORA EM 26/02/2002

João Maurício Rosário  
ORIENTADOR

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

**“Projeto e Implementação de  
uma Plataforma Didática de Manipulação,  
Utilizando Conceitos de Automação e Integração”**

Autor: **Antônio Carlos Baffi**  
Orientador: **João Maurício Rosário**

19/02

**UNIVERSIDADE ESTADUAL DE CAMPINAS  
FACULDADE DE ENGENHARIA MECÂNICA  
DEPARTAMENTO DE PROJETOS MECÂNICOS**

**“Projeto e Implementação de  
uma Plataforma Didática de Manipulação,  
Utilizando Conceitos de Automação e Integração”**

**Autor: Antônio Carlos Baffi**  
**Orientador: João Maurício Rosário**

Curso : Engenharia Mecânica  
Área de Concentração : Projetos Mecânicos

Dissertação de tese de mestrado apresentada à comissão de Pós Graduação da Faculdade de Engenharia Mecânica , para obtenção do título de Mestre em Engenharia Mecânica.

Campinas, 2002  
S.P. – Brasil

UNIDADE 30  
Nº CHAMADA T/UNICAMP  
B145p  
V \_\_\_\_\_ EX \_\_\_\_\_  
TOMBO BCI 50576  
PROC 16.837102  
C \_\_\_\_\_ DX \_\_\_\_\_  
PREÇO R\$ 11,00  
DATA 24/08/02  
Nº CPD \_\_\_\_\_

CMO0172426-4

BIB ID 253636

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

B145p Baffi, Antônio Carlos  
"Projeto e implementação de uma plataforma didática de manipulação, utilizando conceitos de automação e integração" / Antônio Carlos Baffi.--Campinas, SP: [s.n.], 2002.

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

1. Automação. 2. Tecnologia. 3. Robôs - Sistemas de controle. I. Rosário, João Maurício. II. Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica. III. Título.

UNIVERSIDADE ESTADUAL DE CAMPINAS  
FACULDADE DE ENGENHARIA MECÂNICA  
DEPARTAMENTO DE PROJETOS MECÂNICOS

DISSERTAÇÃO DE MESTRADO

**“Projeto e Implementação de  
uma Plataforma Didática de Manipulação, Utilizando  
Conceitos de Automação e Integração”**

Autor: **Antônio Carlos Baffi**  
Orientador: **João Maurício Rosário**



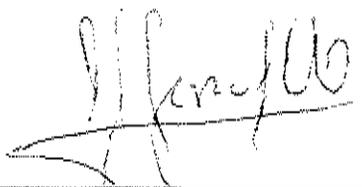
---

Prof. Dr. João Maurício Rosário – Orientador  
FEM/ UNICAMP



---

Prof. Dr. Marcius Fabius Henriques de Carvalho  
FEM/ UNICAMP



---

Prof. Dr. Humberto Ferasoli Filho  
Unesp/Baurú

Campinas, 26 de fevereiro de 2002.

UNICAMP  
BIBLIOTECA CENTRAL

002 39579

**Dedicatória:**

**Dedico este trabalho à minha mãe e às minhas filhas.**

## **Agradecimentos**

Este trabalho não seria concluído sem a ajuda de várias pessoas, as quais agradeço e presto minha homenagem e consideração.

A minha filha Daniela, sempre presente com sua preocupação e carinho. Ao meu orientador, prof. Dr. João Maurício Rosário, pela forma como conduziu este trabalho, assim como a confiança depositada em mim.

Aos amigos: Almiro, Édna, Marcílio, Marcelo, Maurício e todos os demais que de uma forma direta ou indireta colaboraram na conclusão deste trabalho.

“ Um professor educador sempre afeta a realidade. Ele nunca saberá onde sua influência termina”.

( Henry Adams )

## **RESUMO**

BAFFI, Antonio Carlos, Projeto e Implementação de uma Plataforma Integrada de Manipulação, Campinas, Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2001, 124 p. Dissertação ( mestrado ).

Os avanços da microeletrônica e dos sistemas computacionais proporcionaram um grande impulso a automação industrial. A automatização dos meios de produção de forma integrada e flexível, tornou decisivo para que as indústrias atingissem um elevado grau de competitividade. Focando esse novo contexto do setor produtivo industrial, este trabalho tem como objetivo abordar o tema e apresentar o projeto e implementação de uma plataforma constituída de dois manipuladores ( um pneumático e outro robótico ) e uma mesa de transferência como elo de ligação. Para modelagem do sistema em estudo foi utilizado o grafcet ( grafos de etapa e transição ), permitindo o gerenciamento lógico entre entradas e saídas. A parte comando é representada por sensores, chaves, CLP e botões e a parte operativa consiste de atuadores pneumáticos, motor cc, solenóide e sinalizadores. O projeto implementado, além de servir para validação do trabalho proposto , propiciará a formação de profissionais em Automação Industrial ( propiciando a consolidação dos conhecimentos adquiridos nas aulas teóricas e conceituais ).

Palavras chaves :

Automação, Grafcet, Integração, Tecnologia, CLP, Flexibilidade, Robótica e Controle.

## **Abstract**

BAFFI, Antônio Carlos, Design and Implementation of a platform Automated of manipulation to Validation of Conceptual of Integration of Systems, Campinas : Faculdade de Engenharia Mecânica , Universidade Estadual de Campinas , 2002, 124p. dissertação ( mestrado )

Advances in microelectronics and in computational systems led to significant development in industrial automation. The production automatization in an integrated and flexible way has become an important step for the industries to reach a high degree of competitiveness. By focusing on this new context of the industrial productive sector, the aim of the present work is to approach the theme and to present the design and implementation of a platform with two manipulators ( a pneumatic one and a robotic one ) having and a transfer rotating table as a link. The grafcet was used to model the whole system, allowing the logical management of inputs and outputs. The command part is represented by sensors, keys, PLC and buttons while the operative part consists of pneumatic performers, motors, solenoids and signallers. The results of the implantation of this project will allow not only the validation of the present work, but will also contribute to the training of professionals in Industrial Automation, as a complementary means to theoretical and conceptual classes.

Key words

PLC, Grafcet, Technology, Control, Robotic, Industrial Automation, Integration Systems.

## ÍNDICE

<b>CAPÍTULO 1 - Introdução</b>	01
<b>1.1 Objetivos do Trabalho</b> .....	02
<b>1.2 Delineamento do Trabalho</b> .....	02
<b>CAPÍTULO 2 - Automação Industrial</b>	04
<b>2.1 Vantagens da Automação</b> .....	06
<b>2.2 Definição e História da Automação</b> .....	06
<b>2.3 Inovação Tecnológica</b> .....	07
<b>2.4 Evolução do Controle Industrial</b> .....	09
2.4.1 Controle Numérico.....	09
2.4.2 Controle de Processos Industriais.....	10
2.4.3 Definição de Processo Industrial.....	10
<b>2.5 Classificação de Sistemas Automatizados</b> .....	12
2.5.1 Sistemas Flexíveis de Manufatura – SFM.....	14
2.5.2 Sistemas Integrados de Manufatura.....	15
<b>2.6 O Impacto da Automação na Sociedade</b> .....	16
<b>2.7 Conclusão</b> .....	18
<b>CAPÍTULO 3 - Sistema Automatizado de Produção – SAP</b>	19
<b>3.1 Introdução</b> .....	19
<b>3.2 Grafcet</b> .....	24
3.2.1 Etapas.....	26
3.2.2 Transições.....	26
3.2.3 Ação.....	28
3.2.3.1 Ordem Memorizada ( S ).....	29
3.2.3.2 Ordem Condicional ( C ).....	29
3.2.3.3 Com Retardo ( D ).....	30
3.2.3.4 Limitada no Tempo ( L ).....	30

3.2.3.5	Ordem Impulsional ( P ).....	31
3.2.3.6	Ordem Armazenada com Retardo ( SD ).....	32
3.2.3.7	Ordem com Retardo e Armazenada ( DS ).....	32
3.2.3.8	Ordem Condicional, Armazenada e Limitada no Tempo ( CSL ).....	33
3.2.3.9	Em Etapas Diversas.....	34
3.2.3.10	Memorizada ( S ou R ).....	35
<b>3.3</b>	<b>Elementos Integradores.....</b>	<b>36</b>
3.3.1	Elementos de Alimentação.....	36
3.3.2	Elementos de Atuação.....	36
3.3.3	Elementos de Transferência.....	37
3.3.4	Elementos de Controle.....	37
3.3.5	Controladores Programáveis.....	37
<b>3.4</b>	<b>Exemplo de Manipulação de Peças.....</b>	<b>38</b>
3.4.1	Relação de Entradas e Saídas dos Atuadores.....	39
3.4.2	Grafcet Tecnológico do Manipulador.....	40
<b>3.5</b>	<b>Conclusão.....</b>	<b>42</b>

## **CAPÍTULO 4 – Implementação e Integração Final de uma Plataforma de Manipulação**

		43
<b>4.1</b>	<b>Especificação Funcional.....</b>	<b>44</b>
<b>4.2</b>	<b>Manipulador Pneumático.....</b>	<b>47</b>
4.2.1	Montagem do Manipulador Pneumático.....	48
4.2.2	Descrição da Parte Operativa.....	49
4.2.3	Relação de Entradas e Saídas.....	50
4.2.4	Grafcet Seqüencial do Manipulador Pneumático.....	52
<b>4.3</b>	<b>Sistema de Transferência.....</b>	<b>53</b>
4.3.1	Mesa de Transferência.....	53
4.3.2	Montagem da Mesa de Transferência.....	54
4.3.3	Relação de Entradas e Saídas.....	55

4.3.4	Grafcet Funcional da Mesa Rotativa de Transferência.....	56
<b>4.4</b>	<b>Manipulador Robótico – Robix™</b> .....	57
4.4.1	Entradas e Saídas e Pc/Po do Manipulador Robótico.....	58
<b>4.5</b>	<b>Conclusão</b> .....	59
<b>CAPÍTULO 5 - Integração Final dos Postos de Trabalho da Plataforma</b>		60
<b>5.1</b>	<b>Parte Comando ( Pc ) e Parte Operativa ( Po ) dos Postos de Trabalho</b> .....	61
<b>5.2</b>	<b>Estruturação da Plataforma</b> .....	64
5.2.1	Modularização da Plataforma.....	65
5.2.2	Grafcet Tecnológico Geral da Plataforma.....	67
<b>5.3</b>	<b>Conclusão</b> .....	69
<b>CAPÍTULO 6 - Conclusões e Perspectivas para Trabalhos Futuros</b>		70
<b>6.1</b>	<b>Sugestões para Continuação deste Trabalho</b> .....	71
<b>REFERÊNCIA BIBLIOGRÁFICA</b>		72
<b>ANEXO I – Programação em Ladder para CLP1 e CLP2</b>		74
<b>ANEXO II – Programação em Linguagem C para Comunicação do CLP com Manipulador Robótico</b>		85
<b>ANEXO III – Programação das Trajetórias do Manipulador Robótico – Robix™</b>		96
<b>ANEXO IV – Sistemas Robóticos aplicados em Automação Industrial</b>		98
<b>ANEXO V – Controlador Lógico Programável ( CLP )</b>		106
<b>APÊNDICE I - Ilustração do Projeto e Implementação</b>		119

## ÍNDICE DE FIGURAS

### **CAPÍTULO 2 - Automação Industrial**

2.1	Histórico das Inovações Tecnológicas.....	08
2.2	Evolução dos Controladores de Base Tecnológica.....	11
2.3	Área de Aplicação dos Diversos Conceitos de Manufatura.....	12

### **CAPÍTULO 3 - Sistema Automatizado de Produção – SAP**

3.1	Elementos de um Grafcet.....	25
3.2	Exemplo de Ação.....	28
3.3	Ordem Memorizada.....	29
3.4	Ordem Condicional.....	29
3.5	Ordem com Atraso de Tempo.....	30
3.6	Ordem com Tempo Limitado.....	30
3.7	Ordem Impulsional e Impulsional Condicionado.....	31
3.8	Ordem SD.....	32
3.9	Ordem DS.....	32
3.10	Ordem CSL.....	33
3.11	a) Repetição de Ações em Etapas Consecutivas.....	34
3.11	b) Estrutura em Paralelo Equivalente.....	34
3.12	Ordem Memorizada usando Indicadores S e R.....	35
3.13	Manipulador Pneumático para Alimentar a Forja com Peças Provenientes de Três Fornos.....	38
3.14	Grafcet Tecnológico do Manipulador.....	40

### **CAPÍTULO 4 - Descrição e Funcionamento da Plataforma**

4.1	Postos de Trabalho Automatizado e Integrado da Plataforma.....	43
4.2	Diagrama de Blocos da Plataforma.....	45
4.3	Grafcet Funcional da Plataforma.....	46
4.4	Esquema de Montagem do Manipulador Pneumático.....	48
4.5	Diagrama de Blocos do Manipulador Pneumático.....	49

4.6	Grafcet Funcional do Manipulador Pneumático.....	52
4.7	Parte Comando e Operativa da Mesa de Transferência.....	53
4.8	Conjunto da Mesa de Transferência.....	54
4.9	Grafcet Funcional da Mesa Rotativa.....	56
4.10	Configuração Estrutural do Robix <sup>TM</sup> .....	57
4.11	Representação da Pc / Po e Entrada/Saída do Manipulador Robótico.....	58

## **CAPÍTULO 5 - Integração Final dos Postos de Trabalho da Plataforma**

5.1	Grafcet Funcional da Plataforma.....	60
5.2	Postos de Trabalho da Plataforma.....	62
5.3	Plataforma Projetada e Implementada.....	63
5.4	Grafcet Funcional das Macro-Etapas.....	66
5.5	Grafcet Tecnológico Geral de Integração da Plataforma.....	67

## **ANEXO IV – Sistemas Robóticos aplicados em Automação Industrial**

A4.1	Organização Funcional de um Robô.....	99
------	---------------------------------------	----

## **ANEXO V – Controlador Lógico Programável**

A5.1	Elementos que Compõe o CLP.....	108
A5.2	Princípio de Funcionamento de um CLP.....	109
A5.3	Ciclo de Varredura.....	110
A5.4	Componentes da Programação em Linguagem Ladder.....	115

## **APÊNDICE I - Ilustração do Projeto e Implementação**

A1	Projeto do Conjunto Esquemático da Plataforma.....	119
A2	Plataforma Implementada.....	119
A3	Projeto do Manipulador Pneumático.....	120
A4	Manipulador Pneumático Implementado.....	121
A5	Projeto da Mesa Rotativa de Transferência.....	122
A6	Mesa Rotativa de Transferência Implementada.....	123
A7	Configuração Estrutural do Robix <sup>TM</sup> .....	124
A8	Configuração Estrutural do Robix <sup>TM</sup> Implementada.....	124

## ÍNDICE DE TABELAS

### **CAPÍTULO 3 - Sistema Automatizado de Produção - SAP**

3.1	Simbologia de Etapas e Ações.....	27
3.2	Relação de Entradas do Manipulador.....	39
3.3	Relação de Saídas do Manipulador.....	39

### **CAPÍTULO 4 – Implementação e Integração Final de uma Plataforma de Manipulação**

4.1a	Relação de Entradas do Manipulador Pneumático.....	50
4.1b	Relação de Saídas do Manipulador Pneumático.....	51
4.2a	Relação de Entradas da Mesa Rotativa de Transferência.....	55
4.2b	Relação de Saídas da Mesa Rotativa de Transferência.....	55

### **CAPÍTULO 5 - Integração Final dos Postos de Trabalho da Plataforma**

5.1	Postos de Trabalho e as Pc / Po.....	61
5.2	Relação das Entradas.....	64
5.3	Relação das Saídas.....	64

### **ANEXO V – Controlador Lógico Programável ( CLP )**

A5.1	Dispositivos de Entrada/Saída Discretos.....	112
------	--	-----

## NOMENCLATURA

ABNT .....	Associação Brasileira de Normas Técnicas
AFRI.....	Association Française de Robotique Industrielle
CA.....	Corrente Alternada
CAD.....	Computer Aided Design
CC.....	Corrente Contínua
CIM.....	Computer Integrated Manufacturing
CLP.....	Controlador Lógico Programável
CPU.....	Central Processing Unit
EPROM.....	Erasable and Programmable Read Only Memory
GRAF CET....	Gráfico Funcional de Comandos Etapas - Transição
IEC.....	International Electrotechnical Committee
JIT.....	Just in Time
MES.....	Manufacturing Execution System
MRP.....	Manufacturing Resource Planning
NA.....	Normalmente Aberto
NEMA.....	National Electric Manufacturing Association
NF.....	Normalmente Fechado
PID.....	Preditivo Integral Derivativo
PLC.....	Programmable Logic Controller
RAM.....	Random Access Memory
RIA.....	Robot Institute of America
SAP.....	Sistema Automatizado de Produção
SCADA.....	Supervisory Control and Data Acquisition
SFC.....	Sequential Function Chart
TQC.....	Total Quality Control

## **CAPÍTULO 1**

### **Introdução**

Este capítulo faz uma breve consideração sobre o cenário que as indústrias estão situadas diante da competitividade atual e apresenta também, os objetivos e o delineamento deste trabalho.

Nos últimos anos houve uma liberação para o livre comércio entre organizações ( globalização ). Esta nova realidade forçou a modernização dos parques industriais em todos os continentes, tornando-os mais competitivos, principalmente nos países emergentes.

A automação industrial é o principal meio de se atingir esse objetivo, porém, para que um sistema industrial seja competitivo, é necessário um planejamento criterioso. A utilização de uma tecnologia apropriada, integrando as diversas atividades internas e externas de uma indústria, visando produtividade, qualidade e flexibilidade a fim de surpreender e corresponder as expectativas do cliente é fundamental.

No Brasil, esta realidade não é diferente. Durante um longo período ( até final dos anos 80 ) havia reserva de mercado, onde muitos produtos nacionais eram protegidos contra os produtos estrangeiros, fazendo com que o consumidor não tivesse opções de escolha. Desta forma, as indústrias nacionais não investiram o suficiente em tecnologia para se modernizarem.

A complexidade dos Sistemas Automatizados de Produção implica em grande dificuldade na definição clara das especificações associada a estes sistemas. Por ser uma área multidisciplinar, as variedades de temas fundamentais exige do projetista a escolha de ferramentas adequadas para estruturar os sistemas de maneira que os mesmos possam ser integrados, flexíveis e expansíveis.

Dentro desta realidade, este trabalho visa mostrar conceitos de integração de sistemas automatizados através da implementação de uma plataforma de manipulação.

### **1.1 – Objetivos do Trabalho**

O objetivo deste trabalho é apresentar um projeto e sua implementação através de uma plataforma integrada e automatizada com manipuladores, cujas etapas a serem atingidas são:

1 – apresentar ferramentas para descrição e implementação de SAPs, levando-se em consideração as exigências das normas existentes;

2 – apresentar uma referência bibliográfica para utilização acadêmica e industrial que sustentem os conceitos utilizados em automação industrial;

3– apresentar exemplos práticos utilizando os conceitos de automação e integração de sistemas industriais automatizados.

## 1.2 – Delineamento do Trabalho

A dissertação apresentada está subdividida em capítulos, anexos e apêndice delineados da seguinte forma:

- Capítulo 1 – Introdução: é feita uma breve consideração sobre o cenário industrial, referente a automação, uma apresentação do trabalho, incluindo os objetivos e o delineamento do mesmo.
- Capítulo 2 – Automação Industrial: é feita uma consideração geral sobre automação industrial, justificando sua importância, considerando os aspectos econômicos e sociais.
- Capítulo 3 – Sistemas Automatizados de Produção: é apresentada a norma internacional para Diagramas Funcionais e Seqüenciais e sua importância como ferramenta de integração.
- Capítulo 4 – Descrição e Funcionamento da Plataforma: são apresentadas as descrições de funcionamento dos postos de trabalho da plataforma e a referência dos mesmos dentro dos Sistemas Automatizados de Produção.
- Capítulo 5 – Integração Final dos Postos de Trabalho da Plataforma: é apresentada a integração final dos postos de trabalho, utilizando o grafcet como ferramenta de integração.
- Capítulo 6 – Conclusões e Perspectivas para trabalhos futuros: são apresentadas as considerações finais e as sugestões para trabalhos futuros.
- Anexo I – Programação em Ladder para CLP<sub>1</sub> e CLP<sub>2</sub>.
- Anexo II - Programação em Linguagem C para comunicação do CLP com Manipulador Robótico.
- Anexo III – Programação das trajetórias do Manipulador Robótico – Robix<sup>TM</sup>.
- Anexo IV – Sistemas Robóticos aplicados na Automação Industrial.
- Anexo V – Controlador Lógico Programável.
- Apêndice 1 – Ilustração do Projeto e Implementação.

## **CAPÍTULO 2**

### **Automação Industrial**

O trabalho desde as mais antigas civilizações é fruto da conjugação de três fatores básicos, ou seja, de mão de obra, equipamento ( ferramentas, dispositivos, máquinas, etc. ) e processo utilizado para realizar uma determinada tarefa.

Sempre foi uma preocupação e uma busca constante para aprimorar os recursos e os processos do trabalho, visando melhorar a qualidade de vida, na redução de esforços, jornada de trabalho e redução de custo.

A evolução tecnológica tem sido marcante na mudança de comportamento da humanidade, cuja primeira revolução de maior impacto foi a industrial ocorrida por volta de 1760, na Inglaterra. O uso de teares mecânicos, o aperfeiçoamento da máquina a vapor e da siderurgia deram início a uma sucessão de inovações tecnológicas, permitindo os sistemas automatizados de produção ( SAP ) aplicados aos processos industriais atuais.

Até início dos anos 70 a produção industrial se caracterizava pela produção em massa, oferecendo ao mercado consumidor um número limitado de produto, cuja demanda superava a capacidade de produção das empresas. Já no final desta década, em função do grande avanço tecnológico baseado na microeletrônica foi possível, aprimorar os meios produtivos e diversificação de produtos e pela primeira vez na história da produção industrial, a demanda foi superada pela oferta de produto. ( Agostinho 95 )

Nas últimas décadas o setor produtivo vem sofrendo profundas modificações, por conta de uma competitividade globalizada, o que força a modernização dos parques industriais em todos os segmentos produtivos.

Nesse contexto, a automação desempenha um papel fundamental nas organizações industriais, para que as mesmas possam atingir uma alta produtividade, qualidade, baixo custo, confiabilidade e flexibilidade.

Com as mudanças rápidas nas plantas industriais, tem surgido o conceito de automação integrada, tais como: Processo de engenharia assistida por computador, robótica e engenharia de novos materiais. Essas mudanças, alteraram profundamente a relação do trabalhador e sua qualificação com as empresas, alterando o perfil profissional dos mesmos.

Os fatores decisivos que levaram a uma profunda reestruturação no setor produtivo industrial foram:

- Inovações tecnológicas na microeletrônica
- Necessidade do mercado e tempo de resposta
- Livre comércio entre organizações internacionais – Globalização
- Aumento da competitividade
- Expectativa do cliente para novos produtos
- Qualidade assegurada e direito do consumidor

A utilização de novas técnicas de gestão industrial como: TQC, JIT, MRP, MES e as inovações tecnológicas foram fundamentais na eficácia dos processos de manufaturas, tornado-os mais racionais e otimizados. A implantação de células flexíveis na produção, tornou-se uma necessidade das empresas, para atender a demanda de mercado, com mais rapidez. ( Silveira 98 )

## 2.1 - Vantagens da Automação

Pode-se destacar como as principais vantagens proporcionadas pela automação :

- Redução no custos do produto, reduzindo mão de obra, economia de energia, economia de material e etc.
- Substituição do trabalho humano em atividades repetitivas e perigosas e melhoria das condições de trabalho.
- Aumento da produtividade, melhora da qualidade, confiabilidade e uniformidade limitada a atividade humana.
- Facilita o controle de operações manualmente ou intelectualmente, como operações rápidas e coordenações complexas.

## 2.2 - Definição e Histórico da Automação

A forma de produção, após a segunda guerra mundial, alterou-se muito. Contribuiu para essa mudança vários fatores, como avanço tecnológicos na área da microeletrônica, busca da qualidade, conforto do cliente, preços competitivos, etc.

Quanto a definição de automação, ainda causa uma certa ambiguidade na literatura técnica.

Alguns autores com uma visão mais holística ( Oxford English Dictionary, por exemplo) definem automação como sendo: “O controle automático da fabricação de produtos através de uma sequência de estágios sucessivos”. Outros autores consideram automação essencialmente uma mecanização. Dentro de uma visão econômica, prevalece a seguinte definição: “É um método tecnológico que busca reduzir custos de produção em termos de homem/hora por unidade de produto”. ( Silveira-98 ).

Já para os autores mais técnicos (Agostinho-95), automação tem conceito bem distinto de mecanização, ou seja, automação esta voltada em substituir ou auxiliar no esforço mental do trabalhador/ operador, enquanto mecanização apenas substitui os esforços físicos, como aplicação de motores em máquinas ferramentas e de processo.

A partir da revolução industrial (meados do séc. XVIII ), iniciou-se a automação de transformação, com a utilização de máquinas hidráulicas e posteriormente os motores de combustão interna e elétricos.

No final do século XIX, em resposta a necessidade de maior rapidez e confiabilidade nos processos produtivos, deram início a automação dos cinematismo das máquinas. Nesta época prevalecia os modelos “Taylorista” e “Fordista” nas linhas de produção.

Após a segunda guerra mundial ( meados dos anos 50 ) surgia as primeiras iniciativas para automação de controle, como forma de atender uma necessidade cada vez mais flexível dos processos produtivos industriais.

Nas últimas décadas, com a competitividade crescente entre empresas, deu-se início à automação das células flexíveis de manufatura. Com isso houve uma grande transformação não apenas na concepção das máquinas, mas também nos procedimentos de operação e utilização dos recursos de materiais.

### **2.3 - Inovação Tecnológica**

As inovações produzidas pela humanidade, sempre foram motivos de apreensão, quanto a seus impactos tecnológicos e sociais, das mudanças provocadas no estilo decorrente de novos hábitos assumidos, desafios e necessidades.

O homem sempre buscou simplificar seu trabalho de forma a substituir o seu esforço físico por meio de mecanismo, com objetivo de aumentar sua riqueza através do

aumento da produtividade, maior tempo livre para uso de seu intelecto na forma de pesquisa e arte, melhorar sua qualidade de vida com maior tempo para lazer e conforto.

Desde a primeira utilização da roda, ocorrida na Ásia por volta de 3500 a 3200 a. C. o homem vem procurando controlar e otimizar o tempo. Isso significa ganhar tempo, ou seja, tornar os processos produtivos industriais sobretudo mais rápidos, embora algumas inovações foram utilizadas para fins bélicos como meio de destruição do próprio homem.

Com o compromisso pautado na esperança da qualidade de vida, as inovações têm trazido consequências sociais desastrosas, como aumento do desemprego, mesmo que seja temporário. Porém, a marca mais contundente das inovações tecnológicas tem sido a concentração de renda em prejuízo da maioria da população, algumas das inovações mais significativas ao longo da história são apresentadas na figura 2.1.

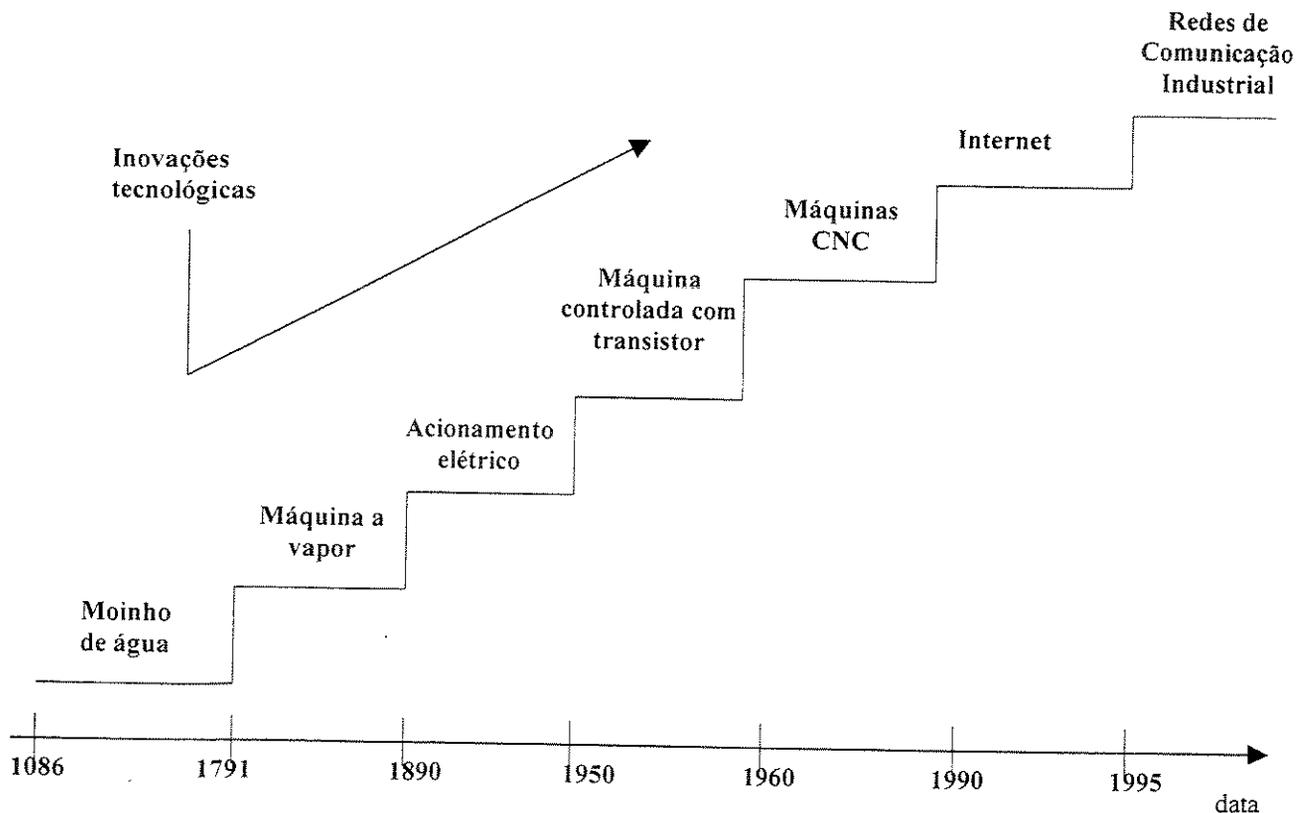


Fig. 2.1 – Histórico das Inovações Tecnológicas

## **2.4 - Evolução do Controle Industrial**

Os primeiros controles em acionamentos mecânicos, empregados na indústria, eram acionados através de dispositivos pneumáticos e/ou hidráulicos por volta de 1788, e logo após a segunda guerra mundial o controle passou a ser eletrônico com tecnologia analógica.

Com o decorrer da evolução tecnológica, foram desenvolvidas máquinas-ferramentas com controle automático simples para executarem uma sequência de operação. Em 1950, tais máquinas eram comandadas por circuitos transistorizados e em 1960, os controles passaram ser executados por microprocessadores, seguindo instruções codificadas e transmitidas por fitas perfuradas ou fitas magnéticas.

O desenvolvimento de todos os equipamentos dos quais a civilização moderna tem conhecimento, não aconteceria caso as máquinas-ferramentas não fossem empregadas nas várias indústrias, nos diferentes ramos de atividades que se conhece atualmente.

### **2.4.1 - Controle Numérico**

A máquina-ferramenta foi aprimorada até a concepção do comando numérico (C.N.), e tal concepção viabilizou a fabricação de peças com geometria complexa por meio de programação eletrônica, de pequenos lotes de peças.

A primeira máquina de controle numérico foi uma fresadora com três eixos, desenvolvida pelo Instituto de Tecnologia de Massachusetts (MIT), constituindo-se em um sistema em que as ações eram controladas pela aplicação direta de dados numéricos em alguns pontos específicos, como rotação e avanço. Na indústria moderna as máquinas-ferramentas automáticas, estão sempre presentes um sistema autômato no processo de produção. No Brasil, a primeira máquina de controle numérico foi um torno mecânico, fabricado pela indústria ROMI, em 1971. ( Vendrometo 94 ).

## **2.4.2 – Controle de Processos Industriais**

Os processos industriais atuais, podem ser divididos em dois tipos conforme a natureza das variáveis a serem controladas. Quando se trata de variáveis do tipo analógica ou de tempo contínuo, tem-se um processo do tipo contínuo. Caso a variável seja discreta ou digital, tem-se um processo do tipo discreto.

## **2.4.3 - Definição de Processo Industrial**

“ Processo industrial constitui-se na aplicação do trabalho e do capital para transformar a matéria prima em bens de produção e consumo, por meio de técnicas de controle , agregando valor ao produto para atingir o objetivo pelo qual o mesmo foi idealizado” . ( Silveira 98 ).

As indústrias manufatureiras são as que se caracterizam pelo controle do processo do tipo discreto, com fabricação em lotes, onde o maior exemplo ocorre na industria automobilística, já o controle de processo do tipo contínuo, e característico das indústrias químicas, petroquímicas e farmacêuticas.

Até o início da década de 60, a utilização de relés eletromecânicos era praticamente a única opção viável. Quando surgiram os dispositivos microprocessados, vieram também os controladores lógicos programáveis ( CLP ), onde a forma de programação segue a lógica dos diagramas elétricos dos relés ( mais detalhes no capítulo 3 ). Adequados para as situações industriais, os controladores realizavam uma rotina cíclica de operação, apenas com variáveis digitais. Quando tais equipamentos manipulavam variáveis, denominam-se controladores programáveis ( CP ).

Para controlar os processos industriais, surgiram os controladores de processos contínuos de malha única, conhecidos como malha aberta, muito utilizado para controle de

temperatura de ambiente ou de processo, pressão em líquidos e vazão em gases. A evolução desse controle deu origem aos sistemas “multi-look”. Além de implementar e executar todos os tipos de controles, tecnicamente possíveis de forma simultânea controlam inúmeros pontos do processo por meio de controladores de grande porte.

A figura 2.2 , mostra a evolução dos controladores discretos e analógicos, quanto ao tipo de equipamento utilizado.

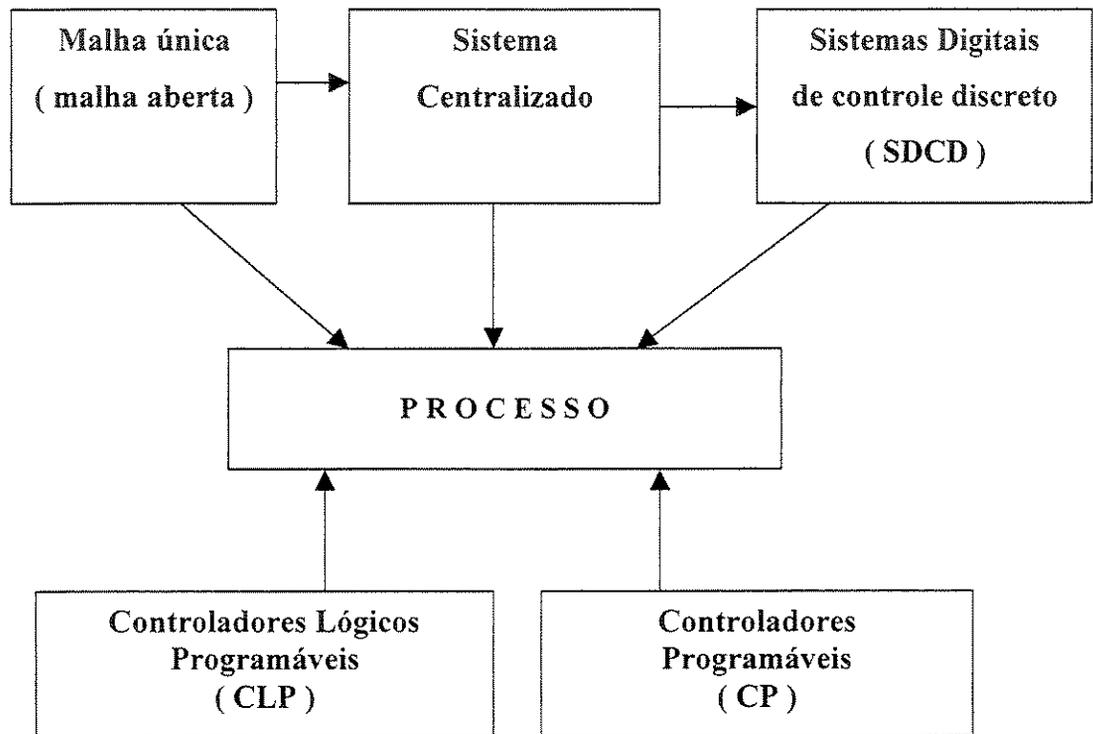


Figura 2.2 Evolução dos Controladores de Base Tecnológica

## 2.5 – Classificação de Sistemas Automatizados

A forma de produção nas últimas décadas alterou de forma substancial. Essa mudança deve-se a vários fatores, como avanço tecnológico da microeletrônica que permitiu o surgimento dos microprocessadores, aumento da competitividade entre as organizações, comportamento dos consumidores exigindo mais qualidade, conforto e menores preços. A figura 2.3 mostra de forma quantitativa, as várias formas de aplicação e os diversos conceitos de fabricação.

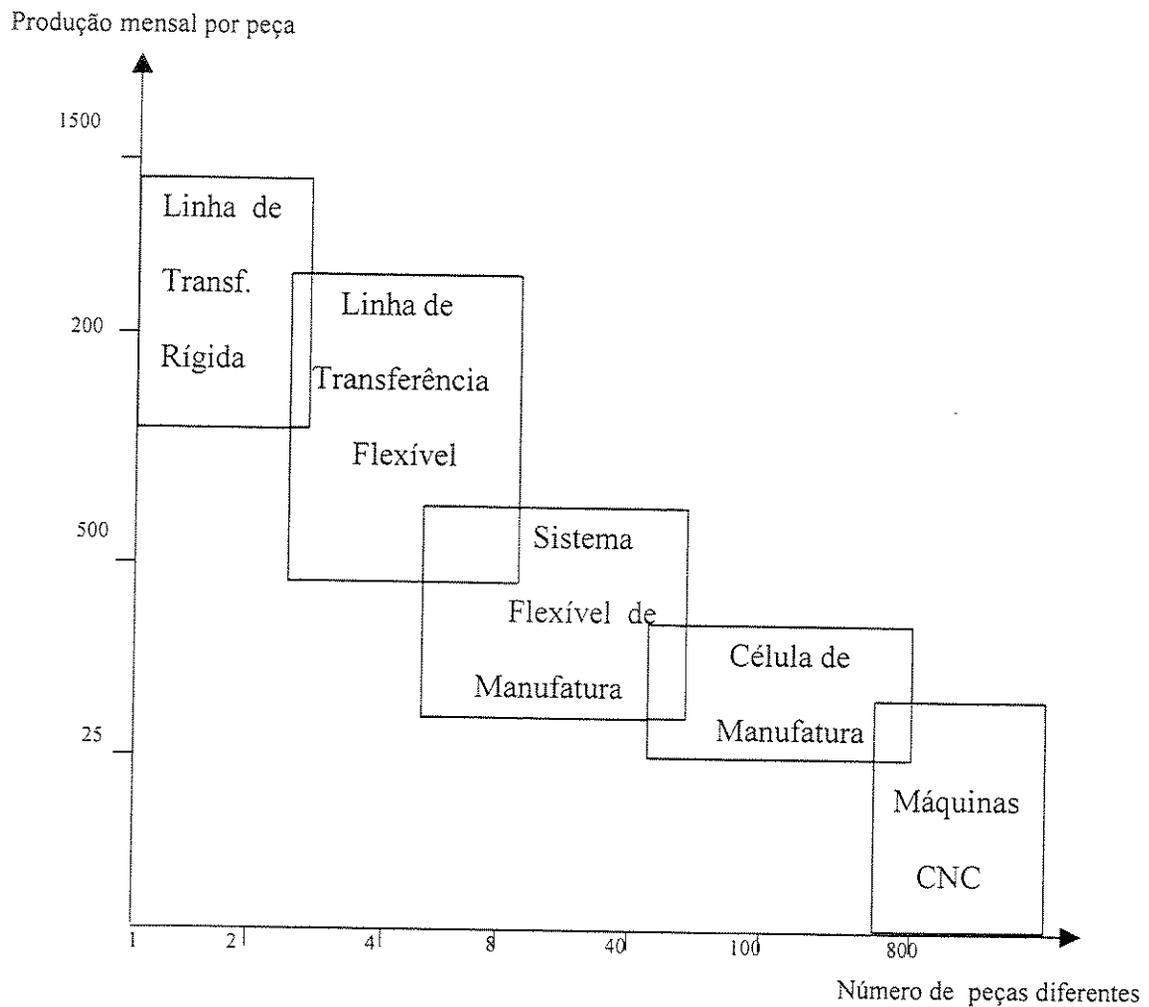


Fig. 2.3 – Área de Aplicação dos Diversos Conceitos de Manufatura ( Agostinho 95 )

Diante desse novo contexto a automação industrial divide-se em três formas:

- Automação rígida
- Automação flexível
- Automação programável

**Automação Rígida** - é utilizada quando o volume de produção ( quantidade de peças) é grande, voltada para um único tipo de produto. As características principais da automação são:

- alto custo de investimento inicial.
- as operações básicas são normalmente simples.
- qualquer alteração em decorrência de mudança do produto tem custo elevado.

**Automação Flexível** - utiliza-se dos recursos mecânicos, microeletrônicos e da informática. É essencialmente destinada a volume de produção médio, para situações onde requer variedades de produtos, sem que haja troca de equipamento.

**Automação Programável** - tem semelhança com automação flexível, porém as diferenças são notadas no volume de produção ( lotes de produção menores ). Na automação programável, a reprogramação do equipamento é muito rápida, reduzindo o tempo para troca de produto e ferramentas ( set up ) utilizando programas computacionais.

### **2.5.1 - Sistemas Flexíveis de Manufatura - SFM**

A denominação SFM é frequentemente utilizada para descrição genérica de vários sistemas similares, embora existam outros termos para descrever estes equipamentos como:

- Sistema de manufatura integrado por computador – SMIC
- Manufatura de peças gerenciada por computador – MPGC
- Complexo flexível de manufatura – CFM
- Sistemas automatizados e integrados por computador – SAIC

A característica única e peculiar dos SFM é de ligar a um só conjunto, de maneira integrada, postos de trabalho, sistema automatizado de manuseio de materiais e o controle computacional.

Um SFM tem como objetivo produzir, de maneira eficiente, diferentes tipos de peças em volumes médios e pequenos. Todas as atividades no sistema de preparação de matéria prima, monitoramento de desgaste de ferramentas, movimentação de peças entre máquinas, cargas e descargas, inspeção, planejamento e programação, são feitos por computador em tempo real.

### **2.5.2 - Sistemas Integrados de Manufatura**

O controlador lógico programável (CLP), é um importante equipamento utilizado pelas indústrias, e devido aos recursos que lhe são adicionados, deverá continuar sendo largamente aplicado como equipamento de controle discreto e facilitador na integração dos sistemas de manufatura. Existe uma forte tendência em definir padrões entre os fabricantes de CLP, facilitando o técnico em automação na utilização de uma tecnologia aberta.

Uma característica marcante dos sistemas integrados de manufatura é a grande quantidade de informações a serem manipuladas, tanto na fabricação, como no produto e na gestão. Ao se estabelecer a integração das informações por um controle discreto, através de uma rede de computadores, entre diversos níveis hierárquicos na produção industrial, reduz de maneira substancial os custos de fabricação, aumenta a qualidade, produtividade e a confiabilidade do sistema.

As decisões essenciais dentro de um sistema organizacional de produção, são tomadas tendo como referência às informações vindas do chão de fábrica, com dados precisos e confiáveis. Informações confiáveis no processo de fabricação permitem uma integração eficaz, assim como um planejamento mais adequado quanto à capacidade de produção, regulando os níveis de estoques, definindo e projetando melhor o produto, estabelecendo estratégias e procedimentos de gerenciamento e diagnósticos.

## 2.6 - O Impacto da Automação na Sociedade

A grande questão desde o final do século XX é como equilibrar a relação entre a automação industrial e o desemprego. Percebe-se ao fazer uma contextualização histórica, que a preocupação é bastante antiga e que sempre preocupou a humanidade ou pelo menos a grande maioria.

Quando HARGREAVES em 1767 construiu uma máquina de fiação para indústria têxtil, onde substituiu o trabalho equivalente a 100 homens, e posteriormente ARKWRIGHT idealizou e construiu o tear mecânico (movido à força da água em queda livre), calculava-se haver na Inglaterra, em torno de 7900 pessoas empregadas na produção de tecidos de algodão. Após 25 anos, o número de pessoas trabalhando com fiação e tecelagem era da ordem de 320.000, um aumento de quase 4000%. O que se verificou neste caso foi um aumento da produção em série com a mecanização, que revolucionou o paradigma da produção da época.

No final do século XIX, entre 1870 e 1880, a marinha mercante inglesa aumentou seu movimento em torno de 22.000 toneladas, devido à introdução de máquinas de içar e elevadores de grãos a vapor. Verificou-se que o número de trabalhadores reduziu neste período, porém outros empregos foram criados na fabricação dessas máquinas e equipamentos.

Por volta de 1940, a preocupação de novos empregos era tanta que nas cidades de Houston e Texas (EUA), o sindicato dos mestres encanadores, decidiu que os canos pré-fabricados para instalação não deveria ter roscas nas extremidades para que, as mesmas fossem feitas no local da instalação, utilizando assim mais mão de obra. Essa preocupação estendeu-se a outras categorias.

Em 1961, líderes sindicais e funcionários do governo dos EUA, falavam com convicção que a automação industrial era a principal responsável pelo desemprego no país.

Embora com automação cria-se novos empregos, uma vez que as máquinas e os equipamentos para automação devem ser projetados e construídos, verifica-se que o deslocamento da mão de obra não se equivalem, gerando, portanto, desemprego.

Constata-se que devido à automação e as novas técnicas de gestão, como JIT, TQC, MRP entre outras, alguns postos de trabalho foram eliminados, que foi o caso do datilógrafo, torneiro mecânico, linotipista, etc, e outras profissões sofreram transformações como agrônomo, médico, engenheiro, costureiro entre outras.

No Brasil a automação constitui-se um segmento emergente e juntamente com a reestruturação das empresas aqui instaladas provocaram uma grande preocupação com o desemprego, já que o ministério do trabalho, constatou que entre 1990 e 1997, houve uma redução de 2 milhões de empregos formais, e que as atividades relacionadas com a automação contribuíram de forma substancial.

A questão do desemprego deve-se muito ao contexto socioeconômico global, como as crises internacionais, falta de crescimento econômico e a competitividade entre as organizações, contribuem para que o desemprego seja expressivo em todo o mundo.

## **2.7 – Conclusão**

Toda tecnologia aplicada na Automação Industrial tem como objetivo principal à redução de custo, a fim de tornar a indústria mais competitiva. Com a necessidade do cliente ser atendido com mais eficiência, surgiram os critérios de classificação dos sistemas industriais automatizados, em função do tamanho dos lotes a serem produzidos. Os impactos sociais sempre foram sentidos com as inovações tecnológicas, porém, os aspectos econômicos sempre prevaleceram.

## CAPÍTULO 3

### Sistema Automatizado de Produção – S A P

#### 3.1 – Introdução

Um sistema automatizado de produção envolve várias áreas do conhecimento humano, as quais são divididas em dois grandes grupos : Grupo técnico e Grupo humano.

As áreas que compõem o **grupo técnico** são essencialmente : Mecânica .Eletrônica e a Informática. As áreas envolvidas no **grupo humano** são quase sempre circunstanciais, alternando sua importância , de acordo com o momento político e econômico de cada país, tais como: Economia, administração, ciências sociais e psicologia. Os segmentos e/ou componentes das áreas que compõem o grupo técnico e que estão presentes principalmente nas organizações competitivas, formam toda a base que sustenta a automação industrial.

#### **Grupo Técnico:**

- CLP – Controladores Lógicos Programáveis.
- CIM – Manufatura Integrada por Computador.
- FMS – Sistemas Flexíveis de Manufatura.
- CNC – Comando Numérico Computadorizado.
- CAE – Engenharia assistida por Computador.
- CAD – Desenho Auxiliado por Computador.
- CAM – Manufatura Auxiliada por Computador.
- Robôs e Manipuladores industriais.
- Atuadores
- Sensores
- Redes de Comunicação.

### **Grupo Humano:**

- Sistema educacional - formação básica e tecnológica
- Política econômica voltada para o crescimento.
- Gestão organizacional – qualidade, “just in time”, tecnologia de grupo, MRP, ERP, MES, etc.
- Treinamento motivacional e psicologia do trabalho
- Doenças profissionais – ergonomia.
- Sindicatos de classes e centrais sindicais
- Desemprego e aspectos sociais.

Os elementos pertencentes aos grupos citados acima possuem variáveis e características independentes, relacionadas e integradas dentro de sistemas automatizados de produção. A integração de todos esses elementos constitui algo novo dentro e fora das organizações modernas.

A automação tem complexidade distinta nas empresas em função do processo a ser automatizado, porém os objetivos são semelhantes, visando sempre aumentar a competitividade (mencionado no capítulo 1), através da produtividade e/ou qualidade, confiabilidade, flexibilidade e menor tempo de resposta para o cliente.

Os sistemas automatizados e integrados de produção estão em pleno processo de evolução e transição em relação principalmente à administração científica proposta por Taylor/Ford no início do século passado. O componente multidisciplinar que caracteriza os processos industriais automatizados impõe ao sistema uma complexidade até então nunca vista nos meios produtivos. Tudo isso na prática significa que, descrever um sistema automatizado de forma clara, segura, sem ambiguidade e universal, passa a ser a fase mais importante de todo processo de automação.

Visando padronizar uma linguagem na descrição dos sistemas automatizados a norma **IEC 1131-3**, através de seu capítulo 351, padroniza um vocabulário internacional para controles automáticos, dividindo um sistema de controle em duas partes distintas, são elas:

- **Parte Operativa ( Po ) ou Sistema Controlado** – é o sistema que realiza as operações físicas e;
- **Parte Comando ( Pc )** – caracterizado por receber as informações vindas do operador e/ou do processo a ser controlado e emitir informações ao sistema controlado ( Po ).

A primeira etapa do desenvolvimento de um SAP consiste em descrevê-lo de modo a não ficar nenhuma dúvida, sobre os objetivos a serem atingidos no projeto proposto, onde deve prevalecer o conjunto, sem a preocupação com detalhes tecnológicos, quando então se deve descrever os elementos específicos do sistema de automação. É nesta etapa que surgem as maiores dificuldades, porque as informações devem chegar ao projetista, com todos os detalhes necessários.

A comunicação verbal não é a forma mais indicada, isso porque pode levar a mais de uma interpretação e até mesmo informações ambíguas. Para sistemas complexos, com ações simultâneas e decisões com múltiplas possibilidades, deve-se evitar a utilização de textos. Sempre que possível, e necessário, às descrições de sistemas automatizados deverão ser representadas de forma gráfica, que são mais fáceis de serem interpretadas e executadas. Porém encontrar uma forma que seja aceita e entendidas por todos, torna-se muito difícil. Para unificar e padronizar os símbolos gráficos e a sequência do sistema, a norma **IEC 1131-3** padronizou 5 linguagens de programação (para CLP), definindo os pontos de intertravamento de sistemas sequenciais.

A norma IEC 1131-3, também incorpora 5 grandes grupos como objetivos gerais, são eles:

- Visão geral
- Equipamentos
- Linguagens de programação
- Manuais ou guias do usuário
- Comunicação

Sabe-se que para sistemas complexos de controles em automação, a utilização de controladores lógicos programáveis é fundamental. As dificuldades surgem na medida que aumenta o número de fabricantes de CLP, com linguagens diferentes e próprias para o mesmo processo, sem critérios de padronização. A norma IEC-1131-3 visa proporcionar em ambientes integrados, principalmente uma maior rapidez de programação e minimizando os erros no desenvolvimento dos programas. As 5 linguagens de programação foram divididas em duas partes; uma gráfica e outra textual da seguinte forma:

- **Linguagens Gráficas**

- Diagramas de funções sequenciais (SFC – Sequential Function Chart ) também conhecidos como Grafcet.
- Diagrama em ladder ( LD – Ladder Diagram ) ou diagramas de contatos.
- Diagramas de blocos de funções ( FBD – Function Block Diagram ) ou diagramas de blocos.

- **Linguagens Textuais**

- Listas de instruções ( IL – Instruction List ) ou mnemônicos Booleanos , baseados em comandos – load , store , move , add, são utilizados em aplicações envolvendo elementos tais como: sensores e atuadores inteligentes é considerada uma linguagem de baixo nível.
- Textos estruturados – ( ST – Structured Text ) ou parâmetro idiomáticos. É uma linguagem de alto nível similar ao pascal, utilizada em sistemas complexos, onde fica difícil interpretar uma linguagem gráfica de algoritmos de otimização de processos e inteligência artificial.

### 3.2 - GRAFCET

O Grafcet foi desenvolvido na França por volta 1975, através de um grupo de pesquisadores e gerentes industriais, envolvidos com sistemas discretos de grande complexidade. Após ser testado em sistemas educacionais e em empresas privadas francesas, mostrou-se ser muito interessante e eficaz para representação de sistemas sequenciais. Em 1988, ele foi incorporado a norma IEC 60848 (preparação e diagrama funcional para sistema de controle), sob a sigla **SFC** conforme publicação 848, e regularizado pela norma Francesa NF C03 190. Posteriormente, os fabricantes de CLP e produtores de software escolheram o Grafcet como linguagem de entrada para controle sequencial booleano e propuseram implementações em computadores e controladores. Seu uso industrial vem se ampliando, assim como o interesse em pesquisá-lo quanto ao valor teórico desse modelo. O nome Grafcet derivou-se de Graf, devido à representação gráfica e, AFCET (Association Française de Cybernétique Economique et Technique), que deu suporte ao trabalho de definição. Em 1982, o Grafcet foi incorporado como norma pela AFNOR (Association Française de Normalization).

Algumas técnicas utilizadas atualmente para descrever comportamento sequenciais em sistemas automatizados incluem : fluxogramas , diagramas de variáveis de estado, rede de petri, diagrama trajeto-passo e grafcet.

O Grafcet é um modelo de representação gráfica do comportamento de comando de um sistema automatizado. Ele é constituído por simbologias em arcos orientados que interligam as etapas e transições, interpretadas por variáveis de entrada e saída da parte comando, identificadas como receptividade de ações e por sequências de evolução que caracterizam o comportamento dinâmico dos elementos comandados.

A figura 3.1 representa os elementos do Grafcet. Uma progressão clara do significado de cada elemento do Grafcet é fundamental para que se implemente o controle sequencial em CLP.

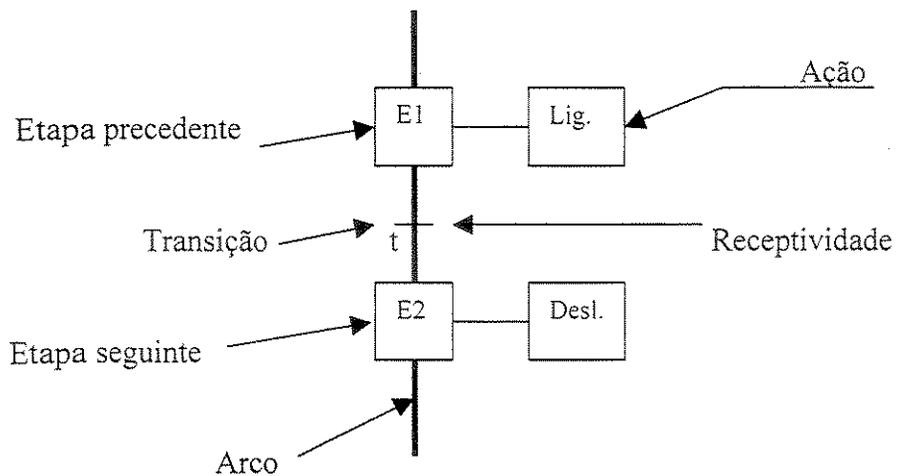


Figura 3.1 Elementos de um Grafcet

### **3.2.1 - Etapas**

Uma etapa significa um estado onde o comportamento do conjunto do comando não se altera em relação às entradas e saídas. A precisão do diagrama é diretamente proporcional ao número de etapas utilizadas para descrever o sistema. As etapas são representadas por retângulos, e devem ser identificadas por números no seu interior.

A etapa inicial deve ser necessariamente ativada no início do controle de um sistema. As etapas iniciais serão tantas quantas forem necessárias, sendo todas ativadas simultaneamente. A etapa inicial é diferenciada através de um duplo quadrilátero representada na tabela 3.1, que mostra as etapas relacionadas com as ações.

### **3.2.2 - Transições**

Uma transição indica a possibilidade de evolução entre etapas. Em um determinado instante uma transição pode ser válida ou não. Uma transição é representada por um traço curto e grosso ( figura 3.1 ) situado sobre cada uma das ligações entre etapas. A cada transição associa-se uma condição lógica chamada receptividade. Esta condição lógica permite distinguir, dentre todas as informações disponíveis num dado instante, apenas aquelas que permitem a evolução da parte comando, tornando a transição válida.

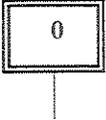
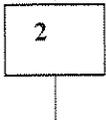
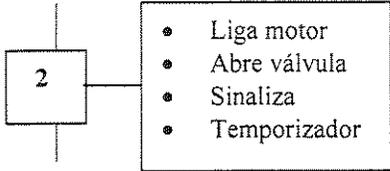
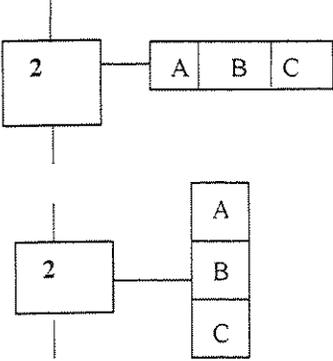
Simbologia	Denominação
	Etapa inicial
	Etapa
	Ação associada à Etapa
	Várias Ações associadas à mesma Etapa

Tabela 3.1 Simbologia de Etapas e Ações

### 3.2.3 Ação

As ações representam os efeitos que devem ser obtidos sobre a operação controlada em uma determinada situação (tabela 3.1). Em geral, as ordens de comando contidas em ações podem atuar sobre elementos físicos da operação controlada (saídas de CLP), sobre elementos auxiliares de comando (temporizadores, contadores, memórias, etc.) ou ainda em interface homem – máquina (vídeo, painéis de controle, impressora, etc.).

Cada ação é representada graficamente no interior de retângulos associados a uma etapa, e isto será realizado somente e apenas quando a etapa correspondente estiver ativa (Fig. 3.2).

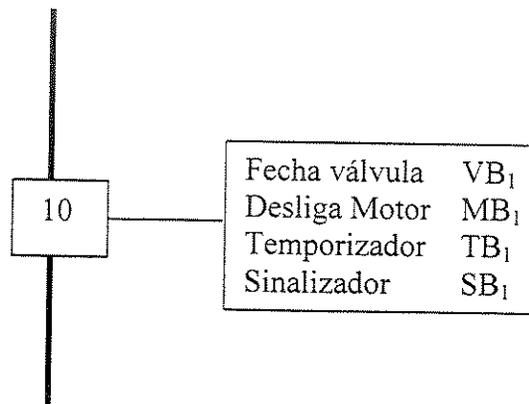


Fig. 3.2 – Exemplo de Ação

Uma ação pode emitir ordens de comando da seguinte forma :

- Memorizada – S ( stored )
- Condicional – C ( conditional )
- Com retardo – D ( delayed )
- Limitada no tempo - L ( time limited )
- Impulsional – P ( pulse )
- Armazenada com retardo ( SD )
- Com retardo e armazenada ( DS )
- Condicional, armazenada e limitada no tempo ( CSL )

### 3.2.3.1 – Ordem Memorizada ( S )

Tipo de ordem em que a ação é mantida após a desativação da etapa até ser finalizada por outra etapa, conforme indicado na figura 3.3.

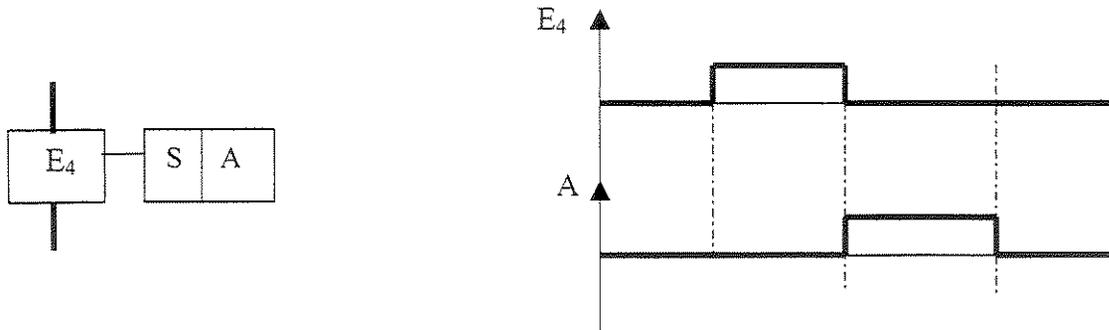


Fig. 3.3 – Ordem Memorizada

### 3.2.3.2. – Ordem Condicional ( C )

Neste caso a ação é indicada e mantida enquanto a etapa estiver ativa, desde que a condição lógica especificada seja verdadeira (satisfeita) e pode ser indicada interna ou externa ao símbolo (Fig. 3.4).

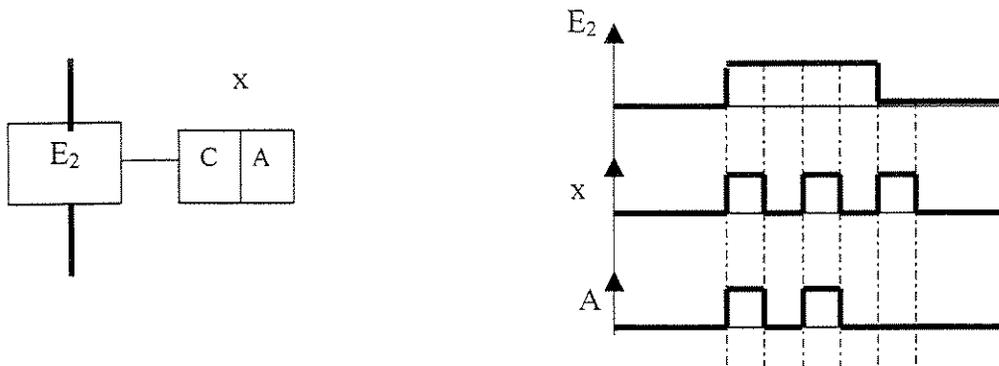


Fig. 3.4 – Ordem Condicional ( C )

### 3.2.3.3. – Com Retardo ( D )

Refere-se ao caso particular de ordem condicional em que a dependência é relacionada a um retardo de tempo conforme figura 3.5.

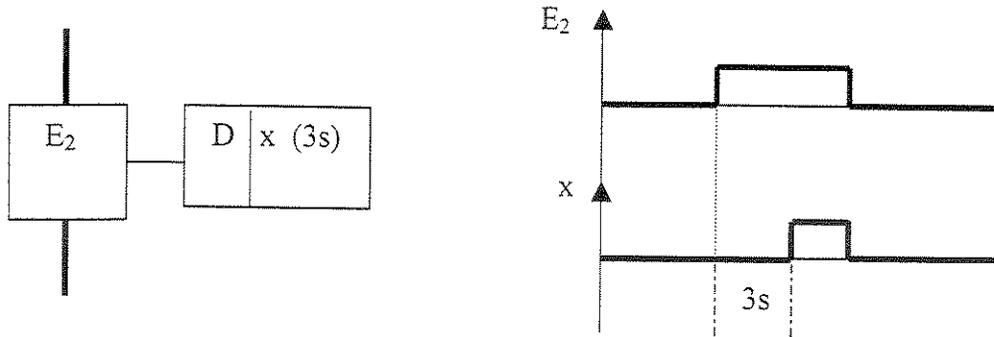


Fig. 3.5 – Ordem com Atraso de Tempo

### 3.2.3.4. – Limitada no Tempo ( L )

Trata-se de uma ordem limitada no tempo a qual é emitida logo após a ativação da etapa, porém sua duração é limitada a um valor de tempo específico ( fig. 3.6 ).

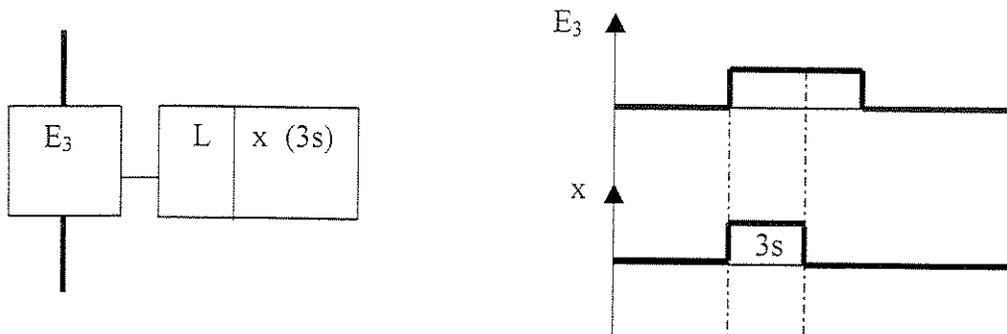


Fig. 3.6 – Ordem com Tempo Limitado

### 3.2.3.5 – Ordem Impulsional ( P )

Sua finalidade é atuar em elementos de comando tais como inicializador de temporizações, incremento ou decremento de contadores, armazenamento de dados em memória, etc. Pode ser de duas formas : Emissão apenas associada à ativação da etapa, ou condicionada ao aparecimento de uma outra variável, conforme apresentado na figura 3.7.

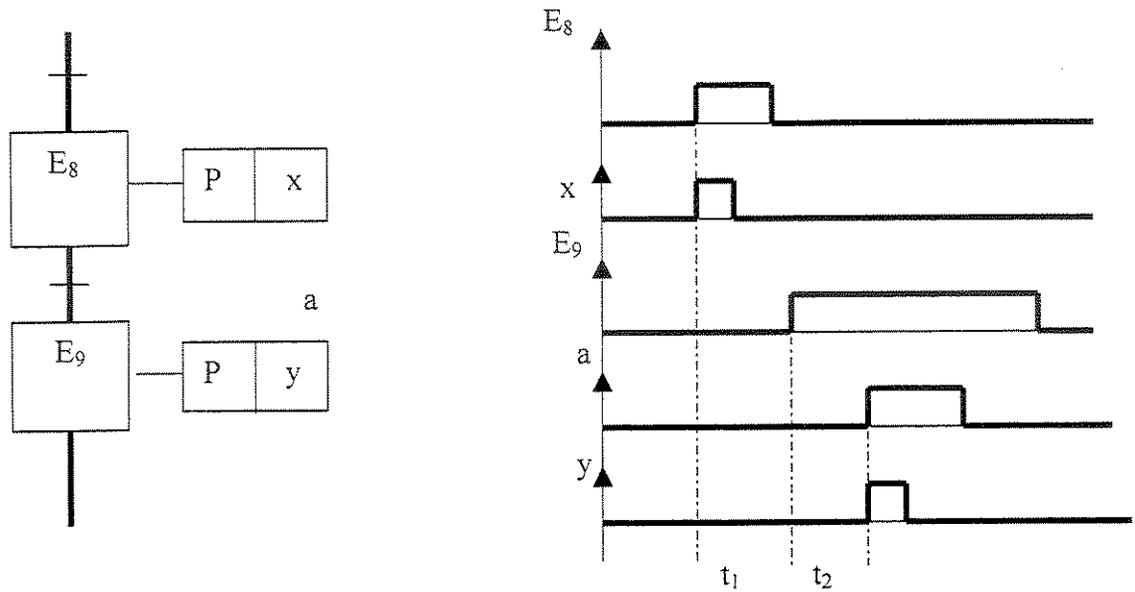


Fig. 3.7 – Ordem Impulsional e Impulsional Condicionado

### 3.2.3.6 – Ordem Armazenada com Retardo (SD)

Trata-se de uma ação associada a uma etapa onde a mesma é armazenada (S) e iniciada após um tempo especificado (D), mesmo que a etapa não esteja mais ativa. A ação é continuada até ser finalizada pela etapa seguinte (fig. 3.8).

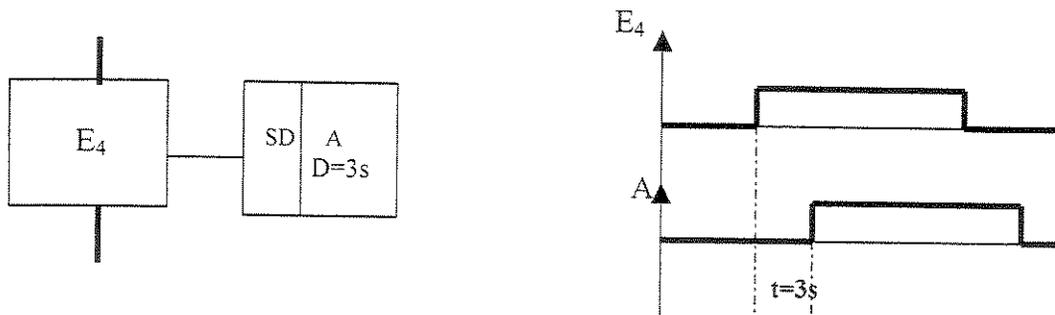


Fig. 3.8 – Ordem SD

### 3.2.3.7 – Ordem com retardo e Armazenada (DS)

Neste caso a ação associada à etapa é iniciada após o tempo especificado (D) e continuada (S) até ser finalizada pela etapa seguinte.

Se a etapa permanecer ativa por um período menor que o especificado, a ação não é iniciada (fig. 3.9).

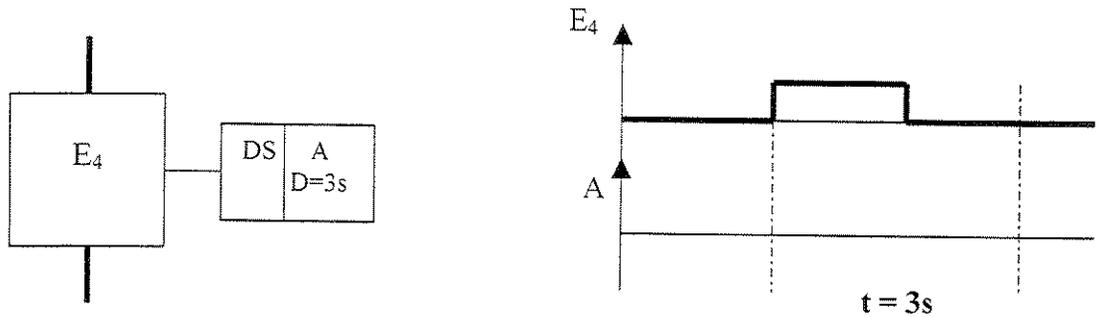


Fig. 3.9 – Ordem DS

### 3.2.3.8 – Ordem Condicional, Armazenada e Limitada no Tempo ( CSL )

A ação é iniciada desde que a condição lógica seja satisfeita ( C ) e mantida ( S ) até ser atingido o tempo especificado ( L ) mesmo que a etapa não seja mais ativa, ou até ser finalizada pela etapa seguinte ( fig. 3.10 ).

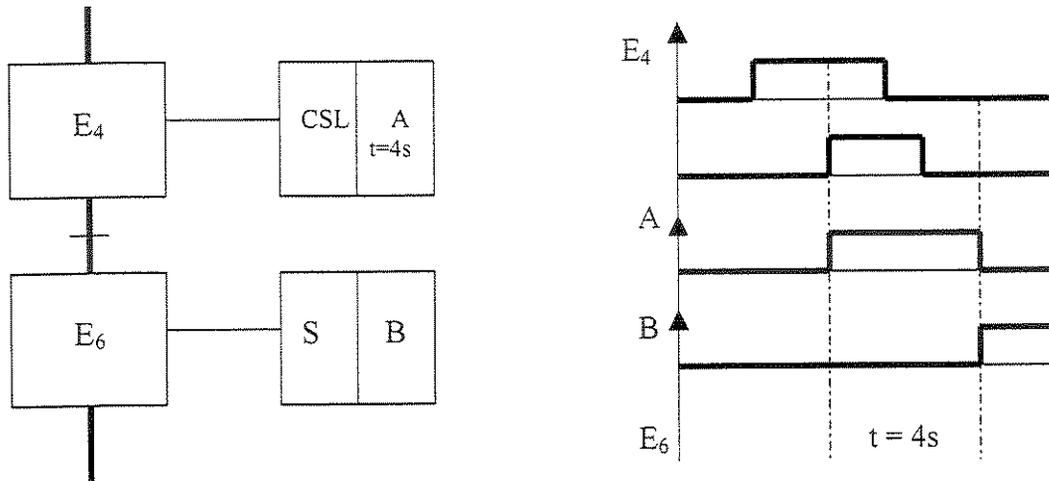


Fig. 3.10 – Ordem CSL

### 3.2.3.9. – Em Etapas Diversas

Quando se necessita que a mesma ação atue em mais de uma etapa, é possível a repetição da ordem de comando quantas vezes for necessário ( fig. 3.11.a ). Porém, convém observar que no caso da repetição ocorrer em etapas consecutivas ( fig. 3.11.b ), uma estrutura em paralelo pode ser usada alternadamente.

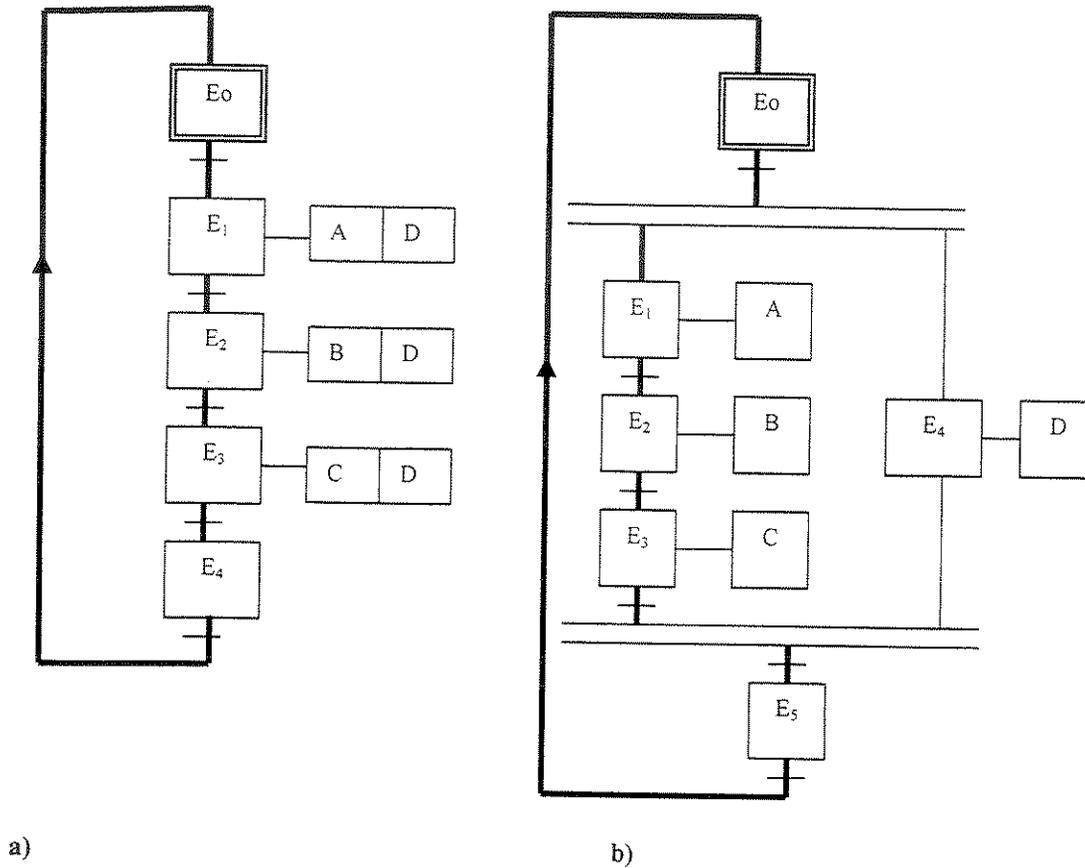


Fig. 3.11 – a) Repetição de Ações em Etapas Consecutivas  
b) Estrutura em Paralelo Equivalente

### 3.2.3.10. – Memorizada ( S ou R )

Uma ação cujo comportamento seja equivalente ao descrito pela figura 3.9 (b), pode ainda ser modelada por elementos de memorização. Neste caso, utiliza-se uma ação específica para ligar ( SET ) o elemento comandado , e outra ação para desligá-lo ( RESET ). No caso abordado na figura 3.12, a saída D permanecerá ligada durante o intervalo entre as transições  $T_1$  e  $T_2$ .

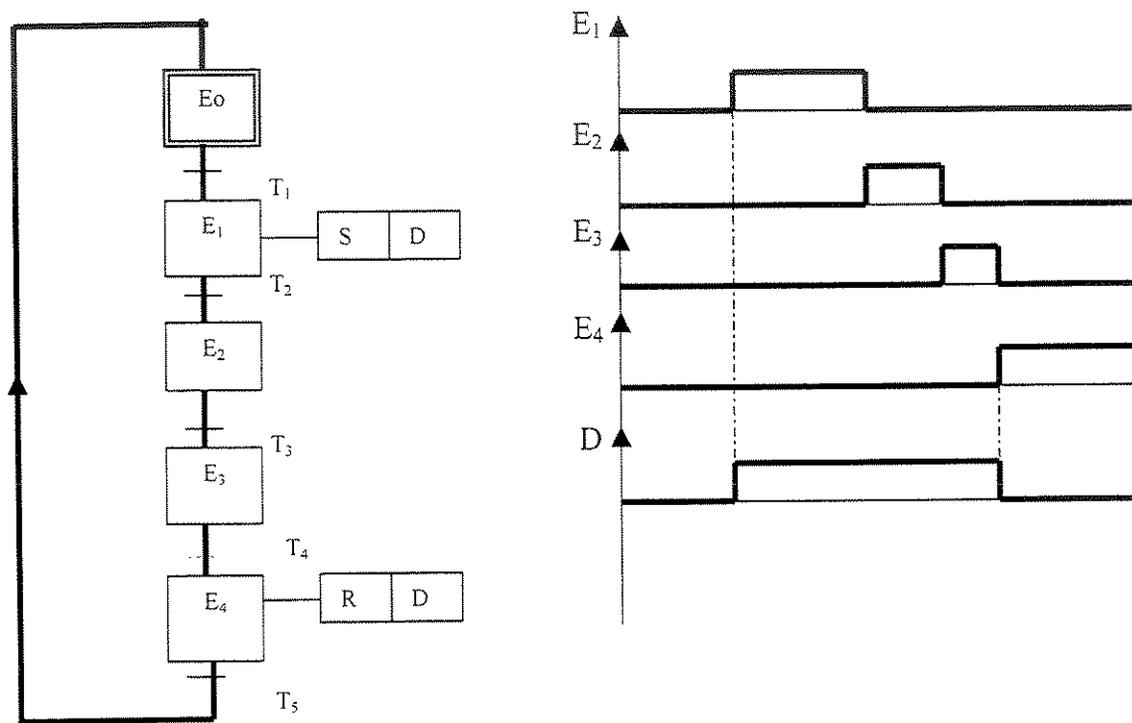


Fig. 3.12 – Ordem Memorizada Usando Indicadores S e R

### **3.3. Elementos Integradores**

A integração dos elementos do SAP é a fase mais importante e complexa nos projetos de automação. Isso porque as eficiências da parte operacional e da parte comando dependem de como os elementos serão integrados. Em um sistema automatizado e integrado alguns elementos são intrínsecos ao sistema, tais como :

- elementos de alimentação
- elementos de atuação
- elementos de transferência
- elementos de controle
- controladores lógicos programáveis

#### **3.3.1 – Elementos de Alimentação**

Nos processos de fabricação automatizados ( SAP ), utiliza-se vários recursos tecnológicos para inserir a matéria prima no processo, como sistema de vácuo, manipuladores robóticos, etc.

#### **3.3.2 – Elementos de Atuação**

Os elementos de atuação são responsáveis pela parte operativa, ou seja, os elementos que realizam trabalho no sistema.

Quanto à fonte de energia utilizada, pode-se classificar os atuadores em três tipos, ou seja, elétrico, pneumático e hidráulico.

### **3.3.3 – Elementos de Transferência**

Como um SAP é formado por vários postos ou células de trabalho, é necessário que haja uma transferência física de produtos entre os mesmos, compatível com o tipo de produto e integrado com os elementos anteriores e posteriores. Essa transferência pode ser feita através de mesa rotativa, acionada por motor de corrente contínua ou por um passo, por indexador, por transportadores, por sistema de vácuo, etc.

### **3.3.4 – Elementos de Controle**

Os elementos de controle assumem dentro do SAP uma função de emitir sinais de comando, cujos principais são os sensores e as chaves fim de curso.

### **3.3.5 – Controladores Programáveis**

Trata-se de um equipamento eletrônico de grande importância no SAP, pelo controle dos dados de entrada e de saída, já que é um equipamento com memória programável. As características dos controladores lógicos programáveis ( CLP ) são :

- Permite a entrada e saída de dados com fácil reposição.
- Funcionamento eficaz em ambientes industriais agressivos como : poeira, calor, vibração, corrosão, etc.
- Facilidade de programação, manutenção, instalação e ampliação.
- Repetibilidade de operação e confiabilidade.

Pode-se perceber que pelas características dos controladores lógicos ( descritas no anexo V ), eles têm grande importância como elementos de integração de sistemas automatizados.

### 3.4 – Exemplo de Manipulação de Peças

A figura 3.13 apresenta o esquema de um manipulador de peças, cuja função é alimentar uma forja com peças aquecidas, obtidas de três fornos ( $F_1$ ,  $F_2$  e  $F_3$ ).

Os fornos  $F_1$  e  $F_2$  e a entrada da forja se situam num mesmo plano vertical, enquanto que o forno  $F_3$  está mais à frente, num plano alcançado pelo avanço do atuador C.

A forja e os fornos  $F_1$  e  $F_3$  estão numa posição mais alta, num mesmo plano horizontal, correspondendo à posição do atuador A recuado, conforme figura.

As peças podem ser colocadas aleatoriamente nos três fornos, com tempos de aquecimento não necessariamente iguais, o que implica que as peças também podem ficar prontas em ordem aleatória.

Caso os três fornos disponibilizem as peças ao mesmo tempo, prevalece a ordem dos fornos, primeiro  $F_1$  antes de  $F_2$  e este antes que  $F_3$ .

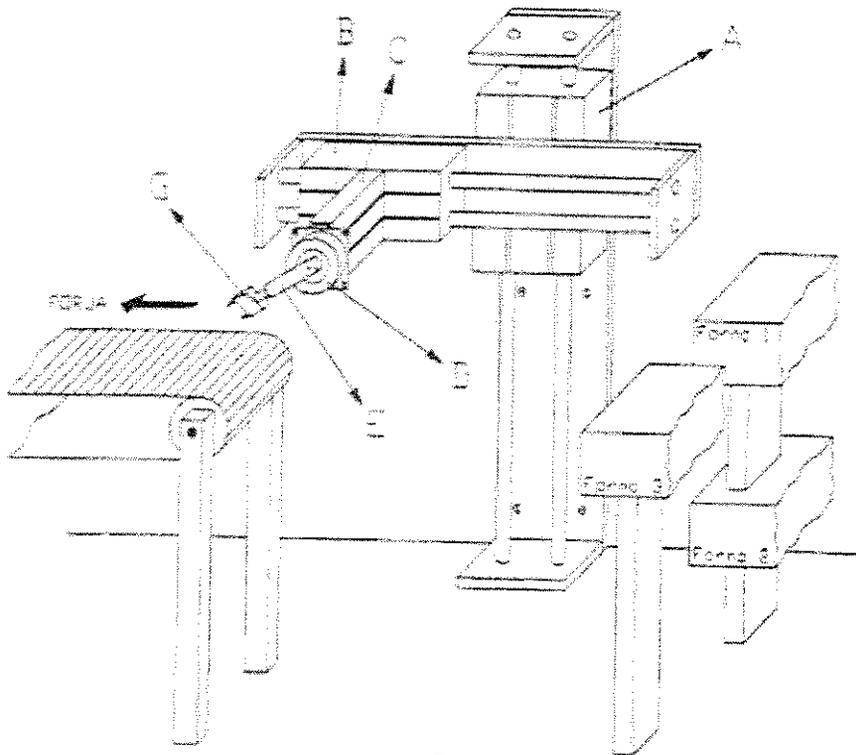


Figura 3.13 – Manipulador Pneumático para Alimentação de Forja .

### 3.4.1 – Relação de Entradas e Saídas dos Atuadores.

Para descrever a função das variáveis em relação às entradas e saídas , utilizou-se as tabelas 3.2 e 3.3 respectivamente.

Variáveis	In	Descrição da Função
Sens_ atead A _ rec	X <sub>1</sub>	Atuador A recuado
Sens_ atead A _ av	X <sub>2</sub>	Atuador A avançado
Sens_ atead B _ rec	X <sub>3</sub>	Atuador B recuado
Sens_ atead B _ av	X <sub>4</sub>	Atuador B avançado
Sens_ atead C _ rec	X <sub>5</sub>	Atuador C recuado
Sens_ atead C _ av	X <sub>6</sub>	Atuador C avançado
Sens_ atead D _ rec	X <sub>7</sub>	Atuador D recuado
Sens_ atead D _ av	X <sub>8</sub>	Atuador D avançado
Sens_ atead E _ rec	X <sub>9</sub>	Atuador E recuado
Sens_ atead E _ av	X <sub>10</sub>	Atuador E avançado
Sens_ atead G _ rec	X <sub>11</sub>	Atuador G recuado
Sens_ atead G _ av	X <sub>12</sub>	Atuador G avançado

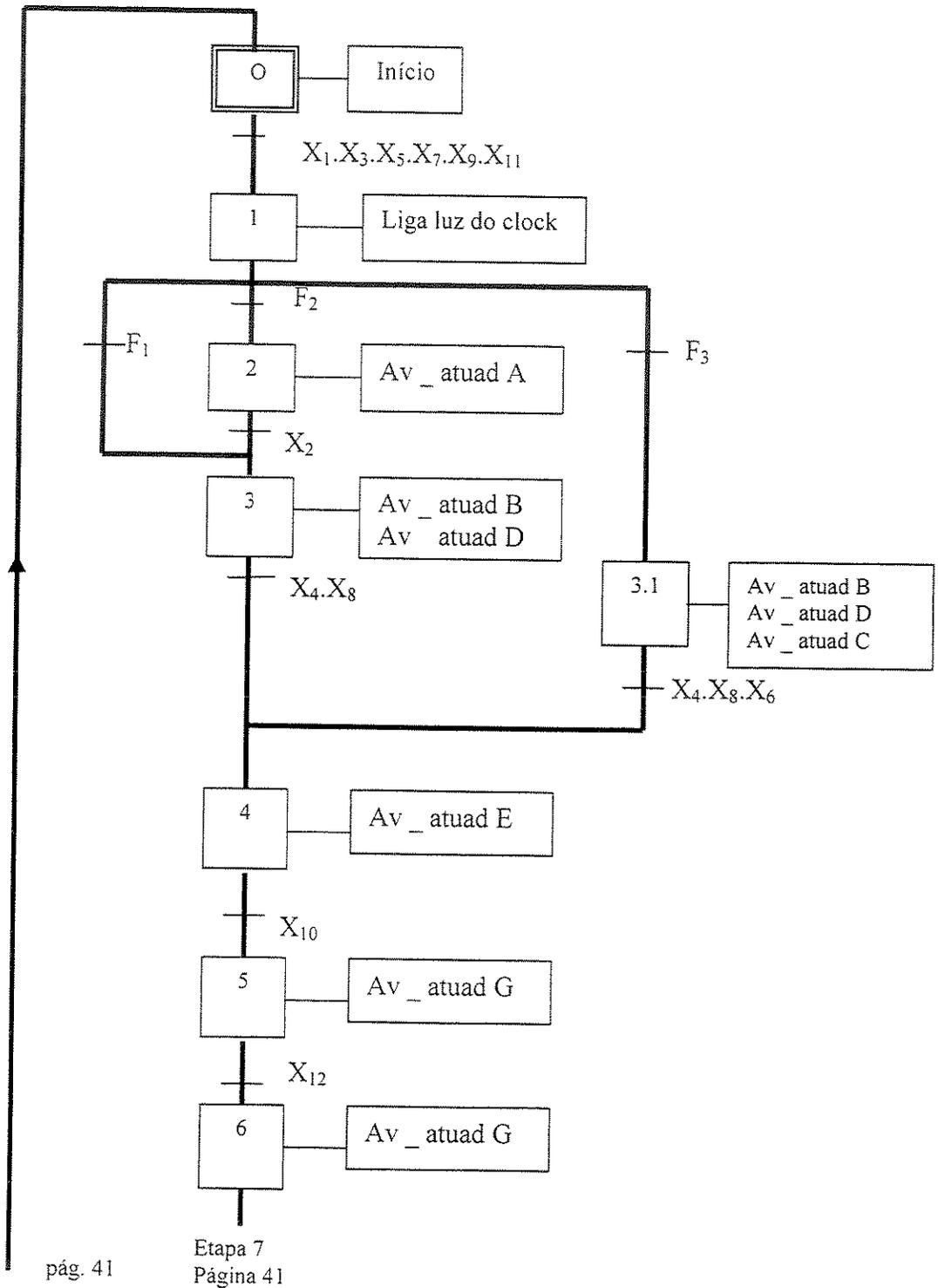
Tabela 3.2 – Relação de Entradas do Manipulador.

Variáveis	Out	Descrição da Função
Sol _ av _ atead A	Y <sub>1</sub>	Aciona solenóide _ avança atuador A
Sol _ av _ atead B	Y <sub>2</sub>	Aciona solenóide _ avança atuador B
Sol _ av _ atead C	Y <sub>3</sub>	Aciona solenóide _ avança atuador C
Sol _ av _ atead D	Y <sub>4</sub>	Aciona solenóide _ avança atuador D
Sol _ rec _ atead D	Y <sub>5</sub>	Aciona solenóide _ recua atuador D
Sol _ av _ atead E	Y <sub>6</sub>	Aciona solenóide _ avança atuador E
Sol _ av _ atead G	Y <sub>7</sub>	Aciona solenóide _ avança atuador G

Tabela 3.3 – Relação de Saídas do Manipulador.

### 3.4.2 – Grafet Tecnológico do Manipulador

A seguir é apresentado o grafet funcional tecnológico do manipulador conforme figura 3.14.



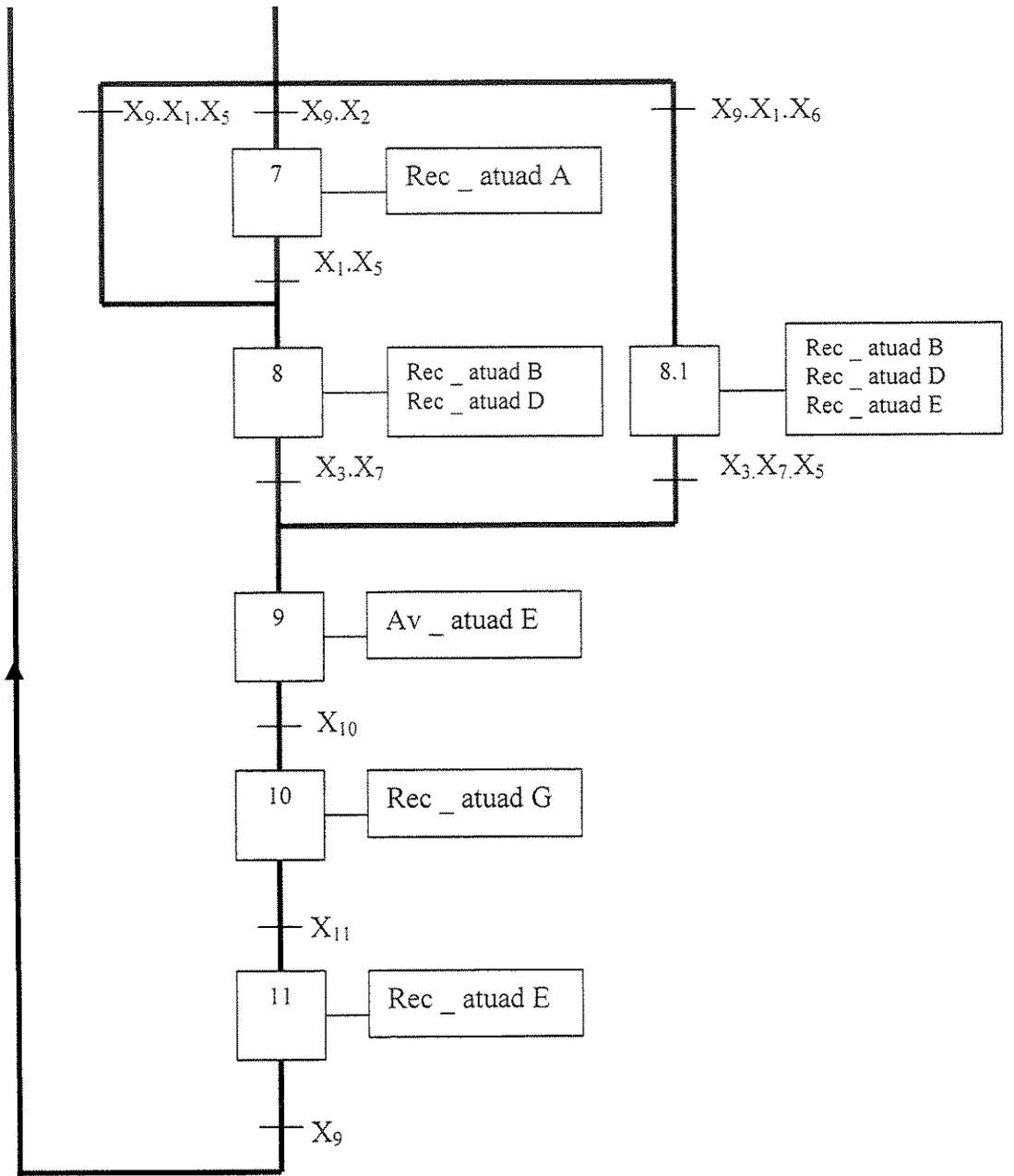


Figura 3.14 – Grafset Tecnológico

### **3.5 – Conclusão**

Para um sistema automatizado de produção ( SAP ) a parte fundamental é a definição estrutural de todo processo a ser automatizado. Definindo de forma clara a parte comando e a parte operativa, assim como a descrição do sistema como um todo, de forma objetiva. As entradas e saídas devem ser relacionadas por postos de trabalho, todas as vezes que houver integração dos mesmos. A ferramenta de integração deve facilitar a visualização da evolução de todo conjunto automatizado e integrado. Neste caso, o grafcet é a ferramenta mais eficaz para sistemas complexos.

Deve-se , sempre que possível, tornar o processo automatizado aberto a novas tecnologias, modulando-o para facilitar futuras instalações.

Este capítulo teve como objetivo, dar as diretrizes gerais para implantação de um SAP.

## CAPÍTULO 4

### Implementação e Integração Final de Uma Plataforma de Manipulação

Este capítulo mostra o projeto e implementação de três postos de trabalho integrados e automatizado de produção ( SAP ) , são representados, o sistema de alimentação, transferência e descarregamento do produto. A figura 3.1 mostra os postos de trabalho da plataforma.

Os postos de trabalho são compostos pelos seguintes elementos :

- Manipulador Pneumático – responsável pela alimentação do produto.
- Mesa rotativa de Transferência – Serve como elo de ligação entre alimentação e descarga.
- Manipulador Robótico – Faz o descarregamento do produto.

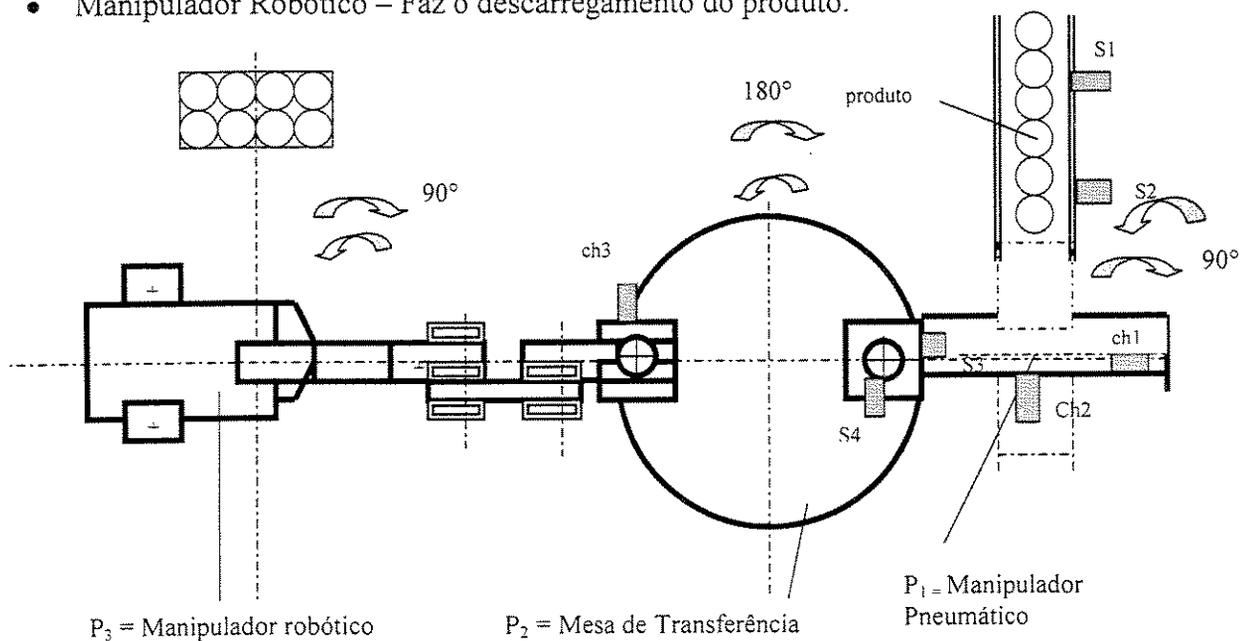


Fig. 4.1 – Postos de Trabalho Automatizado e Integrado da Plataforma.

#### 4.1 – Especificação Funcional

O conjunto que forma a plataforma com elementos integrados para SAP, foi dividido em parte operativa ( Po ), constituída de um manipulador pneumático, mesa de transferência acionado com motor de C. C. e um manipulador robótico. A parte comando ( Pc ) contém botão de partida, sensores, chave óptica, controlador lógico programável ( CLP ) e um microcomputador para “controlar o manipulador robótico”.

Conforme esquema apresentado na fig. 4.1, os sensores  $S_1$  e  $S_2$  detectam o estoque ( 5 produtos ) de produto na calha de alimentação e o posicionamento do primeiro produto respectivamente.

Acionando um botão de inicialização, o braço giratório do manipulador pneumático deverá estar posicionado pela chave 1 (ch1). O atuador linear avança, sensor 3 aciona o solenóide, aciona o vácuo através de uma eletroválvula direcional pneumática. O produto é succionado pelo vácuo e preso por uma ventosa fixada na extremidade do atuador, que ao recuar, inicia-se o movimento através do atuador rotativo em  $90^\circ$ . A chave óptica ( ch1 ) permite a parada do braço giratório, após avanço do atuador linear, interrompe-se o vácuo, assim o produto é colocado na mesa de transferência. O sensor  $S_3$  detecta a presença de produto na mesa de transferência e permite que essa gire  $180^\circ$  anti-horário através de um motor C.C.. A presença de produto e a chave óptica (ch2) da mesa no final do giro dá-se o início aos movimentos do manipulador robótico, para descarregamento, colocando-o em caixa de embalagem. Os movimentos do manipulador robótico são programados por uma linguagem própria, cujas entradas são I/O e as saídas são as trajetórias para realização da tarefa. Após a realização da tarefa pelo manipulador robótico ( Robix ) retorna a sua posição original e a mesa é acionada em  $180^\circ$  horário, completando assim o ciclo da plataforma.

A arquitetura da plataforma e os respectivos postos de trabalho são representados pelo diagrama de blocos conforme figura 4.2.

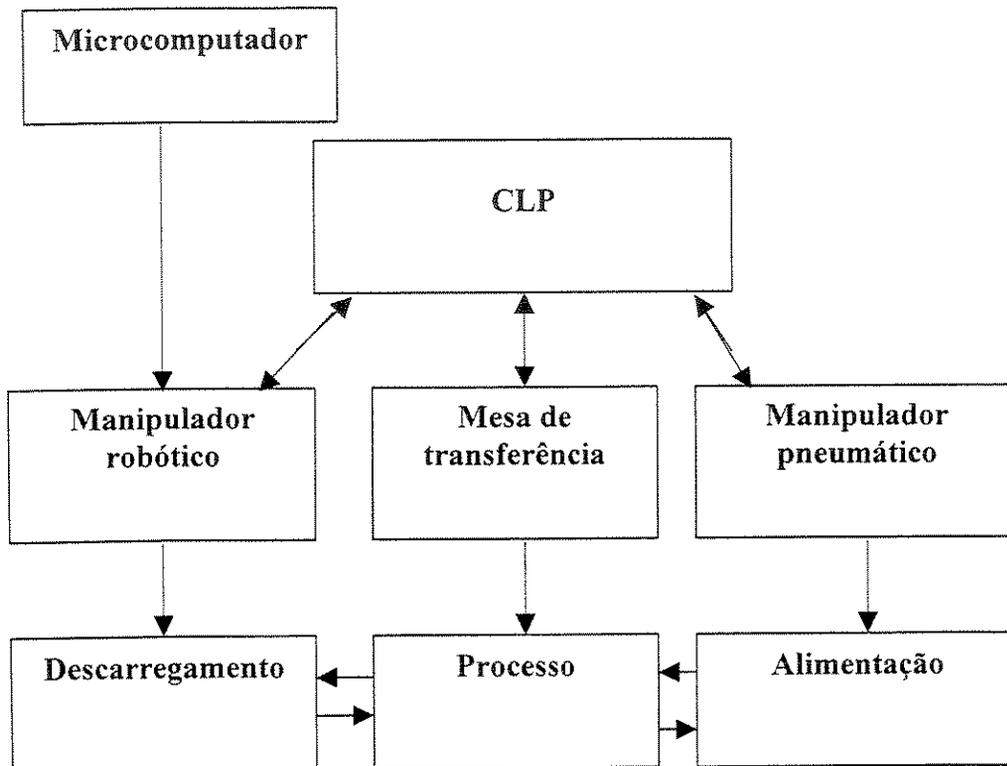


Fig. 4.2 – Diagrama de Blocos da Plataforma

A integração dos postos de trabalho é o propósito deste trabalho. A ferramenta de integração utilizada é o grafcet ( figura 4.3 ) por ser eficiente, de fácil interpretação e visualização.

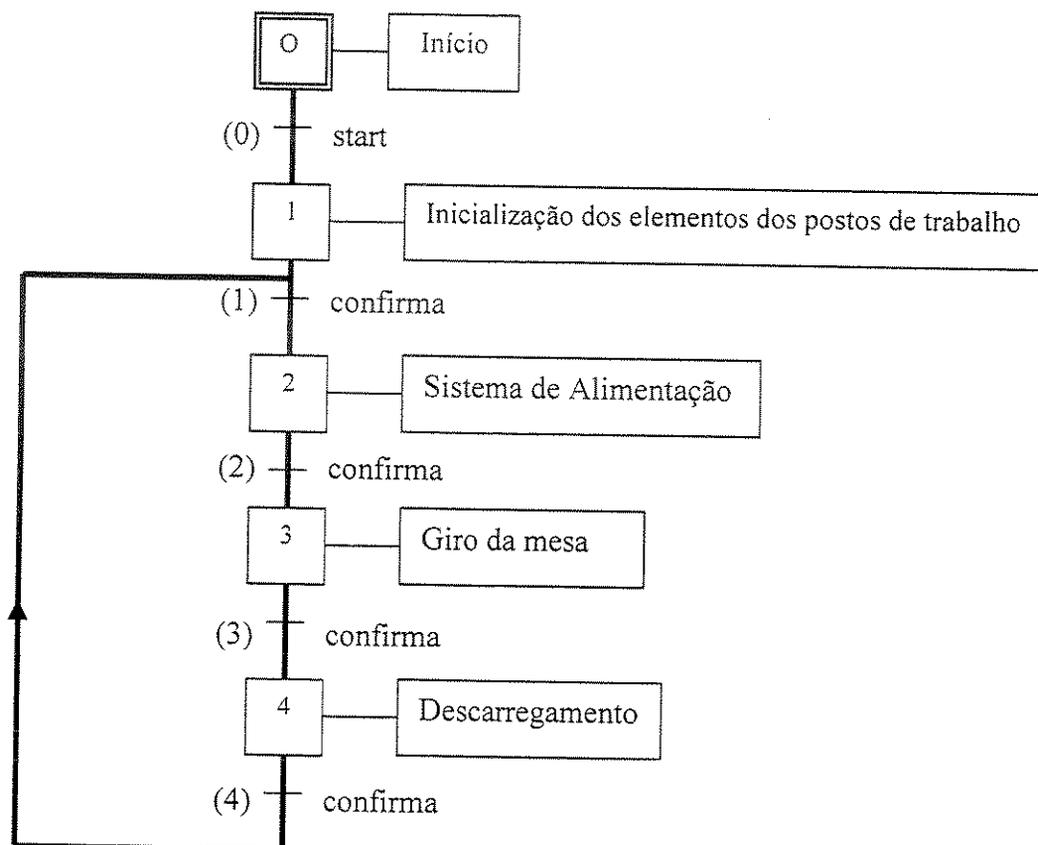


Figura 4.3 - Grafcet Funcional da Plataforma

Como pode ser observado na figura 3.3, fica mais fácil de visualizar o funcionamento da plataforma em comparação com a descrição verbal, que pode causar ambiguidade e/ou confundir na interpretação.

A seguir, os elementos do SAP serão descritos, considerando-se a Parte Operativa ( Po ) e a Parte Comando ( Pc ).

#### **4.2 – Manipulador Pneumático**

A alimentação da plataforma constitui-se de um manipulador pneumático composto, em sua parte operativa, de :

- Atuador Pneumático Linear
- Sistema de Vácuo
- Atuador Pneumático Rotativo
- Eletroválvulas Pneumáticas
- Conjunto Mecânico

Na parte comando é representada pelos sensores que determinam o início e o fim dos movimentos dos atuadores ( linear e rotativo ), abertura e fechamento do vácuo, posicionamento do conjunto mecânico e do produto ( estoque e presença ).

### 4.2.1 – Montagem do Manipulador Pneumático

O conjunto que compõe o manipulador pneumático é formado por um atuador linear, adaptado para aplicação de vácuo através de sua haste furada onde uma ventosa é adaptada, formando a garra que segura o produto.

Um braço acoplado ao eixo do atuador rotativo que gira em ângulo de 90° e todo conjunto é suportado por uma estrutura metálica, conforme figura 4.4.

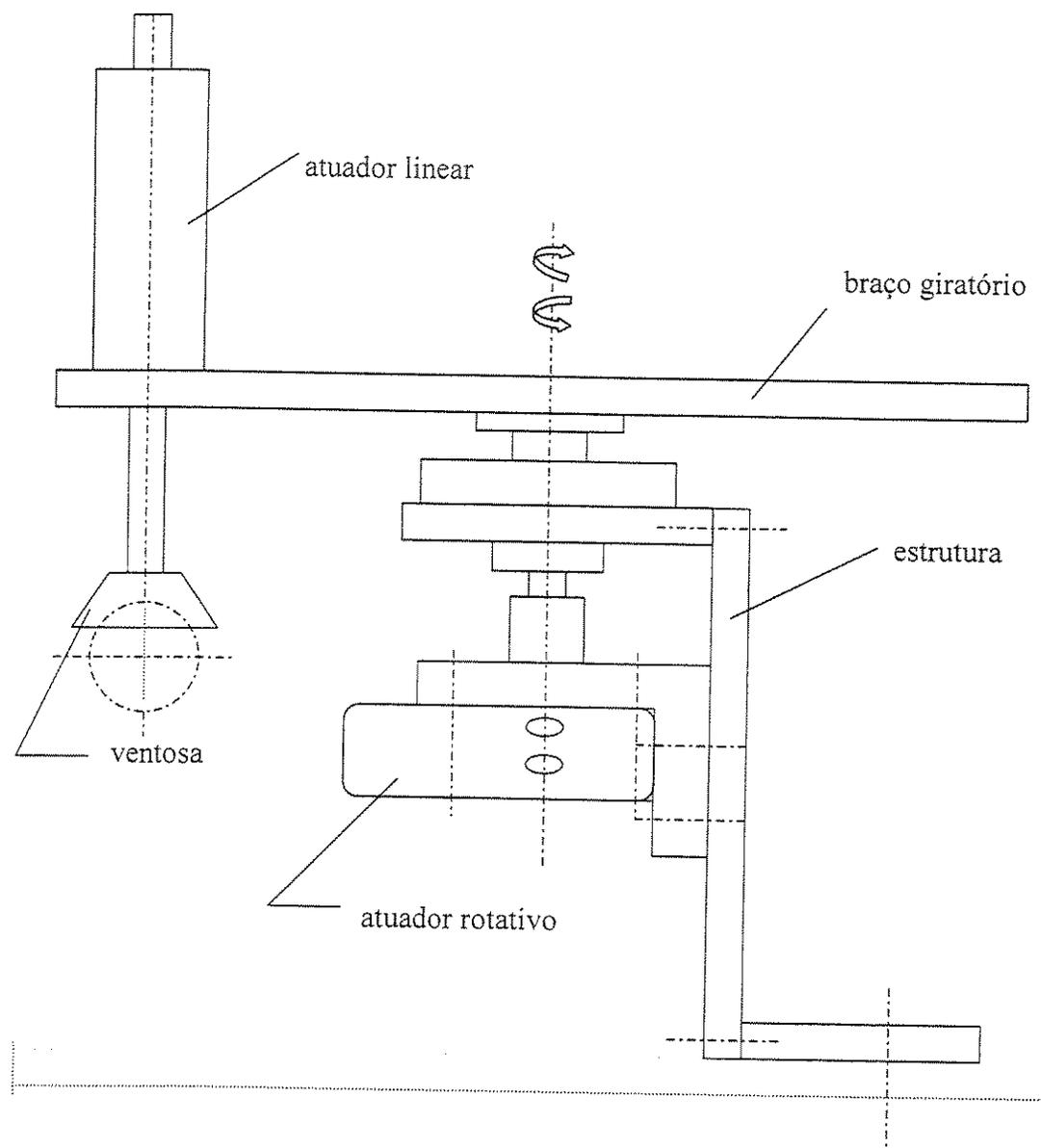


Fig. 4.4 – Esquema de Montagem do Manipulador Pneumático

#### 4.2.2 – Descrição da Parte Operativa

O manipulador pneumático tem a parte operativa ( Po ), representada por atuadores ( linear e rotativo ) e as eletroválvulas e a parte comando ( Pc ) corresponde a botão de partida, sensores, CLP e PC. Através do diagrama de bloco da figura 4.5, pode-se também visualizar além das partes operativa e comando, as entradas e saídas.

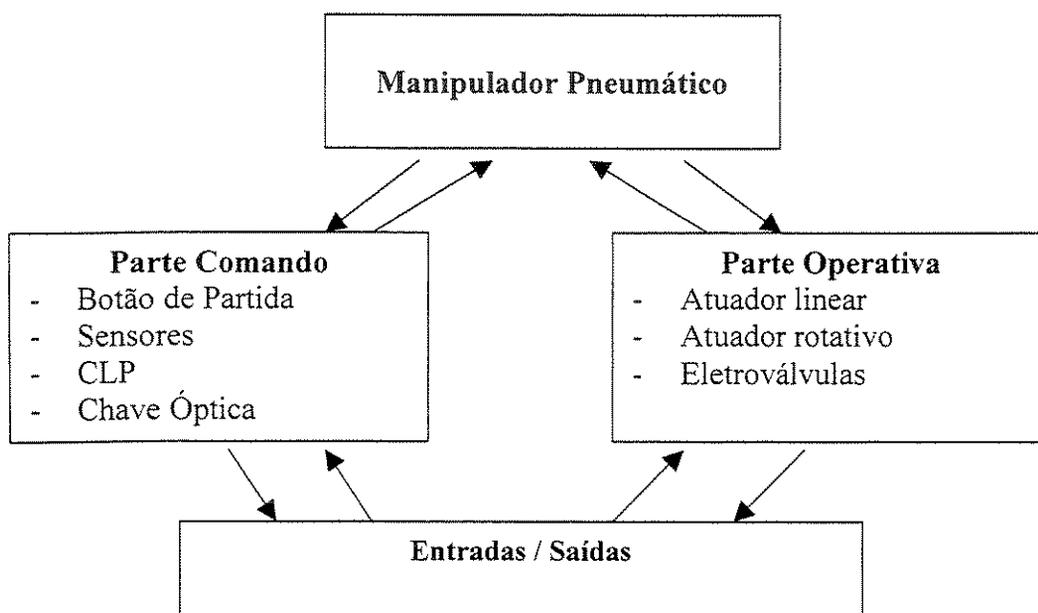


Fig. 4.5 – Diagrama de Blocos do Manipulador Pneumático.

### 4.2.3 – Relação das Entradas e Saídas

A fim de tornar mais compreensível a sequência de funcionamento do manipulador pneumático, as tabelas 4.1 a e 4.1 b relacionam as variáveis com a função de cada entrada e saída no manipulador, respectivamente. As etapas e transições que ocorrem serão representadas através do grafcet ( fig. 4.6 ).

Variáveis	Entradas	Descrição da Função
Sensor _ estoque	X <sub>0</sub>	Estoque de produto na calha
Sensor _ presença de produto	X <sub>1</sub>	Presença de produto na calha
Sensor _ posicionamento do braço	X <sub>2</sub>	Posicionamento do braço do manipulador pneumático
Sensor _ avanço do cilindro	X <sub>3</sub>	Acionamento da solenóide – avanço do cilindro
Sensor _ liga o vácuo	X <sub>4</sub>	Acionamento da solenóide – liga o vácuo
Sensor _ retorno do cilindro	X <sub>5</sub>	Acionamento da solenóide – retorno do cilindro
Sensor _ gira atuador	X <sub>6</sub>	Acionamento da solenóide – gira atuador 90° anti-horário
Sensor _ posicionamento da mesa	X <sub>7</sub>	Posicionamento da mesa de transferência
Sensor _ avanço do cilindro	X <sub>8</sub>	Acionamento da solenóide – avanço do cilindro
Sensor _ desliga o vácuo	X <sub>9</sub>	Acionamento da solenóide – desliga o vácuo
Sensor _ retorno do cilindro	X <sub>10</sub>	Acionamento da solenóide – retorno do cilindro
Sensor _ gira atuador	X <sub>11</sub>	Acionamento da solenóide – gira atuador 90° horário

Tabela 4.1 a – Relação de Entradas do Manipulador Pneumático.

Variáveis	Saídas	Descrição da Função
Cilindro pneumático	Y <sub>0</sub>	Acionamento do cilindro – avança
Vácuo	Y <sub>1</sub>	Acionamento do vácuo
Cilindro pneumático	Y <sub>2</sub>	Acionamento do cilindro – retorno
Atuador rotativo pneumático	Y <sub>3</sub>	Acionamento do atuador rotativo – 90° anti-horário
Cilindro pneumático	Y <sub>4</sub>	Acionamento do cilindro – avança
Vácuo	Y <sub>5</sub>	Desligamento do vácuo
Cilindro pneumático	Y <sub>6</sub>	Acionamento do cilindro – retorno
Atuador rotativo	Y <sub>7</sub>	Acionamento do atuador rotativo – 90° horário

Tabela 4.1 b – Relação das Saídas no Manipulador Pneumático

#### 4.2.4 – Grafset Sequencial do Manipulador Pneumático

Na figura 4.11 é apresentado o grafset seqüencial do manipulador pneumático.

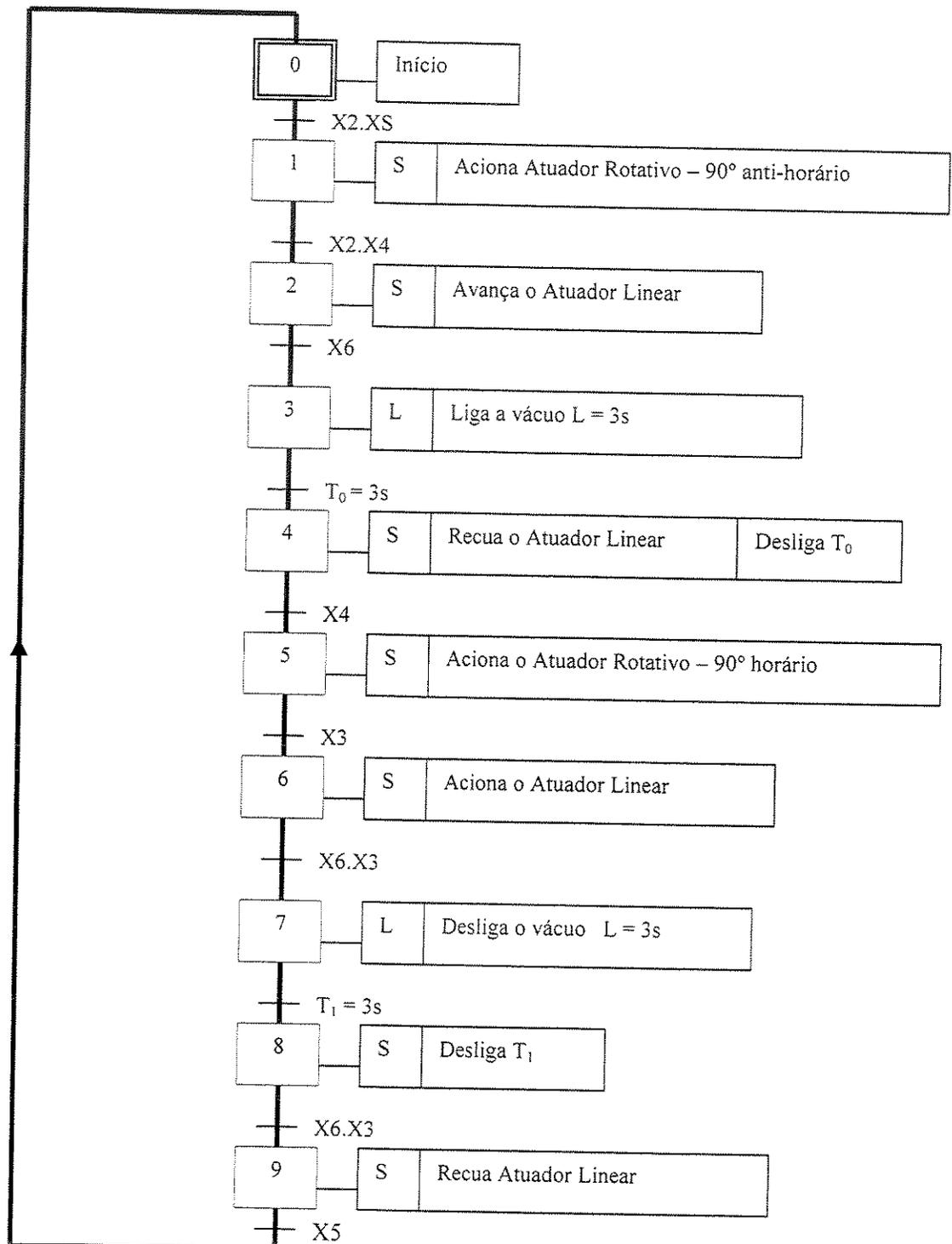


Fig. 4.6 – Grafset Funcional do Manipulador Pneumático

### 4.3 – Sistema de Transferência

O sistema é composto por uma mesa rotativa, acionada por um moto-reductor de C.C.. Os movimentos são alternados e o ângulo de giro corresponde a 180°, com o objetivo de transferir o produto do posto de alimentação ( manipulador pneumático ) para o poste de descarregamento ( manipulador robótico ). A parte comando é representada por sensores que determinam a posição de parada da mesa, seu sentido de giro e a presença de produto na mesma e o CLP.

#### 4.3.1 – Mesa de Transferência

Na plataforma, a mesa rotativa de transferência é um posto intermediário. A parte operativa é composta por um conjunto mecânico ( mesa, eixo, acoplamento e suportes ) e um moto-reductor de c.c.. A parte de comando é representada por sensores, CLP conforme a figura 4.7.

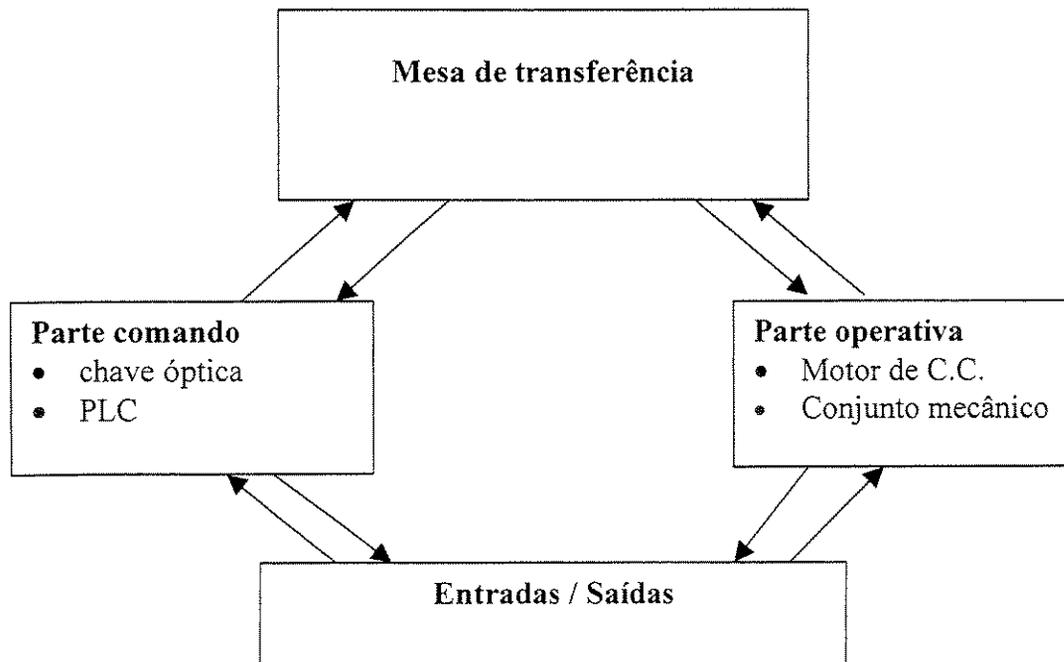


Fig. 4.7 – Parte Comando e Operativa da Mesa de Transferência.

### 4.3.2 – Montagem da Mesa de Transferência

A parte estrutural da mesa é constituída de um disco que faz a transferência propriamente dita, a qual é suportada por apoios. O motor é preso por suporte conforme figura 4.8.

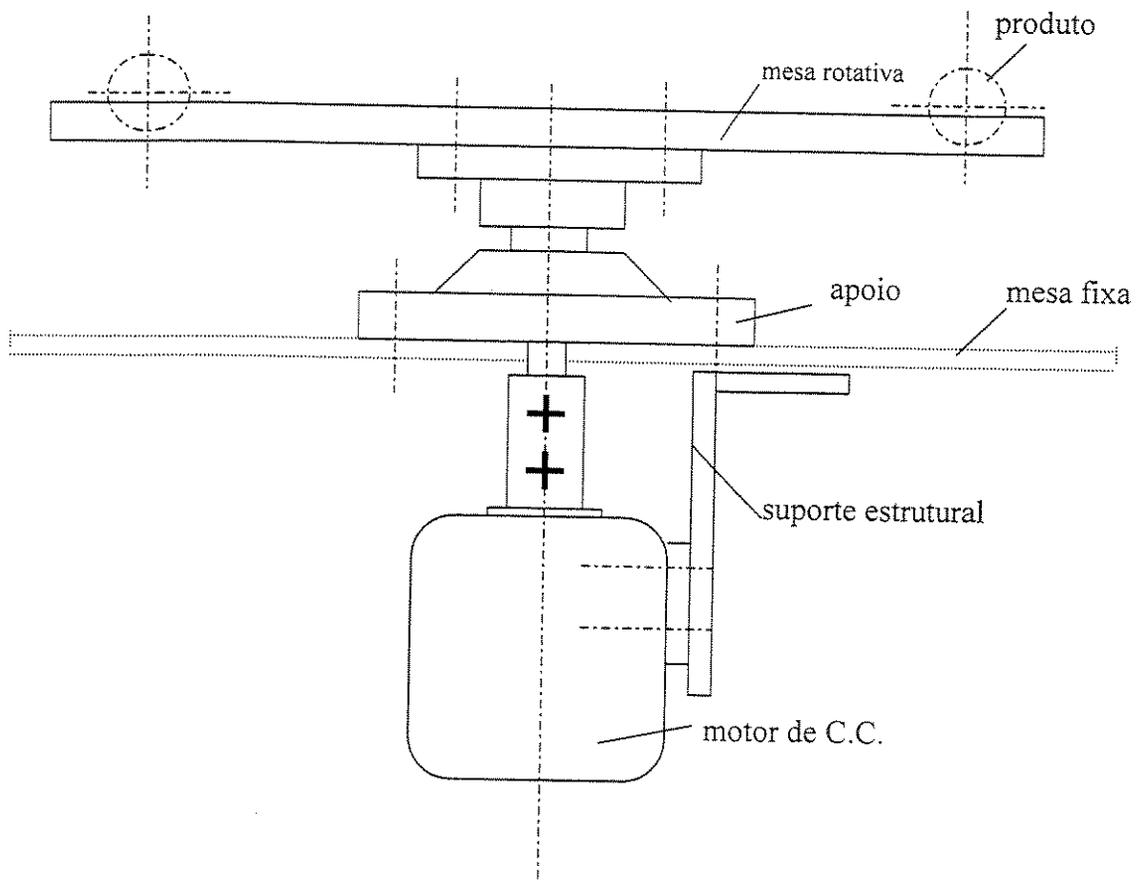


Fig. 4.8 – Conjunto da Mesa de Transferência

### 4.3.3 – Relação de Entradas e Saídas

O detalhamento das entradas ( tab. 4.2 a ) e saídas ( tab. 4.2 b ) e o graficet ( fig. 4.9 ) são representados a seguir.

Variáveis	Entradas	Descrição da Função
Chave óptica – 3 – desliga o motor da mesa rotativa de transferência	X <sub>7</sub>	Desliga o motor da mesa rotativa de transferência, para movimentos de 180° no sentido horário e anti-horário

Tab. 4.2 a – Relação de Entradas da Mesa Rotativa de Transferência

Variáveis	Saídas	Descrição da Função
Motor da mesa rotativa de transferência	Y <sub>6</sub>	Liga o motor da mesa rotativa de transferência – giro de 180° no sentido horário
Motor da mesa rotativa de transferência	Y <sub>7</sub>	Liga o motor da mesa rotativa de transferência – giro de 180° no sentido anti-horário

Tab. 4.2 b – Relação de Saídas da Mesa Rotativa de Transferência

#### 4.3.4 – Grafet Funcional da Mesa Rotativa de Transferência

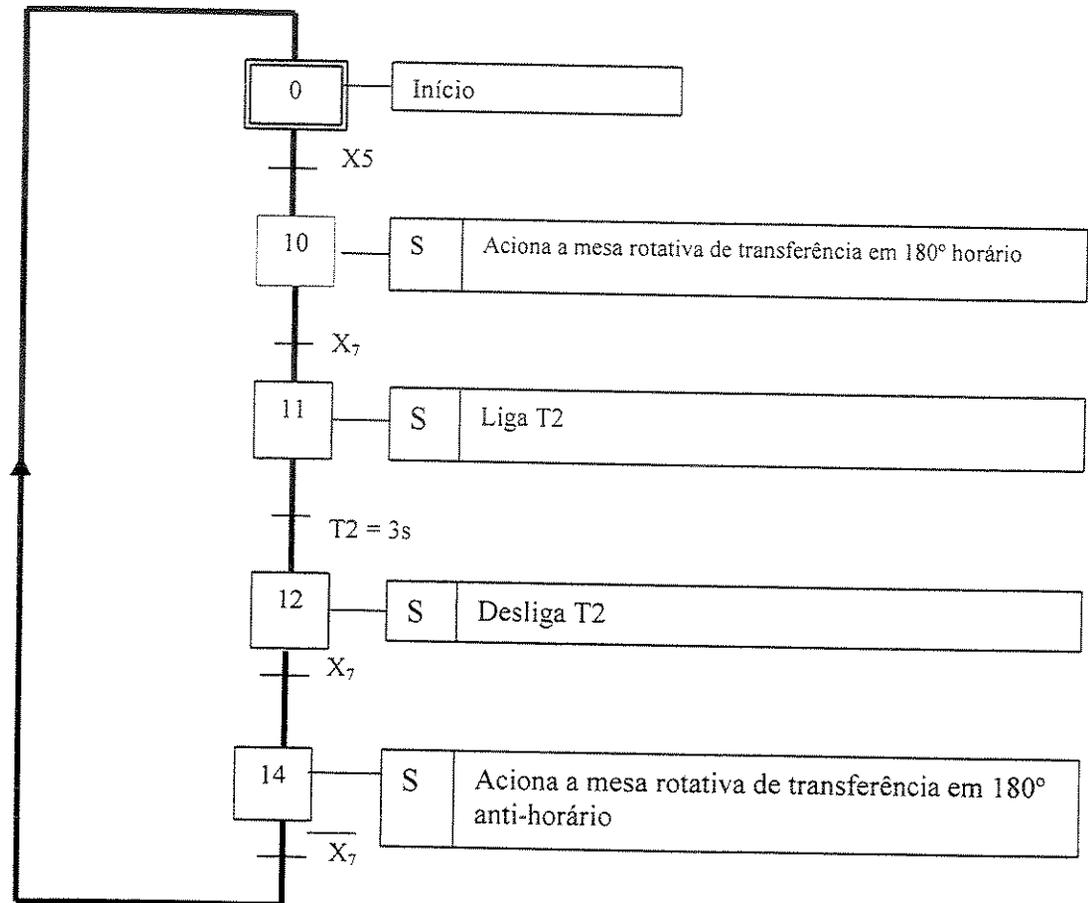


Fig. 4.9 – Grafet Funcional da Mesa Rotativa

#### 4.4 – Manipulador Robótico – Robix™

O sistema robótico utilizado na plataforma, é um kit didático para aprendizado de robótica. Trata-se de um elemento que pode ser montado em várias configurações, possui 5 graus de liberdade, mais um servo motor para garra. Utilizou-se para esse trabalho, uma disposição semelhante a um braço mecânico articulado, denominado de manipulador robótico ( figura 4.10 ).

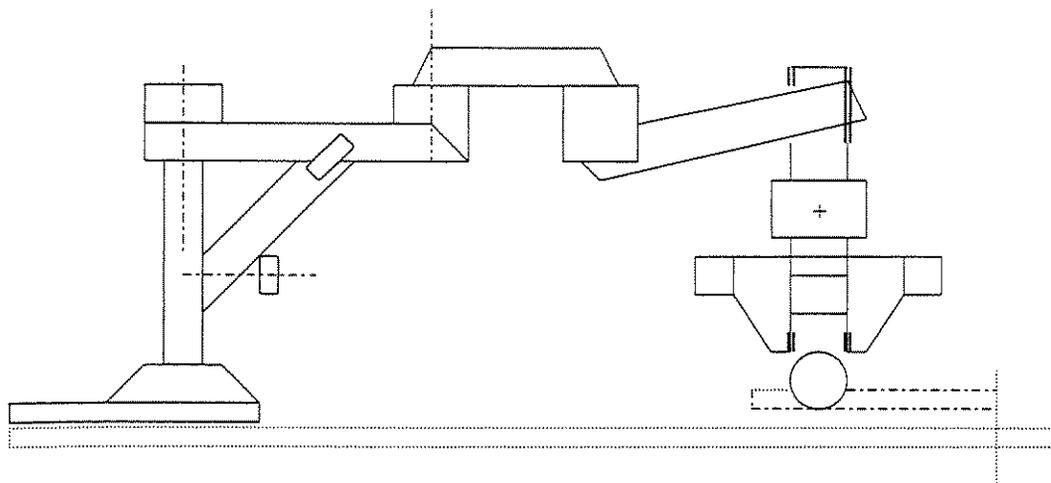


Fig. 4.10 – Configuração Estrutural do Robix™

Será utilizado apenas 4 graus de liberdade do Robix para realização da tarefa composta por 4 trajetórias. O manipulador robótico foi introduzido na plataforma com a finalidade de descarregar o produto. Isso ocorre quando a mesa de transferência estiver devidamente posicionada com o produto. As trajetórias do Robix serão programadas para que os servomotores operem segundo parâmetros pré-estabelecidos pelo usuário, como aceleração, desaceleração, posição inicial, etc... No caso da plataforma, a operação consiste em pegar o produto e levá-lo à posição de embalagem, deslocando-o em 90°.

#### 4.4.1 – Entradas e Saídas e PC / PO do Manipulador Robótico

O manipulador Robix™ tem a parte comando ( PC ) representada por sensores, CLP e PC e a parte operativa ( PO ) através dos servo-motores. Já as entradas e saídas são determinadas por I/O e trajetórias respectivamente ( figura 4.11 ).

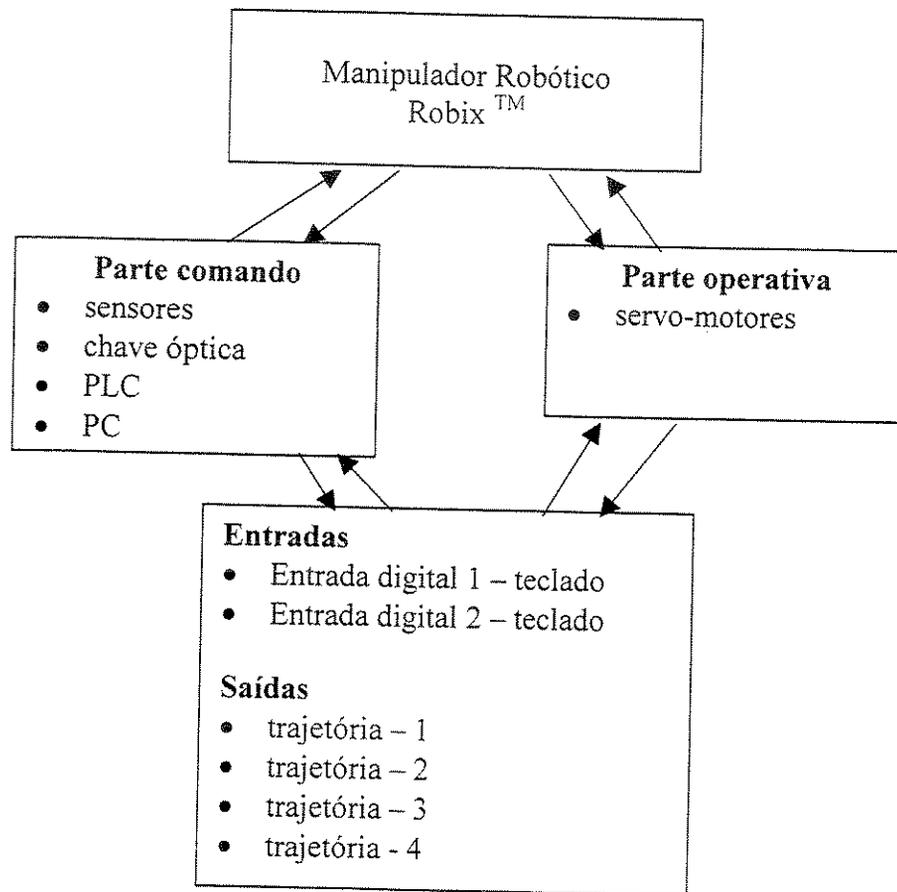


Fig. 4.11 – Representação da PC / PO e Entrada/Saída do Manipulador Robótico.

#### **4.5 - Conclusão**

A plataforma foi projetada e implementada com o objetivo de convalidar este trabalho, porém também permite a utilização prática de um sistema automatizado de produção ( SAP ), já que os postos de trabalho também retratam a realidade do chão da fábrica. Foi possível perceber que um sistema integrado precisa ser estruturado de forma clara para facilitar o entendimento, a implantação, modificações e ampliações. A importância em utilizar a ferramenta de integração adequada cresce na mesma proporção que o grau de complexidade do sistema a ser automatizado aumenta.

## CAPÍTULO 5

### Integração Final dos Postos de Trabalho da Plataforma

Como já mencionado ( capítulo 3 ) a integração dos elementos de um Sistema Automatizado de Produção ( SAP ) constitui-se em uma das principais fases nos projetos de automação. Portanto, a linguagem a ser adotada deve ter consistência e clareza, a fim de evitar dúvidas na interpretação dos dados da estruturação. Outra condição importante é garantir a versatilidade, manutenção e a expansão do projeto.

Este capítulo tem como objetivo a integração dos postos de trabalho que formam o conjunto da plataforma. Para integrar os postos de trabalho, utilizou-se o grafcet como ferramenta, pelas vantagens que essa linguagem oferece como, facilidade na interpretação, normalização e inexistência de ambiguidade. A plataforma foi dividida em macro-etapas para facilitar a programação do CLP, e o controle sobre o acionamento dos elementos. A figura 5.1 mostra o Grafcet funcional da plataforma.

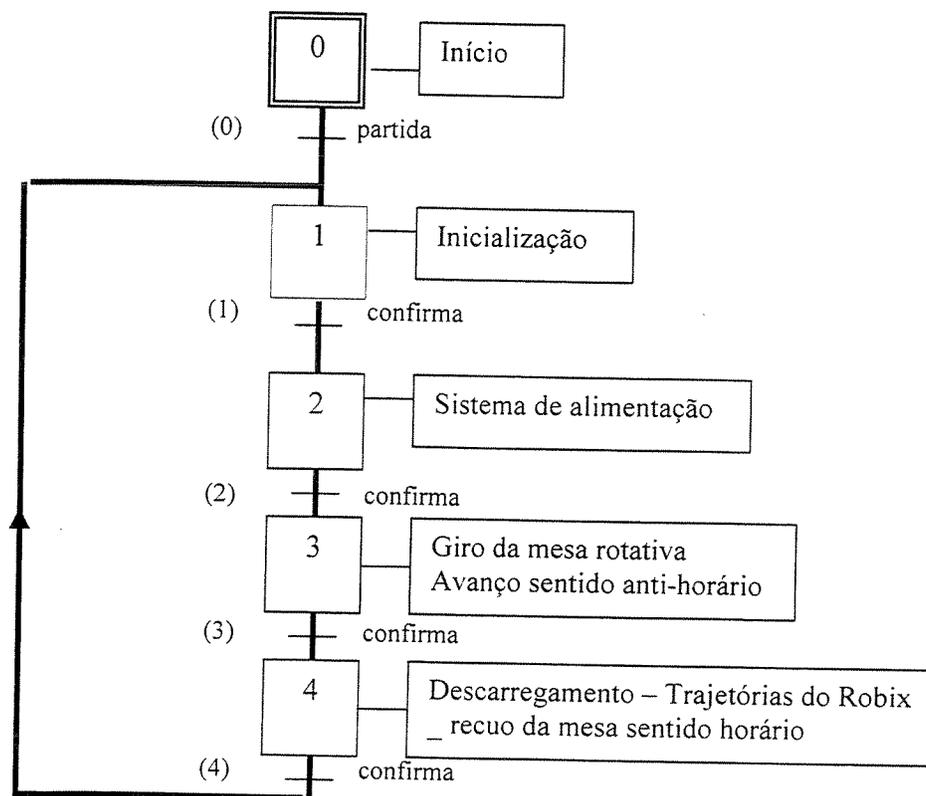


Figura 5.1 – Grafcet Funcional da Plataforma

### 5.1 – Parte Comando ( Pc ) e Parte Operativa ( Po ) dos Postos de Trabalho

Os três postos de trabalho da plataforma foram divididos em parte comando ( Pc ) e parte operativa ( Po ). Na tabela 5.1 são apresentados os postos de trabalho e as respectivas Pc e Po.

Postos de Trabalho	Parte Comando	Parte Operativa
Manipulador Pneumático	<ul style="list-style-type: none"><li>• Sensores</li><li>• CLP</li></ul>	<ul style="list-style-type: none"><li>• Solenóides</li><li>• Atuadores</li><li>• Conjunto Mecânico</li></ul>
Mesa de Transferência	<ul style="list-style-type: none"><li>• Sensor</li><li>• CLP</li></ul>	<ul style="list-style-type: none"><li>• Motor</li><li>• Conjunto Mecânico</li></ul>
Manipulador Robótico	<ul style="list-style-type: none"><li>• CLP</li><li>• PC</li></ul>	<ul style="list-style-type: none"><li>• Servo-motores</li></ul>

Tabela 5.1 – Postos de Trabalho e as Pc / Po

A integração entre Pc e Po dos postos de trabalho é representada pela figura 5.2 e a plataforma projetada e implementada é apresentada na figura 5.3.

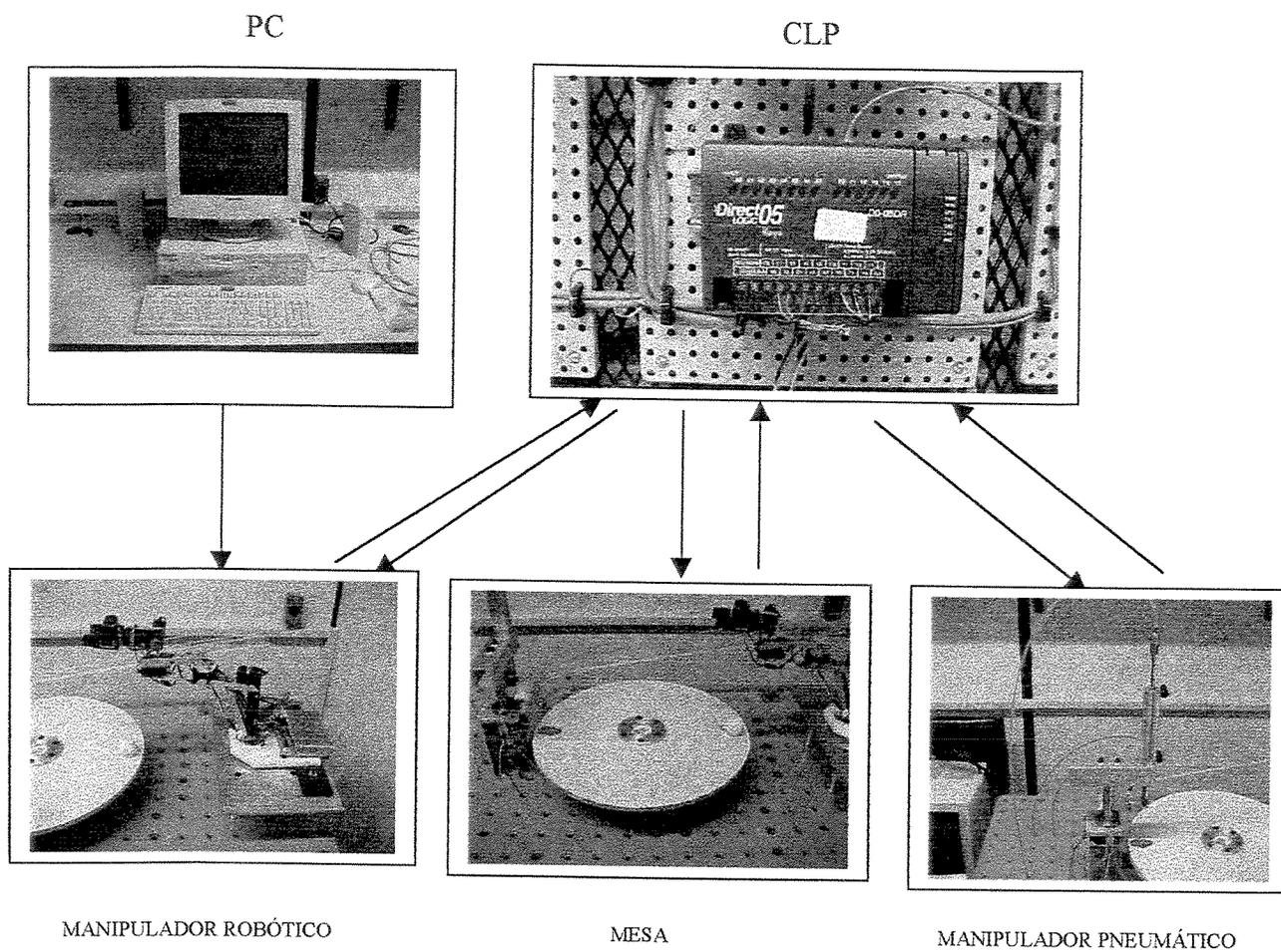
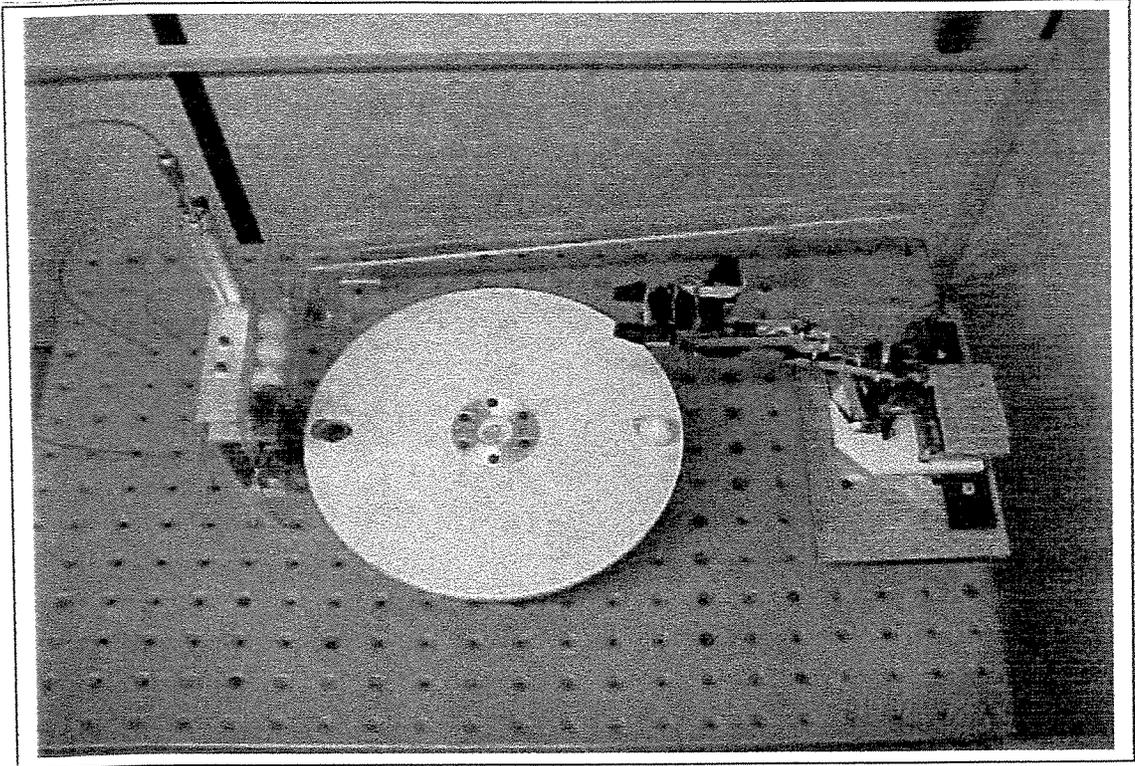


Figura 5.2 – Postos de Trabalho Integrados da Plataforma.



**Figura 5.3– Plataforma Implementada.**

## 5.2 – Estruturação da Plataforma

Para facilitar a programação do CLP, estruturou-se todas as variáveis de controle dos elementos da plataforma. Inicialmente relacionou-se as variáveis de entrada ( tabela 5.2 ) e saída ( tabela 5.3 ) de forma mnemônica e a descrição correspondente de cada variável.

Variáveis	In	Descrição da Função
Bot _ LD	X <sub>0</sub>	Inicialização so sistema
Sens _ Alim	X <sub>1</sub>	Sistema de Alimentação – estoque
Sens _ Prod	X <sub>2</sub>	Presença de produto
Braço _ rot _ rec	X <sub>3</sub>	Braço rotativo recuado
Braço _ rot _ av	X <sub>4</sub>	Braço rotativo avançado
Atuad _ lin _ rec	X <sub>5</sub>	Atuador linear recuado
Atuad _ lin _ av	X <sub>6</sub>	Atuador linear avançado
Sens _ mês _ rot	X <sub>7</sub>	Sensor da mesa rotativa
AuxA	X <sub>10</sub>	Saída do Rodix ( A )
AuxB	X <sub>11</sub>	Saída do Rodix ( B )
Bot _ D	X <sub>12</sub>	Emergência

Tabela 5.2 – Relação das Entradas

Variáveis	Out	Descrição da Função
Alarm _ alim	Y <sub>0</sub>	Alarme de alimentação de produtos
Alarm _ prod	Y <sub>1</sub>	Alarme não presença de produtos
Braço _ rot _ h	Y <sub>2</sub>	Movimento horário do braço rotativo
Braço _ rot _ ah	Y <sub>3</sub>	Movimento anti-horário do braço rotativo
Atuad _ linear	Y <sub>4</sub>	Movimento do atuador linear
Vácuo	Y <sub>5</sub>	Sistema de vácuo ( garra )
Motor _ h	Y <sub>6</sub>	Movimento da mesa no sentido horário
Motor _ ah	Y <sub>7</sub>	Movimento da mesa no sentido anti-horário
Ent _ rob0	Y <sub>10</sub>	Entrada Rodix 0
Ent _ rob1	Y <sub>11</sub>	Entrada Rodix 1
Ent _ rob2	Y <sub>12</sub>	Entrada Rodix 2
Alarm _ Emerg	Y <sub>13</sub>	Alarme

Tabela 5.3 – Relação das Saídas

### 5.2.1 – Modularização da Plataforma

Para integrar todos os postos de trabalho da plataforma , foram criadas macro etapas (  $S_0, S_1, S_2, S_3, S_4, S_5$  ), que podem ser testadas separadas.

O objetivo é modularizar o projeto implementado, flexibilizando-o, para possíveis ampliações e alterações.

As macros são denominadas da seguinte forma :

- $S_0$  – Início
- $S_1$  – Pega \_ produto
- $S_2$  – Depósito \_ produto
- $S_3$  – Devolver \_ produto
- $S_4$  – Emergência
- $S_5$  - Inicialização

A figura 5.4 representa o grafcet funcional das macro-etapas.

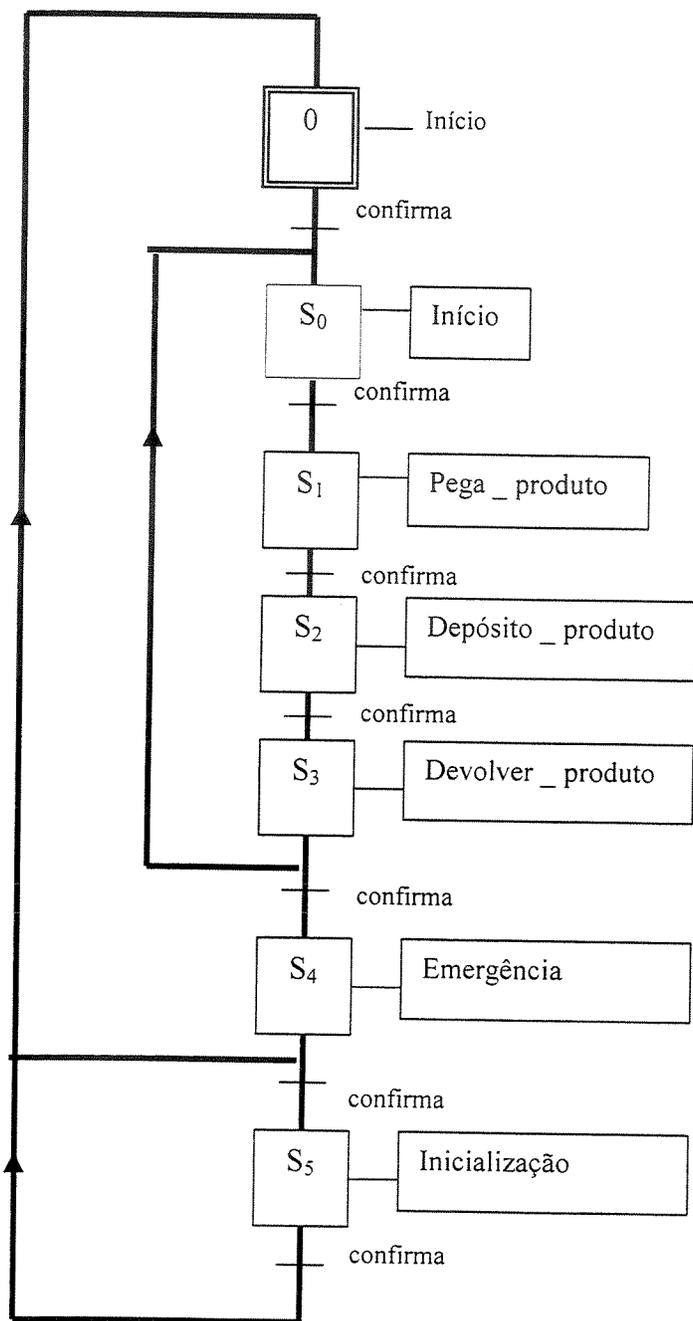
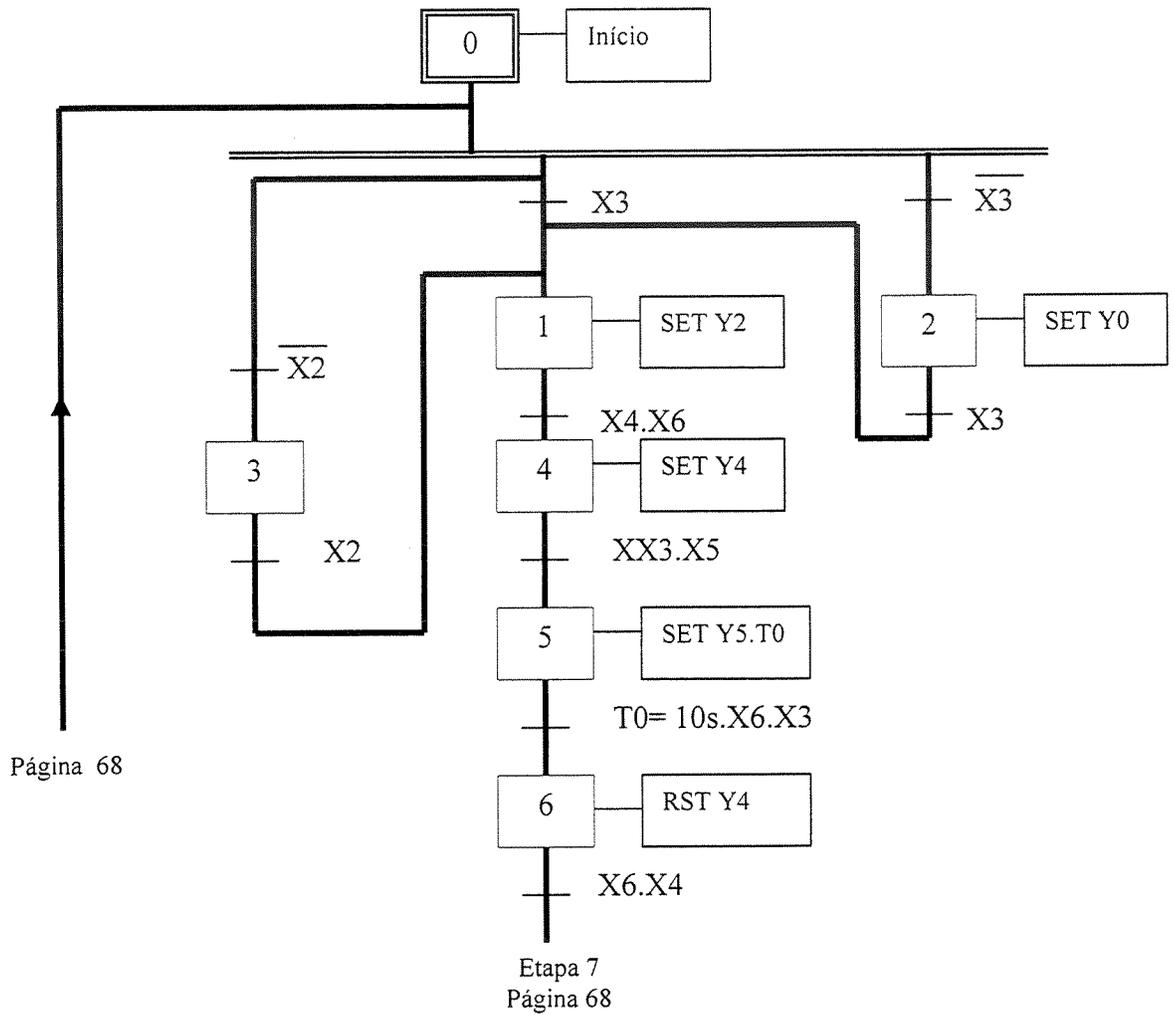


Figura 5.4 – Grafcet Funcional das Macro-Etapas

## 5.2.2 – Grafcet Tecnológico Geral da Plataforma

A figura 5.5 representa o Grafcet Tecnológico incluindo todas as macro-etapas da plataforma.



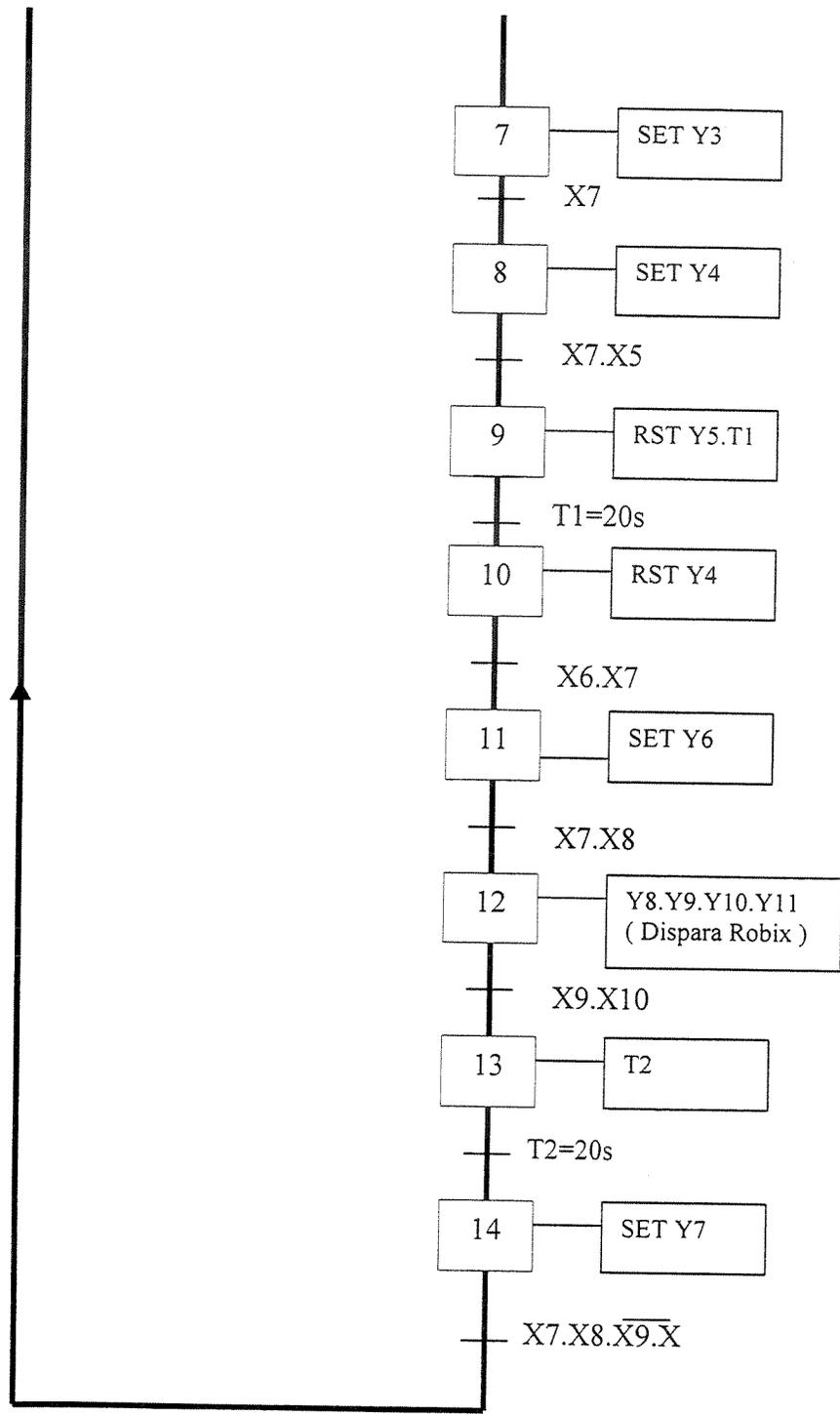


Figura 5.5 – Grafcet Tecnológico Geral de Integração da Plataforma

### **5.3 – Conclusão**

Para que um sistema automatizado seja implementado, o mesmo deve estar muito bem estruturado e documentado, para evitar interpretações ambíguas. Para que a integração de um sistema seja completa é necessário saber a sequência que as etapas devem cumprir. Para isso, deve-se dividir todo o conjunto em subconjuntos e testá-los individualmente. Dessa forma, pode-se detectar falhas com mais rapidez e facilidade.

## **CAPÍTULO 6**

### **Conclusões e Perspectivas para Trabalhos Futuros**

No trabalho apresentado, abordou-se os principais elementos de um sistema automatizado de produção, desde a idealização, descrição e sua implementação. Constatou-se que para sistemas automatizados, mais importante que sua implementação é definir corretamente suas tarefas e documentá-las. Dentre as linguagens existentes para descrever um SAP, adotou-se o Grafcet.

Todas as referências de um sistema automatizado de produção são feitas a partir do projeto e implementação de uma plataforma integrada de manipulação. A plataforma contém uma parte comando representada por botões de emergência e partida, sensores, CLP e microprocessador, e uma parte operativa contendo solenóides, atuadores pneumáticos, motor elétrico e conjuntos mecânicos.

A complexibilidade dos processos industriais automatizados tem crescido de forma sistemática e para acompanhar esta evolução, as redes de comunicação de dados têm um papel cada vez mais importante no apoio aos sistemas de automação e controle. A busca de confiabilidade nas redes de dados em ambientes industriais permite uma maximização na troca de informações, minimiza custos e aumenta a flexibilização das transmissões de dados.

## **6.1 – Sugestões para Continuação deste Trabalho**

- Otimizar o tempo de operação da plataforma com acionamentos simultâneos de seus elementos.
- Implantar um sistema supervisório.
- Implementar um transportador automatizado para alimentação e integrado à plataforma.
- Implementar um sistema de gestão para completar a cadeia produtiva, integrando a produção controle e supervisão.

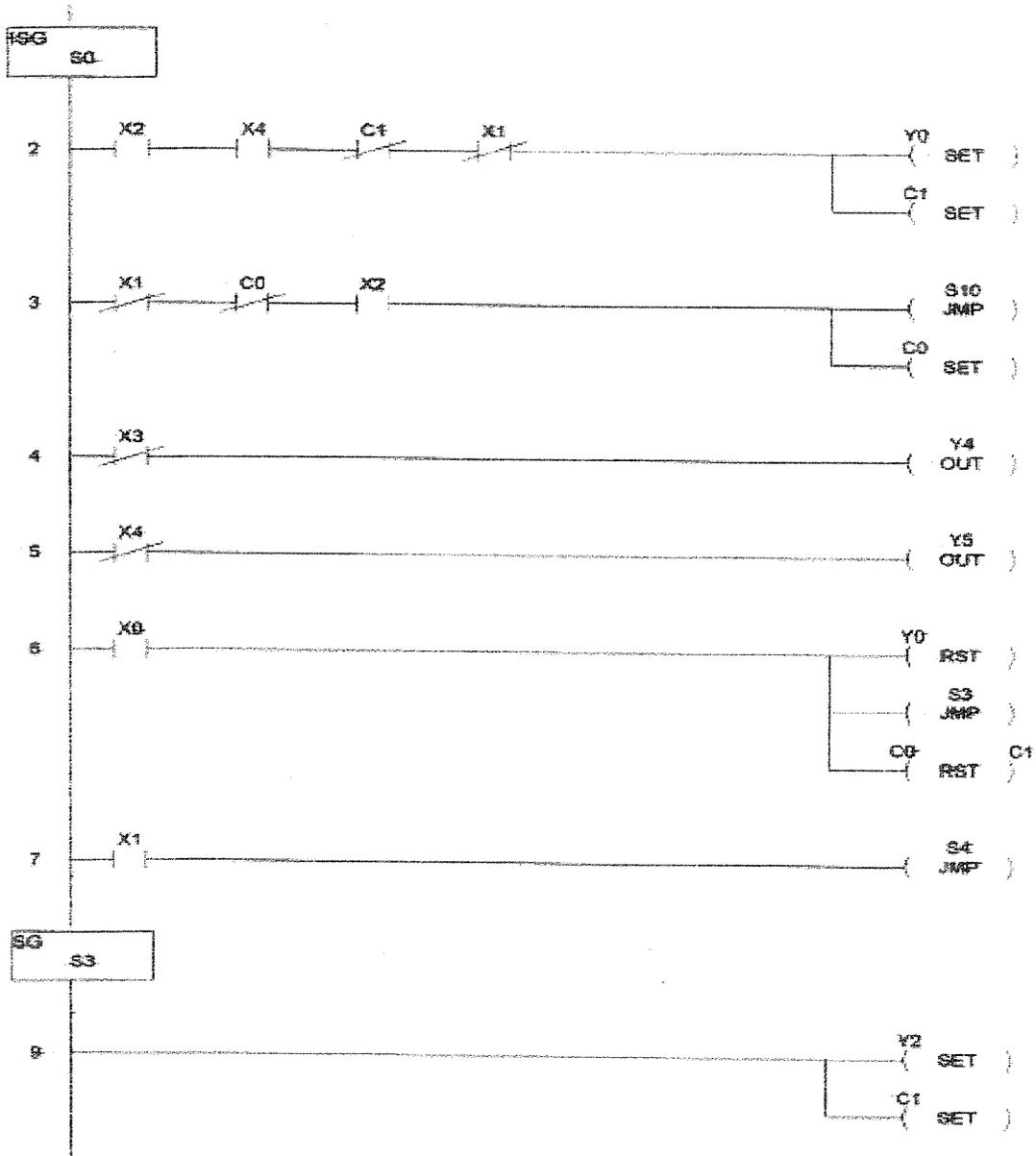
## Referência Bibliográfica

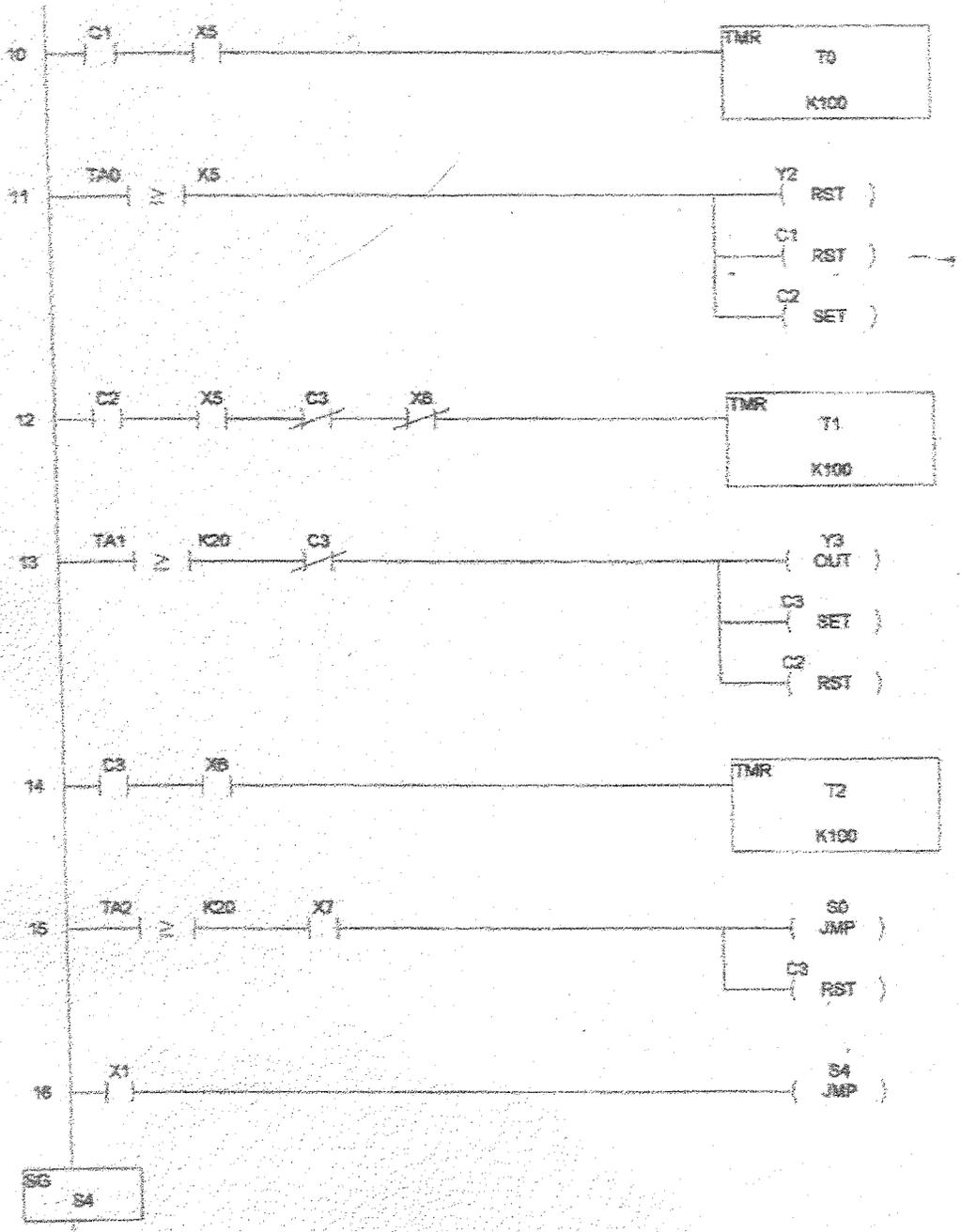
- 1 - Agostinho, Oswaldo Luiz – Integração Estrutural nos Sistemas de Manufatura como Pré-Requisito de Competitividade-Tese de livre docência-Unicamp-1995, 230 p.
- 2 - Bittar, Rita de Cássia da S.M.A utilização do grafcet como ferramenta na automação industrial. Campinas : Faculdade de Engenharia Mecânica, Unicamp, 1993. 143 p. Tese ( mestrado ).
- 3 - Bollinger, John G, Duffie, Neil A. Computer Control of Machines and Processes. 10ª edição. USA : Addison Wesley Publishing Company, 1988.
- 4 - Bollmann, Arno.Fundamentos da automação industrial pneutrônica. ABHP – Associação Brasileira de Hidráulica e Pneumática – 1997 – p. 111-129.
- 5 - Charles Froseer e John Milne – Integreted electrical and eletronic engineering, 1998.
- 6 - Controle & Instrumentação, Automação e Tecnologia da Informação. São Paulo : Valente, n.21, abril, 1998.
- 7 - Dorf C. Richard e Bishop H. Robert, Sistema de Controle Moderno, 8ª edição – LTC – Editora ( Livros Técnicos e Científicos Editora ), 2001.
- 8- Fabricantes de CLP ; <http://www.automacao.net/hardware.htm> -2000.
- 9 - Ferreira, Edson P. , Robótica Básica e Modelagem de Robôs, R. Vieira Gráfica e Editora Ltda., 1991.
- 10 - Georgini, João Marcelo. Elementos para Estruturação e Implementação de Sistemas Automatizados de Produção. Campinas : Faculdade de Engenharia Mecânica, Unicamp, 1999. 209 p. Tese ( mestrado ).

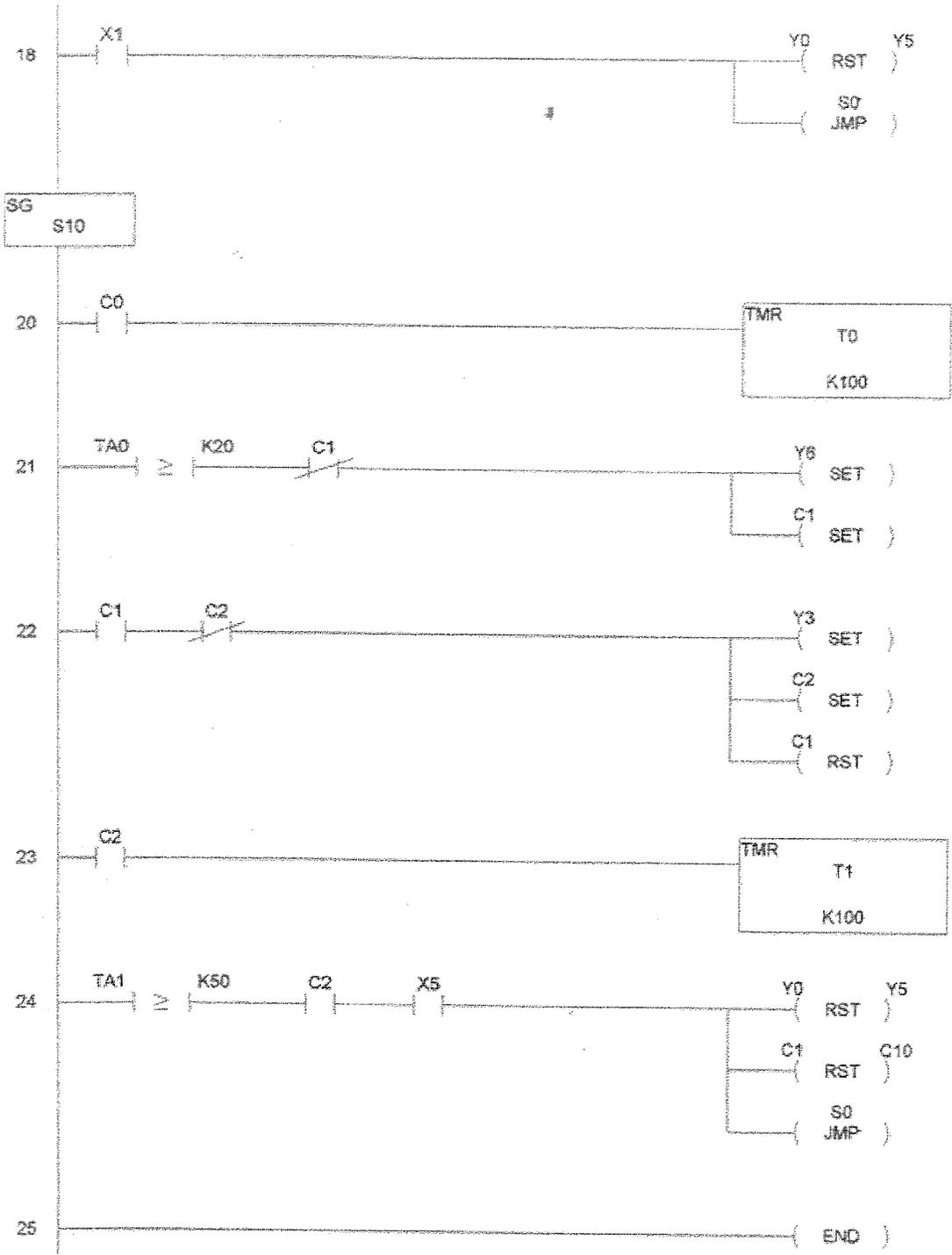
- 11 - In Tech Brasil. Automação e controle via internet,nº 22, março, 2000.
- 12 - Natale, Ferdinando. Automação Industrial . 1ª edição, São Paulo : Editora Érica, 1995.
- 13 - Silveira, Paulo R. e Santos, Winderson, E. Automação e controle discreto. Editora Érica, 1998.
- 14 - Vendrameto, Oduvaldo. Bases de conhecimento para a automação de manufatura. São Paulo : Escola Politécnica / USP – São Paulo, 1994. 155 p. Tese ( doutorado ).
- 15 - Wolovich, William A. Automatic Control Systems – Basic Analysis and Design. USA : International Edition, 1994, 450 p.

# ANEXO I

## Programação em Ladder para CLP<sub>1</sub> e CLP<sub>2</sub>.

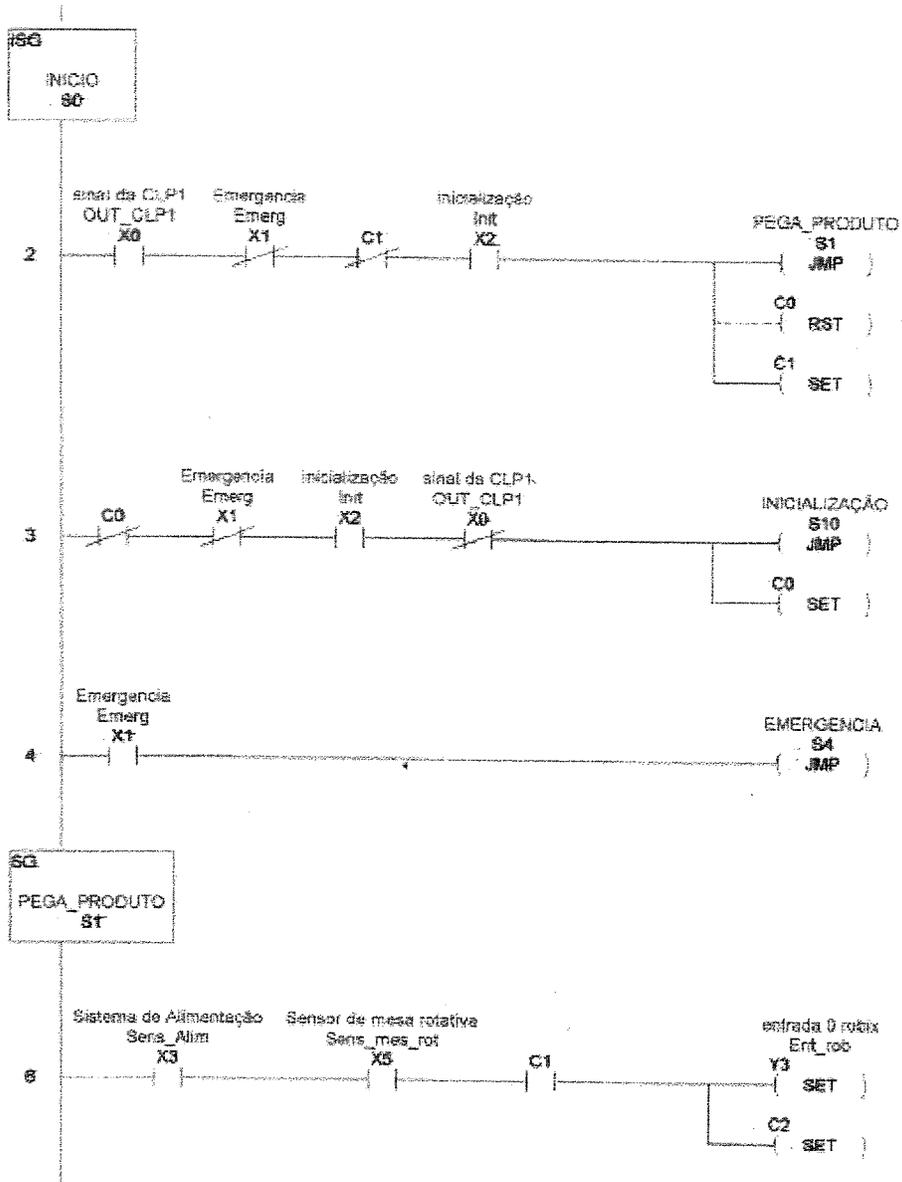


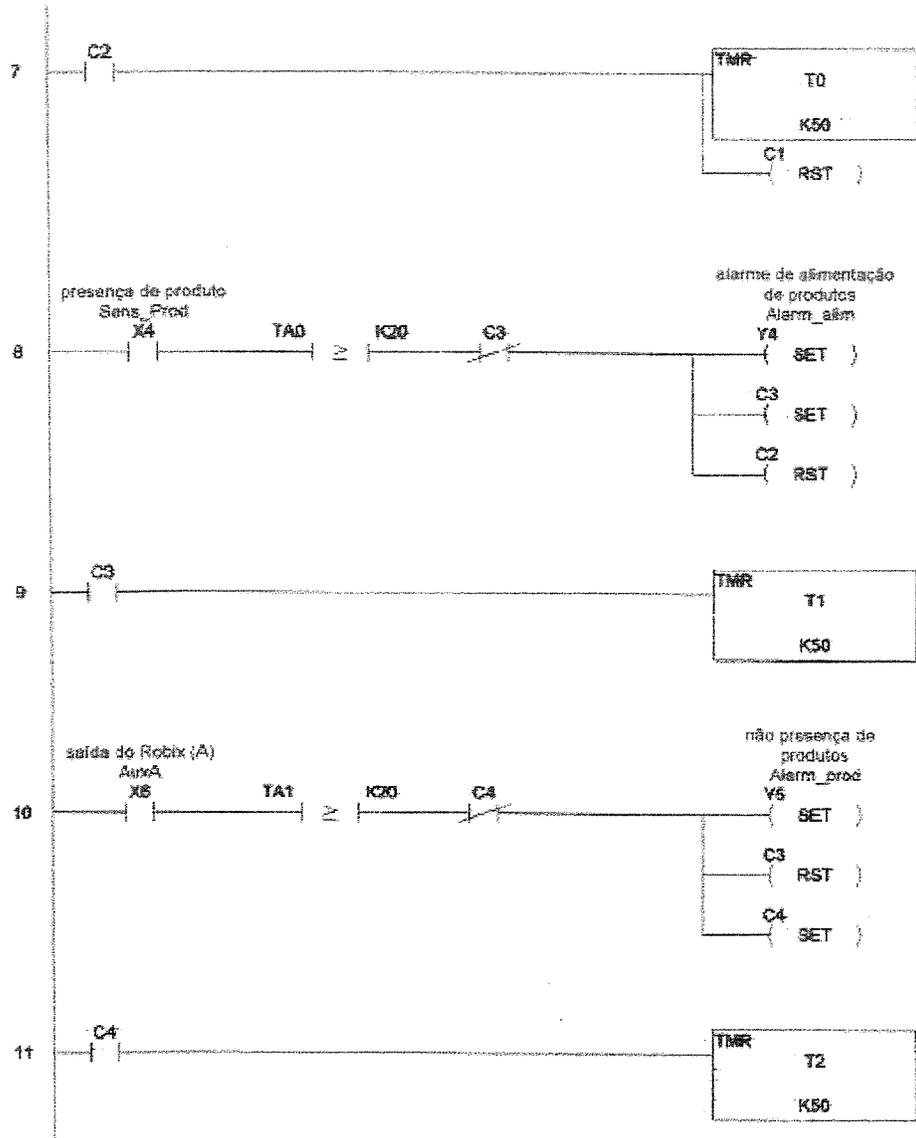


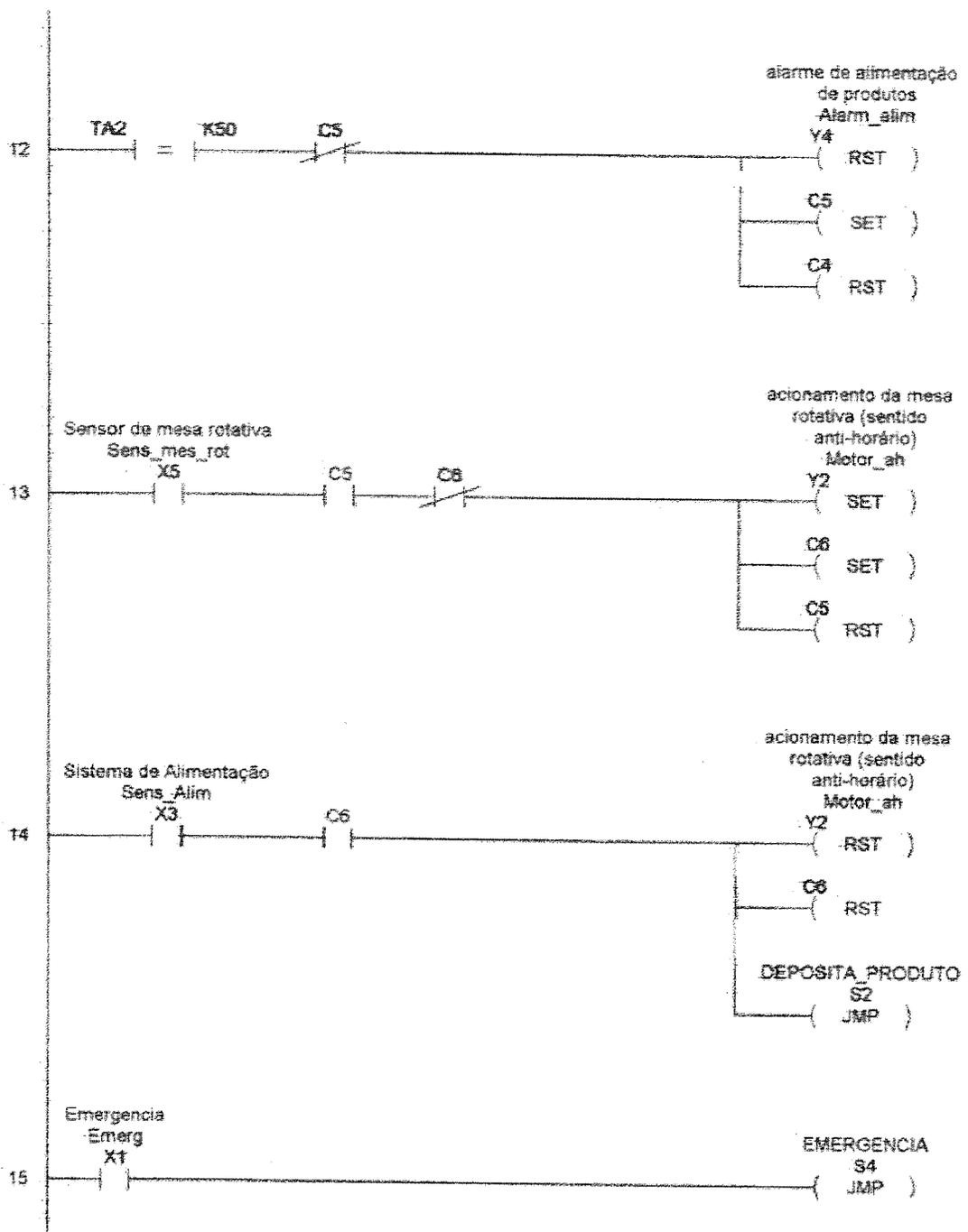


28

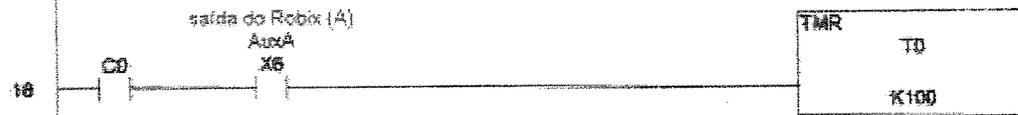
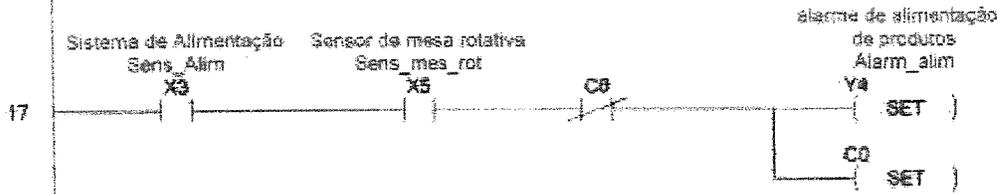
( NOP )

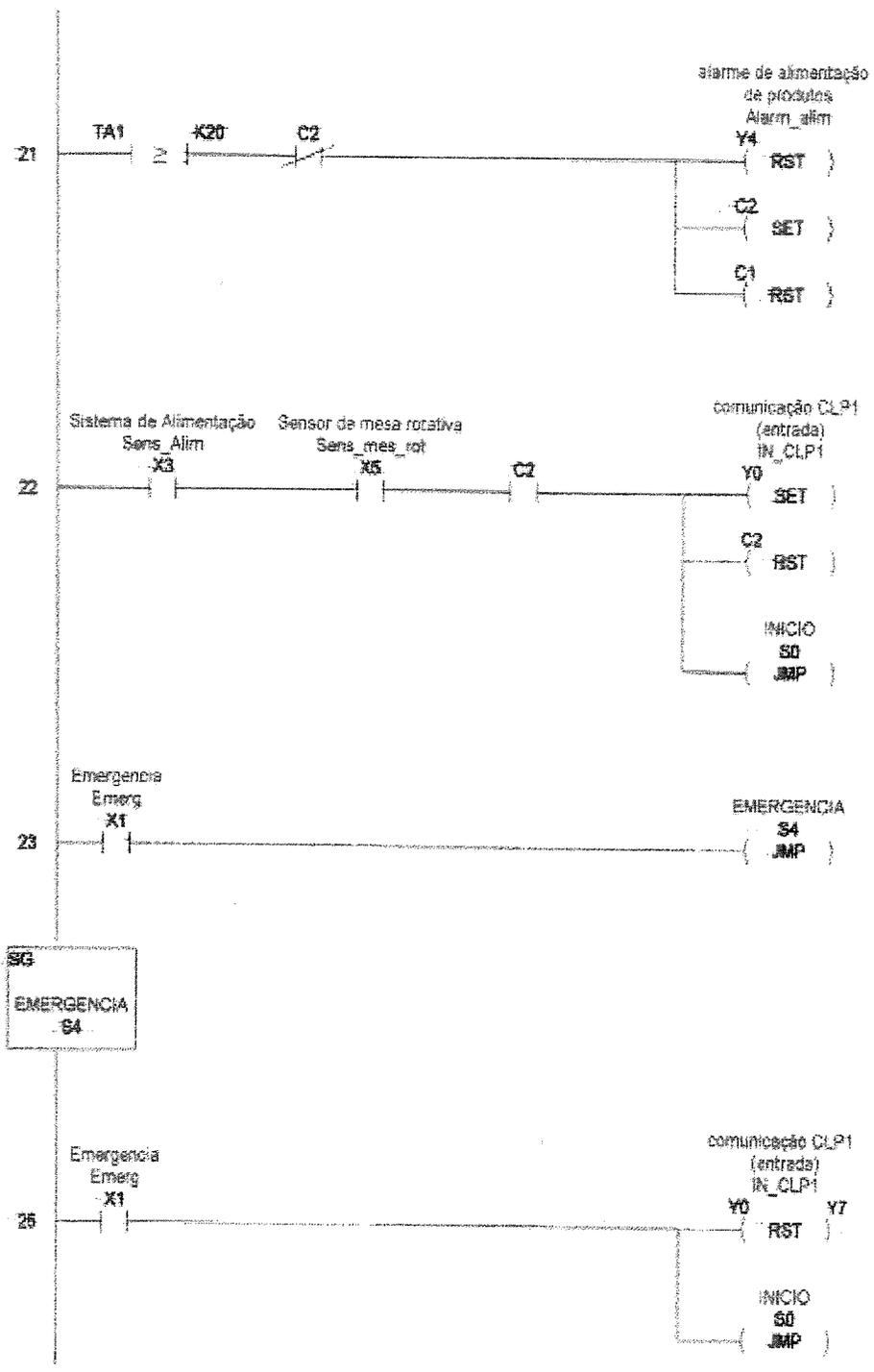




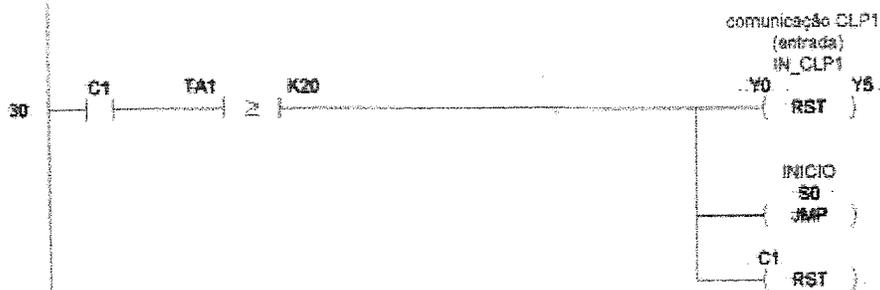
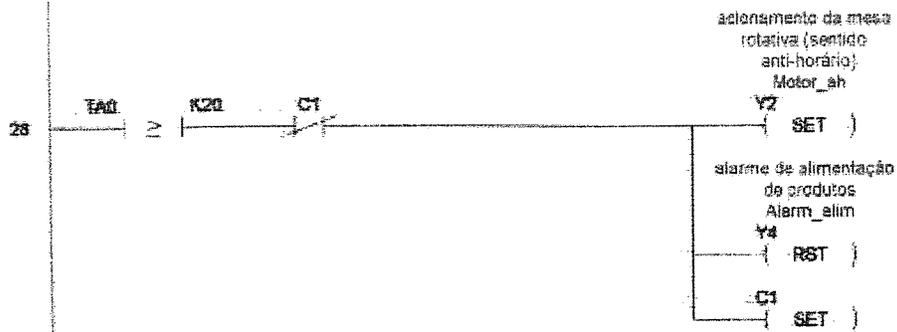


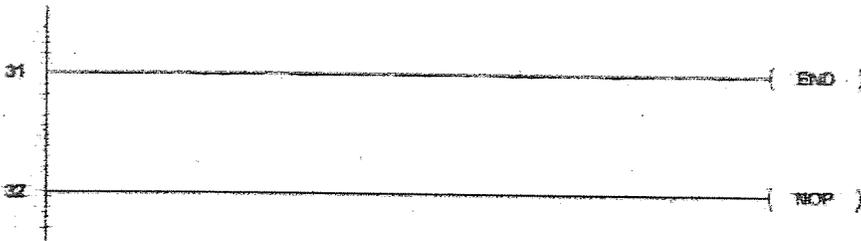
SG  
DEPOSITA\_PRODUTO  
S2





SG  
INICIALIZAÇÃO  
S10





## ANEXO II

O programa a seguir é utilizado para comunicação do CLP e o manipulador robótico. Trata-se de uma linguagem de alto nível ( C ).

```
/* file prog_rob.c, build with makecdem.bat, uses prog_rob.h, prog_rob.inc */
/* Sample command line interpreter for scripting language.
/* Demonstrates use of many functions from rbx?.lib's */

#include "prg1.h"
#include "Luizleo.h"

/* increments for jog commands */
#define LARGE_INC 30
#define SMALL_INC 3

static char acCommandBuf[RBX_COMMAND_LEN_MAX+1];

/* get the literals tables; this is a convenient C technique */
/* to keep literals to in an easy-to-edit table, while */
/* using the "names" of the literals in the code. */
/* The technique is general and is not unique to this program; */
/* it is useful during preparation of multi-lingual apps. */

enum
{
#define MAC(a,b) a,
#include "prg1.inc"
#undef MAC
};

char *aszLits[] =
{
#define MAC(a,b) b,
#include "prg1.inc"
#undef MAC
};

/* wrapper around rbxSaveCfg call to save configuration file */
static void SaveConfig(char *szFspec)
{
switch(rbxSaveCfg(szFspec))
{
case 0:
printf("%s: %s\n", aszLits[CfgSavedLit],
aszLits[DefaultCfgFnameLit]);
return;

case 1: goto bad_open;
case 2: goto bad_close_write;
}
```

```

    }

    bad_close_write:
        unlink(szFspec);

    bad_open:
        printf("%s: %s\n", aszLits[FileWriteErrLit], szFspec);
    }

    /* noise to make when no action */
    void NoActionBeep(void)
    {
        sound(100);
        delay(200);
        nosound();
    }

    void OutOfRangeBeep(void)
    {
        sound(2000);
        delay(200);
        nosound();
    }

    void EnterBeep(void)
    {
        sound(2000);
        delay(200);
        nosound();
    }

    void ErrorBeep(void)
    {
        sound(2000);
        delay(200);
        nosound();
    }

    void Stat(void)
    {
        short sEBase = rbxEnumBaseR();
        short si, sStat;

        static short *psMotorsParam = NULL; /* for clarity; statics are anyway */

        /* could also just declare the array with RBX_MOTOR_PARAM_COUNT_MAX */
        if (!psMotorsParam)
            psMotorsParam = malloc(sizeof(short) * rbxMotorCount);

        printf("%s ", rbxMotorTtl);

        for (si = sEBase; si < rbxMotorCount + sEBase; si++)
            printf("%*d", RBX_READING_FMT_LEN+1, si);
    }

```

```

printf("\n");

for (si = sEBase; si < rbxMotorParamCount + sEBase; si++)
{
    short si1;

    printf("%s ", rbxMotorParamTtl[si-sEBase /* it's a C-array */]);
    rbxMotorsParamR(si, psMotorsParam);
    for (si1 = 0 /* not sEBase, it's a C-ary */; si1 < rbxMotorCount; si1++)
        printf("%*d", RBX_READING_FMT_LEN+1, psMotorsParam[si1]);
    printf("\n");
}
printf("\n");

#if 0
/* alternate, less "automatic" approach */
printf("%s\n", aszLits[ShowTitleLit]);

for (si = sEBase; si < rbxParameterCount + sEBase; si++)

for (si = sEBase; si < rbxMotorCount + sEBase; si++)
{
    short sPos, sMn, sMx, sInit, sAcc, sDec, sSpd, sInvert, sPwr, sPin;

    sPos = rbxPosR(si);
    sInit = rbxInitPosR(si);
    sMn = rbxMinPosR(si);
    sMx = rbxMaxPosR(si);
    sAcc = rbxAccelR(si);
    sDec = rbxDecelR(si);
    sSpd = rbxMaxSpdR(si);
    sInvert = rbxInvertR(si);
    sPwr = rbxPowerR(si);
    sPin = rbxPinnedR(si);

    printf(" %2d %5d %4d %4d %4d  %d %5d %5d %5d %5d %5d\n",
        si, sPos, sSpd, sAcc, sDec, sPwr, sMn, sMx, sInit, sInvert, sPin);
}
printf("\n");
#endif

sStat = rbxStatusR();

printf("statuses:\n"
    "EnumBase: %d  "
    "Compiling: %c  "
    "Running: %c  "
    "Ready: %c\n"
    "Continued: %c  "
    "CmdWasTxt: %c  "
    "AuxA: %c  "
    "AuxB: %c\n",

    rbxEnumBaseR(),

```

```

(sStat & RBX_COMPILING_STAT) ? 'X' : '-',
(sStat & RBX_RUNNING_STAT) ? 'X' : '-',
(sStat & RBX_READY_STAT) ? 'X' : '-',
(sStat & RBX_CONTINUED_STAT) ? 'X' : '-',
(sStat & RBX_CMD_WAS_TEXT_STAT) ? 'X' : '-',
(sStat & RBX_AUX_A_STAT) ? 'X' : '-',
(sStat & RBX_AUX_B_STAT) ? 'X' : '-'
);

printf("\n");

/* read switches and adc's */
{
    short asAdc[8];
    short asSwitch[8];
    short si;

    printf(rbxSensorTtl);
    rbxAdcR(asAdc);
    for (si = 0; si < rbxAdcCount; si++)
        printf("%6d", asAdc[si]);

    rbxSwitchesR(asSwitch);
    printf(" ");
    for (si = 0; si < rbxSwitchCount; si++)
        printf("%c", asSwitch[si] ? 'X' : '-');
}

printf("\n\n");
}

/* Jog a motor. Returns 0 if character not recognized; 1 if processed; */
short Jog(int iCh)
{
    //printf("jogging"); //LeoSoft
    short siM, slnc;
    char *pc;

    if (!iCh)
    {
        iCh = getch();
        goto extended;
    }

    iCh = (char)toupper(iCh);

    if ((pc = strchr(aszLits[JogLargePlusLit], iCh)) != NULL)
    {
        siM = (short)(pc - aszLits[JogLargePlusLit]);
        slnc = LARGE_INC;
        goto movit;
    }

    if ((pc = strchr(aszLits[JogLargeMinusLit], iCh)) != NULL)
    {

```

```

    siM = (short)(pc - aszLits[JogLargeMinusLit]);
    sInc = -LARGE_INC;
    goto movit;
}

if ((pc = strchr(aszLits[JogSmallPlusLit], iCh)) != NULL)
{
    siM = (short)(pc - aszLits[JogSmallPlusLit]);
    sInc = SMALL_INC;
    goto movit;
}

if ((pc = strchr(aszLits[JogSmallMinusLit], iCh)) != NULL)
{
    siM = (short)(pc - aszLits[JogSmallMinusLit]);
    sInc = -SMALL_INC;
    goto movit;
}

return 0;

movit:
if (siM > rbxMotorCount - 1)
    return 0;

rbxJumpRel(siM + rbxEnumBaseR(), sInc);
printf("jogging");    //LeoSoft

if (rbxPinnedR(siM))
    OutOfRangeBeep();

return 1;

extended:

if (iCh >= 16 && iCh <= 24)
{
    short sPwr;

    siM = iCh - 16;
    if (siM > rbxMotorCount - 1)
        return 0;

    /* toggle power */
    rbxPowerW(siM, !rbxPowerR(siM));

    printf("%s %d: %s %d\n",
        rbxMotorTtl, siM+1,
        rbxMotorParamTtl[rbxPowerMotorParamNdx], sPwr);
    return 1;
}

return 0;
}

```

```

void ReportError(void)
{
    char *tkn;

    if (rbxCmdWasTextR())
    {
        printf("%s\n%*c\n", rbxLastCmdR(), rbxErrColR(), '^');
    }

    printf("%s", rbxErrMsgR());
    tkn = rbxErrTokenR();
    if (*tkn)
        printf(" -- \"%s\"", tkn);
    printf("\n");
}

```

```

static void DirectControl(void)
{
    printf("%s\n\n", aszLits[DirectPromptLit]);

    for (;;)
    {
        int ch;
        switch(ch = getch())
        {
            case '^': Stat(); break;
            case '\x1b': goto done; /* escape */
            default:
                if (!Jog(ch))
                {
                    NoActionBeep();
                    printf("%s\n", aszLits[DirectPromptLit]);
                }
                break;
        }
    }
done:
    ;
}

```

```

static void Help(void)
{
    short sEBase = rbxEnumBaseR();
    short si, scltems;

    /* macros */
    scltems = rbxMacroCountR();

    printf("\n\n%s:\n", aszLits[DrvMacrosTitleLit]);
    for (si = sEBase; si < scltems + sEBase; si++)
        printf("%-20s", rbxMacroNameR(si));

    if (!scltems)
        printf(aszLits[NoMacrosLit]);
}

```

```

    /* resident primitive commands */
    scItems = rbxPrimitiveCount;

    printf("\n\n%s:\n", aszLits[DrvPrimitivesTitleLit]);

    for (si = sEBase; si < scItems + sEBase; si++)
        printf("%-20s", rbxPrimitiveNameR(si));

    /* resident keywords */
    scItems = rbxKeywordCount;
    printf("\n\n%s:\n", aszLits[DrvKeywordsTitleLit]);

    for (si = sEBase; si < scItems + sEBase; si++)
        printf("%-20s", rbxKeywordR(si));

    /* console commands */
    printf("\n\n%s:\n", aszLits[ConCommandsTitleLit]);
    for (si = CommandBeginsLit + 1; si < CommandEndsLit; si++)
        printf("%-20s", aszLits[si]);

    done:
    printf("\n\n");
}

static void GetCommands(void)
{
    char acXBuf[RBX_COMMAND_LEN_MAX+1]; /* extra buffer for strtok'ing */
    char *npos;

    printf("\n\n%s\n\n", aszLits[TypeHelpForHelpLit]);

    for (;;)
    {
        char *pcPrompt;
        ushort usStat;
        char *pcTkn;
        short scTkns;

        usStat = rbxStatusR();

        if (usStat & RBX_COMPILING_STAT)
        {
            if (usStat & RBX_CONTINUED_STAT)
                pcPrompt = aszLits[CompPlusPromptLit];
            else
                pcPrompt = aszLits[CompPromptLit];
            printf("%3d %s ", rbxPartialMacroSizeR(), pcPrompt);
        }
        else
        {
            if (usStat & RBX_CONTINUED_STAT)
                pcPrompt = aszLits[ExecPlusPromptLit];
            else
        }
    }
}

```

```

    pcPrompt = aszLits[ExecPromptLit];
    printf("%s ", pcPrompt);
}

fgets(acCommandBuf, sizeof(acCommandBuf), stdin);
if ((npos = strchr(acCommandBuf, '\n')) != NULL)
    *npos = '\0';

strcpy(acXBuf, acCommandBuf);

scTkns = 0;
if ((pcTkn = strtok(acXBuf, " ")) != NULL)
{
    ++scTkns;
    while (strtok(NULL, " "))
        ++scTkns;
}

/* note that there is no explicit check for a blank line: */
/* blank lines fall through to the rbx call. */

if (scTkns == 1)
{
    if (!stricmp(pcTkn, aszLits[ExitLit]))
        return;

    if (!stricmp(pcTkn, aszLits[SaveConfigLit]))
    {
        SaveConfig(aszLits[DefaultCfgFnameLit]);
        continue;
    }

    if (!stricmp(pcTkn, aszLits[StatusLit]))
    {
        Stat();
        continue;
    }

    if (!stricmp(pcTkn, aszLits[MemleftLit]))
    {
        printf("%d\n", rbxMemleftR());
        continue;
    }

    if (!stricmp(pcTkn, aszLits[HelpLit]))
    {
        Help();
        continue;
    }

    if (!stricmp(pcTkn, aszLits[DirectLit]))
    {
        DirectControl();
        continue;
    }
}
}

```

```

rbxCmd(acCommandBuf);

if (rbxErrNo)
{
    ReportError();
    ErrorBeep();
}
continue;
}
}

void main(void)
{
    short lsensor1;          //LeoSoft Estado do Sensor 1
    short lsensor2;          //LeoSoft Estado do Sensor 2
    short lsensor3;          //LeoSoft Estado do Sensor 3
    short lsensor4;          //LeoSoft Estado do Sensor 4
    short lsensor5;          //LeoSoft Estado do Sensor 5
    short lsensor6;          //LeoSoft Estado do Sensor 6
    short lsensor7;          //LeoSoft Estado do Sensor 7

    short sInitErr;
    switch (sInitErr = rbxAttach())

    {
        case 0:
            break;

        case 1:
            printf("%s\n", aszLits[DrvNotInstalledLit]);
            exit(1);

        default:
            printf("rbx_Attach error [%d], not recognized", sInitErr);
            exit(1);
    }

    printf("%s: %s...", aszLits[CfgLoadingLit], aszLits[DefaultCfgFnameLit]);
    switch (rbxLoadCfg(aszLits[DefaultCfgFnameLit]))
    {
        case 1:
            printf("%s: %s\n", aszLits[FileLoadErrLit],
                aszLits[DefaultCfgFnameLit]);
            printf("%s: %d\n", aszLits[LineLit], rbxErrLine);
            ReportError();
            break;

        case 2:
            printf("%s: %s\n", aszLits[CfgNotFoundLit],
                aszLits[DefaultCfgFnameLit]);
            break;

        default: /* went ok */
    }
}

```

```

    printf("%s\n", aszLits[DoneLit]);
    break;
}

rbxRestart(); /* reprogram Adapter in case power has been interrupted */

printf("%s: %s...",
    aszLits[StartupLoadingLit], aszLits[DefaultStartupFnameLit]);
switch (rbxLoad(aszLits[DefaultStartupFnameLit]))
{
    case 1:
        printf("%s: %s\n", aszLits[FileLoadErrLit],
            aszLits[DefaultStartupFnameLit]);
        printf("%s: %d\n", aszLits[LineLit], rbxErrLine);
        ReportError();
        break;

    case 2:
        printf("%s: %s\n", aszLits[StartupNotFoundLit],
            aszLits[DefaultStartupFnameLit]);
        break;

    default: /* went ok */
        printf("%s\n", aszLits[DoneLit]);
        break;
}

while(!kbhit())          // LeoSoft leitura dos Sensores e comparacao
{
    int valor;
    for (valor = 0; valor < 100; valor++)
    {
        delay (1);
        lsensor1=rbxSwitchR(1); //Leitura do sensor 1 (traj1)
        lsensor2=rbxSwitchR(2); //Leitura do sensor 2 (traj2)
        lsensor3=rbxSwitchR(3); //Leitura do sensor 3
        lsensor4=rbxSwitchR(4); //Leitura do sensor 4
        lsensor5=rbxSwitchR(5); //Leitura do sensor 5
        lsensor6=rbxSwitchR(6); //Leitura do sensor 6 (retorno)
        lsensor7=rbxSwitchR(7); //Leitura do sensor 7 (fim)
    }
    if (lsensor1==1 && lsensor2==1 && lsensor3==1)
    {
        printf ("\n posicao de emergencia");
        /* printf ("\n 111 --> traj11" ); */
        rbxLoad("TRAJ11.RBX");
    }

    if (lsensor1==1 && lsensor2==1 && lsensor3==0)
    {
        printf ("\n trajetoria pegar recipiente de mistura");
        /* printf ("\n 110 --> traj12" ); */
        rbxLoad("TRAJ12.RBX");
    }
    if (lsensor1==1 && lsensor2==0 && lsensor3==1)
    {

```

```

printf ("\n trajetoria devolver o recipiente de mistura");
/* printf ("\n 101 --> traj13" ); */
rbxLoad("TRAJ13.RBX");
}

if (lsensor1==0 && lsensor2==1 && lsensor3==1)
{
printf ("\n trajetoria cor1");
/* printf ("\n 011 --> traj14" ); */
rbxLoad("TRAJ14.RBX");
}

if (lsensor1==0 && lsensor2==0 && lsensor3==1)
{
printf ("\n trajetoria cor2");
/* printf ("\n 001 --> traj15" ); */
rbxLoad("TRAJ15.RBX");
}

if (lsensor1==0 && lsensor2==1 && lsensor3==0)
{
printf ("\n trajetoria cor3");
/* printf ("\n 010 --> traj16" ); */
rbxLoad("TRAJ16.RBX");
}

if (lsensor1==1 && lsensor2==0 && lsensor3==0)
{
printf ("\n trajetoria cor4");
/* printf ("\n 100 --> traj17" ); */
rbxLoad("TRAJ17.RBX");
}

if (lsensor4==1)
{
printf ("\n Sensor 4");
}

if (lsensor5==1)
{
printf ("\n Sensor 5");
}

if (lsensor6==1)
{
printf ("\n Sensor 6");
}

if (lsensor7==1)
{
printf ("\n Sensor 7");
}
}
}
}

```

### ANEXO III

#### Programação das Trajetórias do Manipulador Robótico - Robix™

```
forget all
macro inicio
move 1 to -1051, 2 to 366, 3 to 127, 4 to -9, 5 to -1382;
end
macro aproxima_bola
move 3 to 368, 4 to -1400;
end
macro pega_bola
move 1 to -1051, 3 to 839, 4 to -875;
end
macro aprox_canaleta
move 1 to -148, 3 to 127, 4 to -9;
end
macro deixa_canaleta
move 3 to 913, 4 to 747;
end
macro abre_garra
move 6 to 1120;
end
macro segura
move 6 to 520;
end
macro fecha_garra
move 6 to -1400;
end
macro executa
auxa on;
auxb off;
fecha_garra;
inicio;
aproxima_bola;
abre_garra;
pega_bola;
segura;
aproxima_bola;
inicio;
aprox_canaleta;
deixa_canaleta;
abre_garra;
aprox_canaleta;
fecha_garra;
inicio;
auxa off;
auxb on;
end
executa
```

```
forget all
macro executa
auxa on;
auxb off;
move 6 to -1400;
move 1 to -1051, 2 to 366, 3 to 127, 4 to -9, 5 to -1382;
move 3 to 368, 4 to -1400;
move 6 to 1120;
move 1 to -1051, 3 to 839, 4 to -875;
move 6 to 520;
move 3 to 368, 4 to -1400;
move 1 to -1051, 2 to 366, 3 to 127, 4 to -9, 5 to -1382;
move 1 to -148, 3 to 127, 4 to -9;
move 3 to 913, 4 to 747;
move 6 to 1120;
move 1 to -148, 3 to 127, 4 to -9;
move 6 to -1400;
move 1 to -1051, 2 to 366, 3 to 127, 4 to -9, 5 to -1382;
auxa off;
auxb on;
end
executa
```

## **ANEXO IV**

### **Sistemas Robóticos Aplicados em Automação Industrial**

A robótica vem despertando interesse dos pesquisadores desde o início do século passado ( 1920 ), quando surgiram as primeiras aplicações com objetivo de substituir algumas atividades do homem. Essas atividades estão relacionadas com força, velocidade, precisão ou em atividades que oferecem riscos a saúde do homem, como temperatura ( alta ou baixa ), ambientes tóxicos, etc.

A robótica compreende uma área de aplicação bem ampla e multidisciplinar, envolvendo conhecimento nas áreas de mecânica, eletrônica, informática, teorias de otimização e controle e de métodos matemáticos. Os aspectos científicos e tecnológicos têm sido alvo das pesquisas, porém a robótica gera uma expectativa junto ao leigo, muito além de sua real condição de realizar trabalho industrial.

A caracterização dos robôs está associada a tecnologia de manipuladores, por estarem sempre integradas a processos produtivos. Esses processos podem ser centros de usinagem, como máquinas-ferramentas CNC ou em linhas de montagem em geral, atuando no carregamento e descarregamento de produtos.

Cada bloco do diagrama ( fig. A4.1 ) representa uma parte de toda organização funcional de um robô, onde os fluxos de informações são passadas pela inter-relação entre os blocos.

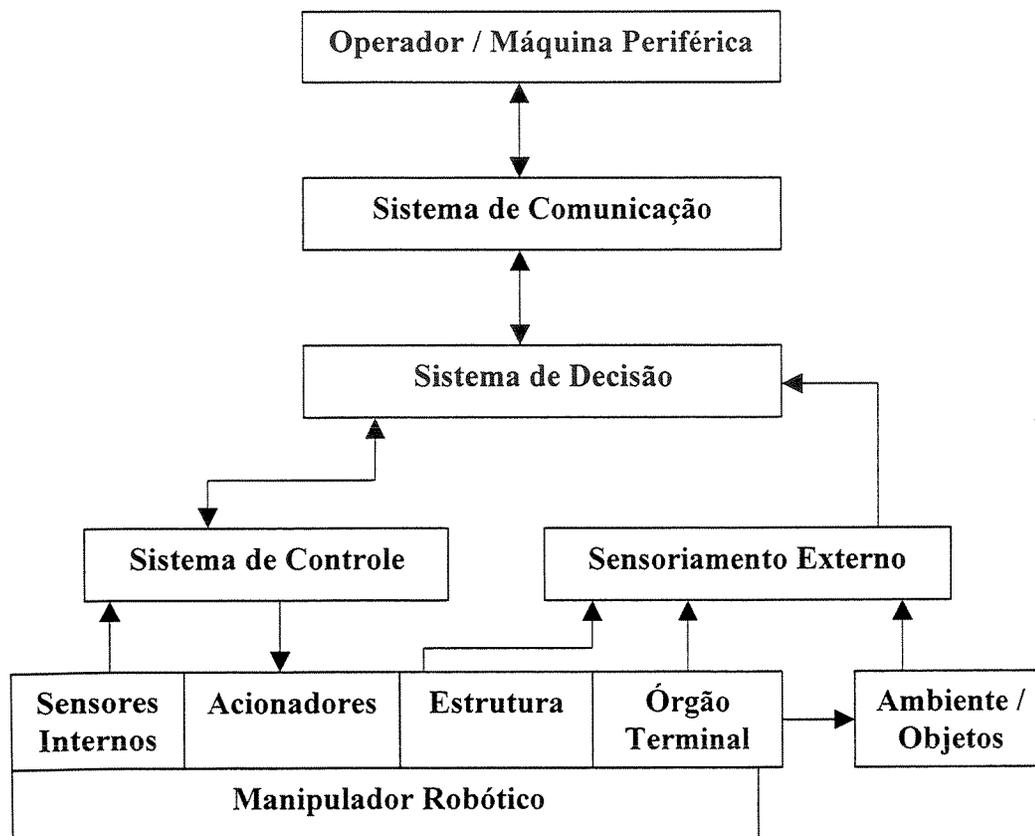


Fig. A4.1 – Organização Funcional de um Robô

A robótica tornou-se fundamental para evolução dos meios de produção, essencialmente pela versatilidade que a mesma proporciona. Os robôs podem ser programados e isto permite uma flexibilidade grande, tornando-os adaptáveis a várias tarefas com rapidez e precisão.

A robótica tem um amplo domínio de aplicações tais como : industrial, médica, espacial e submarina. A robótica integra as disciplinas necessárias ao desenvolvimento de máquinas com estágios de evolução avançados, na maioria dos casos repetindo, especializando ou substituindo algumas funções humanas. Desta forma, a robótica procura dotar as máquinas das seguintes funções :

- Comunicação, para integração, fazendo a interface homem-máquina ou máquina-máquina;
- Percepção sensorial;
- Inteligência – poder de decisão e controle;
- Manipulação, deslocamento autônomo e ações mecânicas, interagindo com o ambiente.

Ao considerar as funções anteriores dos robôs, pode-se dividi-los em dois grupos, quanto à utilização :

❖ Robôs de exploração – As aplicações ficam voltadas para ambientes hostis, como alta temperatura, profundidades elevadas, etc. Estes robôs precisam de um grau de independência elevado, com maior número de funções motoras, usando softwares complexos, dotando-os de uma inteligência artificial para maior autonomia nas decisões. Esses robôs ainda estão em fase de desenvolvimento e pesquisa cuja utilização ainda é muito específica.

❖ Robôs de Produção – No âmbito industrial predominam as ações de manipulação, sendo que a necessidade de ações autômatas e inteligentes é quase que inexistente. Neste contexto, praticamente os robôs industriais em atividades são manipuladores, com atuação muito limitada de percepção e decisão.

Os robôs de produção para atuação em sistemas mais complexos, como os sistemas flexíveis de manufatura ( FMS ) e sistemas integrados por computadores ( CIM ), precisam ser dotados de sensores sofisticados de percepção para que haja uma comunicação de integração eficaz.

Com a necessidade das indústrias terem grande parte de sua produção feita de forma flexível, os robôs passam a ter uma importância fundamental. Pelo fato de serem programáveis permitem uma ação rápida, para mudança e adaptação para novos produtos e/ou processos.

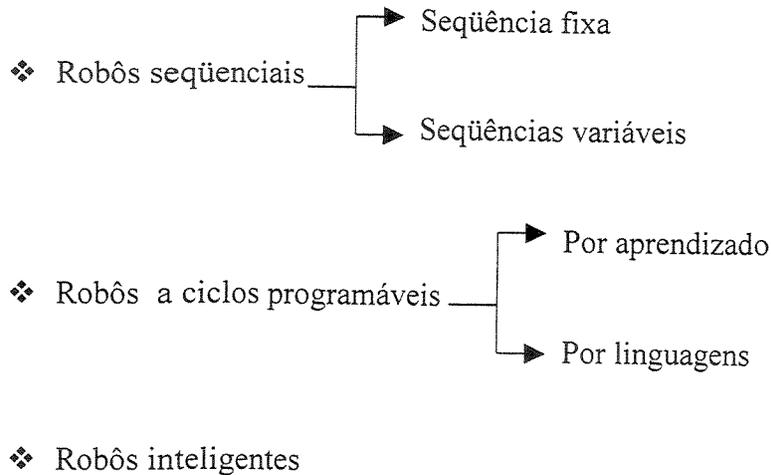
### **Caracterização dos Robôs**

Os robôs atuais não são caracterizados como máquinas de aparência humanóide, como prevalece no imaginário popular, mas sim, por critérios funcionais. Além disso, os critérios funcionais tomam conotações diferentes para cada país. Senso assim, ainda não há um consenso claro sobre os limites que separam uma máquina de um robô. Os japoneses, em um levantamento junto às suas indústrias, informam a presença de mais de 60.000 robôs. Contudo, a maioria desses robôs, limitam-se a sistemas articulados bem elementares, utilizados na alimentação de máquinas. Já nos EUA e Europa, esses robôs seriam apenas mecanismos sofisticados para alimentação ou transferência.

Segundo a AFRI ( Association Française de Robotique Industrielle ), manipuladores modestos, efetuando automaticamente seqüências de trabalho, variáveis ou fixas, devem ser denominados robôs. Complementando ainda, que toda robótica deve adaptar-se a inovações concebidas às máquinas de CNC, da informática e de novos sensores.

O RIA ( Robot Institute of América ) dá uma definição menos restritiva, substituindo a noção de adaptação pela noção de reprogramação : “um robô industrial é um manipulador multifuncional e reprogramável, concebido para mover cargas, peças, ferramentas ou dispositivos especiais, seguindo movimentos programados variáveis, para execução de tarefas variadas”.

O JIRA ( Japan Industrial Robot Association ) dá uma definição e função dos tipos de robôs industriais, distinguindo-os da seguinte forma :



### **A Robótica na Automação Industrial**

A automação pode ser dividida em dois tipos : **Flexível** e **Rígida**. É sobretudo na automação flexível que a robótica tem uma importância maior, pela versatilidade de seus movimentos que são programáveis. Vê-se assim que onde a robótica industrial pode interferir, proporcionando melhor desenvolvimento e otimização, está na automação flexível, que corresponde a cerca de 70% de toda produção industrial manufatureira. Isso significa a grande importância dada atualmente à robótica Industrial ( Ferreira 91 ).

### **Aplicações dos Robôs**

As primeiras operações dos robôs ocorreram na indústria automobilística, onde os setores escolhidos foram as operações de solda e pintura. Além da natureza das operações que oferecem riscos à saúde do trabalhador, também verificou-se um grande aumento na produtividade.

Considerando o conjunto de robôs, as indústrias automobilísticas absorvem 20% nos EUA, 40% no Japão e 58% na Europa, de toda a produção em robótica. Para fins industriais é praticamente o único setor que investe em pesquisas de robôs.

A classificação dos robôs por atividades é uma tarefa difícil. Os tipos de manipuladores apresentados ( Ferreira 91 ) oferecem um amplo leque de aplicações industriais.

- Manipuladores seqüenciais
  - serviços de prensas
  - periférico para máquinas-ferramentas
  - transferência de peças para montagem
  - montagens repetitivas
  - extração de moldes em máquinas
  - serviços de forno e forjaria
  - manipulação em microeletrônica
  
- Manipuladores programáveis
  - solda
  - pintura
  - montagem
  - controle de qualidade
  - rebitagem
  - posicionamento de peças para solda
  
- Manipuladores inteligentes
  - montagem e inserção de peças
  - manipulação com controle visual
  - soldagem com controle visual

## **A Evolução dos Robôs**

A idéia que prevalecia na concepção dos primeiros robôs estava direcionada para um modelo humano, ou seja, universalidade e características antropomórficas.

O conceito de robô universal está relacionado com sua versatilidade, segundo dois aspectos marcantes : polivalência e adaptabilidade.

A **polivalência** é entendida neste caso sob o ponto de vista mecânico. Quanto mais graus de liberdade tem o robô ( estrutura mecânica ), maior a sua polivalência. Para posicionar e orientar um corpo no espaço é necessário um conjunto de articulações mecânicas, com 6 graus de liberdade. Assim, um robô universal deve ter no mínimo seis ligações. Robôs com graus de liberdade acima de seis, o tornam abundantes, possibilitando contornar obstáculos e dando maior versatilidade para a execução de tarefas. Os elementos terminais ou conjunto de garras ou mãos-ferramentas também favorecem para estabelecer a polivalência dos robôs.

A **adaptabilidade** é a capacidade de executar uma tarefa numa vizinhança variável, cuja adaptação está relacionada com os sistemas de comunicação, decisão e percepção do robô.

Para a área industrial, a princípio, a idéia predominante foi a de robôs universais, que embora caros, serviam para todas as tarefas. Porém, para algumas aplicações, é necessário apenas 2 ou 3 graus de liberdade, como por exemplo carregar e descarregar peças ou máquinas-ferramentas. Com isso, a polivalência cede lugar para a modularidade, cuja tecnologia permite uma concepção que adapte a arquitetura mecânica do robô à sua aplicação. Diferentes arquiteturas podem ser obtidas pela montagem de elementos modulares de rotação, translação, módulos tipo “punho” ( com três graus de liberdade ), rótulos múltiplos ( tromba de elefante para robôs de pintura ) e múltiplos elementos de garra e serviços ( mãos-ferramentas ).

Simultaneamente ao desenvolvimento das máquinas de controle numérico, os robôs também foram desenvolvidos. Historicamente os robôs foram vistos pelas pessoas em geral, como sendo uma máquina com aparência humana, inspirados nos filmes de ficção científica, novelas, livros e peças teatrais. Essa idéia foi desfeita, quando da utilização dos robôs em processos industriais.

A robótica ( mais detalhes no capítulo-4 ) enquadra-se na automação programável, cuja característica principal é sua adaptação ao produto, possibilitando sua reprogramação sempre que houver alteração no produto, tendo uma atuação importante nas células flexíveis de manufatura.

## **ANEXO V**

### **Controlador Lógico Programável - CLP**

Na década de 50 os dispositivos utilizados para efetuar controles lógicos nos processos industriais eram eletromecânicos, baseados principalmente nos relés.

Apesar de funcionais, tais sistemas de controle apresentavam sérios problemas práticos. Havia uma quantidade muito grande de relés e, portanto um número elevado de interconexões, além do que, quando ocorria uma falha em um elemento comprometia os demais provocando interrupção na produção. A proteção contra umidade, gases inflamáveis, oxidações era necessária além de ocupar grande espaço.

No final da década de 60, alguns dispositivos transistorizados, substituíram as válvulas a vácuo, reduzindo os problemas causados pelos relés. Porém foi com surgimento dos componentes integrados em larga escala, que novas aplicações surgiram, como os computadores digitais e principalmente as tecnologias para automação industrial.

A primeira experiência de um controlador lógico programável que permitisse o uso de software foi em 1968, na divisão de hidramáticos da General Motors Corporation. O primeiro dispositivo a atender as especificações foi desenvolvida pela Gould Modicom em 1969.

## **Definição**

A norma NEMA define CLP como sendo: “Suporte eletrônico digital para armazenar instruções de funções específicas, como lógica, sequencialização, contagem e aritméticas, todas dedicadas ao controle de máquinas e/ou processos”.

Para a norma ABNT, a definição de CLP é: “Controlador lógico programável é um equipamento eletrônico-digital com hardware e software compatíveis com as aplicações industriais”.

As características desejadas de um controlador lógico, associada à flexibilidade do computador eram:

- Preços competitivos em relação aos sistemas de relés.
- Dispositivo de entrada e saída de fácil reposição.
- Funcionamento eficaz em ambientes industriais adversos ( vibração, calor, poeira, corrosão
- Facilidade de programação, manutenção, instalação e ampliação.
- Repetibilidade de operação e confiabilidade.

## Princípio de Funcionamento

Para facilitar a compreensão do que vem a ser um CLP e seu funcionamento, é necessário conhecer sua estrutura ( fig.A5.1 ) com os respectivos elementos.

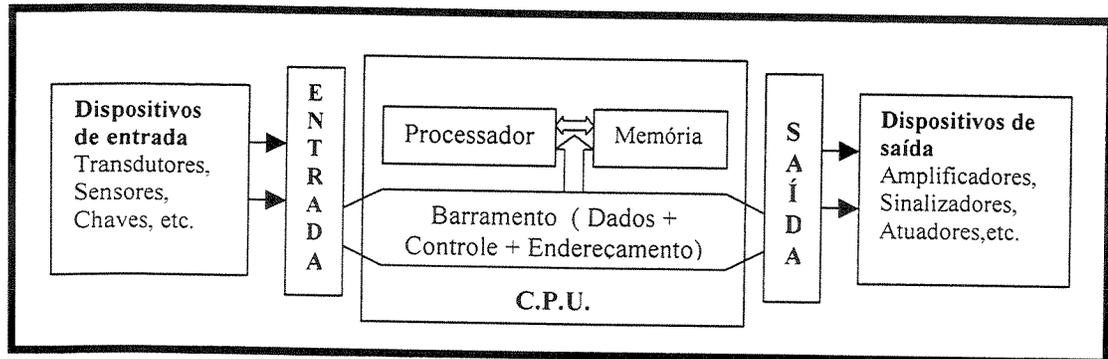


Figura A5.1 – Elementos que Compõe o CLP

**Sinais de Entrada** – são variáveis externas que entram no CLP, cuja origem esta no processo controlado. Tais sinais são gerados por elementos como sensores, chaves ou botoeiras, etc.

**Sinais de Saída** – são variáveis que controlam dispositivos em cada ponto de saída do CLP. Esses pontos poderão servir para intervenção direta no processo controlado por acionamento próprio ou servir para sinalização de estado em painel sinótipo, como exemplos podem ser citados os contadores, solenóides, displays, motores, sinalizadores, etc.

A figura A5.2 relaciona as entradas e saídas com programa ( linguagem ladder) e a memória do CLP.

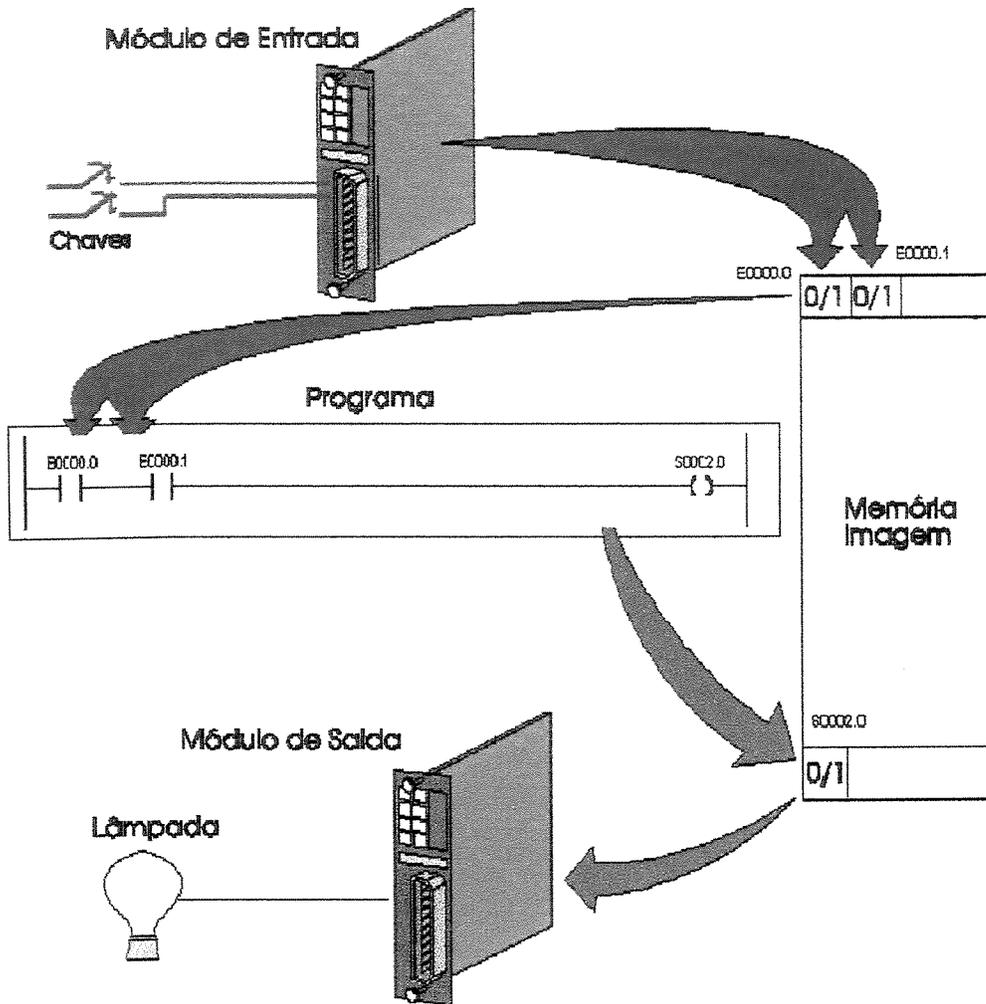


Figura 2.4 – Princípio de Funcionamento de um CLP

## Unidade Central de Processamento ( CPU )

Segue padrões semelhantes às arquiteturas dos computadores digitais, os quais são compostos essencialmente por um processador, banco de memória ( para dados e programação ) e um barramento de interligação.

A CPU executa um programa, conhecido como executivo ( princípio fundamental de funcionamento do CLP ) e é de responsabilidade do fabricante, que realiza ciclicamente as ações de leitura das entradas, execução do programa de controle do usuário e atualização das saídas (fig.A5.3).

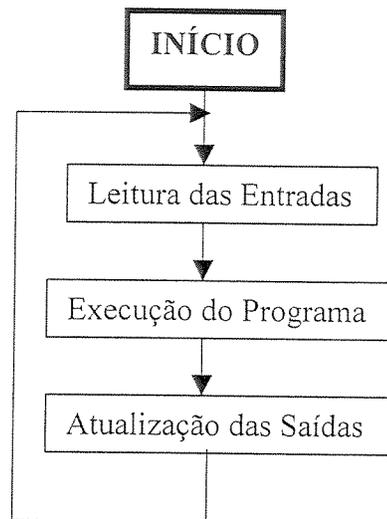


Figura A5.3 - Ciclo de Varredura

O tempo total para execução dessas tarefas, chamadas de ciclo de varredura ou scanning, depende principalmente da CPU e das características do programa, além da quantidade e tipo de pontos de entrada e saída. O valor do tempo está na ordem de milisegundos em média.

## **Evolução dos CLPs**

No início da década de 60, já surgiam as primeiras preocupações e necessidades de aumentar a produtividade da indústria. A competitividade aumentava, principalmente no setor automobilístico e apontou um sistema alternativo para substituir os relés até então utilizados. Uma possibilidade pensada pela General Motors seria um sistema em computador, que aumentaria a produtividade através da confiabilidade e qualidade.

Foi assim que, em 1968 a divisão de hidramático da G.M., fez a primeira experiência com um controlador lógico que permitia uma programação por software. Utilizando recursos externos, capazes de realizar operações de entrada e saída, em um mini-computador com recursos de programação para obter vantagens técnicas de controle que superou o custo de investimento.

Assim iniciava-se a era dos controladores lógicos programáveis. Em 1971 ocorreram as primeiras aplicações dos CLPs fora da indústria automobilística.

Em 1975 introduziu-se o controlador PID no CLP e com evolução da microeletrônica, a partir de 1977, foi possível substituir os componentes eletrônicos discretos por microprocessadores, com grande aceitação pela indústria e também pelo setor de serviço, como para controle de tráfego através dos semáforos.

Com os aperfeiçoamentos recebidos o CLP, passou a ser nos anos 80 um dos equipamentos mais atraentes para automação industrial, por possuir as seguintes características:

- Comunicação em rede
- Possibilidade de expansão por acoplamento
- Tamanho físico reduzido.

## **Interfaces de Entrada e Saída**

O sistema de entrada/saída é responsável pela conexão física entre a CPU e o meio externo, através de vários tipos de circuitos de interfaceamento.

Tais circuitos possuem padrões semelhantes entre os principais fabricantes de CLP.

### **Entradas / Saídas Discretas**

São os tipos de sinais mais encontrados em automação, utilizando CLP, nessas interfaces, a informação consiste em um único bit, em seu estado pode se apresentar, ligado ou desligado, por isso a sua característica discreta.

A tabela 4.1 apresenta uma lista com os principais dispositivos de entrada / saída com características discretas.

<b>Dispositivo de entrada</b>	<b>Dispositivo de saída</b>
Chave seletora	Relés de controle
Pushbothons	Solenóides
Sensores	Partida de motores
Chaves fim-de-curso	Válvulas
Sensores de proximidade	Ventiladores
Chaves sensoras de nível	Alarmes
Contatos de partida	Lâmpadas
Contatos de relés	Alarmes

**Tabela 4.1 - Dispositivos de Entrada/Saída Discretos**

Para o dispositivo citado as fontes de alimentação são distintas, cujo valor ou natureza não são as mesmas.

Por essa razão, as interfaces com dispositivos de entrada /saída discretos são encontrados no mercado em níveis de CA ou CC conforme necessidade.

## **Processador**

O processador ou CPU ( unidade central de processamento ) é constituído de um microprocessador que controla todas as ações de CLP e, os elementos que o constitui, são as memórias e o sistema de interligação.

A capacidade e as características dos CLPs estão intimamente relacionadas com o processador utilizado. Suas funções principais são : Soluções de operações lógico-matemática, manipulação de dados e controle de fluxo de programa. Essas operações são de ordem que jamais poderiam ser implementadas nos tradicionais diagramas de relés.

Os processadores mais utilizados ( 8031,8051,80188, Z80, 8085, 8088, etc), nos CLPs podem ser classificados, a princípio pelo número de bits que podem manipular. Valores como 8bits, 16bits ou 32bits, são os mais encontrados na atualidade.

## **Memórias**

Armazenagem temporária ou permanente dos dados de entrada, o programa a ser executado e os dados de monitoração e controle do próprio CLP são algumas das funções das memórias.

### **Tipos de Memórias mais Utilizadas**

- **Memória RAM** ( random access memory ) onde são armazenadas temporariamente os dados e variáveis internas, os valores atuais dos contadores e temporizadores. O programa do CLP também pode ser armazenado por tempo limitado . A memória RAM é volátil, ou seja, ao ser desenergizada os dados informados se perdem, motivo pelo qual é provida de bateria para protegê-la.
- **Memória EPROM** ( erasable and programmable read only memory ) ou EEPROM (eletrically) possível de ser programada pelo usuário, mas apenas de leitura . Nesta memória são armazenados os programas do usuário que não serão modificados, são memórias não voláteis do controlador.

### **Programação dos CLPs**

As linguagens de programação permitem aos usuários se comunicarem com o CLP através de um dispositivo de programação e definir as tarefas que CLP deve executar.

#### **As linguagens mais utilizadas são:**

- Diagrama de contatos ou Diagrama ladder
- Diagramas de funções ou Blocos de funções
- Listas de instruções

## Diagramas de Contatos

É a linguagem utilizada em CLP que apresenta de forma mais clara, uma lógica de controle para os técnicos e engenheiros, já que os contatos NA, NF e relés são conhecidos.

O diagrama ladder é uma escada (daí o nome ladder), e é feito entre duas linhas verticais, onde a da esquerda será conectada a tensão da fonte e a da direita o terra. As linhas horizontais são interligadas às verticais e nelas são colocados os símbolos de entrada e saída.

Entradas em série formam a função “E” entre as mesmas, já as entradas em paralelo formam a função “OU”. A negação da entrada é representada pelo contato NF.

As saídas cujo símbolo é  $-( )-$  são colocados do lado direito no final da linha horizontal (fig.A5.4).

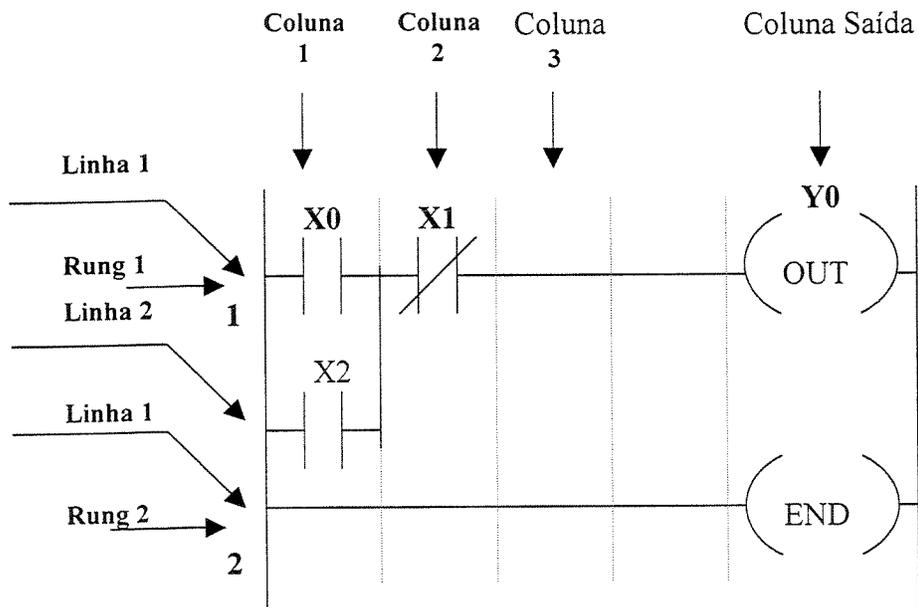


Figura A5.4 – Componentes da Programação em Linguagem Ladder

## **Diagrama de Funções ou Blocos de Funções**

Pode ser utilizado para pequenos programas de funções lógicas e para representação de programas sequenciais. Há uma semelhança entre as portas lógicas usadas em eletrônica digital.

## **Listas de Instruções**

Esta linguagem não é uma representação gráfica, mas sim uma instrução literal da programação.

A lista de instrução é formada por “linhas de instruções”, cada qual mostra um dado individual sendo que tanto à direita ou à esquerda (da instrução) pode ser feito comentários em linguagem normal. Assim é possível uma descrição precisa dos elementos de comutação.

## **Principais Fabricantes de CLPs**

A escolha criteriosa do fornecedor de hardware é importante na estratégia de automação de uma empresa.

Para se efetuar uma boa escolha de um fabricante de hardware, é necessário avaliar o produto, a confiabilidade do fabricante e de seu distribuidor, além do suporte técnico local.

Os principais fornecedores de CLPs no Brasil são: Allen-Bradley, GE Fanuc, Siemens e Atos, somente o último fabricante citado é nacional. Por muito tempo o CLP Allen-Bradley foi considerado um dos melhores do mercado nacional. Atualmente, a Siemens e GE-Fanuc disputam com a Allen-Bradley o mercado como os grandes fornecedores de CLPs, devido às vantagens oferecidas.

A Allen-Bradley possui um ótimo CLP para automação de manufatura discreta. Os seus CLPs possuem um bom desempenho para tratamento de entradas e saídas discretas e uma

variedade de instruções ladder que facilitam a automação deste tipo de processo. Para o controle de processos contínuos ou de bateladas, o CLP da Allen-Bradley tem algumas desvantagens, como limitação de memória, nas redes de comunicação e falta de instruções de controle avançado de malhas PID. A sua principal linguagem de programação é a Ladder (diagrama de contatos) .

A Siemens possui um excelente CLP para automação de processos contínuos e bateladas, além de um bom desempenho para tratamento de entradas e saídas digitais e analógicas. Possuem também mais de uma linguagem de programação o que facilita a sua utilização para processos com grande complexidade de automação, mas dificulta a sua utilização para processos simples. A Siemens possui uma grande variedade de redes de comunicações baseadas em padrões europeus como Profibus.

A GE-Fanuc possui um CLP que devido as suas características tem um ótimo desempenho em processos de manufatura discreta e também pode ser utilizado para processos contínuos e de bateladas. O CLP da GE-Fanuc não possui uma grande variedade de instruções na linguagem Ladder, mas isto não chega a comprometer sua utilização em processos que possuem muitos intravamentos.

## **Aplicações dos CLPs**

São nas indústrias que os CLPs têm mais aplicações, como na implementação de painéis sequenciais de intertravamento, controle de malhas, servo-posicionamento, sistema SCADA ( supervisory control and data acquisition ), sistema de controle estatístico de processo, sistema de controle de célula de manufatura, entre outras aplicações. A interface homem-máquina ( H/M ), possibilitando a visualização dos dados previamente programados em tempo real.

## Apêndice 1

### Ilustração do Projeto Idealizado e como foi Implementado

A seguir é apresentada de forma esquemática o projeto da plataforma com os subconjuntos, representados pelos postos de trabalho. As figuras A<sub>1</sub> e A<sub>2</sub> representam respectivamente o conjunto da plataforma projetado e a concepção implementada respectivamente.

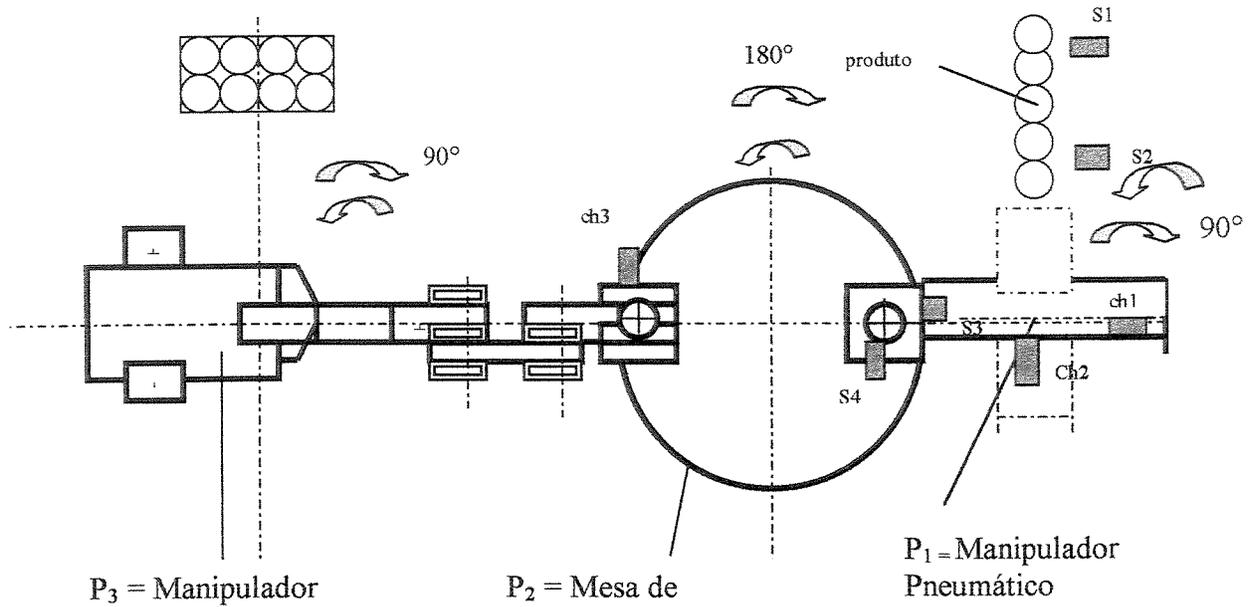


Fig. A<sub>1</sub> – Projeto Esquemático da Plataforma Idealizada.

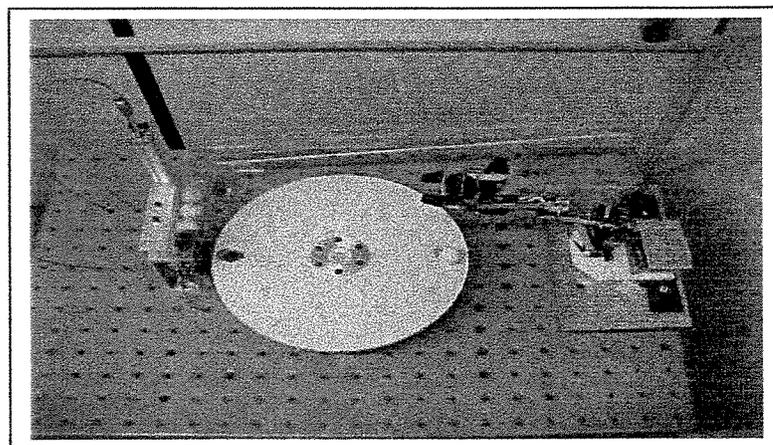


Fig. A<sub>2</sub> – Plataforma Implementada.

## Manipulador Pneumático

As figuras A<sub>3</sub> e A<sub>4</sub> a seguir representam respectivamente o projeto do manipulador pneumático e como o mesmo foi implementado.

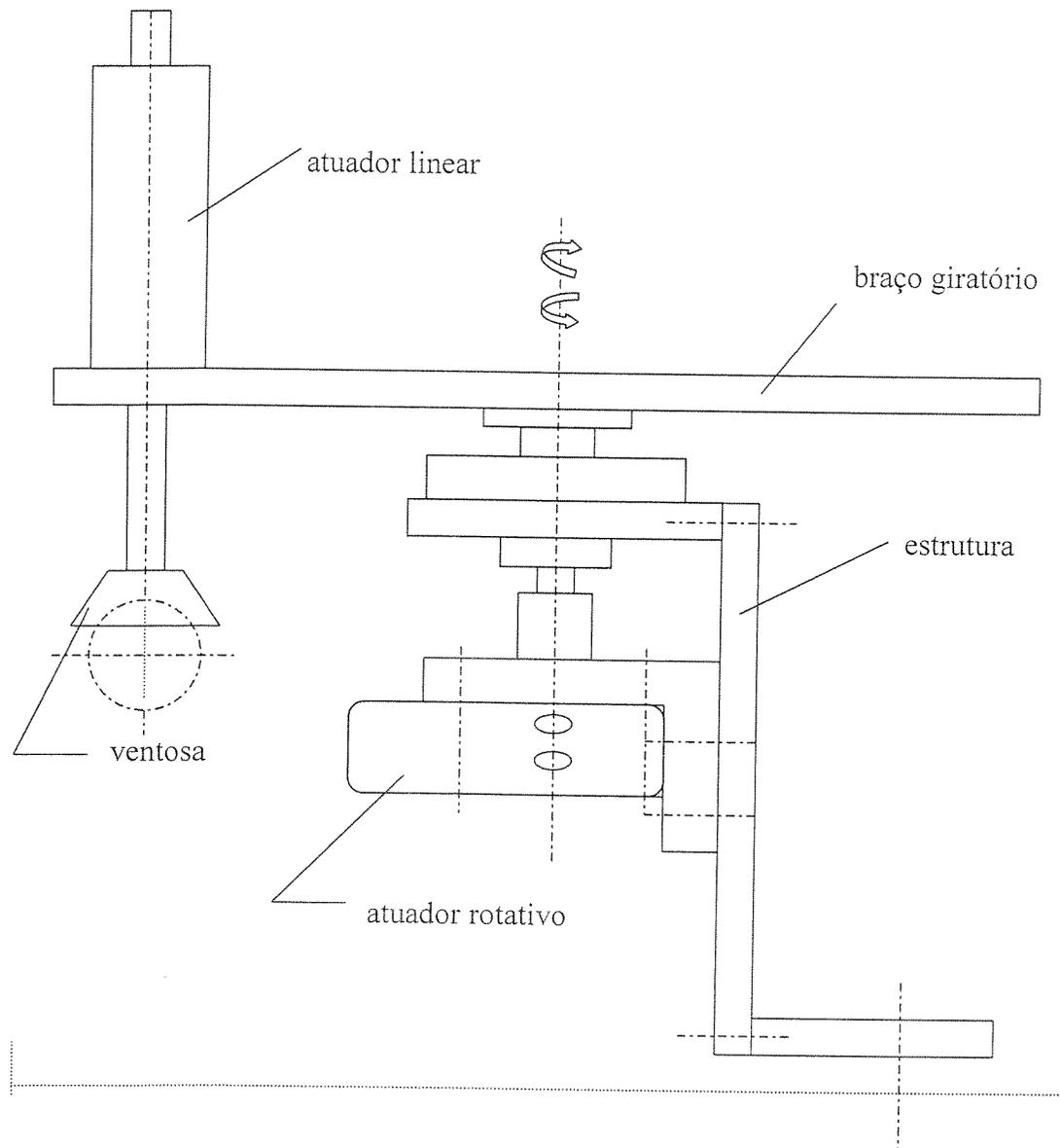
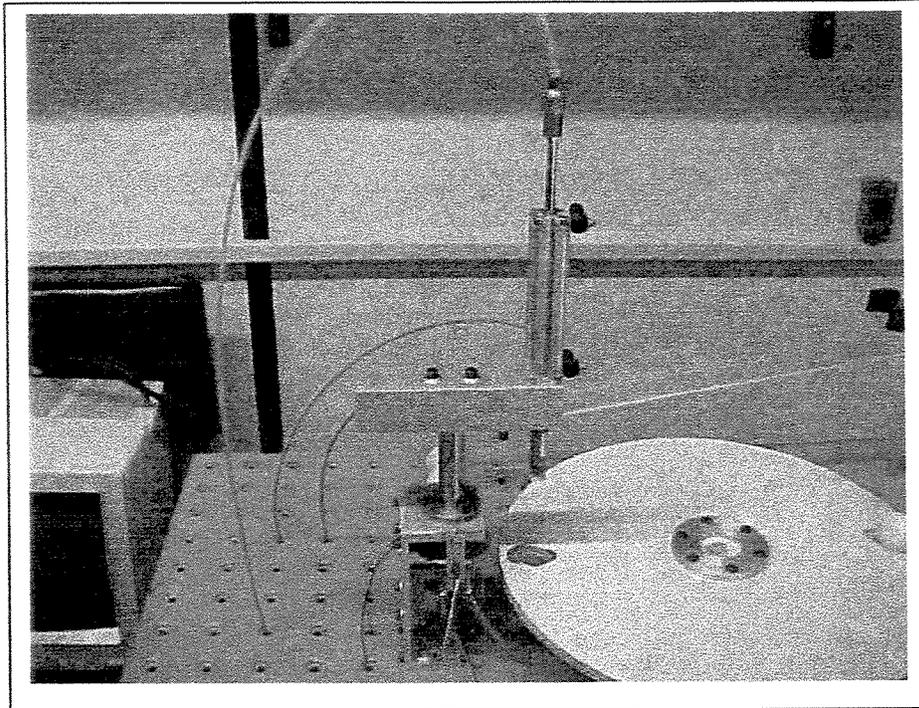


Fig. A<sub>3</sub> – Projeto do Manipulador Pneumático.



**Fig. A<sub>4</sub> – Manipulador Pneumático Implementado.**

## Mesa Rotativa de Transferência

A seguir são apresentadas as figuras que representam a mesa rotativa de transferência projetada conforme figura A<sub>5</sub> e a forma como a mesma foi implementada conforme figura A<sub>6</sub>.

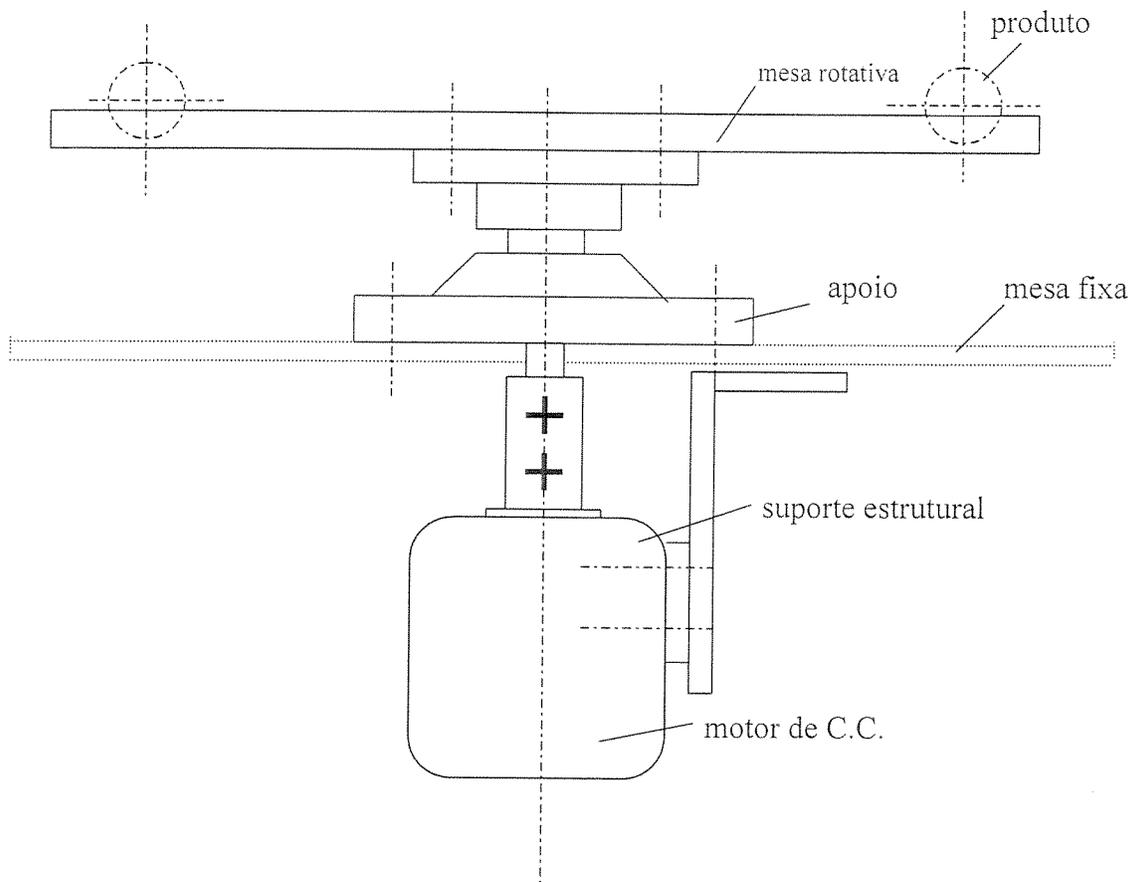
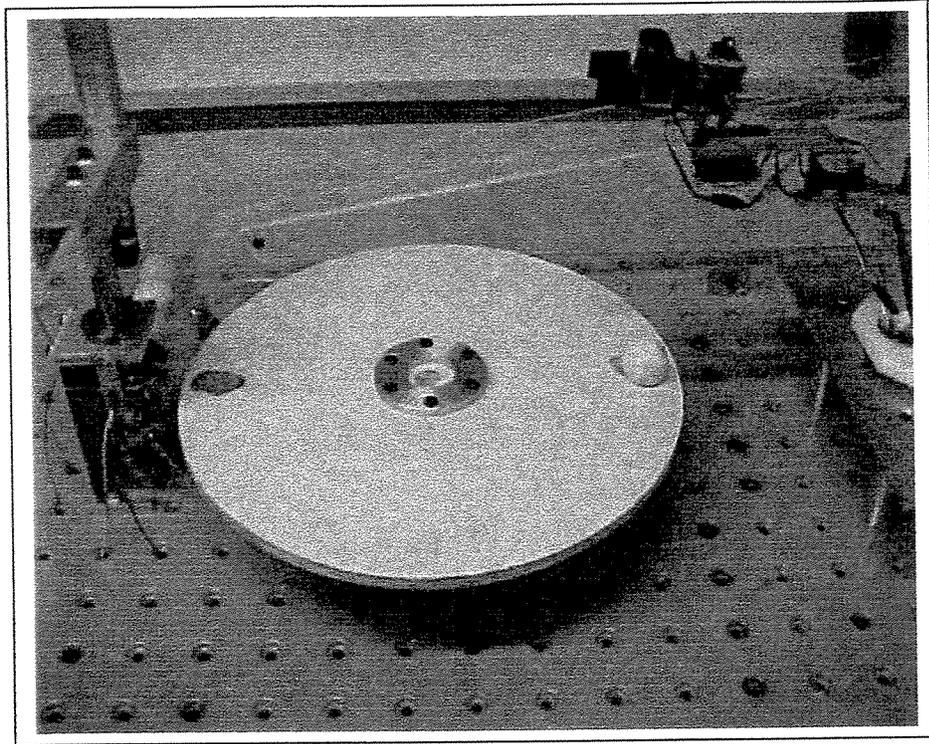


Fig. A<sub>5</sub> – Projeto da Mesa Rotativa de Transferência.



**Fig. A<sub>6</sub> – Mesa Rotativa de Transferência Implementada.**

## Manipulador Robótico -Robix<sup>TM</sup>

Trata-se de um kit didático denominado Robix<sup>TM</sup> RCS-6, para ensino de robótica, com seis graus de Liberdade, cujo acionamento ocorre através de servo-motores. A disposição dos braços pode ser montada em várias configurações, conforme aplicação. Para utilização deste trabalho foi usado apenas quatro graus de liberdade. A figura A7 representa esquematicamente a forma idealizada no projeto. A figura A8 mostra o manipulador robótico (Robix) como foi implementado, montado em base, para facilitar remoção do produto da mesa de transferência .

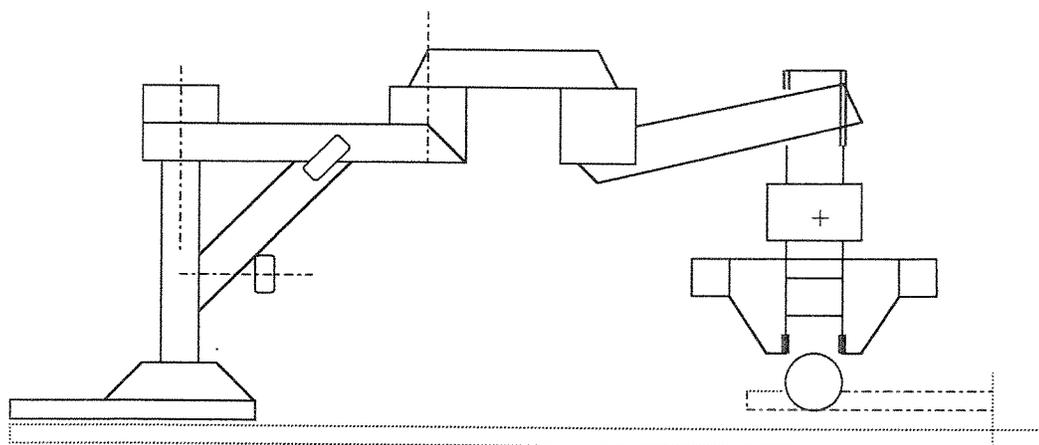


Fig. A7 – Configuração Estrutural do Manipulador Robótico Idealizado- Robix<sup>TM</sup>

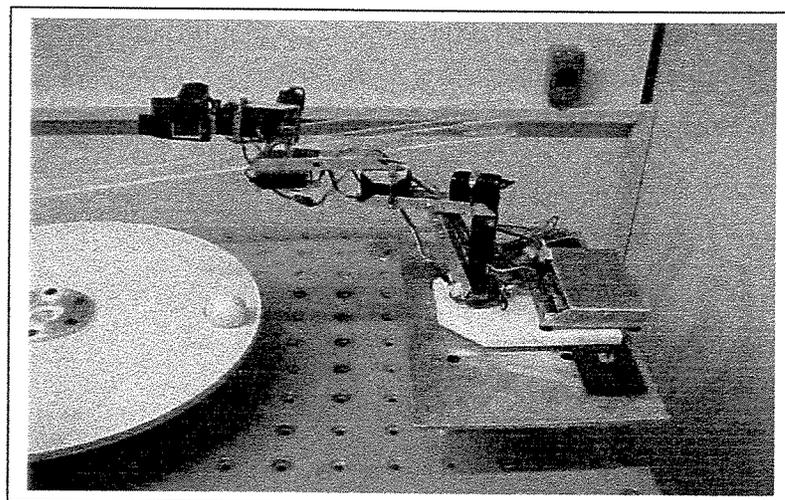


Fig. A8 – Montagem Implementada do Manipulador Robótico -Robix<sup>TM</sup>