

UM MODELO EM REDE DE PETRI PARA O SISTEMA AUTOMÁTICO DE  
INJEÇÃO DE UMA MÁQUINA INJETORA DE PLÁSTICO

Ilka Cristina Fernandes de Souza Telles

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO  
DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA  
UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE  
DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE  
MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Aprovada por:

---

Prof. Amit Bhaya , Ph. D.

---

Prof. Fernando Cesar Lizarralde, D. Sc.

---

Prof. José Stockler Canabrava Filho, Ph. D.

---

Prof. Antonio Eduardo Carrilho da Cunha, Dr. Eng.

RIO DE JANEIRO, RJ - BRASIL

MARÇO DE 2007

TELLES, ILKA CRISTINA FERNANDES  
DE SOUZA

Um modelo em rede de Petri para  
o sistema automático de injeção de  
uma máquina injetora de plástico [Rio de  
Janeiro] 2007

XVII, 118 p. 29,7 cm (COPPE/UFRJ,  
M.Sc., Engenharia Elétrica, 2007)

Dissertação - Universidade Federal do Rio  
de Janeiro, COPPE

1. Sistemas a Eventos Discretos
  2. Máquina Injetora
  3. Redes de Petri
  4. Linguagem Ladder
  5. Automação de Sistemas de Manufatura
- I. COPPE/UFRJ II. Título ( série )

# AGRADECIMENTOS

Ao meu orientador e professor João Carlos dos Santos Basílio por seu apoio, sua metodologia de orientação e por sua valiosa contribuição no desenvolvimento e revisão teórica deste trabalho.

Ao professor José Stockler Canabrava Filho por seu apoio e por sua contribuição no desenvolvimento deste trabalho.

Aos professores Fernando Lizarralde, Amit Bhaya, Liu Hsu, Marcelo Werneck e Ramon Romankevicius pelo apoio e ensinamentos importantes obtidos.

Aos meus familiares pela compreensão e confiança, que apostaram em mim ao longo de minha formação pessoal e profissional e que foram essenciais para superar cada uma das etapas de minha vida.

Ao meu noivo e seus familiares, pela compreensão e pelo apoio moral.

Aos meus colegas da COPPE, pela amizade e apoio ao longo do curso.

Ao órgão de fomento e amparo à pesquisa CAPES.

E ainda aos professores Bernardo Severo da Silva Filho, Orlando Bernardo Filho, Flávio Joaquim de Souza e o reitor Nival Nunes de Almeida, da UERJ, pelo reconhecimento de meus esforços, confiança e apoio durante a graduação, a Pós-graduação Lato Sensu em Mecatrônica na UERJ e pela recomendação para o Mestrado em Controle, Automação e Robótica da COPPE/UFRJ.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

## UM MODELO EM REDE DE PETRI PARA O SISTEMA AUTOMÁTICO DE INJEÇÃO DE UMA MÁQUINA INJETORA DE PLÁSTICO

Ilka Cristina Fernandes de Souza Telles

Março/2007

Orientador: João Carlos dos Santos Basilio

Programa: Engenharia Elétrica

A máquina injetora de plástico é um equipamento utilizado no processo de moldagem por injeção de resinas termoplásticas. Neste processo, a resina é plastificada e forçada ao interior do molde frio, onde sofre solidificação. O processo industrial de moldagem por injeção é utilizado para a confecção de diversos produtos como conexões em PVC e peças técnicas para indústria.

O objetivo deste trabalho é derivar um modelo em redes de Petri para o sistema de automação da máquina injetora de plástico IHP 180/350 (Mecânica Oriente). A abordagem empregada baseia-se na modelagem do sistema de automação da máquina injetora como um sistema a eventos discretos, uma vez que sua dinâmica é definida a partir da ocorrência de eventos.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

A PETRI NET MODEL FOR THE AUTOMATIC INJECTION SYSTEM OF A  
PLASTIC INJECTION MACHINE

Ilka Cristina Fernandes de Souza Telles

March/2007

Advisor: João Carlos dos Santos Basilio

Department: Electrical Engineering

The plastic injection machine is an equipment used in the thermoplastic resin injection molding process. In this process, the resin is plasticized and is forced towards the inner part of a cold mold, where it suffers solidification. The industrial injection molding process is used for confection of diverse products as PVC connections and industry technique parts.

The objective of this work is to derive a Petri net model for the automation system of the IHP 180/350 plastic injection machine (Mecânica Oriente). The approach is based on the modeling of the plastic injection machine automation system as a discrete event system since the machine dynamics is defined by means of event occurrences.

# Conteúdo

<b>Resumo</b>	<b>iv</b>
<b>Abstract</b>	<b>v</b>
<b>1 Introdução</b>	<b>1</b>
<b>2 Fundamentos da teoria de sistemas a eventos discretos</b>	<b>7</b>
2.1 Sistemas a eventos discretos . . . . .	8
2.1.1 Evento . . . . .	8
2.1.2 Sistemas e modelos . . . . .	8
2.1.3 Sistemas controlados pelo tempo e sistemas baseados em eventos	11
2.1.4 Sistemas a eventos discretos . . . . .	11
2.2 Linguagens e autômatos . . . . .	12
2.2.1 Modelos de linguagens de SED . . . . .	13
2.2.2 Operações em linguagens . . . . .	13
2.2.3 Autômatos . . . . .	13
2.2.4 Representação de linguagem por autômatos . . . . .	15
2.3 Redes de Petri . . . . .	16
2.3.1 Redes de Petri marcadas . . . . .	19
2.3.2 Dinâmica de redes de Petri . . . . .	19
2.3.3 Linguagens de redes de Petri . . . . .	22
2.3.4 Redes de Petri temporizadas . . . . .	23
2.3.5 Comentários Finais . . . . .	25

<b>3</b>	<b>Máquina injetora IHP-180/350</b>	<b>27</b>
3.1	Moldagem de plásticos por injeção . . . . .	27
3.1.1	Ciclo de moldagem por injeção . . . . .	28
3.2	Máquinas de moldagem por injeção . . . . .	29
3.2.1	Etapas de moldagem da máquina injetora . . . . .	32
3.3	Máquina injetora IHP-180/350 . . . . .	36
3.3.1	Fechamento . . . . .	37
3.3.2	Encoste . . . . .	41
3.3.3	Injeção . . . . .	43
3.3.4	Retorno do conjunto injetor . . . . .	46
3.3.5	Dosagem . . . . .	47
3.3.6	Descompressão . . . . .	48
3.3.7	Abertura do molde . . . . .	49
3.3.8	Extração . . . . .	52
3.4	Dispositivos de segurança e acionamento da máquina injetora . . . . .	54
3.5	Comandos . . . . .	56
3.6	Restrições importantes . . . . .	57
3.7	Problemas que podem ser causados no produto devido a variações dos parâmetros controlados . . . . .	58
<b>4</b>	<b>Modelagem do sistema de injeção por redes de Petri</b>	<b>61</b>
4.1	Modelagem por rede de Petri para o nível inferior . . . . .	62
4.1.1	Função fechamento . . . . .	64
4.1.2	Função encoste . . . . .	67
4.1.3	Função injeção . . . . .	69
4.1.4	Função retorno . . . . .	73
4.1.5	Funções Dosagem e Descompressão . . . . .	74
4.1.6	Função abertura . . . . .	79
4.1.7	Função extração . . . . .	83
4.2	Modelagem por rede de Petri completa . . . . .	86

4.3	Simulação . . . . .	93
4.3.1	O SIMULINK e o Real-time Windows Target . . . . .	93
4.3.2	O CLP SLC500 e o software RSLogix500 . . . . .	94
4.3.3	Simulação da variação de velocidade na função fechamento . .	96
<b>5</b>	<b>Conclusões e trabalhos futuros</b>	<b>106</b>
5.1	Trabalhos futuros . . . . .	106
<b>A</b>	<b>Circuito hidráulico da Máquina Injetora 180/350</b>	<b>108</b>



# Lista de Figuras

2.1	Processo de modelagem simples. . . . .	9
2.2	Sistema de filas simples. . . . .	12
2.3	Diagrama de Transição de Estados. . . . .	14
2.4	Autômato. . . . .	15
2.5	As quatro estações e suas mudanças. . . . .	18
2.6	Seqüência de disparos de transição em uma rede de Petri. . . . .	21
2.7	(a) Transição com disparo instantâneo e (b) Transição temporizada com atraso $\underline{v}_i$ . . . . .	25
2.8	Transição temporizada com atraso (a) Estado inicial e (b) Estado final.	25
3.1	Ciclo de Injeção. Figura originalmente apresentada em Ipiranga (1998).	29
3.2	Máquina de moldagem por injeção de parafuso simples. Figura origi- nalmente apresentada em Blass (1985). . . . .	30
3.3	Injetora coaxial de parafuso e êmbolo. Figura originalmente apresen- tada em Blass (1985). . . . .	30
3.4	Início do processo de injeção: placas abertas e canhão recuado. . . . .	32
3.5	Primeira etapa do processo de injeção: sistema hidráulico movimenta a placa móvel em direção à placa fixa (fechamento). . . . .	33
3.6	Segunda etapa do processo de injeção: avanço da unidade de injeção (canhão). . . . .	33
3.7	Terceira e quarta etapas do processo de injeção: Avanço da rosca pelo sistema hidráulico, realizando a injeção por transferência e, em seguida, o recalque. . . . .	34

3.8	Quinta etapa do processo de injeção: recuo da unidade de injeção (canhão), enquanto, simultaneamente, há a refrigeração do produto. . . . .	34
3.9	Sexta etapa do processo de injeção: recuo da rosca, dosagem do polímero em conjunto com o movimento de plastificação. . . . .	35
3.10	Sétima etapa do processo de injeção: abertura das placas, acionada por sistema hidráulico. . . . .	35
3.11	Parte final da última etapa do processo de injeção: extração da peça.	36
3.12	Máquina injetora. Figura originalmente apresentada em Canabrava Filho (1998). . . . .	37
3.13	Função fechamento. Figura originalmente apresentada em Oriente (1998). . . . .	38
3.14	Processo de fechamento complexo. Figura originalmente apresentada em Oriente (1998). . . . .	40
3.15	Função encoste. Figura originalmente apresentada em Oriente (1998).	41
3.16	Função injeção. . . . .	42
3.17	Processo de injeção complexo. Figura originalmente apresentada em Oriente (1998). . . . .	45
3.18	Função retorno. Figura originalmente apresentada em Oriente (1998).	46
3.19	Funções Dosagem e Descompressão. Figura originalmente apresentada em Oriente (1998). . . . .	47
3.20	Função abertura. Figura originalmente apresentada em Oriente (1998).	50
3.21	Processo de abertura complexo. Figura originalmente apresentada em Oriente (1998). . . . .	51
3.22	Função extração avanço. Figura originalmente apresentada em Oriente (1998). . . . .	52
3.23	Função extração retorno. Figura originalmente apresentada em Oriente (1998). . . . .	54
4.1	Diagrama esquemático para a função fechamento. . . . .	63
4.2	Diagrama da rede de Petri para a função fechamento. . . . .	64

4.3	Diagrama esquemático para a função encoste. . . . .	67
4.4	Diagrama da rede de Petri para a função encoste. . . . .	68
4.5	Diagrama esquemático para a função injeção. . . . .	69
4.6	Diagrama da rede de Petri para a função injeção. . . . .	71
4.7	Diagrama esquemático para a função retorno. . . . .	74
4.8	Diagrama da rede de Petri para a função retorno. . . . .	75
4.9	Diagrama esquemático para as funções dosagem e descompressão. . .	76
4.10	Diagrama da rede de Petri para as funções dosagem e a descompressão.	77
4.11	Diagrama esquemático para a função abertura. . . . .	80
4.12	Diagrama da rede de Petri para as funções abertura e extração. . . .	82
4.13	Diagrama esquemático para a função extração avanço. . . . .	83
4.14	Diagrama esquemático para a função extração retorno. . . . .	84
4.15	Rede de Petri total com solenóides. . . . .	87
4.16	Janela principal do RS Logix 500. . . . .	95
4.17	Esquema do sistema de simulação. . . . .	96
4.18	Componentes do sistema de simulação. . . . .	97
4.19	Planta no simulink. . . . .	98
4.20	Placa PCI1710. . . . .	98
4.21	Protoboard com três resistores e $10\Omega$ em série, somando $30\Omega$ . . . . .	99
4.22	Lógica Ladder da simulação (pág. 1). . . . .	100
4.23	Lógica Ladder da simulação (pág. 2). . . . .	101
4.24	Lógica Ladder da simulação (pág. 3). . . . .	102
4.25	Controlador SLC500. . . . .	103
4.26	Sinal de entrada da planta, correspondente à velocidade. . . . .	105
4.27	Sinal de saída da planta, correspondente à posição. . . . .	105
A.1	Circuito hidráulico da máquina injetora. . . . .	109

# Lista de Tabelas

3.1	Parâmetros de regulação da função fechamento. . . . .	40
3.2	Parâmetros de regulação da função injeção. . . . .	44
3.3	Parâmetros de regulação da função dosagem. . . . .	48
3.4	Parâmetros de regulação da função descompressão. . . . .	49
3.5	Parâmetros de regulação da função abertura. . . . .	50
3.6	Parâmetros de regulação da função extração. . . . .	53
3.7	Dispositivos de segurança. . . . .	55
3.8	Funções e solenóides acionados para execução das funções. . . . .	56
4.1	Definição do conjunto dos lugares para a rede de Petri $(P_f, T_f, A_f, w_f, E_f, l_f, x_f, V_f)$ da função fechamento. . . . .	65
4.2	Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri $(P_f, T_f, A_f, w_f, E_f, l_f, x_f, V_f)$ da função fechamento. . . . .	65
4.3	Definição do conjunto dos lugares para a rede de Petri $(P_e, T_e, A_e, w_e, E_e, l_e, x_e, V_e)$ da função $t_{f1}$ encoste. . . . .	68
4.4	Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri $(P_e, T_e, A_e, w_e, E_e, l_e, x_e, V_e)$ da função encoste. . . . .	68
4.5	Definição do conjunto dos lugares para a rede de Petri $(P_i, T_i, A_i, w_i, E_i, l_i, x_i, V_i)$ da função injeção. . . . .	70
4.6	Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri $(P_i, T_i, A_i, w_i, E_i, l_i, x_i, V_i)$ da função injeção. . . . .	70

4.7	Definição do conjunto dos lugares para a rede de Petri $(P_r, T_r, A_r, w_r, E_r, l_r, x_r, V_r)$ da função retorno. . . . .	75
4.8	Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri $(P_r, T_r, A_r, w_r, E_r, l_r, x_r, V_r)$ da função retorno. . . . .	75
4.9	Definição do conjunto dos lugares para a rede de Petri $(P_d, T_d, A_d, w_d, E_d, l_d, x_d, V_d)$ das funções dosagem e descompressão. . . . .	78
4.10	Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri $(P_d, T_d, A_d, w_d, E_d, l_d, x_d, V_d)$ das funções dosagem e descompressão. . . . .	78
4.11	Definição do conjunto dos lugares para a rede de Petri $(P_a, T_a, A_a, w_a, E_a, l_a, x_a, V_a)$ da função abertura. . . . .	81
4.12	Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri $(P_a, T_a, A_a, w_a, E_a, l_a, x_a, V_a)$ da função abertura. . . . .	81
4.13	Definição do conjunto dos lugares para a rede de Petri $(P_x, T_x, A_x, w_x, E_x, l_x, x_x, V_x)$ da função extração. . . . .	85
4.14	Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri $(P_x, T_x, A_x, w_x, E_x, l_x, x_x, V_x)$ da função extração. . . . .	85
4.15	Definição do conjunto dos lugares adicionais na rede de Petri $(P_c, T_c, A_c, w_c, E_c, l_c, x_c, V_c)$ completa. . . . .	88
4.16	Caracterização das transições adicionais e das correspondentes associações aos eventos para a rede de Petri $(P_c, T_c, A_c, w_c, E_c, l_c, x_c, V_c)$ completa. . . . .	88
4.17	Funções e solenóides acionados para execução das funções. . . . .	89
4.18	Definição do conjunto dos lugares relativos aos solenóides que ativam as funções. . . . .	89
4.19	Conversão dos pontos de corrente. . . . .	104
4.20	Conversão dos pontos de posição. . . . .	104

# Lista de Abreviaturas e Símbolos

Abreviaturas:

<i>CLP</i>	—	Controlador lógico programável
<i>CPN</i>	—	Rede de Petri colorida
<i>CVDS</i>	—	Ciclo de vida do desenvolvimento de sistemas
<i>MIC</i>	—	Manufatura integrada ao computador
<i>PC</i>	—	Computador pessoal (Personal computer)
<i>SED</i>	—	Sistemas a eventos discretos
<i>SIPN</i>	—	Rede de Petri interpretada por sinais
<i>SMF</i>	—	Sistema de manufatura flexível

Símbolos:

Sistemas Dinâmicos:

$e$	—	Evento
$E$	—	Conjunto discreto cujos elementos são eventos
$r(t)$	—	Sinal de referência
$t$	—	Tempo
$t_0$	—	Instante inicial
$u(t)$	—	Entrada
$u(t) = \gamma[r(t), t]$	—	Lei de Controle ou Controle
$x(t)$	—	Estado
$x(t_0) = x_0$	—	Estado inicial
$\mathcal{X}$	—	Espaço de Estados
$y(t)$	—	Saída

Linguagens formais e autômatos:

$\epsilon$	—	Palavra vazia
$\varepsilon$	—	Evento nulo
$G$	—	Autômato
$\mathcal{L}$	—	Linguagem
$\bar{L}$	—	Operação em linguagem fechamento de prefixo
$L^*$	—	Operação em linguagem fechamento de Kleene
$\mathcal{L}(G)$	—	Linguagem gerada por um autômato
$\mathcal{L}_m(G)$	—	Linguagem marcada por um autômato
$X$	—	Conjunto de estados
$X_m$	—	Conjunto dos estados marcados

Redes de Petri:

$\mathcal{L}(N)$	—	Linguagem gerada pela Rede de Petri marcada
$\mathcal{L}_m(N)$	—	Linguagem marcada pela Rede de Petri marcada
$\mathcal{PNL}$	—	Classe de linguagens que pode ser representada por redes de Petri marcadas
$T_0$	—	Conjunto de transições que sempre incorrem em atraso de disparo zero
$T_D$	—	Conjunto de transições que geralmente incorrem em algum atraso de disparo
$V$	—	Estrutura de temporização
$v_j$	—	Seqüências de tempos de vida $v_j = v_{j,1}, v_{j,2}, \dots$
$\tau_{r,k}$	—	k-ésimo tempo em que uma ficha é depositada em pi é precisamente o k-ésimo tempo de disparo de $t_r$
$\pi_{i,k}$	—	Instante de tempo no qual o lugar pi recebe sua k-ésima ficha, $k = 1, 2, \dots$

Máquina Injetora:

- P1(01)* — Pressão inicial de fechamento
- P1(02)* — Pressão de segurança do molde
- P1(03)* — Pressão de travamento
- P1(05)* — pressão de transferência
- P1(06)* — 1ª pressão de recalque
- P1(09)* — Pressão de dosagem
- P1(10)* — Pressão de descompressão
- P2(07)* — 2ª pressão de recalque
- P3(08)* — 3ª pressão de recalque
- S1* — Solenóide do sistema de segurança de passagem de óleo para a máquina.
- S1(01)* — 1ª posição de variação de velocidade
- S1(17)* — ponto de comutação entre V1 (05) e V2 (06)
- S1(21)* — Término da dosagem/início da descompressão
- S2(02)* — 2ª posição de variação de velocidade
- S2(18)* — ponto de comutação entre V2 (06) e V3 (07)
- S2(22)* — Término da descompressão
- S3(10)* — Posição de término da pressão inicial de fechamento e início da pressão de segurança do molde
- S3(19)* — ponto de comutação entre V3 (07) e V4 (08)
- S4(11)* — Posição de término da pressão de segurança do molde e início da pressão de travamento
- S4(20)* — posição final de transferência/início de recalque
- S82* — Solenóide que aciona o fechamento
- S83* — Solenóide que aciona o encoste
- S84* — Solenóide que aciona a injeção
- S85* — Solenóide que aciona a dosagem
- S86* — Solenóide que aciona a descompressão F1 e F2



- S87* — Solenóide que aciona retorno do conjunto injetor
- S88* — Solenóide que aciona abertura
- S89* — Solenóide que aciona o extrator hidráulico extrair
- S8A* — Solenóide que aciona o extrator hidráulico retornar
- S94* — Solenóide que aciona o aquecimento da zona 1
- S95* — Solenóide que aciona o aquecimento da zona 2
- S96* — Solenóide que aciona o aquecimento da zona 3
- S97* — Solenóide que aciona o aquecimento da zona 4
- S98* — Solenóide que aciona o aquecimento da zona 5
- S99* — Solenóide que aciona o aquecimento da placa fixa
- S9A* — Solenóide que aciona o aquecimento da placa móvel
- SIZ* — Solenóide do sistema de controle de passagem de óleo para a máquina.
- T1(01)* — Tempo de retardo de fechamento
- T1(03)* — 1º tempo de recalque
- T1(09)* — Tempo de retardo da dosagem (opcional)
- T1(17)* — tempo de transferência
- T2(04)* — 2º tempo de recalque
- T3(05)* — 3º tempo de recalque
- V1(01)* — Velocidade inicial de fechamento de fechamento
- V1(04)* — Velocidade de encoste
- V1(05)* — 1ª velocidade de recalque
- V1(09)* — Velocidade de dosagem
- V1(10)* — Velocidade de descompressão
- V2(02)* — Velocidade intermediária
- V2(06)* — 2ª velocidade de recalque
- V3(03)* — Velocidade final de fechamento de fechamento
- V3(07)* — 3ª velocidade de recalque
- V4(08)* — última velocidade de transferência/velocidade de recalque

# Capítulo 1

## Introdução

A teoria de sistemas a eventos discretos (Peterson, 1981; Ramadge e Wonham, 1989; Cassandras e Lafortune, 2000) constitui o arcabouço principal para a modelagem e projeto de sistemas de automação industrial, uma vez que, nesses, as dinâmicas são definidas pela ocorrência de eventos e manutenção de estados discretos. No passado, os sistemas a eventos discretos eram tão simples que soluções intuitivas eram adequadas a vários problemas. Hoje os sistemas precisam lidar com manufatura integrada ao computador, dados de alto nível processados e fluxo de material automatizado usando máquinas controladas por computador: são os chamados sistemas de manufatura flexível (Zhou e Venkatesh, 1999).

O aumento da complexidade dos sistemas de automação levou os sistemas a eventos discretos a um nível em que as aplicações requerem controle e coordenação para garantir o fluxo ordenado de eventos, surgindo a necessidade de métodos formais mais detalhados para sua análise e projeto.

Uma ferramenta importante no processo de controle de sistemas a eventos discretos é o controlador lógico programável, cujo propósito é monitorar parâmetros de processos cruciais e ajustá-los às operações de processos correspondentes. Tradicionalmente diagramas de lógica *ladder* são usados para capturar a seqüência de operações executadas pelo software de controle do sistema, principalmente porque assemelham-se aos diagramas dos circuitos com relés *ladder* e, por isso, são mais familiares aos profissionais da área (Lucas e Tilbury, 2003). Porém, os diagramas de lógica *ladder* não são de fácil manutenção.

A manufatura ágil e flexível mantém o produtor em competitividade de mercado. Isso ocasionou o surgimento de uma classe emergente de sistemas de automação: os reconfiguráveis, nos quais os sistemas e seus controladores lógicos programáveis freqüentemente precisam ser modificados devido a mudanças de partes do produto ou do processo. Por isso, uma característica desejável de um sistema de controle é a capacidade de reprojeto e de implementação rápida de tais controladores lógicos em resposta à reconfiguração mecânica do sistema de automação.

A natureza complexa dos modernos sistemas industriais e a exigência de alterações rápidas e testes sem interrupção de produção fazem com que o projeto e operação dos sistemas de automação requeiram ferramentas de modelagem e análise. A rede de Petri tem se mostrado uma ferramenta apropriada para a modelagem de tais sistemas, provendo uma solução integrada para modelagem, análise, simulação e controle de processos industriais, pois apresenta facilidade de modificação e capacidade de análise de propriedades importantes para controle, tais como longevidade<sup>1</sup>, segurança<sup>2</sup> e reversibilidade<sup>3</sup> (Cassandras e Lafortune, 2000). Uma das maiores vantagens do uso de modelos de rede de Petri é que o mesmo modelo pode ser usado para análise de propriedades comportamentais e avaliação de desempenho e também para a construção de simuladores e controladores de eventos discretos.

As redes de Petri têm despertado interesse há muito tempo, constituindo-se em uma das maneiras principais de se modelar sistemas a eventos discretos (Crockett et al., 1987). Murata (1989) apresenta um tutorial de revisão de redes de Petri onde são mostrados exemplos de modelagem, propriedades estruturais e comportamentais, métodos de análise e aplicações. Outro tutorial sobre redes de Petri foi apresentado por Zurawski e Zhou (1994), onde são apresentadas a descrição, as propriedades, os métodos de análise e as aplicações industriais das redes de Petri.

Dentro do contexto do resultado teórico, destacam-se: Valavanis (1990), que

---

<sup>1</sup>A longevidade significa que existe alguma trajetória amostra tal que qualquer transição pode eventualmente disparar de qualquer estado alcançado a partir do estado inicial.

<sup>2</sup>A segurança diz respeito a questões como alcançabilidade de estados indesejáveis na rede, a bloqueios do ponto de vista de uma linguagem de rede, comportamento de uma rede contido no comportamento de outra rede do ponto de vista da linguagem de rede, entre outras.

<sup>3</sup>A reversibilidade significa que se pode voltar ao estado inicial a partir de um estado qualquer da rede de Petri.

apresentou uma metodologia de modelagem hierárquica, com decomposição das operações com relações de precedência especificadas, gerando subsistemas modelados por redes de Petri estendidas por hierarquia, com elementos adicionais, como, por exemplo, surgimento e perda de fichas, indicando passagem de uma ficha de uma rede a outra; Freedman (1991), que definiu uma classe particular de rede usada na modelagem de tarefas repetitivas e mostrou como o comportamento repetitivo de tal rede pode ser estudado pela decomposição deste conflito em componentes deterministas associados a alternativas mutuamente exclusivas; Zhou et al. (1992), que apresentaram uma metodologia híbrida para síntese sistemática de modelos de rede de Petri para sistemas de manufatura automatizados, através de uma abordagem por metodologia “*bottom-up*”, no qual o levantamento e a modelagem são realizados gerando-se os macroprocessos principais e depois detalhando-se os processos que os compõem; Ferrarini (1992), que propôs uma metodologia prática e conceitual para projetar sistemas de controle lógico com redes de Petri; Zhou et al. (1993), que apresentaram a modelagem e análise de um sistema de manufatura flexível usando redes de Petri, descreveram seu processo de síntese e apresentaram o modo como redes de Petri com determinadas propriedades desejadas podem ser construídas; Venkatesh et al. (1994a), que trataram de simuladores de fichas; e Ferrarini et al. (1994), que ilustraram uma abordagem “*bottom-up*” para a síntese de modelos baseados em redes de Petri para projeto de controladores lógicos, mostrando uma nova abordagem para análise de longevidade modular.

Na segunda metade dos anos 1990, destacam-se os seguintes trabalhos: Cazola et al. (1995), que investigaram se o desempenho e a correção de um algoritmo de controlador lógico são afetados por um modo particular onde um esquema de redes de Petri é convertido em um algoritmo para rodar em um controlador lógico programável; Jones et al. (1996), que apresentaram a descrição de estruturas de redes de Petri com transições temporizadas (t-temporizadas) e sua lógica *ladder* equivalente e o desenvolvimento de um controlador de redes de Petri heurístico para o controle de sistemas a eventos de um sistema de manufatura, derivando-o para código *ladder*; Uzam et al. (1996), que estenderam o conceito de redes de Petri para lidar com

controladores de redes de Petri, incluindo atuadores e sensores como estruturas formais neste conceito; Park et al. (1999), que mostraram como um controlador lógico para uma linha com alto volume de transferência pode ser representado por redes de Petri, através da representação modular.

Mais recentemente tem-se os seguintes trabalhos: Holloway et al. (2000), que apresentaram uma técnica que pode converter especificações de metas de controle de alto nível em seqüências detalhadas de atuação e sensores de sinais, capacidade necessária aos métodos de síntese de controle automático para sistemas a eventos discretos; Schneeweiss (2001), que mostrou como a construção de redes relevantes por intermédio de modelagem de redes de Petri funciona na prática e como esta ferramenta pode auxiliar na análise de sistemas não reparáveis; Park et al. (2001), que apresentaram um novo método de representação para controladores lógicos com três modos de controle por intermédio de redes de Petri e introduziram e formalizaram uma estrutura de controlador lógico modular para linhas de transferência de alto volume; Mellado (2002), que apresentou a simulação de modelos baseados em redes de Petri temporizadas, com algoritmos de execução eficientes; e Koriem et al. (2004), que propuseram uma nova técnica baseada na teoria de redes de Petri em conjunto com a teoria de controle ótimo para análise de desempenho de sistemas a eventos discretos dinâmicos.

Uma abordagem recente sugere que um modelo de redes de Petri pode ser obtido da síntese de módulos individuais. Nessa linha, Tsinarakis et al. (2003) apresentaram uma decomposição de um sistema em módulos, sua derivação em modelos de rede de Petri modulares e a derivação de dois conjuntos de módulos (genérico, mais simples, e generalizado, mais geral). Os autores fizeram, ainda, uma síntese dos módulos individuais para gerar o modelo de todo o sistema. Tomando como ponto de partida o artigo de Ferrarini (1992), Tsinarakis et al. (2003) abordaram redes de Petri modulares generalizadas com transições temporizadas (t-temporizadas) e utilizaram tais redes para modelagem, análise, síntese e simulação de topologia aleatória dos sistemas de produção dedicados, em que as máquinas falham e são reparadas aleatoriamente.

Recentemente outros trabalhos importantes sobre redes de Petri foram desenvolvidos por Han et al. (2004), que apresentaram um modelo de redes de Petri estendidas temporizadas para transações concorrentes de um sistema de bancos de dados tendo como base a WEB e utilizaram-se da técnica de redução do modelo de redes de Petri para fazerem composição sincronizada. Peng e Zhou (2004) apresentaram uma visão dos métodos recentes de projeto de controle a eventos discretos, considerando as redes de Petri como o método mais efetivo de projeto e implementação de sistemas de controle a eventos discretos e que a conversão destas em aplicações de tempo real tem sido simplificada por muitas abordagens. Marranghello (2005) propôs a hierarquia como possibilidade de modelar partes do sistema em subredes associadas a lugares ou transições de uma rede. Bouyekhf e Moudni (2005) derivaram condições suficientes para analisar as propriedades estruturais de redes de Petri. Novos resultados que permitem a construção sistemática de abstrações funcionais para uma classe de modelos por redes de Petri que pode ser usada para representar os componentes primários do sistema de manufatura flexível de um veículo guiado automatizado foram apresentados por Zurawski (2005) e Tsinarakis et al. (2005), que apresentaram também técnicas de redução para uma classe de modelos de redes de Petri. Há ainda pesquisadores que geraram diagramas *ladder* de redes de Petri para utilizar métodos formais de projeto com *hardware* industrial existente baseado em *ladder*, como é o caso de Lee e Hsu (2000), Uzam et al. (2000) e Peng e Zhou (2001a).

Neste trabalho é considerado o desenvolvimento da modelagem por redes de Petri de um sistema de automação para uma máquina de injeção de plásticos, atendendo aos seguintes requisitos: (i) capacidade autônoma de reprojeção, evitando dependência do fornecedor da máquina, e (ii) facilidade na manutenção, permitindo alterações rápidas e testes simulados fora de linha. O processo de moldagem por injeção representa um importante processo industrial, sendo um método de conformação de resinas termoplásticas, em que o material fundido é forçado ao interior de um molde frio, onde endurece. O processo industrial de moldagem por injeção é utilizado para a confecção de diversos produtos como conexões em PVC, peças

técnicas para indústria automobilística, produtos para indústria de calçados, peças técnicas para indústria eletro-eletrônica e de informática e muitos outros. Nos últimos anos, percebe-se que os produtos utilizados e comercializados por diferentes setores da indústria brasileira vêm sendo fabricados cada vez mais com diferentes resinas termoplásticas (Galdamez e Carpinetti, 2004). Em Galdamez e Carpinetti (2004) foram estudadas e aplicadas as técnicas de planejamento e análise de experimentos na melhoria da qualidade industrial em um processo de moldagem por injeção plástica. Identificaram os níveis ótimos de regulagem e os parâmetros mais importantes da injeção plástica, temperatura da máquina e pressão de injeção e avaliaram o procedimento de implementação das técnicas de experimentação e as dificuldades práticas encontradas na empresa. Um outro trabalho envolvendo o processo de moldagem por injeção de plásticos foi apresentado por Liao et al. (2004), que propuseram uma técnica de modelagem baseada em componentes de sistema de monitoração e simulação de tempo-real, demonstrando os benefícios do uso de técnicas inteligentes para criar um modelo de solução de problemas de processos de moldagem.

Este trabalho está estruturado da seguinte forma: no capítulo 2 são revisados os conceitos principais da teoria de sistema a eventos discretos e de redes de Petri. O capítulo 3 apresenta a máquina injetora IHP-180/350, fabricada pela Mecânica Oriente. O capítulo 4 apresenta o modelo do sistema de injeção utilizando redes de Petri. Um exemplo de simulação de uma das partes do processo utilizando um CLP e o simulink também é apresentado no capítulo 4. Finalmente, no capítulo 5 são apresentadas as conclusões e sugestões de futuros trabalhos que darão continuidade ao trabalho aqui desenvolvido.

## Capítulo 2

# Fundamentos da teoria de sistemas a eventos discretos

Um sistema a eventos discretos (SED) é um sistema de estados discretos baseado em eventos, isto é, a evolução dos estados depende somente da ocorrência de eventos discretos assíncronos.

Neste sentido, Cury (2001) escreve que “De um modo geral, um sistema é uma parte limitada do Universo que interage com o mundo externo através das fronteiras que o delimitam.”...“Os sistemas de interesse percebem as ocorrências no mundo externo através da recepção de estímulos, denominados eventos.”...“A ocorrência de um evento causa, em geral, uma mudança interna no sistema, a qual pode ou não se manifestar a um observador externo. Além disso, uma mudança pode ser causada pela ocorrência de um evento interno ao próprio sistema, tal como o término de uma atividade ou o fim de uma temporização. Em qualquer caso, essas mudanças se caracterizam por serem abruptas e instantâneas: ao perceber um evento, o sistema reage imediatamente, acomodando-se em tempo nulo numa nova situação, onde permanece até que ocorra um novo evento. Desta forma, a simples passagem do tempo não é suficiente para garantir que o sistema evolua; para tanto, é necessário que ocorram eventos, sejam estes internos ou externos. Note ainda que a ocorrência desses eventos pode depender de fatores alheios ao sistema, de modo que este não tem, em geral, como prevê-los”.

Neste capítulo será feita uma revisão dos principais conceitos e dos fundamentos da teoria de sistemas a eventos discretos, estando estruturado da seguinte forma:



na seção 2.1 são apresentados os principais conceitos e os fundamentos da teoria de sistemas a eventos discretos; na seção 2.2 são estudados os conceitos de linguagem e autômato e na seção 2.3 é introduzido o conceito de redes de Petri.

## 2.1 Sistemas a eventos discretos

Nesta seção será feita uma revisão dos principais conceitos e dos fundamentos da teoria de sistemas a eventos discretos.

### 2.1.1 Evento

“*Evento*” é um conceito que precisa de boa base intuitiva para que seu significado seja compreendido. Pode ser pensado como uma ação acontecendo instantaneamente e que causa transições de um valor de estado para outro. Nesse trabalho será usado o símbolo “*e*” para denotar um evento e “*E*” para denotar o conjunto discreto cujos elementos são eventos.

### 2.1.2 Sistemas e modelos

Segundo descrição do dicionário padrão de termos elétricos e eletrônicos do IEEE, um sistema é uma combinação de componentes que agem juntos para desempenhar uma função que seria impossível de ser desempenhada por qualquer uma das partes individualmente.

No desenvolvimento do ciclo de vida de sistemas dinâmicos a eventos discretos (SED), a modelagem é um importante estágio no qual testes aplicados ao modelo permitem detectar problemas de projeto surgidos devido a especificações funcionais incompletas ou ambíguas. O modelo é um dispositivo que reproduz o comportamento do sistema. É uma abstração do sistema, que apenas se aproxima do seu comportamento verdadeiro do sistema. Para realizar a modelagem de um sistema, é necessário definir um conjunto de variáveis mensuráveis associadas ao sistema dado. Medindo-se essas variáveis por um período de tempo, é possível coletar dados. Selecionando-se um conjunto destas variáveis e supondo que seja possível variá-las pelo tempo, define-se um conjunto de funções de tempo chamadas variá-

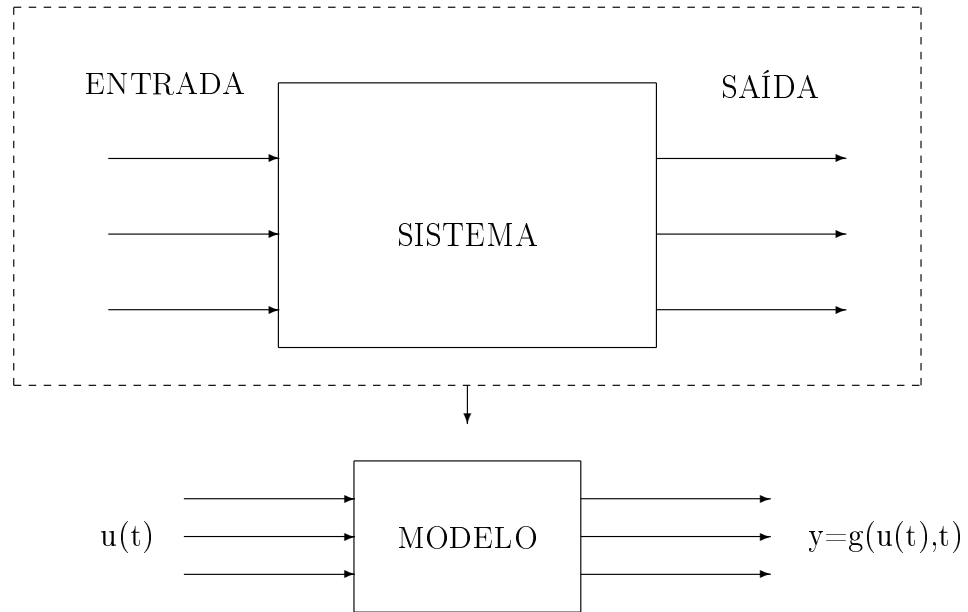


Figura 2.1: Processo de modelagem simples.

veis de entrada (Cassandras e Lafortune, 2000). Por outro lado, as variáveis que podem ser medidas diretamente, enquanto as de entrada são variadas, são definidas como o conjunto de variáveis de saída, ou seja, que descrevem as respostas providas por funções de entrada selecionadas. Para completar um modelo, existem algumas relações matemáticas entre entrada e saída, gerando o modelo na forma matemática (figura 2.1).

Quando se deseja também levar em conta no modelo do sistema as suas variáveis internas, surge a necessidade de se introduzir as chamadas variáveis de estado. Sendo  $u(t)$ ,  $y(t)$  e  $x(t)$ , respectivamente, a entrada, a saída, o estado e o estado inicial  $x(t_0)$ , o processo de modelagem pode ser definido como o conjunto de equações necessárias para especificar a saída  $y(t)$  para todo  $t \geq t_0$ , supondo conhecidos  $y(t_0)$  e  $u(t)$ ,  $t > t_0$ .

A maioria dos modelos de sistemas é baseada em equações diferenciais da forma:

$$\dot{x}(t) = f(x(t), u(t), t), \quad (2.1)$$

e, assim, o modelo de espaço de estados de um sistema será dado pelo seguinte

conjunto de equações:

$$\dot{x}(t) = f(x(t), u(t), t), x(t_0) = x_0, \quad (2.2a)$$

$$y(t) = g(x(t), u(t), t). \quad (2.2b)$$

A equação (2.2a) forma um conjunto de equações de estado com condições iniciais especificadas e a equação (2.2b) o conjunto de equações de saída.

A definição completa de um sistema contém a idéia de desempenho de uma função particular, ou seja, o sistema precisa ser controlado, selecionando-se a entrada certa para alcançar algum comportamento desejado. Sendo  $r(t)$  o sinal de referência, tem-se que a entrada de controle para o sistema é referida como a Lei de Controle ou apenas controle, sendo representada pela equação:

$$u(t) = \gamma(r(t), t). \quad (2.3)$$

O conceito de realimentação consiste em usar toda a informação disponível sobre o comportamento do sistema para ajustar continuamente a entrada de controle. Matematicamente falando, usar uma realimentação implica em estender a lei de controle (equação 2.3) para incluir a referência  $r(t)$ , a saída observada  $y(t)$  ou o estado  $x(t)$ . Pode-se representá-la na forma:

$$u(t) = \gamma(r(t), x(t), t) \text{ ou } u(t) = \gamma(r(t), y(t), t). \quad (2.4)$$

Os objetivos da teoria de sistemas são modelagem, projeto e síntese do controlador e avaliação de desempenho. A etapa de modelagem é o primeiro passo para se entender como um sistema trabalha. Desenvolve-se, para tanto, um modelo para tentar reproduzir o sistema físico. Se o modelo estiver preciso, o objetivo passa a ser o estudo do comportamento do sistema em diferentes condições, com diferentes valores de parâmetros ou funções de entrada. As etapas de projeto e síntese podem ser iniciadas quando já se tem técnicas de modelagem acuradas à disposição. É necessário juntar os componentes e selecionar os valores de parâmetros que levem a um projeto satisfatório para o controlador. A fase de avaliação de desempenho depende da aplicação. Normalmente é necessário limitar ou refinar os esquemas de controle

para selecionar alguns que satisfaçam os objetivos de desempenho (Cassandras e Lafortune, 2000).

### **2.1.3 Sistemas controlados pelo tempo e sistemas baseados em eventos**

Em sistemas controlados pelo tempo, a todo sinal de “clock”, um evento  $e$  de  $E$  é selecionado. Se nenhum evento ocorrer, isso significa que o “evento nulo” ocorreu, isto é, um evento cuja propriedade é não causar nenhuma transição de estado. Note que, nesse caso, as transições de estado são sincronizadas pelo relógio e a cada instante um evento (ou nenhum evento) é selecionado. O tempo é responsável por toda transição de estado.

Em sistemas dirigidos por eventos, em instantes de tempo, não necessariamente conhecidos com antecedência, e coincidindo ou não com as marcações de tempo, algum evento  $e$  irá ocorrer. Todo evento  $e$  de  $E$  define um processo distinto pelo qual os instantes de tempo em que  $e$  acontece são determinados. As transições de estado são o resultado de combinações destes processos de eventos assíncronos e simultâneos, processos estes que não precisam ser independentes um do outro.

### **2.1.4 Sistemas a eventos discretos**

Um Sistema a Eventos Discretos (SED) é um sistema de estado discreto, no qual o espaço de estados é um conjunto discreto, cujo o mecanismo de transição de estados é baseado em eventos. A evolução dos estados depende somente da ocorrência de eventos discretos, que acontecem pontualmente no tempo, e assíncronos, que ocorrem em intervalos irregulares, e não importa o que ocorre entre eles (Cury, 2001; Ramadge e Wonham, 1989).

Sistemas industriais são bons exemplos de sistemas a eventos discretos. Em um processo industrial, os clientes são as peças ou partes de peças da produção, que estão dispostas para o acesso às máquinas, que executam operações específicas, e dispositivos de manipulação de material, como robôs e correias transportadoras, que desempenham o papel de servidores. Quando as peças não estão sendo trabalhadas,

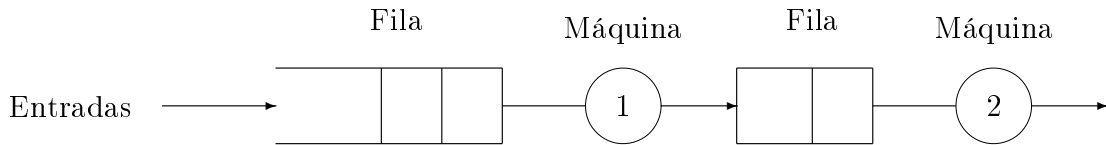


Figura 2.2: Sistema de filas simples.

elas são armazenadas em uma fila até que o servidor libere o acesso para a próxima operação que está disponível. As filas em um sistema industrial têm normalmente capacidades finitas devido às limitações físicas. Na figura 2.2, é apresentado um exemplo de uma linha de produção onde as peças passam por duas máquinas, sendo a capacidade da primeira fila infinita, enquanto a quantidade de peças que podem ficar esperando para serem processadas pela segunda máquina é limitada a dois (Cassandras e Lafortune, 2000). Como resultado, é possível que uma parte do serviço da máquina 1 seja completada, porém a máquina 2 esteja ainda ocupada e/ou a fila esteja completa. Neste caso, a peça tem que permanecer na máquina 1, embora não requeira mais nenhum serviço desta; além disso, outras peças são forçadas a esperar para ter o acesso à máquina 1 permanecendo em fila na entrada. Assim, o conjunto de eventos neste caso é  $E = \{a, c_1, d_2\}$ , sendo que  $a$  denota chegada de peça na primeira máquina,  $c_1$  denota conclusão de serviço da primeira máquina e  $d_2$  denota partida da segunda máquina. O estado do sistema pode ser definido como um vetor  $x = [x_1, x_2]^T$  correspondendo aos comprimentos das fila das duas máquinas. Observe-se que  $x_2$  é restrito aos valores  $\{0, 1, 2, 3\}$ , e quando  $x_2 = 3$ , a máquina 1 é bloqueada (acabou de executar o serviço na peça e a fila da segunda máquina está completa). Para modelar este fenômeno de bloqueio, é necessário introduzir um valor adicional  $B$  que  $x_2$  pode assumir, significando que o comprimento da fila é 3 (uma peça sendo processada e duas em espera) e que a primeira máquina está bloqueada (embora se possa ter, nesse caso,  $x_1 \neq 0$ ). O espaço de estado se torna o conjunto discreto  $X = \{(x_1, x_2) : x_1 \geq 0, x_2 \in \{0, 1, 2, 3, B\}\}$ .

## 2.2 Linguagens e autômatos

Nesta seção serão apresentados alguns dos formalismos necessários para modelar o comportamento lógico de um sistema a eventos discretos (SED): a teoria de lin-

guagens e autômatos. Um SED possui um conjunto de eventos  $E$  associado a ele, que pode ser visto como o “alfabeto” de uma linguagem. As seqüências de eventos associados a este conjunto são definidas como “palavras” desta linguagem.

### 2.2.1 Modelos de linguagens de SED

A linguagem definida sobre um conjunto de eventos  $E$  é o conjunto formado por seqüências de comprimentos finitos dos eventos de  $E$ . Por exemplo, associado ao conjunto de eventos  $E = \{a, b, g\}$ , pode-se definir a linguagem  $L_1 = \{\epsilon, a, abb\}$ , que consiste somente de três seqüências, onde  $\epsilon$  representa evento nulo.

### 2.2.2 Operações em linguagens

Seja  $E^*$  o conjunto de todas as seqüências finitas de elementos de  $E$ , incluindo  $\epsilon$  (evento nulo). O conjunto  $E^*$  é denominado fechamento Kleene de  $E$ . Uma vez que as linguagens são conjuntos, então as operações fixas habituais (união, interseção, diferença e complemento com respeito a  $E^*$ ) são aplicáveis a linguagens. Além disso, são definidas ainda, as seguintes operações envolvendo linguagens:

**Concatenação:** Dadas as linguagens  $L_1$  e  $L_2$ , então a linguagem  $L$  representativa da concatenação de  $L_1$  e  $L_2$  é definida como:

$$L = L_1L_2 = \{s_1s_2 : s_1 \in L_1 \text{ e } s_2 \in L_2\}.$$

**Prefixo-fechamento:** Dada uma linguagem  $L \in E^*$ , então o fecho de prefixo de  $L$  ( $\bar{L}$ ) é definido como:

$$\bar{L} = \{s \in E^* : \exists t \in E^* \text{ de tal forma que } st \in L\}.$$

**Fechamento Kleene:** Dada uma linguagem  $L \subseteq E^*$  então o fecho de Kleene de  $L$  é definido como:

$$L^* = \{\epsilon\} \cup L \cup LL \cup LLL \cup \dots$$

### 2.2.3 Autômatos

Um autômato é um dispositivo capaz de representar uma linguagem de acordo com regras claras. Assim, dado um conjunto de eventos  $E$ , um conjunto de estados  $X$

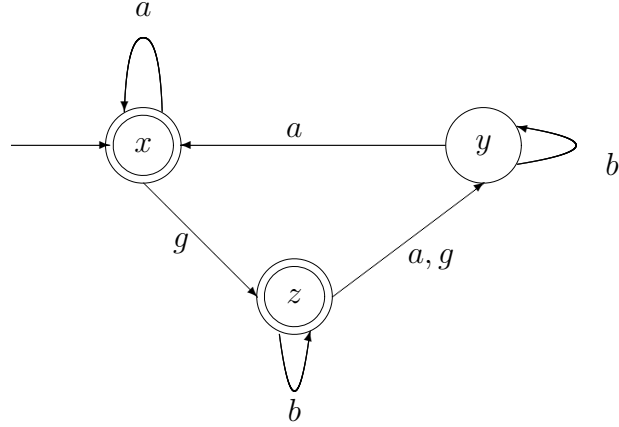


Figura 2.3: Diagrama de Transição de Estados.

e uma linguagem  $L$ , um diagrama de transições de estados é uma representação esquemática da evolução dos estados de acordo com as seqüências definidas em  $L$  (figura 2.3). Na figura utiliza-se  $\bigcirc$  para representar os estados e  $\rightarrow$  para representar as transições de estados, onde a letra em cima da seta denota o evento que causou a transição. A seta apontada para um estado e que não inicia em um outro estado ( $\rightarrow\bigcirc$ ) define o estado inicial. Finalmente,  $\odot$  indicam os estados marcados, aqueles que determinam que uma tarefa ou operação foi completada.

Uma definição formal para um autômato é a seguinte.

**Definição 2.1** *Um autômato determinista, denotado por  $G$ , é uma sextupla  $G = (X, E, f, \Gamma, x_0, X_m)$ , onde  $X$  é o conjunto de estados,  $E$  é o conjunto finito de eventos associados com as transições em  $G$ ,  $f : X \times E \rightarrow X$  é a função de transição em que  $f(x, e) = y$  significa que existe uma transição rotulada pelo evento  $e$  que leva do estado  $x$  para o estado  $y$ ,  $\Gamma : X \rightarrow \mathcal{P}(E)$  é função de evento ativa, onde  $\mathcal{P}(E)$  denota o conjunto potência de  $E$  (conjunto formado por todos os subconjuntos de  $E$ ),  $x_0$  é o estado inicial, e  $X_m \subseteq X$  é o conjunto de estados marcados (Cassandras e Lafortune, 2000).*

O autômato da figura 2.3 pode então, ser caracterizado pela seguinte sêxtupla:  $X = \{x, y, z\}$ ,  $E = \{a, b, g\}$ ,  $f(x, a) = x$ ,  $f(x, g) = z$ ,  $f(y, a) = x$ ,  $f(y, b) = y$ ,  $f(z, a) = f(z, g) = y$  e  $f(z, b) = z$ ,  $\Gamma(x) = \{a, g\}$ ,  $\Gamma(y) = \{a, b\}$  e  $\Gamma(z) = E$ ,  $x_0 = x$  e  $X_m = \{x, z\}$ .

A evolução de um autômato se dá da seguinte forma: partindo do estado inicial  $x_0$  e na ocorrência de um evento  $e \in \Gamma(x_0) \subseteq E$  haverá uma transição para o estado  $f(x_0, e) \in X$ . Este processo então continua baseado nas transições para as quais  $f$  é definida.

## 2.2.4 Representação de linguagem por autômatos

A partir do diagrama de transição de estados de um autômato, é possível fazer a conexão entre linguagem e autômato. Começando do estado inicial, podem-se traçar todos os possíveis caminhos diretos no diagrama de transição de estado, e, dentre esses, os que terminam em um estado marcado. Assim, é possível chegar aos conceitos de linguagens gerada e marcada por um autômato. A linguagem gerada por um autômato é um conjunto formado por todas as seqüências de eventos que podem ser definidas começando-se pelo estado inicial; a linguagem marcada por um autômato é um subconjunto da linguagem gerada que contém todas as seqüências de eventos que terminam em um estado marcado. Uma definição formal para as linguagens gerada e marcada é a seguinte:

**Definição 2.2** A linguagem gerada por  $G = (X, E, f, \Gamma, x_0, X_m)$ , é  $\mathcal{L}(G) := \{s \in E^* : f(x_0, s) \text{ é definida}\}$ . A linguagem marcada por  $G$  é  $\mathcal{L}_m(G) := \{s \in \mathcal{L}(G) : f(x_0, s) \in X_m\}$ .

Note que, na definição acima, faz-se uso da forma estendida da função de transição de estados (Cassandras e Lafortune, 2000).

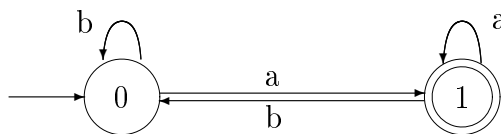


Figura 2.4: Autômato.

Para melhor compreensão pode-se considerar o seguinte exemplo (figura 2.4): seja  $E = \{a, b\}$  um conjunto de eventos e considere o autômato de estado finito  $G = (X, E, f, \Gamma, x_0, X_m)$ , sendo  $X = \{0, 1\}$ ,  $x_0 = 0$ ,  $X_m = \{1\}$ , e  $f$  definida



como  $f(0, a) = 1$ ,  $f(0, b) = 0$ ,  $f(1, a) = 1$ ,  $f(1, b) = 0$ , representado na figura 2.4. Como  $f$  é uma função definida em todos os estados para todos os eventos (função completa), então a linguagem gerada por esse autômato é o próprio  $E^*$ , isto é,  $\mathcal{L}(G) = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, \dots\}$ . Como o único estado marcado é o 1, então esse estado só pode ser atingido pelas seqüências de  $\mathcal{L}(G)$  em que o último evento é o  $a$ . Portanto  $\mathcal{L}_m(G) = \{b\}^*\{a\}L^*$ , sendo  $L = \{a\}^*\{b\}\{b\}^*\{a\}$ . Pode-se, então, concluir que um autômato  $G$  é a representação de duas linguagens,  $\mathcal{L}(G)$  e  $\mathcal{L}_m(G)$ .

## 2.3 Redes de Petri

Nesta seção será apresentado um outro formalismo para a modelagem de SED: as redes de Petri. As redes de Petri distinguem-se pelas características resultantes de sua natureza gráfica e seu suporte matemático simples, clareza e compactação para representar comportamentos complexos como causalidade, concorrência, paralelismo, sincronização e troca de informação (Mellado, 2002).

As redes de Petri, desenvolvidas por Carl A. Petri no início dos anos 60 (Reisig, 1985), são uma ferramenta matemática do tipo rede, descrita graficamente por grafos que são intuitivos e capturam um conjunto de informações estruturais sobre o sistema. Uma rede de Petri é um dispositivo que manipula eventos de acordo com certas regras. Uma das características é que inclui condições explícitas sob as quais um evento pode ser habilitado. Isto permite a representação de um SED muito geral cuja operação depende do controle de esquemas potencialmente complexos.

As principais razões para se considerar o uso de redes de Petri para modelagem de SED são:

- As redes de Petri podem representar uma classe maior de linguagens que a classe de linguagens regulares<sup>1</sup> (Cassandras e Lafortune, 2000);
- O arcabouço de análises técnicas que foi desenvolvido para estudá-las;
- A existência de modelos de redes de Petri temporizados;

---

<sup>1</sup>Uma linguagem é denominada regular se ela puder ser associada à linguagem marcada de um autômato de estados finitos.

- As redes de Petri podem ser usadas para modelar propriedades como processos de sincronização, eventos assíncronos, operações concorrentes e conflitos ou compartilhamento de recursos (Mellado, 2002);
- O mesmo modelo pode ser usado para análise de comportamento e avaliação de desempenho (Zurawski, 2005).
- Do ponto de vista analítico, tem-se que um modelo de rede de Petri pode ser descrito por um conjunto de equações algébricas lineares ou outros modelos matemáticos que reflitam o comportamento do sistema.

O processo de definição de redes de Petri envolve dois passos (Cassandras e Lafortune, 2000):

1. Define-se o grafo de redes de Petri, também chamada estrutura de redes de Petri, que é análogo ao diagrama de transição de estados de um autômato;
2. Junta-se a este grafo um estado inicial, um conjunto de estados marcados, e a função de transição rotulada, resultando num modelo completo de redes de Petri, sua dinâmica associada, e as linguagens que gera e marca.

Com base nos conceitos de condições, eventos, lugares e transições, é possível dizer que, em redes de Petri, os eventos são associados a transições e para uma transição ocorrer, várias condições devem ser satisfeitas. A informação relacionada a estas condições está contida nos lugares. Alguns lugares são vistos como a entrada para uma transição; eles estão associados com as condições necessárias para que esta transição ocorra. Outros lugares são vistos como a saída de uma transição; eles estão associados a condições que são afetadas pela ocorrência destas transições.

Transições, lugares e as relações entre eles definem os componentes básicos de um grafo de redes de Petri. Um grafo de redes de Petri tem dois tipos de nós (lugares e transições) e arcos conectando-os. É um grafo bipartido no sentido de que os arcos não podem conectar nós diretamente do mesmo tipo; isto é, arcos conectam lugares a transições e transições a lugares. Um lugar é uma entrada para uma transição se

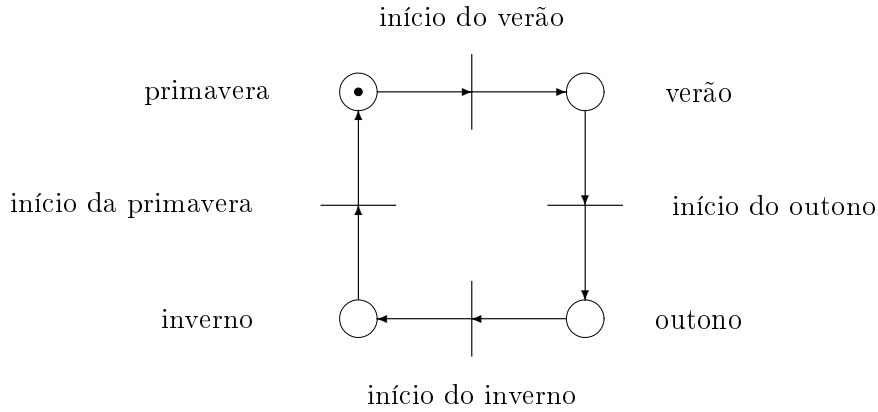


Figura 2.5: As quatro estações e suas mudanças.

há um arco orientado conectando este lugar à transição. Um lugar é uma saída de uma transição se há um arco orientado conectando a transição ao lugar. Deve-se salientar ainda que lugares de entrada ou de saída podem representar pré-condições ou pós-condições, respectivamente.

A figura 2.5 mostra um sistema no qual as condições são “é inverno”, “é verão”, “é outono”, “é primavera” e os eventos são “início da primavera”, “início do verão”, “início do outono”, “início do inverno”. Vê-se que cada condição é representada por um círculo e cada evento por uma barra. Cada condição que é gerada é marcada com uma ficha. No caso da figura 2.5 há uma ficha no lugar “é primavera”, mostrando que o sistema está atualmente na primavera. Assim, para que o evento “início do outono” ocorra é necessário que haja uma ficha no lugar “é verão”. Desta forma, quando a transição associada ao evento “início do verão” disparar, então o lugar “é primavera” perderá uma ficha, que irá para o lugar “é verão”, habilitando, assim, a transição “início do outono” (Reisig, 1985).

A definição precisa de grafo de redes de Petri é dada a seguir (Cassandras e Lafortune, 2000).

**Definição 2.3** Um grafo de uma rede de Petri é um grafo bipartido ponderado  $(P, T, A, w)$  sendo,  $P$  um conjunto finito de lugares,  $P = \{p_1, p_2, \dots, p_n\}$ ,  $T$  um conjunto finito de transições,  $T = \{t_1, t_2, \dots, t_m\}$ ,  $A \subseteq (P \times T) \cup (T \times P)$  é o conjunto de arcos ligando lugares a transições e transições a lugares no grafo, e  $w : A \rightarrow \{1, 2, 3, \dots\}$  é a função de ponderação dos arcos, que indica seu peso.

Para descrever um grafo da rede de Petri é conveniente usar  $I(t_j)$  para representar o conjunto de lugares de entrada para a transição  $t_j$ . Analogamente,  $O(t_j)$  representa o conjunto de lugares de saída da transição  $t_j$ . Tem-se, então, que:  $I(t_j) = \{p_i \in P : (p_i, t_j) \in A\}$  e  $O(t_j) = \{p_i \in P : (t_j, p_i) \in A\}$ . Uma notação similar pode ser usada para descrever transições de entrada e saída para um dado lugar  $p_i$ , isto,  $I(p_i)$  e  $O(p_i)$ .

### 2.3.1 Redes de Petri marcadas

Os mecanismos que identificam se as transições (eventos) podem disparar são as fichas atribuídas aos lugares. As fichas definem uma marcação, formalmente definida como uma função  $x : P \rightarrow \mathbb{N}^n$  e levam à definição de redes de Petri marcadas.

**Definição 2.4** *Uma rede de Petri marcada é uma quintupla  $(P, T, A, w, x)$  onde  $(P, T, A, w)$  é uma rede de Petri e  $x$  é uma marcação do conjunto de lugares  $P$ , isto é,  $x : P \rightarrow \mathbb{N}^n$ ,  $x = [x(p_1), x(p_2), \dots, x(p_n)]$ , onde  $x(p_i)$  representa o número de fichas no lugar  $p_i$ . O vetor  $x = [x(p_1), x(p_2), \dots, x(p_n)]$  é também referido como o estado da rede de Petri.*

A evolução de uma rede de Petri se dá a partir do disparo das transições. Contudo, para que uma transição dispare, a mesma deve estar habilitada. Uma transição é dita estar habilitada quando o número de fichas em  $p_i$ , é maior ou igual que o peso do arco que conecta  $p_i$  a  $t_j$ , para todos os lugares  $p_i$  que são entradas para a transição  $t_j$ , ou seja, a transição  $t_j \in T$  em uma rede de Petri é dita estar habilitada se  $x(p_i) \geq w(p_i, t_j)$  para todo  $p_i \in I(t_j)$  (Cassandras e Lafortune, 2000).

### 2.3.2 Dinâmica de redes de Petri

A representação da dinâmica das redes de Petri dá-se pela movimentação das fichas, à medida que as transições disparam ou, equivalentemente, que os eventos ocorrem. As seguintes regras são usadas para governar o fluxo de fichas. Uma transição  $t_j$  habilitada pode ou não disparar, dependendo de interpretação adicional. O disparo de uma transição habilitada  $t_j$  remove de cada lugar de entrada  $p_i$  um número de

fichas igual ao peso do arco que conecta  $p_i$  a  $t_j$  e, em seguida, deposita em cada lugar de saída  $p_k$  o número de fichas igual ao peso do arco que conecta  $t_j$  a  $p_k$ .

Uma definição formal para a dinâmica de redes de Petri é a seguinte:

**Definição 2.5** *A função de transição de estados,  $f : \mathbb{N}^n \times T \rightarrow \mathbb{N}^n$ , da rede de Petri  $(P, T, A, w, x)$  está definida para a transição  $t_j \in T$  se e somente se  $t_j$  está habilitada, isto é,*

$$x(p_i) \geq w(p_i, t_j) \text{ para todo } p_i \in I(t_j). \quad (2.5)$$

Se  $f(x, t_j)$  estiver definida, então, quando  $t_j$  disparar, ter-se-á

$$x' = f(x, t_j), \text{ onde } x'(p_i) = x(p_i) - w(p_i, t_j) + w(t_j, p_i), i = 1, \dots, n. \quad (2.6)$$

A partir dessa definição chega-se as seguintes conclusões:

- 1) Uma transição está pronta para disparar somente quando está habilitada;
- 2) O próximo estado depende dos lugares de entrada e saída ligados a transição  $t_j$  e dos pesos dos arcos.

É importante notar que o número de fichas não precisa necessariamente permanecer constante após o disparo de uma transição em uma rede de Petri. Isto vem da equação (2.6), já que é inteiramente possível que  $\sum_{p_i \in P} w(t_j, p_i) > \sum_{p_i \in P} w(p_i, t_j)$  ou que  $\sum_{p_i \in P} w(t_j, p_i) < \sum_{p_i \in P} w(p_i, t_j)$ .

A definição 2.5 conduz a um importante conceito, que é o da alcançabilidade de estados, isto é, os estados a que se pode chegar após o disparo das transições a partir do estado inicial. Uma definição formal para estados alcançáveis é a seguinte:

**Definição 2.6** *O conjunto de estados alcançáveis da rede de Petri  $(P, T, A, w, x)$  é  $\mathcal{R}[(P, T, A, w, x)] := \{y \in \mathbb{N}^n : \exists s \in T^* \text{ e } f(x, s) = y\}$*

Note que, na definição acima, faz-se uso da forma estendida da função de transição de estados (Cassandras e Lafortune, 2000).

Para ilustrar o processo de disparo de transições e mudanças de estado de uma rede de Petri deve-se considerar a rede de Petri da figura 2.6(a), onde o estado

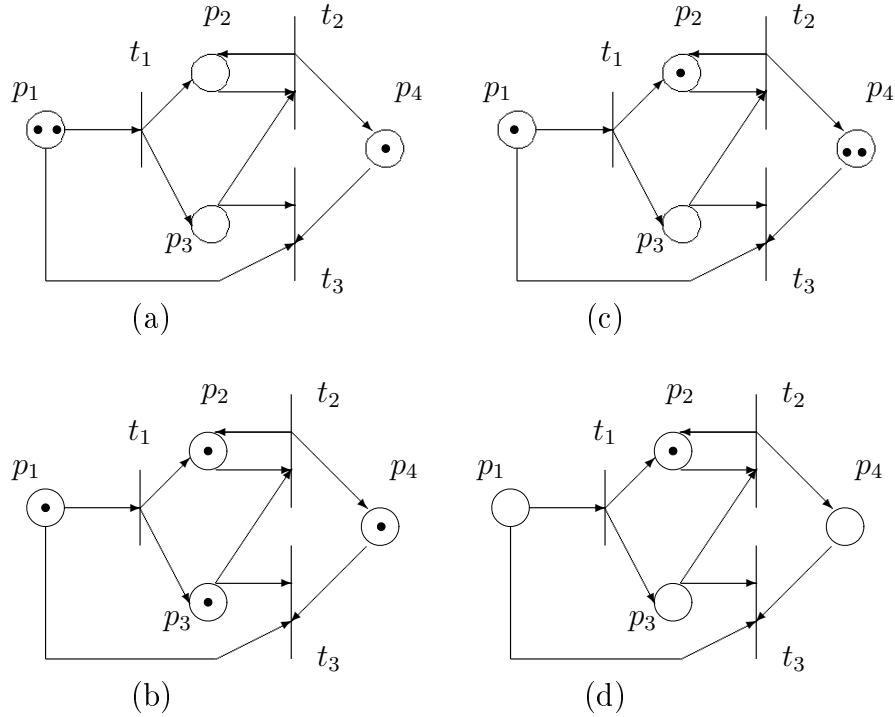


Figura 2.6: Seqüência de disparos de transição em uma rede de Petri.

inicial é  $x_0 = [2, 0, 0, 1]$ . Pode-se observar que a única transição habilitada é  $t_1$ , já que requer uma única ficha do lugar  $p_1$  e tem-se que  $x_0(p_1) = 2$ . Quando  $t_1$  dispara, uma ficha é removida de  $p_1$ , e uma ficha é colocada em cada um dos lugares  $p_1$  e  $p_3$ . O novo estado é  $x_1 = [1, 1, 1, 1]$ , como mostrado na figura 2.6(b). Todas as transições  $t_1$ ,  $t_2$  e  $t_3$  estão habilitadas.

Supondo que a transição  $t_2$  dipare, uma ficha é removida do lugar  $p_3$ ,  $p_2$  mantém sua ficha, já que é lugar de entrada e de saída, e o lugar  $p_4$  recebe uma ficha. O novo estado é  $x_2 = [1, 2, 0, 2]$ , como mostrado na 2.6(c). Neste estado apenas a transição  $t_1$  está habilitada.

Supondo, a partir da 2.6(b), que, ao invés de  $t_2$  disparar,  $t_3$  dispare, os lugares  $p_1$ ,  $p_3$  e  $p_4$  perdem suas fichas e não há lugares de saída. O novo estado  $x'_2 = [0, 1, 0, 0]$ , como mostrado na 2.6(d), e não há nenhuma transição habilitada (Cassandras e Lafortune, 2000).

Como mencionado anteriormente, o modelo de rede de Petri pode ser descrito por um conjunto de equações algébricas lineares ou outros modelos matemáticos que reflitam o comportamento do sistema, como as equações de estado. Seja, portanto,

$x = [x(p_1), x(p_2), \dots, x(p_n)]$  o estado atual e  $x' = [x'(p_1), x'(p_2), \dots, x'(p_n)]$  o próximo estado e suponha que uma transição particular,  $t_j$ , foi disparada. Para isso, primeiro define-se o vetor disparo  $u$ , um vetor linha  $m$ -dimensional da forma

$$u = [0, \dots, 0, 1, 0, \dots, 0], \quad (2.7)$$

onde o único 1 aparece na  $j$ -ésima posição, para indicar o fato de que a  $j$ -ésima transição é correntemente disparada. Além disso, define-se a matriz de incidência de uma rede de Petri,  $A$ , uma matriz  $m \times n$  cujo elemento  $(j, i)$  é da forma

$$a_{ji} = w(t_j, p_i) - w(p_i, t_j), \quad (2.8)$$

que representa a diferença de pesos que aparece na atualização de  $x(p_i)$ , na equação (2.6). Usando a matriz de incidência  $A$ , pode-se agora escrever a seguinte equação de estados:

$$x' = x + uA, \quad (2.9)$$

que descreve o processo de transição de estados como resultado de uma entrada  $u$ , ou seja, um disparo particular de transição. A  $i$ -ésima linha da equação 2.9 é precisamente a equação 2.6. Vê-se, então, que  $f(x, t_j) = x + uA$ , onde  $f(x, t_j)$  é a função de transição definida anteriormente. O argumento  $t_j$  nesta função indica que a  $j$ -ésima entrada em  $u$  é não nula. A equação de estados provê uma ferramenta algébrica apropriada e uma alternativa ao significado puramente gráfico da descrição do processo de disparo de transições e troca de estados de uma rede de Petri.

### 2.3.3 Linguagens de redes de Petri

Seja  $E$  o conjunto de eventos de um SED sob consideração e cuja linguagem será modelada pela rede de Petri. Uma definição formal para rede de Petri rotulada é:

**Definição 2.7** *Uma rede de Petri rotulada  $N$  é uma óctupla  $N = (P, T, A, w, E, l, x_0, X_m)$  onde  $(P, T, A, w)$  é um grafo de rede de Petri,  $E$  é um conjunto de eventos para rotular transições,  $l : T \rightarrow E$  é a função de rotular transição,  $x_0 \in \mathbb{N}^n$  é o estado inicial da rede e  $X_m \subseteq \mathbb{N}^n$  é o conjunto de estados marcados da rede (Cassandras e Lafortune, 2000).*

De posse do conceito de rede de Petri rotulada pode-se apresentar os conceitos de linguagens gerada e marcada por uma rede de Petri.

**Definição 2.8** *A linguagem gerada pela rede de Petri rotulada  $N=(P,T,A,w,E,l,x_0,X_m)$  é  $\mathcal{L}(N) := \{l(s) \in E^* : s \in T^* \text{ e } f(x_0, s) \text{ é definido}\}$ . A linguagem marcada por  $N$  é  $\mathcal{L}_m(N) := \{l(s) \in \mathcal{L}(N) : s \in T^* \text{ e } f(x_0, s) \in X_m\}$ .*

A linguagem  $\mathcal{L}(N)$  representa todas as seqüências de rótulos de transição que são obtidos por todas as possíveis seqüências (finitas) de disparos de transições em  $N$ , começando no estado inicial  $x_0$  de  $N$ ; a linguagem marcada  $\mathcal{L}_m(N)$  é um subconjunto destas seqüências que deixa a rede de Petri em um estado que é um membro do conjunto de estados marcados dado na definição de  $N$ .

A classe de linguagens que pode ser representada por redes de Petri rotuladas é  $\mathcal{PNL} := \{K \subseteq E^* : \exists N = (P, T, A, w, E, l, x_0, X_m)[\mathcal{L}_m(N) = K]\}$ .

**Observação 2.1** *Autômatos e redes de Petri podem ser ambos usados para representar o comportamento de um SED. Há dois importantes critérios de comparação que apontam redes de Petri como um modelo mais abrangente: expressividade da linguagem e construção de módulos. Quanto à expressividade da linguagem a classe PNL é estritamente maior do que a classe R, ou seja, as redes de Petri com conjuntos finitos de lugares e transições podem representar mais linguagens e  $E^*$  do que os autômatos de estados finitos. Quanto à construção de módulos, em autômatos representa um aumento significativo de complexidade enquanto que em redes de Petri é possível decompor um sistema complexo em módulos e obter um sistema integrado apenas através da manutenção dos módulos e adição de alguns poucos lugares e ou transições representando o acoplamento entre os módulos.*

### 2.3.4 Redes de Petri temporizadas

No caso de modelos de SED não temporizados, as redes de Petri fornecem uma estrutura muito geral. Para que seja possível especificar sistemas de tempo real, avaliar o desempenho de sistemas dinâmicos de um modo geral e examinar questões referentes ao seu escalonamento, por exemplo, é necessário que se considerem



informações relativas ao tempo em que ocorrem os eventos no sistema considerado. Assim, a estrutura das redes de Petri pode ser estendida para modelos temporizados, equipando a rede de Petri com uma estrutura de temporização e transformando-a em uma rede de Petri temporizada (Cassandras e Lafortune, 2000).

Seja  $(P, T, A, w, x)$  uma rede de Petri marcada. Acrescentar a essa rede de Petri uma estrutura de temporização é definir uma seqüência de temporização  $v_j$  associada a transições  $t_j \in T$ . Um número real positivo,  $v_{j,k}$ , associado a  $t_j$  tem o seguinte significado: quando a transição  $t_j$  é habilitada pela  $k$ -ésima vez, não dispara imediatamente, mas incorre em um atraso no disparo de  $v_{j,k}$ ; durante este período as fichas são mantidas nos lugares de entrada de  $t_j$ .

Nem todas as transições precisam ter atraso no disparo. Assim, o conjunto de transições  $T$  pode ser escrito como  $T = T_0 \cup T_D$ , onde  $T_0$  é o conjunto de transições que disparam imediatamente após serem habilitadas e  $T_D$  é o conjunto de transições que geralmente incorrem em algum atraso de disparo.

Uma definição formal para a estrutura de temporização é a seguinte:

**Definição 2.9** *A estrutura temporizada associada a um conjunto de transições temporizadas  $T_D \subseteq T$  de uma rede de Petri marcada  $(P, T, A, w, x)$  é um conjunto  $V = \{\underline{v}_j : t_j \in T_D\}$  de seqüências de tempos de vida  $\underline{v}_j = v_{j,1}, v_{j,2}, \dots, t_j \in T_D$ ,  $v_{j,k} \in \mathbb{R}^+$ ,  $k = 1, 2, \dots$*

Note que  $v_{j,k}$  significa que quando a transição  $t_j$  é habilitada pela  $K$ -ésima vez, ela não dispara imediatamente, mas espera  $v_{j,k}$  unidades de tempo.

Uma definição formal para redes de Petri temporizadas é a seguinte:

**Definição 2.10** *Uma rede de Petri temporizada é uma sextupla  $(P, T, A, w, x, V)$  onde  $(P, T, A, w, x)$  é uma rede de Petri marcada e  $V = \{\underline{v}_j : t_j \in T_D\}$  é uma estrutura de temporização.*

Para diferenciar transições com disparo instantâneo das transições temporizadas, tem-se que as primeiras são representadas por barras, enquanto as últimas são representadas por retângulos conforme mostrado na figura 2.7.

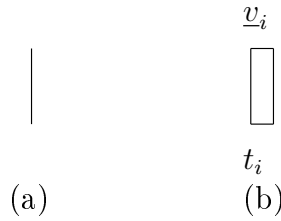


Figura 2.7: (a) Transição com disparo instantâneo e (b) Transição temporizada com atraso  $\underline{v}_i$ .

Considere o exemplo da figura 2.8. A transição  $t_1$  é temporizada e seu atraso é representado por  $\nu_1$ . Como o lugar  $p_1$  tem uma ficha (figura 2.8 (a)) e o arco que o une à transição  $t_1$  tem peso 1, a transição  $t_1$  é habilitada e, após o tempo  $\nu_1$ , ela é disparada. Com isso, o lugar  $p_1$  perde sua ficha e o lugar  $p_2$  recebe uma ficha (figura 2.8 (b)).

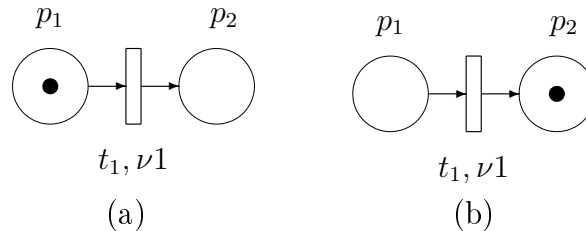


Figura 2.8: Transição temporizada com atraso (a) Estado inicial e (b) Estado final.

**Observação 2.2** *Um dos principais pontos fracos das redes de Petri é sua complexidade. Por isto, o desenvolvimento de métodos que possibilitem a sua redução, preservando suas propriedades e, conseqüentemente, facilitando sua análise, é de suma importância. Algumas técnicas de redução são: fusão de lugares em série, fusão de transições em paralelo, fusão de transições em série e fusão de lugares em paralelo (Murata, 1989).*

### 2.3.5 Comentários Finais

Um sistema de manufatura deve possuir duas características: flexibilidade e agilidade. A flexibilidade é a habilidade de ajustar o sistema às preferências dos clientes e a agilidade é a velocidade de configuração dos sistemas determina pela demanda (Zhou e Venkatesh, 1999). O desempenho de sistemas de produção envolvendo linhas de produção simples, sistemas de manufatura flexíveis e outras aplicações foi

estudado em Al-Jaar e Desrochers (1990), Freedman (1991), Hillion e Proth (1989), Mazigh et al. (1993), Zhou et al. (1993).

As ferramentas convencionais de modelagem (equações diferenciais, a diferenças, entre outras) muitas vezes não são apropriadas para lidar com sistemas de manufatura flexíveis, que podem ser caracterizados como sistemas dirigidos a eventos discretos. A modelagem e análise de sistemas de manufatura, representação de linhas de produção simples com armazenamento, sistemas de produção automotiva, sistemas de manufatura flexíveis, linhas de montagem automatizadas, sistemas de compartilhamento de recursos, sistemas de manufatura “*just-in-time*”<sup>2</sup> e “*kanban*”<sup>3</sup>, entre outras aplicações, têm sido, em geral, feitas utilizando redes de Petri.

Seguindo essa linha de desenvolvimento, tem-se que, nesse trabalho, o sistema de automação da máquina injetora de plásticos será considerado como um sistema a eventos discretos, tendo em vista que a sua dinâmica é definida a partir da ocorrência de eventos. O modelo a ser adotado será o de redes de Petri, já que esta mostra-se uma ferramenta adequada para esta finalidade por se tratar de um sistema com diversas temporizações, eventos assíncronos e processos de sincronização. As funções da máquina injetora serão decompostas em módulos e será gerada a sua derivação em modelos de rede de Petri modulares.

---

<sup>2</sup>Just-in-time é um modelo de gestão da produção, em que os insumos são fornecidos apenas no momento em que serão processados.

<sup>3</sup>Kanban é o sistema de puxar a produção a partir da demanda, produzindo somente os itens necessários, nas quantidades necessárias e no momento necessário, baseado no uso de cartões para a transmissão de informações entre os centros produtivos.

# Capítulo 3

## Máquina injetora IHP-180/350

A proposta deste capítulo é apresentar a máquina injetora IHP-180/350 (Mecânica Oriente). Essa máquina de injeção, em sua configuração original, faz o controle automático do processo de injeção e extração do molde utilizando um microprocessador MCP3. Embora a proposta desse trabalho seja iniciar o desenvolvimento de um modelo para um sistema de automação que será, no futuro, implementado utilizando um CLP e, portanto, dispensando o uso do microprocessador MCP3, as nomenclaturas e notações originalmente utilizadas serão mantidas.

Neste capítulo será feita uma revisão dos principais conceitos, instruções e funções da máquina injetora IHP-180/350, estando estruturado da seguinte forma: na seção 3.1 são apresentados os principais conceitos e as fases do processo de moldagem por injeção; na seção 3.2 são descritos seus elementos, são apresentados os tipos de máquinas de injeção, o ciclo de injeção e a operação da máquina injetora; na seção 3.3 são estudadas as funções da máquina injetora; na seção 3.4 são vistos os dispositivos de segurança e acionamento da máquina injetora; na seção 3.5 são apresentados os comandos da máquina injetora; na seção 3.6 são vistas as principais restrições da máquina injetora; e na seção 3.7 são apresentados os problemas que podem ser causados no produto devido a variações dos parâmetros controlados.

### 3.1 Moldagem de plásticos por injeção

A palavra “plástico” significa “capaz de ser moldado” (Blass, 1985). Os materiais plásticos são naturais ou artificiais, de origem orgânica, podem ser moldados quando

submetidos a determinadas condições. Seu componente básico é a resina, que pode ser classificada em termoplástica e termorrígida. Na moldagem das resinas termorrígidas, os componentes são misturados no molde ou na máquina e o endurecimento ocorre por meio de reações químicas, chamadas de reações de cura. Na moldagem das resinas termoplásticas, a resina é plastificada (amolecida) no interior do cilindro pela rosca e injetada neste estado no molde. No interior do molde, ela é solidificada por meio de resfriamento e, portanto, não ocorrem reações químicas.

As resinas termoplásticas tornam a moldagem simples e rápida por não sofrerem reações químicas durante a moldagem. Elas são utilizadas para produção de artigos de forma geométrica variada, suprimindo, eficaz e economicamente, requisitos funcionais diversos. O método clássico de conformação das resinas termoplásticas é a chamada moldagem por injeção. Na moldagem por injeção, o material plastificado é forçado para o interior de um molde frio, onde, então, sofre solidificação.

### **3.1.1 Ciclo de moldagem por injeção**

O ciclo de operação, durante a moldagem por injeção, compreende as seguintes etapas, ilustradas na figura 3.1:

1. Fechamento do molde;
2. Avanço da unidade de injeção (ou encoste);
3. Injeção, período em que a unidade de injeção permanece na posição avançada durante um tempo determinado em função da natureza do material, do tamanho e da forma do produto a ser moldado;
4. Recalque, onde é feita a compactação do material plástico para compensação da contração do mesmo no interior da cavidade do molde;
5. Recuo do canhão;
6. Dosagem do material a ser injetado;
7. Abertura do molde e extração da peça, após o produto moldado ter resfriado.

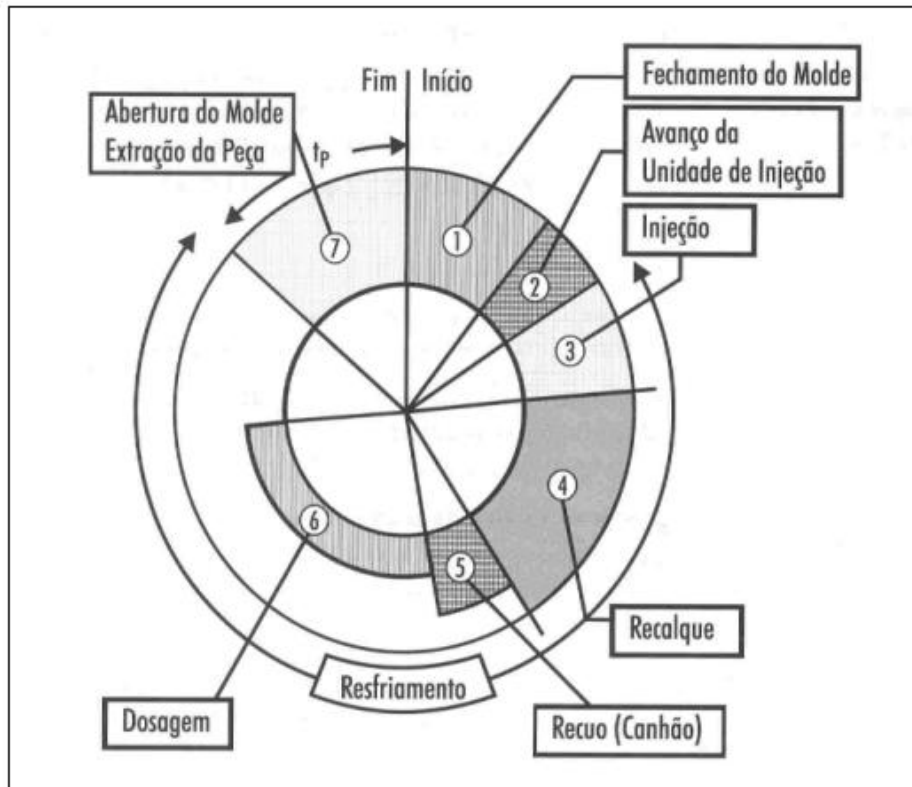


Figura 3.1: Ciclo de Injeção. Figura originalmente apresentada em Ipiranga (1998).

## 3.2 Máquinas de moldagem por injeção

A figura 3.2 apresenta um desenho esquemático de uma máquina injetora. Ela é composta de uma estrutura, onde estão fixados o sistema de injeção e o sistema de fixação, abertura e fechamento do molde. No interior desta estrutura se encontra o sistema de acionamento hidráulico e o sistema de controle.

O sistema de injeção é o responsável pela alimentação, plastificação e injeção do polímero, e está representado esquematicamente na figura 3.3. Nesta unidade o polímero é alimentado pela tremonha ou funil até o cilindro de plastificação. No interior do cilindro de plastificação encontra-se a rosca e no exterior as resistências de aquecimento, subdivididas em diferentes zonas. A plastificação do polímero ocorre no interior dos filetes da rosca. O movimento de rotação da rosca faz com que os grãos de plástico se atritem com a mesma e sofram aquecimento e cisalhamento. Assim, à medida que o material é transportado ao longo da rosca, ele vai sofrendo a plastificação e vai se homogeneizando. A rosca é subdividida em três zonas: (i) zona

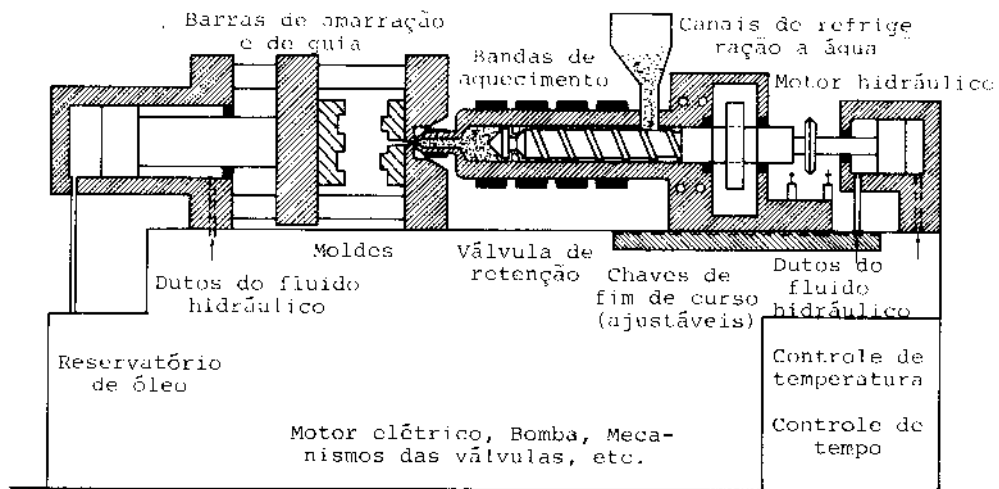


Figura 3.2: Máquina de moldagem por injeção de parafuso simples. Figura originalmente apresentada em Blass (1985).

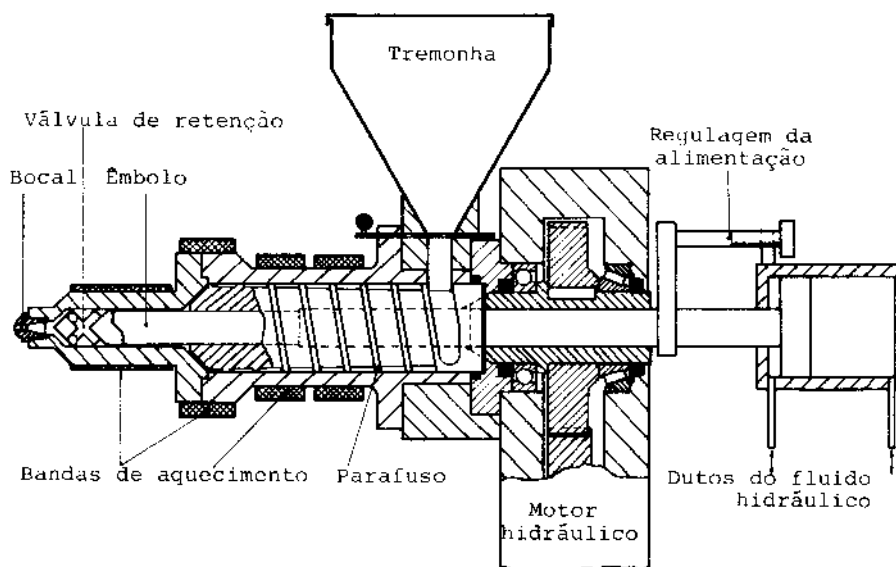


Figura 3.3: Injetora coaxial de parafuso e êmbolo. Figura originalmente apresentada em Blass (1985).

de alimentação, em que a altura do filete é constante, devendo o material ser apenas transportado, não podendo sofrer deformação; (ii) zona de compressão, em que a altura do filete diminui progressivamente e o material sofre plastificação; (iii) zona de calibragem ou de homogeneização, onde o material já plastificado é homogeneizado. A outra função da rosca é avançar para injetar e compactar o plástico no molde. As zonas de aquecimento do cilindro servem para controlar a temperatura, sendo compostas de resistências elétricas que fornecem calor para compensar as perdas de calor do cilindro para o ambiente, tanto por radiação como por convecção, e para a máquina (por condução). A temperatura de cada zona deve ser regulada de acordo com o material, as características da ferramenta e as características da peça.

O sistema de fixação, abertura e fechamento do molde é composto de uma placa fixa, uma placa móvel, guias das placas (normalmente colunas), pistão hidráulico para abertura e fechamento da placa móvel e pistão do sistema extrator. Neste sistema, a parte do molde que recebe o plástico do sistema de injeção é montada na placa fixa e a parte do molde, por onde normalmente a peça é extraída, é montada na placa móvel. Em cada ciclo, a placa móvel se desloca em direção à placa fixa para fechar o molde, acionada por um pistão hidráulico. Durante a injeção, a força de fechamento ou travamento do molde precisa ser suficiente para manter as duas faces do molde encostadas. Caso contrário, o material penetrará na fresta entre as partes do molde e formará rebarba na peça. O sistema de extração também é acionado por um pistão hidráulico e pode ser movimentado durante ou após a abertura da placa móvel, dependendo das características da peça a ser extraída.

O sistema hidráulico é composto de bomba hidráulica, válvulas, pistões, motor hidráulico e outros. A bomba hidráulica fornece a pressão necessária ao óleo para mover os pistões e o motor. As válvulas recebem sinais do sistema de controle e permitem ou não a passagem de óleo, conforme a necessidade.



### 3.2.1 Etapas de moldagem da máquina injetora

As etapas de moldagem<sup>1</sup> da máquina injetora são descritas a seguir. Para iniciar o processo, o molde deve estar aberto, o canhão recuado e a tremonha abastecida, conforme ilustrado na figura 3.4. O primeiro passo é encher a unidade de injeção com resina plástica da tremonha (dosagem). Em seguida, deve-se fechar o molde, o que é realizado pelo sistema hidráulico, que movimenta a placa móvel na direção da fixa (figura 3.5). Na fase seguinte dá-se o avanço da unidade de injeção (canhão), ocorrendo o encoste, conforme mostrado na figura 3.6. A seguir ocorre a injeção, que apresenta as etapas de transferência e recalque, como mostra a figura 3.7.

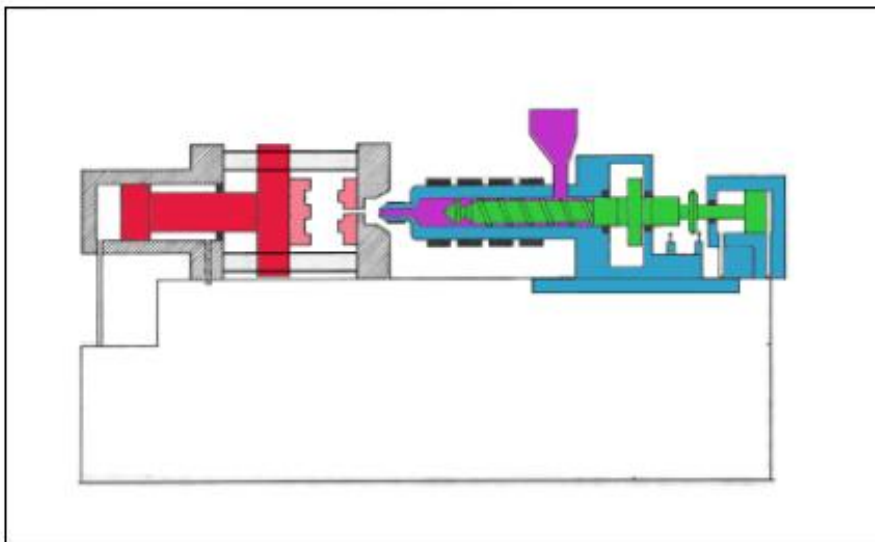


Figura 3.4: Início do processo de injeção: placas abertas e canhão recuado.

---

<sup>1</sup>As figuras 3.4 a 3.11 utilizadas para ilustrar as etapas da máquina injetora foram originalmente apresentadas em Canabrava Filho (1998)

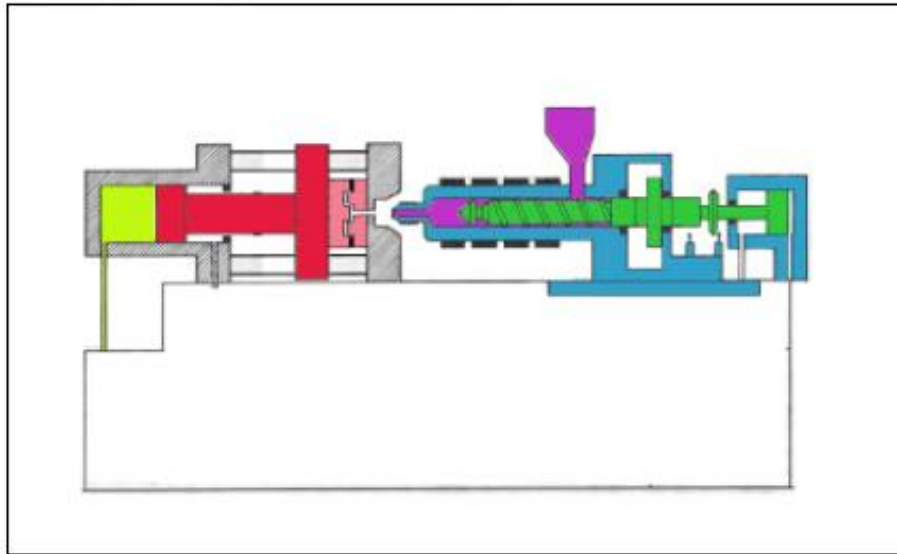


Figura 3.5: Primeira etapa do processo de injeção: sistema hidráulico movimenta a placa móvel em direção à placa fixa (fechamento).

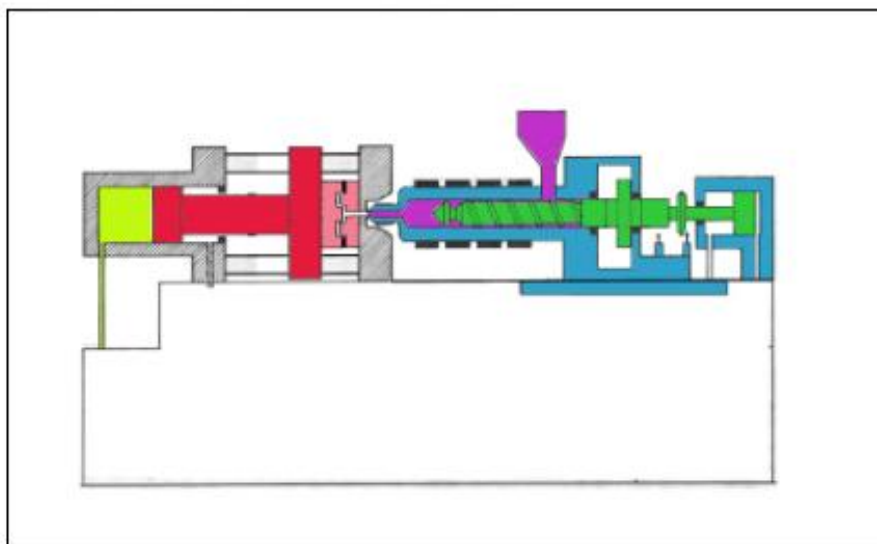


Figura 3.6: Segunda etapa do processo de injeção: avanço da unidade de injeção (canhão).

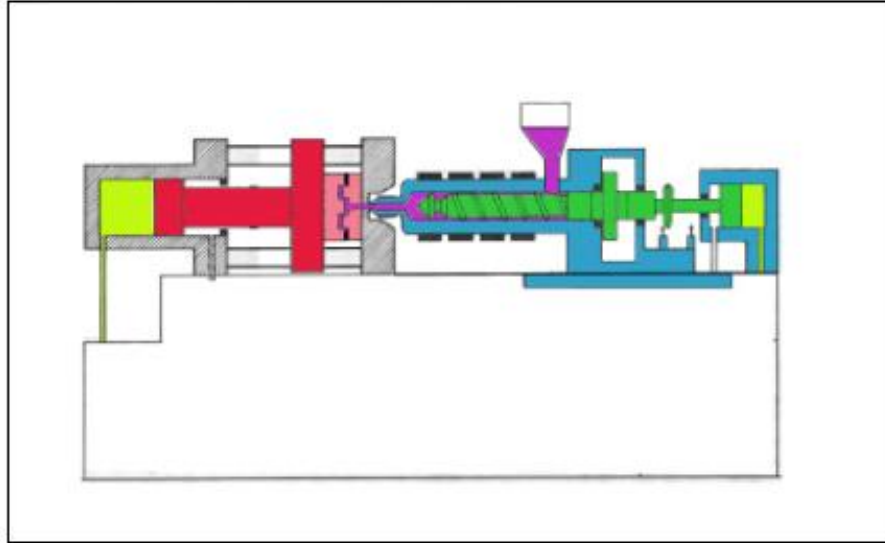


Figura 3.7: Terceira e quarta etapas do processo de injeção: Avanço da rosca pelo sistema hidráulico, realizando a injeção por transferência e, em seguida, o recalque.

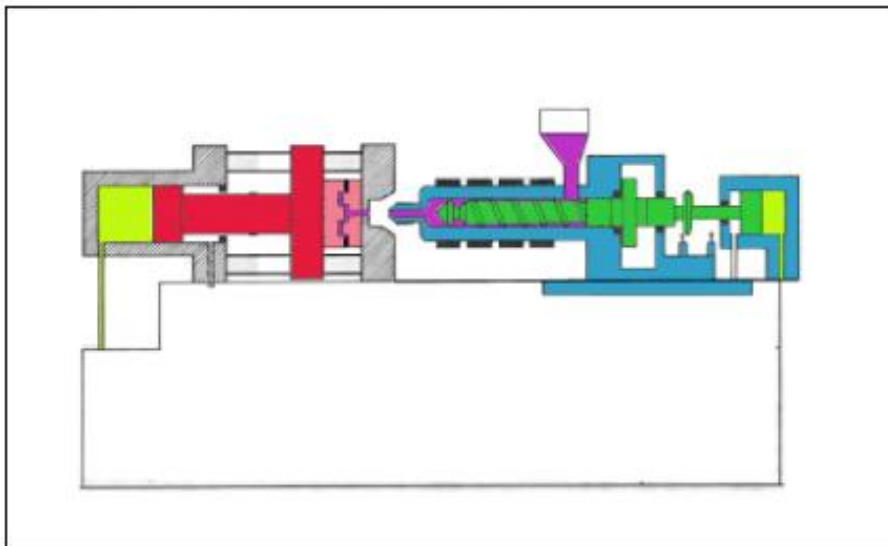


Figura 3.8: Quinta etapa do processo de injeção: recuo da unidade de injeção (cânhamo), enquanto, simultaneamente, há a refrigeração do produto.

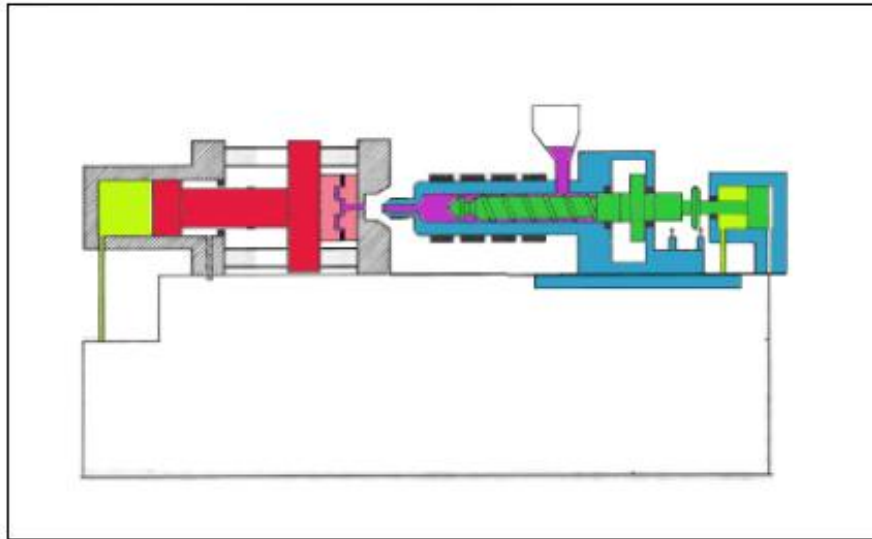


Figura 3.9: Sexta etapa do processo de injeção: recuo da rosca, dosagem do polímero em conjunto com o movimento de plastificação.

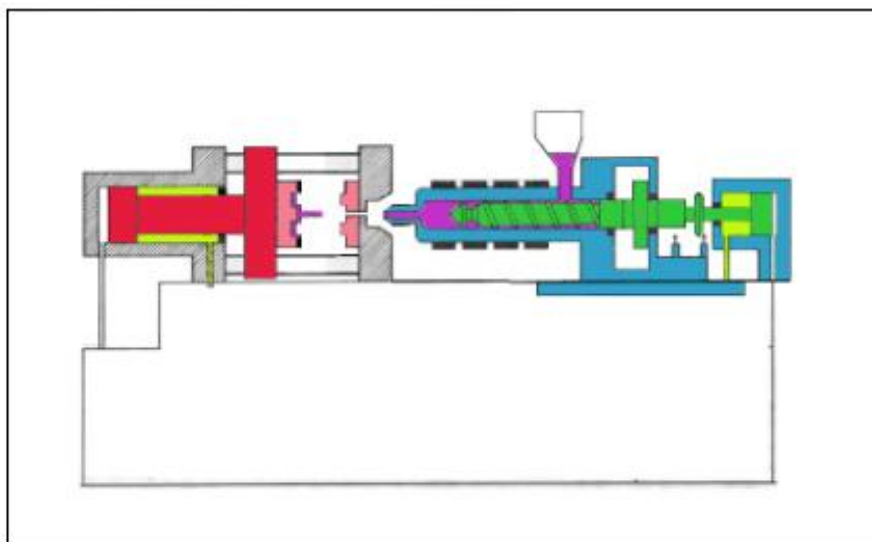


Figura 3.10: Sétima etapa do processo de injeção: abertura das placas, acionada por sistema hidráulico.

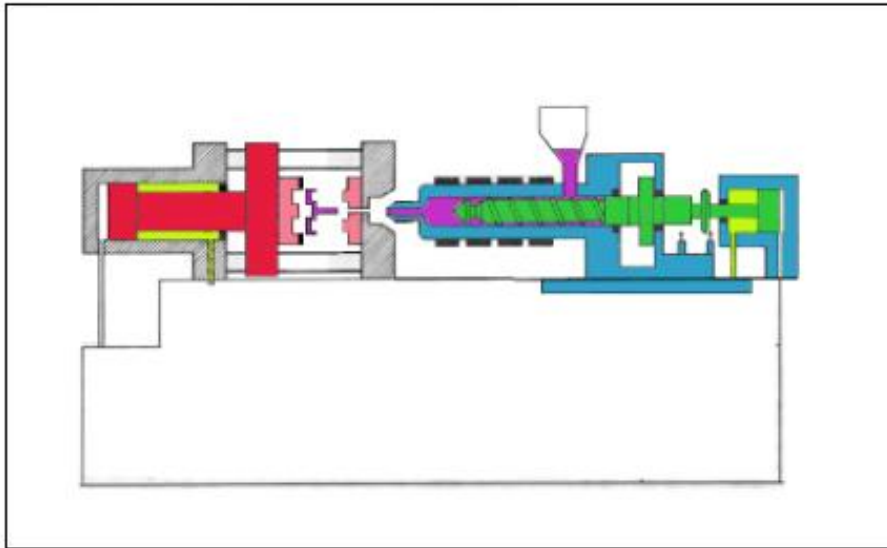


Figura 3.11: Parte final da última etapa do processo de injeção: extração da peça.

Após o recalque é possível efetuar o recuo do canhão enquanto ocorre a refrigeração do produto (figura 3.8). A próxima operação é a dosagem de polímero que se dá em conjunto com o movimento de plastificação (figura 3.9). Após o resfriamento da peça ocorre a abertura do molde, na qual a placa móvel da máquina é acionada por sistema hidráulico (figura 3.10). Em seguida ocorre a extração da peça (figura 3.11).

### 3.3 Máquina injetora IHP-180/350

Nesta seção serão vistas as funções da máquina injetora IHP-180/350, conforme o seu manual técnico (Oriente, 1998). Uma representação esquemática da máquina injetora IHP-180/350 é apresentada na figura 3.12.

As principais funções da máquina injetora IHP-180/350 são:

1. Fechamento;
2. Encoste;
3. Injeção;
4. Retorno do conjunto injetor;



Figura 3.12: Máquina injetora. Figura originalmente apresentada em Canabrava Filho (1998).

5. Dosagem;
6. Descompressão;
7. Abertura do molde;
8. Extração.

### 3.3.1 Fechamento

O fechamento é o deslocamento da placa móvel em direção à placa fixa, provocado pelo conjunto de fechamento da máquina.

O ajuste do sistema automático de fechamento requer a especificação dos seguintes parâmetros: tempo, pressão, velocidade e posições. Os valores intermediários desses parâmetros estão definidos na tabela 3.1, sendo o seu ajuste feito de acordo com a figura 3.13, da seguinte forma:

(i) O tempo de retardo de fechamento,  $T1(01)$ , é o tempo que a placa móvel permanece parada no final do curso de abertura.

(ii) Existem três pressões atuantes nesta função. A pressão  $P1(01)$  é a pressão de início de fechamento, podendo trabalhar em conjunto com as velocidades  $V1(01)$ ,

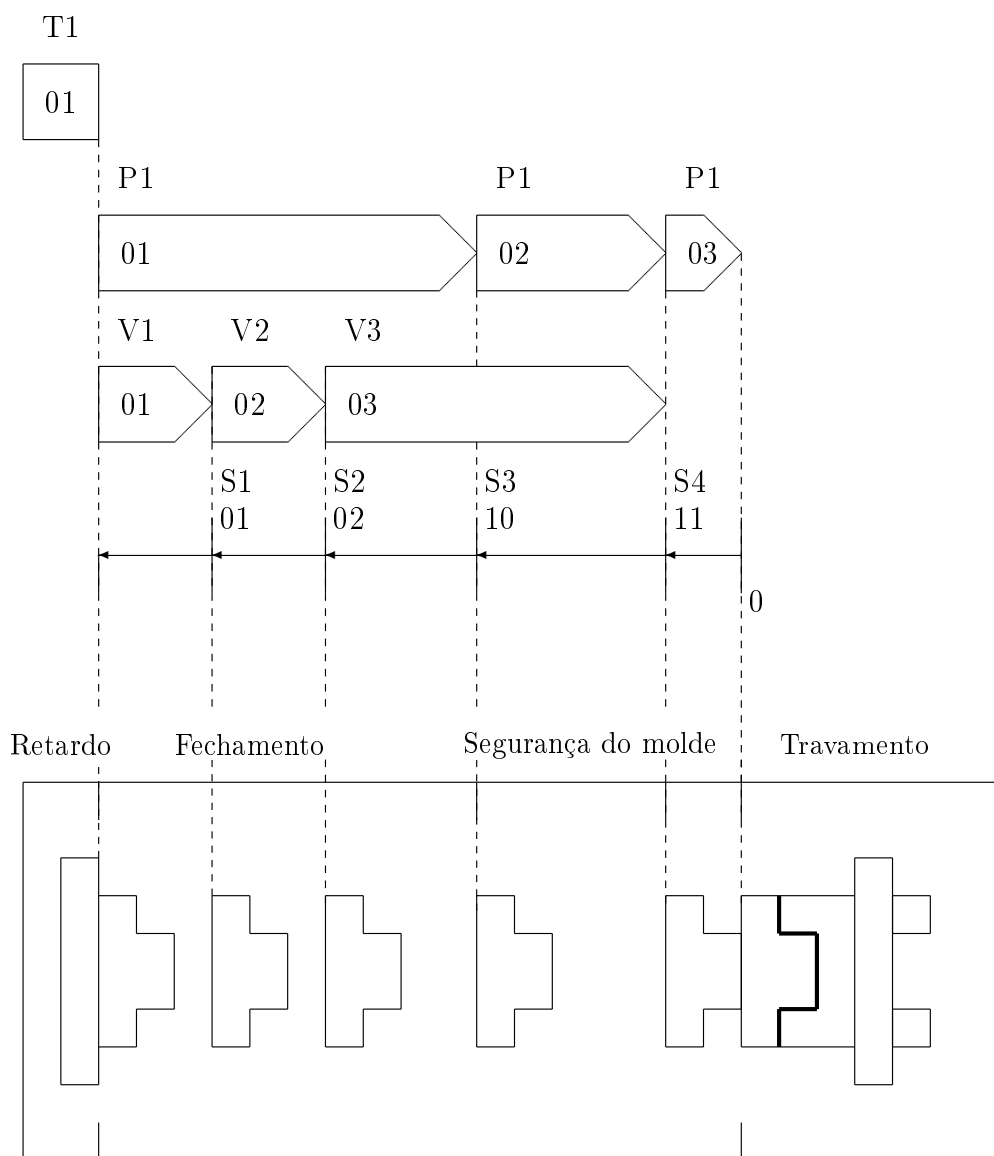


Figura 3.13: Função fechamento. Figura originalmente apresentada em Oriente (1998).

V2(02) e V3(03) e com as posições S1(01), S2(02) e S3(10). A pressão P1(02) é a pressão de segurança do molde, e pode também trabalhar com as três velocidades. Essa pressão tem seu início de atuação na posição S3(10). Finalmente, tem-se P1(03), que é a pressão de travamento, com seu início determinado pela posição S4(11), sendo a velocidade nesse trecho fixada pelo fabricante.

(iii) Existem três velocidades atuantes nesta função: V1(01), velocidade inicial de fechamento, que pode ou não atuar como freio; V2(02), velocidade intermediária e/ou de segurança do molde; V3(03), velocidade final de fechamento final de fechamento, que pode atuar como freio ou de segurança do molde. Estas três velocidades podem atuar com a pressão P1(01) ou P1(02) e são determinadas em função das posições S1(01) e S2(02).

(iv) Existem quatro posições de comutação de pressão ou de velocidade no fechamento: sendo que as posições S1(01) e S2(02) só atuam sobre as velocidades V1(01), V2(02) e V3(03). S1(01) é o ponto de comutação entre as velocidades V1(01) e V2(02), S2(02) é o ponto de comutação entre as velocidades V2(02) e V3(03). A posição S3(10) só atua sobre a pressão, determinando o final da pressão P1(01) e início da P1(02), podendo atuar com as três velocidades, porém não interferindo no ponto de comutação entre elas. Finalmente, S4(11) é a posição que determina o final de segurança do molde e início do travamento. Atua com pressão P1(03) e sua velocidade é fixada pelo fabricante.

**Observação 3.1** *A segurança do molde faz com que o fechamento da máquina se dê a uma baixa pressão, até as faces do molde se tocarem. Após o término da segurança do molde, começa o início do travamento, em que se dá uma elevação da pressão.*

O sistema de controle a ser desenvolvido para a máquina deve permitir trabalhar com variações de velocidade e pressão, reguláveis de acordo com as necessidades do processo, ou ainda, com velocidades e pressões uniformes. Um exemplo de processo de fechamento complexo é apresentado na figura 3.14. Neste exemplo, o deslocamento da placa móvel tem velocidade inicial V1(01) e pressão P1(01). Quando a placa móvel atinge a posição S3(10) a pressão muda para P1(02), porém



Tabela 3.1: Parâmetros de regulagem da função fechamento.

Parâmetro	Nomenclatura	Definição
Tempo	T1(01)	Tempo de retardo de fechamento
Pressão	P1(01)	Pressão inicial de fechamento
	P1(02)	Pressão de segurança do molde
	P1(03)	Pressão de travamento
Velocidade	V1(01)	Velocidade inicial de fechamento/freio inicial de fechamento
	V2(02)	Velocidade intermediária
	V3(03)	Velocidade final de fechamento/freio final de fechamento
Posição	S1(01)	1ª posição de variação de velocidade
	S2(02)	2ª posição de variação de velocidade
	S3(10)	Posição de término da pressão inicial de fechamento e início da pressão de segurança do molde
	S4(11)	Posição de término da pressão de segurança do molde e início da pressão de travamento

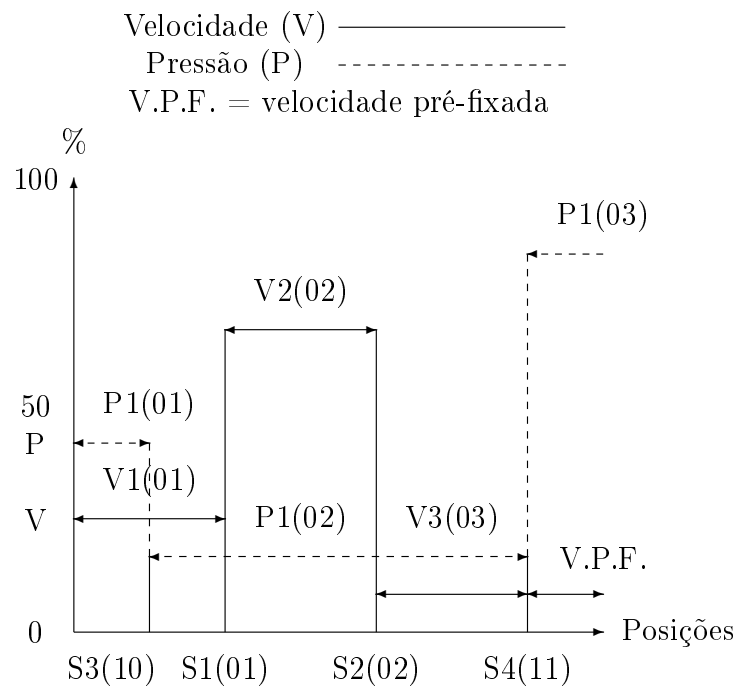


Figura 3.14: Processo de fechamento complexo. Figura originalmente apresentada em Oriente (1998).

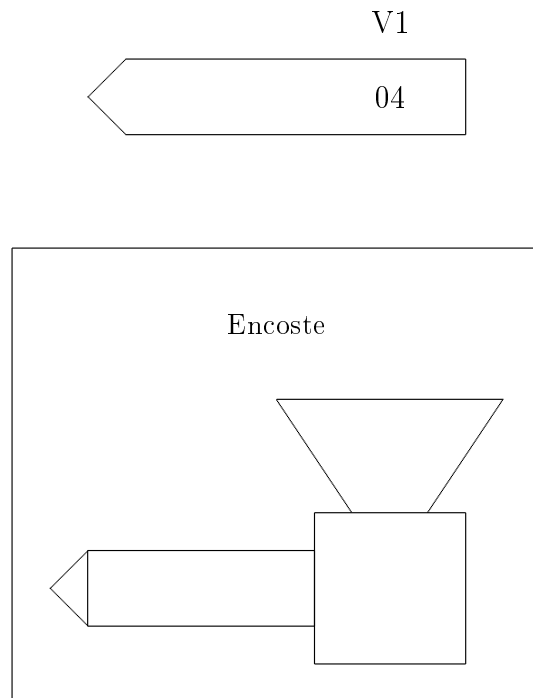


Figura 3.15: Função encoste. Figura originalmente apresentada em Oriente (1998).

a velocidade permanece igual a  $V1(01)$ . No ponto  $S1(01)$  (definido pelo usuário) a velocidade de deslocamento muda para  $V2(02)$ , porém não há variação na pressão. Note que o valor  $S3(10)$  é menor que  $S2(02)$ , isto é, embora possa haver mudança na velocidade, o sistema de fechamento já entrou na pressão de segurança. A velocidade mudará, ainda, nas posições  $S2(02)$  e  $S4(11)$ ; na primeira a nova velocidade pode ser especificada pelo usuário enquanto que na segunda, por se tratar do trecho final do fechamento, a velocidade é determinada pelo fabricante. Em  $S4(11)$  haverá também uma última mudança de pressão.

### 3.3.2 Encoste

O encoste é o avanço da unidade de injeção em direção ao molde da máquina (figura 3.15). Os parâmetros de regulação da função encoste são a pressão de encoste (fixada pelo fabricante) e a velocidade  $V1(04)$  de encoste (ajustável até o limite pré-fixado de 30%).

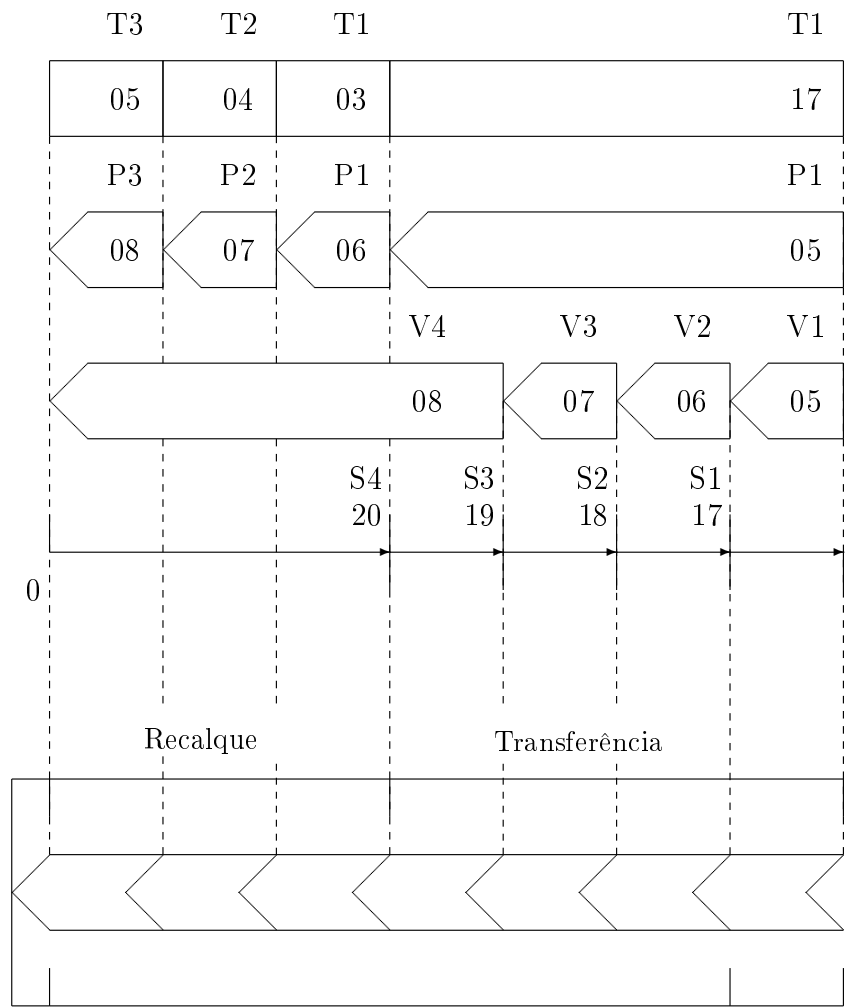


Figura 3.16: Função injeção.

### 3.3.3 Injeção

Após assegurado o encoste da unidade de injeção, libera-se a pressão de injeção. Isto é feito por meio de um pressostato que deve ser checado antes de se iniciar a injeção.

A injeção é a função que engloba transferência e recalque. A transferência é o transporte do material existente no interior do cilindro plastificador à frente da rosca plastificadora, para o interior do molde. O recalque é a compactação do material plástico para compensação de contração do mesmo no interior da cavidade do molde, ocorrendo logo após o término da transferência. É necessário para evitar que o material retorne para fora da cavidade e ajuda a compensar a contração da peça, durante o resfriamento. Isto é feito introduzindo-se uma quantidade extra de material.

Após o processo de injeção, quando é necessária a limpeza da máquina ou a eliminação de material degradado, há um sistema que permite a descarga de material do cilindro plastificador, sem encostar o bico no molde, esse evento é referido como a descarga do cilindro, que para ser executado é necessário somente para isto que a seleção da função esteja ligada, no painel de comandos. A descarga é usada para esvaziar o cilindro na região à frente da rosca.

O ajuste do sistema automático de injeção requer a especificação dos parâmetros tempo, pressão, velocidade e posições de comutação, mostrados na tabela 3.2, sendo ajustados de acordo com a figura 3.16. Na etapa de transferência os parâmetros a serem ajustados são os seguintes:

**Tempo:**

O tempo de transferência, T1(17), é o tempo durante o qual atua uma única pressão P1(05), quatro velocidades V1(05), V2(06), V3(07) e V4(08) determinadas pelas três posições S1(17), S2(18) e S3(19).

**Pressão:**

Existe uma única pressão P1(05) atuante nesta etapa, trabalhando dentro do tempo T1(17), podendo atuar com quatro velocidades V1(05), V2(06), V3(07)

Tabela 3.2: Parâmetros de regulagem da função injeção.

Parâmetro	Nomenclatura	Definição
Tempo	T1(17)	Tempo de transferência
	T1(03)	1º tempo de recalque
	T2(04)	2º tempo de recalque
	T3(05)	3º tempo de recalque
Pressão	P1(05)	Pressão de transferência
	P1(06)	1ª pressão de recalque
	P2(07)	2ª pressão de recalque
	P3(08)	3ª pressão de recalque
Velocidade	V1(05)	1ª velocidade de transferência
	V2(06)	2ª velocidade de transferência
	V3(07)	3ª velocidade de transferência
	V4(08)	Última velocidade de transferência/velocidade de recalque
Posição	S1(17)	Ponto de comutação entre V1(05) e V2(06)
	S2(18)	Ponto de comutação entre V2(06) e V3(07)
	S3(19)	Ponto de comutação entre V3(07) e V4(08)
	S4(20)	Posição final de transferência/início de recalque

e V4(08) determinadas pela posição inicial e pelas três posições S1(17), S2(18) e S3(19).

#### **Velocidade:**

Existem quatro velocidades atuantes nesta função: (i) V1(05), primeira velocidade de transferência; (ii) V2(06), segunda velocidade de transferência; (iii) V3(07), terceira velocidade de transferência; (iv) V4(08), última velocidade de transferência e atuante também na função de recalque. Estas quatro velocidades são determinadas pela posição inicial e pelas três posições S1(17), S2(18) e S3(19).

#### **Posições:**

Existem três posições de comutação de velocidade na etapa de transferência: (i) S1(17), que é o ponto de comutação entre V1(05) e V2(06); (ii) S2(18), que é o ponto de comutação entre V2(06) e V3(07); (iii) S3(19), que é o ponto de comutação entre V3(07) e V4(08).

Na etapa de recalque os parâmetros são os seguintes:

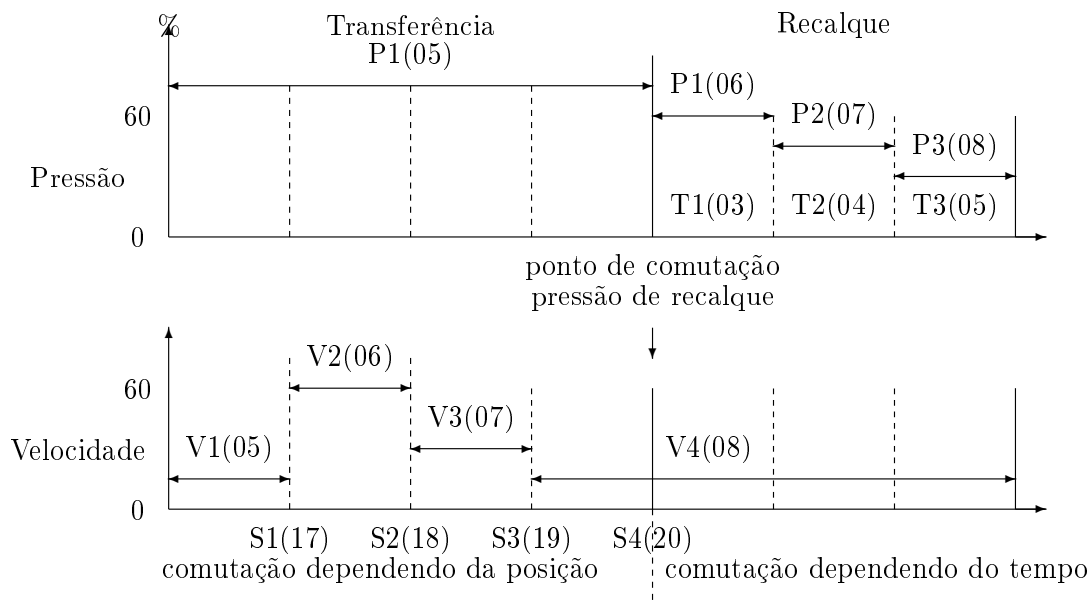


Figura 3.17: Processo de injeção complexo. Figura originalmente apresentada em Oriente (1998).

### Velocidade:

Existe apenas uma velocidade  $V4(08)$  atuante nesta etapa, que é a última velocidade da função transferência, permanecendo inalterada.

### Tempo:

Os três tempos atuantes são: (i)  $T1(03)$ , que é o tempo de atuação da pressão  $P1(06)$ . É determinado pela posição  $S4(20)$  ou pelo término do tempo  $T1(17)$ , pela situação que ocorrer primeiro; (ii)  $T2(04)$ , que ocorre após  $T1(03)$ , atuando com uma pressão  $P2(07)$ ; (iii)  $T3(05)$ , que ocorre após  $T2(04)$ , atuando com uma pressão  $P3(08)$ .

### Pressão:

Existem três pressões atuantes nesta etapa: (i) a pressão  $P1(06)$ , que atua no tempo  $T1(03)$  a partir do acionamento da posição  $S4(20)$  ou do término do tempo  $T1(17)$ , pela situação que ocorrer primeiro; (ii) a pressão  $P2(07)$ , que atua no tempo  $T2(04)$ ; (iii) finalmente, tem-se  $P3(08)$ , que é a pressão atuante no tempo  $T3(05)$ .

### Posição:

Existe uma única posição  $S4(20)$  atuante nesta etapa, trabalhando com três tempos, três pressões e uma única velocidade.

O sistema de controle a ser desenvolvido para a máquina deve permitir traba-

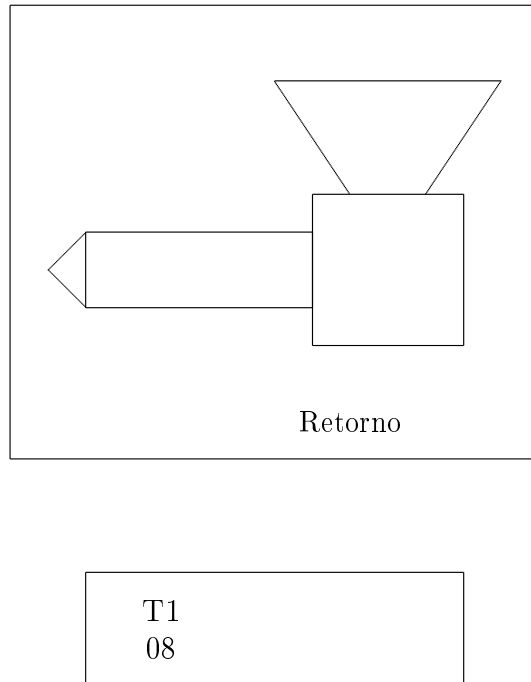


Figura 3.18: Função retorno. Figura originalmente apresentada em Oriente (1998).

lhar com variações de velocidade e pressão, reguláveis de acordo com as necessidades do processo, ou ainda, com velocidades uniformes, na etapa de transferência, e com pressões uniformes, na etapa do recalque. Um exemplo de processo de injeção complexo é apresentado na figura 3.17. Nesse exemplo, a injeção por transferência tem velocidade inicial  $V1(05)$  e pressão  $P1(05)$ , esta última mantendo-se constante durante todo o processo de transferência. No ponto  $S1(17)$  (definido pelo usuário), a velocidade de deslocamento muda para  $V2(06)$ . A velocidade mudará, ainda, nas posições  $S2(18)$ , para  $V3(07)$ , e  $S3(19)$ , para  $V4(08)$ , também definidas pelo usuário. A injeção por recalque tem início com a mudança da pressão  $P1(05)$  para a pressão  $P1(06)$ , a partir da posição  $S4(20)$ , ainda com velocidade  $V4(08)$ , que permanecerá por todo o recalque. A pressão mudará, ainda, para  $P2(07)$  após um intervalo de tempo igual a  $T1(03)$  e para  $P3(08)$  após um intervalo de tempo igual a  $T2(04)$ , permanecendo nesse valor por um intervalo de tempo igual a  $T3(05)$ .

### 3.3.4 Retorno do conjunto injetor

O retorno é o recuo da unidade de injeção, em relação ao molde. Ocorre após uma descompressão, que puxa a rosca para trás, com o objetivo de tirar a pressão do

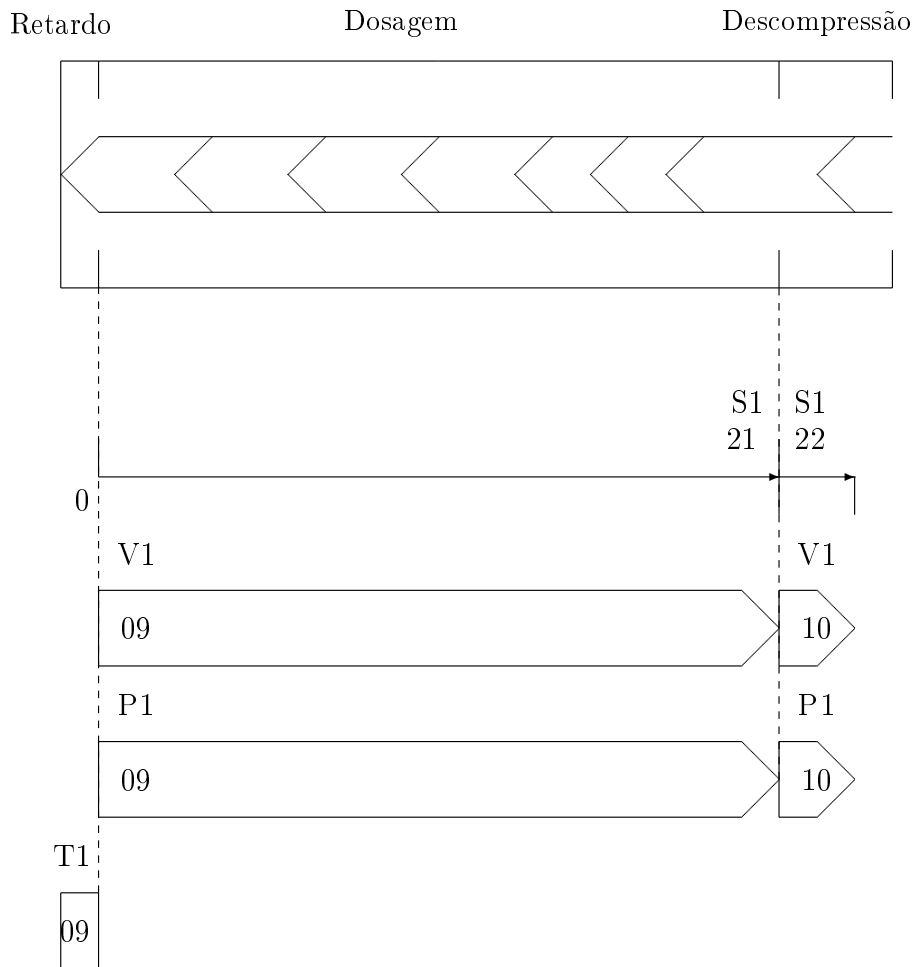


Figura 3.19: Funções Dosagem e Descompressão. Figura originalmente apresentada em Oriente (1998).

plástico que está na saída do cilindro, evitando, assim, que o material fique pingando. Essa descompressão é determinada pelo intervalo de tempo T1(09), mostrado na figura 3.19. Os parâmetros de regulagem da função retorno são T1(08), que é o tempo de comando da função; a pressão (fixada pelo fabricante) e a velocidade V1(04), a mesma de encoste (figura 3.18).

### 3.3.5 Dosagem

A dosagem é a função empregada para dosar o material para o ciclo seguinte, e ocorre após o término da função recalque. Durante esta função verificam-se dois movimentos conjugados da rosca plastificadora: o de retorno (horizontal) e o de rotação (sentido anti-horário). Esses movimentos permitem que a rosca carregue o cilindro plastificador com o material proveniente do funil alimentador, na quantidade



Tabela 3.3: Parâmetros de regulagem da função dosagem.

Parâmetro	Nomenclatura	Definição
Tempo	T1(09)	Tempo de retardo da dosagem (opcional)
Pressão	P1(09)	Pressão de dosagem
Velocidade	V1(09)	Velocidade de dosagem (rotação da rosca plastificadora)
Posição	S1(21)	Término da dosagem/início da descompressão

pré-determinada, necessária para a próxima injeção.

O ajuste do sistema automático de dosagem requer a especificação dos parâmetros pressão, velocidade e posições, mostrados na tabela 3.3 e ajustados de acordo com a figura 3.19, da seguinte forma:

**Pressão:**

Existe apenas uma pressão P1(09) atuante nesta função, trabalhando em conjunto com a velocidade V1(09) desde o início do curso até a posição S1(21).

**Velocidade:**

Existe somente uma velocidade (rotação da rosca plastificadora) V1(09) atuante nesta função, com pressão P1(09) desde o início do curso até a posição S1(21).

**Posição:**

Existe apenas uma posição S1(21) atuante nesta função, que determina o curso de dosagem (quantidade de material). Durante o curso de dosagem atuam a velocidade V1(09) e a pressão P1(09).

### 3.3.6 Descompressão

A descompressão pode ser realizada por duas funções, (i) a função 1, que ocorre após o término da função dosagem e consiste no retorno da rosca plastificadora até a posição S1(22) sem que haja simultaneamente o movimento de rotação, bastando para isto que a seleção da função descompressão esteja ligada em P1; (ii) a função 2, na qual a rosca plastificadora carrega e descomprime ao mesmo tempo.

O ajuste do sistema automático de descompressão requer a especificação dos parâmetros pressão, velocidade e posição, mostrados na tabela 3.4, sendo realizados

Tabela 3.4: Parâmetros de regulagem da função descompressão.

Parâmetro	Nomenclatura	Definição
Pressão	P1(10)	Pressão de descompressão
Velocidade	V1(10)	Velocidade de descompressão
Posição	S1(22)	Término da descompressão

em conformidade com a figura 3.19, qual seja:

**Pressão:**

Existe apenas uma única pressão P1(10) atuante nesta função, trabalhando com uma velocidade V1(10), a partir da posição S1(21) até a posição S1(22).

**Velocidade:**

Existe apenas uma única velocidade V1(10) atuante nesta função, trabalhando com uma pressão P1(10), a partir da posição S1(21) até a posição S1(22).

**Posição:**

Existem duas posições, S1(21), que determina o início da função, e S1(22), que determina o seu término. Durante esse curso atuam a velocidade V1(10) e a pressão P1(10).

### 3.3.7 Abertura do molde

A abertura é o deslocamento da placa móvel em sentido contrário ao da função fechamento.

O ajuste do sistema automático de abertura requer a especificação dos parâmetros tempo, pressão, velocidade e posições mostrados na tabela 3.5, sendo realizado em conformidade com a figura 3.20, isto é:

**Tempo:**

O tempo T1(10) atuante nesta função é o tempo que a placa móvel permanece parada após o término da função recalque e antes do início da função abertura.

**Pressão:**

Existe apenas uma única pressão P1(11) atuante nesta função, trabalhando com três velocidades V1(11), V2(12) e V3(13) e três posições S1(03), S2(04) e S3(05).

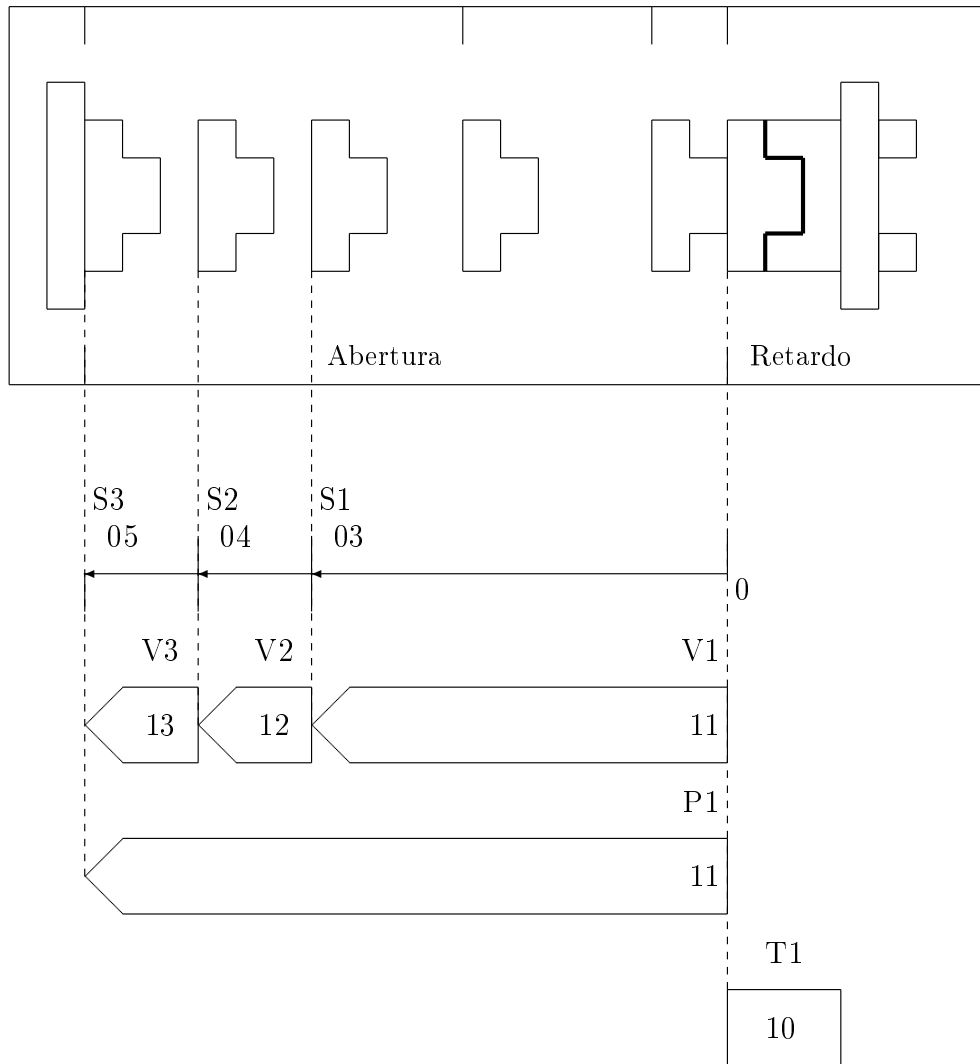


Figura 3.20: Função abertura. Figura originalmente apresentada em Oriente (1998).

Tabela 3.5: Parâmetros de regulação da função abertura.

Parâmetro	Nomenclatura	Definição
Tempo	T1(10)	Tempo de retardo da abertura
Pressão	P1(11)	Pressão de abertura
Velocidade	V1(11)	Velocidade inicial de abertura/freio inicial de abertura
	V2(12)	Velocidade intermediária
	V3(13)	Velocidade final de abertura/freio final de abertura
Posição	S1(03)	1ª posição de variação de velocidade
	S2(04)	2ª posição de variação de velocidade
	S3(05)	Posição limite de abertura

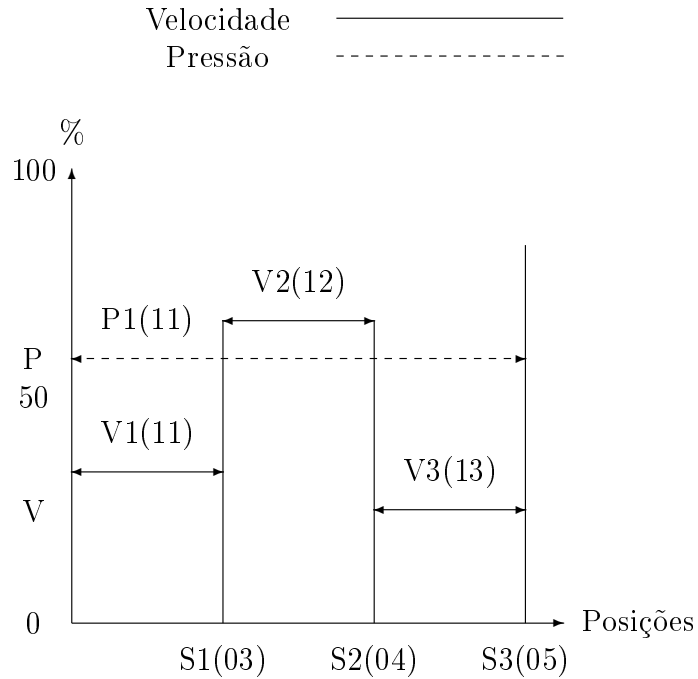


Figura 3.21: Processo de abertura complexo. Figura originalmente apresentada em Oriente (1998).

#### **Velocidade:**

Existem três velocidades atuantes nesta função, todas trabalhando com a pressão  $P1(11)$ : (i)  $V1(11)$ , velocidade inicial de abertura/freio inicial de abertura, atuando até a posição  $S1(03)$ ; (ii)  $V2(12)$ , velocidade intermediária da função, atuando a partir da posição  $S1(03)$  até a posição  $S2(04)$ ; (iii)  $V3(13)$ , velocidade final de abertura/freio final de abertura, atuando a partir da posição  $S2(04)$  até a posição  $S3(05)$ .

#### **Posições:**

Existem três posições de comutação de velocidade. (i) a posição  $S1(03)$ , que é o ponto de comutação entre as velocidades  $V1(11)$  e  $V2(12)$ ; (ii) a posição  $S2(04)$ , que representa o ponto de comutação entre  $V2(12)$  e  $V3(13)$ ; e (iii)  $S3(05)$ , que é a posição limite de abertura da máquina.

O sistema de controle a ser desenvolvido para a máquina deve permitir trabalhar com variação de velocidade, que deve ser regulada de acordo com as necessidades do processo, ou ainda, com velocidades uniformes. Um exemplo de um processo de abertura é apresentado na figura 3.21 (Oriente, 1998). Nesse exemplo,

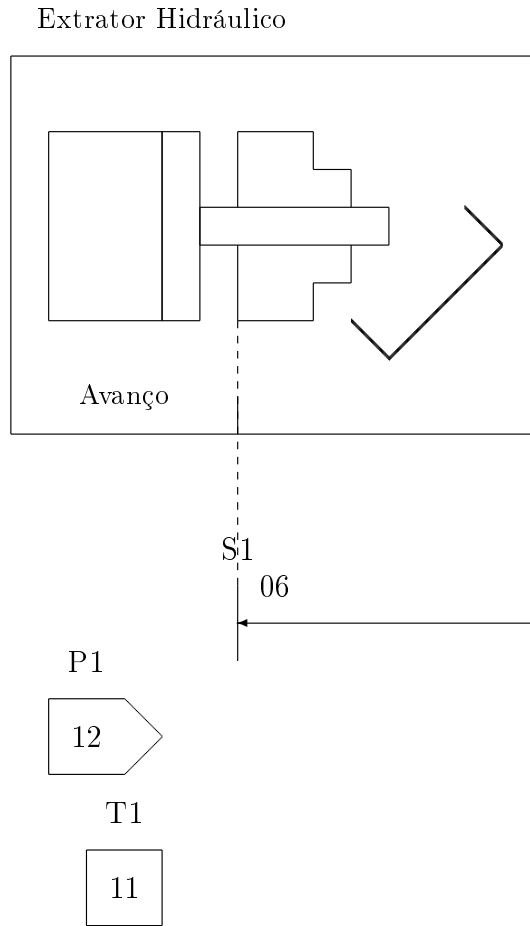


Figura 3.22: Função extração avanço. Figura originalmente apresentada em Oriente (1998).

o deslocamento da placa móvel tem velocidade inicial  $V1(11)$  e pressão  $P1(11)$ ; esta última atuando durante todo o processo de abertura. No ponto  $S1(03)$  (definido pelo usuário) a velocidade de deslocamento muda para  $V2(12)$  e no ponto  $S2(04)$  (também definido pelo usuário), a velocidade de deslocamento muda para  $V3(13)$ , até a posição  $S3(05)$ , que é a posição limite de abertura da máquina.

### 3.3.8 Extração

O extrator hidráulico é um sistema acionado por um cilindro hidráulico exclusivo e tem por finalidade extrair a peça moldada após a placa móvel ter percorrido uma determinada distância de abertura. A extração é dividida em extração avanço e extração retorno.

O ajuste do sistema automático de extração (avanço e retorno) requer a espe-

Tabela 3.6: Parâmetros de regulagem da função extração.

Parâmetro	Nomenclatura	Definição
Tempo	T1(11)	Tempo de avanço do extrator hidráulico
	T1(12)	Tempo de retorno do extrator hidráulico
Pressão	P1(12)	Pressão de movimento (avanço e retorno) do extrator hidráulico
Posição	S1(06)	Posição de início da função do extrator hidráulico

cificação dos parâmetros tempo, pressão e posição, definidos de acordo com a tabela 3.6 e com as figuras 3.22 e 3.23. A especificação dos parâmetros para a etapa da extração avanço deve satisfazer as seguintes exigências:

**Tempo:**

Existe um único tempo T1(11) atuante nesta função, o tempo de avanço do extrator hidráulico.

**Pressão:**

Existe uma única pressão P1(12) atuante nesta função, comandada pela posição S1(06) de início da extração hidráulica. Atua durante todo o curso de avanço do extrator hidráulico.

**Posição:**

Existe uma única posição S1(06) atuante nesta função, que determina o início da função do extrator hidráulico e representa uma posição no curso de abertura do molde. Quando a placa móvel passa por S1(06) durante a abertura, o pistão do sistema extrator se desloca para iniciar a extração.

O ajuste do sistema automático deve satisfazer às seguintes exigências:

**Tempo:**

Existe apenas um tempo T1(12) atuante nesta função, o tempo de retorno do extrator hidráulico.

**Pressão:**

A pressão P1(12) atuante durante o retorno do extrator hidráulico é a mesma que atua durante o avanço.

T1

12

Extrator Hidráulico

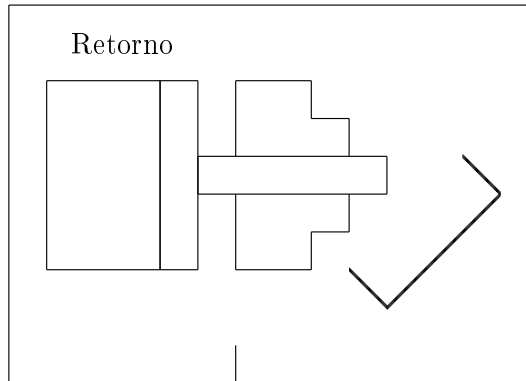


Figura 3.23: Função extração retorno. Figura originalmente apresentada em Oriente (1998).

**Observação 3.2** *As velocidades de avanço e retorno do extrator hidráulico são reguladas manualmente, independentemente uma da outra, através de registros apropriados existentes na máquina. Quanto maiores as velocidades dentro de um mesmo tempo, maiores serão os números de batidas (avanço e retorno) do extrator. Quanto maiores os tempos com as mesmas velocidades, também serão maiores os números de batidas (avanço e retorno) do extrator.*

### 3.4 Dispositivos de segurança e acionamento da máquina injetora

Para que a máquina injetora entre em operação, existe, em sua configuração original, uma chave com duas posições: bloqueado e livre. Quando a chave está na posição bloqueado, não há possibilidade de inserção de dados no programa. Essa somente poderá ser feita com a chave na posição livre. Além disso, nunca se deve dar a partida na máquina com o cilindro plastificador a frio, porque pode causar danos aos componentes do conjunto plastificador. Devem ser ajustadas as temperaturas das zonas de aquecimento do cilindro plastificador para cada tipo de material, conforme especificação do fabricante. Os ciclos automático e semi-automático de-

Tabela 3.7: Dispositivos de segurança.

Dispositivo	Função
D1	Alarmes sonoro e visual da botoneira para indicação de falhas no funcionamento da máquina.
D2	Alarme da porta de segurança, que indica que a porta está aberta. Aciona alarme sonoro e dá indicação no painel de controle. Desliga os circuitos elétricos.
D3	Alarme da proteção lateral, que indica que uma das portas de proteção do mecanismo de fechamento está aberta. Aciona o alarme sonoro, indica a falha na tela do controle e desliga o motor elétrico da máquina.
D4	Alarme de temperatura do óleo, que é acionado quando o óleo atingir 60°C. Deve acionar alarmes da botoeira, impedir o funcionamento dos ciclos automático e semi-automático e permitir o ciclo manual. Quando o óleo atingir 55°C, deve haver essa indicação na tela de controle.
D5	Alarme de interrupção de ciclo (automático e semi-automático), que interrompe os ciclos quando o tempo de ciclo é superior ao tempo pré-programado no controlador. Aciona os alarmes da botoeira.
D6	Relé térmico motor-máquina, que atua quando há sobrecarga ou falha de fase na alimentação. Aciona os alarmes da botoeira e indica a falha no painel de controle.

vem ser bloqueados em caso de falha no aquecimento das zonas. Quando o CLP for implementado pode-se trabalhar de forma idêntica à existente na configuração original ou, ainda, é possível pensar em uma solução com alguma proteção (do tipo senha) para que o programa não possa ser alterado por uma pessoa não autorizada, usando, assim, o sistema de segurança do CLP e não uma chave.

A injetora IHP-180/350 está equipada com vários dispositivos de segurança para o operador e para a máquina, descritos na tabela 3.7. Durante todo o período de funcionamento da máquina injetora, esses alarmes devem ser permanentemente checados.

Para a execução das funções descritas na seção 3.3, devem ser acionados solenóides, conforme mostrado na tabela 3.8. Por exemplo, para a execução do fechamento, devem ser acionados os solenóides S1, S8A, SIZ e S82. Ao final do



Tabela 3.8: Funções e solenóides acionados para execução das funções.

Funções	S1	S89	S8A	SIZ	S82	S88	S83	S87	S86	S85	S84
1) Fechamento	X		X	X	X						
2) Encoste	X			X			X				
3) Injeção	X			X			X				X
4) Retorno	X			X				X			
5) Dosagem	X			X			X			X	
6 a) Descompressão F1	X			X					X		
6 b) Descompressão F2	X			X					X	X	
7 a) Abertura	X			X		X					
7 b) Extrator avança	X	X									
7 c) Extrator retorna	X		X								

fechamento, e para que se inicie o encoste, deve-se, de acordo com a segunda linha da tabela 3.8, manter os solenóides S1 e SIZ acionados, desabilitar os solenóides S8A e S82 e acionar o solenóide S83. É importante ressaltar que todos os solenóides acionam as válvulas hidráulicas, representam uma interface entre o circuito elétrico e o hidráulico (anexo A).

### 3.5 Comandos

Os comandos da máquina injetora, em sua configuração original, são classificados como manual, semi-automático e automático. Para acionar o comando manual deve-se colocar a chave comutadora da botoeira em manual. Neste comando, para efetuar uma ação é necessário apertar o botão correspondente na botoeira. Deve-se utilizar o comando manual na preparação do produto e para as regulagens da máquina e do molde. Para acionar o comando semi-automático, deve-se colocar a chave comutadora da botoeira em semi-automático. Neste comando, a cada ciclo completo, a máquina pára e reinicia novamente as operações após abrir e fechar a porta de segurança do lado do operador. É usado quando o molde não extrai as peças com facilidade, precisando da intervenção externa, ou quando é usado qualquer tipo de

inserto<sup>2</sup> no molde, que precisa ser colocado antes de cada ciclo. Finalmente, para acionar o comando automático, deve-se colocar a chave comutadora da botoeira em automático. Neste comando, a cada ciclo completo da máquina, esta reiniciará automaticamente, sem intervenção externa. Quando o CLP for implementado, sugere-se manter a possibilidade de operação manual porque durante a instalação do molde, que muitas vezes ocorre antes da programação do ciclo, é preciso executar algumas vezes funções como abrir e fechar as placas e acionar o sistema extrator para verificar se a instalação está correta. Quando ocorrem problemas durante a injeção, às vezes é preciso efetuar operações como abrir as placas, recuar o canhão ou purgar (injetar fora da ferramenta para esvaziar o cilindro) também manualmente.

### 3.6 Restrições importantes

Há algumas restrições que devem ser observadas pelo operador:

1. A faixa de temperatura ideal para trabalho do óleo é entre 30°C e 40°C. A temperatura máxima permitida para o sistema é 60°C.
2. A máquina só pode ser ligada se a tensão da rede estiver dentro da tolerância de mais ou menos 10 % da tensão especificada.
3. Só pode ser dada partida na máquina quando o cilindro plastificador estiver na temperatura ideal de início do processamento (definida em função do material a ser utilizado na moldagem). Quando o CLP for implantado, o operador não deverá ter como iniciar o funcionamento da máquina (em manual por exemplo) se as condições de temperatura não o permitirem. A não observância dessa restrição pode causar sérios danos aos componentes do conjunto plastificador.

---

<sup>2</sup>Os insertos são peças que devem ser colocadas no molde para que façam parte do produto injetado. Quando a produção de peças com inserto é muito grande, usam-se geralmente robôs e dispositivos automáticos para colocar o inserto. Assim basta manter a placa móvel da máquina aberta pelo tempo suficiente para o robô colocar os insertos na posição, e, com isso, não é preciso interromper o ciclo por muito tempo. No caso de uma produção muito pequena, a moldagem com inserto pode ser efetuada de duas maneiras: (i) usar o ciclo semi-automático, no qual a máquina executa um ciclo e só reinicia o próximo após a abertura e o fechamento da porta de segurança; (ii) usar o modo automático e abrir a porta da máquina depois de cada extração, pois a abertura da máquina faz com que o sistema de segurança da máquina interrompa o ciclo de moldagem até que ela seja fechada novamente.

### 3.7 Problemas que podem ser causados no produto devido a variações dos parâmetros controlados

A injeção de plásticos é influenciada por muitas variáveis, tais como temperatura, pressão e tempo, eficiência da injetora, funcionamento dos controles e instrumentação da injetora, o tipo de polímero processado, o grau de atenção do operador e as suas habilidades, o projeto da peça e a sua geometria e o projeto e construção do molde (Maraghi, 1997). Uma outra variável corresponde às exigências do usuário final (Maraghi, 1997). Serão apresentadas, a seguir, as principais influências que justificam a necessidade de controle das variáveis deste trabalho.

A temperatura do material fundido tem grande influência na qualidade e produtividade. Alguns dos efeitos devido a altas temperaturas são a degradação térmica, as rebarbas e as variações na contração. A degradação térmica envolve a quebra das cadeias moleculares dos polímeros, afetando as propriedades do produto final, as rebarbas na linha de separação do molde devem-se à redução da viscosidade do material fundido e as variações na contração devem-se à dificuldade que há em controlar a taxa de resfriamento da peça no molde se uma temperatura excessiva do material for usada para preencher o molde, podendo até mesmo gerar peças com dimensões diferentes das especificadas. No que se refere a baixas temperaturas, os principais efeitos são peças incompletas, formação de linhas de fluxo e acabamento superficial ruim. As peças incompletas e as linhas de fluxo são devido à solidificação prematura da frente de material fundido, o que pode fazer com que o avanço da frente de material fundido adquira um padrão de linhas de fluxo, similar a um disco de vinil e o acabamento superficial pobre deve-se ao fato de que, se o conteúdo de calor do material fundido for baixo, a massa de material começa a se solidificar antes que possa reproduzir o acabamento superficial das cavidades.

Altas e baixas pressões têm também influência na qualidade e produtividade, devendo a pressão ser devidamente controlada em cada fase do processo de injeção, conforme justificativa a seguir.

1. A pressão de injeção por transferência é a força que expelle o material para fora

do canhão e o injeta através do bico de injeção para o interior do molde, criando uma pressão nas cavidades, tão logo elas estejam preenchidas com o polímero. Se esta pressão for alta demais, a peça será compactada em excesso, podendo também ocorrer formação de rebarbas na linha de separação do molde e um alto nível de tensões residuais pode permanecer na peça moldada, devido à pressão excessiva. Por outro lado, baixas pressões podem causar o enchimento incompleto do molde e assim o acabamento superficial da peça não irá refletir adequadamente a superfície do molde. Além disso, uma baixa pressão não permite a fusão completa de duas frentes de material fundido que venham a se encontrar.

2. A pressão de recalque é menor que a de transferência, sendo aplicada até que o ponto de injeção se solidifique, a fim de evitar que o material retorne para fora da cavidade e ajuda a compensar a contração da peça durante o resfriamento, introduzindo uma quantidade extra de material proveniente da região final do cilindro da injetora para as peças contraídas, o que mantém as dimensões e peso das peças conforme planejado. Se a pressão de recalque é alta demais, podem resultar peças altamente tensionadas e ocorrer variações na contração. Se for baixa demais ocorrerá retro-descarga do material para os canais de alimentação, assim como vazios na peça moldada. A aplicação da pressão de recalque deve ocorrer durante o resfriamento do material fundido, enquanto os pontos de injeção estiverem, ainda, derretidos, porém não solidificados. Se for removida antes que os pontos de injeção se solidifiquem ocorrerá os mesmos problemas que se não for aplicada, o que fará com que as tolerâncias dimensionais não sejam, necessariamente, atingidas, e se é removida depois é um desperdício de energia e tempo no ciclo.
3. A pressão de fechamento é a pressão exercida na placa móvel para manter o molde fechado durante a aplicação da pressão de injeção. A força de fechamento deve ser maior do que a força causada pela pressão de injeção, caso contrário o molde se abrirá. Altas pressões de fechamento podem ser um risco

para as colunas da máquina e podem achatam e bloquear os canais para remoção de ar do molde. Baixas pressões de fechamento podem causar rebarbas na peça na linha de separação do molde e compactação excessiva do material do molde.

4. A pressão de extração é a pressão na peça que está sendo extraída devido ao grau variável de força usado quando os extratores são empurrados. Se a força de extração é grande demais, os pinos extratores deixarão uma marca indesejável nas peças e, podem, em alguns casos perfurá-las, originando buracos.

Há, ainda, influência de altas e baixas velocidades na qualidade e na produtividade. Uma alta velocidade de injeção pode criar uma taxa de cisalhamento excessivo, causando degradação térmica, viscosidade muito baixa no material e fratura do material no ponto de injeção. Uma baixa velocidade provoca a solidificação do material fundido antes que as cavidades estejam completamente cheias, formando peças incompletas. Pode causar linhas de solda deficientes, linhas de fluxo e acabamento superficial ruim, correndo o risco do ponto de injeção se solidificar antes da compactação da peça e da aplicação da pressão de recalque.

## Capítulo 4

# Modelagem do sistema de injeção por redes de Petri

Um modelo é uma abstração da realidade e a simulação é a criação de ambientes virtuais nos quais é possível gerar modelos cuja relação causa-efeito é imaginada e comparada com a realidade. O modelo é, portanto, uma abstração formal, que visa funcionar como a réplica computacional da estrutura, do comportamento ou das propriedades de um fenômeno real ou imaginário. A simulação, por sua vez, consiste basicamente numa experimentação simbólica do modelo (Machado, 1993).

A modelagem do sistema de injeção da máquina injetora IHP-180/350 por redes de Petri, requer dois níveis hierárquicos: (i) superior, representando aqueles sistemas que podem ser acionados a qualquer tempo (dispositivos de segurança e de comando descritos nas seções 3.4 e 3.6 respectivamente), e (ii) inferior, representando as funções da máquina (fechamento, encoste, injeção, dosagem e descompressão, retorno, abertura e extração). Neste trabalho será considerado apenas o nível hierárquico inferior.

Este capítulo está estruturado da seguinte forma: na seção 4.1 será feita a modelagem por redes de Petri para cada função do ciclo de injeção e na seção 4.2 será apresentada a modelagem completa por redes de Petri para o nível hierárquico inferior. Como aplicação do modelo desenvolvido serão apresentados na seção 4.3 os resultados da simulação visando controlar a velocidade de deslocamento da placa móvel em relação à placa fixa, na etapa de fechamento do ciclo de injeção da máquina injetora de plásticos IHP-180/350. Nessa simulação será utilizado um CLP

para realizar as funções de controle e um computador, rodando um programa Simulink, para simular o avanço da unidade injetora, fornecendo, em cada instante de amostragem, a posição atual da placa móvel.

## 4.1 Modelagem por rede de Petri para o nível inferior

A técnica de modelagem adotada neste trabalho é a rede de Petri. A razão principal para a adoção do modelo por redes de Petri é que ela permite, de forma direta, a modelagem de eventos concorrentes. Nos modelos por redes de Petri, (seção 2.3), os lugares representam as condições para a ocorrência dos eventos e as transições representam os eventos. As fichas (*tokens*) são utilizadas para simular a dinâmica e as atividades concorrentes do sistema. A presença de uma ficha em um lugar é interpretada como tornando verdadeira a condição atribuída ao lugar. Quando todas as condições para que um evento ocorra são satisfeitas, a transição é disparada, isto é, o evento ocorre.

De acordo com as definições 2.8 e 2.10, uma rede de Petri fica completamente caracterizada a partir da óctupla  $(P, T, A, w, E, l, x, V)$ , onde  $P$  denota o conjunto dos lugares,  $T$  o conjunto das transições,  $A$  o conjunto dos arcos,  $w$  a função de ponderação,  $E$  o conjunto de eventos,  $l$  a função que associa transições a eventos,  $x$  representa os estados da rede (vetor cujas componentes são o número de fichas em cada lugar da rede) e  $V$  é a estrutura de temporização. Neste trabalho, para cada etapa do ciclo de injeção será desenvolvida uma rede de Petri. A definição dos arcos ( $A$ ) e dos correspondentes pesos dos mesmos ( $w$ ) e a estrutura de temporização serão indicados no diagrama de rede de Petri. Os conjuntos dos lugares ( $P$ ) e das transições ( $T$ ) serão definidos explicitamente utilizando-se tabelas enquanto o conjunto de eventos ( $E$ ) e a função de associação de transições a eventos ( $l$ ) serão indicados de forma apenas descritiva nas tabelas correspondentes às definições das transições.

**Observação 4.1** *Nos diagramas das redes de Petri para as funções do ciclo de*

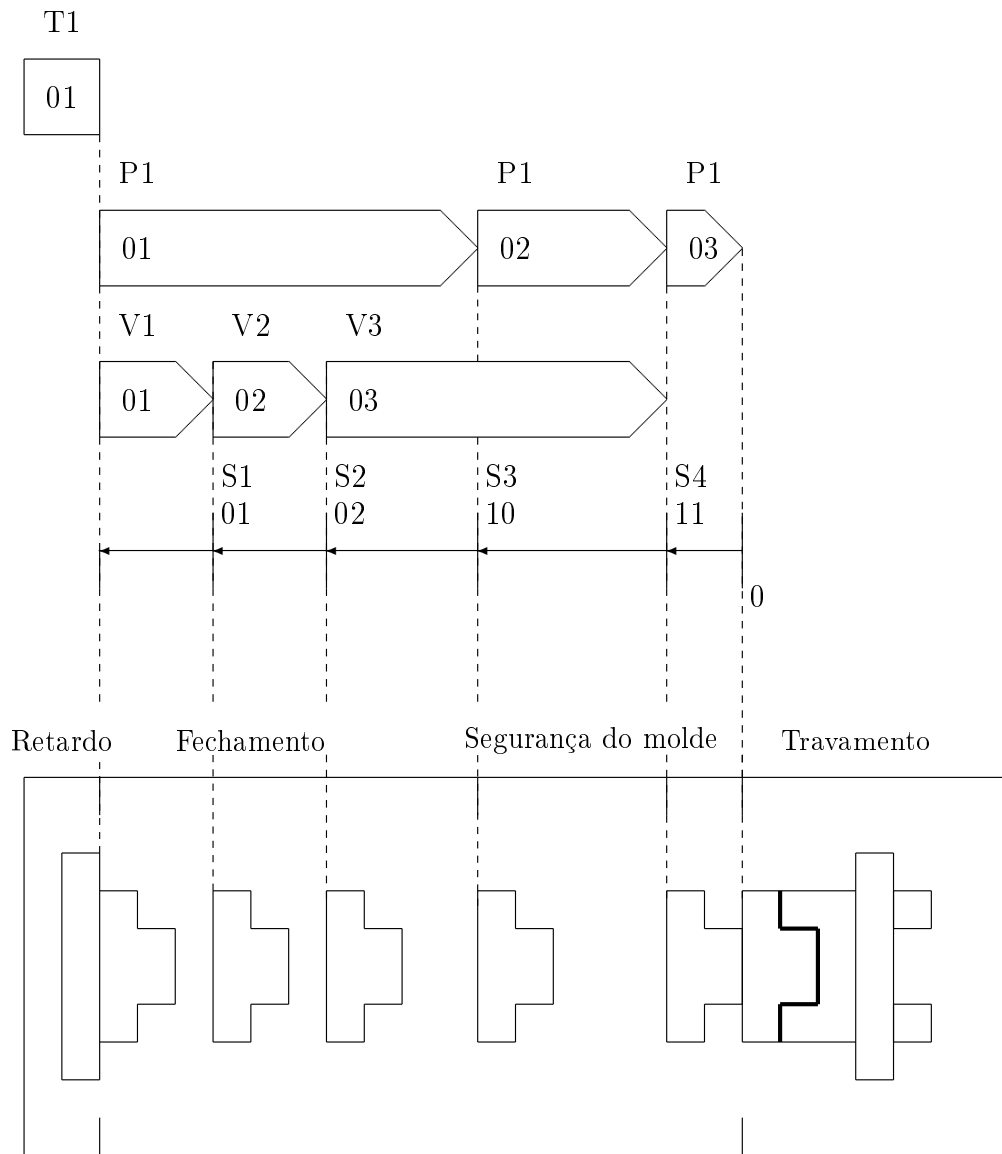


Figura 4.1: Diagrama esquemático para a função fechamento.

*injeção a seguir, não será levada em conta a atuação dos solenóides, uma vez que, em algumas funções não há mudança no acionamento de determinados solenóides (ver tabela 3.8). Isso implicaria na existência de lugares isolados, o que não é usual em redes de Petri (Murata, 1989). Os lugares associados aos solenóides serão, contudo, considerados na modelagem completa do ciclo de injeção a ser apresentada na seção 4.2.*



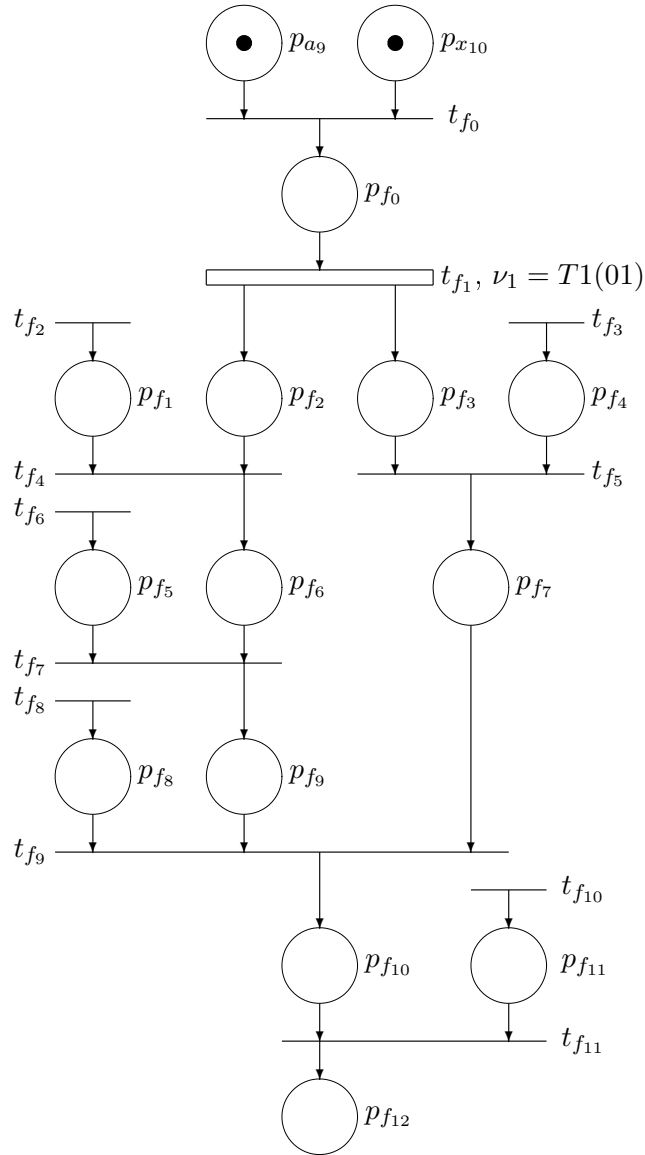


Figura 4.2: Diagrama da rede de Petri para a função fechamento.

### 4.1.1 Função fechamento

A partir da figura 4.1 e do exposto na seção 3.3, pode-se chegar à rede de Petri  $(P_f, T_f, A_f, w_f, E_f, l_f, x_f, V_f)$  da função fechamento, cujo conjunto dos lugares  $(P_f)$  está descrito na tabela 4.1 e o conjunto das transições juntamente com as associações aos eventos a que correspondem (conjunto  $E$  e função  $l$ ) estão definidos na tabela 4.2. Os outros elementos da rede estão mostrados no diagrama da figura 4.2.

No diagrama da rede de Petri da figura 4.2, as condições para início do fechamento são: extração concluída ( $p_{x_{10}}$ ) e placas abertas ( $p_{a_9}$ ). Quando todas essas

Tabela 4.1: Definição do conjunto dos lugares para a rede de Petri ( $P_f, T_f, A_f, w_f, E_f, l_f, x_f, V_f$ ) da função fechamento.

Elemento de $P_f$	Descrição
$p_{f_0}$	Fechamento iniciado
$p_{f_1}$	Posição S1(01) alcançada
$p_{f_2}$	Velocidade de deslocamento igual a V1(01)
$p_{f_3}$	Pressão de fechamento igual a P1(01)
$p_{f_4}$	Posição S3(10) alcançada
$p_{f_5}$	Posição S2(02) alcançada
$p_{f_6}$	Velocidade de deslocamento igual a V2(02)
$p_{f_7}$	Pressão de fechamento igual a P1(02)
$p_{f_8}$	Posição S4(11) alcançada
$p_{f_9}$	Velocidade de deslocamento igual a V3(03)
$p_{f_{10}}$	Velocidade de deslocamento igual a V. P. F. (velocidade pré-fixada)
$p_{f_{11}}$	Régua resistiva na posição 0
$p_{f_{12}}$	Placas travadas (fechamento concluído)
$p_{a_9}$	Placas abertas (abertura concluída)
$p_{x_{10}}$	Extração concluída

Tabela 4.2: Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri ( $P_f, T_f, A_f, w_f, E_f, l_f, x_f, V_f$ ) da função fechamento.

Elemento de $T_f$	Evento associado
$t_{f_0}$	Início do fechamento
$t_{f_1}$	retardo após o início do fechamento
$t_{f_2}$	Sistema de fechamento alcança a posição S1(01)
$t_{f_3}$	Sistema de fechamento alcança a posição S3(10)
$t_{f_4}$	Mudar velocidade de V1(01) para V2(02)
$t_{f_5}$	Mudar pressão de P1(01) para P1(02)
$t_{f_6}$	Sistema de fechamento alcança a posição S2(02)
$t_{f_7}$	Mudar velocidade de V2(02) para V3(03)
$t_{f_8}$	Sistema de fechamento alcança a posição S4(11)
$t_{f_9}$	Mudar velocidade e pressão
$t_{f_{10}}$	Posição final de fechamento alcançada
$t_{f_{11}}$	Acionar travamento

condições forem satisfeitas (uma ficha em cada lugar), a transição  $t_{f_0}$  será habilitada e o lugar  $p_{f_0}$  recebe uma ficha. Assim, a transição  $t_{f_1}$  se tornará ativa (habilitada), disparando após um tempo de retardo igual a T1(01). Após o disparo da transição  $t_{f_1}$ , os lugares  $p_{a_9}$  e  $p_{x_{10}}$  perderão suas fichas e os lugares  $p_{f_2}$  e  $p_{f_3}$  recebem fichas. Quando a placa móvel chega à posição S1(01), indicada pela régua resistiva de posição, há o disparo da transição  $t_{f_2}$ , que é uma transição fonte (Murata, 1989), fornecendo uma ficha ao lugar  $p_{f_1}$ . Como o lugar  $p_{f_2}$  também tem uma ficha, então a transição  $t_{f_4}$  dispara, fazendo a velocidade mudar de V1(01) para V2(02). Na rede de Petri, os lugares  $p_{f_1}$  e  $p_{f_2}$  perdem suas fichas enquanto o lugar  $p_{f_6}$  recebe uma ficha. Quando a placa móvel chega à posição S2(02), indicada pela régua resistiva de posição, a transição  $t_{f_6}$  dispara e o lugar correspondente a esta condição,  $p_{f_5}$ , recebe uma ficha. Como os lugares  $p_{f_5}$  e  $p_{f_6}$  têm fichas, a transição  $t_{f_7}$  dispara, fazendo com que a velocidade mude de V2(02) para V3(03), os lugares  $p_{f_5}$  e  $p_{f_6}$  percam suas fichas e o lugar  $p_{f_9}$  receba uma ficha. Quando a placa móvel chega à posição S3(10), indicada pela régua resistiva de posição, a transição externa  $t_{f_3}$  dispara e o lugar  $p_{f_4}$  recebe uma ficha. Como os lugares  $p_{f_3}$  e  $p_{f_4}$  têm, cada um, uma ficha, a transição  $t_{f_5}$  dispara, fazendo com que a pressão mude de P1(01) para P1(02), os lugares  $p_{f_3}$  e  $p_{f_4}$  fiquem sem fichas e o lugar  $p_{f_7}$  receba uma ficha. Após a placa móvel chegar à posição S4(11), indicada pela régua resistiva de posição, a transição externa  $t_{f_8}$  dispara, fazendo com que o lugar  $p_{f_8}$  receba uma ficha. Como os lugares  $p_{f_7}$ ,  $p_{f_8}$  e  $p_{f_9}$  têm agora fichas, a transição  $t_{f_9}$  irá disparar, o que fará com que a velocidade mude de V3(03) para VPF e a pressão mude de P1(02) para P1(03). Na rede de Petri, ter-se-á que os lugares  $p_{f_7}$ ,  $p_{f_8}$  e  $p_{f_9}$  perderão suas fichas, e o lugar  $p_{f_{10}}$  receberá uma ficha. Quando a placa móvel chega à posição zero, indicada pela régua resistiva de posição, a transição  $t_{f_{10}}$  dispara, fornecendo uma ficha para o lugar  $p_{f_{11}}$ . Nesse ponto, como os lugares  $p_{f_{10}}$  e  $p_{f_{11}}$  têm fichas, a transição  $t_{f_{11}}$  (acionar travamento) dispara. O lugar  $p_{f_{12}}$  (placas travadas) recebe uma ficha, e os lugares  $p_{f_{10}}$  e  $p_{f_{11}}$  perdem fichas, o que corresponde à conclusão da função fechamento.

**Observação 4.2** *O diagrama da rede de Petri da função fechamento mostrado na figura 4.2 mostra uma importante característica da rede de Petri: a capacidade de*

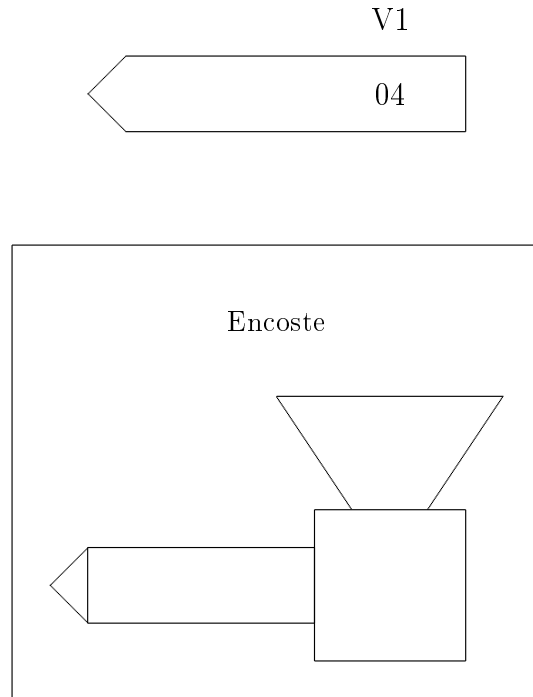


Figura 4.3: Diagrama esquemático para a função encoste.

*modelar a execução de eventos concorrentes. Veja na figura 4.2 que as transições  $t_{f_4}$  e  $t_{f_7}$  disparam de forma independente, porém concorrente, em relação à transição  $t_{f_5}$ . Essa modelagem utilizando autômato implicaria na introdução de diversos estados para modelar todas as possíveis seqüências de ocorrência dos eventos associados às transições  $t_{f_4}$ ,  $t_{f_7}$  e  $t_{f_5}$ .*

#### 4.1.2 Função encoste

A partir da figura 4.3 e do exposto na seção 3.3, pode-se chegar à rede de Petri  $(P_e, T_e, A_e, w_e, E_e, l_e, x_e, V_e)$  da função encoste em que o conjunto dos lugares está descrito na tabela 4.3 e o conjunto das transições, juntamente com as associações aos eventos a que correspondem, estão definidos na tabela 4.4. Os outros elementos da rede estão mostrados no diagrama da figura 4.4.

No diagrama da rede de Petri da figura 4.4 tem-se que, quando as condições para encoste são satisfeitas, o que corresponde a se ter uma ficha nos lugares  $p_{d_{10}}$  (dossagem concluída) e  $p_{f_{12}}$  (fechamento concluído), a transição  $t_{e_1}$  é disparada. Como conseqüência o lugar  $p_{e_1}$  que representa a pressão pré-estabelecida pelo fabricante e a velocidade V1(04) recebe uma ficha e os lugares  $p_{r_5}$ ,  $p_{f_{12}}$  e  $p_{d_{10}}$  perdem suas fichas.

Tabela 4.3: Definição do conjunto dos lugares para a rede de Petri ( $P_e, T_e, A_e, w_e, E_e, l_e, x_e, V_e$ ) da função  $t_{f_1}$  encoste.

Elemento de $P_e$	Descrição
$p_{e_1}$	Pressão de encoste pré-estabelecida pelo fabricante e Velocidade de encoste V1(04)
$p_{e_2}$	Posição final de encoste alcançada
$p_{e_3}$	Encoste concluído
$p_{f_{12}}$	Placas travadas (fechamento concluído)
$p_{d_{10}}$	Dosagem concluída (descompressão concluída)

Tabela 4.4: Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri ( $P_e, T_e, A_e, w_e, E_e, l_e, x_e, V_e$ ) da função encoste.

Elemento de $T_e$	Evento associado
$t_{e_1}$	Início do encoste
$t_{e_2}$	Sistema de encoste atinge posição final (sensor de fim de curso)
$t_{e_3}$	Finalizar encoste

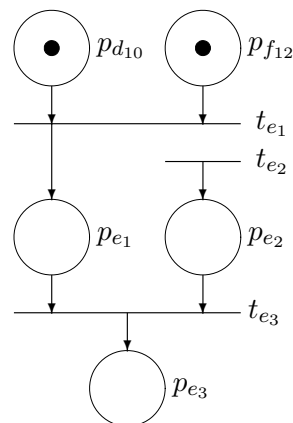


Figura 4.4: Diagrama da rede de Petri para a função encoste.

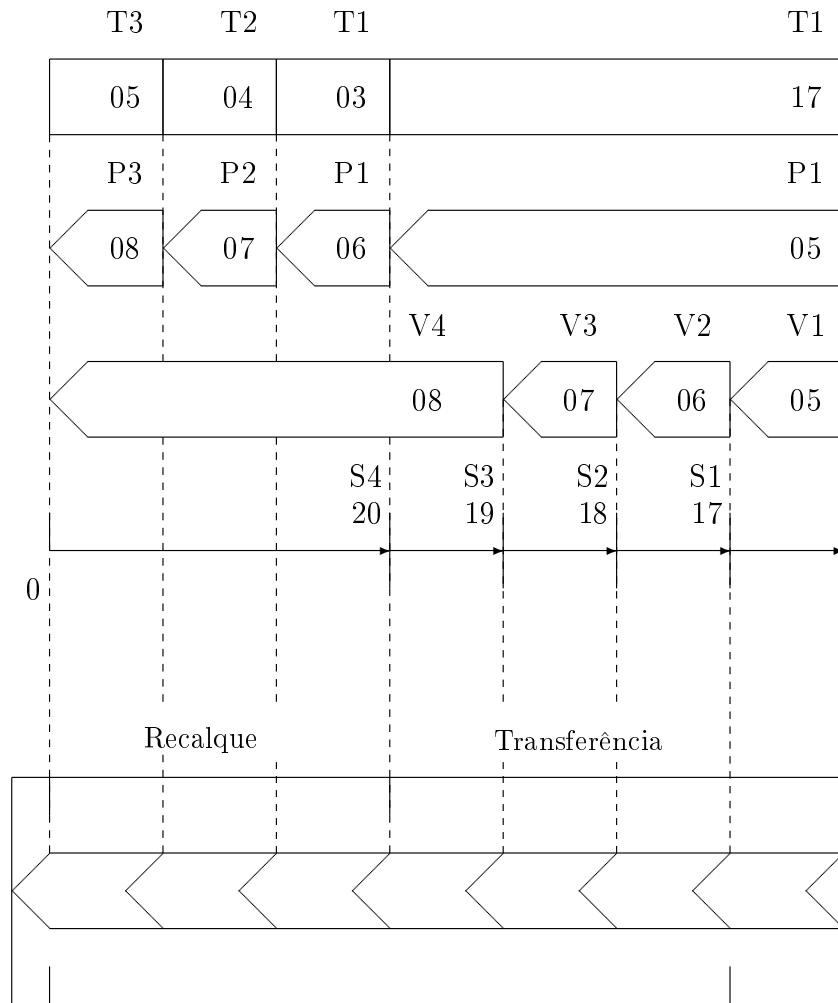


Figura 4.5: Diagrama esquemático para a função injeção.

Quando o conjunto injetor chega à posição de encoste, indicada por um sensor de fim de curso, a transição  $t_{e_2}$  dispara e o lugar  $p_{e_2}$  recebe uma ficha. Como os lugares  $p_{e_1}$  e  $p_{e_2}$  têm fichas, a transição  $t_{e_3}$  (finalizar o encoste) dispara, o lugar  $p_{e_3}$  (encoste concluído) recebe uma ficha e os lugares  $p_{e_1}$  e  $p_{e_2}$  perdem suas fichas, o que corresponde à função encoste ter sido concluída.

### 4.1.3 Função injeção

A partir da figura 4.5 e do exposto na seção 3.3, pode-se chegar à rede de Petri  $(P_i, T_i, A_i, w_i, E_i, l_i, x_i, V_i)$  da função injeção em que o conjunto dos lugares  $(P_i)$  está descrito na tabela 4.5 e o conjunto das transições juntamente com as associações aos eventos a que correspondem estão definidos na tabela 4.6. Os outros elementos da rede estão mostrados no diagrama da figura 4.6.

Tabela 4.5: Definição do conjunto dos lugares para a rede de Petri ( $P_i, T_i, A_i, w_i, E_i, l_i, x_i, V_i$ ) da função injeção.

Elemento de $P_i$	Descrição
$p_{i_1}$	Pressostato checado
$p_{i_2}$	Transferência em execução
$p_{i_3}$	Posição S4(20) alcançada
$p_{i_4}$	Pressão de injeção igual a P1(05)
$p_{i_5}$	Velocidade de deslocamento igual a V1(05)
$p_{i_6}$	Posição S1(17) alcançada
$p_{i_7}$	Pressão de injeção igual a P1(06)
$p_{i_8}$	Velocidade de deslocamento igual a V2(06)
$p_{i_9}$	Posição S2(18)) alcançada
$p_{i_{10}}$	Pressão de injeção igual a P2(07)
$p_{i_{11}}$	Velocidade de deslocamento igual a V3(07)
$p_{i_{12}}$	Posição S3(19) alcançada
$p_{i_{13}}$	Pressão de injeção igual a P3(08)
$p_{i_{14}}$	Velocidade de deslocamento igual a V4(08)
$p_{i_{15}}$	Injeção concluída
$p_{i_{16}}$	Condição para término da transferência
$p_{i_{17}}$	Recalque concluído
$p_{e_3}$	Encoste concluído

Tabela 4.6: Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri ( $P_i, T_i, A_i, w_i, E_i, l_i, x_i, V_i$ ) da função injeção.

Elemento de $T_i$	Evento associado
$t_{i_1}$	Sistema de injeção atinge pressão para injeção
$t_{i_2}$	Início da injeção
$t_{i_3}$	Sistema de injeção atinge posição S4(20)
$t_{i_4}$	Sistema de injeção atinge posição S1(17)
$t_{i_5}$	Finalizar transferência por tempo T1(17)
$t_{i_6}$	Iniciar recalque
$t_{i_7}$	Sistema de injeção atinge posição S2(18)
$t_{i_8}$	Mudar velocidade de V1(05) para V2(06)
$t_{i_9}$	Mudar a pressão de P1(06) para P2(07)
$t_{i_{10}}$	Mudar velocidade de V2(06) para V3(07)
$t_{i_{11}}$	Sistema de injeção atinge posição S3(19)
$t_{i_{12}}$	Mudar a pressão de P2(07) para P3(08)
$t_{i_{13}}$	Mudar velocidade de V3(07) para V4(08)
$t_{i_{14}}$	Finalizar recalque
$t_{i_{15}}$	Iniciar recalque
$t_{i_{16}}$	Não associada a evento (limpar lugares $p_{i_{16}}$ e $p_{i_3}$ )
$t_{i_{17}}$	Finalizar injeção

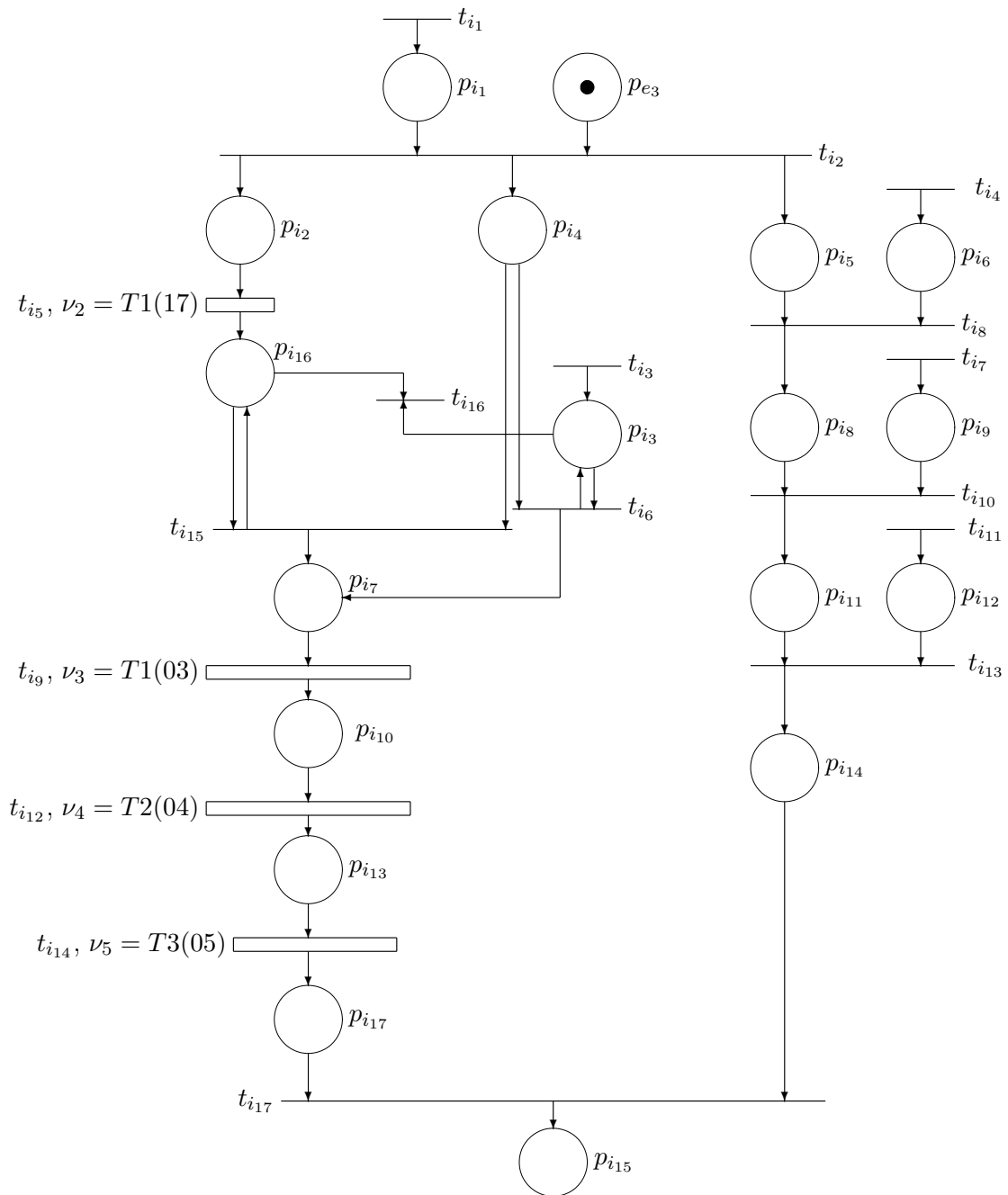


Figura 4.6: Diagrama da rede de Petri para a função injeção.



No diagrama da rede de Petri da figura 4.6 tem-se que, quando o sistema de injeção atinge a pressão para injeção, a transição  $t_{i_1}$  dispara, o lugar  $p_{i_1}$  (pressostato checado) recebe uma ficha. Além disso, uma vez que o encoste tiver sido concluído, ter-se-á também uma ficha no lugar  $p_{e_3}$  (conforme mostrado na seção 4.1.2), o que fará com que a transição  $t_{i_2}$  (início da injeção) seja disparada. Assim, os lugares  $p_{e_3}$  e  $p_{i_1}$  perdem suas fichas, os lugares  $p_{i_2}$ ,  $p_{i_4}$  e  $p_{i_5}$  recebem fichas. Note, de acordo com a figura 4.5, que a evolução da pressão se dá em paralelo (concorrentemente) com a velocidade. Essa evolução será capitada pelo modelo em rede de Petri dado pela figura 4.6.

Considere, inicialmente, a evolução da velocidade. Quando a rosca avança e chega à posição S1(17), indicada pela régua resistiva de posição, a transição  $t_{i_4}$  dispara e o lugar  $p_{i_6}$  recebe uma ficha. Como os lugares  $p_{i_5}$  e  $p_{i_6}$  têm agora fichas, a transição  $t_{i_8}$  (mudar velocidade de V1(05) para V2(06)) é disparada, o que faz com que os lugares  $p_{i_5}$  e  $p_{i_6}$  percam suas fichas e o lugar  $p_{i_8}$  receba uma ficha. Quando a rosca chega à posição S2(18), a transição  $t_{i_7}$  dispara e o lugar  $p_{i_9}$  recebe uma ficha. Isso fará com que a transição  $t_{i_{10}}$  (mudar velocidade de V2(06) para V3(07)) dispare em seguida, fazendo com que os lugares  $p_{i_8}$  e  $p_{i_9}$  percam suas fichas e o lugar  $p_{i_{11}}$  receba uma ficha. Quando a rosca chegar à posição S3(19), indicada pela régua resistiva de posição, a transição  $t_{i_{11}}$  irá disparar e o lugar  $p_{i_{12}}$  irá receber uma ficha. Como os lugares  $p_{i_{11}}$  e  $p_{i_{12}}$  têm fichas, a transição  $t_{i_{13}}$  (mudar velocidade de V3(07) para V4(08)) será disparada, os lugares  $p_{i_{11}}$  e  $p_{i_{12}}$  perderão suas fichas e o lugar  $p_{i_{14}}$  receberá uma ficha.

Considere, agora, a variação da pressão. Na transferência, a única pressão atuante é a P1(05) e há duas possibilidades para o início do recalque, sendo que aquela que ocorrer primeiro o desencadeia: (i) atinge-se a posição S4(20) ou; (ii) esgota-se o intervalo de tempo T1(17) antes de se atingir a posição S4(20). Na primeira, quando a rosca chega à posição S4(20), indicada pela régua resistiva de posição, a transição  $t_{i_3}$  (sistema de injeção atinge posição S4(20)) dispara e o lugar  $p_{i_3}$  recebe uma ficha. Como os lugares  $p_{i_3}$  e  $p_{i_4}$  têm fichas, a transição  $t_{i_6}$  (iniciar recalque) irá disparar e o lugar  $p_{i_7}$  (pressão de injeção igual a P1(06)) receberá

uma ficha. Na segunda possibilidade, como a transição  $t_{i_5}$  (tempo de transferência T1(17)) é habilitada logo após o disparo de  $t_{i_2}$ , tem-se que após decorrido o tempo de transferência T1(17), a ficha de  $p_{i_2}$  se move para o lugar  $p_{i_{16}}$ , o que provoca o disparo da transição  $t_{i_{15}}$ , iniciando-se o recalque. Em qualquer dos dois casos, o lugar  $p_{i_7}$  recebe uma ficha e habilita a transição  $t_{i_9}$  (mudar a pressão de P1(06) para P2(07)). Note a presença da transição  $t_{i_{16}}$ , destinada a eliminar as fichas de  $p_{i_{16}}$  e de  $p_{i_3}$ , quando ambas as condições, tempo de transferência T1(17) decorrido e posição S4(20) alcançada, forem satisfeitas. Após decorrido o tempo T1(03), o lugar  $p_{i_{10}}$  (pressão de injeção igual a P2(07)) recebe uma ficha e o lugar  $p_{i_7}$  perde sua ficha. Como o lugar  $p_{i_{10}}$  tem ficha, a transição  $t_{i_{12}}$  (mudar a pressão de P2(07) para P3(08)) é habilitada. Após decorrido o tempo T2(04), a ficha do lugar  $p_{i_{10}}$  se move para o lugar  $p_{i_{13}}$  (pressão de injeção igual a P3(08)), habilitando a transição  $t_{i_{14}}$ . Decorrido o tempo T3(05), a transição  $t_{i_{14}}$  dispara, movendo a ficha do lugar  $p_{i_{13}}$  para o lugar  $p_{i_{17}}$ . Isso fará com que a transição  $t_{i_{17}}$  dispare, movendo a ficha dos lugares  $p_{i_{17}}$  e  $p_{i_{14}}$  para o lugar  $p_{i_{15}}$ , concluindo, então, a injeção.

**Observação 4.3** *Na etapa de injeção podem ser observados dois pontos importantes: i) duas transições ( $t_{i_6}$  e  $t_{i_{15}}$ ) estão associadas a um mesmo evento; ii) a transição  $t_{i_{16}}$  não está associada a nenhum evento, sendo necessária para limpar os lugares  $p_{i_{16}}$  e  $p_{i_3}$  após o disparo de ambas as transições  $t_{i_3}$  e  $t_{i_5}$ . É chamada transição dissipadora (Murata, 1989).*

A seguir, será visto que, finalizada a injeção, mais uma vez será feito uso da capacidade da rede de Petri de modelar eventos concorrentes. Nesse caso os eventos referem-se às funções: (i) funções retorno, dosagem e descompressão (em seqüência) e (ii) abertura e extração em paralelo. O ciclo de injeção iniciar-se-á após a conclusão de ambas as funções descompressão e extração.

#### 4.1.4 Função retorno

A partir das figuras 4.7 e 4.9 e do exposto na seção 3.3, pode-se chegar à rede de Petri  $(P_r, T_r, A_r, w_r, E_r, l_r, x_r, V_r)$  da função retorno em que o conjunto dos lugares

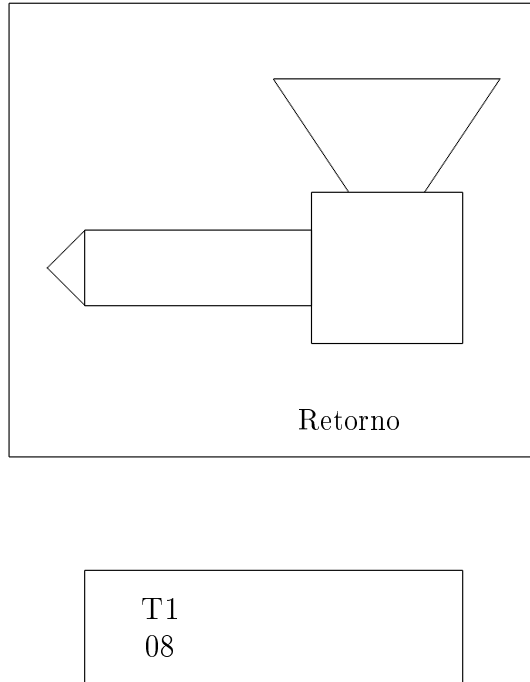


Figura 4.7: Diagrama esquemático para a função retorno.

( $P_r$ ) está descrito na tabela 4.7 e o conjunto das transições ( $T_r$ ) juntamente com as associações aos eventos a que correspondem (conjunto  $E_r$  e função  $l_r$ ) estão definidos na tabela 4.8. Os outros elementos da rede estão mostrados no diagrama da figura 4.8.

No diagrama da rede de Petri da figura 4.8 tem-se que, quando há uma ficha no lugar  $p_{i_{15}}$  (injeção concluída), a transição  $t_{ra}$  é disparada, movendo uma ficha do lugar  $p_{i_{15}}$  para o lugar  $p_{r_1}$ , iniciando, portanto, a descompressão. Assim a transição  $t_{r_2}$  é habilitada e, após um intervalo T1(09) em que a rosca avança (de acordo com a figura 3.19), a transição  $t_{r_2}$  dispara, o lugar  $p_{r_1}$  perde sua ficha e o lugar  $p_{r_2}$  recebe uma ficha. Isso fará com que a transição  $t_{r_3}$  seja habilitada e, após decorrido o intervalo de tempo de comando da função retorno, o lugar  $p_{r_2}$  perde a sua ficha e o lugar  $p_{r_3}$  recebe uma ficha, o que corresponde à função retorno ter sido concluída.

#### 4.1.5 Funções Dosagem e Descompressão

A partir da figura 4.9 e do exposto na seção 3.3, pode-se chegar à rede de Petri ( $P_d, T_d, A_d, w_d, E_d, l_d, x_d, V_d$ ) das funções dosagem e descompressão em que o conjunto dos lugares está descrito na tabela 4.9 e o conjunto das transições, juntamente

Tabela 4.7: Definição do conjunto dos lugares para a rede de Petri ( $P_r, T_r, A_r, w_r, E_r, l_r, x_r, V_r$ ) da função retorno.

Elemento de $P_r$	Descrição
$p_{r_1}$	Retorno iniciado
$p_{r_2}$	Velocidade de deslocamento igual a V1(04) e Pressão de retorno pré-fixada pelo fabricante
$p_{r_3}$	Canhão (conjunto injetor) retornado (retorno do conjunto injetor concluído)
$p_{i_{15}}$	Injeção concluída

Tabela 4.8: Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri ( $P_r, T_r, A_r, w_r, E_r, l_r, x_r, V_r$ ) da função retorno.

Elemento de $T_r$	Evento associado
$t_{ra}$	Iniciar retorno do canhão e abertura do molde
$t_{r_2}$	Descompressão
$t_{r_3}$	Finalizar retorno

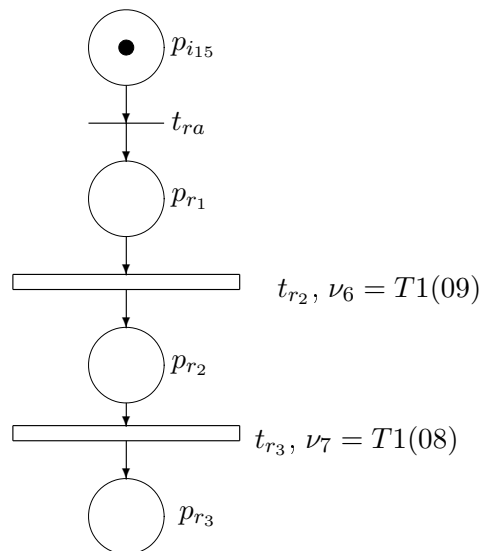


Figura 4.8: Diagrama da rede de Petri para a função retorno.

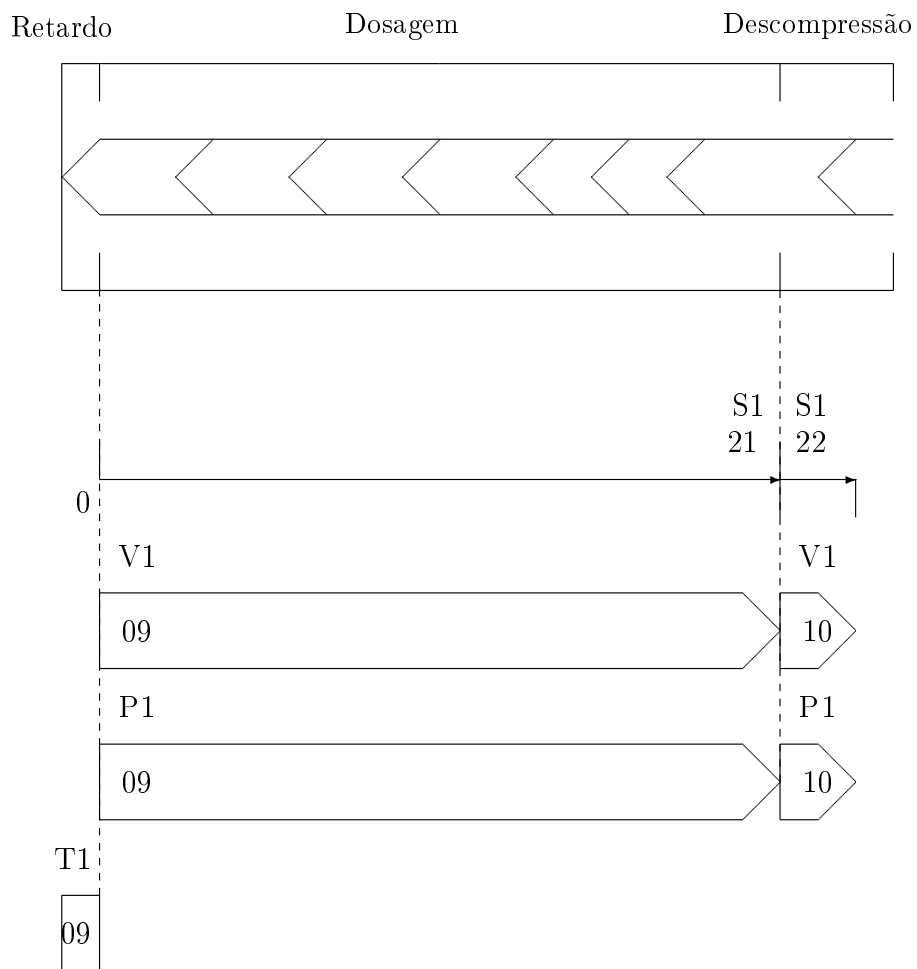


Figura 4.9: Diagrama esquemático para as funções dosagem e descompressão.

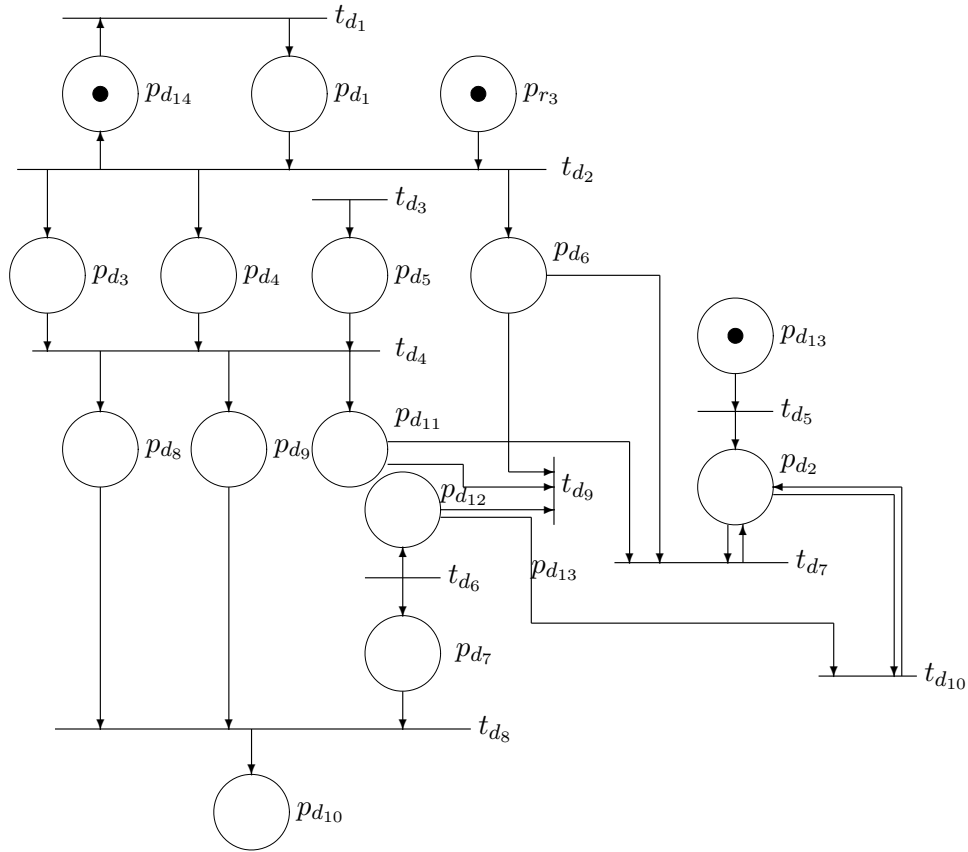


Figura 4.10: Diagrama da rede de Petri para as funções dosagem e a descompressão.

com as associações aos eventos a que correspondem, estão definidos na tabela 4.10. Os outros elementos da rede estão mostrados no diagrama da figura 4.10.

No diagrama da rede de Petri da figura 4.10 tem-se que, supondo que o retorno do canhão tenha sido concluído, ter-se-á uma ficha no lugar  $p_{r_3}$ . Além disso, quando o nível de material na tremonha é maior que o mínimo, a transição  $t_{d_1}$  dispara, colocando uma ficha no lugar  $p_{d_1}$ . Como os lugares  $p_{d_1}$ ,  $p_{d_{15}}$  e  $p_{r_3}$  têm fichas, a transição  $t_{d_2}$  dispara, fazendo com que os lugares  $p_{d_3}$ ,  $p_{d_4}$  e  $p_{d_6}$  que representam, respectivamente, velocidade de deslocamento igual a V1(09), pressão de dosagem igual a P1(09) e rosca girando em sentido anti-horário na dosagem, recebem fichas e os lugares  $p_{d_1}$  e  $p_{r_3}$  perdem uma ficha cada. Note que o lugar  $p_{d_{14}}$  também receberá uma ficha, o que fará com que a transição  $t_{d_1}$  dispare novamente caso o nível de material na tremonha seja maior que o mínimo adotado. Quando a rosca chega à posição S1(21), indicada pela régua resistiva de posição, a transição  $t_{d_3}$  dispara e o lugar  $p_{d_5}$  (posição S1(21) alcançada) recebe uma ficha. Como os lugares  $p_{d_3}$ ,

Tabela 4.9: Definição do conjunto dos lugares para a rede de Petri ( $P_d, T_d, A_d, w_d, E_d, l_d, x_d, V_d$ ) das funções dosagem e descompressão.

Elemento de $P_d$	Descrição
$p_{d_1}$	Tremonha carregada
$p_{d_2}$	F1 selecionada
$p_{d_3}$	Velocidade de deslocamento igual a V1(09)
$p_{d_4}$	Pressão de dosagem igual a P1(09)
$p_{d_5}$	Posição S1(21) alcançada
$p_{d_6}$	Rosca girando em sentido anti-horário
$p_{d_7}$	Posição S1(22) alcançada
$p_{d_8}$	Velocidade de deslocamento igual a V1(10)
$p_{d_9}$	Pressão de descompressão igual a P1(10)
$p_{d_{10}}$	Descompressão concluída
$p_{d_{11}}$	Descompressão iniciada
$p_{d_{12}}$	Termina F1
$p_{d_{13}}$	Condição para seleção da função F1
$p_{d_{14}}$	"Dummy State"
$p_{r_3}$	Canhão retornado

Tabela 4.10: Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri ( $P_d, T_d, A_d, w_d, E_d, l_d, x_d, V_d$ ) das funções dosagem e descompressão.

Elemento de $T_d$	Evento associado
$t_{d_1}$	Tremonha carregada
$t_{d_2}$	Início da dosagem
$t_{d_3}$	Rosca atinge a posição S1(21)
$t_{d_4}$	Iniciar descompressão
$t_{d_5}$	Selecionar a função F1
$t_{d_6}$	Rosca atinge a posição S1(22)
$t_{d_7}$	Interromper rotação da rosca
$t_{d_8}$	Términar descompressão
$t_{d_9}$	Não associada a evento (limpar lugar $p_{d_6}$ )
$t_{d_{10}}$	Não associada a evento (limpar lugar $p_{d_{12}}$ )

$p_{d_4}$  e  $p_{d_5}$  têm agora fichas, a transição  $t_{d_4}$  dispara, dando início à descompressão. Com isso, os lugares  $p_{d_8}$  e  $p_{d_9}$  recebem uma ficha cada e os lugares  $p_{d_3}$ ,  $p_{d_4}$  e  $p_{d_5}$  perdem suas fichas. Se a função F1 foi selecionada, a transição  $t_{d_5}$  (seleção de F1) dispara e o lugar  $p_{d_2}$  (F1 selecionada) recebe uma ficha. Como  $p_{d_6}$ ,  $p_{d_{12}}$  e  $p_{d_2}$  têm fichas, a transição  $t_{d_7}$  (interromper rotação da rosca) dispara, retirando as fichas dos lugares  $p_{d_6}$  e  $p_{d_{11}}$ , o que significa que a rosca não mais girará em sentido anti-horário. Note que o lugar  $p_{d_2}$  mantém sua ficha. Quando a rosca chega à posição S1(22), indicada pela régua resistiva de posição, a transição  $t_{d_6}$  (rosca atinge a posição S1(22)) dispara e os lugares  $p_{d_7}$  e  $p_{d_{12}}$  (posição S1(22) alcançada) recebem fichas. Caso a função F1 tenha sido selecionada, não haverá mais ficha em  $p_{d_6}$  e, então, a transição  $t_{d_{10}}$  dispara, limpando  $p_{d_{12}}$ . Caso a função F1 não tenha sido selecionada, haverá, ainda, uma ficha em  $p_{d_6}$ , o que fará com que  $t_{d_9}$  dispare, limpando  $p_{d_6}$  e  $p_{d_{12}}$ . Em ambos os casos, como os lugares  $p_{d_7}$ ,  $p_{d_8}$  e  $p_{d_9}$  têm fichas, a transição  $t_{d_8}$  (terminar a descompressão) dispara, o lugar  $p_{d_{10}}$  (descompressão concluída) recebe uma ficha e os lugares  $p_{d_7}$ ,  $p_{d_8}$  e  $p_{d_9}$  perdem suas fichas, o que corresponde às funções dosagem e descompressão terem sido concluídas.

**Observação 4.4** *Note que, como a transição  $t_{d_1}$  está associada à leitura do sensor que indica o nível mínimo de material na tremonha, então, enquanto o nível de material na tremonha for maior que o mínimo, a transição  $t_{d_1}$  irá disparar indefinidamente, fazendo com que o número de fichas em  $p_{d_1}$  se tornasse infinito. Para se limitar a  $m$  o número de fichas em  $p_{d_1}$ , introduz-se o lugar  $p_{d_{13}}$  com a configuração mostrada na figura 4.10, para garantir que  $t_{d_1}$  só dispare uma vez antes do disparo de  $t_{d_2}$  (Murata, 1989).*

#### 4.1.6 Função abertura

A partir da figura 4.11 e do exposto na seção 3.3, pode-se chegar à rede de Petri  $(P_a, T_a, A_a, w_a, E_a, l_a, x_a, V_a)$  da função abertura em que o conjunto dos lugares está descrito na tabela 4.11 e o conjunto das transições juntamente com as associações aos eventos a que correspondem estão definidos na tabela 4.12. Os outros elementos da rede estão mostrados no diagrama da figura 4.12.



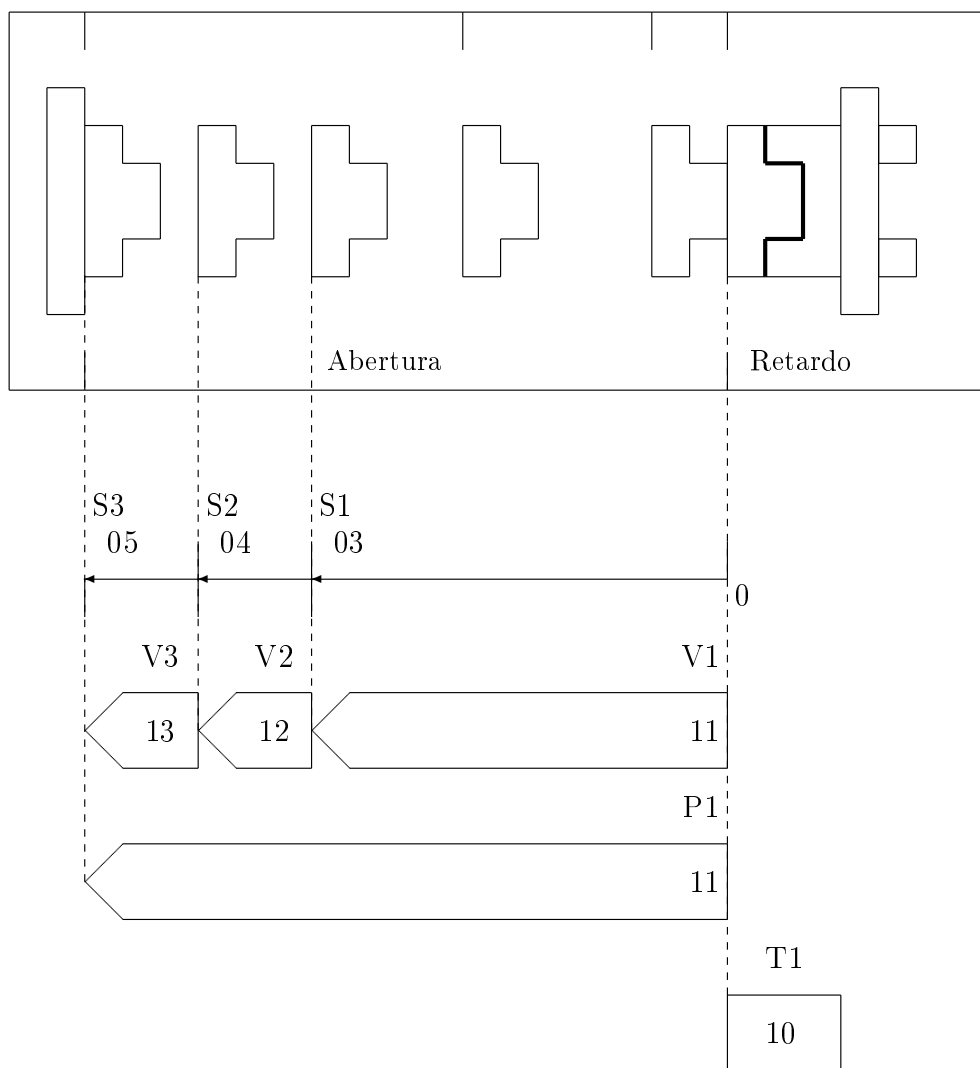


Figura 4.11: Diagrama esquemático para a função abertura.

Tabela 4.11: Definição do conjunto dos lugares para a rede de Petri ( $P_a, T_a, A_a, w_a, E_a, l_a, x_a, V_a$ ) da função abertura.

Elemento de $P_a$	Descrição
$p_{a_1}$	Condição para abertura
$p_{a_2}$	Posição S1(03) alcançada
$p_{a_3}$	Velocidade de deslocamento igual a V1(11)
$p_{a_4}$	Pressão de abertura igual a P1(11)
$p_{a_5}$	Posição S2(04) alcançada
$p_{a_6}$	Velocidade de deslocamento igual a V2(12)
$p_{a_7}$	Posição S3(05) alcançada
$p_{a_8}$	Velocidade de deslocamento igual a V3(13)
$p_{a_9}$	Placas abertas (abertura concluída)

Tabela 4.12: Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri ( $P_a, T_a, A_a, w_a, E_a, l_a, x_a, V_a$ ) da função abertura.

Elemento de $T_a$	Evento associado
$t_{ra}$	Iniciar retorno do canhão e abertura do molde
$t_{a_1}$	Resfriamento
$t_{a_2}$	Sistema de abertura atinge a posição S1(03)
$t_{a_3}$	Mudar velocidade de V1(11) para V2(12)
$t_{a_4}$	Sistema de abertura atinge a posição S2(04)
$t_{a_5}$	Mudar velocidade de V2(12) para V3(13)
$t_{a_6}$	Sistema de abertura atinge a posição S3(05)
$t_{a_7}$	Finalizar abertura

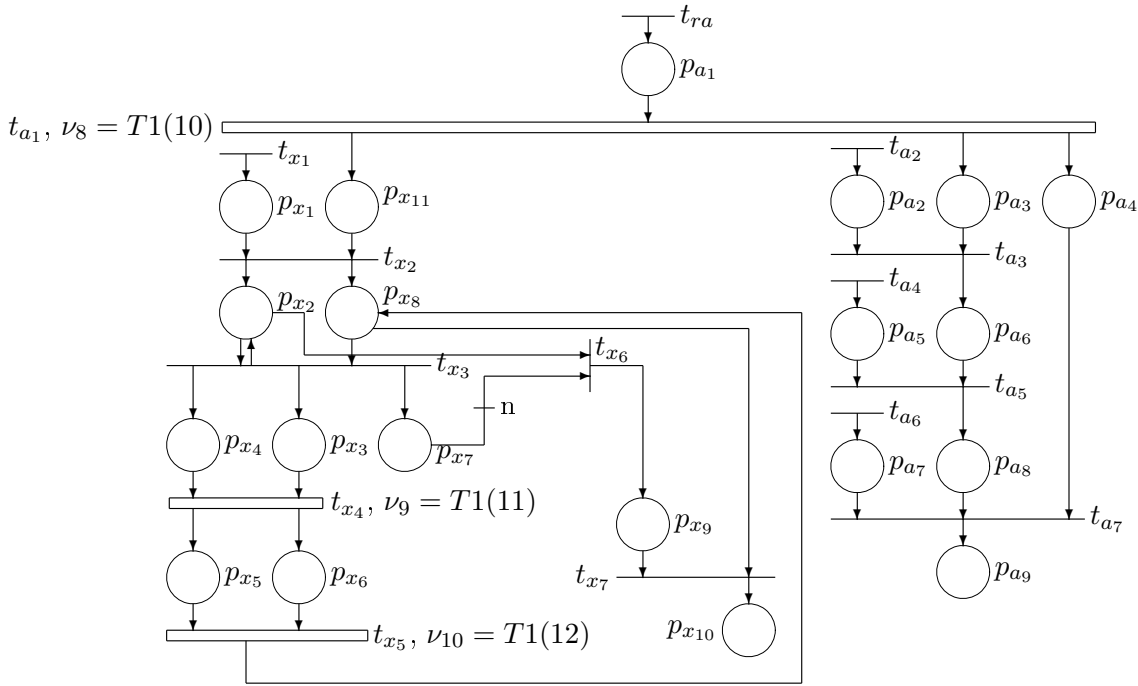


Figura 4.12: Diagrama da rede de Petri para as funções abertura e extração.

No diagrama da rede de Petri da figura 4.12 tem-se que, quando a transição  $t_{ra}$  dispara, o lugar  $p_{a1}$  (condição para abertura) recebe uma ficha. Assim, a transição  $t_{a1}$ , que corresponde ao atraso de início da abertura, determinado pelo tempo de resfriamento da peça, é habilitada. Após decorrido o tempo de retardo  $T1(10)$ , os lugares  $p_{a3}$  e  $p_{a4}$  que representam, respectivamente, velocidade de deslocamento igual a  $V1(11)$  e pressão de abertura igual a  $P1(11)$ , recebem fichas. Note que o lugar  $p_{x11}$ , que representa a condição para iniciar a extração também recebe uma ficha. A evolução dessa ficha será descrita na seção seguinte. Quando a placa móvel chega à posição  $S1(03)$ , indicada pela régua resistiva de posição, a transição  $t_{a2}$  dispara e o lugar  $p_{a2}$  (posição  $S1(03)$  alcançada) recebe uma ficha. Como os lugares  $p_{a2}$  e  $p_{a3}$  têm agora fichas, a transição  $t_{a3}$  (mudar velocidade de  $V1(11)$  para  $V2(12)$ ) dispara, o lugar  $p_{a6}$  (velocidade de deslocamento igual a  $V2(12)$ ) recebe uma ficha e os lugares  $p_{a2}$  e  $p_{a3}$  perdem fichas. Quando a placa móvel chega à posição  $S2(04)$ , indicada pela régua resistiva de posição, a transição  $t_{a4}$  dispara e o lugar  $p_{a5}$  (posição  $S2(04)$  alcançada) recebe uma ficha. Como os lugares  $p_{a5}$  e  $p_{a6}$  têm fichas, a transição  $t_{a5}$  (mudar velocidade de  $V2(12)$  para  $V3(13)$ ) dispara, o lugar

Extrator Hidráulico

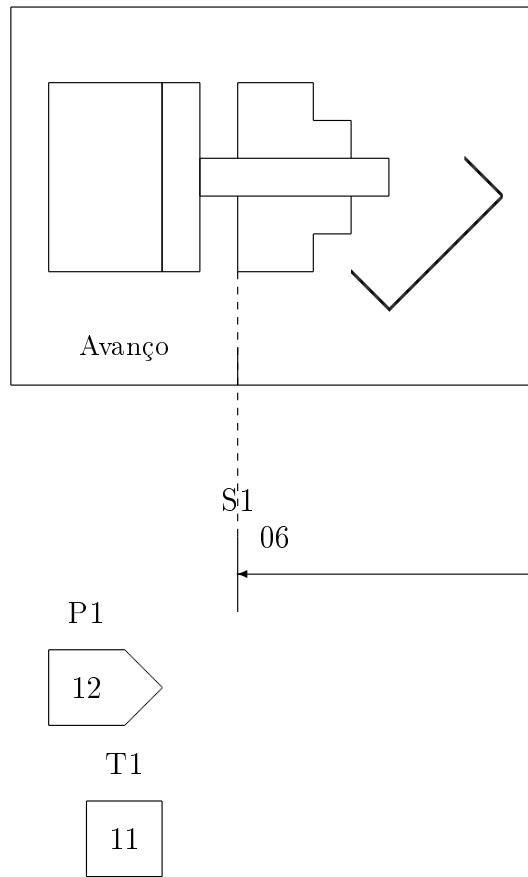


Figura 4.13: Diagrama esquemático para a função extração avanço.

$p_{a8}$  (velocidade de deslocamento igual a  $V3(13)$ ) recebe uma ficha e os lugares  $p_{a5}$  e  $p_{a6}$  perdem fichas. Quando a placa móvel chega à posição  $S3(05)$ , indicada pela régua resistiva de posição, a transição  $t_{a6}$  (sistema de abertura atinge a posição  $S3(05)$ ) dispara e o lugar  $p_{a7}$  (posição  $S3(05)$  alcançada) recebe uma ficha. Como os lugares  $p_{a4}$ ,  $p_{a7}$  e  $p_{a8}$  têm fichas, a transição  $t_{a7}$  (finalizar abertura) dispara, o lugar  $p_{a9}$  (placas abertas ou abertura concluída) recebe uma ficha e os lugares  $p_{a4}$ ,  $p_{a7}$  e  $p_{a8}$  perdem suas fichas, o que corresponde à função abertura ter sido concluída.

#### 4.1.7 Função extração

A partir das figuras 4.13 e 4.14 e do exposto na seção 3.3, pode-se chegar à rede de Petri  $(P_x, T_x, A_x, w_x, E_x, l_x, x_x, V_x)$  da função extração em que o conjunto dos lugares está descrito na tabela 4.13 e o conjunto das transições juntamente com

T1



Extrator Hidráulico

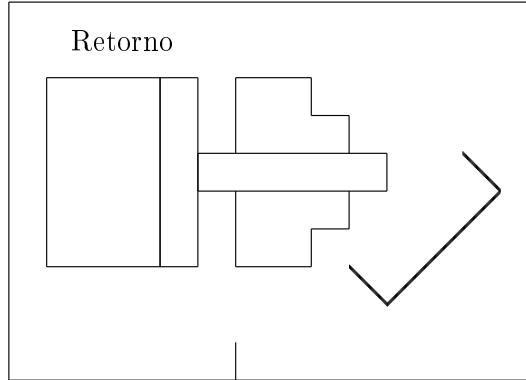


Figura 4.14: Diagrama esquemático para a função extração retorno.

as associações aos eventos a que correspondem estão definidos na tabela 4.14. Os outros elementos da rede estão mostrados no diagrama da figura 4.12.

No diagrama da rede de Petri da figura 4.12 tem-se que, quando  $t_{a_1}$  (resfriamento) dispara, o lugar  $p_{x_{11}}$  recebe uma ficha. Assim, quando a transição  $t_{x_1}$  disparar (sistema de abertura atinge a posição S1(06)), o lugar  $p_{x_1}$  (posição S1(06) alcançada) recebe uma ficha. Assim, as condições para extração são satisfeitas, fazendo a transição  $t_{x_2}$  (acionar extrator) é disparar. Assim, os lugares  $p_{x_2}$  e  $p_{x_8}$ , que representam, respectivamente, condição para continuar a extração e condição para concluir a extração, recebem fichas enquanto os lugares  $p_{x_1}$  e  $p_{x_{11}}$  perdem suas fichas. Como  $p_{x_1}$  e  $p_{x_8}$  têm fichas, e a transição  $t_{x_3}$  (avançar extrator) dispara. Assim, o lugar  $p_{x_8}$  perde sua ficha e os lugares  $p_{x_3}$ ,  $p_{x_4}$  e  $p_{x_7}$ , que representam, respectivamente, pressão de extração igual a P1(12), velocidade de avanço igual a VADF e contador, recebem fichas. Note que  $p_{x_2}$  continua com uma ficha. Como os lugares  $p_{x_3}$  e  $p_{x_4}$  têm fichas, a transição  $t_{x_4}$  (avanço do extrator) é habilitada. Após decorrido o tempo tempo de avanço do extrator (T1(11)), a velocidade passa a ser a de retorno, assim, os lugares  $p_{x_5}$  e  $p_{x_6}$  recebem uma ficha cada e os lugares  $p_{x_3}$  e  $p_{x_4}$  perdem suas ficha. Como os lugares  $p_{x_5}$  e  $p_{x_6}$  têm fichas, a transição  $t_{x_5}$  (retorno do extrator) é habilitada. Após decorrido o tempo de retorno do extrator (T1(12)), o

Tabela 4.13: Definição do conjunto dos lugares para a rede de Petri  $(P_x, T_x, A_x, w_x, E_x, l_x, x_x, V_x)$  da função extração.

Elemento de $P_x$	Descrição
$p_{x_1}$	Posição S1(06) alcançada pela placa móvel
$p_{x_2}$	Condição para concluir a extração
$p_{x_3}$	Pressão de extração igual a P1(12)
$p_{x_4}$	Velocidade de avanço igual a VADF Velocidade de avanço determinada pelo fabricante
$p_{x_5}$	Velocidade de retorno igual a VRDF Velocidade de retorno determinada pelo fabricante
$p_{x_6}$	Pressão de extração igual a P1(12) sentido contrário
$p_{x_7}$	Contador
$p_{x_8}$	Condição para continuar a extração
$p_{x_9}$	Número máximo de repetições alcançado
$p_{x_{10}}$	Extração concluída
$p_{x_{11}}$	Condição para iniciar a extração

Tabela 4.14: Caracterização das transições e das correspondentes associações aos eventos para a rede de Petri  $(P_x, T_x, A_x, w_x, E_x, l_x, x_x, V_x)$  da função extração.

Elemento de $T_x$	Evento associado
$t_{x_1}$	Sistema de abertura atinge a posição S1(06)
$t_{x_2}$	Iniciar extração
$t_{x_3}$	Avançar extrator
$t_{x_4}$	Avanço do extrator
$t_{x_5}$	Retorno do extrator
$t_{x_6}$	Finalizar contagem
$t_{x_7}$	Concluir extração

lugar  $p_{x_8}$  é preenchido novamente por uma ficha. Nessas condições, a função extração recomeça e o processo se repete até que o lugar  $p_{x_7}$  (contador) tenha recebido o número de fichas correspondente à quantidade necessária de repetições, estabelecida para cada produto (definido pelo peso  $n$ , no arco que liga o lugar  $p_{x_7}$  à transição  $t_{x_6}$ ). Neste momento, a transição  $t_{x_6}$  (finalizar contagem) é disparada, os lugares  $p_{x_2}$  e  $p_{x_7}$  perdem suas fichas e o lugar  $p_{x_9}$  (condição para concluir a extração) recebe uma ficha. Quando a última repetição for terminada, os lugares  $p_{x_8}$  e  $p_{x_9}$  terão fichas e a transição  $t_{x_7}$  (concluir extração) é disparada, movendo uma ficha para o lugar  $p_{x_{10}}$  (extração concluída) e os lugares  $p_{x_8}$  e  $p_{x_9}$  perdem suas fichas.

**Observação 4.5** *Os módulos das funções foram validados no HPSim versão 1.1 . O HPSim é um software para simulação de redes de Petri com interface amigável, desenvolvido no contexto de um estudo em C++ no Microsoft Visual Studio e é de domínio público. É executável em plataforma Windows NT, 95-98 e 2000. Permite o acompanhamento da evolução do estado da rede de uma forma gráfica, o que auxilia no desenvolvimento do modelo e na detecção de erros, e a gravação do resultado da simulação e seu posterior tratamento em softwares como o Microsoft Excel, o que é importante para a análise do sistema modelado.*

## 4.2 Modelagem por rede de Petri completa

Para o processo de injeção iniciar-se, o molde deve estar aberto, o canhão recuado e a tremonha abastecida. O primeiro passo é encher a unidade de injeção com resina plástica da tremonha (dosagem). Em seguida, começa o ciclo de injeção, que será modelado nesta seção. Para efetuar a modelagem completa desta seqüência, todos os diagramas de redes de Petri gerados para a máquina injetora da seção 4.1 (figuras 4.2, 4.4, 4.6, 4.8, 4.10, 4.12) são então interligados gerando a rede de Petri da figura 4.15. O conjunto dos lugares adicionais da rede completa está descrito na tabela 4.15 e o conjunto das transições adicionais juntamente com as associações aos eventos a que correspondem estão definidos na tabela 4.16. Na rede de petri completa são também apresentados os lugares relativos aos solenóides que ativam

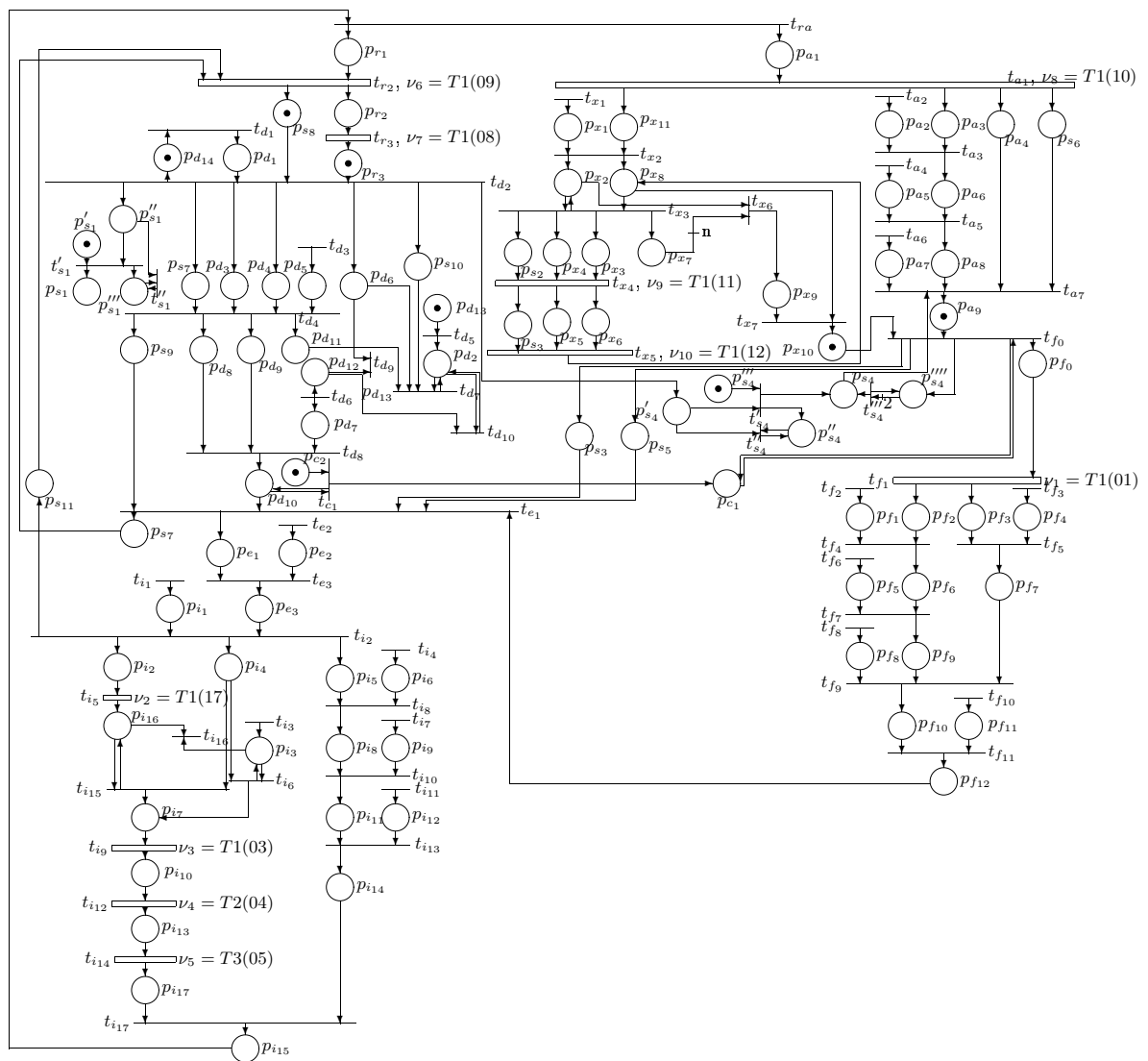


Figura 4.15: Rede de Petri total com solenóides.

as funções, descritos na tabela 4.17. O conjunto dos lugares relativos aos solenóides que ativam essas funções estão descritos na tabela 4.18.

No estado inicial, os seguintes lugares devem conter uma ficha cada um:  $p_{a9}$  (abertura concluída),  $p_{x10}$  (extração concluída) e  $p_{r3}$  (canhão recuado), condições para que o processo de injeção seja iniciado. Além desses lugares, devem haver fichas nos seguintes lugares  $p_{d14}$  ("dummy state"), para que o nível de material na tremonha seja inicialmente maior que o mínimo adotado e possa acionar a condição tremonha carregada;  $p'_{s1}$  (condição para S1 receber uma ficha), que é necessário



Tabela 4.15: Definição do conjunto dos lugares adicionais na rede de Petri  $(P_c, T_c, A_c, w_c, E_c, l_c, x_c, V_c)$  completa.

Elemento de $P_x$	Descrição
$p_{c_1}$	Fechamento acionado na primeira vez
$p_{c_2}$	Condição para ativar o fechamento na primeira vez
$p'_{s_1}$	Lugar auxiliar para que o lugar $p_{s_1}$ receba apenas uma ficha na primeira vez que a transição $t_{d_2}$ disparar.
$p''_{s_1}$	Lugar auxiliar 2 para o solenóide $S1$ ativo.
$p'''_{s_1}$	Lugar auxiliar para que o lugar $p'_{s_1}$ tenha no máximo uma ficha.
$p'_{s_4}$	Condição para acionar SIZ na primeira vez
$p''_{s_4}$	Solenóide SIZ já acionado pelo menos uma vez
$p'''_{s_4}$	Lugar auxiliar para que o lugar $p_{s_4}$ receba apenas uma ficha na primeira vez que a transição $t_{d_2}$ disparar.
$p''''_{s_4}$	Lugar auxiliar para que o lugar $p_{s_4}$ seja acionado no início do fechamento.

Tabela 4.16: Caracterização das transições adicionais e das correspondentes associações aos eventos para a rede de Petri  $(P_c, T_c, A_c, w_c, E_c, l_c, x_c, V_c)$  completa.

Elemento de $T_x$	Evento associado
$t_{c_1}$	Acionar o fechamento na primeira vez
$t'_{s_1}$	Transição auxiliar para que o lugar $p_{s_1}$ receba apenas uma ficha na primeira vez que a transição $t_{d_2}$ disparar.
$t''_{s_1}$	Transição auxiliar para que o lugar $p''_{s_1}$ tenha no máximo uma ficha.
$t'_{s_4}$	Transição auxiliar para acionar SIZ na primeira vez
$t''_{s_4}$	Transição auxiliar para eliminar a ficha de $p'_{s_4}$ nos ciclos seguintes.
$t'''_{s_4}$	Transição auxiliar para que o lugar $p_{s_4}$ seja acionado no início do fechamento.

Tabela 4.17: Funções e solenóides acionados para execução das funções.

Funções	S1	S89	S8A	SIZ	S82	S88	S83	S87	S86	S85	S84
1) Fechamento	X		X	X	X						
2) Encoste	X			X			X				
3) Injeção	X			X			X				X
4) Retorno	X			X				X			
5) Dosagem	X			X			X			X	
6 a) Descompressão F1	X			X					X		
6 b) Descompressão F2	X			X					X	X	
7 a) Abertura	X			X		X					
7 b) Extrator avança	X	X									
7 c) Extrator retorna	X		X								

Tabela 4.18: Definição do conjunto dos lugares relativos aos solenóides que ativam as funções.

Elemento de $P_x$	Descrição
$p_{s_1}$	Solenóide S1 ativo
$p_{s_2}$	Solenóide S89 ativo
$p_{s_3}$	Solenóide S8A ativo
$p_{s_4}$	Solenóide SIZ ativo
$p_{s_5}$	Solenóide S82 ativo
$p_{s_6}$	Solenóide S88 ativo
$p_{s_7}$	Solenóide S83 ativo
$p_{s_8}$	Solenóide S87 ativo
$p_{s_9}$	Solenóide S86 ativo
$p_{s_{10}}$	Solenóide S85 ativo
$p_{s_{11}}$	Solenóide S84 ativo

para que o lugar  $p_{s_1}$  receba apenas uma ficha na primeira vez que a transição  $t_{d_2}$  disparar;  $p_{s_8}$  (solenóide S87 ativo), indicando que o retorno está ativo;  $p_{c_2}$  (condição para acionar o fechamento na primeira vez), já que, na primeira vez, o início do fechamento precisa ser atrelado ao final da dosagem; e  $p_{s_4}''$  (lugar auxiliar para que o lugar  $p_{s_4}$  receba apenas uma ficha na primeira vez que a transição  $t_{d_2}$  disparar).

**Observação 4.6** *O artifício sugerido no artigo Murata (1989) não pode ser usado, uma vez que o lugar  $p_{s_1}$  não é entrada de outra transição. Assim, foi construído o artifício utilizado no diagrama, que consiste em utilizar um lugar inicial  $p'_{s_1}$  e uma transição  $t'_{s_1}$ , para que o lugar  $p_{s_1}$  receba apenas uma ficha na primeira vez que a transição  $t_{f_0}$  disparar, e em utilizar a transição  $t''_{s_1}$  e o lugar  $p'''_{s_1}$ , para que o lugar  $p'_{s_1}$  tenha no máximo uma ficha.*

No diagrama da rede de Petri completa da figura 4.15 tem-se que, para o processo de injeção iniciar-se, na primeira vez o molde deve estar aberto, o canhão recuado e a tremonha abastecida, assim, os lugares  $p_{a_9}$  (abertura concluída),  $p_{x_{10}}$  (extração concluída),  $p_{r_3}$  (canhão recuado) e  $p_{s_8}$  (solenóide S87 ativo) têm fichas inicialmente. Como  $p_{d_{14}}$  ("dummy state"), que indica que o nível de material na tremonha é maior que o mínimo adotado, também tem uma ficha, então a transição  $t_{d_1}$  é disparada e o lugar  $p_{d_1}$  (tremonha carregada) recebe uma ficha, indicando que a tremonha está abastecida.

A seguir deve-se encher a unidade de injeção com resina plástica da tremonha, o que corresponde à função dosagem, já descrita na subseção 4.1.5. Durante esta função devem estar ativos os solenóides S1, SIZ, S83 e S85, como mostra a tabela 4.17. Para que os solenóides S83 e S85 sejam ativados na dosagem é preciso que  $p_{s_7}$  e  $p_{s_{10}}$  recebam fichas, o que ocorre após o disparo da transição  $t_{d_2}$  (início da dosagem). Quanto ao solenóide S1, este deve permanecer ativo durante todo o ciclo e apresenta uma lógica específica para mantê-lo ligado e com no máximo uma ficha. O lugar  $p''_{s_1}$  (estado auxiliar 2 para o solenóide S1 ativo) tem inicialmente uma ficha. Como  $p''_{s_1}$  e  $p'_{s_1}$  têm fichas, a transição  $t'_{s_1}$  é disparada. Os lugares  $p_{s_1}$  e  $p'''_{s_1}$  recebem fichas. Note que, o lugar inicial  $p'_{s_1}$  e a transição  $t'_{s_1}$  são para que o lugar  $p_{s_1}$  receba

apenas uma ficha na primeira vez que a transição  $t_{d_2}$  disparar. Já a transição  $t''_{s_1}$  e o lugar  $p'''_{s_1}$  são para que o lugar  $p''_{s_1}$  tenha no máximo uma ficha. Já o solenóide SIZ deve permanecer ativo por todo o ciclo menos durante a função extração. Para ativá-lo pela primeira vez, foram introduzidos os lugares  $p'_{s_4}$ ,  $p''_{s_4}$  e  $p'''_{s_4}$  e as transições  $t'_{s_4}$  e  $t''_{s_4}$ . Quando  $t_{d_2}$  disparar, o lugar  $p'_{s_4}$  recebe uma ficha. Como o lugar  $p''_{s_4}$  tem uma ficha, a transição  $t'_{s_4}$  dispara, passando uma ficha para o lugar  $p_{s_4}$  (solenóide SIZ acionado) e  $p''_{s_4}$  (que indica que o solenóide SIZ já foi acionado pelo menos uma vez). A transição  $t''_{s_4}$  é para eliminar a ficha de  $p'_{s_4}$  nos ciclos seguintes. Nos ciclos seguintes esse solenóide deve ser acionado no início do fechamento, o que é garantido introduzindo-se o lugar  $p''''_{s_4}$  e a transição  $t''''_{s_4}$ . Note que o peso do arco que liga  $p''''_{s_4}$  e  $t''''_{s_4}$  é igual a 2, o que impede que  $p_{s_4}$  receba uma ficha quando  $t_{f_0}$  disparar pela primeira vez. Durante a descompressão, S83 deve estar inativo, assim  $p_{s_7}$  perde sua ficha com o disparo da transição  $t_{d_4}$ , e S86 deve ser ativado, ou seja,  $p_{s_9}$  recebe uma ficha, e permanece ativo até o disparo da transição  $t_{e_1}$ , início do encoste. Se F1 estiver selecionada S85 deve ser desativada, assim,  $p_{s_{10}}$  perde sua ficha após o disparo da transição  $t_{d_7}$  (interromper rotação da rosca).

Ao término da dosagem pela primeira vez, a função fechamento, que movimenta a placa móvel na direção da fixa, deve ser ativada. Para tanto, como o lugar  $p_{d_{10}}$  (descompressão concluída) tem uma ficha, a transição  $t_{c_1}$  dispara e o lugar  $p_{c_1}$  recebe uma ficha. Como os lugares  $p_{a_9}$  (abertura concluída) e  $p_{x_{10}}$  (extração concluída) já têm fichas, a transição  $t_{f_0}$  é disparada, iniciando-se a função fechamento, já descrita na subseção 4.1.1. Durante esta função os solenóides S1, SIZ, S8A e S82 devem estar ativos. Como S1 já está ativo, quando a transição  $t_{f_0}$  (início do fechamento) é disparada, os lugares  $p_{s_3}$  (solenóide S8A ativo) e  $p_{s_5}$  (solenóide S82 ativo) recebem uma ficha cada.

O passo seguinte é o avanço da unidade de injeção (canhão), ou seja, função encoste, já descrita na subseção 4.1.2, para a qual é preciso que, além de S1 e SIZ já ativos, ou seja, os lugares  $p_{s_1}$  e  $p_{s_4}$  com fichas e S83 esteja ativo, ou seja,  $p_{s_7}$  receba uma ficha, o que ocorre após o disparo da transição  $t_{e_1}$  (início do encoste).

Seguinte ao avanço da unidade de injeção, ocorre a função injeção, propria-

mente dita. Nas etapas de transferência e recalque é preciso, além de S1, SIZ e S83 já ativos, que S84 seja ativado, ou seja, além de  $p_{s_1}$ ,  $p_{s_4}$  e  $p_{s_7}$ , que já possuem uma ficha cada, também  $p_{s_{11}}$  deve receber uma ficha, o que ocorre após o disparo da transição  $t_{i_2}$  (início da injeção) e é mantido ativo por toda a função injeção. O mesmo ocorre com  $p_{s_7}$ .

Após o final do recalque é possível efetuar o recuo do canhão, para o qual é preciso que, além de S1 e SIZ, já ativos, S87 seja ativado. Assim, para ocorrer a função retorno é preciso que, além de  $p_{s_1}$  e  $p_{s_4}$ , já com uma ficha cada,  $p_{s_8}$  receba uma ficha, o que ocorre após o disparo da transição  $t_{r_2}$  (descompressão), sendo  $p_{s_8}$  mantido com uma ficha por toda a função retorno.

De acordo com o ciclo de injeção, a próxima operação é a dosagem de polímero que se dá em conjunto com o movimento de plastificação, já explicada no início por ser a primeira função ativa na primeira vez que o ciclo ocorre.

Após o resfriamento da peça ocorre a abertura do molde, na qual a placa móvel da máquina é acionada por sistema hidráulico, e a extração da peça. Para ocorrerem as funções abertura e extração, quando  $t_{ra}$ , transição que inicia o retorno do canhão e a abertura do molde, é acionada, os lugares  $p_{r_1}$  (retorno iniciado) e  $p_{a_1}$  (condição para abertura) recebem uma ficha cada.

A função abertura faz o caminho oposto da função fechamento, ou seja, afasta a placa móvel da placa fixa. Para ocorrer a função abertura é preciso que, além de S1 e SIZ, já ativos, S88 seja ativado, ou seja, além de  $p_{s_1}$  e  $p_{s_4}$ , terem uma ficha cada,  $p_{s_6}$  deve receber uma ficha, o que ocorre após o disparo da transição  $t_{a_1}$  (resfriamento) e é mantido ativo por toda a função abertura.

A função extração começa após a placa móvel ter percorrido uma determinada distância de abertura, correspondente à posição S1(06). A extração é dividida em extração avanço e extração retorno. Para ocorrer o avanço da extração é preciso que, além de S1, já ativo, S89 seja ativado, ou seja, além de  $p_{s_1}$ , já com ficha,  $p_{s_2}$  receba uma ficha, no avanço do extrator, o que ocorre após o disparo da transição  $t_{x_3}$  (avançar extrator) e seja mantido ativo durante o avanço do extrator. Já para ocorrer o retorno da extração é preciso que, além de S1, já ativo, S89 seja ativado,

ou seja, além de  $p_{s_1}$ , já com ficha, que  $p_{s_2}$  perca sua ficha e  $p_{s_3}$  receba uma ficha, o que ocorre após o disparo da transição  $t_{x_4}$  e é mantido ativo durante o retorno do extrator. Note que SIZ não deve estar ativo durante esta função.

**Observação 4.7** *O modelo da rede de Petri tem como objetivo flexibilizar, facilitar e permitir que sejam feitas virtualmente futuras manutenções, não afetando, assim, o sistema de produção. É fundamental que a especificação baseada em redes de Petri possa ser implementada. Uma das formas de implementação é sua utilização para a geração automática do código em programação ladder, através de aplicativos como o Grafcet (David, 1995). Outra maneira é a conjugação do modelo de redes de Petri com um simulador híbrido, como o Visual Object Net ++, por exemplo, que apresenta suporte a elementos discretos e contínuos (Souza et al., 2005).*

## 4.3 Simulação

Nesta seção será apresentada uma simulação do funcionamento da função fechamento da máquina injetora IHP-180/350 utilizando o CLP SLC500 da Allen-Bradley e um computador digital que fará as vezes da injetora e a realização da evolução dinâmica do movimento da placa móvel em direção à placa fixa utilizando o SIMULINK do Matlab. Antes de se considerar a simulação propriamente dita será feita nas seções 4.3.1 e 4.3.2 uma breve descrição do “hardware” e “software” utilizados na simulação.

### 4.3.1 O SIMULINK e o Real-time Windows Target

O SIMULINK é um acessório do MATLAB que acrescenta uma interface de diagramas de blocos e recursos de simulação gráficos e de linguagem de programação que fazem parte do MATLAB. É um pacote que permite modelagem, simulação e análise de sistemas dinâmicos.

O Real-time Windows Target é a biblioteca de desenvolvimento em tempo real do Simulink, onde o sistema alvo e o ambiente de desenvolvimento estão no mesmo computador. Os requisitos mínimos do sistema são: computador PC com Windows

NT4.0, Millenium Edition ou XP; Matlab 6.5; Simulink; real-time workshop, que converte blocos do Simulink e modelos de sistemas de eventos discretos do State-Flow em código C, a fim de ser utilizado pelo compilador na implementação de simulações em tempo real; compilador C, que converte o código C proveniente do real-time workshop em código executável; target, que é o componente do computador associado à aplicação desenvolvida. Um passo-a-passo de instalação e utilização do Real-time Windows Target está descrito em Calvo (2006).

### **4.3.2 O CLP SLC500 e o software RSLogix500**

O SLC500 é uma família de controladores para aplicações em pequenos e médios processos industriais. O controlador lógico programável SLC5/02 fabricado pela Allen Bradley, utilizado neste trabalho, apresenta uma CPU 1746-P2, trabalha com tensões de 85 a 132 VAC a 50/60 Hz e sua saída varia de -5 a 5 VDC.

O software de programação desenvolvido pela Rockwell Automation para ser usado com o SLC500 é o RS Logix 500. Ele permite programar, monitorar, resolver problemas e documentar sistemas controladores programáveis. Os requisitos mínimos de sistema para sua utilização são Windows NT/2000/XP, Pentium II 500 MHz, 128 MB de memória de trabalho, 45 MB de memória em disco rígido, unidade de disquete de 3,5". Sua tela principal é apresentada na figura 4.16.

O RSLogix500 permite as seguintes instruções:

- Instruções do tipo relé para verificação se um relé está energizado ou desenergizado.
- Instruções de temporizador para contagem dos intervalos da base de tempo e contador para contar as transições de falsa para verdadeira, as quais podem ser causadas por eventos que ocorrem no programa.
- Instruções de Comparação para verificação de igualdade e desigualdade e comparações do tipo “menor que”, “maior que”, “menor que ou igual a” e “maior que ou igual a” entre parâmetros.

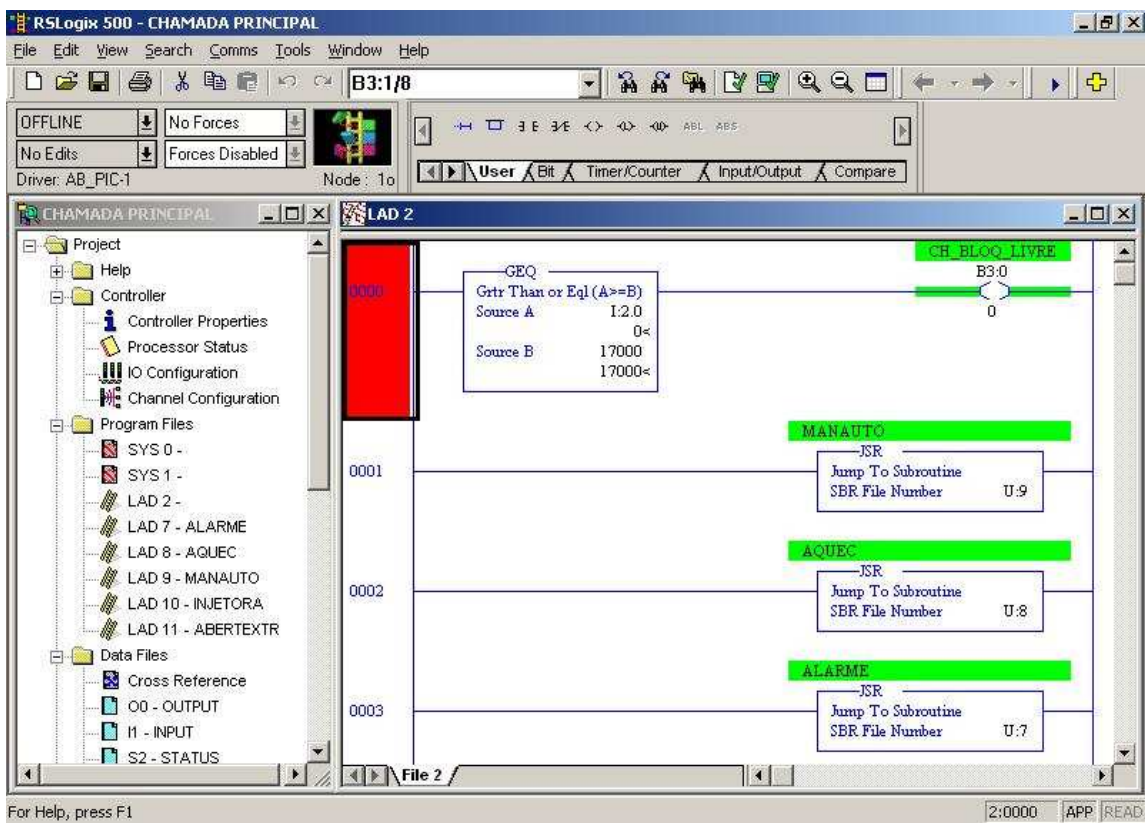


Figura 4.16: Janela principal do RS Logix 500.



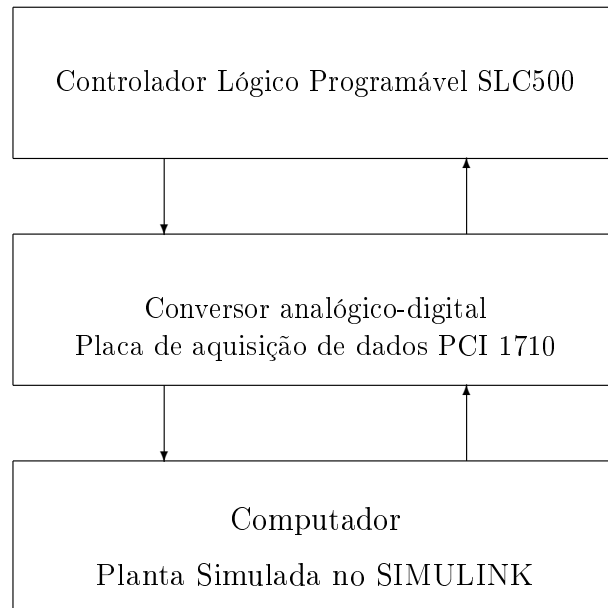


Figura 4.17: Esquema do sistema de simulação.

- Instruções Matemáticas para efetuar “adição”, “subtração”, “multiplicação”, “divisão”, “negação” entre parâmetros e para “zerar um parâmetro”.
- Instruções de Movimentação com o objetivo de mover um valor de uma fonte para um destino.
- Instruções Lógicas para efetuar operações lógicas como “e”, “ou”, “ou exclusivo” e “complemento”.

### 4.3.3 Simulação da variação de velocidade na função fechamento

A figura 4.17 mostra um diagrama de blocos do sistema a ser adotado na simulação. A entrada no Simulink será a velocidade, determinada a partir de um programa em ladder no CLP, e a saída do SIMULINK será a posição, que é informação necessária para a variação da velocidade e da pressão. A figura 4.18 apresenta todos os componentes do sistema de simulação da velocidade na função fechamento da máquina injetora.

Será utilizada uma modelagem simplificada, baseada em movimento retilíneo uniforme, modelo cinemático, representado por  $\Delta v = \Delta x / \Delta t$  e  $x = x_0 + v_0 t$ , uma

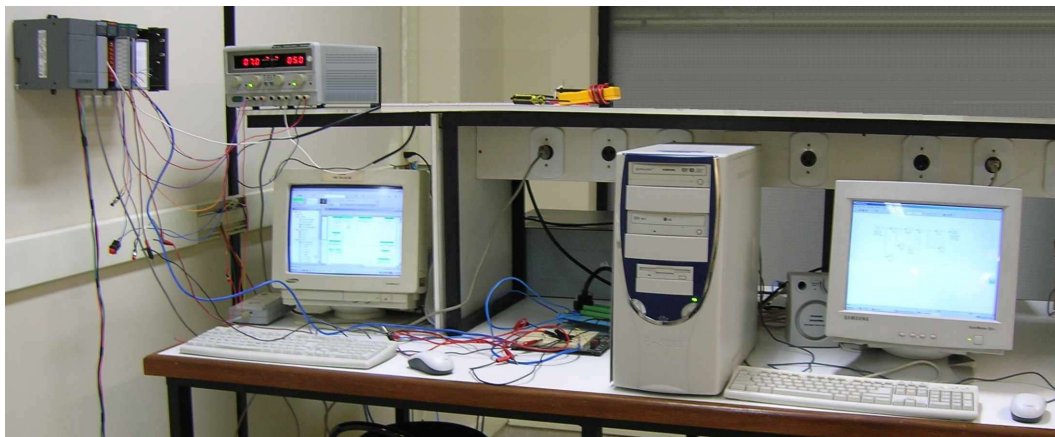


Figura 4.18: Componentes do sistema de simulação.

vez que para gerar uma modelagem mecânica mais sofisticada da máquina seria necessário medir parâmetros como atritos, inércias, massas, entre outros, que não estão disponíveis, já que a máquina não está em funcionamento. Além disso, o objetivo principal aqui é mostrar a aplicação do modelo por redes de Petri do sistema no desenvolvimento de um sistema de automação.

A figura 4.19 apresenta os elementos que compõem a planta simulada no SIMULINK. Seus componentes são: (i) bloco de entrada analógica, (ii) bloco de saída analógica (blocos da biblioteca Real-time Windows Target, que permitem que se selecione e conecte canais analógicos específicos ao modelo projetado), (iii) um bloco integrador, (iv) bloco da biblioteca “continuous”, que alimentará o programa ladder.

Para conectar o computador que comporta o programa de simulação com o computador que comporta o programa em linguagem ladder (que fará o controle da planta simulada) foram utilizados um conversor analógico digital, a placa AD/DA PCI1710, apresentada na figura 4.20, com referência de tensão interna para saída analógica de 5V e cujas especificações podem ser encontradas no manual do usuário PCI1710, (Advantech, 2004), que vem com a placa, e um protoboard, apresentado na figura 4.21, com três resistores de  $10\Omega$  em série, somando  $30\Omega$ , entre o controlador e a placa AD/DA. Isso se faz necessário, uma vez que a saída no SLC500 é medida

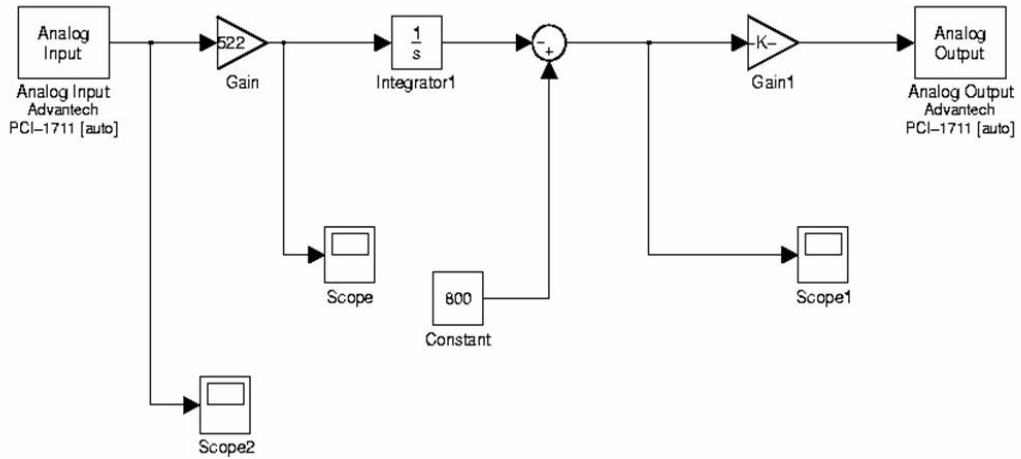


Figura 4.19: Planta no simulink.

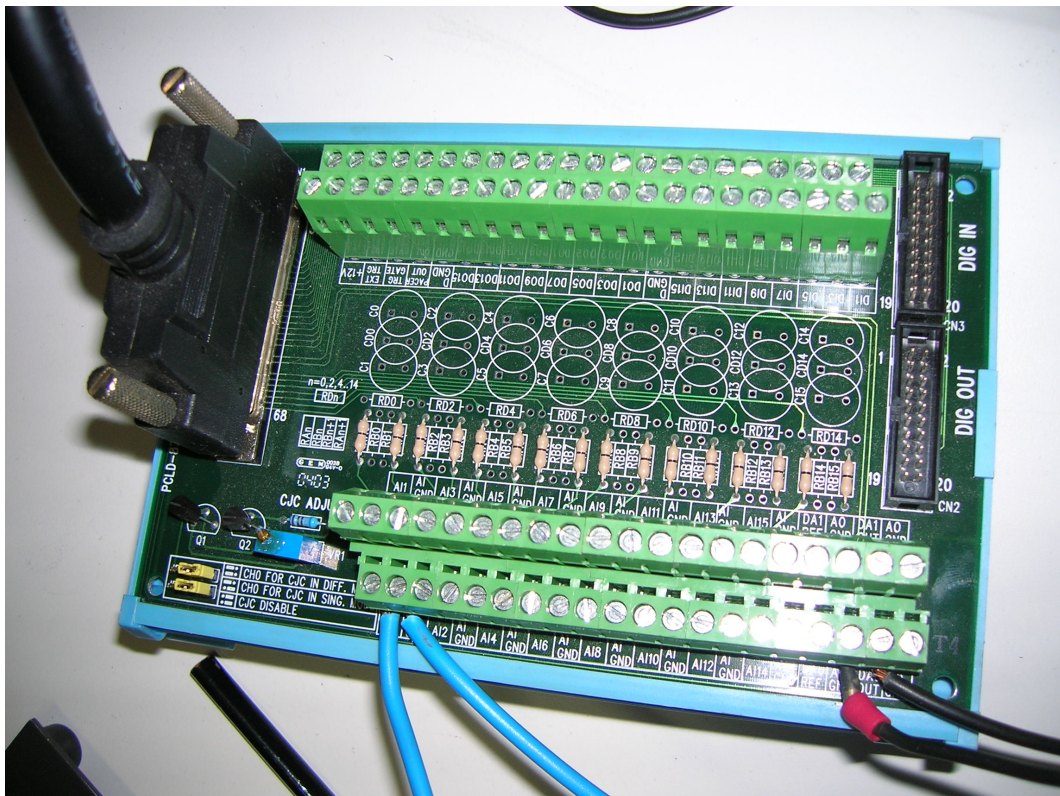


Figura 4.20: Placa PCI1710.

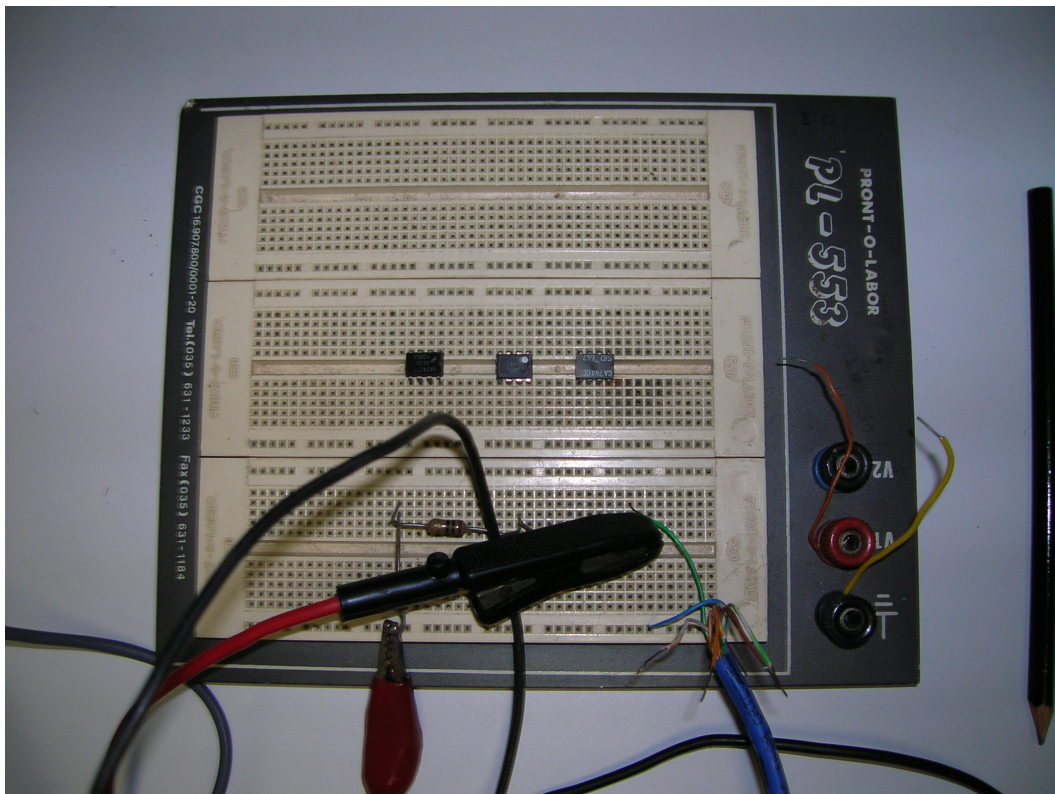


Figura 4.21: Protoboard com três resistores e  $10\Omega$  em série, somando  $30\Omega$ .

em ampères, de 0 a 20 mA, e, assim, para que essa possa servir de entrada para a placa AD/DA, deve ser transformada em tensão. Tal procedimento, por sua vez, é necessário para a entrada do SLC500, já que essa é medida em tensão, que é também a grandeza de saída da placa PCI1710.

O programa em linguagem ladder para o controle da função fechamento da máquina injetora é apresentado nas figuras 4.22, 4.23 e 4.24. Conforme pode ser visto no diagrama da figura 4.22, ela faz uso de duas entradas (denominadas A e B) do CLP SLC5/02, apresentado na figura 4.25, que estão associadas ao funcionamento do parâmetro velocidade na função fechamento da máquina injetora IHP-180/350 e ao controle da chave liga-a-máquina, pré-requisito para o início do funcionamento da máquina injetora.

Os valores atribuídos a  $S$  e  $V$  para esta simulação são fictícios. Foram adotados: (i) abertura máxima: 800 mm; (ii)  $S1(01)$ : 600 mm; (iii) velocidade  $V1(01)$ : 261 mm/s; (iv) distância entre  $S1(01)$  e  $S2(02)$ : 200 mm; (v) velocidade  $V2(02)$ : 157 mm/s; (vi) distância entre  $S2(02)$  e  $S4(11)$ : 200 mm; (vii) velocidade  $V3(03)$ : 104 mm/s; (viii) velocidade VPF entre  $S4(11)$  e 0: 52 mm/s. É importante ressaltar que, de acordo com o exposto na seção 4.1.1, a posição é a variável através da qual se faz a comutação de velocidade.

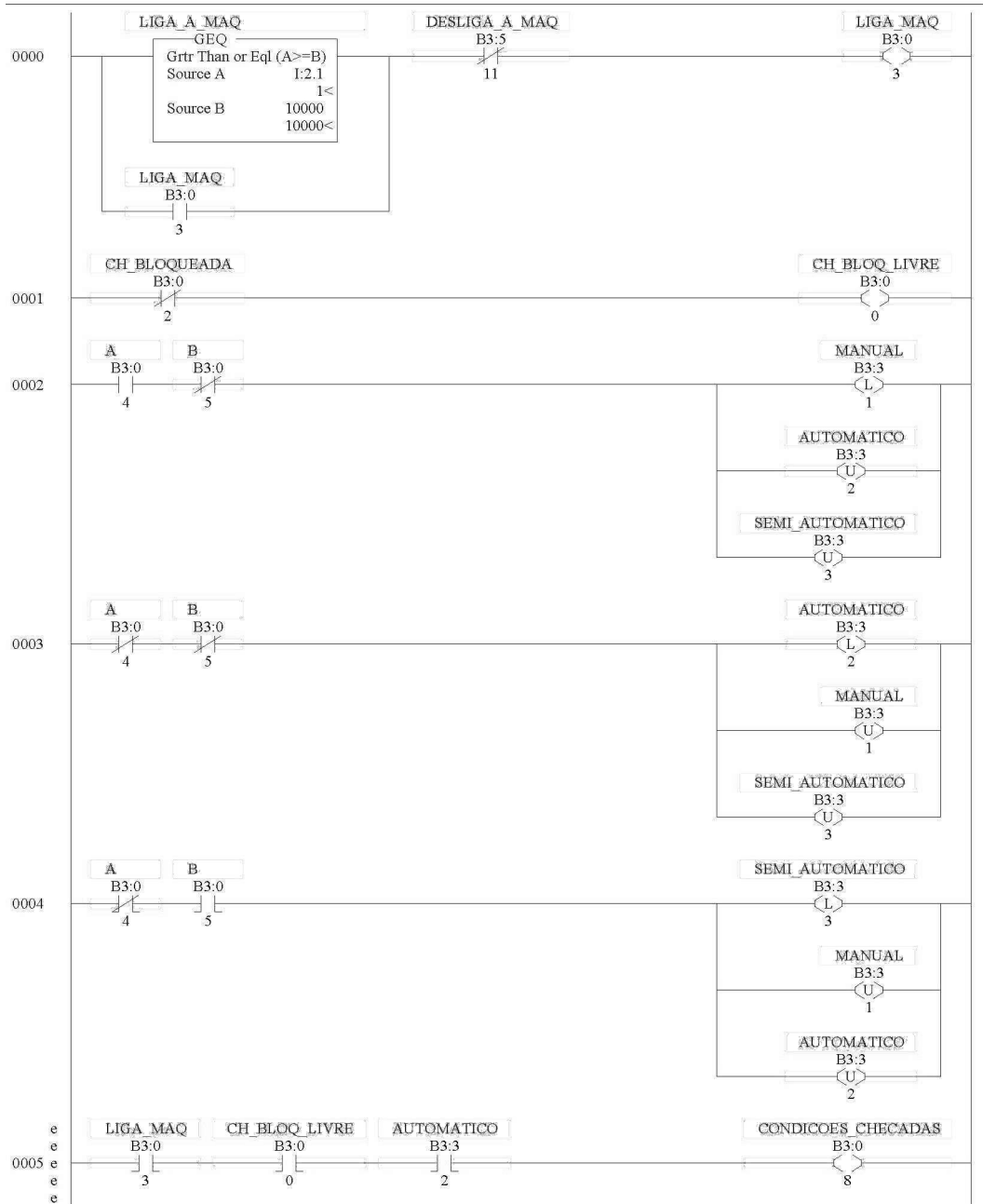


Figura 4.22: Lógica Ladder da simulação (pág. 1).

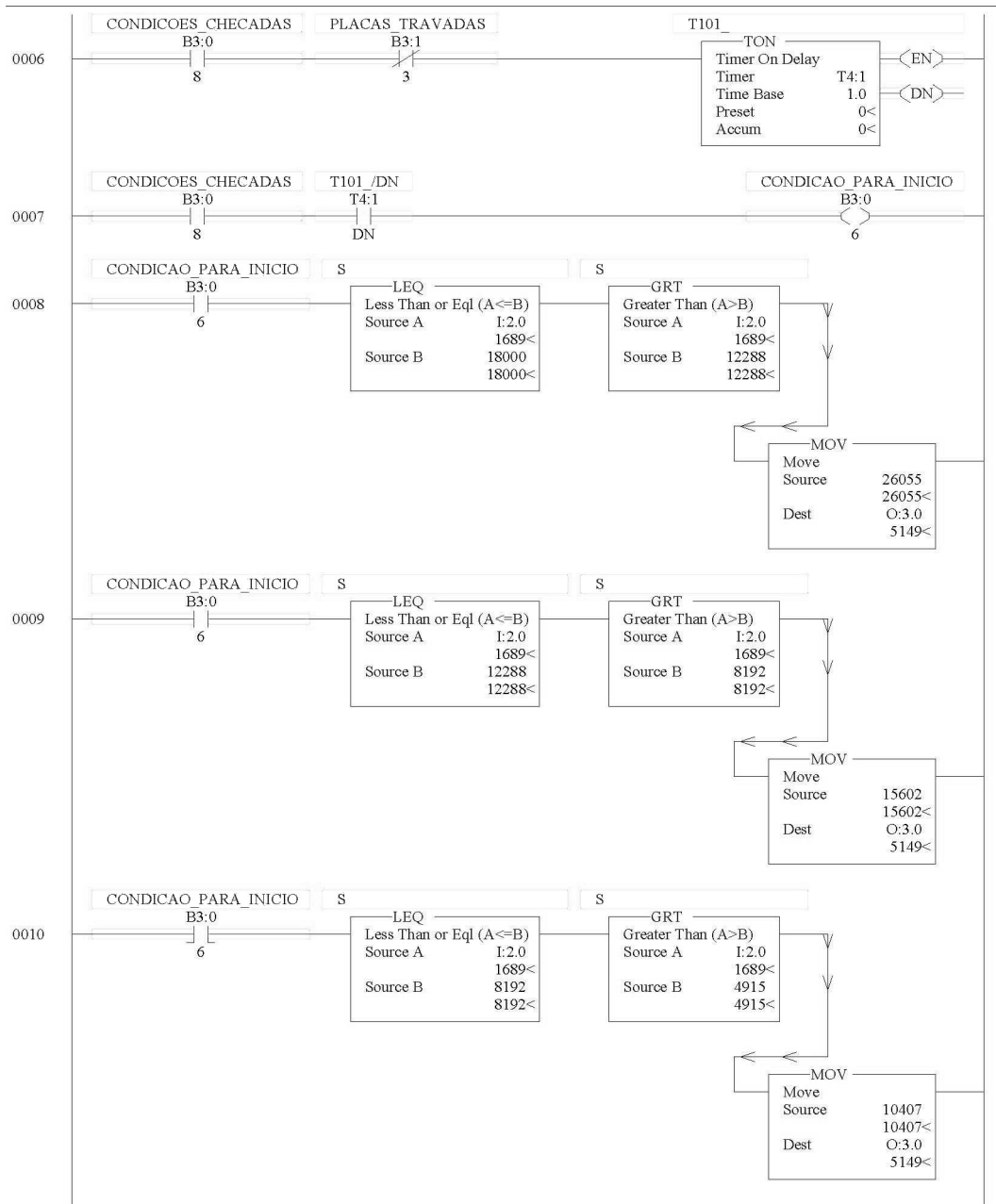


Figura 4.23: Lógica Ladder da simulação (pág. 2).

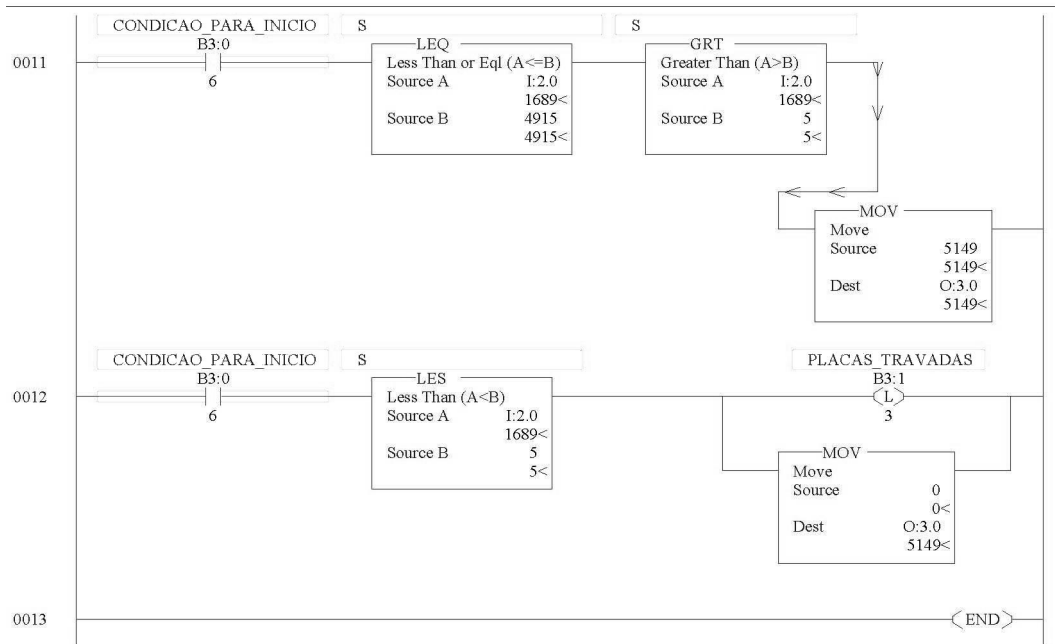


Figura 4.24: Lógica Ladder da simulação (pág. 3).



Figura 4.25: Controlador SLC500.

Considerando-se que ao iniciar o processo a máquina está desligada e que para ligá-la é preciso apertar o botão da entrada I:2.1, correspondente na lógica ao botão *liga – a – maq*, supõe-se que chave bloqueado/livre não está bloqueada, representando-se este estado através do uso da instrução examinar se desenergizado (XIO) para o flag *chave – bloqueada*, que é o padrão inicial (chave em 0). O botão manual/automático/semi-automático é suposto em automático e, para representar este estado, é utilizado um conjunto de chaveamento auxiliar, *A* e *B*, ambos desenergizados (condição padrão) para *automatico*. As condições para início da função fechamento são: máquina ligada, chave bloqueado/livre liberada e controle automático escolhido.

Para simular a função fechamento da máquina injetora, foram calculados os parâmetros que seriam utilizados para posições de comutação de velocidade. Foi utilizado o seguinte processo: considerando que a tensão é dada pelo produto entre resistência e corrente elétrica, e que, segundo o manual do SLC500, a faixa de codificação da saída analógica é de 0 a 20 mA, correspondendo a 0 a 31203,81 bits, para a variação de velocidades, foram escolhidos os pontos apresentados na tabela 4.19. Estes valores em bits serão enviados para a saída do controlador através da instrução MOV, transferindo os valores de corrente correspondentes às velocidades



Tabela 4.19: Conversão dos pontos de corrente.

Correntes em mA	Correntes convertidas
$0,5V/30\Omega = 16,7mA$	$31203,81 * 16,7/20 = 26055,18 \approx 26055$
$0,3V/30\Omega = 10mA$	$31203,81 * 10/20 = 15601,91 \approx 15602$
$0,2V/30\Omega = 6,67mA$	$31203,81 * 6,67/20 = 10406,47 \approx 10407$
$0,1V/30\Omega = 3,3mA$	$31203,81 * 3,3/20 = 5148,63 \approx 5149$

Tabela 4.20: Conversão dos pontos de posição.

Posição em V	Posição em bits
5 V	16400 bits
3,75 V	12288 bits
2,5 V	8192 bits
1,25 V	4915 bits

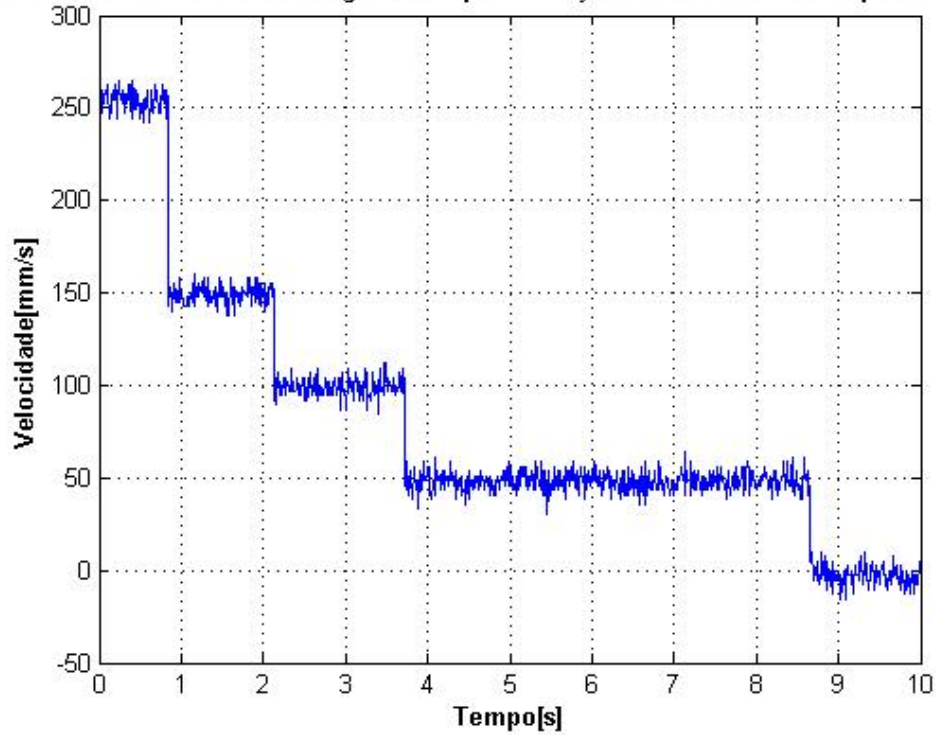
para o protoboard, a fim de serem convertidos em Volts e serem enviados à placa AD/DA.

Ainda, segundo o manual do SLC500, a faixa de codificação da entrada analógica é de 10 V, correspondendo a 0 a 32768 bits, para a variação de posições, foram escolhidos os pontos apresentados na tabela 4.20.

Estes valores em bits serão comparados com os valores da entrada I:2.0 no diagrama ladder, recebidos da placa AD/DA, provenientes do sistema testeinjetera.mdl montado no simulink.

A figura 4.26 mostra a representação gráfica da entrada, já convertida, referente à velocidade, e a figura 4.27 mostra a representação gráfica da saída, referente à posição, obtidas após a realização da simulação. É possível observar que os resultados obtidos indicam que a variação da posição está de acordo com a velocidade prescrita e que as mudanças de velocidade ocorreram nas posições esperadas.

**Gráfico de velocidade ao longo do tempo da função fechamento da máquina injetora**



**Gráfico de posição ao longo do tempo da função fechamento da máquina injetora**

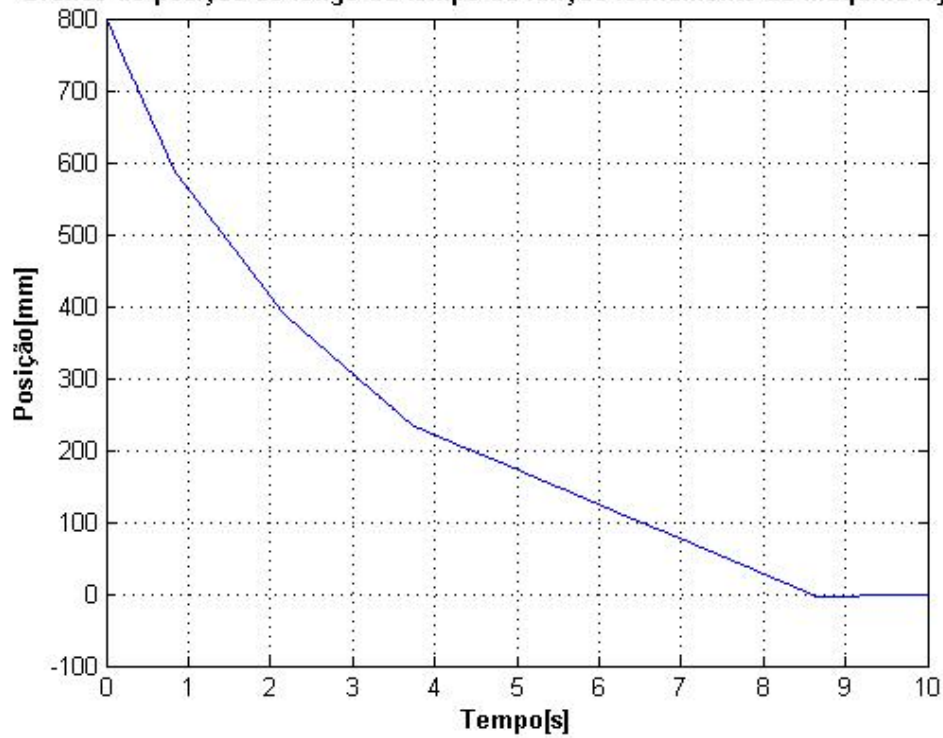


Figura 4.27: Sinal de saída da planta, correspondente à posição.

# Capítulo 5

## Conclusões e trabalhos futuros

Neste trabalho, o sistema de automação da máquina injetora IHP 180/350 (Mecânica Oriente), é considerado, em um nível de abstração, como um sistema a eventos discretos e é modelado por redes de Petri, representando o primeiro passo para a revitalização da máquina injetora existente no Laboratório de Tecnologia Mecânica. A modelagem por redes de Petri foi escolhida devido à sua capacidade de modelar eventos concorrentes. Os objetivos iniciais dessa modelagem são: flexibilizar e facilitar futuras manutenções, já que a modelagem por redes de Petri permite uma melhor visualização do que é modificado; permitir modificações virtuais que não afetem a produção. Outros objetivos são evitar a dependência do fabricante, que, como detentor da tecnologia, impõe seu preço, reduzir os custos e obter melhores respostas na produção, já que automatizando e reduzindo o trabalho manual é possível diminuir mão de obra humana e especializá-la. O fato da implementação de um controle para a máquina injetora poder reduzir ou evitar problemas causados no produto devido à variação de parâmetros como temperatura, pressão e velocidade (seção 3.7) é uma motivação para trabalhos futuros .

### 5.1 Trabalhos futuros

Como trabalho futuro tem-se ainda a implementação do controlador utilizando um CLP. Para tanto será necessário contratar um técnico para identificar os circuitos de controle e as interfaces da máquina existente no laboratório para poder refazer todo o controle usando o CLP. Todo o hardware elétrico e eletrônico da máquina precisa

ser testado e identificado para que se possa implementar o novo sistema de controle (*software* e *hardware*). Para a implementação do controlador para este modelo proposto, é necessário que sejam satisfeitos os requisitos mínimos de suportar vinte e cinco entradas e vinte e sete saídas. Deve-se, ainda, prever uma margem de pelo menos 10% a mais de entradas e saídas e o uso de controladores PID para suavizar as variações de velocidades nas funções fechamento/abertura da placa móvel e injeção, como alternativas para acréscimos futuros.

Outro trabalho que pode ser realizado no futuro é adicionar um controle híbrido, capaz de combinar as partes discreta e contínua. No processo de injeção pode-se utilizar ferramentas inteligentes, como redes neurais, capaz de realizar as tarefas e também resolver problemas na linha durante a produção, reduzindo perdas. Deve-se ressaltar que um sistema de controle flexível pode ter seus parâmetros modificados entre os ciclos de injeção, cujos valores podem ser obtidos a partir de sistemas baseados em inteligência artificial que tenham como dados de entrada características desejadas da peça e os sinais de temperatura e pressão em diferentes regiões do molde.

Quanto ao modelo, como trabalhos futuros pode-se implementar a rede de Petri para a geração do programa *ladder*, a ser usado com o CLP e fazer a conjugação do modelo de redes de Petri com um simulador híbrido, que tenha suporte a elementos discretos e contínuos.

Há ainda algumas abordagens de otimização do processo de injeção recentes, capazes de melhorar a eficiência da linha de produção, como é o caso de simuladores de moldagem de plástico por injeção (Carneiro, 2006), ou o surgimento de novas técnicas, como a desmoldagem com tempo estabelecido, na qual a peça moldada alcança um determinado estado térmico (Thienel et al., 2006). Nestes casos pode ser necessária a aquisição de equipamentos, como por exemplo um sensor integrado para medição de temperatura e pressão. Assim, um trabalho futuro também pode vir a contemplar a introdução na máquina injetora de dispositivos que permitam a utilização desses novos recursos tecnológicos.

# Apêndice A

## Circuito hidráulico da Máquina Injetora 180/350

O circuito hidráulico da máquina injetora de plástico 180/350 mostrando as solenóides nas válvulas correspondentes é apresentado na figura A.1.

O pressostato nessa máquina é um equipamento de proteção e envia um sinal para que a máquina seja desligada caso a pressão fique alta demais. Para o início da injeção há um pressostato. Nas funções fechamento e injeção há sensores de pressão. Para o controle com CLP, este deve receber do sensor de pressão o dado e passá-lo para a válvula reguladora de pressão.

O controle de velocidade em malha aberta é feita com o envio de uma vazão, ou seja, uma quantidade de óleo por unidade de tempo que está entrando no cilindro. Em malha fechada é calculado a partir de duas posições da régua resistiva.

Quanto ao controle de posição, com CLP, este deve receber da régua resistiva de posição o dado e passá-lo para um cartão (circuito impresso).

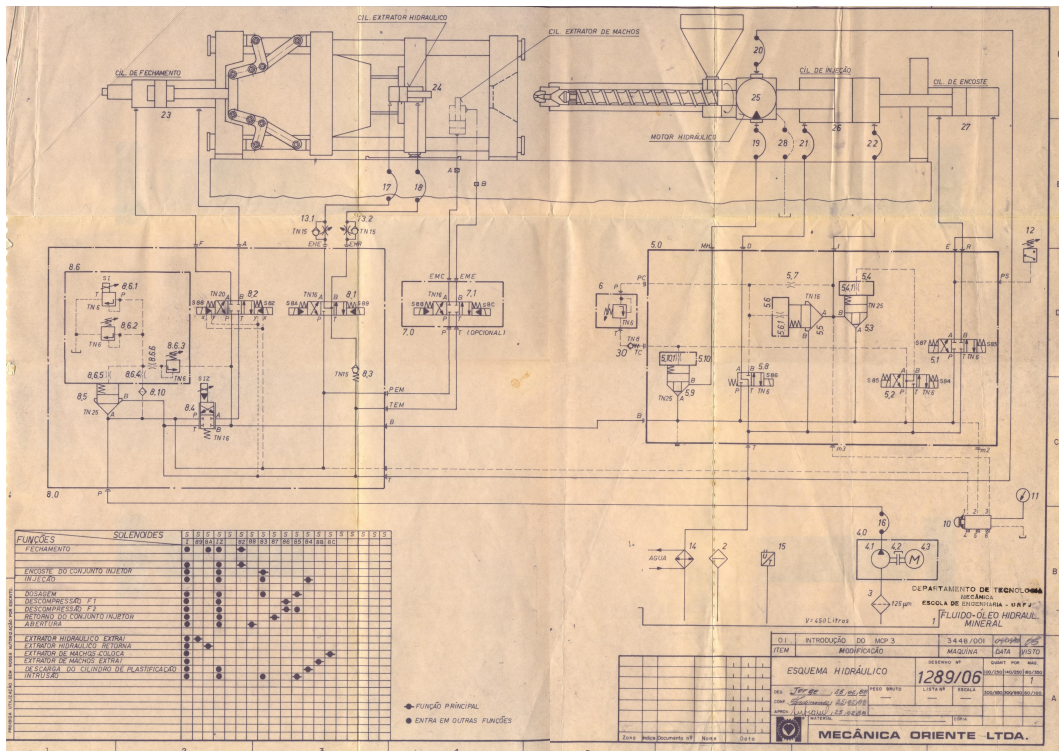


Figura A.1: Circuito hidráulico da máquina injetora.

# Bibliografia

- Advantech (2004). *User's Manual PCI-1710 Series 12/16bit Multifunction Card*.
- Al-Jaar, R. Y. e Desrochers, A. A. (1990). Performance evaluation of automated manufacturing systems using generalized stochastic Petri nets, *IEEE Transactions on Robotics and Automation*, vol. 6(6), pp. 621–639.
- Basilio, J. C. (2005). *Notas de aula de Automação Industrial*, UFRJ Departamento de Eletrotécnica, baseada em Cassandras e Lafortune, 2000, Rio de Janeiro : Brasil.
- Blass, A. (1985). *Processamento de Polímeros*, Editora da UFSC, Florianópolis : Brasil.
- Bouyekhf, R. e Moudni, A. E. (2005). On the analysis of some structural properties of Petri nets, *IEEE Transactions on Systems, Man and cybernetics* , vol. 35(6): parte A, pp. 784–794.
- Calvo, R. N. (2006). *Uso de Observadores em Malha Fechada no Controle de Velocidade de Motores de Corrente Contínua*, Projeto Final de Curso, UFRJ, Departamento de Eletrotécnica, Rio de Janeiro : Brasil.
- Canabrava Filho, J. S. (1998). *Processo de injeção*, Rio de Janeiro : Brasil. Apresentação em PowerPoint, UFRJ, Laboratório de Tecnologia Mecânica, baseada em Blass, 1985.
- Carneiro, M. S. (2006). Transferência térmica determina o bom desempenho de moldes de injeção, *Revista Plástico Industrial* , nº 91, pp. 108–119.

- Cassandras, C. G. e Lafortune, S. (2000). *Introduction to Discrete Event Systems*, Kluwer Academic Publishers. 2ª Edição, Boston : EUA.
- Cazzola, F., Ferrarini, F. e Preziosa, M. (1995). Interpretation rules of Petri net models for logic control, *IEEE Conference on Emerging Technology and Factory Automation* , vol. 2: pp. 289–297.
- Crockett, D., Desrochers, A., Dicesare, F. e Ward, T. (1987). Implementation of a Petri net controller for a machining workstation, *Proc. of IEEE Int'l Conf on Robotics and Automation*, Raleigh : EUA, pp. 1861–1867.
- Cury, J. E. R. (2001). Teoria de controle supervisorio de sistemas a eventos discretos, *V Simpósio Brasileiro de Automação Inteligente*, Canela : Brasil, pp. 9–52.
- David, R. (1995). Grafcet: A powerful tool for specification of logic controllers, *IEEE Transactions on Control Systems Technology* , vol. 3(3): pp. 253–268.
- Dias, J. R. S. (2005). Um laboratório para um curso de automação industrial utilizando a teoria de sistemas a eventos discretos, *Tese de Mestrado, UFRJ, COPPE, Programa de Engenharia Elétrica*, Rio de Janeiro : Brasil.
- Feldman, K., Colombo, A. W., Schur, C. e Stockel, T. (1999). Specification, design, and implementation of logic controllers based on colored Petri net models and the standard IEC 1131 part I: specification and design, *IEEE Transactions on Control Systems Technology* , vol. 7(6): pp. 657–665.
- Ferrarini, L. (1992). An incremental approach to logic controllers design with Petri nets, *IEEE Transactions on Systems, Man and Cybernetics* , vol. 22(3): pp. 461–473.
- Ferrarini, L., Narduzzi, M. e Tassan-Solet, M. (1994). A new approach to modular liveness analysis conceived for large logic controllers' design, *IEEE Trans. Robot. Automat.* , vol. 10(2): pp. 169–184.
- Freedman, P. (1991). Time, Petri nets, and robotics, *IEEE Transactions on Robotics and Automation*, vol. 7(4): pp. 417–433.



- Frey, G. (2000). Automatic implementation of Petri net based control algorithms on PLC, *Proceedings of the American Control Conference*, Chicago : EUA, pp. 2819–2823.
- Frey, G. e Litz, L. (1999). A measure for transparency in net based control algorithms, *IEEE International Conference on Systems, Man, and Cybernetics*, Tóquio : Japão, pp. 887–892.
- Frey, G. e Litz, L. (2000a). Formal methods in PLC programming, *Proceedings of the IEEE SMC 2000*, Nashville : EUA, pp. 2431–2436.
- Frey, G. e Litz, L. (2000b). Transparency analysis of Petri net based logic controllers: a measure for software quality in automation, *Proceedings of the American Control Conference*, Chicago : EUA, pp. 3182–3186.
- Galdamez, E. V. C. e Carpinetti, L. C. R. (2004). Aplicação das técnicas de planejamento e análise de experimentos no processo de injeção plástica, *Gestão & Produção* , vol. 11(1): pp. 121–134.
- Giua, A. e Dicesare, F. (1994). Petri net structural analysis for supervisory control, *IEEE Trans. Robot. Automat.* , vol. 10: pp. 185–195.
- Gomes, L. e Barros, J. P. (2003). On structuring mechanisms for Petri net based system design, *ETFA 2003: 2003 IEEE Conference on Emerging Technologies and Factory Automation Proceedings*, Lisboa : Portugal, pp. 16–19.
- Gomes, L. e Barros, J. P. (2005). Structuring and composability issues in petri nets modeling, *IEEE. Transactions on Industrial Informatics* , vol. 1(2): pp. 112–123.
- Han, Y., Jang, C. e Luo, X. (2004). A study of concurrency control in web-based distributed real-time database system using extended time Petri nets, *2004 International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN'04)*, Hong Kong : China, pp. 67–72.

- Hillion, H. P. e Proth, J. (1989). Performance evaluation of job-shop system using timed event-graphs, *IEEE Transaction on Automatic Control* , vol. 34(1): pp. 3–9.
- Holloway, L. E., Guan, X., Sundaravadivelu, R. e Ashley Jr., J. (2000). Mtl internet: a university microelectronics laboratory www-based management system, *IEEE Transactions on Industrial Eletronics* , vol. 30(5): pp. 696–712.
- Ipiranga (1998). *Apostila de Injeção da Petróleo Ipiranga*, Laboratório de Tecnologia Mecânica, Rio de Janeiro : Brasil. UFRJ.
- Jiang, J., Azzopardi, D., Holding, D. J., Carpenter, G. F. e Sagoo, J. S. (1996). Real-time synchronization of multi axis high-speed machines, from SFC specification to Petri net verification, *Proc. Inst. Elect. Eng., Control Theory Appl.* , vol. 143(2): pp. 164–170.
- Jones, A. H., Uzam, M. e Ajlouni, N. (1996). Design of discrete event control systems for programmable logic controllers using t-timed Petri nets, *Proc. 1996 IEEE Int. Symp. on Computer-Aided Control System Design*, Dearborn : EUA, pp. 212–217.
- Kang, B. S. e Cho, K. H. (2001). Design of PLCs for automated industrial systems based on discrete event models, *IEEE International Symposium on Industrial Electronics* , vol. 8, Pusan : Coréia do Sul, pp. 1431–1434.
- Koh, I. e Dicesare, F. (1991). Synthesis rules for colored Petri nets and their applications to automated manufacturing systems, *Proceedings of IEEE International Symposium on Intelligent Control*, Virginia: EUA, pp. 152–157.
- Koriem, S. M., Dabbous, T. E. e El-Kilani, W. S. (2004). A new Petri net modeling technique for the performance analysis of discrete event dynamic systems, *J. Syst. Softw.* , vol. 72(3): pp. 335–348.

- Koutsoukos, X. D., Antsaklis, P. J., Stiver, J. A. e Lemmon, M. D. (2000). Supervisory control of hybrid systems, *Proceedings of the IEEE*, , vol. 88(7), pp. 1026–1049.
- Kumar, R. e Garg, V. K. (1995). Modeling and control of logical discrete event systems, *Kluwer Academic Publishers*, Norwell: EUA.
- Lamport, L. (1986). *A Document Preparation System Latex: User's Guide & Reference Manual*, Addison-Wesley Publishing Company, Massachusetts : EUA.
- Lauzon, S. C., Ma, A. K. L., Mills, J. K. e Benhabib, B. (1996). Application of discrete event system theory to flexible manufacturing, *IEEE Control Systems Magazine* , vol. 16(1): pp. 41–48.
- Lee, J. S. e Hsu, P. L. (2000). A PLC-based design for the sequence controller in discrete event systems,, *IEEE International Conference on Control Applications*, Anchorage : EUA, pp. 929–934.
- Lee, J. S. e Hsu, P. L. (2001). A new approach to evaluate ladder logic diagrams and Petri nets via the IF-THEN transformation, *IEEE Conference on Systems, Man and Cybernetics*, Tucson : EUA, pp. 2711–2716.
- Liao, X., Lao, J. e Jiang, M. (2004). Modeling of real-time monitoring and simulation for plastic injection molding process, *Proceedings of the 2004 international Conference on Intelligent Mechatronics and Automation*, Chengdu : China.
- Lin, M. e Fu, L. (1999). Modeling of priority queueing service in discrete event systems using hybrid Petri nets, *IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC'99)*, vol. 1, Tóquio : Japão, pp. 818–823.
- Lucas, M. R. e Tilbury, D. M. (2003). A study of current logic design practices in the automotive manufacturing industry, *International Journal of Human-Computer Studies* , vol. 59: pp. 725–753.
- Machado, A. (1993). *O imaginário numérico*, Edusp, São Paulo : Brasil.

- Maraghi, R. (1997). *Defeitos de moldagem na injeção de plásticos*, Plassoft, Salvador : Brasil.
- Marranghello, N. (2005). *Redes de Petri: Conceitos e Aplicações*, DCCE/IBILCE/UNESP, São Paulo : Brasil.
- Mazigh, B., Gresser, J. e Simon, F. (1993). Well suited modelling and evaluation techniques based on GSPN for real production systems, *Proceedings of the Workshop on Emerging Technologies and Factory Automation: Design and Operations of Intelligent Factory*, Queensland : Austrália, pp. 169–176.
- Mellado, E. L. (2002). Analysis of discrete event systems by simulation of timed Petri net models, *Mathematics and Computers in Simulation* , vol. 61(1): pp. 53–59.
- Minas, M. e Frey, G. (2002). Visual PLC-programming using signal interpreted Petri nets, *Proceedings of the 2002 American Control Conference* , vol. 6: pp. 5019–5024.
- Murata, T. (1989). Petri nets: properties, analysis and applications, *Proc. of the IEEE* , vol. 77: pp. 541–580.
- Oriente M. (1998). *Catálogo da Injetora Oriente IHP-180/350*, São Paulo: Brasil.
- Park, E., Tilbury, D. M. e Khargonekar, P. P. (1999). Modular logic controllers for machining systems: formal representation and performance analysis using Petri nets, *IEEE Transactions on Robotics and Automation* , vol. 15(6): pp. 1046–1061.
- Park, E., Tilbury, D. M. e Khargonekar, P. P. (2001). A modeling and analysis methodology for modular logic controllers of machining systems using Petri net formalism, *IEEE Transactions on Systems, Man, and Cybernetics: Part C: Applications and Reviews* , vol. 31(2): pp. 168–188.

- Peng, S. S. e Zhou, M. C. (2001a). Conversion between ladder diagrams and Petri net in discrete event control design: A survey, *IEEE Conference on Systems, Man and Cybernetics*, Tucson : EUA, pp. 2682–2687.
- Peng, S. S. e Zhou, M. C. (2001b). Petri net based PLC stage programming for discrete-event control design, *Proc. 2001 IEEE Int. Conf. on Systems, Man, and Cybernetics*, vol. 4, Tucson : EUA, pp. 2706–2710.
- Peng, S. S. e Zhou, M. C. (2004). Ladder diagram and Petri net based discrete event control design methods, *IEEE Transactions on Systems, Man, and Cybernetics: Part C: Applications and Reviews* pp. 1–9.
- Peterson, J. L. (1981). *Petri Net theory and the modeling of systems*, Prentice-Hall, Englewood Cliffs : EUA.
- Ponizil, D. (2001). Back to basics: the essentials of structured PLC, *Control Engineering* , vol. 48(76).
- Ramadge, P. J. G. e Wonham, W. M. (1989). The control of discrete event systems, *Proceedings of the IEEE* , vol. 77(1): pp. 81–98.
- Reisig, W. (1985). *Petri Nets: An Introduction*, Springer-Verlag, Berlim : Alemanha.
- Satoh, T., Oshima, H., Nose, K. e Kumagai, S. (1992). Automatic generation system of ladder list program by Petri net, *Proceedings of the IEEE International Workshop on Emerging Technologies on Factory Automation*, Melbourne : Australia, pp. 128–133.
- Schneeweiss, W. G. (2001). Tutorial: Petri nets as a graphical description medium for many reliability scenarios, *IEEE Transactions on reliability* , vol. 50(2): pp. 159–164.
- Souza, J. R. B., Lima, A. M. N., Perkusich e A., (2005). Redes de Petri híbridadas diferenciais: aplicação na modelagem do consumo de energia de um nó sensor sem fio , *Anais do SBAI 2005*, Maranhão : Brasil, pp. 123–131.

- Thienel, P., Schumacher, B. e Schnerr, O. (2006). Pesquisadores propõem nova abordagem do tempo de resfriamento, *Revista Plástico Industrial* , nº 91, pp. 42–49.
- Tsinarakis, G. J., Valavanis, K. P. e Tsourveloudis, N. C. (2003). Modular Petri net based modeling, analysis and synthesis of dedicated production systems, *International Conference on Robotics & Automation 2003*, Taiwan : China, pp. 3559–3564.
- Tsinarakis, G. J., Valavanis, K. P. e Tsourveloudis, N. C. (2005). Modular Petri net based modeling, analysis, synthesis and performance evaluation of random topology dedicated production systems, *Journal of Intelligent Manufacturing* , vol. 16(1): pp. 67–92.
- Uzam, M., Jones, A. e Ajlouni, N. (1996). Conversion of Petri net controllers for manufacturing systems into ladder logic diagrams, *IEEE Symposium on Emerging Technology and Factory Automation, ETFA*, vol. 2, Hawaii : USA, pp. 649–655.
- Uzam, M., Jones, A. e Yucel, I. (2000). Using a Petri-net-based approach for the real-time supervisory control of an experimental manufacturing system, *International Journal of Advanced Manufacturing Technology* , vol. 16(7): pp. 498–515.
- Valavanis, K. P. (1990). On the hierarchical modeling analysis and simulation of flexible manufacturing systems with extended Petri nets, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20(1), pp. 94–110.
- Vandoren, V. (1996). Designing PLC-based control without ladder logic, *Control Engineering*, vol. 43: pp. 110.
- Venkatesh, K., Zhou, M. C. e Caudill, R. J. (1994a). Comparing ladder logic diagrams and Petri nets for sequence controller design through a discrete manufacturing system, *IEEE Transactions on Industrial Electronics* , vol. 41(6): pp. 611–619.

- Venkatesh, K., Zhou, M. C. e Caudill, R. J. (1994b). Evaluating the complexity of Petri nets and ladder logic diagrams and for sequence controllers design in flexible automation, *Proceedings of the 1994 IEEE Symposium on Emerging Technologies e factory Automation*, pp. 428–435.
- Webb, J. e Reis, R. A. (1999). *Programmable Logic Controllers: Principles and Applications*, Fourth Edition - Prentice-Hall, Nova Jersey : EUA.
- Wegrzyn, M. e Wolanski, P. (1997). Coloured Petri net model of application specific logic controller programs, *Proceedings of IEEE International Symposium on Industrial Eletronics*, vol. 8, Guimarães : Portugal, pp. pp. 350–361.
- Zhou, M. C., Dicesare, F. e Desrochers, A. (1992). A hybrid methodology for synthesis of Petri nets for manufacturing systems, *IEEE Transactions on Robotics and Automation* , vol. 8(3): 350–361.
- Zhou, M. C. e Twiss, E. (1998). Design of industrial automated systems via relay ladder logic programming and Petri nets, *IEEE Transactions on systems, man, and cybernetics: Part C: applications and reviews* , vol. 28(1): pp. 137–150.
- Zhou, M. C., Mcdermott, K. e Patel, P. A. (1993). Petri net synthesis and analysis of a flexible manufacturing system cell, *IEEE Trans. on Systems, Man, and Cybernetics* , vol. 23(2): pp. 523–531.
- Zhou, M. e Venkatesh, K. (1999). *Modeling, Simulation and Control of Flexible Manufacturing Systems: A Petri Net Approach*, World Scientific Publishing Co, Nova York : EUA.
- Zurawski, R. (2005). Petri net models, functional abstractions, and reduction techniques: Applications to the design of automated manufacturing systems, *IEEE Transactions on Industrial Electronics* , vol. 52(2): pp. 595–609.
- Zurawski, R. e Zhou, M. C. (1994). Petri nets and industrial application: A tutorial, *IEEE Transactions on Industrial Electronics* , vol. 41(6): pp. 567–583.