

OK

CONSULTA  
FD-3675  
Ed.rev.

SÃO PAULO  
2004

Dissertação apresentada à Escola  
Politécnica da Universidade de São  
Paulo para obtenção do título de  
Mestre em Engenharia

MODELAGEM ORIENTADA A OBJETOS DE  
SISTEMAS A EVENTOS DISCRETOS: ESTUDO DE  
CASO NA SÍNTESE DE CONTROLE DE SISTEMAS  
PREDIAIS

PEDRO LUIS ANGEL RESTREPO

PEDRO LUIS ANGEL RESTREPO

**MODELAGEM ORIENTADA A OBJETOS DE  
SISTEMAS A EVENTOS DISCRETOS: ESTUDO DE  
CASO NA SÍNTESE DE CONTROLE DE SISTEMAS  
PREDIAIS**

Dissertação apresentada à Escola  
Politécnica da Universidade de São  
Paulo para obtenção do título de  
Mestre em Engenharia

Area de Concentração:  
Engenharia Mecatrônica

Orientador:

Prof. Dr. Jose Reinaldo Silva

SÃO PAULO

2004

*A mi viejo*  
*In Memoriam*

## RESUMO

Neste trabalho apresenta-se a definição da rede *GHENeSsys* (*General Hierarchical Enhanced Net System*), sua equação de estado, os algoritmos computacionais para um jogador de marcas baseado na sua equação de estado, a ferramenta computacional que os implementa e uma aplicação na síntese de controle.

A *GHENeSsys* é uma Rede de Petri, estendida e orientada a Objetos que permite o tratamento hierárquico do processo de projeto. O tema base é uma proposta na direção da fusão do *Petri Net Inside Objects* e do *Object Inside Petri Nets*, mantendo a representação algébrica e a análise de propriedades clássica.

Apresenta-se um estudo de caso no campo da automação predial, consistindo de um prédio com uma política de utilização definida mediante regras heurísticas. Estas regras são utilizadas para a síntese de um sistema supervisorio para o Prédio, mediante o conceito de objetos de controle.

# ABSTRACT

This work presents a formal definition of a Petri Net named *GHENeSys Ge-neral Hierarchical Enhanced Net System*, We show its state equation, algorithms for a computational tool based on classic state equation, the software that make it useful and an application on control synthesis.

*GHENeSys* is an Extended Object Oriented Net than allow hierarchical design of systems, the principal interest of this work is an intent of melting *Petri Net Inside Objects* and *Object Inside Petri Nets*, but keeping the classic algebraic representation and properties analysis.

Is presented an case study on Building Automation, where the use politics represented by heuristic rules, are used to make the synthesis of an supervisory system using the concept of *Control Object*.

# Sumário

RESUMO	4
ABSTRACT	5
1 INTRODUÇÃO	1
2 OBJETIVO	4
3 REVISÃO BIBLIOGRÁFICA	5

3.1	Redes de Petri . . . . .	5
3.1.1	Princípios de Representação Alébrica . . . . .	7
3.1.1.1	Vetor de Capacidades . . . . .	8
3.1.1.2	Vetor de Marcação . . . . .	9
3.1.1.3	Transições Habilitadas . . . . .	9
3.1.1.4	Vetor de Habilitação . . . . .	10
3.1.1.5	Equação de Estado . . . . .	10
3.1.1.6	Deteção de Conflitos . . . . .	11
3.1.2	Dinâmica da Rede de Petri Ordinária . . . . .	12
3.1.3	Redes Estendidas e de Alto Nível . . . . .	13
3.1.4	Linguagens . . . . .	13
3.2	Orientação a Objetos . . . . .	14
3.2.1	Conceitos Básicos . . . . .	14
3.2.2	Análise Orientada a Objetos . . . . .	16
3.3	Orientação a Objetos e Redes de Petri . . . . .	16
3.3.1	Objetos Dentro de Redes de Petri . . . . .	17
3.3.2	Redes de Petri Dentro de Objetos . . . . .	17
3.3.3	Mútua Integração das Técnicas de Orientação a Objetos e Redes de Petri . . . . .	18
3.4	Hierarquia em Redes de Petri . . . . .	21
3.4.1	Composição Hierárquica . . . . .	22
3.4.2	Refinamento . . . . .	22
3.4.3	Refinamento Hierárquico . . . . .	22
3.5	Controle de Sistemas a Eventos Discretos com Redes de Petri . . . . .	23
3.5.1	Controle de Sistemas de Eventos Discretos . . . . .	23

24	3.6	Prédios Inteligentes . . . . .
24	3.6.1	Inteligência nos Prédios . . . . .
26	3.6.2	Diferença entre um Prédio Inteligente e um Prédio Auto- matizado . . . . .
26	3.6.2.1	Prédio Automatizado . . . . .
26	3.6.2.2	Prédio Inteligente . . . . .
26	3.6.3	Sistemas do Prédio Inteligente . . . . .
28	3.7	Realidade Virtual . . . . .
28	3.7.1	Conceitos Fundamentais . . . . .
29	3.7.2	Programação de Ambientes Virtuais . . . . .
30	4	<b>REDE <i>GHENeSys</i></b> . . . . .
30	4.1	Conceitos Básicos . . . . .
32	4.2	Diagrama de Classes . . . . .
33	4.3	Regras de Compatibilidade entre Redes e MacroElementos . . . . .
33	4.4	Efeito da Marcação dos Pseudobox sobre os Elementos Ativos . . . . .
33	4.5	Efeito do Fluxo de mensagens na Marcação dos Pseudobox . . . . .
35	4.6	Definição Formal . . . . .
35	4.7	Representação Algebrica . . . . .
37	4.7.1	Matriz de Incidência . . . . .
38	4.7.2	Efeito da Marcação dos Pseudobox no Fluxo de Marcas . . . . .
39	4.7.3	Vetor de Capacidades . . . . .
39	4.7.4	Vetor de Marcação . . . . .
40	4.7.5	Vetor de Habilitação . . . . .
40	4.7.5.1	Habilitação por Marcação . . . . .
40	4.7.5.2	Efeito dos arcos Habilitadores e Inibidores . . . . .
43	4.7.6	Equação de Estado . . . . .
43	4.7.7	Deteção de Conflitos . . . . .
44	4.7.8	Cálculo de Efeito de Mensagens nos Pseudobox . . . . .
45	4.7.9	Dinâmica da Rede <i>GHENeSys</i> . . . . .
46	4.8	Exemplo de Abstração . . . . .
51	4.9	Projeto de Sistemas com a Rede <i>GHENeSys</i> . . . . .
54	5	<b><i>GHENeSys</i>: O SISTEMA COMPUTACIONAL</b> . . . . .
54	5.1	Caraterísticas Gerais . . . . .
65	5.2	Caraterísticas Específicas . . . . .

65	5.3	Caraterísticas Adicionais . . . . .	65
66	5.4	Crítérios para a Seleção do Compilador . . . . .	66
66	5.5	Aspectos Computacionais . . . . .	66
66	5.5.1	Algoritmo de Hierarquização . . . . .	66
68	5.5.2	Algoritmo de Execução da Rede <i>GHENeSys</i> . . . . .	68
68	5.6	Exemplo da Dinâmica da Rede <i>GHENeSys</i> . . . . .	68
70	5.6.1	Gerção da Arvore Hierarquica . . . . .	70
72	5.6.2	Matrizes do Sistema . . . . .	72
73	5.6.3	Vetores de Capacidade . . . . .	73
73	5.6.4	Vetores de Marcação . . . . .	73
73	5.6.5	Cálculo das Transições Habilitadas . . . . .	73
75	5.6.6	Cálculo do Efeito da Presença de Arcos Habilitadores e Inibidores . . . . .	75
76	5.6.7	Cálculo de Conflitos . . . . .	76
79	5.6.8	Aplicação da Equação de Estado . . . . .	79
80	5.7	Descrição Geral da <i>GHENeSys</i> . . . . .	80
84	5.7.1	Interface Gráfica da <i>GHENeSys</i> . . . . .	84
84	5.7.1.1	Módulo de Sistema Operacional . . . . .	84
85	5.7.1.2	Módulo de Desenho . . . . .	85
87	5.7.1.3	Módulo de Administração do Projeto . . . . .	87
89	5.7.1.4	Módulo de Execução . . . . .	89
91	5.7.1.5	Módulo de Ajuda . . . . .	91
92	<b>6</b>	<b>ESTUDO DE CASO</b>	92
92	6.1	Controle de Sistemas a Eventos Discretos com a Rede <i>GHENeSys</i> . . . . .	92
92	6.2	Identificação do Sistema . . . . .	92
92	6.2.1	Descrição do Prédio . . . . .	92
92	6.2.2	Políticas de Utilização do Prédio . . . . .	92
93	6.3	Decomposição do Sistema . . . . .	93
93	6.3.1	Objetos do Prédio . . . . .	93
94	6.3.2	Dispositivos de Automação do Prédio . . . . .	94
94	6.3.3	Objetos Controladores . . . . .	94
96	6.4	Modelos dos Objetos . . . . .	96
96	6.4.1	Modelos dos Objetos com o Sistema de Mensagens . . . . .	96
99	6.4.2	Modelos dos Objetos Controladores . . . . .	99
113	6.5	Encapsulamento dos Objetos em MacroElementos . . . . .	113



122	6.6	Árvore de Hierarquia
128	7	CONCLUSÕES E TRABALHOS FUTUROS
128	7.1	Conclusões
129	7.2	Trabalhos Futuros
130		PLANTAS
131	A4-01	
132	A4-02	
133	A4-03	
134	A4-04	
135	A4-05	
136	A4-06	
137	A4-07	
138	A4-08	
139	A4-09	
140	A4-10	
141	A4-11	
142	A4-12	
143	A4-13	
144	A2-01	
145		APÊNDICES
145	A.1	Critérios para a Seleção do Compilador
147		REFERÊNCIAS BIBLIOGRÁFICAS

# Lista de Figuras

6	3.1	Elementos de uma Rede de Petri	6
11	3.2	Conflito em Redes de Petri	11
12	3.3	Algoritmo da Dinâmica de uma Rede de Petri Ordinária	12
17	3.4	Conceito de Objetos Dentro de Redes de Petri	17
19	3.5	Conceito de Redes de Petri Dentro de Objetos	19
20	3.6	Mútua Integração das Técnicas de OO e Redes de Petri	20
23	3.7	Transformações Preservando Vivacidade, Segurança, e Limitação	23
24	3.8	Controle Supervisório de Sistemas a Eventos Discretos	24
30	4.1	Elementos Estendidos da Rede <i>GHENeSys</i>	30
30	4.2	Representação dos Elementos Marcados na Rede <i>GHENeSys</i>	30
32	4.3	Exemplo de um modelo em Rede <i>GHENeSys</i>	32
32	4.4	Diagrama de Classes da Rede <i>GHENeSys</i>	32
34	4.5	Efeito da Marcação dos Pseudobox sobre os Elementos Ativos	34
45	4.6	Algoritmo da Dinâmica de uma Rede <i>GHENeSys</i>	45
47	4.7	Exemplo de Abstração e Encapsulamento	47
49	4.8	Subrede Encapsulada em uma MacroAtividade	49
52	4.9	Abstração na Rede sem Marcação Consistente	52
53	4.10	Método de Projeto de Sistemas com a Rede <i>GHENeSys</i>	53
67	5.1	Algoritmo de Hierarquização da Rede <i>GHENeSys</i>	67
68	5.2	Sintaxe para Fazer Referência aos Nós da Rede <i>GHENeSys</i>	68
69	5.3	Algoritmo de Execução da Rede <i>GHENeSys</i> (a)	69
70	5.4	Algoritmo de Execução da Rede <i>GHENeSys</i> (b)	70
71	5.5	Exemplo de Dinâmica da Rede <i>GHENeSys</i>	71
71	5.6	Árvore de Hierarquia para o Exemplo da figura 5.5	71
81	5.7	Resultado Exemplo da Dinâmica da Rede <i>GHENeSys</i>	81
82	5.8	Estrutura interna de um Arquivo de Projeto na <i>GHENeSys</i>	82
83	5.9	Estrutura de um Arquivo de Rede na <i>GHENeSys</i>	83
84	5.10	Janela Principal da <i>GHENeSys</i>	84
85	5.11	Ícones correspondentes aos Módulos da <i>GHENeSys</i>	85
85	5.12	Menus e Barra de Ícones do Módulo Sistema Operacional	85
86	5.13	Menus e Barra de Ícones do Módulo Desenho	86
87	5.14	Movimentação de elementos na <i>GHENeSys</i>	87

87	5.15	Mudança de Propriedades dos Elementos na <i>GHENESYS</i>
88	5.16	Menus e Barra de Ícones do Módulo Administração do Projeto
88	5.17	Janela de Opções da <i>GHENESYS</i>
89	5.18	Janela de Estrutura da <i>GHENESYS</i>
89	5.19	Janela de Gestão de mensagens da <i>GHENESYS</i>
90	5.20	Menus e Barra de Ícones do Módulo Administração do Projeto
90	5.21	Menu de Seleção das Instâncias de um Projeto <i>GHENESYS</i>
91	5.22	Menus e Barra de Ícones do Módulo Administração do Projeto
91	5.23	Janela <i>About</i> da <i>GHENESYS</i>
96	6.1	Modelos dos Sistemas de Automação do Prédio
96	6.2	Modelos dos Sistemas de Automação do Prédio com o Sistema de mensagens
99	6.3	Modelo do Objeto Presença Iluminação Interior
100	6.4	Modelo do Objeto Presença Iluminação Exterior
101	6.5	Modelo do Objeto Presença Acesso Iluminação
102	6.6	Modelo do Objeto Duplo Acesso Iluminação Presença
103	6.7	Modelo do Objeto Detecção de Incêndios
104	6.8	Modelo do Objeto Duplo Presença Iluminação Exterior
105	6.9	Modelo do Objeto Duplo Presença Iluminação Interior
106	6.10	Modelo do Objeto Economia de Energia do HVAC
107	6.11	Modelo do Objeto Sistema Acesso
108	6.12	Modelo do Objeto Sistema Vigilância
109	6.13	Modelo do Objeto Presença Iluminação Vigilância
110	6.14	Modelo do Objeto Presença Iluminação Acesso Vigilância
111	6.15	Modelo do Objeto Duplo Acesso Iluminação Presença Vigilância
112	6.16	Modelo do Objeto Duplo Presença Iluminação Interior Vigilância
113	6.17	Modelo do Objeto Acesso Vigilância
114	6.18	Encapsulamento do Objeto Presença Iluminação Exterior
115	6.19	Encapsulamento do Objeto Presença Iluminação Interior Vigilância
116	6.20	Encapsulamento do Objeto Presença Iluminação Acesso Vigilância
117	6.21	Encapsulamento do Objeto Duplo Presença Iluminação Vigilância
118	6.22	Encapsulamento do Objeto de Detecção de Incêndios
119	6.23	Encapsulamento do Objeto Duplo Presença Iluminação Exterior
120	6.24	Encapsulamento do Objeto Duplo Acesso Iluminação Presença Vigilância

6.25 Encapsulamento do Objeto Economia de Energia do HVAC . . . . .	121
6.26 Encapsulamento do Objeto Acesso . . . . .	122
6.27 Sistema de Comunicação dos Objetos Controladores do Andar Terreo124	
6.28 Sistema de Comunicação dos Objetos Controladores Subsolo 1 . . . . .	124
6.29 Arvore de Hierarquia do Estudo de Caso . . . . .	126
6.30 Nó Inicial do Modelo de Controle . . . . .	127

## Lista de Tabelas

3.1	Trabalhos na Tendência de Objetos Dentro de Redes de Petri . . .	18
3.2	Trabalhos na Tendência de Redes de Petri Dentro de Objetos . . .	19
3.3	Mútua Integração de Técnicas de Orientação a Objetos e Redes de Petri . . . . .	20
3.4	Sistemas Requeridos em um Predio Inteligente . . . . .	27
4.1	Descrição dos Elementos da Rede Ghensys . . . . .	31
4.2	Efeito do Fluxo de mensagens nos PseudoBox . . . . .	45
5.1	Caraterísticas das Ferramentas de Redes de Petri . . . . .	55
5.2	Caraterísticas dos Sistemas de Desenvolvimento Disponíveis . . . . .	65
6.1	Descrição dos Elementos de Controle . . . . .	95
6.2	Objetos Controladores do Andar Terreo . . . . .	123
6.3	Objetos Controladores do Subsolo 1 . . . . .	123
6.4	Sistema de Comunicação dos Objetos Controladores Subsolo 2 . . . . .	125
6.5	Objetos Controladores do Subsolo 2 . . . . .	125
A.1	Critérios e Escala de Avaliação . . . . .	145
A.2	Escolha do Ambiente de desenvolvimento para a Construção da Ferramenta Computacional <i>GHENESYS</i> . . . . .	146

## Lista de Símbolos

$L$	Conjunto finito de lugares.
$T$	Conjunto finito de transições.
$F$	Conjunto finito de arcos.
$W$	Função de peso.
$M_0$	Marcação inicial de uma rede de Petri.
$N$	Rede de Petri sem marcação inicial.
$(N, M_0)$	Rede de Petri com uma marcação inicial.
$G$	Grafo de uma rede de Petri.
$V$	Conjunto dos vértices do grafo de uma rede de Petri.
$F$	Conjunto de arcos orientados do grafo de uma rede de Petri.
$m$	Número de transições.
$n$	Número de lugares.
$A^+$	Matriz que indica a quantidade de <i>tokens</i> adicionadas aos lugares após o disparo de uma transição.
$A^-$	Matriz que indica a quantidade de <i>tokens</i> subtraídos dos lugares após o disparo de uma transição.
$C$	Vetor que representa a capacidade dos lugares de uma rede de Petri.
$H$	Vetor que representa as transições habilitadas para disparar em uma rede de Petri.
$U_k$	Vetor que representa a marcação dos lugares de uma rede de Petri no $k$ -ésimo instante de execução.
$X^T$	Inversa da matriz $X$ .
$H_k$	Vetor que representa as transições habilitadas de uma rede de Petri no $k$ -ésimo instante de execução.
$U_s^k$	Vetor que representa a marcação dos lugares de uma rede de Petri simulando o disparo no $k$ -ésimo instante de execução.
$U_k^c$	Vetor de marcação do conflito.
$\Sigma$	Alfabeto
$\Sigma^*$	Conjunto das palavras de um Alfabeto $\Sigma$
$\lambda$	Alfabeto vazio
$P$	Conjunto de elementos passivos da rede <i>GHNESys</i> .
$Mb$	Conjunto finito de Macro elementos do tipo passivo de uma

$P_b$	Conjunto finito de elementos do tipo Pseudobox de uma rede <i>GHENeSys</i> .
$A$	Conjunto finito de elementos ativos de uma rede <i>GHENeSys</i> .
$M_a$	Conjunto finito de Macro elementos do tipo ativo de uma rede <i>GHENeSys</i> .
$H_e$	Matriz que representa a conexão entre os elementos Pseudobox e os elementos ativos por arcos habilitadores em uma rede <i>GHENeSys</i> .
$H^d$	Matriz que representa a conexão entre os elementos Pseudobox e os elementos ativos por arcos inibidores em uma rede <i>GHENeSys</i> .
$A_g$	Matriz de incidência da rede <i>GHENeSys</i> .
$D$	Matriz Diagonal unitária.
$U'$	Vetor de Marcação dos elementos passivos não-Pseudobox
$U_{pb}$	Marcação dos Pseudobox
$H^-$	Vetor que representa o efeito dos arcos inibidores e a marcação dos Pseudobox em as transições em uma rede <i>GHENeSys</i> .
$H^+$	Vetor que representa o efeito dos arcos habilitadores e a marcação dos Pseudobox em as transições em uma rede <i>GHENeSys</i> .
$T$	Matriz diagonal unitária.
$M_r$	Vetor que representa o fluxo de mensagens.
$F_m$	fator do efeito dos mensagens na marcação dos Pseudoboxes.

# 1. INTRODUÇÃO

As Redes de Petri são uma ferramenta adequada para o modelagem de sistemas a eventos discretos [Murata, 1989], [Valette, 1979], [Cassandras, 1993]. Há um problema ainda em aberto no que concerne à geração de um modelo em Redes de Petri: a forma simples da Rede ordinária leva a redes muito grandes que só podem ser analisadas através de sua representação algébrica. Além disso, a construção de modelos de sistemas de maior complexidade requer algum tipo de modularização para romper a complexidade [Lakos, 1991].

Múltiplas tentativas de resolver esta questão, são apresentadas na literatura: em primeiro lugar há a tentativa de dobramento da rede, [Jensen, 1997] e [Genrich, 1981], que em princípio seria um paliativo para a explosão de estados, mas que também aumenta o nível de abstração com que se trata os elementos da rede. Além disso, a rede colorida hierárquica [Jensen, 1997] apresenta a possibilidade de expressar a hierarquia e o conceito de estruturação. Do mesmo modo na rede Predicado/Transição (Pr/T) clássica a abstração é introduzida pela utilização da lógica de primeira ordem associada à rede.

Mais recentemente, a discussão foi ampliada com a introdução das Redes Orientadas a Objeto, que seriam mais uma possibilidade de solução para o problema, quer seja pela introdução do conceito de objetos e seus respectivos métodos de modelagem e projeto na síntese da rede (*Objects Inside Petri Nets*), ou pela utilização do conceito de herança para ampliar o nível de encapsulamento no processo de modelagem, onde os *tokens* da rede são objetos e possuem além de atributos, métodos, eventualmente representados como subredes (*Petri Nets Inside Objects*) [Lakos, 1995].

Porém, os esforços para Integrar Redes de Petri com Orientação a Objetos têm a grande desvantagem de serem essencialmente *flat* (sem hierarquia), enquanto o desenvolvimento de sistemas de maior complexidade necessitam de uma abordagem hierárquica [Schiel, 1990], que permita que os modelos possam ser construídos de maneira estruturada, compostos a partir de unidades simples e facilmente compreensíveis em cada nível de abstração. O modelo hierárquico permite uma representação do sistema no qual as subredes que o integram



são suficientemente simples para serem modificadas de forma rápida e eficiente [Brauer, 1990].

No entanto, qualquer que seja a abordagem, o problema persiste, seja com redes de alto nível ou com redes Orientadas a Objetos (incluindo-se aí a fusão do *Petri Net inside objects* e *Object inside Petri Nets*). O ganho que se tem no processo de síntese da rede é perdido pela impossibilidade de se usar os métodos clássicos de análise de propriedades.

Assim, continua sendo muito atraente a hipótese de combinar uma Rede de Petri com um processo de síntese onde se utiliza o paradigma de objetos, o modelo hierárquico e os métodos de análise algébrica das redes já consagrados.

Neste contexto, o presente trabalho tem como tema uma proposta na direção da fusão do *Petri Net Inside Objects* e do *Object Inside Petri Nets*, mantendo a análise de propriedades baseada na representação algébrica clássica. A rede proposta é também hierárquica uma vez que não é possível dispensar a hierarquia como forma de abstração [Lakos, 1995]; esta abordagem tem ainda a vantagem de produzir componentes reutilizáveis, que reduzem em muito o trabalho de modelagem [Fabian, 1995].

Neste trabalho é apresentada a definição da rede *GHENESys* (*General Hierarchical Enhanced Net System*), sua equação de estado, os algoritmos computáveis para um jogador de marcas estritamente baseado na equação de estado, a ferramenta computacional que os implementa e uma aplicação na síntese de controle supervisorio de um sistema de automação predial. A aplicação é uma proposta de tratamento do *Intelligent building* como descrito em [Finley, 1991], com as considerações das propostas para a formalização do processo de modelagem de sistemas de controle [Silva, 1996].

Diferentes metodologias para a construção de sistemas supervisorios baseados em Redes de Petri podem ser encontradas em [Brandin, 1992], [Brandin, 1992a], [Zhou, 1995], [Giua, 1991], [Giua, 1994], [Moody, 2000], [Bourdon, 2002]. Em geral, Redes de Petri de Controle são uma classe de Redes de Petri com condições de habilitação externas chamadas *lugares de controle* que permitem a um controlador externo influenciar a progressão de *tokens* na rede [Holloway, 1994].

Neste contexto o processo de síntese apresentado neste trabalho baseia-se na heurística de uso do Prédio e aposia-se na Teoria de Sistemas Supervisórios de Ramadge and Wonham [Ramadge, 1989]. Note-se que a *GHENeSys* possui um tratamento especial para os eventos observáveis mas não-controláveis através de elementos estendidos chamados *Pseudoboxes*, que representam os elementos de conexão com o contexto e permitem inibir e/ou habilitar a ocorrência de eventos.

## 2. OBJETIVO

O objetivo deste trabalho é desenvolver um método para o projeto de sistemas procurando a integração do formalismo de Redes de Petri com o paradigma de objetos, sem no entanto perder a possibilidade de fazer análise algébrica de propriedades da maneira clássica. Assim, é apresentada uma rede estendida orientada a objetos que permite o tratamento hierárquico do processo de projeto, na rede *GHENeSys* (*General Hierarchical Enhanced Net System*), onde é possível a utilização de Redes de Petri como Objetos:

O trabalho abrange o projeto e desenvolvimento de uma ferramenta computacional para a construção e execução de modelos formais. É apresentado um estudo de caso no campo da automação predial, consistindo de um prédio com uma política de utilização definida mediante regras heurísticas. Estas regras são utilizadas na síntese de *Objetos de Controle* baseados em Redes de Petri do tipo *GHENeSys*. Para a validação do modelo e do jogador de marcas, utiliza-se uma maquete em realidade virtual do prédio modelado.

### 3. REVISÃO BIBLIOGRÁFICA

#### 3.1 Redes de Petri

As Redes de Petri são uma ferramenta que permite a representação de sistemas através de modelos na forma de estruturas matemáticas. Mediante a análise destas estruturas pode-se obter informação importante acerca do comportamento, propriedades, morfologia e dinâmica do sistema objeto de estudo [Peterson, 1981].

Uma Rede de Petri ordinária é uma estrutura matemática composta por três tipos de elementos: *lugares*, *transições* e *arcos*. Convencionalmente, os lugares simbolizam os aspectos passivos do sistema, as transições associam-se aos aspectos dinâmicos, e os arcos representam a relação entre os elementos passivos e ativos. A definição formal de uma Rede de Petri apresenta-se a seguir:

##### Definição 3.1 Rede de Petri

Uma Rede de Petri é uma 5-Tupla,  $N = (L, T, F, W, M_0)$  onde:

$L = \{l_1, l_2, \dots, l_m\}$  é um conjunto finito de lugares

$T = \{t_1, t_2, \dots, t_n\}$  é um conjunto finito de transições

$F \subseteq (L \times T) \cup (T \times L)$  é um conjunto finito de arcos (Relação de fluxo)

$W : F \rightarrow \{1, 2, 3, \dots\}$  é uma função que associa um peso a cada relação de fluxo

$M_0 : L \rightarrow \{0, 1, 2, \dots\}$  é a marcação inicial

$L \cup T = \phi \text{ e } L \cap T \neq \phi.$

A relação de peso, pode ser entendida como a quantidade de arcos unitário existentes entre dois elementos.

Ao conjunto dos *tokens* presentes nos lugares da rede num dado instante denomina-se *marcação da rede*. Uma estrutura de Rede de Petri  $N = (L, T, F)$  sem uma marcação inicial é denotada por  $N$ . Uma Rede de Petri com uma marcação inicial denota-se por  $(N, M_0)$ .

Uma Rede de Petri possui um grafo subjacente com dois tipos de nós: os lugares e as transições. Este grafo é bipartido, ponderado (*weight*) e direccionado. A definição formal do grafo inerente à Rede de Petri mostra-se a seguir.

O grafo de uma Rede de Petri  $G$  é um grafo bipartido e direcionado,  $G = (V, F)$  onde  $V = \{v_1, v_2, \dots, v_s\}$  é o conjunto dos vértices e  $F = \{f_1, f_2, \dots, f_r\}$  é um conjunto de arcos orientados,  $f_i = (v_j, v_k)$ , com  $v_j, v_k \in V$ . O conjunto  $V$  pode ser partido em dois conjuntos disjuntos  $L$  e  $T$  tal que  $V = L \cup T$ , onde  $L \cup T = \phi$ . A partição de  $V$  em dois conjuntos é a base do Princípio da dualidade [Giraud, 2003], que por sua vez é a base da formalização algébrica das Redes de Petri. O princípio de representação gráfica atribui a cada sistema um grafo composto de arcos, lugares e transições descritos por elementos gráficos.

### Definição 3.2 Grafo Subjacente de uma Rede de Petri

Na representação gráfica, as transições são representadas por barras e/ou retângulos, os lugares por círculos, os arcos são desenhados no interior dos lugares. Quando a quantidade de *tokens* não permite representá-los explicitamente, a representação é feita com um número. Os elementos gráficos da Rede são apresentados na figura 3.1

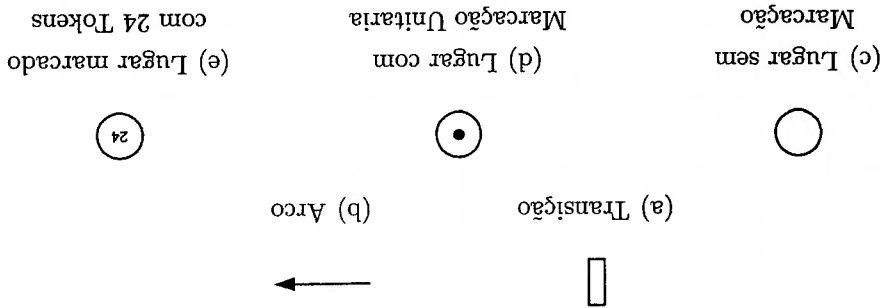


Figura 3.1: Elementos de uma Rede de Petri

A representação gráfica das Redes de Petri facilita o projeto e portanto é bem aceita para a modelagem de sistemas discretos. Entretanto, a forma simples da rede ordinária apresentada leva a redes muito grandes que só podem ser analisadas através de sua representação algébrica.

A dinâmica do sistema é representada mediante a movimentação dos *tokens*, através do disparo (*firing*) das transições. As condições de habilitação e disparo das transições são apresentadas a seguir:

1. Diz-se que uma transição está habilitada se cada lugar de entrada  $l$  está marcado com  $m(l)$  *tokens* e  $m(l) \geq w(l, t)$  onde  $w(l, t)$  é a função peso.

Este princípio propõe uma representação matemática alternativa, baseada em matrizes construídas a partir das relações existentes entre os lugares e as transições. A definição baseada em matrizes é equivalente à forma padrão [Peterson, 1981]. O estudo das propriedades, dinâmica e características do sistema modelado por uma Rede de Petri, baseia-se neste princípio. A representação algébrica supõe uma rede pura, isto é, sem arestas e cujos extremos sejam o mesmo elemento.

### 3.1.1 Princípios de Representação Algébrica

Uma característica das Redes de Petri é sua natureza assíncrona, pois sua execução não depende de parâmetros de tempo. Assim, a única relação entre os eventos a modelar é sua ordem na sequência das ocorrências.

A modelagem com Redes de Petri tem como característica inerente o paralelismo e a concorrência, isto é, em uma Rede de Petri, dois eventos que estão habilitados e não interagem entre si podem ocorrer independentemente, sem que seja necessário sincronizá-los, a menos que esta seja uma característica própria do sistema que se deseja modelar.

Uma transição que não tem lugares de entrada é denominada como transição fonte (*source*) e sempre está habilitada. Uma transição que não tem lugares de saída é denominada sorvedouro (*sink*), e quando disparar remove os *tokens* dos lugares de entrada apagando-os da rede.

Uma Rede de Petri na qual a capacidade dos lugares é limitada se conhece como rede finita, o disparo de uma transição em uma rede finita requer além das condições antes apresentadas, que os lugares de saída tenham uma capacidade livre que permita a alocação dos *tokens*. Esta limitação se conhece como condição de disparo estrita.

2. Uma transição habilitada pode ou não disparar dependendo se o evento que representa ocorre ou não. Uma transição não habilitada não pode disparar.
3. O disparo de uma transição  $t$  remove  $w(l, t)$  *tokens* de cada lugar de entrada  $l$  e adiciona  $w(t, l)$  *tokens* a cada lugar de saída  $l$  de  $t$ .

### Definição 3.3 Representação Algébrica de uma Rede de Petri

Uma Rede de Petri é representada algebricamente por uma 4-tupla:  $(L, T, A^+, A^-)$

onde:

$L$  é um conjunto finito de lugares

$T$  é um conjunto finito de transições

$m$  é o número das transições

$n$  é o número dos lugares

A matriz  $A^+$  de ordem  $m \times n$ , representa a quantidade de *tokens* adicionada aos lugares após o disparo de uma transição:

$$(3.1) \quad A^+ = \begin{matrix} & \begin{matrix} T_1 & T_2 & \dots & T_m \end{matrix} \\ \begin{matrix} L_1 \\ L_2 \\ \dots \\ L_n \end{matrix} & \begin{bmatrix} a_{1,1}^+ & a_{1,2}^+ & \dots & a_{1,m}^+ \\ a_{2,1}^+ & a_{2,2}^+ & \dots & a_{2,m}^+ \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1}^+ & a_{n,2}^+ & \dots & a_{n,m}^+ \end{bmatrix} \end{matrix} \quad \text{onde: } a_{ij}^+ = \begin{cases} w(t_i, l_j) & \text{se existe um arco entre } t_i \text{ e } l_j \\ 0 & \text{caso contrário} \end{cases}$$

A matriz  $A^-$ , de ordem  $m \times n$ , representa a quantidade de *tokens* subtraída dos lugares, após o disparo de uma transição:

$$(3.2) \quad A^- = \begin{matrix} & \begin{matrix} T_1 & T_2 & \dots & T_m \end{matrix} \\ \begin{matrix} L_1 \\ L_2 \\ \dots \\ L_n \end{matrix} & \begin{bmatrix} a_{1,1}^- & a_{1,2}^- & \dots & a_{1,m}^- \\ a_{2,1}^- & a_{2,2}^- & \dots & a_{2,m}^- \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1}^- & a_{n,2}^- & \dots & a_{n,m}^- \end{bmatrix} \end{matrix} \quad \text{onde: } a_{ij}^- = \begin{cases} w(l_j, t_i) & \text{se existe um arco entre } l_j \text{ e } t_i \\ 0 & \text{caso contrário} \end{cases}$$

#### 3.1.1.1 Vetor de Capacidades

A capacidade é o número máximo de *tokens* que um lugar pode armazenar. O vetor de Capacidades  $C$ , de ordem  $n$ , representa a capacidade dos lugares da rede, sendo representado por:

No caso das Redes de Petri finitas, no qual a capacidade dos lugares é finita, é preciso estabelecer se os elementos de saída podem receber os *tokens* que o disparo das transições irão aportar. Assim:

$$\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \geq \begin{pmatrix} [e_1, e_2, \dots, e_m] \end{pmatrix} \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{bmatrix} = \begin{bmatrix} a_{j,1} \\ a_{j,2} \\ \vdots \\ a_{j,n} \end{bmatrix}^T$$

Na forma estendida:

Onde:  $U$  é o vetor de marcação e  $e[j]$  representa um vetor de ordem  $m$  que é zero em todas as entradas exceto na componente  $j$ , Assim é possível comparar a  $j$ -ésima entrada da matriz  $A^-$  com o vetor da marcação  $U$ , e determinar se cada lugar tem o número de *tokens* a serem removidos no disparo da  $j$ -ésima transição.

$$U \geq (e[j] \cdot A^-)^T \quad (3.5)$$

Utilizando o enfoque matricial para determinar se uma transição está habilitada por uma marcação aplica-se a seguinte equação:

### 3.1.1.3. Transições Habilitadas

(3.4)

$$U = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \quad \text{onde: } u_j = \text{Quantidade de } \textit{tokens} \text{ presentes no } j\text{-ésimo lugar}$$

É um vetor  $U_k$  de ordem  $n$ , que representa a distribuição dos *tokens* nos lugares no estado  $k$ . Na forma estendida:

### 3.1.1.2. Vetor de Marcação

$$C = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \quad \text{onde: } c_j = \text{Capacidade do } j\text{-ésimo lugar} \quad (3.3)$$



$$\begin{bmatrix} u_n \\ \vdots \\ u_2 \\ u_1 \end{bmatrix}^{k+1} = \begin{bmatrix} u_n \\ \vdots \\ u_2 \\ u_1 \end{bmatrix}^k + \begin{bmatrix} a_{n,1} & a_{n,2} & \dots & a_{n,m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{2,1} & a_{2,2} & \dots & a_{2,m} \\ a_{1,1} & a_{1,2} & \dots & a_{1,m} \end{bmatrix} \begin{bmatrix} f_m \\ \vdots \\ f_2 \\ f_1 \end{bmatrix}^k$$

A forma explícita da equação 3.8 é mostrada a seguir:

$$F^k \leq H^k \quad (3.9)$$

efetivamente, assim:

a possibilidade de que nem todas as transições habilitadas para disparar, disparem  
 Onde  $F$  é o vetor que representa as transições disparadas. É preciso considerar

$$U^{k+1} = U^k + A^T \cdot F^k \quad (3.8)$$

a equação de estado:

As consequências de um disparo das transições da rede são calculadas mediante

### 3.1.1.5 Equação de Estado

$$\begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_m \end{bmatrix}^k \quad \text{onde: } h_i = \begin{cases} 1 & \text{se verificarem as equações 3.5 e 3.6} \\ 0 & \text{caso contrário} \end{cases} \quad (3.7)$$

habilitação de cada uma das transições da rede

O vetor de habilitação  $H^k$  de ordem  $m$ , é formado pelo resultado da avaliação da

### 3.1.1.4 Vetor de Habilitação

$$\begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} - \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \geq [e_1, e_2, \dots, e_m] \begin{bmatrix} a_{1,1}^+ & a_{1,2}^+ & \dots & a_{1,n}^+ \\ a_{2,1}^+ & a_{2,2}^+ & \dots & a_{2,n}^+ \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1}^+ & a_{m,2}^+ & \dots & a_{m,n}^+ \end{bmatrix} = \begin{bmatrix} a_{j,1}^+ \\ \vdots \\ a_{j,2}^+ \\ a_{j,n}^+ \end{bmatrix}^T$$

Na forma estendida:

$$C - U \geq (e[j] \cdot A^+)^T \quad (3.6)$$

### 3.1.1.6 Detecção de Conflitos

Os conflitos aparecem quando se deseja disparar simultaneamente várias transições, nas quais o disparo de uma desabilita as outras (conflito *preset*), ou quando o disparo de várias transições gera uma quantidade de *tokens* que excede a capacidade de um lugar (Conflito *Postset*). A figura 3.2 ilustra graficamente este conceito.

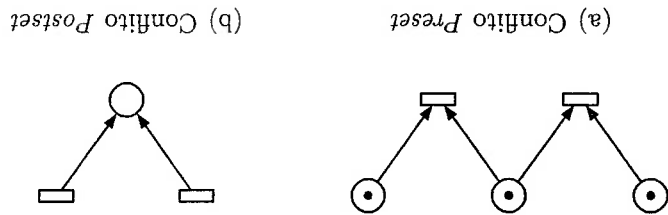


Figura 3.2: Conflito em Redes de Petri

Para detectar os possíveis conflitos calcula-se o vetor da marcação obtido a partir da simulação do disparo de todas as transições habilitadas para disparar (representado por  $U_s^{k+1}$ ), segundo a equação:

$$(3.10) \quad U_s^{k+1} = U^k + A^T \cdot (H^k)$$

Onde:

$A$  é a matriz de incidência construída a partir das duas matrizes definidas em

3.3.

$$(3.11) \quad A = \begin{matrix} & \begin{matrix} T_1 & T_2 & \dots & T_m \\ \begin{matrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{matrix} \end{matrix} \\ \text{onde: } a_{i,j} = a_{i,j}^+ - a_{i,j}^- \end{matrix}$$

$H^k$  é o vetor de Habilitação no  $k$ -ésimo vetor de disparo.

A partir do vetor de simulação de disparo  $U_s^k$  e do vetor de capacidades  $C$  calcula-se o vetor de marcação de conflitos  $U^k$  segundo a seguinte equação:

$$(3.12) \quad U^k = \begin{bmatrix} u_1^k \\ u_2^k \\ \vdots \\ u_n^k \end{bmatrix} \quad \text{onde: } u_j^k = \begin{cases} 0 & \text{caso contrário} \\ 1 & \text{se } (u_s^j)^k < c_j \\ -1 & \text{se } (u_s^j)^k > 0 \end{cases}$$

Uma entrada diferente de zero na  $j$ -ésima posição do vetor de marcação do conflito implica que na coluna correspondente da matriz  $A$  encontram-se presentes transições que apresentam um conflito. Se a entrada é negativa ( $-1$ ) o conflito que se apresenta é de *preset*, se a entrada é positiva ( $1$ ) o conflito é de *postset*.

Para a resolução de conflitos, aplica-se uma política de resolução específica para o sistema modelado, e que depende da interpretação deste modelo. Na ausência de conflitos, seja porque não foram detectados ou porque foram resolvidos pela aplicação de uma política de resolução, é possível avaliar as consequências de um disparo real, através da equação de estado (ver 3.8).

### 3.1.2 Dinâmica da Rede de Petri Ordinária

A dinâmica da Rede representa o comportamento dos aspectos de interesse do sistema modelado. O algoritmo da Rede de Petri Ordinária é apresentado na figura 3.3.

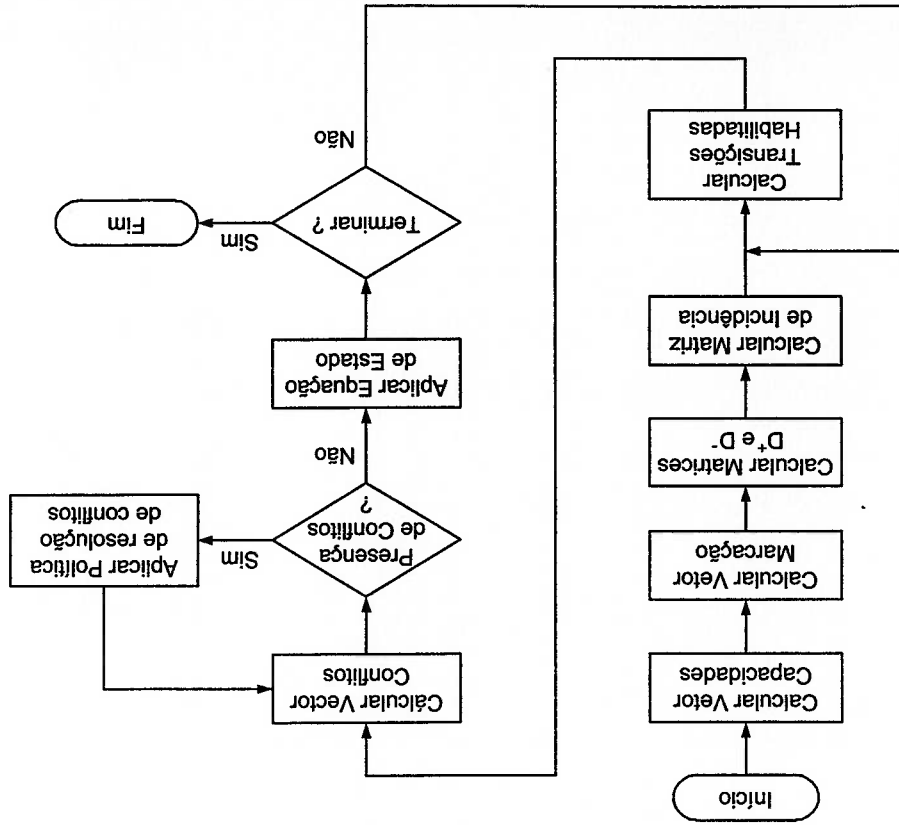


Figura 3.3: Algoritmo da Dinâmica de uma Rede de Petri Ordinária

### 3.1.3 Redes Estendidas e de Alto Nível

Com o objetivo de incrementar o poder de representação dos modelos construídos com Redes de Petri, várias abordagens têm sido desenvolvidas introduzindo novos elementos e técnicas na teoria clássica das Redes de Petri. Estas redes são denominadas Redes de Petri estendidas, as mais importantes são: Redes com Arcos Habilitadores e Inibidores, Redes com Tempo e Redes Estocásticas

Além disso há técnicas nas quais são associados aos *tokens* valores que são modificados durante a execução da rede, este tipo de Rede é conhecida como Redes de Alto Nível e incluem as Redes Coloridas, Predicado/Transição e Redes com *tokens* Individuais.

### 3.1.4 Linguagens

Seja  $\Sigma$  um conjunto de símbolos chamado alfabeto. Uma palavra (*string*) é uma cadeia finita de símbolos do alfabeto  $\Sigma$ . Uma linguagem é um conjunto de palavras e frases bem formadas sobre um alfabeto, definidas por uma gramática. Um alfabeto vazio é denotado por  $\lambda$  e o conjunto das palavras por  $\Sigma^*$ .

Uma Rede de Petri é geradora de uma linguagem [Giua, 1991]. O conjunto de estados de marcação que uma rede  $N$ , pode alcançar desde um estado de marcação inicial  $M_0$ , através do disparo das transições da rede é denominado o conjunto de alcançabilidade de  $M$  e denota-se como  $R(N, M_0)$ . O conjunto de alcançabilidade da rede é equivalente à linguagem gerada pela rede, onde ambos os conceitos se relacionam mediante uma Rede de Petri Rotulada [Giua, 1994].

### Definição 3.4 Rede de Petri Rotulada

Uma Rede de Petri Rotulada é uma 4-Tupla,  $G = (N, l, M_0, F)$  onde:

$N$  = Uma estrutura de Rede de Petri.

$l : T \rightarrow \Sigma$  é uma função que associa a cada transição um rótulo a partir de um

Alfabeto de Eventos  $\Sigma$ .

$M_0$  = Marcação inicial.

$F$  é um conjunto finito de Marcações finais.

A função de rotulação pode ser estendida de maneira que  $T^* \rightarrow \Sigma^*$

Dois Redes de Petri são equivalentes se suas linguagens são iguais, e assim se estabelece a base formal para a equivalência de dois sistemas [Peterson, 1981]. Se uma transformação aplicada a uma rede, produz uma nova rede com a mesma linguagem diz-se que a linguagem é preservada a partir desta transformação.

A representação das Redes de Petri a partir de linguagens permite a otimização dos modelos dos sistemas mediante a aplicação de transformações que preservem a linguagem da rede, procurando reduzir o número de elementos utilizados. Adicionalmente esta representação é útil para técnicas de análise de propriedades e para a especificação da síntese automática da rede a partir da especificação do comportamento desejado do sistema como uma linguagem.

## 3.2 Orientação a Objetos

A Orientação a Objetos (OO) é uma técnica originária da Engenharia de Software, a qual consiste em modelar um sistema decompondo-o em entidades concretas denominadas objetos. O comportamento do sistema é reproduzido pela interação dos objetos assim definidos. O processo de identificação dos objetos de um sistema conhece-se como decomposição [Booch, 1994]. Os objetos são definidos em termos da informação que contém e das operações que disponibilizam para a utilização e manipulação desta informação.

Um objeto é definido como uma estrutura que possui dados, estados e métodos. Os dados são a especificação da informação que caracteriza o objeto, os métodos são a forma de manipular os dados, permitindo alterar o estado do objeto. Os métodos podem ser de três tipos:

- **Métodos de Construção/Destruição:** Permitem que um objeto passe a ser parte do sistema o deixe de sê-lo.
- **Métodos com acesso a Escrita:** Modificam o valor dos dados do Objeto.
- **Métodos com acesso a Leitura:** Permitem obter os valores dos dados do objeto, sem alterá-los.

### 3.2.1 Conceitos Básicos

Os conceitos mais relevantes dos métodos com Orientação a Objetos são descritos a seguir [Jaworski, 1997]:

- **Mensagens.** Os objetos interagem trocando mensagens: quando um objeto recebe uma mensagem, invoca um método para processá-la; este método pode mudar o estado do objeto, retornar a informação nele contida, ou provocar a criação ou eliminação de objetos.
- **Classes.** Uma classe é um conjunto de objetos que compartilham uma estrutura e um comportamento comum, quando um objeto pode ser considerado como um elemento de uma classe, diz-se que este objeto é uma instância da classe. Assim, uma classe define o tipo de dados contidos em um objeto e os métodos que estes utilizam para acessar os dados, e os métodos para criar e/ou destruir instâncias de se mesmo.
- **Classificação:** A classificação é um mapeamento entre objetos e classes conhecidas. Se um objeto não se encaixa em nenhuma das categorias conhecidas, então gera-se uma nova classe.
- **Reutilização:** Consiste em utilizar instâncias de uma classe fazendo referência a os seus métodos de criação. A reutilização de classes proporciona a possibilidade de construir entidades que podem ser depois utilizadas em outros projetos com o consequente incremento na produtividade no projeto de sistemas.
- **Herança:** A propriedade de que uma categoria de objeto (Classe) contém as propriedades de todas as categorias de objetos dos quais este é descendente na árvore de classificação das classes denomina-se herança. A árvore de classificação das classes possui uma organização hierárquica onde os níveis superiores contêm as classes mais simples, isto é, as que têm um número menor de atributos, dados e métodos, descendo até os objetos mais complexos.
- **Composição:** É a propriedade de um objeto poder estar composto de outros objetos ou construído com base nestes, permitindo a construção de objetos complexos a partir de objetos simples.
- **Encapsulamento:** O encapsulamento é a combinação de dados e procedimentos que manipulam os dados em um único componente, isto é um

objeto. O Encapsulamento também faz referência ao controle de acesso aos detalhes da implementação de um objeto. O acesso a um objeto fica limitado a uma interface controlada e definida especificamente, o que permite que os objetos sejam autônomos.

- **Polimorfismo:** É a capacidade de um objeto possuir vários métodos com o mesmo nome porém com formas diferentes, isto é com um número e/ou um tipo de parâmetros diferentes. Tal característica é especialmente útil na Engenharia de Software.

### 3.2.2 Análise Orientada a Objetos

O termo Análise Orientada a Objetos (*AOO*) refere-se a qualquer método que utilize a decomposição para a construção de modelos do mundo real, do ponto de vista da Orientação a Objetos. O Projeto Orientado a Objetos (*POO*) é um método de projeto que utiliza o processo de Análise Orientado a Objetos e algum tipo de notação para representar os modelos lógico, físico, dinâmico e estático do sistema. [Booch, 1994]

As técnicas Orientadas a Objetos já provaram ser extremamente efetivas no modelagem e simulação em geral [Pilgrim, 2001]. Especificamente, as técnicas de modelagem orientadas a Objetos estão sendo amplamente utilizadas na descrição de diferentes tipos de sistemas em diferentes áreas da engenharia, graças ao fato de possuírem importantes características como reutilização de código/classes, independência de implementação e uma grande capacidade de expressão de modelos [Wong, 1995].

## 3.3 Orientação a Objetos e Redes de Petri

Pesquisas recentes mostram que a integração do formalismo de Redes de Petri com técnicas de Orientação a Objetos tem características complementares. [Zapf, 1999]. Da integração destes dois conceitos é possível identificar três tendências [Bastide, 1995]:

- Objetos Dentro de Redes de Petri
- Redes de Petri Dentro de Objetos

- Mútua Integração das Técnicas de Orientação a Objetos e Redes de Petri

### 3.3.1 Objetos Dentro de Redes de Petri

No enfoque de Objetos Dentro de Redes de Petri (*Objects Inside Petri Nets*) o sistema é modelado por uma Rede de Petri na qual os objetos são representados por *tokens* na rede, as transições representam métodos com capacidade de ler e mudar os atributos dos objetos, não existe intercâmbio de mensagens entre os objetos. Assim, a rede é um framework para o fluxo de objetos, o que é conceitualmente semelhante a ter uma rede colorida, já que o mapeamento das marcas em objetos é o mesmo que categorizá-las identificando a sua "cor". A figura 3.4 ilustra graficamente este conceito, a tabela 3.1 apresenta alguns dos trabalhos realizados neste sentido [Zapf, 1999].

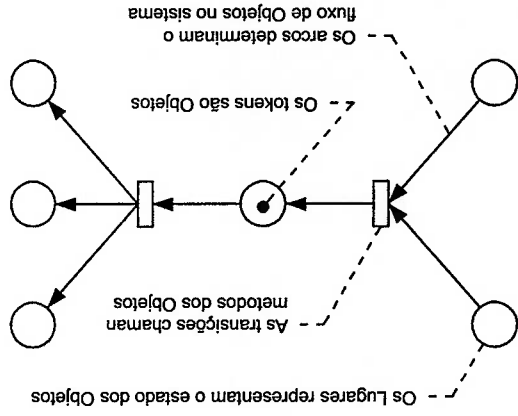


Figura 3.4: Conceito de Objetos Dentro de Redes de Petri

### 3.3.2 Redes de Petri Dentro de Objetos

Na tendência de Redes de Petri Dentro de Objetos (*Petri Nets Inside Objects*) realiza-se primeiro o processo de decomposição do sistema (ver 3.2). Uma vez que os elementos relevantes e suas relações foram estabelecidas, o comportamento de cada objeto é modelado com ajuda das Redes de Petri. A figura 3.5 ilustra graficamente este conceito, e a tabela 3.2 apresenta alguns dos trabalhos realizados nesta tendência [Zapf, 1999].



A integração mútua das técnicas orientadas a Objetos e Redes de Petri tende a se concentrar no desenvolvimento de sistemas que integram de maneira recursiva as Redes de Petri e os objetos. A AOO é utilizada para determinar os objetos e as regras de composição e associação entre estes. O comportamento de cada objeto as relações e o fluxo de mensagens entre objetos são representadas em Redes de Petri ainda que sejam necessárias extensões para abrigar especificamente o conceito de mensagem ou ampliar o espectro de relações entre objetos para incluir relações de vizinhança e ordenação.

### 3.3.3 Mútua Integração das Técnicas de Orientação a Objetos e Redes de Petri

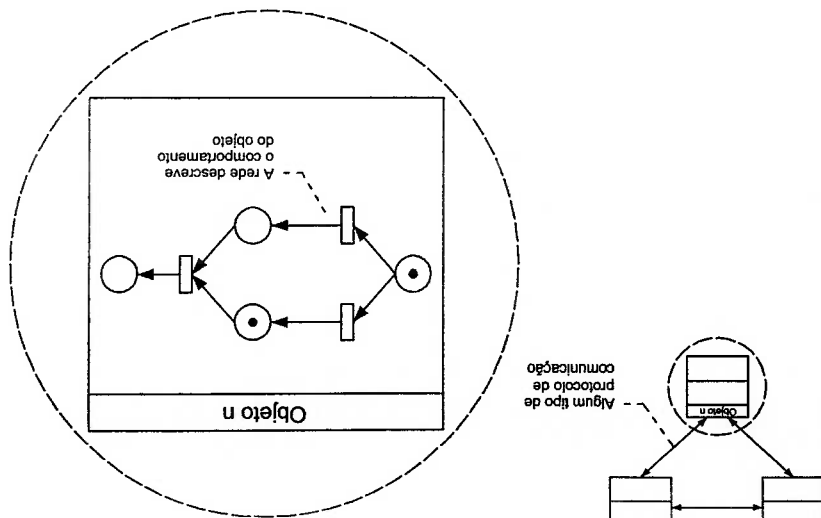
Técnica	OO Notação- Linguagem	Tipo da Rede	Ferramenta
LOOPN (Language for Objects Petri Nets)	Linguagem Própria	Redes Coloridas com Tempo	LOOPN
Macronet)	Extensão da re- apresentação ER	Rede Estendida	MacroTec
MOBY	SmallTalk-80	Rede Estendida	MOBY
NetCASE (Petri Net Based CASE)	Baseado em OMT	Subclasse das redes Pr/T	NEPTUM
OBJSA Nets	OBJ2	Rede Estendida	ONE
OOCPN (Object-Oriented Coloured Petri Nets)	EIFFEL	Redes Coloridas	
OPM (Objects Process Model)	UML	Transformação em Coloridas ou PT.	OTW 2
SimCon (Simple Integrated Model to Complex Object Networks)	Notação Própria	Rede Estendida	ExSpect
THORN (Timed hierarchical Object-Related Nets)	C++	Rede Estendida	THORN/DE

Tabela 3.1: Trabalhos na Tendência de Objetos Dentro de Redes de Petri

técnica	OO Notação- Linguagem	Tipo da Rede	Ferramenta
CLOWN (Class Orientation with Nets)	Linguagem Própria	Rede Estendida	Ambiente ONE
HOOD Nets	Notação Própria, High Level, ADA-pseudo-code	Pr/T Nets	
OBM (Object Behaviour Model)	Notação Própria baseada em Modelo semântico de dados	Transformação em Pr/T	Transformação editor
OCPN (Object Coloured Petri Nets)	Notação Orientada a Objetos	Transformação em Coloridas	
OOBM (Object-Oriented Behaviour Modelling for Real-Time Design)	Notação Orientada a Objetos	Redes Coloridas	
PAM (Petri Net Based Abstract Machine)	Linguagem própria	Redes Coloridas	
PROTOB	Notação e Linguagem própria	Linguagem própria	PROTOB

Tabela 3.2: Trabalhos na Tendência de Redes de Petri Dentro de Objetos

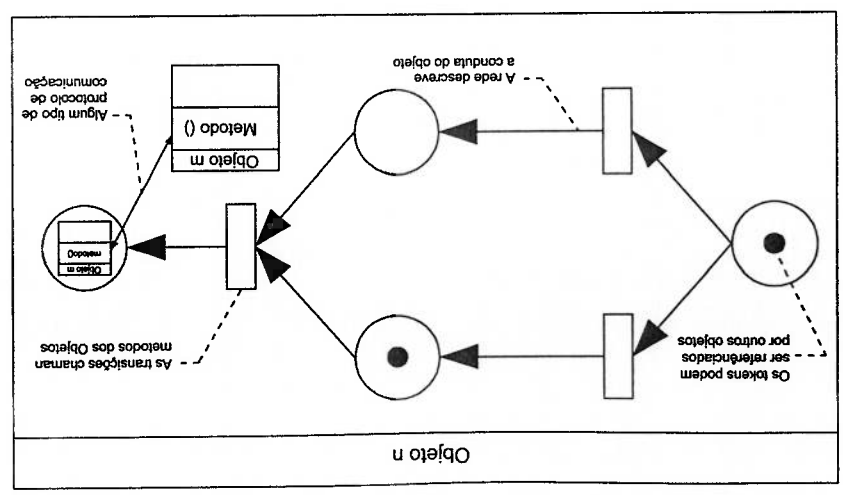
Figura 3.5: Conceito de Redes de Petri Dentro de Objetos



Técnica	OO Notação- Linguagem	Tipo da Rede	Ferramenta
CO-OPN/2 (Concurrent Object-Oriented Petri Nets)	Linguagem Própria	Rede Estendida	Coopn-Tools
COOs (Cooperative Objects)	Linguagem Própria	Rede Estendida	SYROCO
HOON Higher-Order Object Nets	OMT e ER-Notation	Rede Estendida	
LOOPN++ (Language for Object Petri Nets)	Linguagem Própria	Transformação em Coloridas	LOOPN++
OOPN (Object Oriented Petri Nets)	Notação Própria	Rede Estendida	PNTalk
OOPNL (Object Oriented Petri Net Language)	Notação Própria	Rede Estendida	Codesign
OOPr/T (Nets Object oriented Pr/T Nets)	OMT ou UML	Pr/T	
OPN (Object Petri Nets)		Rede Estendida	
PN-TOX (Petri Nets Tools for Object Concurrency Specification)	Notação Própria	Diferentes tipos de Redes	

Tabela 3.3: Mútua Integração de Técnicas de Orientação a Objetos e Redes de Petri

Figura 3.6: Mútua Integração das Técnicas de OO e Redes de Petri



Assim, as Redes de Petri são usadas para modelar o comportamento dos objetos, e portanto não é possível ter uma visão global da interação entre os objetos e da sua troca de mensagens, porém a integração das técnicas de Orientação a Objetos e Redes de Petri facilita a modelagem hierárquica com comportamento concorrente [Zapf, 1999]. A figura 3.6 ilustra graficamente este conceito, a tabela 3.3 mostra alguns trabalhos que utilizam este enfoque.

Em geral, nas propostas apresentadas na tabela 3.3 a formulação analítica das redes e a possibilidade de análise de propriedades é substituída por um conjunto de comportamentos atribuídos a objetos, onde as marcas desempenham um papel fundamental dando ao sistema o status de alto nível.

Porém, algumas das propostas estão baseadas no modelo Redes de Petri dentro de Objetos, incluindo a capacidade de um objeto referenciar outros objetos [Lakos, 1995]. O sistema desta maneira construído é composto por um conjunto de instâncias de classes. Não há portanto como sintetizar as propriedades globais do sistema, por conseguinte não é possível representar agentes integrados de controle.

Os esforços para integrar Redes de Petri com Orientação a Objetos têm a grande vantagem de serem essencialmente *flat* enquanto que o desenvolvimento de sistemas de maior complexidade necessitam de um abordagem ser hierárquico [Schiel, 1990].

### 3.4 Hierarquia em Redes de Petri

A construção de modelos de sistemas de maior complexidade requer algum tipo de modularização para romper a complexidade [Lakos, 1991]. Para a representação eficiente deste tipo de sistema requer-se um mecanismo de composição hierárquica que permita que os modelos possam ser construídos de maneira estruturada, composto a partir de unidades simples e facilmente compreensíveis em cada nível de abstração. A hierarquia é uma ferramenta valiosa que permite que o projeto de um sistema seja construído de maneira estruturada, composto por uma quantidade de entidades totalmente inteligíveis [Cortés, 2001].

### 3.4.1 Composição Hierárquica

O método para a elaboração de um modelo onde o resultado final é a combinação das soluções de pequenos sub-modelos é conhecido como composição hierárquica [Malhotra, 1993]. Vários métodos de composição hierárquica são propostos: [Valette, 1979] define o conceito de *block* que é uma subrede que inicia e termina com uma transição, [Zuzuki, 1983] propõe um método para refinamento e abstração que permite manejar o problema da explosão de estados mediante a transformação de Transições e Lugares em subredes. Transformações realizadas em redes complexas são, em geral, difíceis de provar. O modelo hierárquico permite uma representação do sistema no qual as subredes que o integram são suficientemente simples para serem modificadas de forma rápida e eficiente [Brauer, 1990].

### 3.4.2 Refinamento

A técnica de refinamento consiste em substituir uma transição ou um lugar por uma rede. Um conceito relacionado com o refinamento é a redução que consiste em substituir um subconjunto de elementos por um lugar ou uma transição. [Murata, 1989] apresenta um conjunto de regras de redução que preservam propriedades como vivacidade (*Liveness*), Segurança (*Safeness*) e Limitação (*Boundedness*). A figura 3.7 ilustra estas transformações.

### 3.4.3 Refinamento Hierárquico

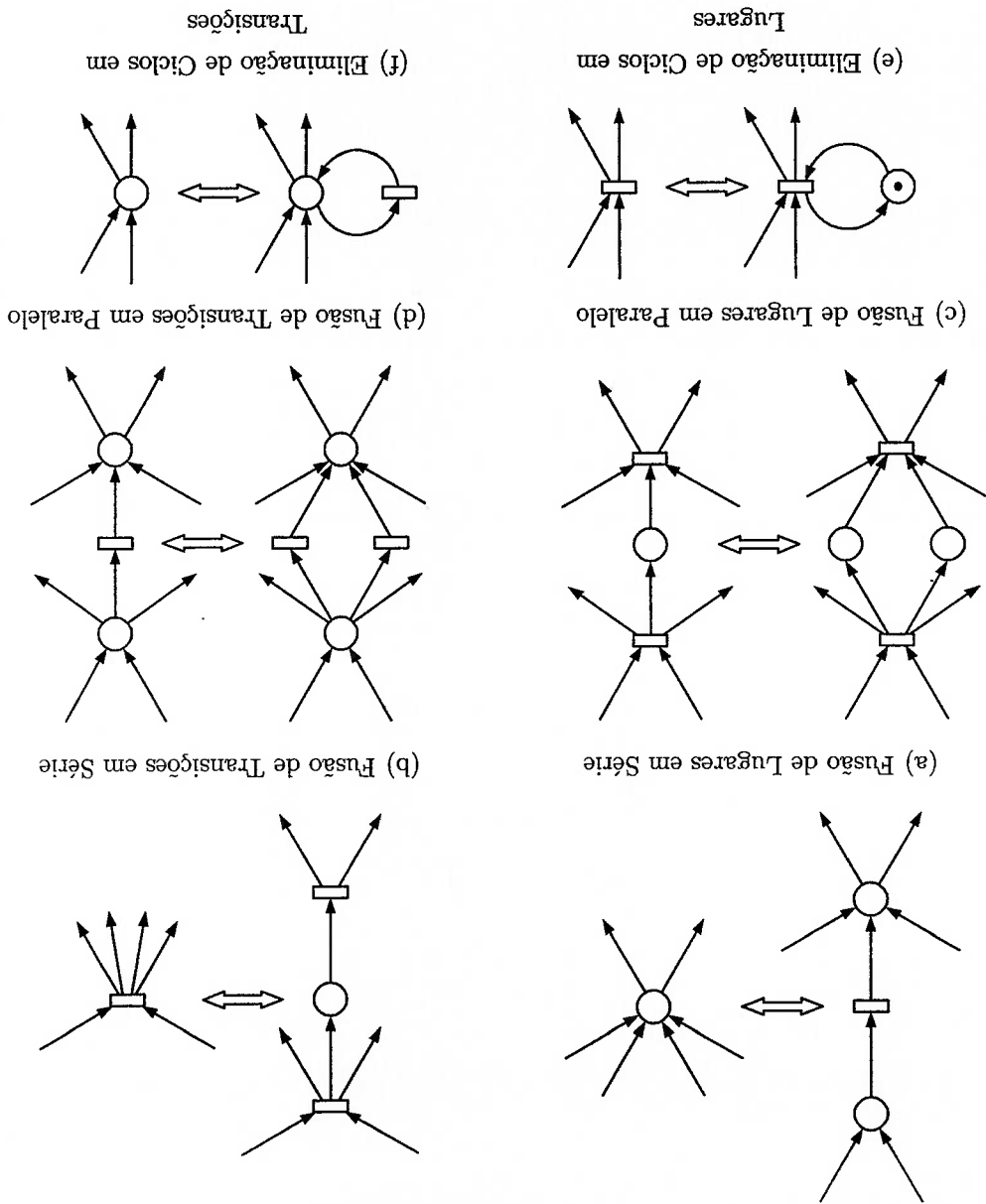
A vantagem da hierarquia associada com refinamento é a possibilidade de poder modelar um sistema em vários níveis de abstração, conforme o estágio de desenvolvimento e o conhecimento que se tem do sistema. Assim, a descrição da rede pode ser representada como um conjunto hierarquicamente organizado de redes [Anisimov, 1994]. Acrescida a esta vantagem, estaria ainda a possibilidade de associar a modelagem conceitual à reutilização de módulos já existentes. Um refinamento ou uma redução são conservativos com relação a vivacidade, segurança e limitação, se mantem a conduta da Rede, isto é, se preserva o comportamento da rede. [Murata, 1989].

Um Sistema de Eventos Discretos (*SED*) é um sistema dinâmico que evolui de entrada correta para que efetuará a conduta desejada. [Cassandras, 1993]. Para que um sistema realize de forma adequada as funções para as quais foi desenvolvido, é preciso que seja controlado, selecionando um conjunto de valores

### 3.5.1 Controle de Sistemas de Eventos Discretos

## 3.5 Controle de Sistemas a Eventos Discretos com Redes de Petri

Figura 3.7: Transformações Preservando Vivacidade, Segurança, e Limitação



acordo com eventos que acontecem em períodos irregulares de tempo. As Redes de Petri são uma ferramenta adequada para o modelagem de sistemas a eventos discretos [Murata, 1989], [Valette, 1979], [Cassandras, 1993].

O Controle de SED baseia-se na capacidade de habilitar ou desabilitar alguns dos eventos do sistema. Assim, o conjunto de eventos  $E$  fica dividido em um subconjunto de eventos controláveis  $E_c$  e incontroleáveis  $E_u$ , isto é:  $E = E_c \cup E_u$  [Ramadge, 1989]. Um sistema de controle deste tipo é conhecido como sistema supervisorio [Cassandras, 1993]. A fig. 3.8 ilustra graficamente o conceito de controle de sistemas a eventos discretos.

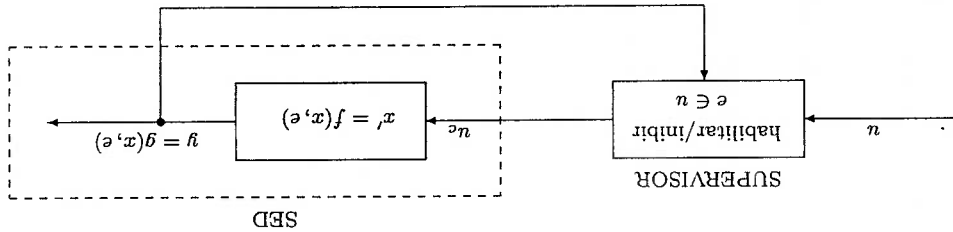


Figura 3.8: Controle Supervisorio de Sistemas a Eventos Discretos

Diferentes metodologias para a construção de sistemas supervisorios baseados em Redes de Petri podem ser encontradas em [Brandin, 1992], [Brandin, 1992a], [Zhou, 1995], [Gina, 1991], [Gina, 1994], [Moody, 2000], [Bourdon, 2002]. Em geral, Redes de Petri de Controle são uma classe de Redes de Petri com condições de habilitação externas chamadas *lugares de controle* que permitem a um controlador externo influenciar a dinâmica da rede [Holloway, 1994].

### 3.6 Prédios Inteligentes

#### 3.6.1 Inteligência nos Prédios

A inteligência dos prédios consiste em sua capacidade para responder às necessidades de seus usuários, provendo-lhes (ao indivíduo e à organização) serviços administrados de forma coordenada para maximizar a ação técnica, minimizar os custos de investimento e a manutenção. Neste caso, a inteligência reside no fato dos sistemas contarem com características para satisfazer as necessidades dos usuários de forma automática, aplicando os critérios que um ser humano aplicaria

na mesma situação. Para chamar a um Prédio de 'inteligente', ele deve reunir as seguintes características: [Fujie, 1991]

- Possuir um projeto que permita prover, todos os serviços de um modo adequado, em relação às exigências de usuários e administradores, permitindo flexibilidade no uso das dependências.

- Contar com serviços integrados e controlados.

- Contar com sistemas históricos de operações.

- Ter um plano que preveja todos os serviços de um modo apropriado, em relação às exigências de usuários e administradores, e que permita flexibilidade no uso das dependências.

"A inteligência de um prédio inteligente reside no grau em que são liberados os administradores e usuários das atividades tediosas e assistidos nas tarefas que deveriam executar. Um prédio altamente automatizado não pode ser considerado inteligente, a menos que esta automatização ajude criar uma atmosfera que promova a criatividade, produtividade, excitação intelectual, troca de informação, e as saúde mental e física. Não pode ser denominado inteligente um Prédio que é projetado ergonomicamente, mas que não possui sistemas de telecomunicações, necessários no mundo atual. Também não é suficiente uma arquitetura brilhante, engenhosa, para considerar-se um Prédio como sendo inteligente." [Finley, 1991]

O nível de Inteligência de um prédio também é medido de forma efetiva pelo nível de conforto do ambiente. As características de conforto são: segurança, eficiência no trabalho, economia de energia, minimização do impacto ambiental, prontidão de canais de comunicação e sistemas de informação. [Stipidis, 1998]

Deste modo o projeto de um Prédio inteligente requer o trabalho de peritos em áreas diversas, como, computação e telecomunicações, construção, projeto de interiores e ainda ecologia [Akimaru, 1991].



### 3.6.2 Diferença entre um Prédio Inteligente e um Prédio

#### Automatizado

##### 3.6.2.1 Prédio Automatizado

Um prédio automatizado, é aquele que possui sistemas e serviços que respondem de um modo automático às necessidades de seus usuários e da administração, com a característica de que cada sistema atua de forma descentralizada. Por exemplo:

- Sistemas de segurança (alarmes, extintores de incêndio, etc.).
- Sistemas de corrente ininterrompida.
- Ar condicionado por zonas.
- Manutenção automatizada.

##### 3.6.2.2 Prédio Inteligente

Um prédio inteligente é aquele que, além de ser automatizado, incorpora técnicas da tecnologia da Informação. Aliás também deve incluir inevitavelmente aspectos de integração, flexibilidade, planejamento do espaço, otimização de recursos e telecomunicações. [Cerdeira, 1989].

### 3.6.3 Sistemas do Prédio Inteligente

Os sistemas do Prédio são os que permitem alcançar uma atmosfera ergonômica para os usuários e equipes. Os sistemas principais de um Prédio são:

- Sistemas de calefação, ventilação y ar condicionado, HVAC (*Heating Ven-tilation Air-Conditioning*).
- Iluminação.
- Energia elétrica.
- Fiação estruturada.
- Elevadores e escadas rolantes.
- Controle de acesso.
- Segurança.

Um sistema de automação predial deve contemplar uma grande quantidade de tarefas, as principais segun [Stipidis, 1998] são apresentadas na tabela 3.4

- Telecomunicações.
- Administração da informação.

Sistema	Aspecto
Administração	Maintenance e Administração de dados Análise econômica Relatório de Operações
Comunicações	Telefonia Internet Dados Internos
Segurança	Controle de Acesso (ID cards, Passwords, etc) Sistema de Alarmes Circuito Fechado de Televisão.
Prevenção de desastres	Controle de Elevadores Controle de Portas e Janelas Sistemas de <i>Sprinkler</i> . Vazamento de água e/ou gás Sistema de apoio a falhas no suprimento de Energia
Controle ambiental	Maintenance dos sistemas de Esgoto Ar Condicionado Iluminação Temperatura Umidade Ventilação
Administração de Energia e Controle	Sistemas automáticos de Iluminação Sistema de Iluminação de Emergência Sistema de Administração do consumo de Energia Sistema geradores de Emergência

Tabela 3.4: Sistemas Requeridos em um Prédio Inteligente

### 3.7 Realidade Virtual

A Realidade Virtual (R.V.) é uma simulação na qual utilizam-se gráficos por computador para criar um universo cujo comportamento reúne as características próprias da realidade [Burdea, 1994]. Com o incremento da capacidade computacional do hardware e a diminuição no custo do mesmo, ocorrido nos últimos anos é possível utilizar de maneira simples modelos gráficos tridimensionais, assim como outras facilidades no acesso a sistemas de entrada e saída para interagir com sistemas gráficos e ainda sonoros.

A R.V. é utilizada hoje em áreas tão variadas como a construção civil [Whyte, 1999], Arquitetura [Navinchandra, 2002], Robótica [Calkin, 1998], [Tianmiao, 2001] e [Freund, 2001], Simulação de Operações [Crook, 1998], Área Militar [Hix, 1999], Ensino [Li, 2002], Medicina [Shelbourn, 1998], [Satava, 1997], [Satava, 1998]. Especificamente com relação aos prédios a RV é utilizada no processo de projeto arquitetônico [Frost, 2000], estudo do desempenho de sistemas modernos de apoio ao funcionamento do prédio [Pilgrim, 2000], comercialização de prédios [Navinchandra, 2002], integração de projetos de grande porte desenvolvidos por equipes multidisciplinares. [Whyte, 1998].

A R.V. tem o potencial para assistir desenvolvedores de *housing* para avaliar e comercializar novos esquemas [Whyte, 1999]. Neste trabalho, a Realidade Virtual é utilizada para a construção de uma maquete de um prédio em forma de ambiente virtual.

#### 3.7.1 Conceitos Fundamentais

As técnicas da Realidade Virtual englobam vários conceitos, entre os quais temos:

- **Realidade Aumentada:** É uma tecnologia mediante a qual o ponto de vista do usuário da realidade é aumentado com informação adicional gerada pelo computador. É uma forma complementar de R.V. que permite ao usuário interagir com um mundo virtual integrado ao mundo real, a realidade aumentada envolve a superposição de informação em forma de imagens tridimensionais geradas por computador sobre uma cena do mundo real. [Whyte, 1999].

- **Ambientes Virtuais:** Um ambiente virtual é um meio que permite a visualização e a simulação de mundos gerados por computador. É preciso fazer a distinção com relação à animação onde imagens são simplesmente reproduzidas em seqüência. Um ambiente virtual pode ser explorado e visualizado de várias formas, com um número potencialmente infinito de perspectivas [Pillgrim, 2001].

- **Sistemas Imersos:** Os sistemas imersos são aqueles sistemas onde se pretende que o usuário fique totalmente imerso no ambiente virtual. Este tipo de sistemas utiliza diferentes acessórios, como luvas, trajes especiais, capacetes; permitindo ao usuário uma visualização integral e uma experiência cognitiva profunda. Estes tipos de sistemas são utilizados especialmente para treino e capacitação. [Burdea, 1994].

- **Desktop Virtual Reality:** É uma técnica que aplica o mesmo software utilizado nos sistemas imersos mas baseado em sistemas de computadores pessoais, onde o usuário interage com o sistema utilizando o teclado, o mouse ou um sistema 3D de mesa como os Joystick. O Desktop Virtual Reality possui os benefícios de visualização e interação com o ambiente virtual sem ter os altos custos dos sistemas imersos. [Tait, 1992].

### 3.7.2 Programação de Ambientes Virtuais

Os modelos de realidade virtual são modelos tridimensionais que podem ser experimentados em tempo real. Um nível de iteração ótimo requer uma taxa de 60 frames por segundo combinado com uma latência de resposta à iteração do usuário de 50 milissegundos. [Whyte, 1999]. Existem várias ferramentas para a construção de Ambientes virtuais, entre elas destaca-se o OpenGL por suas características de portabilidade e desempenho. [Burdea, 1994]. O Ambiente virtual construído para a validação do sistema computacional *GHENESYS* será desenvolvido em OpenGL com uma matriz em Delphi<sup>1</sup>.

<sup>1</sup>Referências na utilização do OpenGL na construção de Sistemas de realidade virtual encontram-se em [Modjeska, 2000] e [Forte, 2001]

## 4. REDE *GHENeSys*

### 4.1 Conceitos Básicos

A rede *GHENeSys* (*General Hierarchical Enhanced Net System*) é uma rede estendida, hierárquica, orientada a objetos, cujos elementos estendidos são apresentados na figura 4.1.

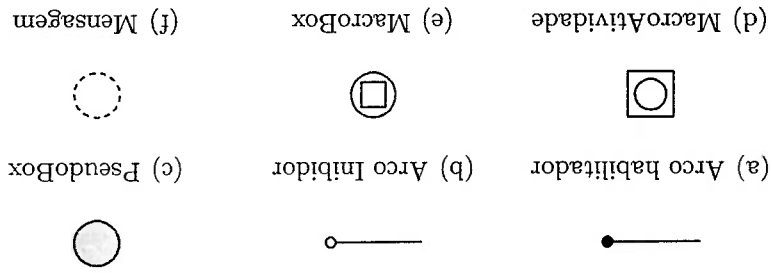


Figura 4.1: Elementos Estendidos da Rede *GHENeSys*

Os elementos passíveis de receber marcações são os passivos: Lugares, Macro-box e PseudoBox, cuja representação gráfica é mostrada na figura 4.2.

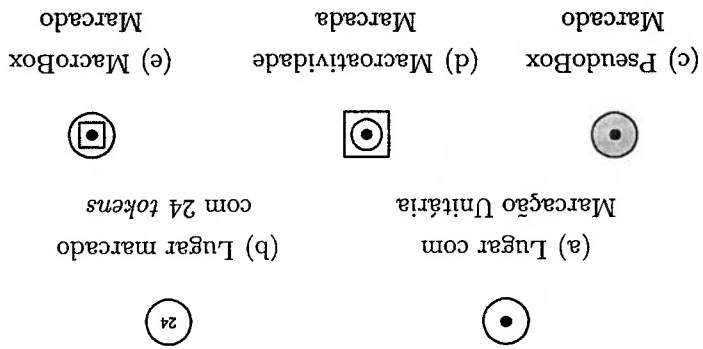

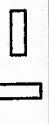









Figura 4.2: Representação dos Elementos Marcados na Rede *GHENeSys*

A tabela 4.1 descreve os elementos que compõem uma rede *GHENeSys* e as relações entre eles. A figura 4.3 apresenta um exemplo de uma rede *GHENeSys*.

Tabela 4.1: Descrição dos Elementos da Rede Ghenesys

NOME	SIMBOLO	DESCRIÇÃO
Lugar		Os lugares na Rede <i>GHENeSys</i> têm as mesmas características da Rede de Petri ordinária
Transição		As transições na Rede <i>GHENeSys</i> têm as mesmas características da Rede de Petri ordinária
Arco		Os arcos normais na rede <i>GHENeSys</i> têm uma função peso ( $w$ ) constante de valor 1.
PseudoBox		Este elemento permite a representação das condições de fronteira do sistema, admitindo marcação vazia ou unitária. deve ser conectado só a elementos ativos e somente através de arcos inibidores ou habilitadores, não pode ter mais de uma conexão com o mesmo elemento ativo.
MacroBox		Representa uma subrede que inicia e finaliza com um lugar, só pode estar conectado com elementos ativos e somente mediante arcos normais. Permite dois tipos de comportamento: Funcionando diretamente com uma rede interna associada ou acumulando uma quantidade de <i>tokens</i> que são liberados em janelas de disparo estabelecidas por um relógio de simulação.
MacroAtividade		Representa uma subrede que inicia e finaliza com uma transição. Só podem estar conectadas a elementos passivos, quando conectada a lugares ou MacroBox terá arcos normais, quando conectada a um PseudoBox só poderá ter arcos inibidores ou habilitadores e pode estar conectada a vários PseudoBox. Possui dois tipos de comportamento: trabalhando com uma Rede associada ou acumulando uma quantidade de <i>tokens</i> e liberando-os em janelas de disparo.
Arco Habilitador		Um Arco habilitador só pode ser traçado entre um PseudoBox e um elemento ativo, se o PseudoBox está marcado a habilitação do elemento ativo depende somente das condições de marcação dos elementos passivos de entrada e saída, se o PseudoBox não está marcado o elemento ativo não está habilitado, independentemente de sua localidade.
Arco Inibidor		Um Arco inibidor só pode ser traçado entre um PseudoBox e um elemento ativo, se o PseudoBox está marcado o elemento ativo está desabilitado, se o PseudoBox não está marcado a habilitação será determinada pelos estados de marcação dos elementos passivos de entrada e saída, isto é de sua localidade.
Mensagem		As mensagens na rede <i>GHENeSys</i> representam uma conexão entre uma transição e um PseudoBox de redes diferentes, assim é possível a marcação ou desmarcação de um PseudoBox por um outro objeto do sistema. Uma mensagem só pode ter arcos de entrada que devem ser traçados partindo de um elemento ativo.

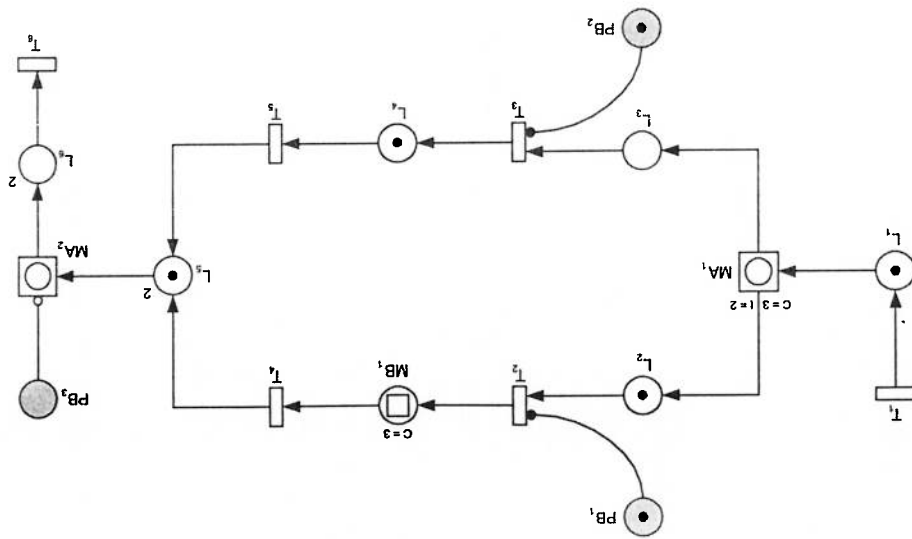


Figura 4.3: Exemplo de um modelo em Rede *GHENesys*

## 4.2 Diagrama de Classes

O diagrama de classes, expressando a descendência e herança dos elementos da rede *GHENesys* e a sua hierarquia, é apresentado na figura 4.4

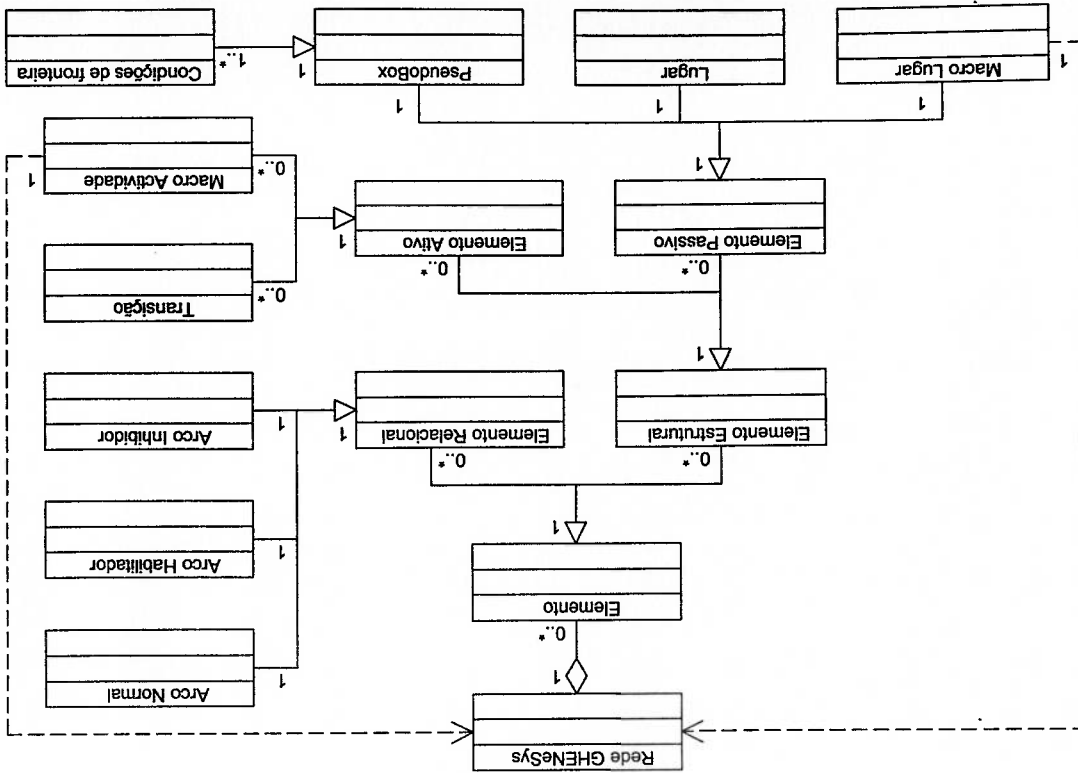


Figura 4.4: Diagrama de Classes da Rede *GHENesys*

Na *GHENeSys* uma mensagem precisa estar vinculada com um PseudoBox, na ocorrência da geração de um mensagem o estado de marcação do PseudoBox pode ser alterado, o fluxo de mensagens é considerado instantâneo. Além disso, múltiplas mensagens, ainda de redes diferentes dentro do projeto, podem estar vinculadas ao mesmo PseudoBox.

## 4.5 Efeito do Fluxo de mensagens na Marcação dos PseudoBox

Quando um elemento ativo (transição ou MacroActividade) está conectado com um PseudoBox, seu estado de habilitação para disparar é influenciado pela marcação do PseudoBox e o tipo de arco que os une. A figura 4.5 apresenta os possíveis casos de conexão entre um PseudoBox e um Elemento Ativo.

## 4.4 Efeito da Marcação dos PseudoBox sobre os Elementos Ativos

1. Uma rede é compatível com um elemento MacroBox se tem um único elemento Inicial e final, e os dois são Lugares.
2. Uma rede é compatível com um elemento MacroActividade se tem um único elemento Inicial e final, e os dois são Transições.
3. Uma rede é incompatível com qualquer MacroElemento se não cumpre nenhuma das condições anteriores.

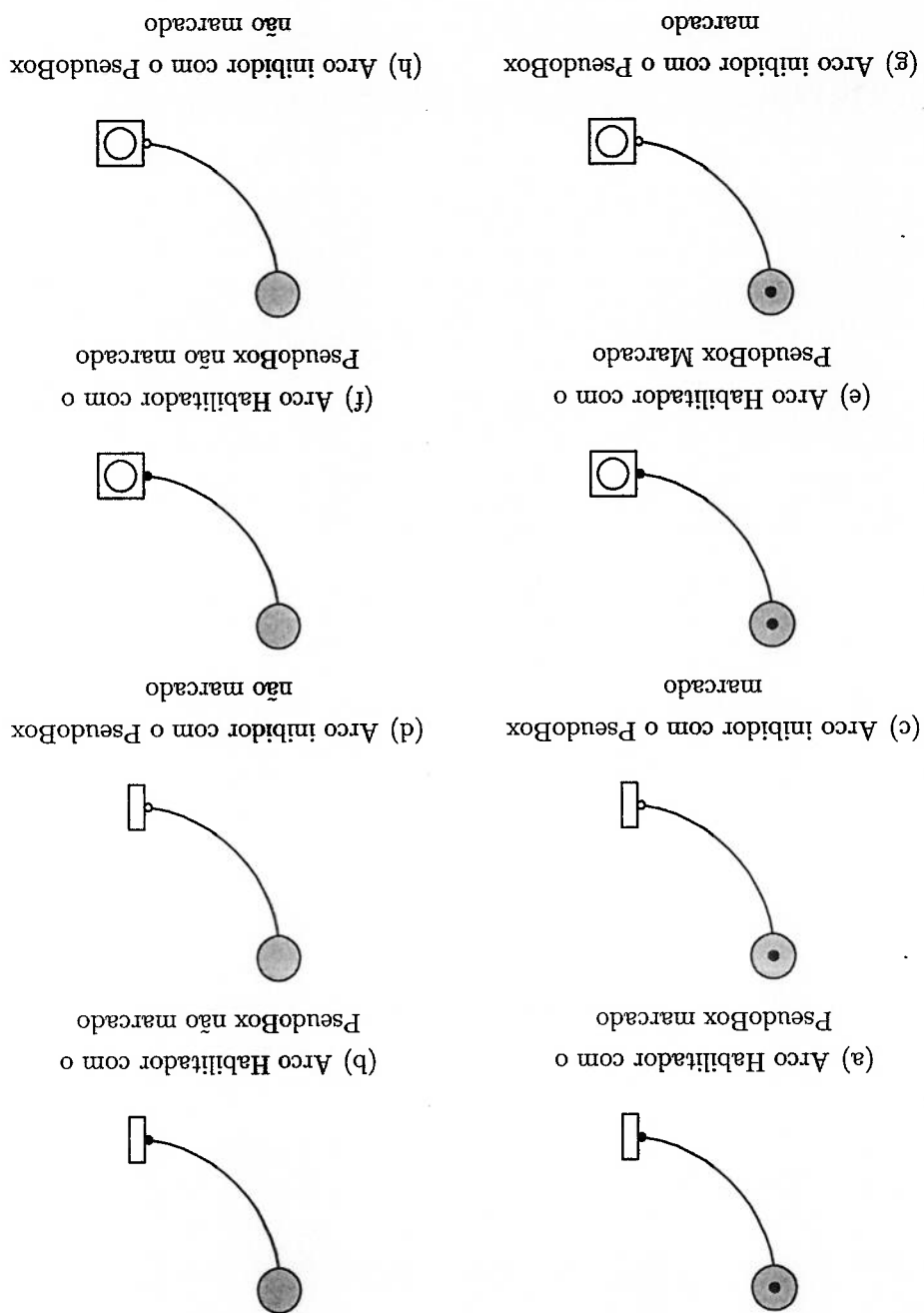
A utilização da *GHENeSys* baseia-se no design de redes que sejam compatíveis com MacroElementos, permitindo o projeto de sistemas a partir de refinamento e composição hierárquica a partir de uma rede inicial, as redes integradas na estrutura se comunicam diante mensagens que alteram a marcação dos PseudoBox das redes no projeto. A seguir se enunciam as regras de compatibilidade das subredes com os MacroElementos:

## 4.3 Regras de Compatibilidade entre Redes e MacroElementos



Figura 4.5: Efeito da Marcação dos PseudoBox sobre os Elementos Ativos

As figuras (b), (c), (f) e (g) apresentam elementos Ativos desabilitados de maneira incondicional, as figuras (a), (d), (e) e (h) mostram elementos ativos cujo estado de habilitação depende da marcação e capacidade de marcação e capacidade de *Preset* e *postset*



## 4.6 Definição Formal

### Definição 4.1 Rede *GHENeSys*

A *GHENeSys* é uma de Rede de Petri estendida definido por uma 4-tupla:  $(P, A, F, w)$ , onde:

$P$  é um conjunto finito de elementos passivos, onde:

$$P = L \cup Mb \cup Pb$$

$L$  é um conjunto finito de lugares.

$Mb$  é um conjunto finito de Macro elementos do tipo passivo.

$Pb$  é um conjunto finito de elementos do tipo Pseudobox.

$A$  é um conjunto finito de Elementos ativos

$$A = T \cup Ma$$

$T$  é um conjunto finito de Transições.

$Ma$  é um conjunto finito de Macro elementos do tipo ativo.

$$P \cap A = \emptyset \text{ e } P \cup A \neq \emptyset.$$

$F$  é a relação de fluxo, sendo um subconjunto do *Bag*  $(P \times A) \cup (A \times P)$  e  $w$  é um função de peso, onde  $w(a) \rightarrow \{1\}$

Na *GHENeSys* os arcos inibidores e/ou habilitadores só podem conectar um Pseudobox e um elemento ativo. Além disso nenhum outro tipo de arco pode ser traçado começando ou terminando em um Pseudobox. Isto é,

$F = (P \times A) \cup (A \times P)$ , como  $P = L \cup Mb \cup Pb$  e  $A = T \cup Ma$  então:

$$F = [(L \cup Mb \cup Pb) \times (T \cup Ma)] \cup [(L \cup Mb \cup Pb)], \text{ assim:}$$

$$F = [(L \cup Mb \cup Pb) \times (T \cup Ma)] \cup [(L \cup Mb)]$$

## 4.7 Representação Algébrica

### Definição 4.2 Representação Algébrica da Rede *GHENeSys*

A rede *GHENeSys* pode ser representada algebricamente por uma 5-tupla de matrizes  $(P, A^+, A^-, H^e, H^d)$  onde:

Se  $|A| = m$  e  $|P| = n$ , então:  $(A^+, A^-)$  são matrizes de ordem  $m \times (n - l)$ , sendo  $m$  o número de elementos ativos,  $n$  o número de elementos passivos e  $l$  o

os elementos ativos da Rede. As colunas destas matrizes relacionam os elementos passivos não Pseudobox com número de Pseudobox presentes na rede;  $(H^e, H^d)$  são matrizes de ordem  $m \times l$ .

A matriz  $A^+$  representa o acréscimo de *tokens* aos lugares e/ou MacroBox depois de disparado um elemento ativo.

$$A^+ = \begin{bmatrix} A_1 & a_{1,1}^+ & a_{1,2}^+ & \dots & a_{1,n-1}^+ \\ A_2 & a_{2,1}^+ & a_{2,2}^+ & \dots & a_{2,n-1}^+ \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_m & a_{m,1}^+ & a_{m,2}^+ & \dots & a_{m,n-1}^+ \end{bmatrix}$$

$$\text{Onde: } a_{ij}^+ = \begin{cases} 1 & \text{se existe um arco entre } T_i \text{ e } P_j \\ 0 & \text{caso contrário} \end{cases}$$

A matriz  $A^-$  representa a subtração de *tokens* aos lugares e/ou MacroBox depois de disparado um elemento ativo.

$$A^- = \begin{bmatrix} A_1 & a_{1,1}^- & a_{1,2}^- & \dots & a_{1,n-1}^- \\ A_2 & a_{2,1}^- & a_{2,2}^- & \dots & a_{2,n-1}^- \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_m & a_{m,1}^- & a_{m,2}^- & \dots & a_{m,n-1}^- \end{bmatrix}$$

$$\text{onde: } a_{ij}^- = \begin{cases} 1 & \text{se existe um arco entre } P_j \text{ e } T_i \\ 0 & \text{caso contrário} \end{cases}$$

As matrizes  $H^e$  e  $H^d$  representam a conexão entre os elementos Pseudobox e os elementos ativos. O valor das entradas destas matrizes não representam fluxo de *tokens*, mas sim informação associada ao tipo de arco: Inibidor  $H^d$  ou habilitador ( $H^e$ )

$$H^e = \begin{bmatrix} T_1 & h_{1,1}^e & h_{1,2}^e & \dots & h_{1,m}^e \\ T_2 & h_{2,1}^e & h_{2,2}^e & \dots & h_{2,m}^e \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ T_m & h_{m,1}^e & h_{m,2}^e & \dots & h_{m,m}^e \end{bmatrix} \quad (4.1)$$

Por ser a rede *GHENeSsys* uma rede hierárquica, além da rede principal, a cada elemento de refinamento hierárquico corresponde uma matriz de incidência e de- mais elementos da representação algébrica.

$$\text{onde: } \left\{ \begin{array}{l} P_j^j \text{ Elemento passivo não Pseudobox na } j\text{-ésima coluna} \\ P_j^{pb} \text{ Pseudobox na } j\text{-ésima coluna} \\ h_{ij} \quad h_{ij}^e + h_{ij}^c \end{array} \right.$$

$$A_g = \begin{bmatrix} A_m \\ \vdots \\ A_2 \\ A_1 \end{bmatrix} = \begin{bmatrix} a_{m,1} & a_{m,2} & \dots & a_{m,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{2,1} & a_{2,2} & \dots & a_{2,n-1} \\ a_{1,1} & a_{1,2} & \dots & a_{1,n-1} \end{bmatrix} \left| \begin{bmatrix} h_{m,n-l+1} & h_{m,n-l+2} & \dots & h_{m,n} \\ \vdots & \vdots & \ddots & \vdots \\ h_{2,n-l+1} & h_{2,n-l+2} & \dots & h_{2,n} \\ h_{1,n-l+1} & h_{1,n-l+2} & \dots & h_{1,n} \end{bmatrix} \right. \begin{bmatrix} P_1^{n-1} & \dots & P_2^{n-1} & P_1^{n-l+1} \\ \dots & \dots & \dots & P_2^{n-l+2} \\ \dots & \dots & \dots & P_1^{n-l+2} \\ \dots & \dots & \dots & P_2^n \end{bmatrix}$$

A matriz de incidência da rede *GHENeSsys*, simbolizada por  $A_g$ , é uma matriz estendida formada a partir de  $A^+$ ,  $A^-$ ,  $H^e$  y  $H^d$ :

#### 4.7.1 Matriz de Incidência

$$\text{onde } h_{ij}^e = \begin{cases} -1 & \text{se existe um arco inibidor entre } P_j^{pb} \text{ e } T_i \\ 0 & \text{caso contrário} \end{cases}$$

$$H^d = \begin{bmatrix} T_m \\ \vdots \\ T_2 \\ T_1 \end{bmatrix} = \begin{bmatrix} h_{m,1}^c & h_{m,2}^c & \dots & h_{m,l}^c \\ \vdots & \vdots & \ddots & \vdots \\ h_{2,1}^c & h_{2,2}^c & \dots & h_{2,l}^c \\ h_{1,1}^c & h_{1,2}^c & \dots & h_{1,l}^c \end{bmatrix} \begin{bmatrix} P_1^{pb} \\ \dots \\ P_2^{pb} \\ \dots \\ P_1^{pb} \end{bmatrix} \quad (4.2)$$

$$\text{onde } h_{ij}^c = \begin{cases} 1 & \text{se existe um arco habilitador entre } P_j^{pb} \text{ e } T_i \\ 0 & \text{caso contrário} \end{cases}$$

### 4.7.2 Efeito da Marcação dos Pseudobox no Fluxo de Marcas

Como foi expresso na tabela 4.1 os Pseudobox representam condições externas ao sistema e possuem marcação unitária. Além disso, é necessário considerar que na rede *GHENeSys* a marcação dos Pseudobox não tem efeito salvo no que se refere à habilitação das transições. Em seguida apresenta-se o procedimento matemático incorporado na equação de estado que garante este comportamento:

Seja  $D$  uma matriz diagonal de ordem  $n$ .

$$D = \begin{bmatrix} d_{1,1} & 0 & \dots & 0 \\ 0 & d_{2,2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & d_{n,n} \end{bmatrix} \quad \text{onde: } d_{i,i} = \begin{cases} 1 & \text{se } i \leq n - l \\ 0 & \text{caso contrário} \end{cases} \quad (4.3)$$

Examinando o produto de  $D \cdot A^g$  tem-se:

$$D \cdot A^g = \begin{bmatrix} d_{1,1} & 0 & \dots & 0 \\ 0 & d_{2,2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & d_{n,n} \end{bmatrix} \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,m} \\ a_{2,1} & a_{2,2} & \dots & a_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n-l,1} & a_{n-l,2} & \dots & a_{n-l,m} \\ b_{n-l+1,1} & b_{n-l+1,2} & \dots & b_{n-l+1,m} \\ b_{n-l+2,1} & b_{n-l+2,2} & \dots & b_{n-l+2,m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n,1} & b_{n,2} & \dots & b_{n,m} \end{bmatrix} = \begin{bmatrix} d_{1,1} \cdot a_{1,1} & d_{1,1} \cdot a_{1,2} & \dots & d_{1,1} \cdot a_{1,m} \\ d_{2,2} \cdot a_{2,1} & d_{2,2} \cdot a_{2,2} & \dots & d_{2,2} \cdot a_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n-l,n-l} \cdot a_{n-l,1} & d_{n-l,n-l} \cdot a_{n-l,2} & \dots & d_{n-l,n-l} \cdot a_{n-l,m} \\ d_{n-l+1,n-l+1} \cdot b_{n-l+1,1} & d_{n-l+1,n-l+1} \cdot b_{n-l+1,2} & \dots & d_{n-l+1,n-l+1} \cdot b_{n-l+1,m} \\ d_{n-l+2,n-l+2} \cdot b_{n-l+2,1} & d_{n-l+2,n-l+2} \cdot b_{n-l+2,2} & \dots & d_{n-l+2,n-l+2} \cdot b_{n-l+2,m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n,n} \cdot b_{n,1} & d_{n,n} \cdot b_{n,2} & \dots & d_{n,n} \cdot b_{n,m} \end{bmatrix}$$

De forma estendida:

$$\text{onde: } \left\{ \begin{array}{l} U' \text{ Marcação dos elementos passivos não-Pseudobox} \\ U^{pb} \text{ Marcação dos Pseudobox} \end{array} \right.$$

$$(4.5) \quad U = U' + U^{pb}$$

com o vetor de marcação  $U$ :  
 O vetor de marcação  $U$  da rede ordinária só considera a existência de lugares, enquanto que a rede *GHENesys* possui elementos passivos, que envolvem tanto os elementos internos da rede (Lugares e MacroBox) como aqueles que representam condições de fronteira (Pseudobox). Assim, é preciso fazer a seguinte distinção

#### 4.7.4 Vetor de Marcação

$$(4.4) \quad C = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \quad \text{onde: } c_j = \text{Capacidade do } j\text{-ésimo lugar}$$

Macrobox da rede:  
 É um vetor  $C$  de ordem  $n$ , que representa as capacidades dos lugares e dos

#### 4.7.3 Vetor de Capacidades

O produto  $D \cdot A^g$  na equação de estado, garante que não ocorra alteração na marcação dos Pseudoboxes da rede. As modificações na marcação dos Pseudobox correspondem ao estado das condições externas do sistema que cada Pseudobox representa, e portanto, não podem estar sob o controle do sistema modelado.

$$D \cdot A^g = \begin{bmatrix} d_{1,1} \cdot a_{1,1} & d_{1,1} \cdot a_{1,2} & \dots & d_{1,1} \cdot a_{1,m} \\ d_{2,2} \cdot a_{2,1} & d_{2,2} \cdot a_{2,2} & \dots & d_{2,2} \cdot a_{2,m} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 \\ d_{n-1,n-1} \cdot a_{n-1,1} & d_{n-1,n-1} \cdot a_{n-1,2} & \dots & d_{n-1,n-1} \cdot a_{n-1,m} \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}$$

Tendo o vetor de habilitação  $H$ , procede-se ao cálculo do efeito dos arcos habilitadores e inibidores. O resultado da presença destes arcos sobre o estado de habilitação das transições do sistema está relacionado com a marcação dos PseudoBoxes. Para relacionar a marcação dos PseudoBoxes e determinar o efeito dos arcos habilitadores e inibidores constrói-se uma matriz diagonal  $T$  de ordem igual

#### 4.7.5.2 Efeito dos arcos Habilitadores e Inibidores

O cálculo da habilitação das MacroAtividades realiza-se mediante uma mensagem que traduz-se em uma consulta sobre o estado interno do elemento.

$$\mu \geq (e[j] \cdot A^-)^T \text{ com: } j \in [1, n-l] \quad (4.7)$$

matriz. Isto é:

O cálculo da habilitação das transições é feito da mesma forma que em 3.7, porém o cálculo da habilitação de cada transição só se aplica à parte não estendida da

#### 4.7.5.1. Habilitação por Marcação

O cálculo do vetor de habilitação da rede *GHENeSys* considera a marcação nos elementos de entrada e saída dos elementos ativos, além da contribuição dos elementos de que representam as condições de fronteira através dos arcos habilitadores e inibidores conectados aos PseudoBox.

#### 4.7.5 Vetor de Habilitação

$$U = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}$$

onde:  $u_j =$  Quantidade de *tokens* presentes no  $j$ -ésimo elemento passivo

$$U = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} u_1' \\ u_{n-l}' \\ \vdots \\ u_{n-l+1}' \\ 0_n \end{bmatrix} + \begin{bmatrix} 0_1 \\ \vdots \\ 0_{n-l} \\ u_{pb}^{n-l+1} \\ \vdots \\ u_{pb}^n \end{bmatrix} \quad (4.6)$$

à quantidade de Pseudobox  $l$ , onde as entradas são os valores de marcação dos

Pseudobox:

$$T_k = \begin{matrix} & \begin{matrix} 1 & 2 & \dots & l \end{matrix} \\ \begin{matrix} 1 \\ \vdots \\ 2 \\ \vdots \\ l \end{matrix} & \begin{bmatrix} t_{1,1} & 0 & \dots & t_{l,1} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & t_{2,2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & t_{l,l} \end{bmatrix} \end{matrix}$$

No caso dos arcos inibidores a relação é estabelecida mediante o produto de  $H^d$

e  $T_k$ , isto é:

$$H^d \cdot T_k = \begin{bmatrix} h_{d,1}^1 & \dots & h_{d,1}^{l,n} \\ h_{d,2}^1 & \dots & h_{d,2}^{l,n} \\ \vdots & \ddots & \vdots \\ h_{d,m,1}^1 & \dots & h_{d,m,1}^{l,n} \end{bmatrix} \begin{bmatrix} t_{1,1} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & t_{2,2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & t_{n,n} \end{bmatrix}$$

Assim é possível calcular o efeito total para cada habilitação através da somatória

de cada linha da matriz anterior.

$$\begin{bmatrix} h_{d,1}^1 \cdot (t_{1,1})^k & h_{d,1}^2 \cdot (t_{2,2})^k & \dots & h_{d,1}^{l,n} \cdot (t_{n,n})^k \\ h_{d,2}^1 \cdot (t_{1,1})^k & h_{d,2}^2 \cdot (t_{2,2})^k & \dots & h_{d,2}^{l,n} \cdot (t_{n,n})^k \\ \vdots & \vdots & \ddots & \vdots \\ h_{d,m,1}^1 \cdot (t_{1,1})^k & h_{d,m,2}^2 \cdot (t_{2,2})^k & \dots & h_{d,m,1}^{l,n} \cdot (t_{n,n})^k \end{bmatrix} \begin{bmatrix} \sum_{j=1}^n h_{d,1,j}^1 \cdot (t_{j,j})^k \\ \sum_{j=1}^n h_{d,2,j}^2 \cdot (t_{j,j})^k \\ \vdots \\ \sum_{j=1}^n h_{d,m,j}^m \cdot (t_{j,j})^k \end{bmatrix}$$

O resultado do efeito dos arcos inabilitadores no estado de habilitação das transições

é representado pelo vetor  $H_-^k$ :

$$H_-^k = \begin{bmatrix} h_{-1}^k \\ h_{-2}^k \\ \vdots \\ h_{-m}^k \end{bmatrix} \quad \text{onde:} \quad (h_{-i}^k)^k = \begin{bmatrix} \sum_{j=1}^n h_{d,1,j}^1 \cdot (t_{j,j})^k \\ \sum_{j=1}^n h_{d,2,j}^2 \cdot (t_{j,j})^k \\ \vdots \\ \sum_{j=1}^n h_{d,m,j}^m \cdot (t_{j,j})^k \end{bmatrix} \quad (4.8)$$



Para o cálculo do efeito da existência de arcos habilitadores é preciso considerar que a marcação dos Pseudobox afeta de forma diferente o estado de habilitação das transições. Assim, a presença ou ausência de *tokens* produz o efeito oposto. A relação entre a marcação dos Pseudoboxes e a existência dos arcos inibido-res está dada pelo produto de  $H^e$  pela matriz resultante da subtração da matriz identidade de  $T^k$ :

$$H^e \cdot (T^k - I) = \begin{bmatrix} h_{1,1}^e & \dots & h_{1,2}^e & \dots & h_{1,n}^e \\ h_{2,1}^e & \dots & h_{2,2}^e & \dots & h_{2,n}^e \\ \vdots & & \vdots & & \vdots \\ h_{m,1}^e & \dots & h_{m,2}^e & \dots & h_{m,n}^e \end{bmatrix} \begin{bmatrix} t_{1,1} - 1 & 0 & 0 & \dots & 0 \\ 0 & t_{2,2} - 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & t_{n,n} - 1 \end{bmatrix}$$

Estabelecida esta relação é possível calcular o efeito total para cada habilitação que é obtido através da somatória de cada fila na matriz anterior.

$$\begin{bmatrix} h_{1,1}^e \cdot (t_{1,1} - 1) & h_{1,2}^e \cdot (t_{2,2} - 1) & \dots & h_{1,n}^e \cdot (t_{n,n} - 1) \\ h_{2,1}^e \cdot (t_{1,1} - 1) & h_{2,2}^e \cdot (t_{2,2} - 1) & \dots & h_{2,n}^e \cdot (t_{n,n} - 1) \\ \vdots & \vdots & \ddots & \vdots \\ h_{m,1}^e \cdot (t_{1,1} - 1) & h_{m,2}^e \cdot (t_{2,2} - 1) & \dots & h_{m,n}^e \cdot (t_{n,n} - 1) \end{bmatrix}$$

$$\begin{bmatrix} \sum_{j=1}^n h_{1,j}^e \cdot (t_{j,j}^k - 1) \\ \vdots \\ \sum_{j=1}^n h_{2,j}^e \cdot (t_{j,j}^k - 1) \\ \vdots \\ \sum_{j=1}^n h_{m,j}^e \cdot (t_{j,j}^k - 1) \end{bmatrix}$$

Assim, o resultado do efeito dos arcos habilitadores é representado pelo vetor  $H^+$ :

Partindo do vetor de simulação de disparo  $U_s^{k+1}$  e do vetor de capacidades  $C$  calcula-se o vetor de marcação do conflito  $U_c^{k+1}$ , isto é:

$$(4.12) \quad U_s^{k+1} = U_k + (D \cdot A_T^g) \cdot (H^k)$$

Realiza-se a detecção de conflitos seguindo as mesmas considerações e procedimentos apresentados em 3.1.1.6, porém esta só se aplica às marcações geradas a partir do vetor de marcação dos elementos passivos excluindo-se os Pseudo-Box. Assim o vetor de simulação de disparo ( $U_s^{k+1}$ ) em uma rede *GHENeSys* é calculado mediante a equação 4.12.

#### 4.7.7 Detecção de Conflitos

Onde  $(F^k) \leq (H^k)$

$$(4.11) \quad U^{k+1} = U_k + (D \cdot A_T^g) \cdot (F^k)$$

Aplica-se a equação de estado da *GHENeSys* a cada um dos elementos hierarquicos além da rede principal. É uma equação linear semelhante à da rede elementar e por tanto o cálculo das propriedades das redes pode ser feito de forma análoga ao da Rede de Petri ordinária.

#### 4.7.6 Equação de Estado

$$(4.10) \quad H^k = \begin{bmatrix} h_1^k \\ h_2^k \\ \vdots \\ h_m^k \end{bmatrix} \quad \text{onde: } (h_i^k)_k = \begin{cases} 0 & \text{caso contrário} \\ 1 & \text{se } (h_i)_k + (h_i^d)_k + (h_i^e)_k > 0 \end{cases}$$

O efeito global dos arcos inibidores e habilitadores presentes na rede sobre o vetor de habilitação  $H$  é dado por:

$$(4.9) \quad (H_+)^k = \begin{bmatrix} h_+^1 \\ h_+^2 \\ \vdots \\ h_+^m \end{bmatrix} \quad \text{onde: } (h_e^i)_k = \begin{bmatrix} \sum_{j=1}^n h_{e_{1,j}}^1 \cdot ((t_{j,j})^k - 1) \\ \vdots \\ \sum_{j=1}^n h_{e_{2,j}}^2 \cdot ((t_{j,j})^k - 1) \\ \vdots \\ \sum_{j=1}^n h_{e_{m,j}}^m \cdot ((t_{j,j})^k - 1) \end{bmatrix}$$

Onde  $M_k$  é o vetor que contém a ocorrência dos mensagens associados ao  $n$ -ésimo PseudoBox do Projeto. Um valor um (1) na  $i$ -ésima entrada do vetor significa que a mensagem correspondente foi gerada, pelo contrário a entrada tem valor zero (0). O efeito do fator  $F^m$  na marcação dos PseudoBox é apresentada na tabela 4.2.

$$(4.14) \quad \text{Seja } M_k = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{bmatrix} \rightarrow F^m(\text{PseudoBox})_n = \begin{cases} 0 & \text{se } M_k > 1 \\ 1 & \text{caso contrário} \end{cases}$$

O fluxo de mensagens é considerado instantâneo, portanto uma mensagem afeta de maneira imediata os PseudoBoxes associados. Mas não se deve considerar como fluxo de *tokens*, porém uma mudança nas condições de fronteira do sub sistema representado pela rede. O fator de efeito das mensagens na marcação dos PseudoBox ( $F^m$ ) é apresentada na equação 4.14

fluxo de mensagens é considerado instantâneo, portanto uma mensagem afeta de maneira imediata os PseudoBoxes associados. Mas não se deve considerar como fluxo de *tokens*, porém uma mudança nas condições de fronteira do sub sistema representado pela rede. O fator de efeito das mensagens na marcação dos PseudoBox ( $F^m$ ) é apresentada na equação 4.14

#### 4.7.8 . Cálculo de Efeito de Mensagens nos PseudoBox

Uma entrada diferente de zero (0) na  $j$ -ésima posição do vetor de marcação de conflito implica que na  $j$ -ésima coluna da matriz  $A^-$ , encontram-se presentes transições que apresentam um conflito. Se a entrada é negativa (-1) o conflito que apresentam é do tipo *preset*, se a entrada é positiva (1) o conflito é de tipo *postset*.

$$(4.13) \quad U_c^{k+1} = \begin{bmatrix} u_1^c \\ u_2^c \\ \vdots \\ u_{n-1}^c \end{bmatrix} \quad \text{onde: } (u_j^c)_k = \begin{cases} -1 & \text{se } (u_j^c)_k > 0 \\ 1 & \text{se } (u_j^c)_k < c_j \\ 0 & \text{caso contrário} \end{cases}$$

Tabela 4.2: Efeito do Fluxo de mensagens nos PseudoBoxes

Estado Inicial da Marcação	valor de $F_m$	Estado Final da Marcação
0	1	1
1	0	1
1	1	1
0	0	0

### 4.7.9 Dinâmica da Rede *GHENeSys*

A dinâmica da Rede *GHENeSys* é apresentada na figura 4.6. Nota-se que este algoritmo aplica-se a todas as Redes associadas aos MacroElementos do sistema. O Algoritmo detalhado da execução da rede *GHENeSys* é apresentado nas seções 5.5.1 e 5.5.2.

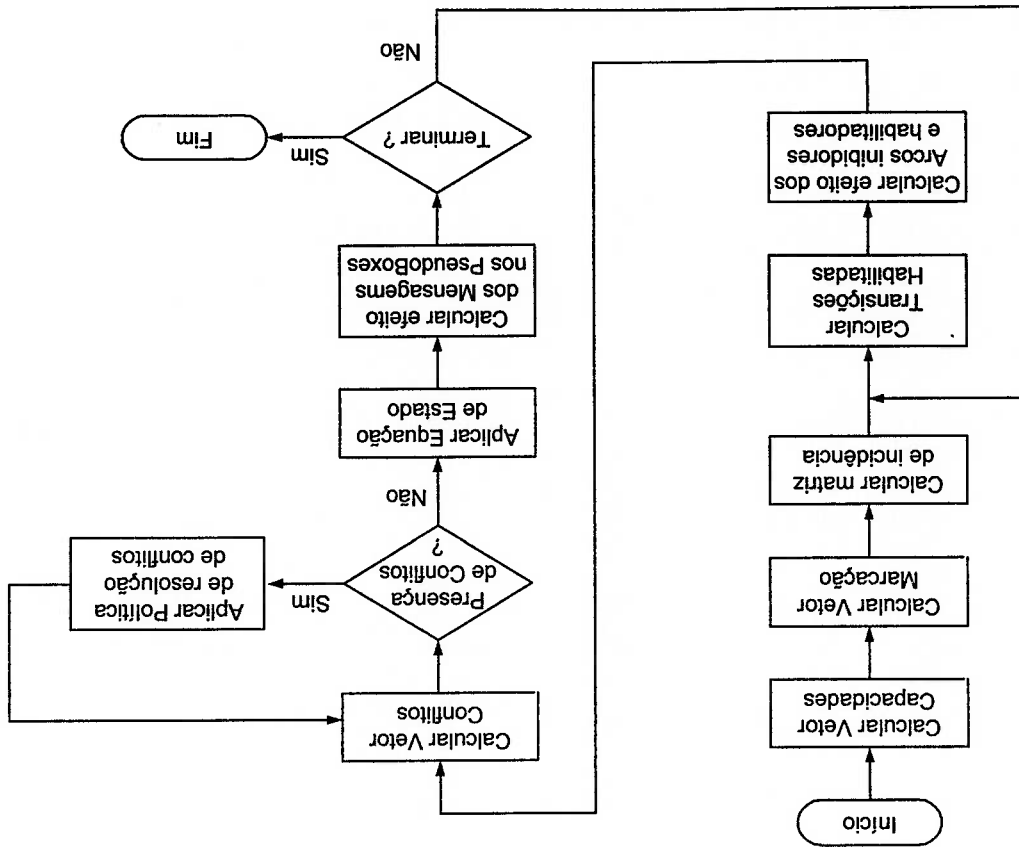
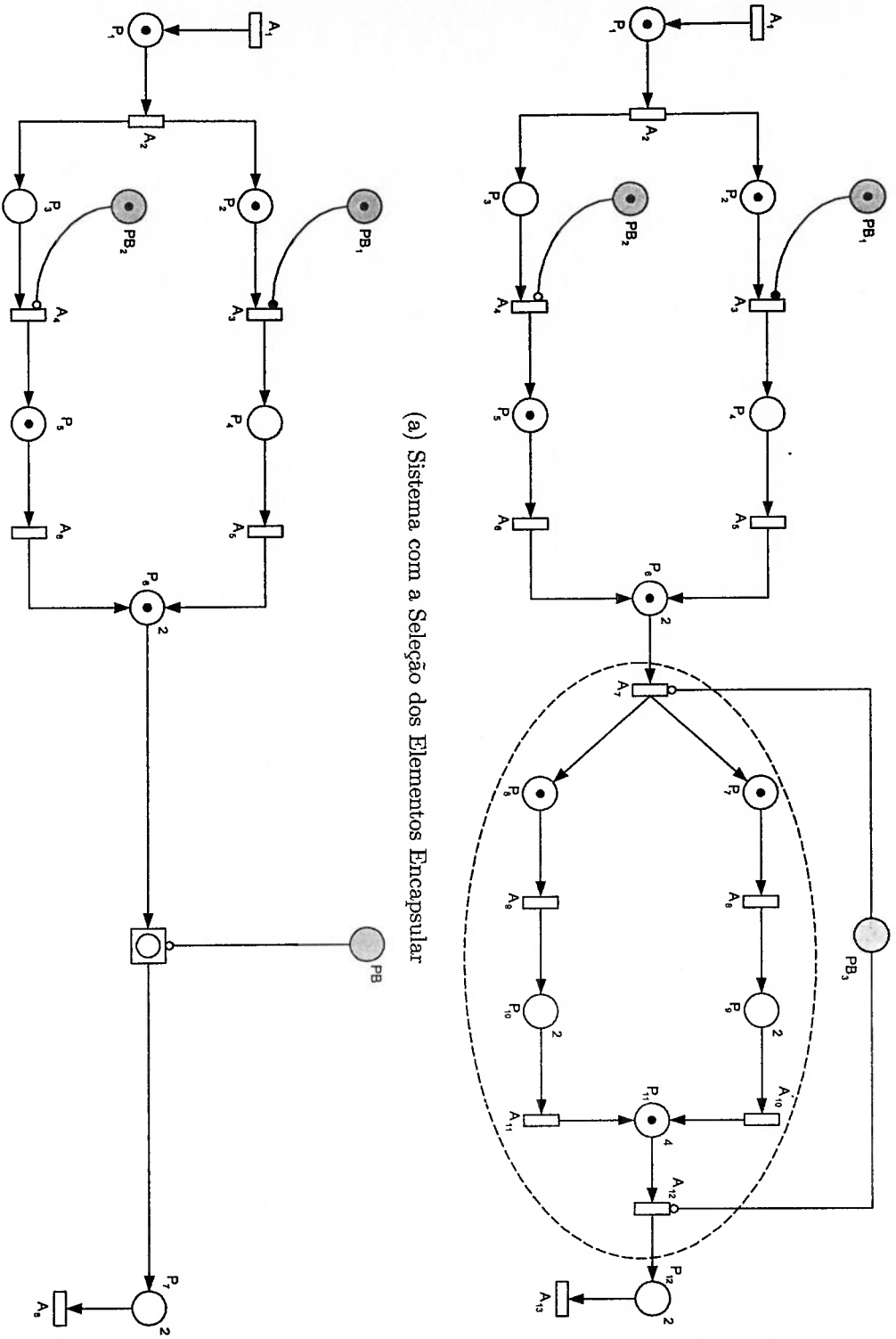


Figura 4.6: Algoritmo da Dinâmica de uma Rede *GHENeSys*





(a) Sistema com a Seleção dos Elementos Encapsular

(b) Resultado do Encapsulamento

Figura 4.7: Exemplo de Abstração e Encapsulamento

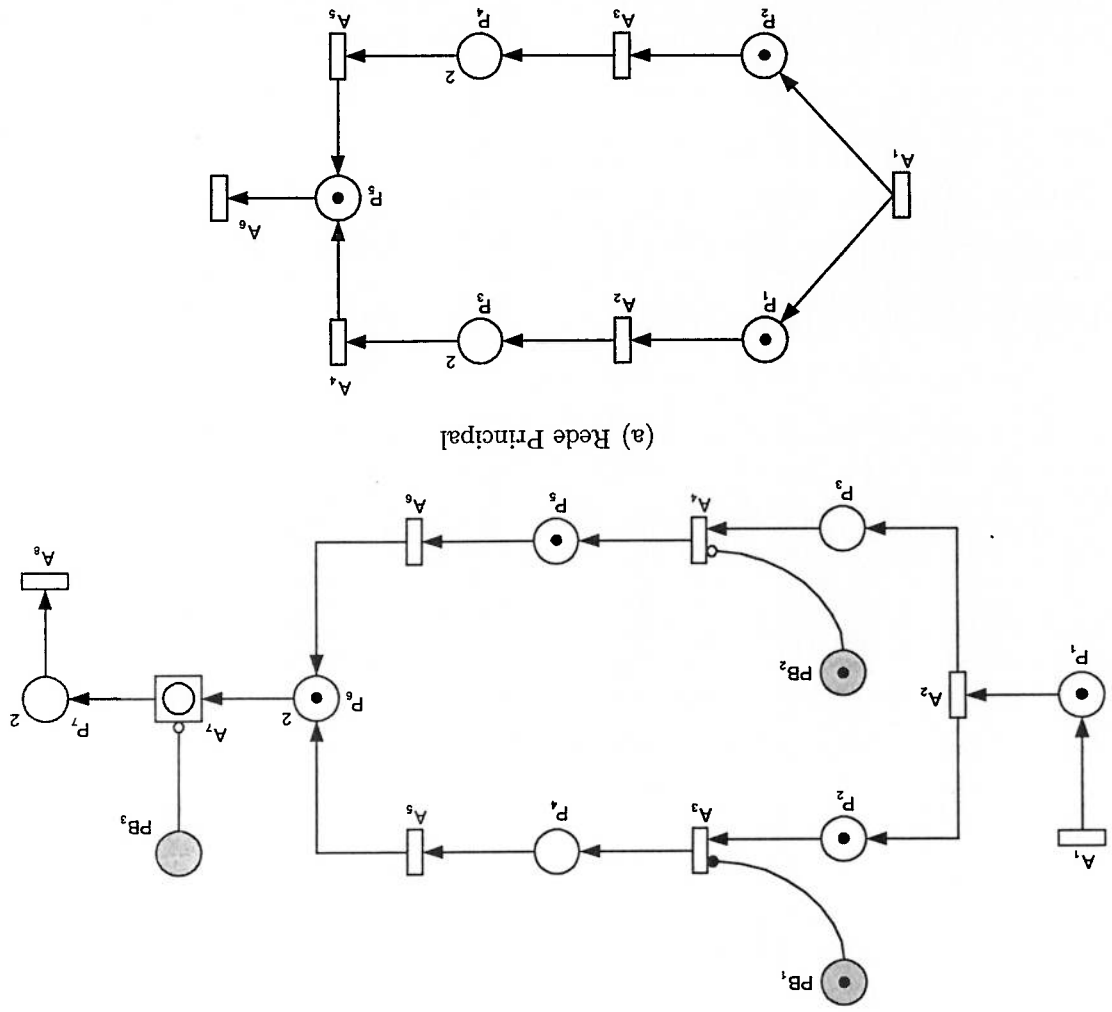
$$A^+ = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_2 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hat{A}_3 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_4 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_5 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ A_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ A_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \\ P_7 \\ P_8 \\ P_9 \\ P_{10} \\ P_{11} \\ P_{12} \\ P_{13} \end{matrix}$$

$$A^- = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_3 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_4 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_5 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_7 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_8 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \\ P_7 \\ P_8 \\ P_9 \\ P_{10} \\ P_{11} \\ P_{12} \\ P_{13} \end{matrix}$$

Assim, o sistema pode ser transformado no sistema equivalente apresentado na figura 4.8 onde 4.8(b) é a rede associada à MacroAtividade  $MA_1$  da 4.8(a). As matrizes de estado destas redes são apresentadas a seguir:

Partindo de este suposto. Macrobox. Partindo de este suposto. de entrada e a marcação do elemento passivo de saída da subred encapsulada no *pre-set* e *post-set* tem que ser calculados com a capacidade do elemento passivo com um elemento MacroBox as condições de habilitação dos elementos ativos durante o comportamento do sistema. No caso que o refinamiento seja realizado simultaneas. Caso contrario, a matriz construida não estará descrevendo adequada acima basea-se no principio que as transições inicial e final das subredes, sejam E preciso considerar que a construção da matriz de incidencia apresentada

$$A_g = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_2 & -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_3 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_4 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_5 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_7 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ A_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ A_9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ A_{10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ A_{11} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ A_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ A_{13} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \\ P_7 \\ P_8 \\ P_9 \\ P_{10} \\ P_{11} \\ P_{12} \\ P_{13} \\ P_{14} \\ P_{15} \\ P_{16} \\ P_{17} \\ P_{18} \\ P_{19} \\ P_{20} \\ P_{21} \\ P_{22} \\ P_{23} \\ P_{24} \\ P_{25} \\ P_{26} \\ P_{27} \\ P_{28} \\ P_{29} \\ P_{30} \\ P_{31} \\ P_{32} \\ P_{33} \\ P_{34} \\ P_{35} \\ P_{36} \\ P_{37} \\ P_{38} \\ P_{39} \\ P_{40} \\ P_{41} \\ P_{42} \\ P_{43} \\ P_{44} \\ P_{45} \\ P_{46} \\ P_{47} \\ P_{48} \\ P_{49} \\ P_{50} \\ P_{51} \\ P_{52} \\ P_{53} \\ P_{54} \\ P_{55} \\ P_{56} \\ P_{57} \\ P_{58} \\ P_{59} \\ P_{60} \\ P_{61} \\ P_{62} \\ P_{63} \\ P_{64} \\ P_{65} \\ P_{66} \\ P_{67} \\ P_{68} \\ P_{69} \\ P_{70} \\ P_{71} \\ P_{72} \\ P_{73} \\ P_{74} \\ P_{75} \\ P_{76} \\ P_{77} \\ P_{78} \\ P_{79} \\ P_{80} \\ P_{81} \\ P_{82} \\ P_{83} \\ P_{84} \\ P_{85} \\ P_{86} \\ P_{87} \\ P_{88} \\ P_{89} \\ P_{90} \\ P_{91} \\ P_{92} \\ P_{93} \\ P_{94} \\ P_{95} \\ P_{96} \\ P_{97} \\ P_{98} \\ P_{99} \\ P_{100} \end{matrix}$$



Assim a matriz de incidência de 4.8(a) é:

$$H^e = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_4 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ A_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$H^d = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_7 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ A_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figura 4.8: Subrede Encapsulada em uma MacroAtividade



Uma consideração final deve ser levada em conta: no caso que da MacroAtividade  $MA_1$  não apresentar *marcação consistente* o sistema representado é o da figura 4.9. Neste caso, somente é limitado o ingresso de *tokens* à MacroAtividade, mas é permitida a saída dos mesmos. A matriz de incidência é então dada pela matriz seguinte:

$$A_g = \begin{matrix} & \begin{matrix} P_1' & P_2' & P_3' & P_4' & P_5' \end{matrix} \\ \begin{matrix} A_1 \\ A_2 \\ A_3 \\ A_4 \\ A_5 \\ A_6 \end{matrix} & \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix} \end{matrix}$$

por:

Assim a matriz de incidência da Subrede encapsulada (figura 4.8(b)) é dada

$$A^+ = \begin{matrix} & \begin{matrix} P_1 & P_2 & P_3 & P_4 & P_5 \end{matrix} \\ \begin{matrix} A_1 \\ A_2 \\ A_3 \\ A_4 \\ A_5 \\ A_6 \end{matrix} & \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{matrix}$$

$$A^- = \begin{matrix} & \begin{matrix} P_1 & P_2 & P_3 & P_4 & P_5 \end{matrix} \\ \begin{matrix} A_1 \\ A_2 \\ A_3 \\ A_4 \\ A_5 \\ A_6 \end{matrix} & \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{matrix}$$

As matrizes de estado da Subrede encapsulada (figura 4.8(b)) são:

$$A_g = \begin{matrix} & \begin{matrix} P_1' & P_2' & P_3' & P_4' & P_5' & P_6' & P_7' & P_8' & P_9' & P_{10}' \end{matrix} \\ \begin{matrix} A_1 \\ A_2 \\ A_3 \\ A_4 \\ A_5 \\ A_6 \\ A_7 \\ A_8 \end{matrix} & \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \end{matrix}$$

## 4.9 Projeto de Sistemas com a Rede *GHENeSys*

O método proposto para o projeto de sistemas com a rede *GHENeSys* é apresentada na figura 4.10, onde:

- **Identificação do Sistema:** Envolve a identificação do sistema, delimitação das condições de fronteira e das funções que se espera que o sistema realize.
- **Análise Orientada a Objetos:** É a identificação das entidades conceituais cuja posterior interação representa o comportamento do sistema, é dizer involucra a identificação dos objetos mediante (AOO). (ver seção 3.2.2).
- **Construção dos Modelos dos Objetos:** Cada um dos objetos identificados na etapa anterior é modelado utilizando-se a rede *GHENeSys*.
- **Encapsulamento:** As redes se encapsulam em MacroElementos constituindo classes. Assim, os objetos que compõem o sistema são representados por instâncias destes MacroElementos, as conexões dos MacroElementos entre si e com os demais elementos representam o fluxo de mensagens em um nível hierárquico do sistema.

• **Construção da Árvore de hierarquia:** Os diferentes níveis hierárquicos são integrados de acordo ao nível de abstração, constituindo a árvore de hierarquia.

$$A_g = \begin{bmatrix} A_1 & A_2 & A_3 & A_4 & A_5 & A_6 & A_7 & A_8 & A_9 & A_{10} & A_{11} & A_{12} & A_{13} \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$P_1 \quad P_2 \quad P_3 \quad P_4 \quad P_5 \quad P_6 \quad P_7 \quad P_8 \quad P_9 \quad P_{10} \quad P_{11} \quad P_{12} \quad P_{13}$

$P_1^{pb} \quad P_2^{pb} \quad P_3^{pb} \quad P_4^{pb} \quad P_5^{pb} \quad P_6^{pb} \quad P_7^{pb} \quad P_8^{pb} \quad P_9^{pb} \quad P_{10}^{pb} \quad P_{11}^{pb} \quad P_{12}^{pb} \quad P_{13}^{pb}$

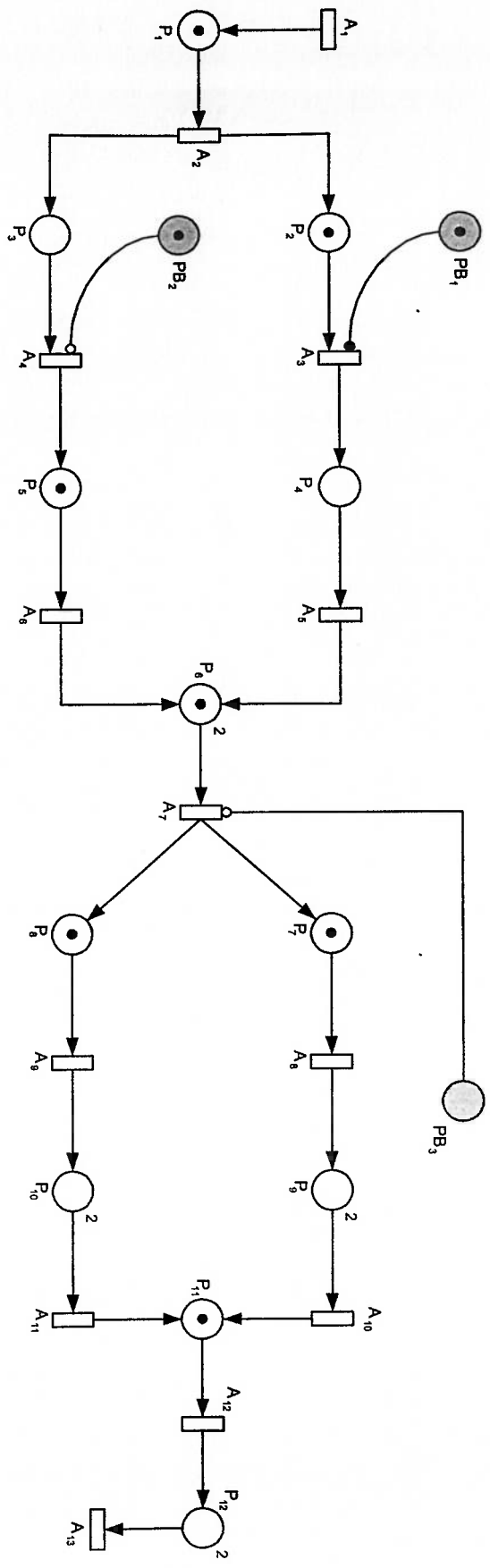
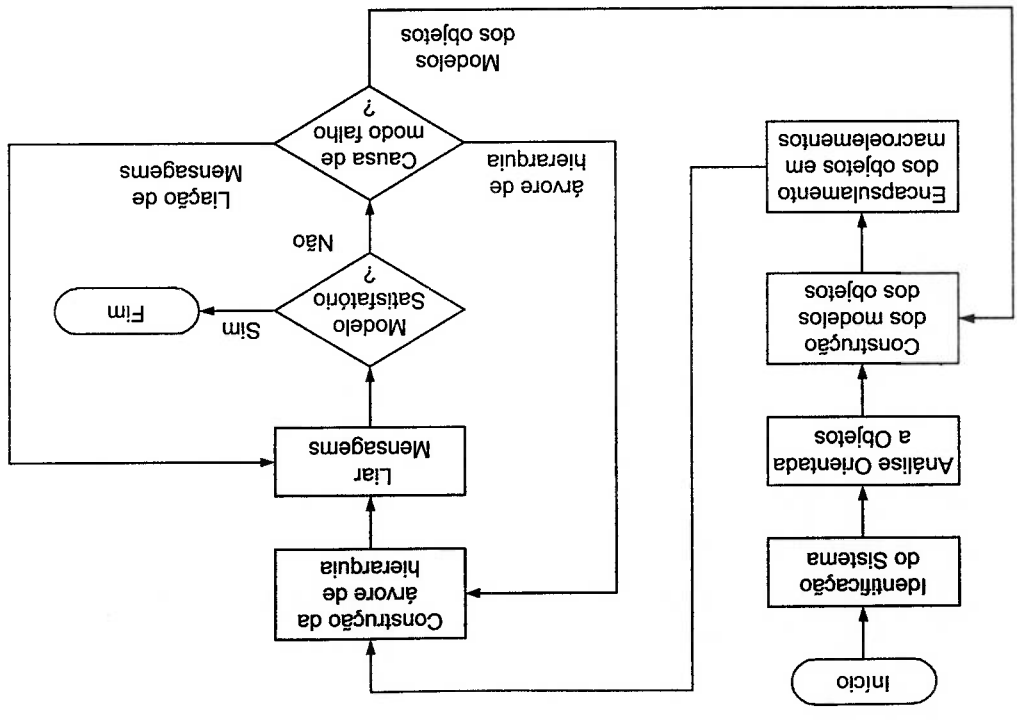


Figura 4.9: Abstração na Rede sem Marcação Consistente

Figura 4.10: Método de Projeto de Sistemas com a Rede *GHENeSys*



## 5. *GHENESYS*: O SISTEMA COMPUTACIONAL

Para desenvolver aplicações práticas com Redes de Petri é preciso utilizar ambiente de Software [Murata, 1989]. No momento não existe uma ferramenta computacional específica para a construção e execução de redes *GHENESys*, portanto é necessário realizar-se o levantamento de requisitos que permitam identificar características necessárias e desejáveis de um ambiente de software que implemente a proposta apresentada na seção 4, e que reúna características comuns às ferramentas existentes para o estudo das redes de Petri. Trata-se de fato de um ambiente de projeto baseado em Redes de Petri Orientadas a Objetos onde se destaca o papel da hierarquia. O Nome escolhido para o software é *GHENESYS*.

### 5.1 Características Gerais

Utilizou-se como referência o site do *DAIMI*<sup>1</sup>, considerado como representativo da área, para o levantamento das características gerais dos sistemas e ambientes de modelagem em Redes de Petri existentes. As principais características dos 51 sistemas de desenvolvimento cadastrados são apresentadas na tabela 5.1. As funcionalidades comuns observadas são:

1. Editor Gráfico
2. Jogador de Marcas
3. Cálculo de Invariantes de Lugar e de Transição

A tabela 5.2 apresenta uma discriminação dos sistemas que apresentaram propriedades similares as características desejadas para a ferramenta computacional *GHENESYS*.

<sup>1</sup><http://www.daimi.au.dk/PetriNets/>, repositório de redes de Petri do departamento de Ciência da Computação da Universidade de Aarhus, visitada em Novembro 25 do 2003

Tabela 5.1: Características das Ferramentas de Redes de Petri

No.	Nome do Software	Tipos de Redes Suportadas	Características	Sistema Operacional	Licença	Página Web
1	ALPHA/Sim	Alto Nível Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida Ana. de desempenho simples	SunOS Solaris Windows	Comercial	<a href="http://www.alpha-tech.com/secondary/techpro/alphasim/alphasim.html">http://www.alpha-tech.com/secondary/techpro/alphasim/alphasim.html</a>
2	ARP	Lugar/Transição Com Tempo	Simulação Rápida Espaço de estados Invariantes de Lugar Invariantes de Transição Análise estrutural Ana. de desempenho simples	MS DOS	Gratuito	<a href="http://www.ppgia.pucpr.br/~maziero/Petri/">http://www.ppgia.pucpr.br/~maziero/Petri/</a>
3	Artifex	Orientada a Objetos Alto Nível Lugar/Transição Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida Análise estrutural Ana. de desempenho avançado	Sun, SunOS HP-Unix S.G. IRIX Linux Windows	Comercial (Desconto acadêmico)	<a href="http://www.rsoftdesign.com/">http://www.rsoftdesign.com/</a>
4	CoopnTools	Alto Nível	Editor Gráfico Simulação Rápida Análise estrutural	Java	Gratuito	<a href="http://iglwww.epfl.ch/Conform">http://iglwww.epfl.ch/Conform</a>
5	CPN-AMI	Alto Nível Lugar/Transição	Editor Gráfico Simulação Rápida Espaço de estados Invariantes de Lugar Invariantes de Transição Análise estrutural Modelagem Modular	Sun Linux Macintosh	Gratuito	<a href="http://www.lip6.fr/cpn-ami">http://www.lip6.fr/cpn-ami</a>
6	CPN-AMI	Alto Nível Com Tempo	Editor Gráfico Jogador de Marcas	Windows	Gratuito	<a href="http://www.dairni.au.dk/CPNtools/">http://www.dairni.au.dk/CPNtools/</a>

			<p>Simulação Rápida  Invariantes de Lugar  Espaço de estados  Ana. de desempenho simples  Formato de intercambio  de arquivos</p>			
7	<b>Design/CPN</b>	Alto Nível Com Tempo	<p>Editor Gráfico  Jogador de Marcas  Simulação Rápida  Invariantes de Lugar  Espaço de estados  Ana. de desempenho simples  Formato de intercambio  de arquivos</p>	Windows HP Irix Linux	Gratuito	<a href="http://www.daimi.au.dk/designCPN/">http://www.daimi.au.dk/ designCPN/</a>
8	<b>DPN Schematic</b>	Lugar/Transição Com Tempo D-extended PN	<p>Editor Gráfico  Simulação Rápida  Ferramenta esquentática</p>	Windows	Gratuito	<a href="http://www.petrtool.narod.ru/">http://www.petrtool.narod.ru/</a>
9	<b>EDS Petri Net Tool</b>	Lugar/Transição Stochastic Petri Nets Com Tempo Coloridas	<p>Editor Gráfico  Jogador de Marcas  Simulação Rápida  Análise estrutural  Ana. de desempenho simples  Formato de intercambio  de arquivos  Linguagem de programação</p>	Windows	Comercial (Desconto acadêmico)	
10	<b>ExSpect</b>	Alto Nível Lugar/Transição Stochastic Petri Nets Com Tempo Modelagem Hierarquico	<p>Editor Gráfico  Jogador de Marcas  Simulação Rápida  Análise estrutural  Ana. de desempenho simples  Ana. de desempenho avançado</p>	Windows	Comercial (Desconto acadêmico)	<a href="http://www.exspect.com/">http://www.exspect.com/</a>

11	<b>F-net</b>	Stochastic Petri Nets Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida Espaço de estados Invariantes de Lugar Invariantes de Transição Análise estrutural Ana. de desempenho simples Ana. de desempenho avançado	Windows OS/2	Comercial (Desconto acadêmico)	<a href="http://www.f.ru/os/petri.php3">http://www.f.ru/os/ petri.php3</a>
12	<b>GDDToolkit</b>	Alto Nível Lugar/Transição	Desenho Automático	Sun Linux Windows	Comercial (Gratuito acadêmico)	<a href="http://www.dia.uniroma3.it/~gdt/">http://www.dia.uniroma3.it/ ~gdt/</a>
13	<b>GreatSPN</b>	Alto Nível Stochastic Petri Nets Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida Espaço de estados Esp. de est. Condensados Invariantes de Lugar Invariantes de Transição Análise estrutural Ana. de desempenho avançado	Sun Linux	Comercial (Desconto acadêmico)	<a href="http://www.di.unito.it/~greatspn/">http://www.di.unito.it/ ~greatspn/</a>
14	<b>HIQPN-Tool</b>	Alto Nível Stochastic Petri Nets Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida Espaço de estados Invariantes de Lugar Invariantes de Transição Análise estrutural Formato de intercambio de arquivos	Sun	Gratuito	<a href="http://ls4-www.informatik.uni-dortmund.de/QPN/">http://ls4-www.informatik.uni- dortmund.de/QPN/</a>



15	<b>HPSim</b>	Lugar/Transição Stochastic Petri Nets Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida Ana. de desempenho simples	Windows	Gratuito	<a href="http://www.winpesim.de/">http://www.winpesim.de/</a>
16	<b>INA</b>	Alto Nível Lugar/Transição Com Tempo	Espaço de estados Esp. de est. Condensados Invariantes de Lugar Invariantes de Lugar Net Reductions Análise estrutural Análise de desempenho Simples e Avançado Formato de intercambio de arquivos	Sun Linux Windows	Gratuito	<a href="http://www.informatik.hu-berlin.de/~starke/ina.html">http://www.informatik.hu-berlin.de/~starke/ina.html</a>
17	<b>INCOME Process Designer</b>	Alto Nível Lugar/Transição Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida	Sun OS HP-Unix Linux Windows Java	Comercial (Desconto acadêmico)	<a href="http://www.get-process.com/">http://www.get-process.com/</a>
18	<b>JARP</b>	Lugar/Transição	Editor Gráfico Jogador de Marcas Espaço de estados	Java	Gratuito	<a href="http://jarp.sf.net/">http://jarp.sf.net/</a>
19	<b>JFern</b>	Orientada a Objetos Alto Nível Lugar/Transição Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida Espaço de estados Ana. de desempenho simples Formato de intercambio de arquivos	Java	Gratuito	<a href="http://sourceforge.net/projects/jfern">http://sourceforge.net/projects/jfern</a>
20	<b>Maria</b>	Alto Nível	Jogador de Marcas	Sun OS	Gratuito	<a href="http://www.tes.hut.fi/">http://www.tes.hut.fi/</a>

		Lugar/Transição Redes Modulares de Alto Nível Redes Etiquetadas Sistemas de transição	Simulação Rápida Espaço de estados Espaço de estados Modular Verificação de modelos LTL	Unix Linux Windows Mac OS		Software/maria/
21	<b>Marigold</b>	Lugar/Transição Construtor de fluxo de dados	Editor Gráfico	Java	Gratuito	<a href="http://www-users.cs.york.ac.uk/~jwillans/VE_Research.html">http://www-users.cs.york.ac.uk/~jwillans/VE_Research.html</a>
22	<b>MISS-RdP</b>	Alto Nível Lugar/Transição Stochastic Petri Nets Com Tempo Modelos Híbridos	Editor Gráfico Jogador de Marcas Simulação Rápida Análise de desempenho Avançado	Sun Windows	Comercial (Desconto acadêmico)	<a href="http://www.ixi.fr/tools/pages/miss/fr.0.htm">http://www.ixi.fr/tools/pages/miss/fr.0.htm</a>
23	<b>The Model Checking Kit</b>	Alto Nível Lugar/Transição	Espaço de estados Esp. de est. Condensados Verificação de modelos CTL-/LTL Verificação de <i>Deadlock</i> Verificação de Alcançabilidade	Sun Linux	Gratuito	<a href="http://www.brainer.in.tum.de/gruppen/theorie/KIT/">http://www.brainer.in.tum.de/gruppen/theorie/KIT/</a>
24	<b>Nevod</b> Arcos inibidores	Redes de Petri com Jogador de Marcas Simulação Rápida	Editor Gráfico	MS DOS	Comercial	<a href="http://geocities.com/zsoftua">http://geocities.com/zsoftua</a>
25	<b>Opera</b>	Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida Espaço de estados Esp. Est. Condensados Invariantes de Lugar Invariantes de Transição Reduções nas Redes Análise estrutural	MS DOS	Comercial	<a href="http://geocities.com/zsoftua">http://geocities.com/zsoftua</a>

			Análise de desempenho Simples e Avançado Formato de intercambio de arquivos			
26	<b>PACE</b>	Alto Nível Lugar/Transição Stochastic Petri Nets Com Tempo Redes de Petri com Atributos	Editor Gráfico Jogador de Marcas Simulação Rápida Reduções nas Redes Lógica Borrosa Otimizações na Rede	Windows	Commercial (Desconto acadêmico)	<a href="http://www.ibepace.com/">http://www.ibepace.com/</a>
27	<b>PEID</b>	Lugar/Transição Com Tempo	Editor Gráfico	Sun Linux	Gratuito	<a href="http://www-dssz.informatik.tu-coitibus.de/~wwwdssz/">http://www-dssz.informatik.tu-coitibus.de/~wwwdssz/</a>
28	<b>PEP</b>	Alto Nível Lugar/Transição Com Tempo	Editor Gráfico Jogador de Marcas Espaço de estados Esp. Est. Condensados Invariantes de Lugar Invariantes de Transição Reduções nas Redes Análise estrutural Verificação dos Modelos Gerador de Redes Formato de intercambio de arquivos	Sun	Gratuito	<a href="http://parsys.informatik.uni-oldenburg.de/~pep">http://parsys.informatik.uni-oldenburg.de/~pep</a>
29	<b>PetriGen</b>	Lugar/Transição	Editor Gráfico Jogador de Marcas Síntese	Linux Windows	Gratuito	<a href="http://www.ec-ille.fr/tomnab">http://www.ec-ille.fr/tomnab</a>
30	<b>Petri Net Kernel</b>	Alto Nível Lugar/Transição Redes tipo <i>DAWN</i> definidas pelo usuário	Editor Gráfico Jogador de Marcas INA-pilot Formato de intercambio	Java	Gratuito	<a href="http://pnlk.sourceforge.net/">http://pnlk.sourceforge.net/</a>

31	<b>Petri Net Toolbox</b>	Lugar/Transição Stochastic Petri Nets Com Tempo Com Tempo	de arquivos Editor Gráfico Jogador de Marcas Simulação Rápida Espaço de estados Invariantes de Lugar Invariantes de Transição Análise estrutural Análise de desempenho Simples e Avançado Formato de intercambio de arquivos	Sum	Comercial (Desconto acadêmico)	<a href="http://www.ac.tuiasi.ro/pntool">http://www.ac.tuiasi.ro/pntool</a>
32	<b>PetriSim</b>	Alto Nível Lugar/Transição Com Tempo	Editor Gráfico Simulação Rápida	MS DOS	Gratuito	<a href="http://staff.um.edu.mt/jsk11/petrisim/index.html">http://staff.um.edu.mt/jsk11/petrisim/index.html</a>
33	<b>Platform Independent Petri Net Editor</b>	Lugar/Transição	Editor Gráfico Jogador de Marcas Simulação Rápida Espaço de estados Esp. Est. Condensados Invariantes de Lugar Invariantes de Transição Análise estrutural Ana. de desempenho simples Formato de intercambio de arquivos	Java	Gratuito	<a href="http://petri-net.sourceforge.net/">http://petri-net.sourceforge.net/</a>
34	<b>PNSim</b>	Lugar/Transição	Editor Gráfico Jogador de Marcas Análise estrutural Análise Simples	Windows Java	Gratuito	<a href="http://www.elyros.com/PNSim/">http://www.elyros.com/PNSim/</a>
35	<b>PNTalk</b>	Alto Nível	Editor Gráfico	Sum	Gratuito	<a href="http://www.fee.vutbr.cz/">http://www.fee.vutbr.cz/</a>

		Com Tempo	Jogador de Marcas Simulação Rápida Ana. de desempenho simples	Windows		~janousek/putalk/putalk.html
36	Poses++	Alto Nível	Simulação Rápida	Sun Linux Windows	Gratuito	<a href="http://www.gpc.de/">http://www.gpc.de/</a>
37	Predator	Lugar/Transição Stochastic Petri Nets Redes Hierarquicas (Subredes)	Editor Gráfico Invariantes de Lugar Invariantes de Transição Formato de intercambio de arquivos	Java	Gratuito	<a href="http://www.mark.wass.com/Petrinets">http://www.mark.wass.com /Petrinets</a>
38	PROD	Alto Nível Lugar/Transição	Espaço de estados Esp. Est. Condensados Verificação de Modelos <i>LTL</i> Verificação de Modelos <i>CTL</i>	Sun HP-Unix Linux Windows	Gratuito	<a href="http://www.tcs.hut.fi/Software/prod/">http://www.tcs.hut.fi/ Software/prod/</a>
39	Renew	Orientada a Objetos Alto Nível Lugar/Transição Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida Formato de intercambio de arquivos	Java	Gratuito	<a href="http://www.renew.de/">http://www.renew.de/</a>
40	Romeo	Com Tempo	Editor Gráfico Espaço de estados	Linux Mac OS Mac OS		<a href="http://www.irccyn.ec-nantes.fr/irccyn/d/em/equipes/TempsReel/logs">http://www.irccyn.ec-nantes.fr/ irccyn/d/em/equipes/ TempsReel/logs</a>
41	SEA	Alto Nível Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida	Sun	Gratuito	<a href="http://www.c-lab.de/sea/">http://www.c-lab.de/sea/</a>
42	SIPN-Editor	Redes de Petri Interpretadas	Editor Gráfico Geração de Código para <i>PLC</i> Tradução em código <i>SMV</i>	Java	Gratuito	<a href="http://www.eit.uni-kl.de/ltiz/ENGLISH/software/SIPNEditor.htm">http://www.eit.uni-kl.de/ltiz/ /ENGLISH/software /SIPNEditor.htm</a>
43	SimulaWorks	Lugar/Transição Redes de Petri Estocásticas	Editor Gráfico Jogador de Marcas	Windows	Gratuito	<a href="http://www.simulaworks.com/">http://www.simulaworks.com/</a>

		Com Tempo	Simulação Rápida				
44	<b>SPNP</b>	Alto Nível Stochastic Petri Nets	Análise de desempenho Avançado	Windows	Comercial (acad. Gratuito)	<a href="http://www.ee.duke.edu/~kst/">http://www.ee.duke.edu/~kst/</a>	
45	<b>StpnPlay</b>	Stochastic Petri Nets Com Tempo	Editor Gráfico Simulação Rápida Ana. de desempenho simples Formato de intercambio de arquivos	Windows	Gratuito	<a href="http://dce.felk.cvut.cz/capeki/StpnPlay">http://dce.felk.cvut.cz/capeki/StpnPlay</a>	
46	<b>SYROCO</b>	Alto Nível Com Tempo Instanização Dinâmica de Redes	Editor Gráfico Simulação Rápida Ana. de desempenho simples Formato de intercambio de arquivos	C++	Comercial (acad. Gratuito)	<a href="http://www.univ-tlse1.fr/ceiss/COOgene.html">http://www.univ-tlse1.fr/ceiss/COOgene.html</a>	
47	<b>TimeNET</b>	Alto Nível Lugar/Transição Redes de Petri Estocásticas Com Tempo	Editor Gráfico Jogador de Marcas Simulação Rápida Espaço de estados Invariantes de Lugar Análise estrutural Análise de desempenho Simples e Avançado Formato de intercambio de arquivos	Sun Linux	Comercial (acad. Gratuito)	<a href="http://pdv.cs.tu-berlin.de/~timenet/">http://pdv.cs.tu-berlin.de/~timenet/</a>	
48	<b>Tina</b>	Lugar/Transição Com Tempo Temporizadas	Editor Gráfico Espaço de estados Esp. Est. Condensados Invariantes de Lugar Invariantes de Transição Análise estrutural Espaço de Estado de Classes	Sun Linux Windows	Gratuito	<a href="http://www.laas.fr/tina">http://www.laas.fr/tina</a>	

49	Visual Object Net ++	Lugar/Transição Com Tempo Temporizadas Redes Híbridadas Redes de Petri Estocásticas	Editor Gráfico Jogador de Marcas Simulação Rápida Análise estrutural Ana. de desempenho simples Hierarquia Orientada a Objetos	Windows	Gratuito	<a href="http://www.systemtechnik.tu-ilmeneau.de/~drath/visual_E.htm">http://www.systemtechnik.tu-ilmeneau.de/~drath/visual_E.htm</a>
50	WebSPN	Redes de Petri Estocásticas	Editor Gráfico Jogador de Marcas Ana. de desempenho avançado	Java	Gratuito	<a href="http://ing-inf.unime.it/webspn">http://ing-inf.unime.it/webspn</a>
51	WINSIM	Redes Simples Redes Estendidas	Simulação Rápida	Windows	Gratuito	<a href="ftp://cmpe.emu.edu.tr/SimSystem">ftp://cmpe.emu.edu.tr/SimSystem</a>

**Nomenclatura:**

Ana. de desempenho avançado      Análise de desempenho avançado  
 Ana. de desempenho simples      Análise de desempenho simples  
 Esp. de est. Condensados      Espaço de estados Condensados

- As características adicionais reúnem aspectos desejáveis em um ambiente computacional destinado à modelagem e projeto de Redes de Petri.
1. Evitar o problema do contato e/ou confusão na execução da rede.
  2. Alta portabilidade do código fonte, permitindo obtenção de ambientes para execução em diferentes sistemas operacionais.

### 5.3 Características Adicionais

1. A rede é Temporalizada.
2. Uso de hierarquia: O sistema deve permitir o projeto de redes hierárquicas.
3. Recursos para permitir o refinamento dos elementos hierárquicos: o sistema deve possuir recursos para garantir a consistência do processo de refinamento das redes.
4. Uso do conceito de Orientação a Objetos.

A seguir enumeram-se os requisitos derivados da formulação teórica da Rede *GHEneSys* :

### 5.2 Características Específicas

CARACTERÍSTICA		SISTEMAS
Refinamento/Hierarquia	ExSpect INA Maria Opera Pep Predator Visual Object Net ++	Tina Visual Object Net ++
Temporalizadas		

Tabela 5.2: Características dos Sistemas de Desenvolvimento Disponíveis



Seja a *GHENeSys* uma rede hierárquica sua dinâmica é diferente da rede de Petri Ordinária. O primeiro aspecto a se considerar é a necessidade de se estabelecer as relações de subordinação dos MacroElementos que estão associados a redes, mediante a construção de uma árvore hierárquica. Isto é necessário por que a determinação da habilitação das transições, da marcação dos elementos passivos não PseudoBox, da detecção de conflitos e da aplicação da equação de estado realiza-se mediante a técnica de análise em profundidade (*depth first*), na qual o nó que possui um maior nível de profundidade, partindo do nó raiz, é avaliado primeiro. O algoritmo de Hierarquização da rede é apresentado na figura 5.1

### 5.5.1 Algoritmo de Hierarquização

## 5.5 Aspectos Computacionais

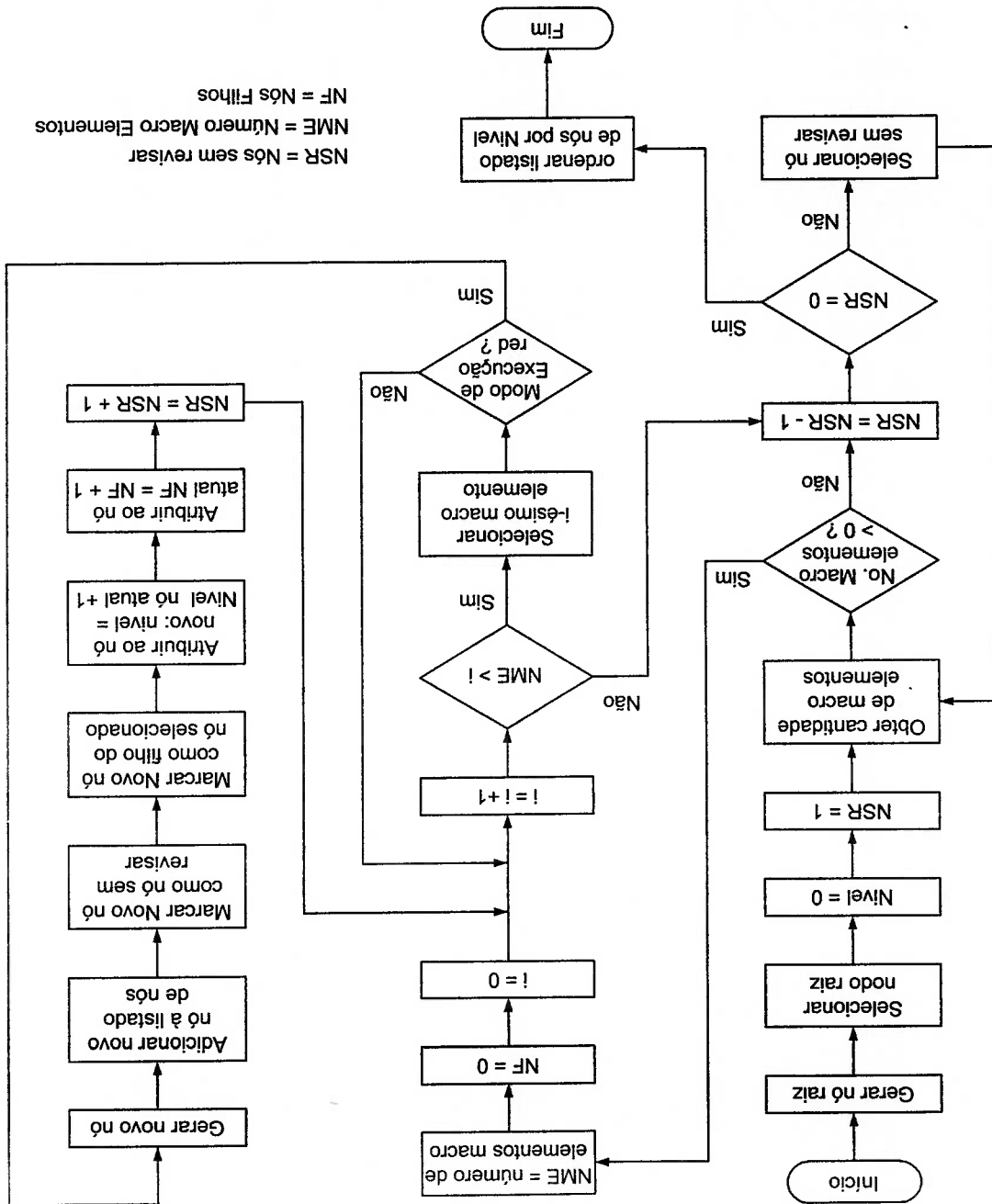
A seleção do ambiente de desenvolvimento para a construção do software foi feita após uma avaliação quantitativa das características relativas ao compilador. O produto e ao desenvolvedor. Os compiladores e aspectos avaliados apresentam-se na tabela A.2. O compilador selecionado foi o Delphi.

## 5.4 Critérios para a Seleção do Compilador

3. Possuir um mecanismo que permita administrar as diferentes subredes que compõem um projeto de forma simples, respeitando sua estrutura hierárquica.
4. Detectar os conflitos de maneira que o usuário identifique claramente quando ocorre um conflito, e possa intervir ativamente na resolução do mesmo.
5. Permitir que os elementos de uma rede *GHENeSys* se comuniquem mediante interface com outros aplicativos e/ou sistema operacional.
6. Possuir restrições no editor e gerenciador de projetos, de modo que as redes desenvolvidas estejam corretas sintaticamente por construção. Por exemplo o sistema não aceita a criação de arcos inibidores entre um lugar simples e uma transição.

Para fazer referência aos diferentes nós, utiliza-se uma sintaxe derivada da estrutura da árvore, onde se percorre toda a rota desde o nó raiz até o nó destino, como se apresenta na figura 5.2. As linhas pontilhadas indicam que os MacroElementos são limites, isto é, não possuem uma rede associada.

Figura 5.1: Algoritmo de Hierarquização da Rede *CHENesys*



Para exemplificar a execução de uma Rede *GHENeSys* utiliza-se a rede apresentada na figura 5.5, contendo dois elementos MacroAtividade e um elemento MacroBox. Como visto no capítulo anterior existem dois modos de operação na rede *GHENeSys*: o modo abstrato temporizado e o modo de rede-associada. No primeiro caso, as MacroAtividades quando chamadas esperam tantos ciclos de tempo quanto o armazenado no seu atributo de temporização. Os MacroBox operam da mesma forma ou eventualmente chamando um dos seus métodos. No exemplo uma das MacroAtividades é executada em modo de rede-associada.

## 5.6 Exemplo da Dinâmica da Rede *GHENeSys*

A execução da Rede *GHENeSys* é apresentada no algoritmo da figura 5.1.

### 5.5.2 Algoritmo de Execução da Rede *GHENeSys*

Figura 5.2: Sintaxe para Fazer Referência aos Nós da Rede *GHENeSys*

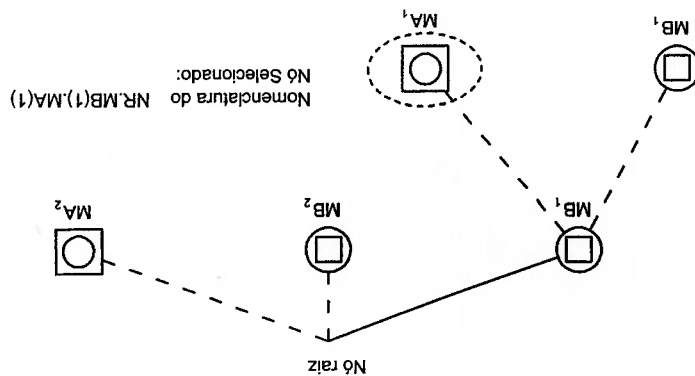
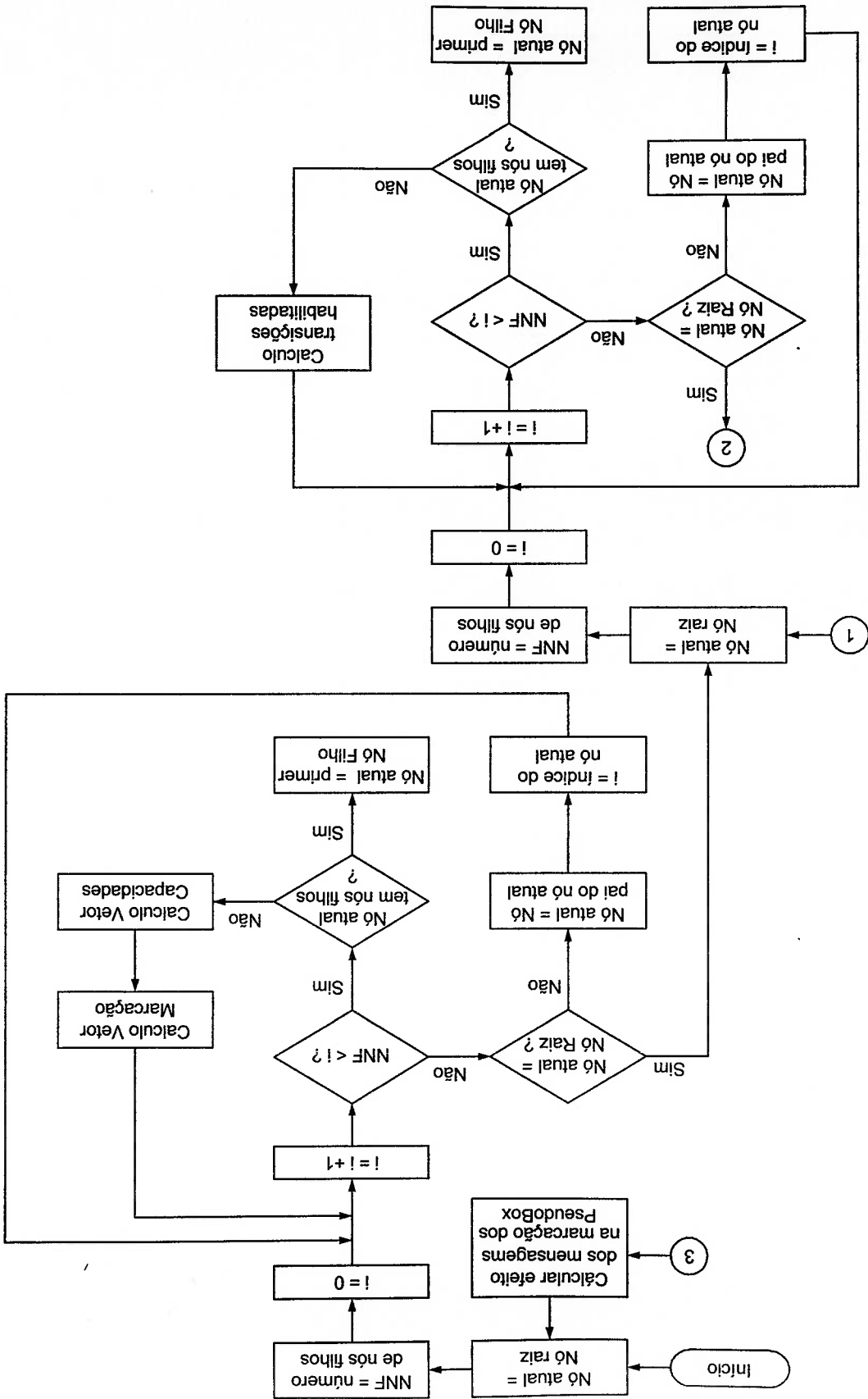
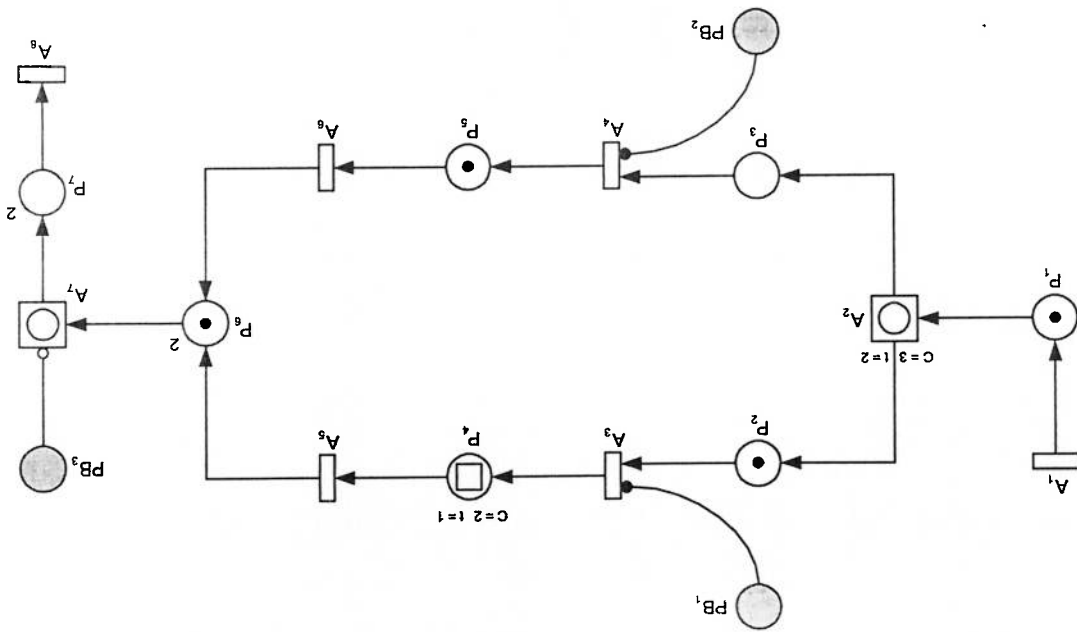


Figura 5.3: Algoritmo de Execução da Rede *GHENeSys* (a)







(a) Rede principal

(b) Subrede associada a MacroAtividade  $NR.MA_2$   
(Elemento Ativo  $A_7$  da rede principal)

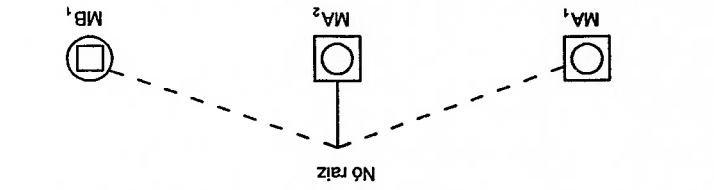


Figura 5.5: Exemplo de Dinâmica da Rede *CHENesys*

Figura 5.6: Árvore de Hierarquia para o Exemplo da figura 5.5

## 5.6.2 Matrizes do Sistema

Matrizes associadas ao nó NR.Ma(2)

$$A^+ = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 \\ A_2 & 0 & 0 & 1 & 0 & 0 \\ A_3 & 0 & 0 & 0 & 1 & 0 \\ A_4 & 0 & 0 & 0 & 0 & 1 \\ A_5 & 0 & 0 & 0 & 0 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad A^- = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 \\ A_2 & 1 & 0 & 0 & 0 & 0 \\ A_3 & 0 & 1 & 0 & 0 & 0 \\ A_4 & 0 & 0 & 1 & 0 & 0 \\ A_5 & 0 & 0 & 0 & 1 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

A matriz de incidência do nó NR.Ma(2) é portanto:

$$A_g = \begin{bmatrix} A_1 & 1 & 0 & 0 & 0 & 0 \\ A_2 & -1 & 0 & 1 & 0 & 0 \\ A_3 & 0 & -1 & 0 & 1 & 0 \\ A_4 & 0 & 0 & -1 & 0 & 1 \\ A_5 & 0 & 0 & 0 & -1 & 1 \\ A_6 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}$$

Matrizes do nó raiz

$$A^+ = \begin{bmatrix} A_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_2 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ A_3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ A_4 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ A_5 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ A_7 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ A_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad A^- = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_3 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ A_4 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ A_5 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ A_7 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ A_8 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

A matriz de incidência portanto é:

$$H^e = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_4 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ A_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad H^d = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A_8 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}$$

Segundo 4.7

### 5.6.5 Cálculo das Transições Habilitadas

$$U_{NR} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}^T$$

Marcação do Nó raiz

$$U_{NR.Ma(2)} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}^T$$

Marcação de Nó NR.Ma(2)

Segundo 5.4

### 5.6.4 Vetores de Marcação

$$C_{NR} = \begin{bmatrix} 1 & 1 & 1 & 2 & 1 & 2 & 2 \\ 1 & 1 & 1 & 2 & 1 & 2 & 2 \end{bmatrix}^T$$

Capacidade do Nó raiz

$$C_{NR.Ma(2)} = \begin{bmatrix} 1 & 1 & 2 & 2 & 1 \\ 1 & 1 & 2 & 2 & 1 \end{bmatrix}^T$$

Capacidade do Nó NR.Ma(2)

### 5.6.3 Vetores de Capacidade

$$A_g = \begin{array}{c} A_8 \\ A_7 \\ A_6 \\ A_5 \\ A_4 \\ A_3 \\ A_2 \\ A_1 \end{array} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{array}{c} P_1^1 \\ P_2^1 \\ P_3^1 \\ P_4^1 \\ P_5^1 \\ P_6^1 \\ P_7^1 \\ P_8^1 \\ P_9^1 \\ P_{10}^1 \end{array} \left| \begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right. \begin{array}{c} P_1^2 \\ P_2^2 \\ P_3^2 \\ P_4^2 \\ P_5^2 \\ P_6^2 \\ P_7^2 \\ P_8^2 \\ P_9^2 \\ P_{10}^2 \end{array}$$



Transições habilitadas No raiz

$$\begin{array}{cc}
 \begin{array}{c} \left[ \begin{array}{c} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_3 = 1 \end{array} & \geq & \begin{array}{c} \left[ \begin{array}{c} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_3 = 1 \end{array} \\
 \\
 \begin{array}{c} \left[ \begin{array}{c} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{array} \right] \\ \leftarrow h_2 = 1 \end{array} & \geq & \begin{array}{c} \left[ \begin{array}{c} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{array} \right] \\ \leftarrow h_2 = 1 \end{array} \\
 \\
 \begin{array}{c} \left[ \begin{array}{c} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_4 = 0 \end{array} & \neq & \begin{array}{c} \left[ \begin{array}{c} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_4 = 0 \end{array}
 \end{array}$$

Isto é:

$$H = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$$

$$\begin{array}{cc}
 \begin{array}{c} \left[ \begin{array}{c} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_5 = 0 \end{array} & \neq & \begin{array}{c} \left[ \begin{array}{c} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_5 = 0 \end{array} \\
 \\
 \begin{array}{c} \left[ \begin{array}{c} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_3 = 0 \end{array} & \neq & \begin{array}{c} \left[ \begin{array}{c} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_3 = 0 \end{array} \\
 \\
 \begin{array}{c} \left[ \begin{array}{c} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_4 = 1 \end{array} & \geq & \begin{array}{c} \left[ \begin{array}{c} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_4 = 1 \end{array} \\
 \\
 \begin{array}{c} \left[ \begin{array}{c} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_2 = 0 \end{array} & \neq & \begin{array}{c} \left[ \begin{array}{c} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] \\ \leftarrow h_2 = 0 \end{array}
 \end{array}$$

Transições habilitadas do Nó NR.Ma(2)

### 5.6.6 Cálculo do Efeito da Presença de Arcos Habilitados e Inibidores

Segundo o algoritmo apresentado em 4.7.5.2 no ciclo de avaliação  $k = 1$ :

$$T_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Arcos inibidores Segundo 4.7.5.2

$$H^p \cdot T_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$$

Isto é:

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}^T$$

Isto é:

$$\begin{array}{l} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \geq \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \rightarrow h_7 = 1 \\ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \not\geq \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \rightarrow h_8 = 0 \end{array}$$
  

$$\begin{array}{l} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \not\geq \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \rightarrow h_5 = 0 \\ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \geq \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \rightarrow h_6 = 1 \end{array}$$

conflitos No NR.Ma(2) A avaliação dos conflitos na rede é feita segundo equação 4.13

### 5.6.7 Cálculo de Conflitos

$$H'_1 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

é:

Assim o efeito dos arcos inibidores e habilitadores sobre o vetor de transições

$$(H^+)_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Isto é:

$$= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} = H \cdot (I - T) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

A avaliação segundo 5.4

### Arcos Habilitadores

$$(H^-)_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_2^z = U_1 + (D \cdot A_T^g) \cdot (H_1^T)$$

**Conflitos do N6 Raiz: Usando 4.13**

Isto é a Subrede associada ao N6 NR[3]. MA não apresenta conflitos em  $k = 1$ .

$$U_2^z = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^z$$

Portanto:

$$\begin{aligned} (1 \geq 0) \vee (1 \leq 0) &\rightarrow u_1^z = 0 \\ (1 \geq 0) \vee (1 \leq 2) &\rightarrow u_2^z = 0 \\ (0 \geq 0) \vee (0 \leq 0) &\rightarrow u_3^z = 0 \\ (0 \geq 0) \vee (0 \leq 0) &\rightarrow u_4^z = 0 \\ (1 \geq 0) \vee (1 \leq 0) &\rightarrow u_5^z = 0 \end{aligned}$$

Aplicando a equação 4.13:

$$U_2^z = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ -1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 2 \\ 1 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



### 5.6.8 Aplicação da Equação de Estado

Nó NR.Ma(2)

Assumindo que todas as transições habilitadas dispararam, isto é:  $(F'_2) = (H'_2)$

$$U'_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{pmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \end{pmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Isto é a rede do nó raiz não apresenta conflitos em  $k = 1$ .

$$U'_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Portanto:

- $(1 \geq 0) \wedge (1 \leq 1) \rightarrow u'_1 = 0$
- $(1 \geq 0) \wedge (1 \leq 1) \rightarrow u'_2 = 0$
- $(1 \geq 0) \wedge (1 \leq 1) \rightarrow u'_3 = 0$
- $(1 \geq 0) \wedge (1 \leq 2) \rightarrow u'_4 = 0$
- $(0 \geq 0) \wedge (0 \leq 1) \rightarrow u'_5 = 0$
- $(1 \geq 0) \wedge (1 \leq 2) \rightarrow u'_6 = 0$
- $(1 \geq 0) \wedge (1 \leq 2) \rightarrow u'_7 = 0$

$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Aplicando 4.13:



faz-se necessário definir um projeto (no contexto da *GHENESYS*) como uma estrutura mediante a qual é possível gerenciar as redes que compõem o sistema, sua hierarquia e o intercâmbio de mensagens. Para o projeto e para as redes definem-se tipos de arquivo diferentes.

Um arquivo de projeto é um arquivo com extensão *.gp* (*GHENESYS Project*). Este é um arquivo compactado em formato *ZIP* que armazena os arquivos das redes e o conjunto de arquivos que contêm características do sistema. tais como: a árvore de hierarquia, lista de redes que integram o projeto, os arquivos das redes, estrutura de mensagens, etc. A estrutura interna de um arquivo típico de projeto se apresenta na figura 5.8.

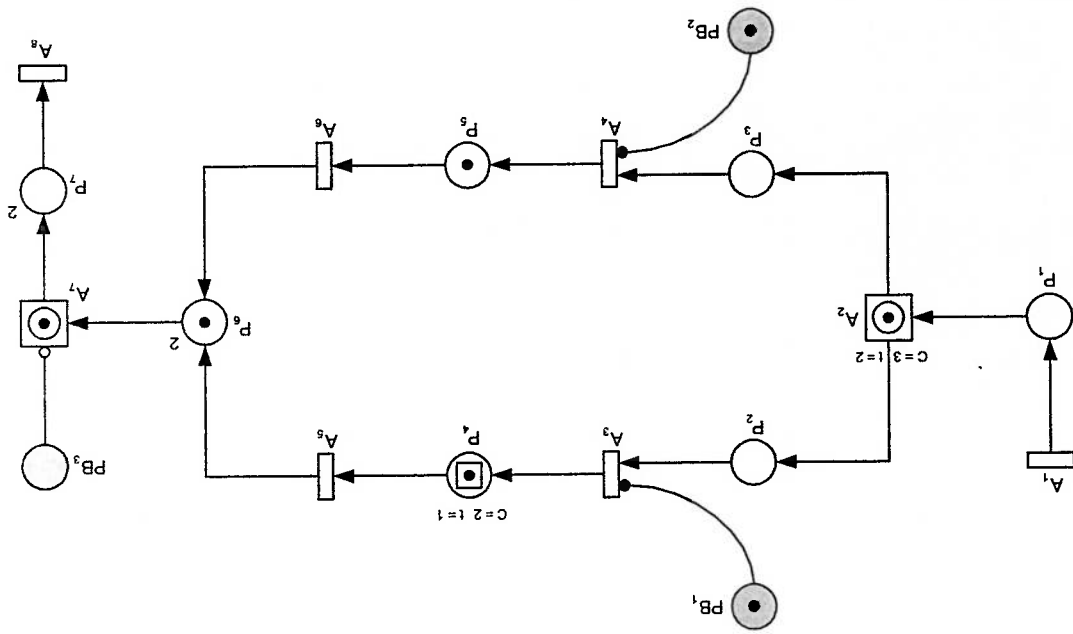


Figura 5.7: Resultado Exemplo da Rede *GHENESYS*



1. **Validade:** Uma rede é válida se não tem elementos isolados. classificação implementados na *GHENESYS* são: é determinada de acordo as regras estabelecidas na seção 4.3. Os dois tipos de condição básica para a construção hierárquica do sistema, Esta compatibilidade A classificação da compatibilidade das redes com os MacroElementos é uma ao projeto.

A modificação deve ser realizada em modo *Design* para depois integrá-la de novo intercambio de mensagens. Assim se é necessário alterar uma rede de um projeto, projeto não é desejável, pois pode afetar a compatibilidade e/ou a estrutura de croBoxes e/ou MacroActividades). A modificação das redes que compõem um ponde à atribuição de redes compatíveis encapsuladas em MacroElementos (Ma- A estrutura da *GHENesys* utiliza um refinamento hierárquico que corres- pode executar um sistema *GHENesys*.

No primeiro modo o usuário pode construir as redes. No segundo, pode adicionar e/ou remover as redes a um projeto, construir a árvore de hierarquia e relacionar os mensagens aos Pseudobox. Somente no modo de Gestão do Projeto o usuário *GHENESYS* pode ser utilizada de dois modos: *Design* e Gestão do Projeto.

A estrutura interna de um arquivo típico apresenta-se na figura 5.9 Um arquivo de rede tem extensão *.gpn* (*GHENESYS Petri Net*), nele são armazenadas as informações próprias de uma rede, tais como a posição dos ele- mento na tela, tipos de elemento, arcos que unem os elementos, tipo de arcos, etc.

Figura 5.8: Estrutura interna de um Arquivo de Projeto na *GHENESYS*

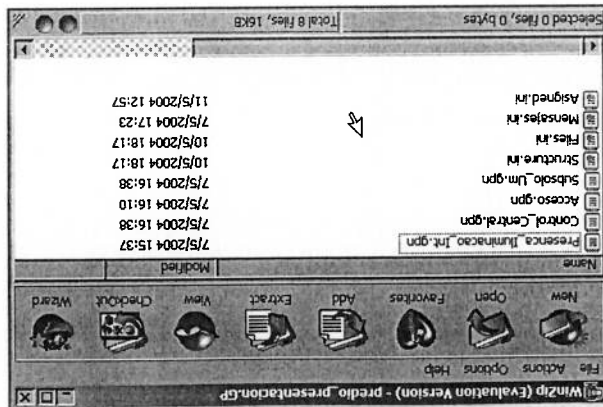


Figura 5.9: Estrutura de um Arquivo de Rede na *GHENESYS*

```

[GPNF]
PaperType= A4
Orientation= polandscape

[PIACES]
Left= 156
Top= 231
Show Name= True
Caption= Presenca
Show Caption= True
Capacity= 1
ShowCapacity= True
Initial Tokens= 0
Current Tokens= 0

Left= 157
Top= 137
Show Name= True
Caption= Mensagem Reportada
Show Caption= True
Capacity= 1
ShowCapacity= True
Initial Tokens= 0
Current Tokens= 0

Left= 161
Top= 40
Show Name= True
Caption= Não Presenca
Show Caption= True
Capacity= 1
ShowCapacity= True
Initial Tokens= 0
Current Tokens= 0

[MESSAGES]

[TRANSITIONS]
Left= 44
Top= 229
Show Name= True
Caption= Gerar Mensagem
Show Caption= True
Position= Vertical

[ARCS]
1 1 0 2 0
0 2 1 3 0
1 3 0 1 0
0 1 1 2 0
0 1 1 1 0
1 0 0 1 0
1 2 0 0 0
0 0 1 0 0

[MACROACTIVITIES]
[MACROBOXES]
[PSEUDOBBOXES]
Left= 264
Top= 39
Show Name= True
Caption= Gerar Mensagem
Show Caption= True
Capacity= 1
ShowCapacity= True
Initial Tokens= 0
Current Tokens= 0

Left= 262
Top= 233
Show Name= True
Caption= Não Detegão Presenca
Show Caption= True
Position= Horizontal

Left= 262
Top= 233
Show Name= True
Caption= Detegão Presenca
Show Caption= True
Position= Horizontal

Time= 0

```

## 2. Compatibilidade: A *GHENESYS* detecta se a rede que introduzida num

projeto é compatível ou incompatível. Se é compatível, determina com qual tipo de elemento, MacroBox ou MacroAtividade.

### 5.7.1 Interface Gráfica da *GHENESYS*

A *GHENESYS* foi desenvolvida procurando uma interface comum às aplicações estilo *Windows*. A figura 5.10 apresenta a janela principal do software apresentando o desenho de uma rede. Para cada função principal da *GHENESYS* são associados um ícone e um menu, que por sua vez são agrupados em módulos de acordo com as funções gerais que desempenham. A figura 5.11 apresenta os ícones correspondentes a cada um dos módulos da *GHENESYS*.

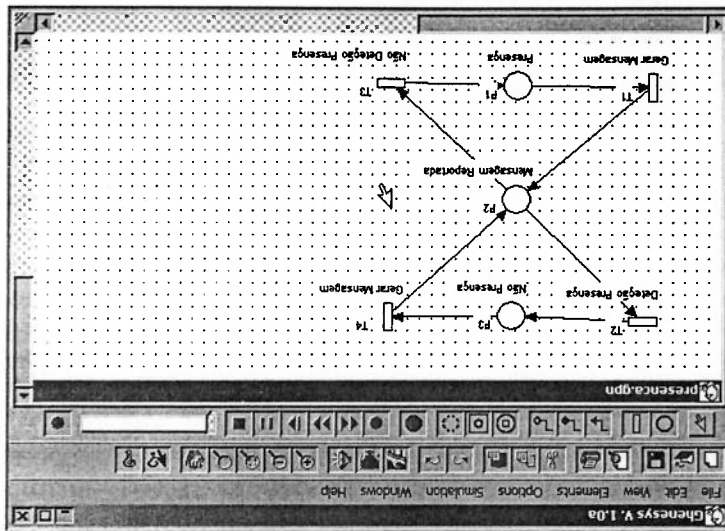


Figura 5.10: Janela Principal da *GHENESYS*

O módulo de Funções *CAD* (*Computer Aided Design*) e algumas das funções do módulo de Sistema Operacional não foram desenvolvidas, pois a formalização da *GHENESYS* e a implementação dos requisitos levantados ( ver secções 5.3 e 5.2) foram priorizados. Os módulos desenvolvidos são apresentados a seguir.

### 5.7.1.1 Módulo de Sistema Operacional

É a interface com o sistema operacional que permite criar, abrir, fechar e/ou salvar arquivos de Projeto e de Rede. Os menus e ícones deste módulo são apresentados na figura 5.12

Este módulo reúne os elementos requeridos para o desenho das redes. Ao seleccionar o menu ou o ícone correspondente ao elemento a *CHENESYS* permite criar

### 5.7.1.2 Módulo de Desenho

Figura 5.12: Menu e Barra de Ícones do Módulo Sistema Operacional

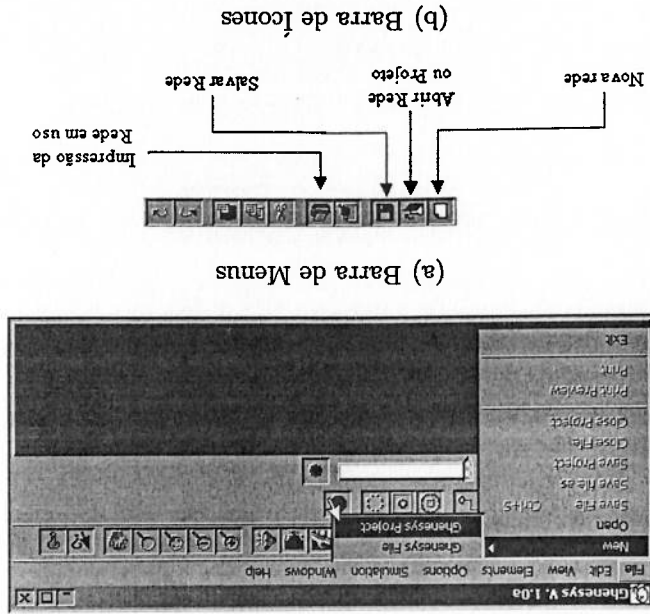
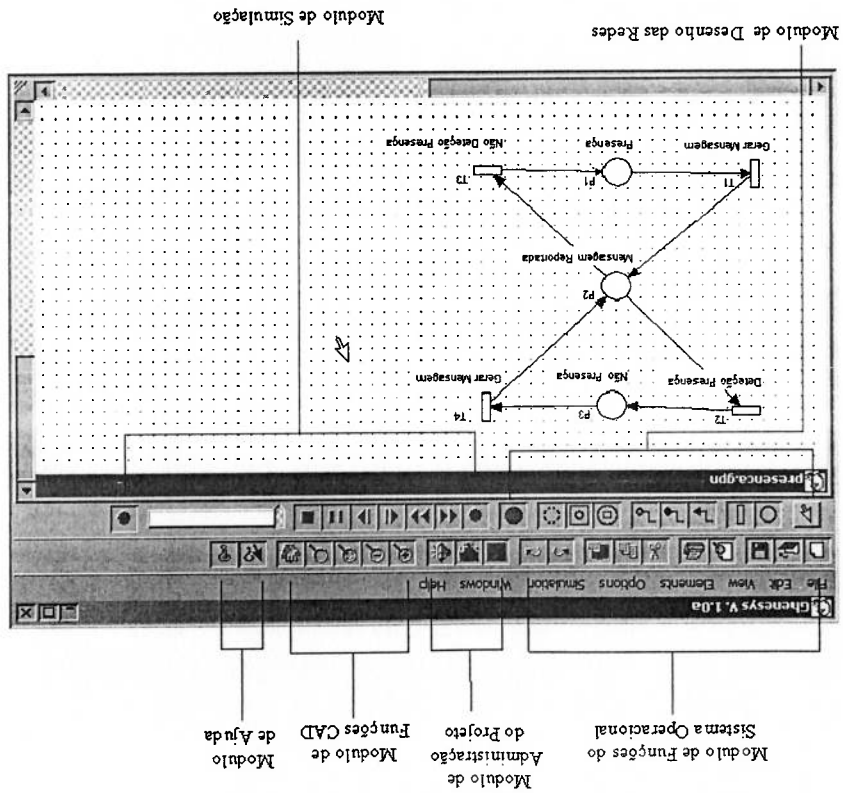
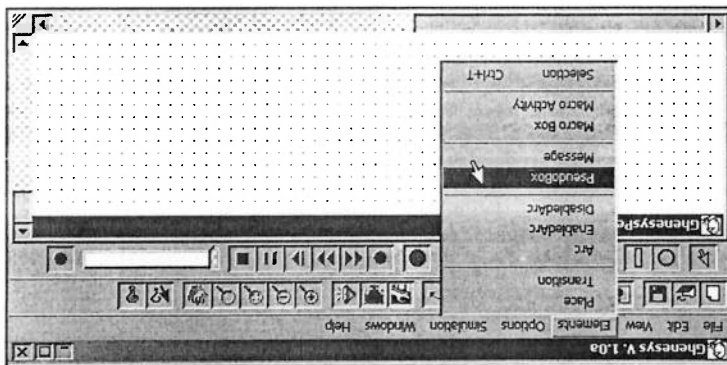


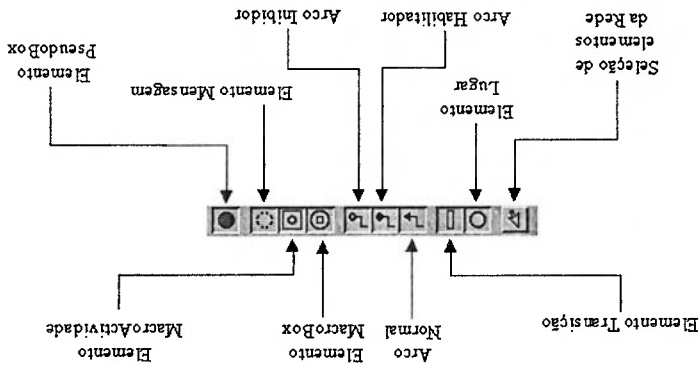
Figura 5.11: Ícones correspondentes aos Módulos da *CHENESYS*



na janela de edição uma instância do elemento escolhido. Os menus e ícones deste módulo são apresentados na figura 5.13, A barra de ícones e os menus de desenho estão habilitados unicamente em modo *Design*.



(a) Barra de Menus



(b) Barra de Ícones

Figura 5.13: Menu e Barra de Ícones do Módulo Desenho

Para desenhar um arco entre dois elementos escolhe-se a opção no menu ou ícone do tipo de arco desejado. Para tragar o arco seleciona-se o elemento de origem e movimentase o mouse com o botão pressionado, até o elemento final onde o botão é liberado e o arco criado. No caso que entre os dois elementos selecionados não seja possível criar um arco a *CHENESYS* apresentara um aviso explicativo.

Quando o elemento *Selection* é escolhido é possível selecionar os elementos para alterar suas propriedades, movimentá-los e ainda apagá-os. A figura 5.14 ilustra a movimentação dos elementos e a figura 5.15 apresenta a maneira de mudar as propriedades dos elementos de uma dada rede.

Este módulo é utilizado para a gestão do projeto, e contém os métodos para construção e modificação da Árvore Hierárquica e ligar as mensagens aos PseudoBox. Além disso, em versões futuras do software planeja-se incluir neste módulo a modificação do formato de espaço de edição da rede e o sistema de resolução, que na versão apresentada é por *default* priorizada as transições. A figura 5.16 apresenta os menus e ícones deste módulo.

### 5.7.1.3 Módulo de Administração do Projeto

Figura 5.15: Mudança de Propriedades dos Elementos na *CHENESYS*

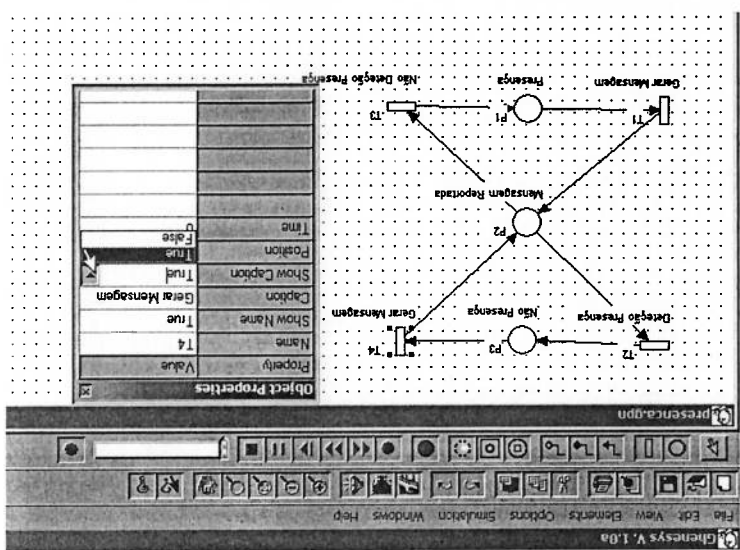
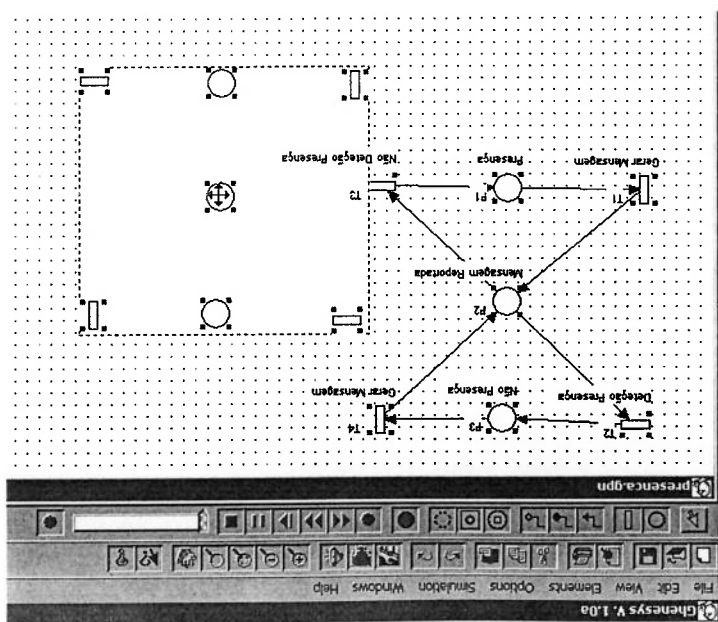
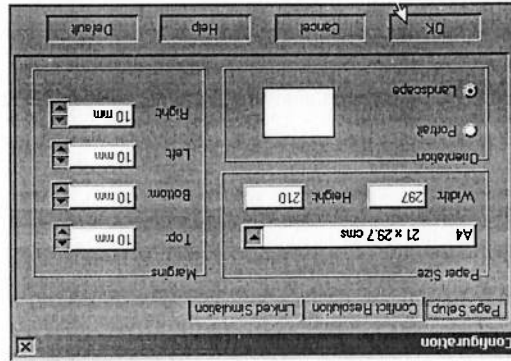


Figura 5.14: Movimentação de elementos na *CHENESYS*



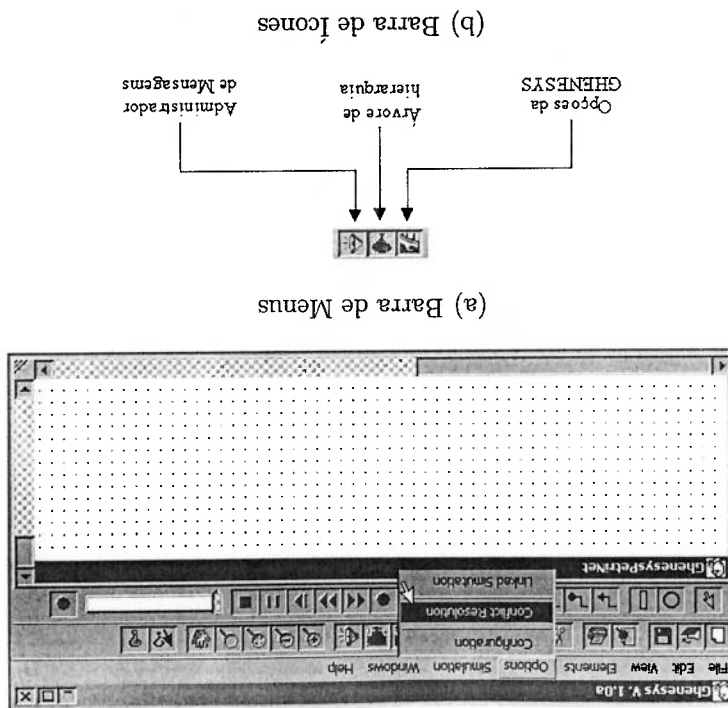
O ícone e o menu *Arvore de Hierarquia* apresentam a janela de atribuição de arquivos ao projeto, e permite a construção da árvore de hierarquia. Além disso possibilita observar a classificação das redes que compõem o projeto. A janela apresentada ao selecionar esta opção é ilustrada na figura 5.18.

Figura 5.17: Janela de Opções da *GHENESYS*



O ícone de opções apresenta a janela de opções do sistema, enquanto que o menu *Options* permite entrar diretamente no submódulo de opções desejado. A figura 5.17 apresenta a janela de opções.

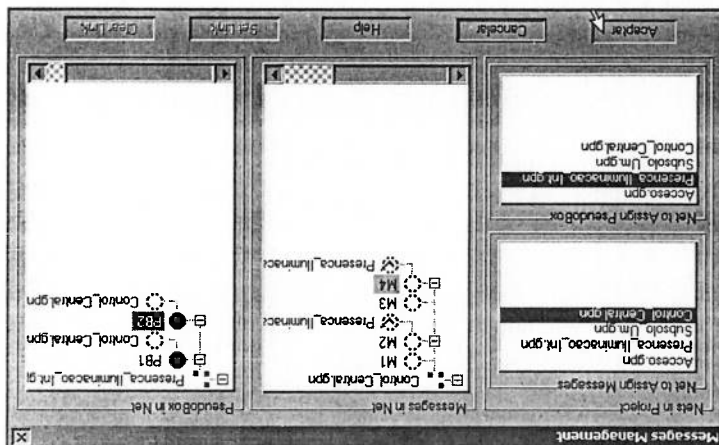
Figura 5.16: Menu e Barra de Ícones do Módulo Administração do Projeto



A *GHENESYS* permite a execução como um jogador de marcas tradicional, mas conta com a possibilidade de funcionar como um controlador acoplada a algum outro sistema computacional. Os menus e ícones deste módulo são apresentados na figura 5.20.

#### 5.7.1.4 Módulo de Execução

Figura 5.19: Janela de Gestão de mensagens da *GHENESYS*



O ícone e o menu *Administrador das Mensagens* apresentam a janela de gestão das mensagens, na qual é possível vincular ou desvincular um conjunto de mensagens a cada *Pseudobox*. No caso de uma rede ser apagada da árvore de hierarquia, as mensagens atribuídas a seus *Pseudobox* serem apagados de forma automática, igualmente as mensagens que a rede contém não estarão mas na estrutura de mensagens do sistema. A figura 5.19 apresenta a janela de gestão das mensagens.

Figura 5.18: Janela de Estrutura da *GHENESYS*

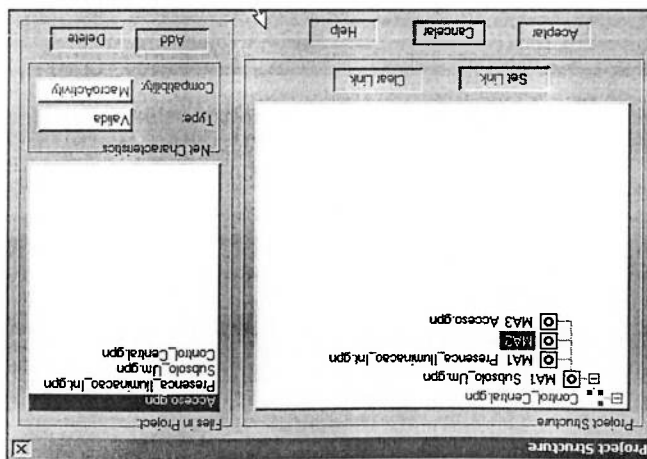
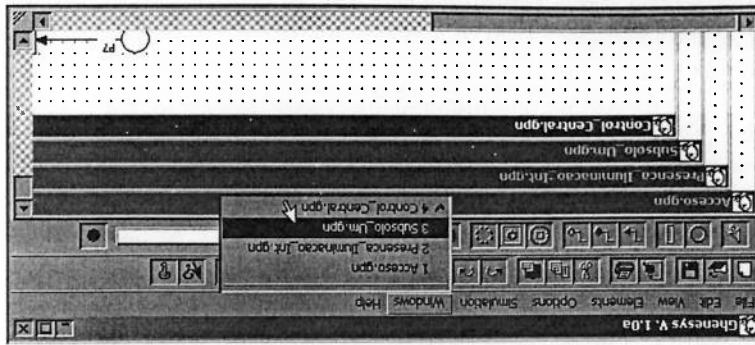


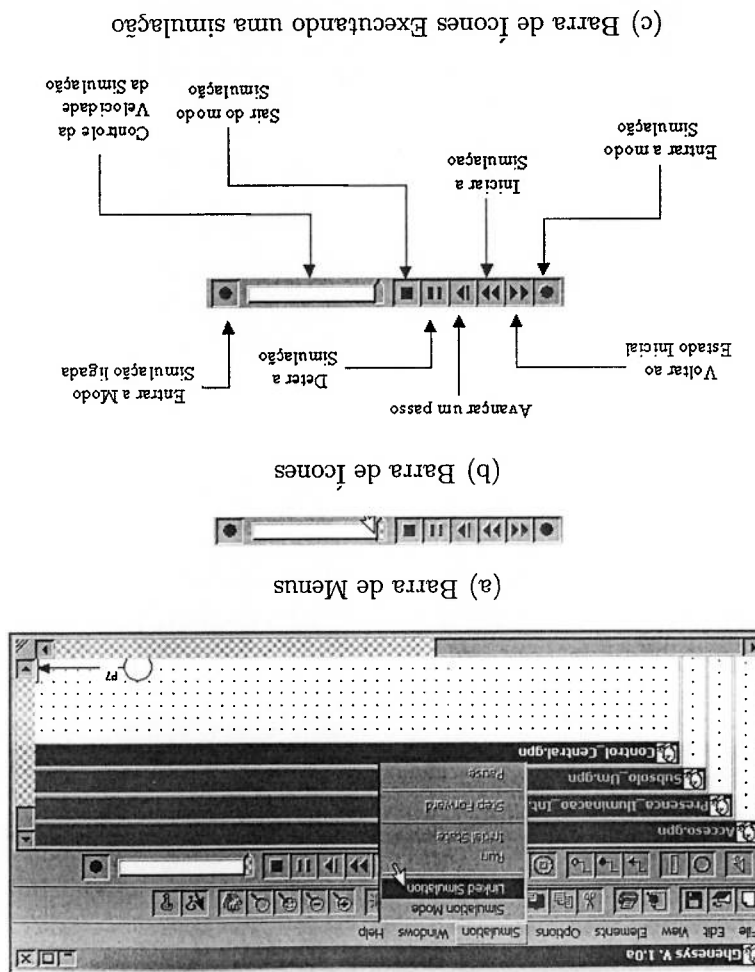


Figura 5.21: Menu de Seleção das Instâncias de um Projeto CHENESYS



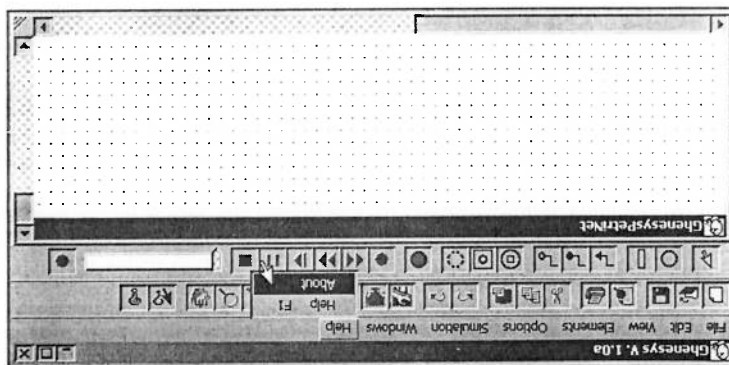
No modo execução cada uma das instâncias do Projeto é apresentada em uma janela, o menu *windows* pode ser utilizado para alternar entre as diferentes janelas do projeto, como se ilustra na figura 5.21.

Figura 5.20: Menus e Barra de Ícones do Módulo Administração do Projeto

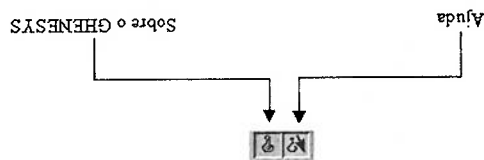


Este módulo reúne o sistema de ajuda da *GHENESYS*, a figura 5.22 apresenta os menus e ícones deste módulo.

### 5.7.1.5 Módulo de Ajuda



(a) Barra de Menus



(b) Barra de Ícones

Figura 5.22: Menu e Barra de Ícones do Módulo Administração do Projeto

O ícone *ajuda* chama o sistema de ajuda e o ícone *Sobre o GHENESYS* ) despijea a janela que apresenta o tipo de licença e os créditos do software, como apresentado na figura 5.23



Figura 5.23: Janela *About da GHENESYS*

## 6. ESTUDO DE CASO

O estudo de caso apresentado neste trabalho insere-se na área da automação predial, e consiste do projeto de um sistema de controle para um prédio automatizado utilizando a rede *GHENeSys*. O modelo de controle é validado utilizando o jogador de marcas desenvolvido acoplado a uma maquete em realidade virtual do prédio.

O objeto de estudo deste trabalho é o controle de um prédio automatizado composto por três andares. O prédio é considerado como uma planta<sup>1</sup> em termos da utilização pelos usuários, a aplicação da metodologia de design do sistema de controle com a *GHENeSys* se apresenta a continuação.

### 6.1 Controle de Sistemas a Eventos Discretos com a Rede *GHENeSys*

O sistema de controle com a rede *GHENeSys* baseia-se na possibilidade de inibir e/ou habilitar elementos ativos mediante a marcação ou demarcação dos elementos PseudoBox. (Ver seções 4.4 e 4.7.5.2) Os PseudoBox representam condições externas ao sistema.

## 6.2 Identificação do Sistema

### 6.2.1 Descrição do Prédio

As plantas dos andares do prédio estão apresentadas nas plantas A4-01, A4-02 e A4-03, uma composição tridimensional do prédio apresenta-se na planta A4-04.

### 6.2.2 Políticas de Utilização do Prédio

O prédio deve ser controlado seguindo-se um conjunto de regras heurísticas enunciadas a seguir:

- Na presença de usuários, quando o sistema de iluminação detecta a necessidade de aumentar a iluminação presente, deve fornecer-la, salvo quando

<sup>1</sup>No contexto da teoria de controle

o prédio não deva ter usuários presentes na área de ação do sensor de Presença. Neste caso deve-se acionar o sistema de alarmes. Os sistemas de Presença e iluminação têm funções diferentes para a área interna e a área externa do prédio: os sistemas de iluminação nas áreas externas não devem ser desligados, não devem ter sua operação manual desabilitada e não geram alarmes.

- No caso de presença de usuários em áreas do prédio, nos horários proibidos, os sistemas de acesso não permitem o ingresso ou saída dos usuários e o sistema de alarmes deve ser acionado. Além disso, o sistema de iluminação é desligado, não podendo ser ligado de maneira manual.

• Na presença de fumaça no prédio, o sistema de Incêndios deve determinar se trata-se de um incêndio, observando um certo tempo antes de acionar os alarmes e demais sistemas. Com o objetivo de economizar energia, os sistemas de HVAC são desligados depois de um tempo predeterminado a partir da não detecção de presença em suas áreas de influência.

## 6.3 Decomposição do Sistema

### 6.3.1 · Objetos do Prédio

Os objetos identificados no prédio, classificam-se de acordo com a função que apresentam. Assim os objetos situados no prédio são:

- Corredor
- Garagem
- Escritório
- Banheiro
- Bodega
- Escadas
- Portas

Os elementos de controle relacionam sensores, actuadores, regras heurísticas e o tipo de área do prédio para realizar o controle de utilização do mesmo. Assim observar-se que é possível a agregação de elementos constituindo elementos de controle. A tabela 6.1 apresenta os elementos de controle do prédio e a descrição da sua função.

### 6.3.3 Objetos Controladores

- Sensor de Presença
- Sensor de Fumaça
- Sistema de Iluminação
- Sistema de acesso
- HVAC

Os objetos identificados nos dispositivos existentes no prédio são:

### 6.3.2 Dispositivos de Automação do Prédio

A utilização do prédio é regulada com base em um conjunto de dispositivos que permitem reagir às diferentes situações que podem se apresentar. Assim, o controle baseia-se nas regras heurísticas provenientes das políticas de utilização do prédio, implementando restrições de segurança e otimização de recursos. a colocação dos dispositivos do prédio apresentam-se nas plantas A4-05, A4-06, A4-07. O modelo de utilização do prédio mostra-se na planta A2-01. As áreas as quais se faz referência neste modelo estão ilustradas nas plantas A4-08, A4-09, A4-10.

- Salão

Tabela 6.1: Descrição dos Elementos de Controle

<b>OBJETO DE CONTROLE</b>	<b>DESCRIÇÃO</b>
Presença-Illuminação-Interior	Permite o controle de espaços como banheiros, bodegas e corredores onde a detecção de presença é requisitada para ter ligada a iluminação só quando se tem presença, economizando energia; aplicar as políticas de segurança gerando alarmes quando se detecta presença em horários não autorizados e impossibilitar a ligação da iluminação manual quando existem alarmes no sistema.
Presença-Illuminação-Exterior	Permite a iluminação de áreas exteriores como garagem e no exterior dos sistemas de acesso ao prédio. A única diferença do Sistema Presença-Illuminação-Interior e que não apresenta geração de alarmes ou bloqueio na iluminação.
Acesso Exterior	São os sistemas que controlam o ingresso ao prédio.
Presença-Illuminação-Acesso	São sistemas que além de controlar as funções do sistema Presença-Illuminação-Interior, bloqueiam a abertura dos sistemas de acesso interior.
Presença(2)-Illuminação-Interior	São sistemas que por controlar áreas maiores requerem dois sensores de Presença, o funcionamento é igual que o Sistema Presença-Illuminação-Interior.
Presença(2)-Illuminação-Exterior	São sistemas que por controlar áreas maiores requerem dois sensores de Presença, o funcionamento é igual ao Sistema Presença-Illuminação-Exterior.
Presença-acesso(2)-Illuminação	Sistemas que controlam dos sistemas de acesso interior, o funcionamento é igual que o Sistema Presença-Illuminação-Acesso.
Controle de Incêndios	Sistemas de controle de presença de fumaça e existência de incêndios, gera alarmes na presença de incêndios.
Controle uso de HVAC	Permite desligar o HVAC quando não é detectada presença na área de utilização do sistema.
Sistema de Vigilância	Aplica as políticas de controle quando é detectada presença não autorizada.

## 6.4 Modelos dos Objetos

A figura 6.2 ilustra os modelos dos objetos identificados no conjunto de dispositivos utilizados na automação do prédio.

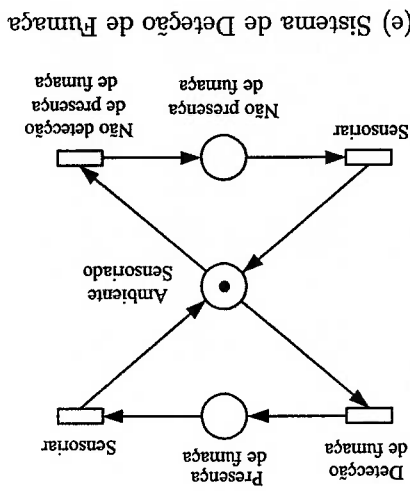
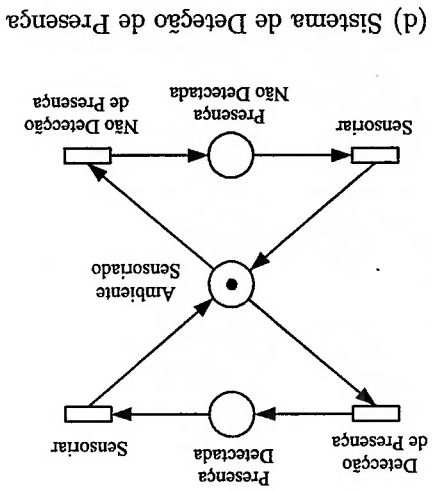
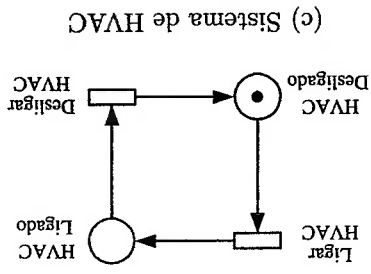
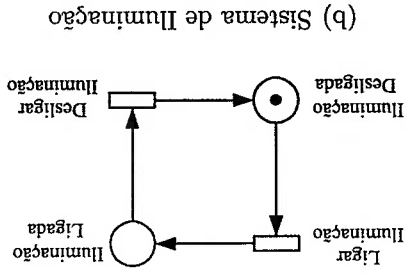
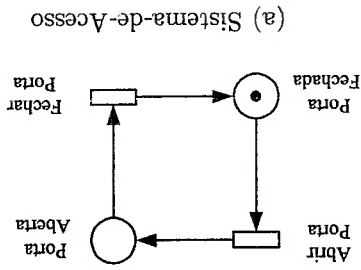


Figura 6.1: Modelos dos Sistemas de Automação do Prédio

## 6.4.1 Modelos dos Objetos com o Sistema de Mensagens

A figura 6.2 apresenta os modelos em rede *GHCENesys* dos objetos do prédio com o correspondente sistema de mensagens .

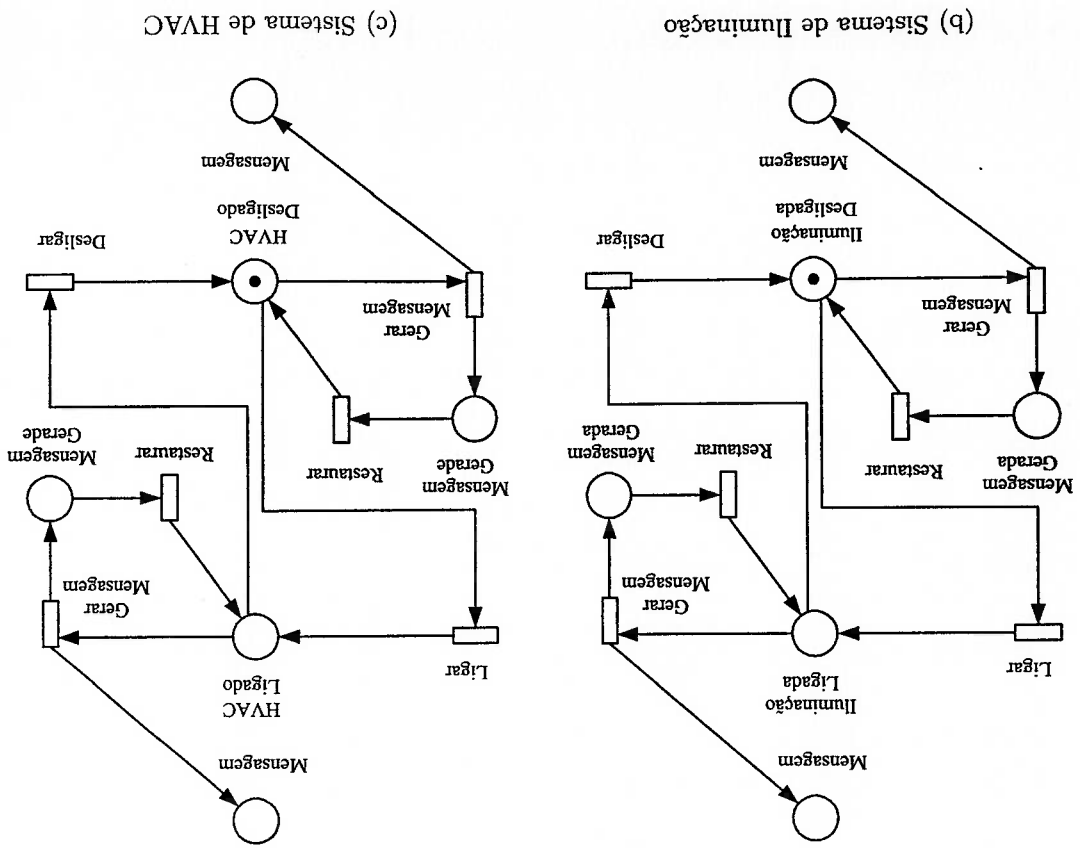
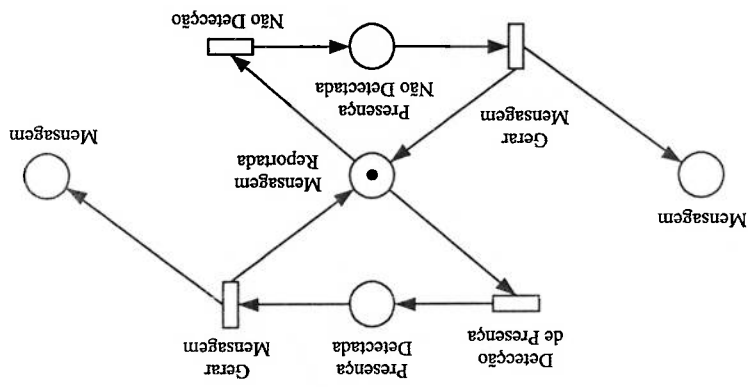
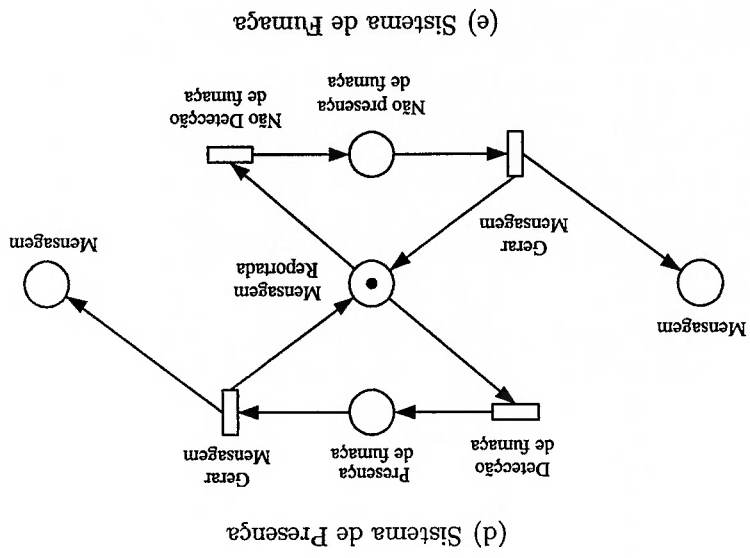




Figura 6.2: Modelos dos Sistemas de Automação do Prédio com o Sistema de Mensagens



Realiza-se o controle em cada área do prédio utilizando as regras de controle, com base nos sinais dos sistemas de sensores (detecção de presença e fumaça), causando ou inibindo eventos nos dispositivos do prédio. Cada objeto de controle apresenta um subsistema *Planta-Control* do prédio, por tanto é possível considerar o controle do prédio como um conjunto de instâncias dos Objetos Controladores. A seguir apresentam-se os modelos dos Objetos Controladores.

### 6.4.2 Modelos dos Objetos Controladores

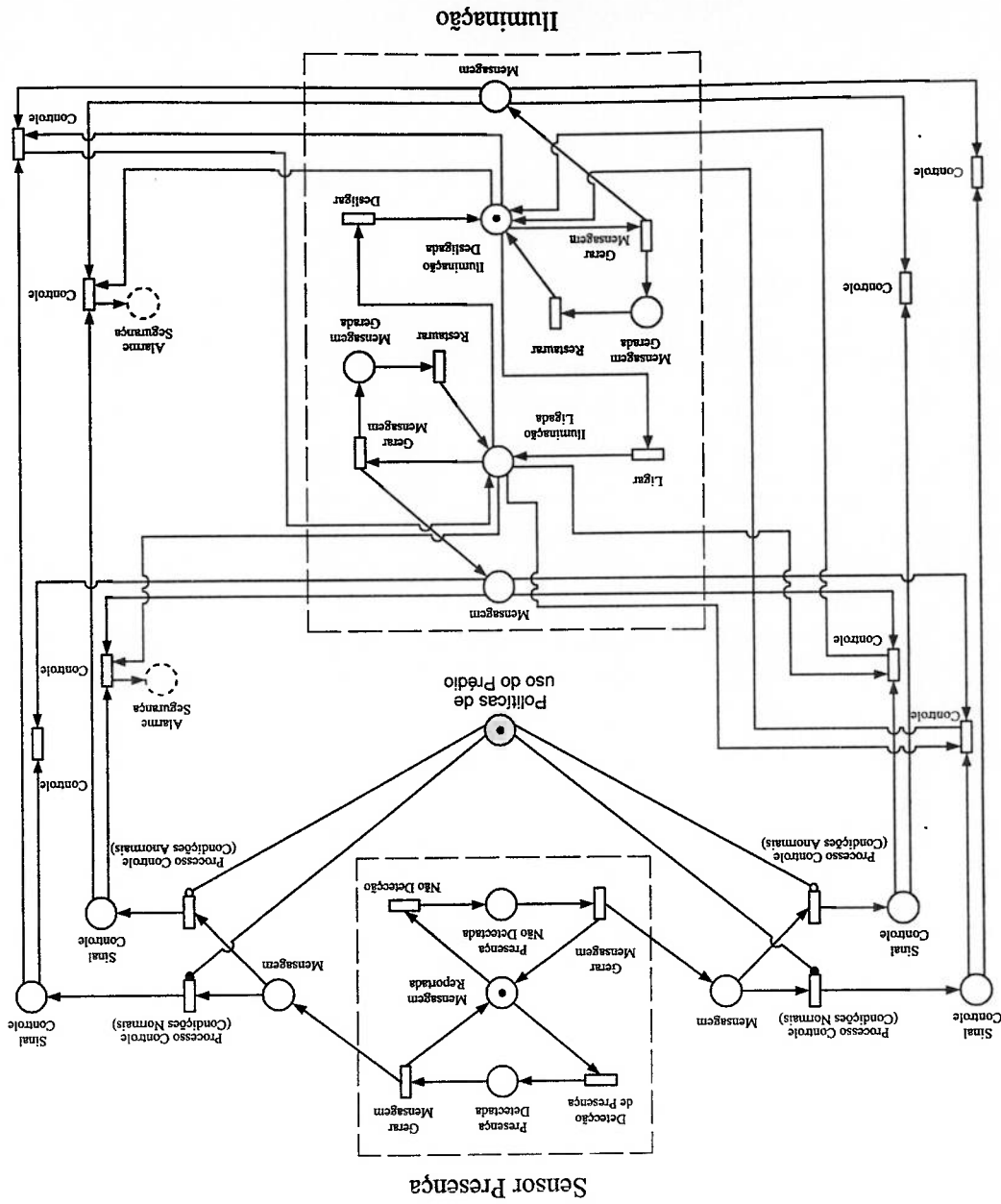


Figura 6.3: Modelo do Objeto Presença Iluminação Interior

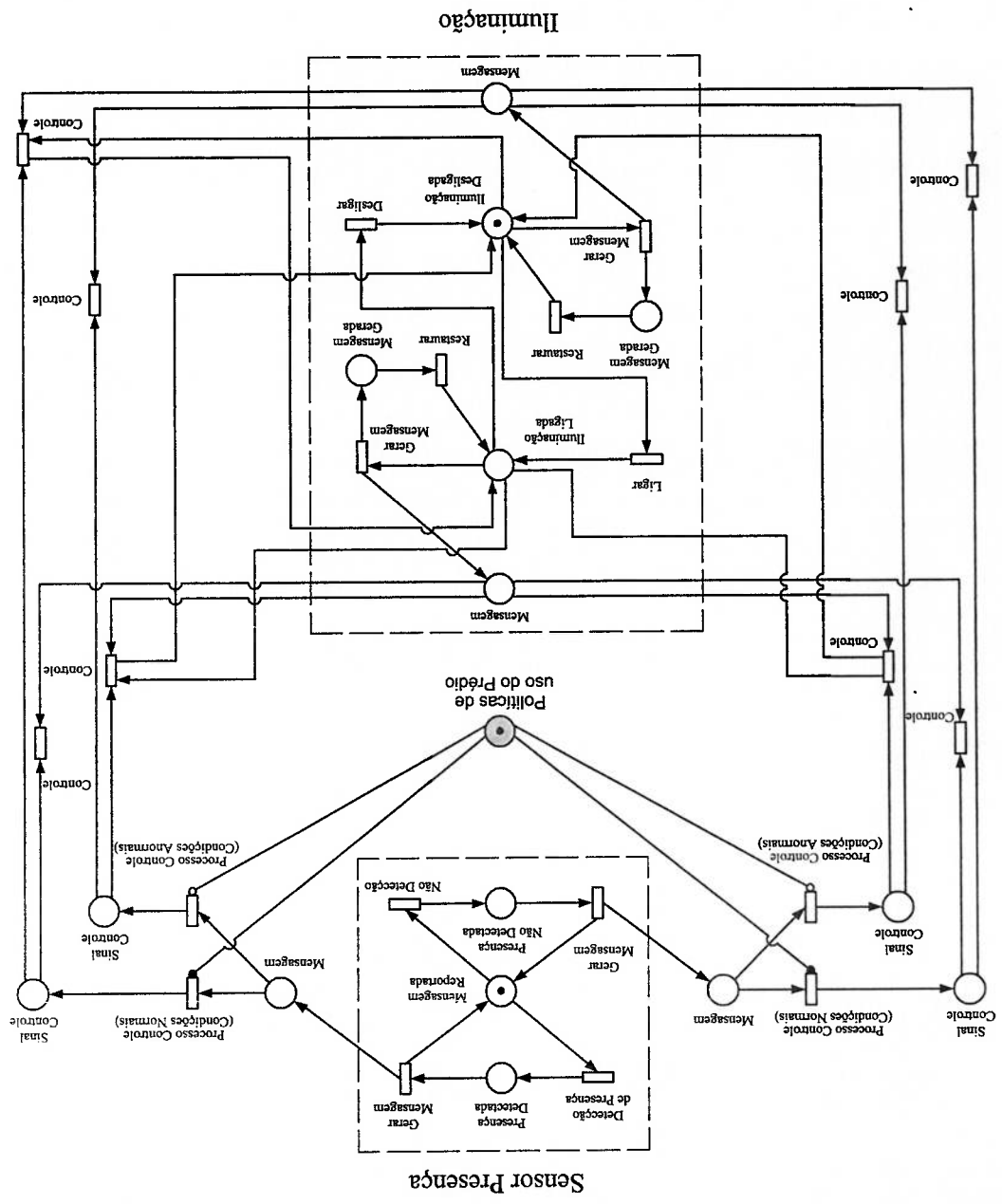
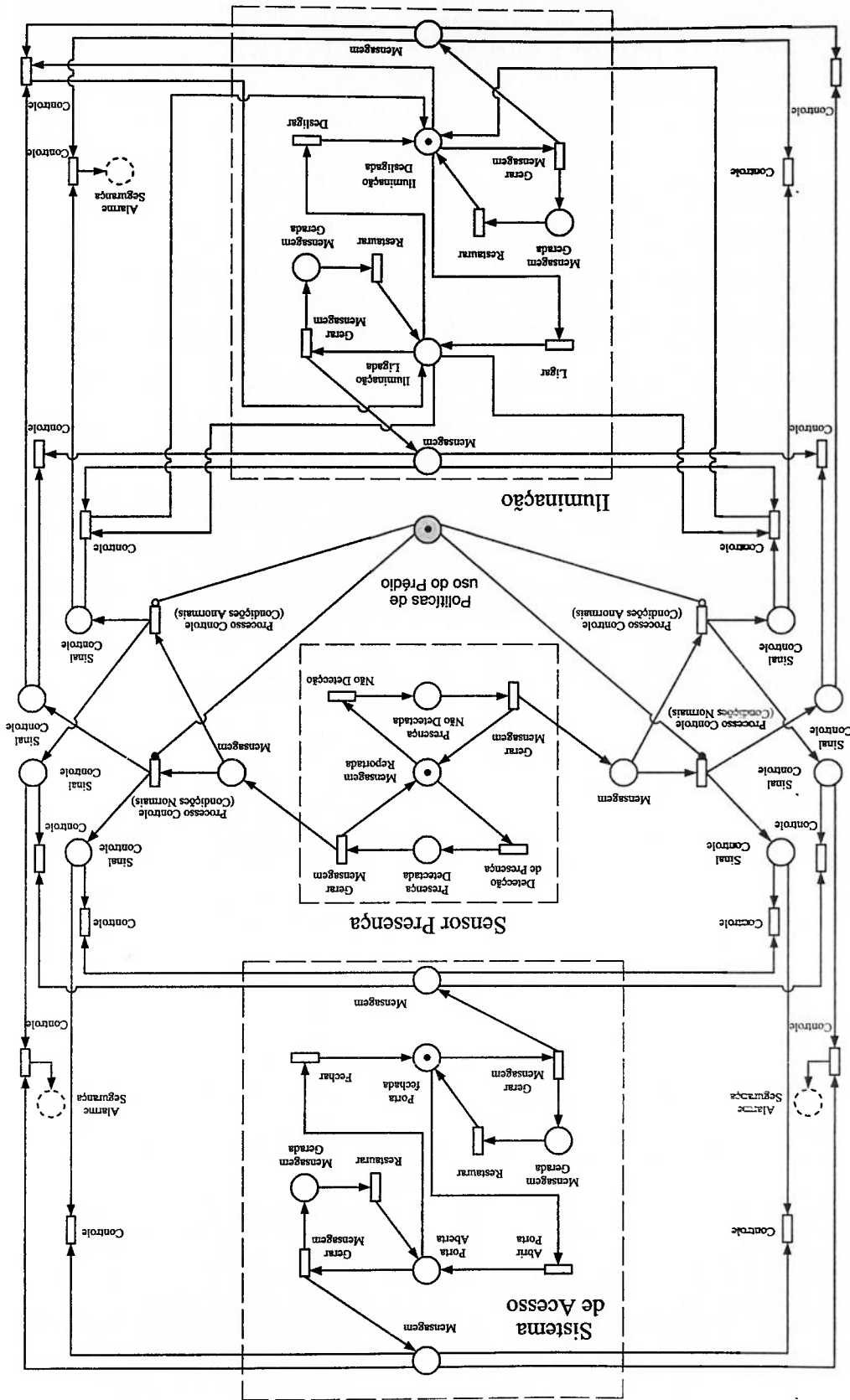


Figura 6.4: Modelo do Objeto Presença Iluminação Exterior

Figura 6.5: Modelo do Objeto Presença Acesso Iluminação



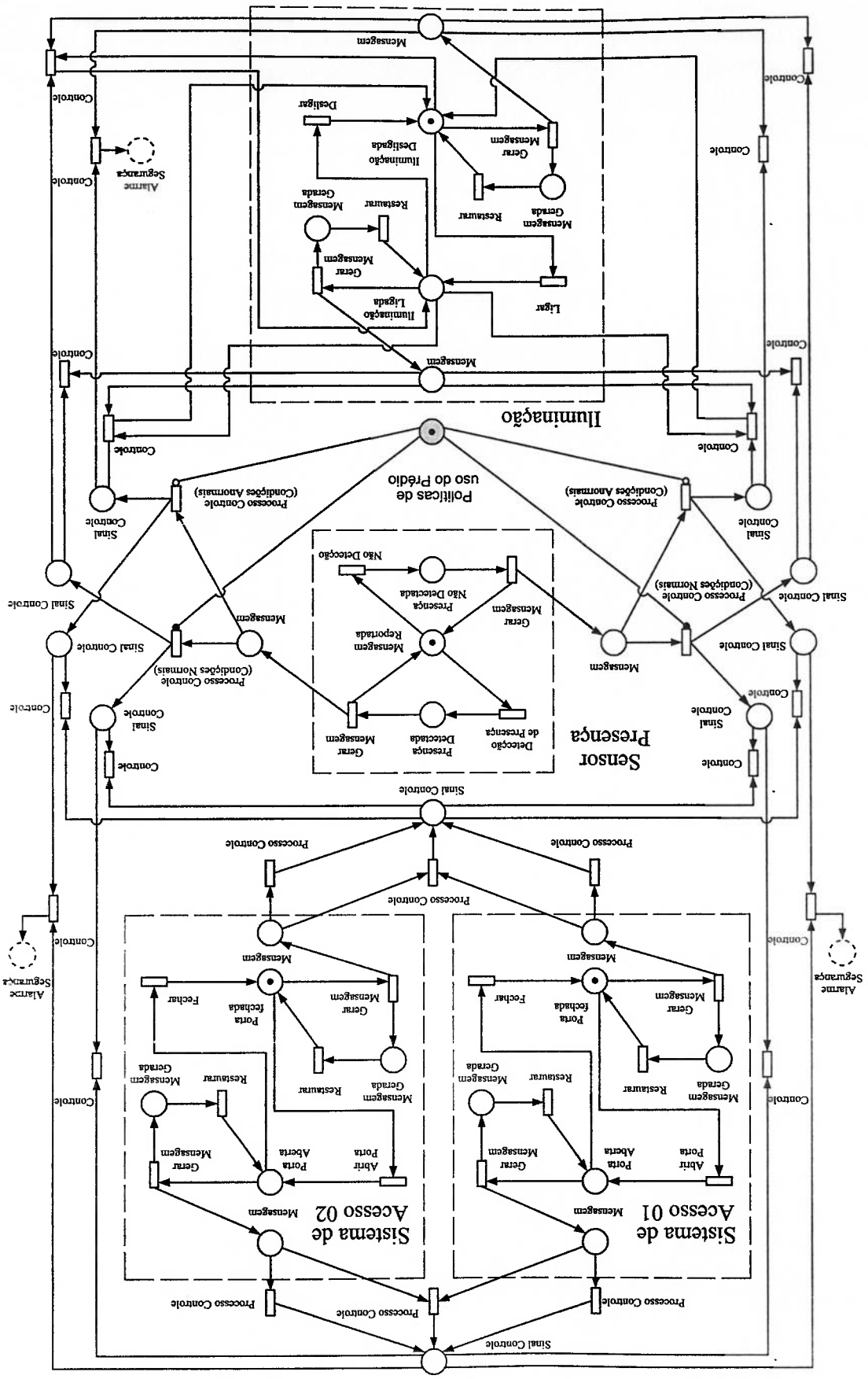
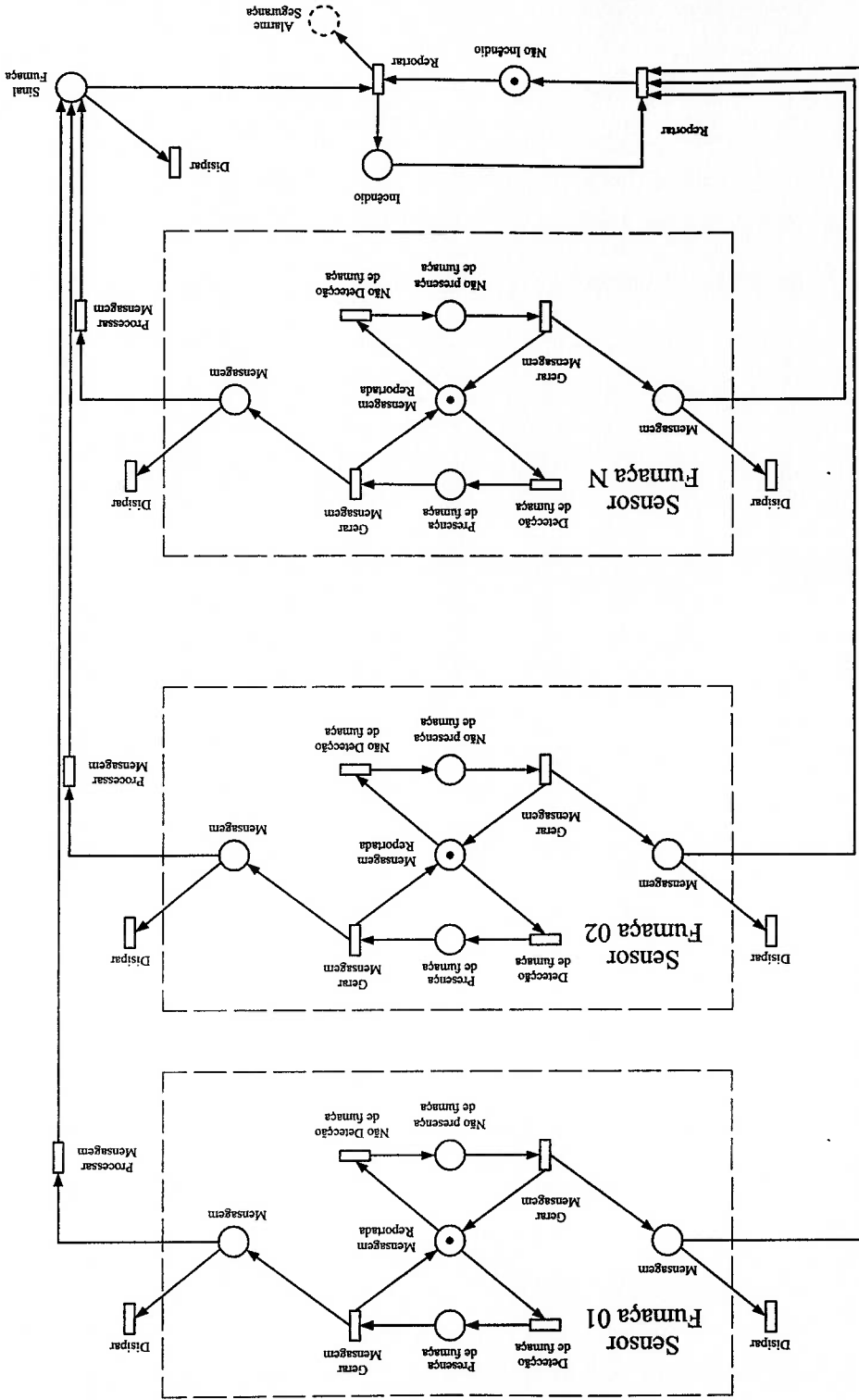


Figura 6.6: Modelo do Objeto Duplo Acesso Iluminação Presença

Figura 6.7: Modelo do Objeto Detecção de Incêndios



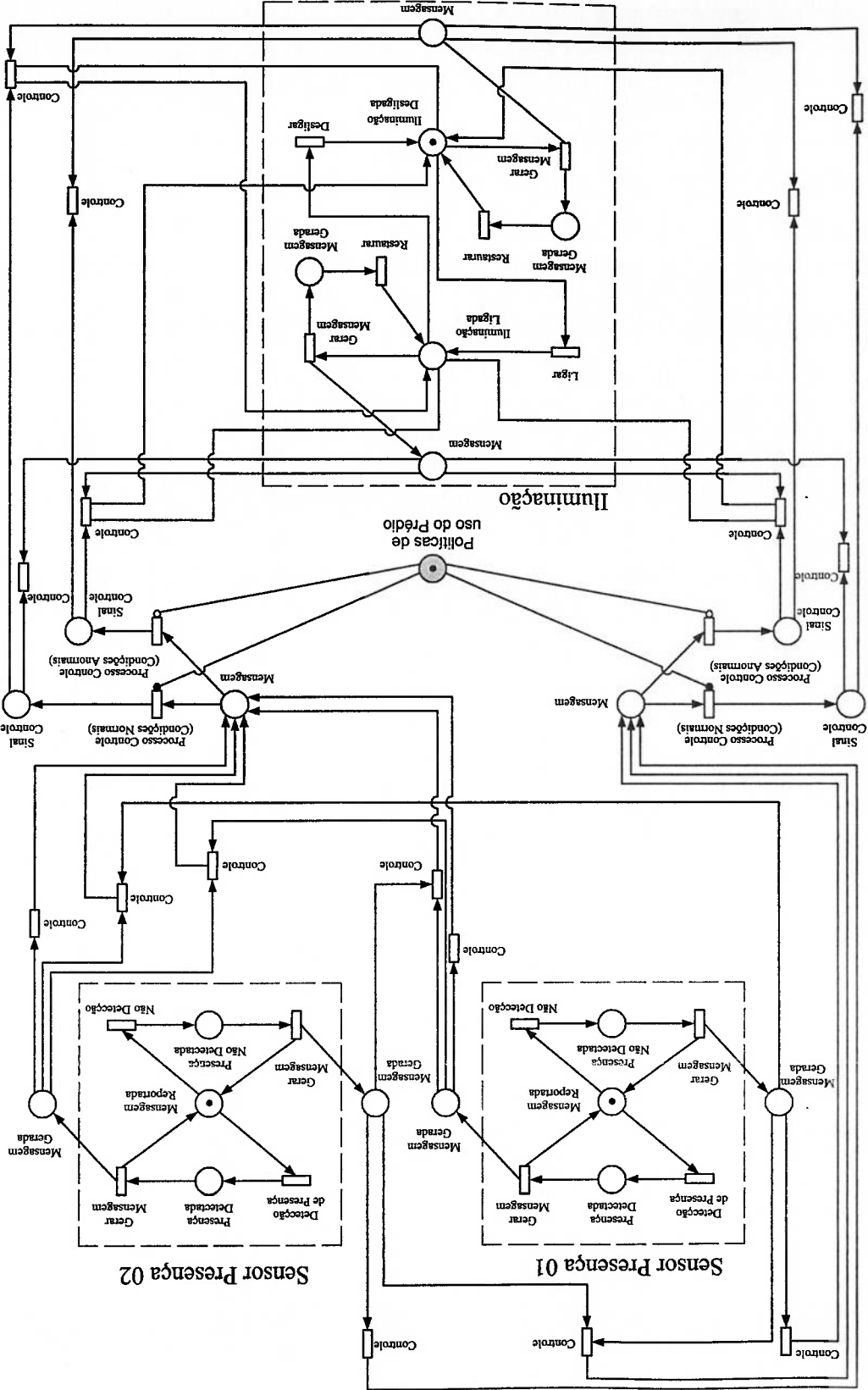


Figura 6.8: Modelo do Objeto Duplo Presença Iluminação Exterior

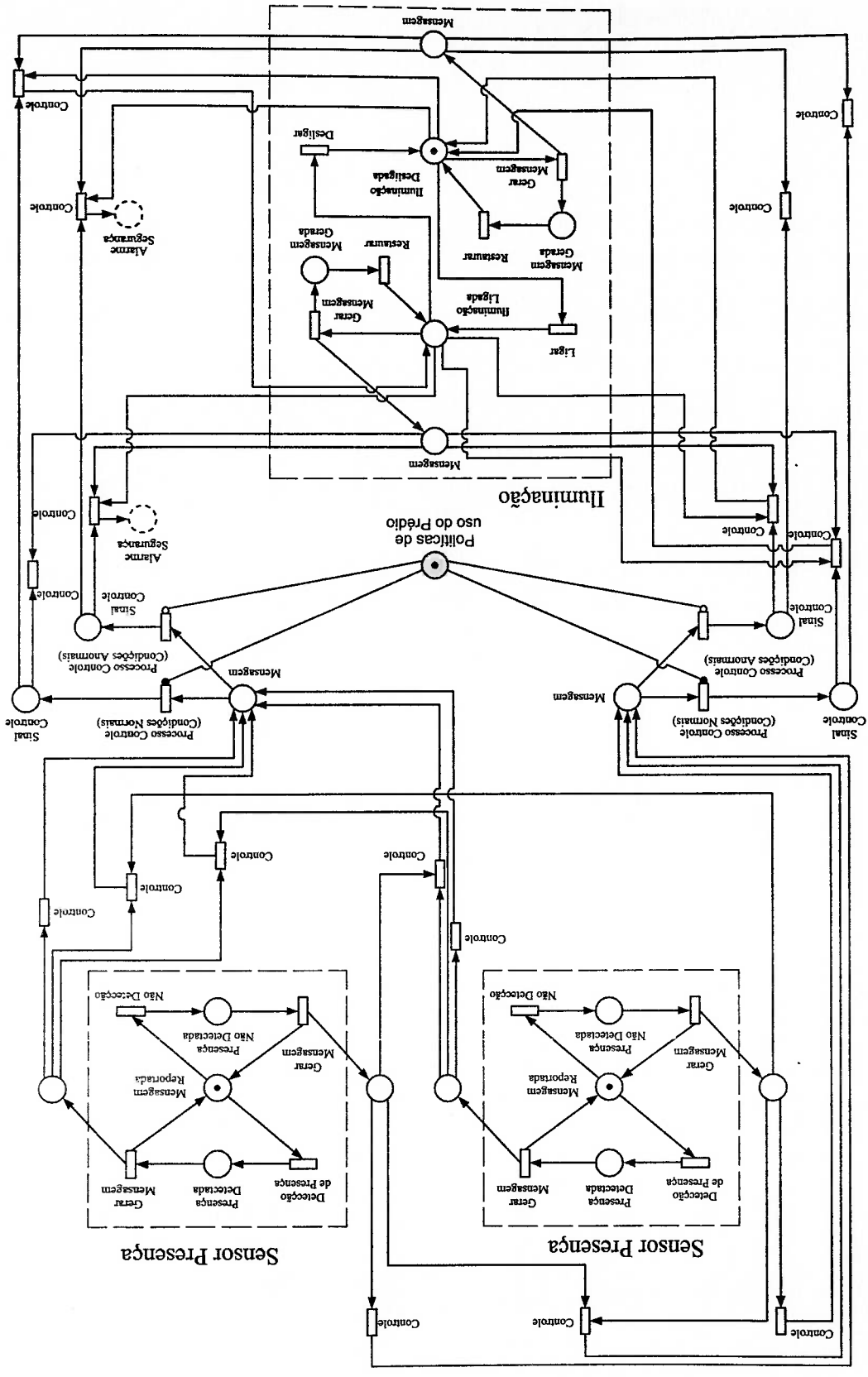


Figura 6.9: Modelo do Objeto Duplo Presença Iluminação Interior



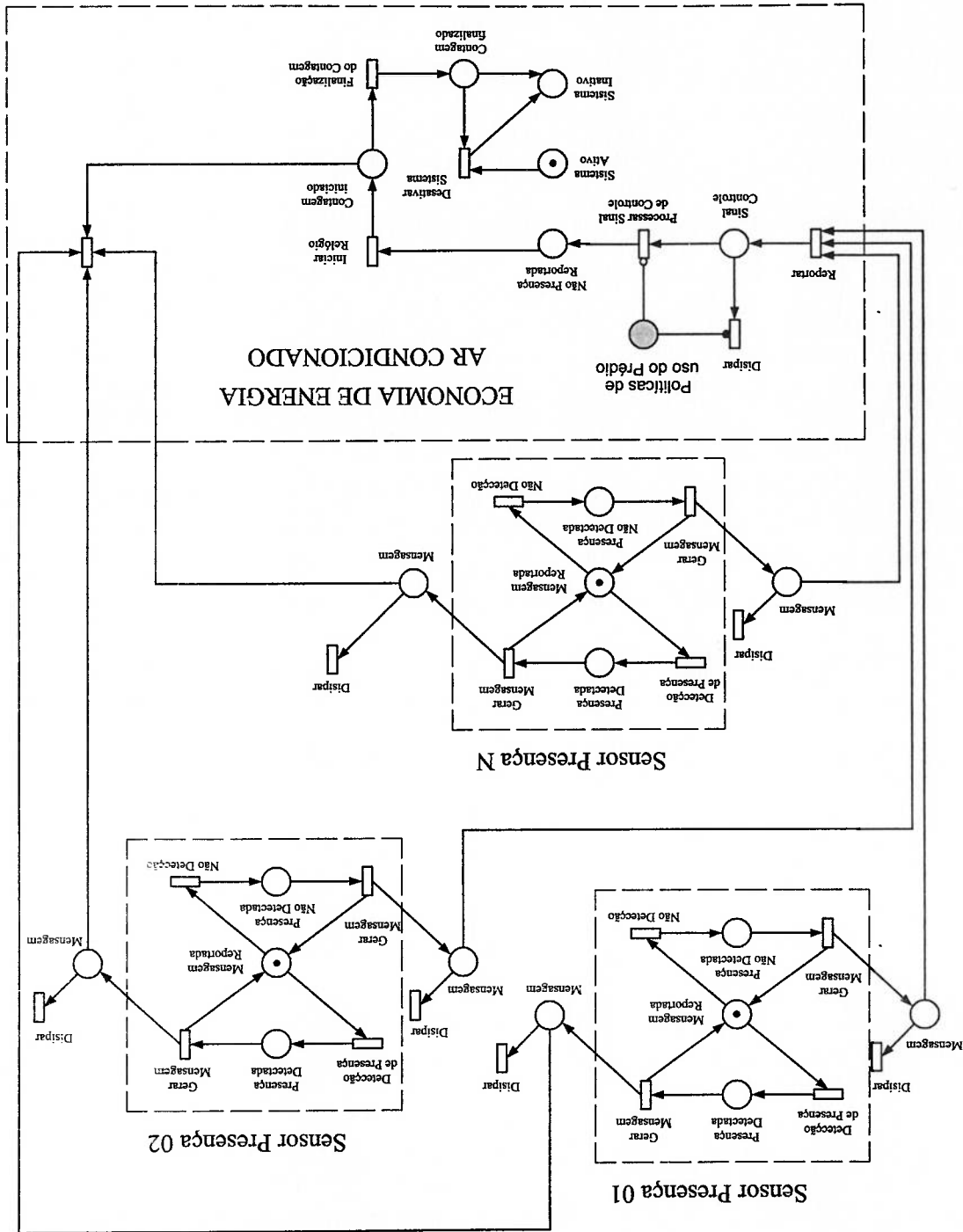
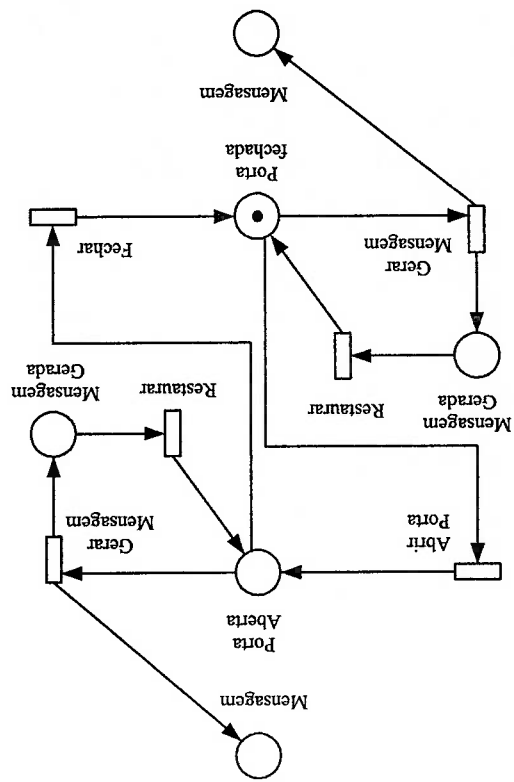


Figura 6.10: Modelo do Objeto Economia de Energia do HVAC

Figura 6.11: Modelo do Objeto do Sistema Acesso





O Sistema de Vigilância gera alarmes que bloqueiam a ocorrência de certos eventos, isto é desabilita algumas transições. Assim alguns modelos dos objetos controladores precisam incorporar mecanismos de resposta aos alarmes do sistema de vigilância. Na *GHENeSys* o método natural para fazer isto é com um Pseu-Box cuja marcação depende de condições externas, neste caso da ocorrência de alarmes. Os modelos dos objetos incorporando o mecanismo de comunicação dos alarmes são apresentados a seguir.

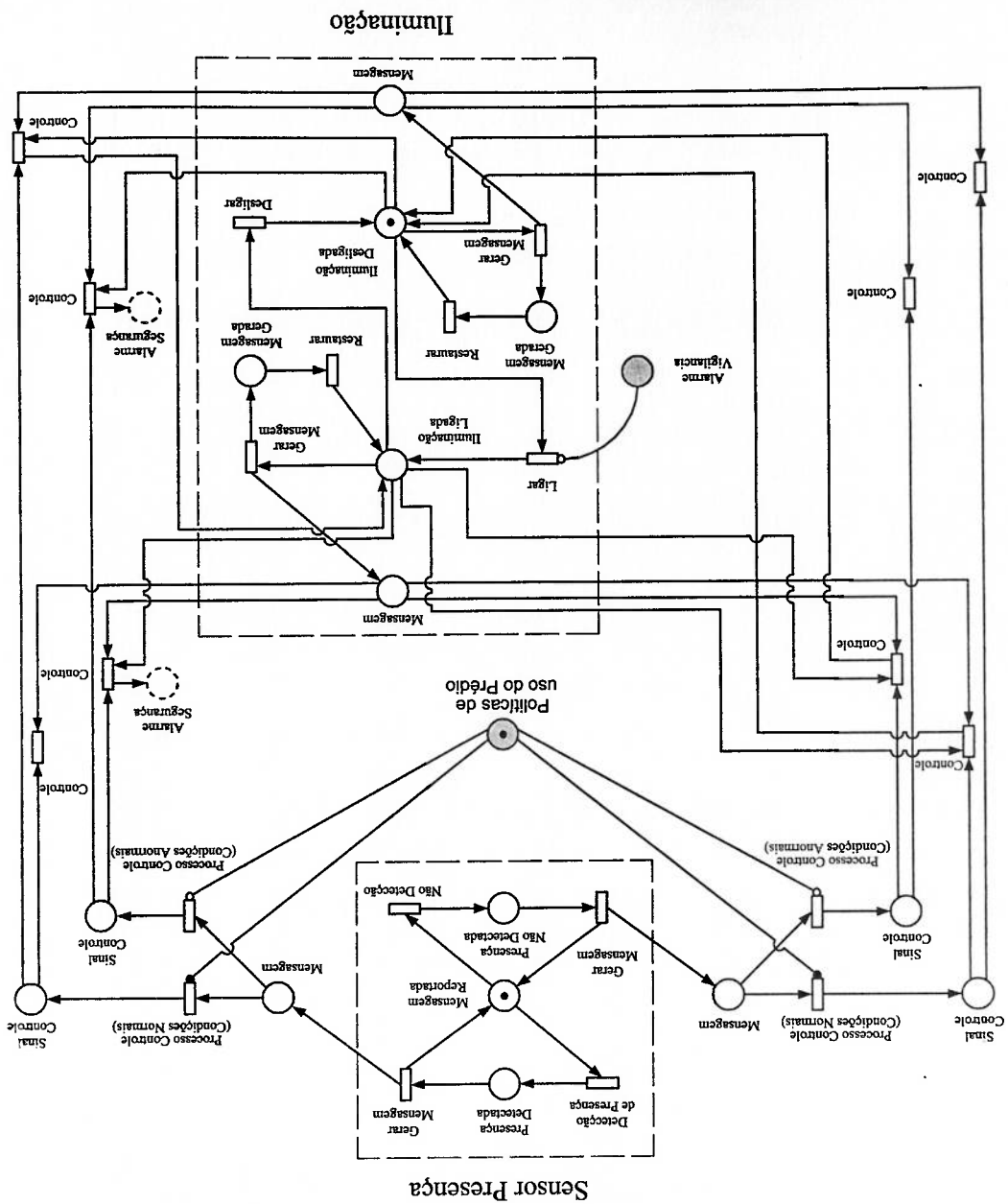
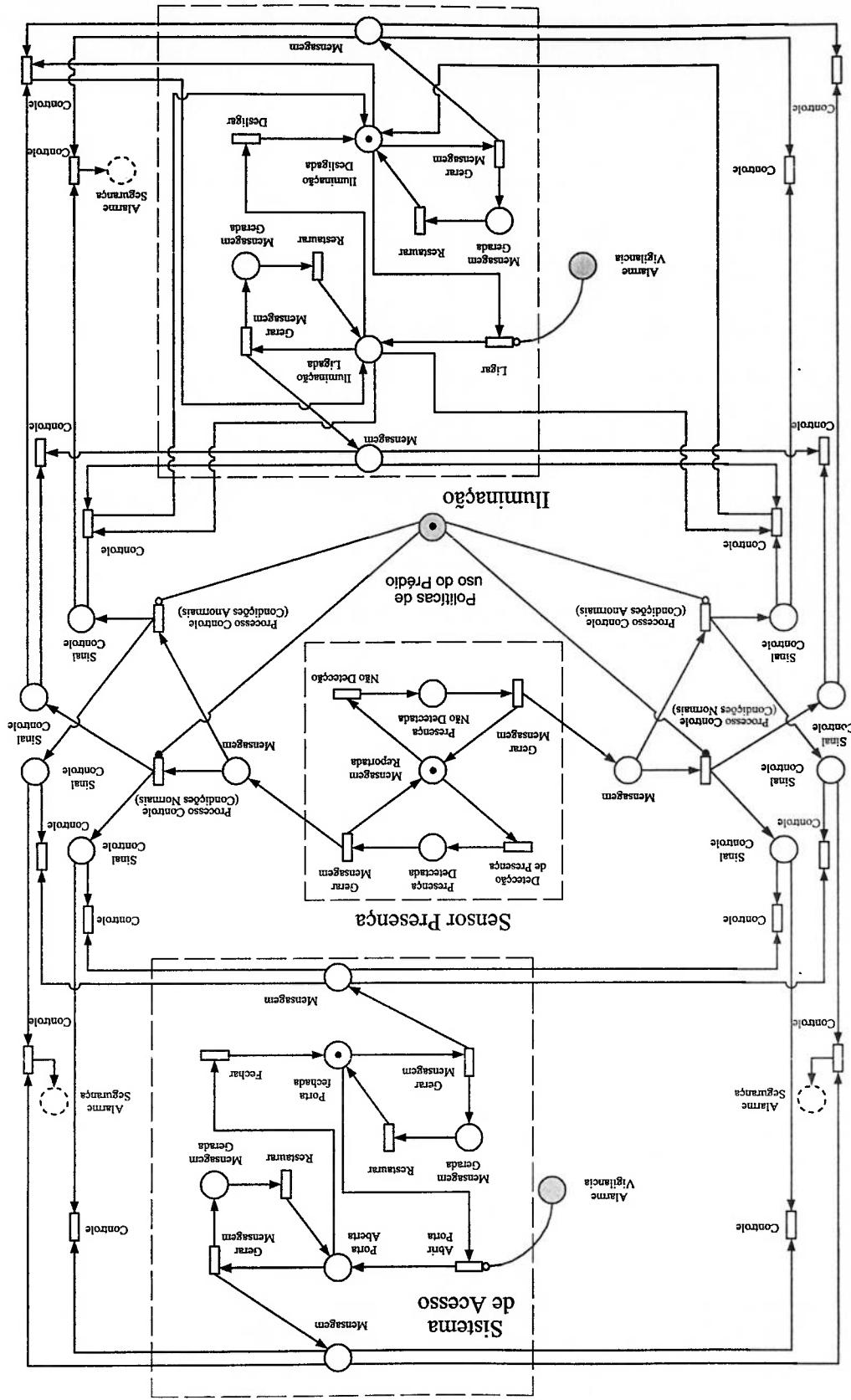


Figura 6.13: Modelo do Objeto Presença Iluminação Vigilância

Figura 6.14: Modelo do Objeto Presença Iluminação Acesso Vigilância



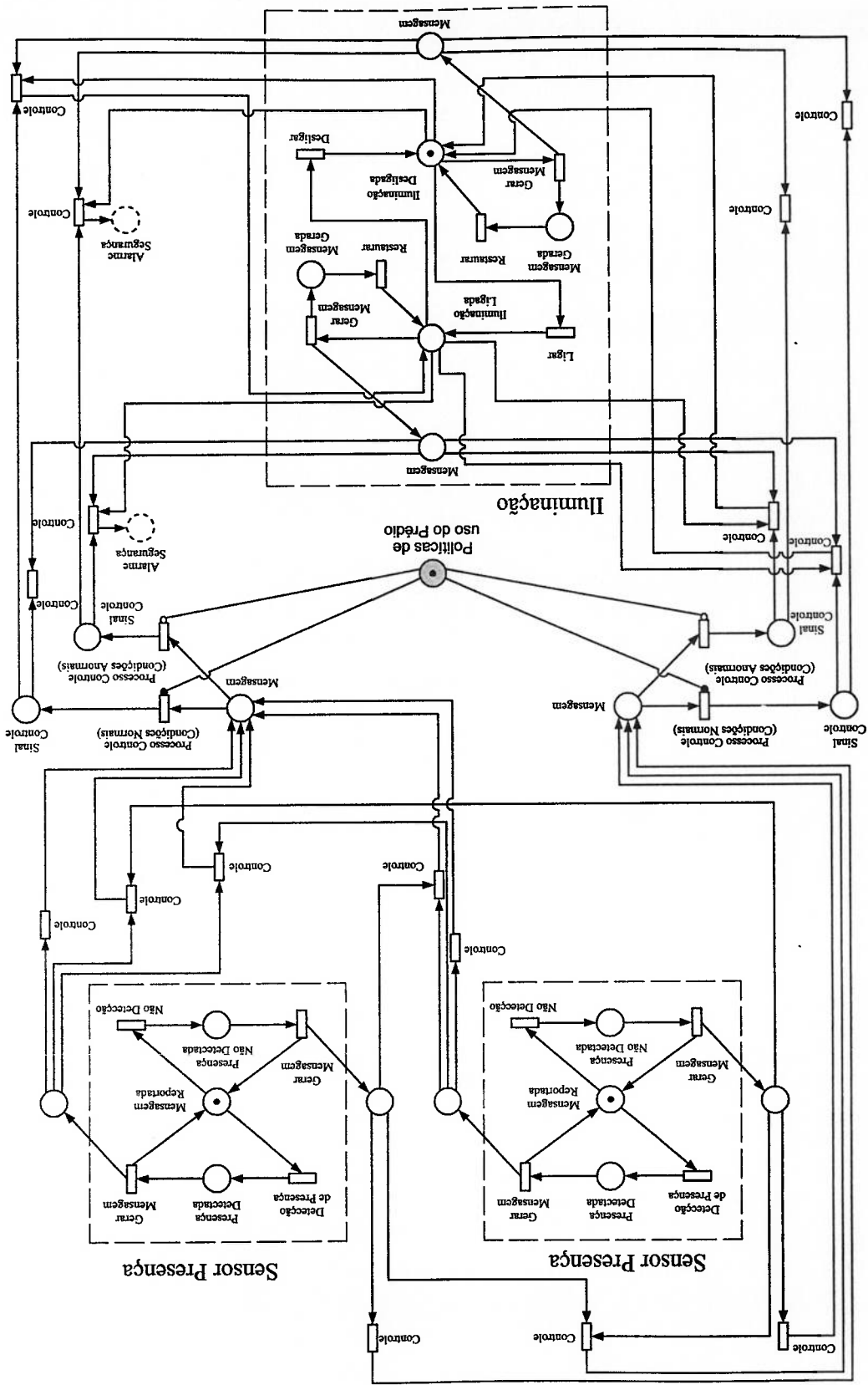
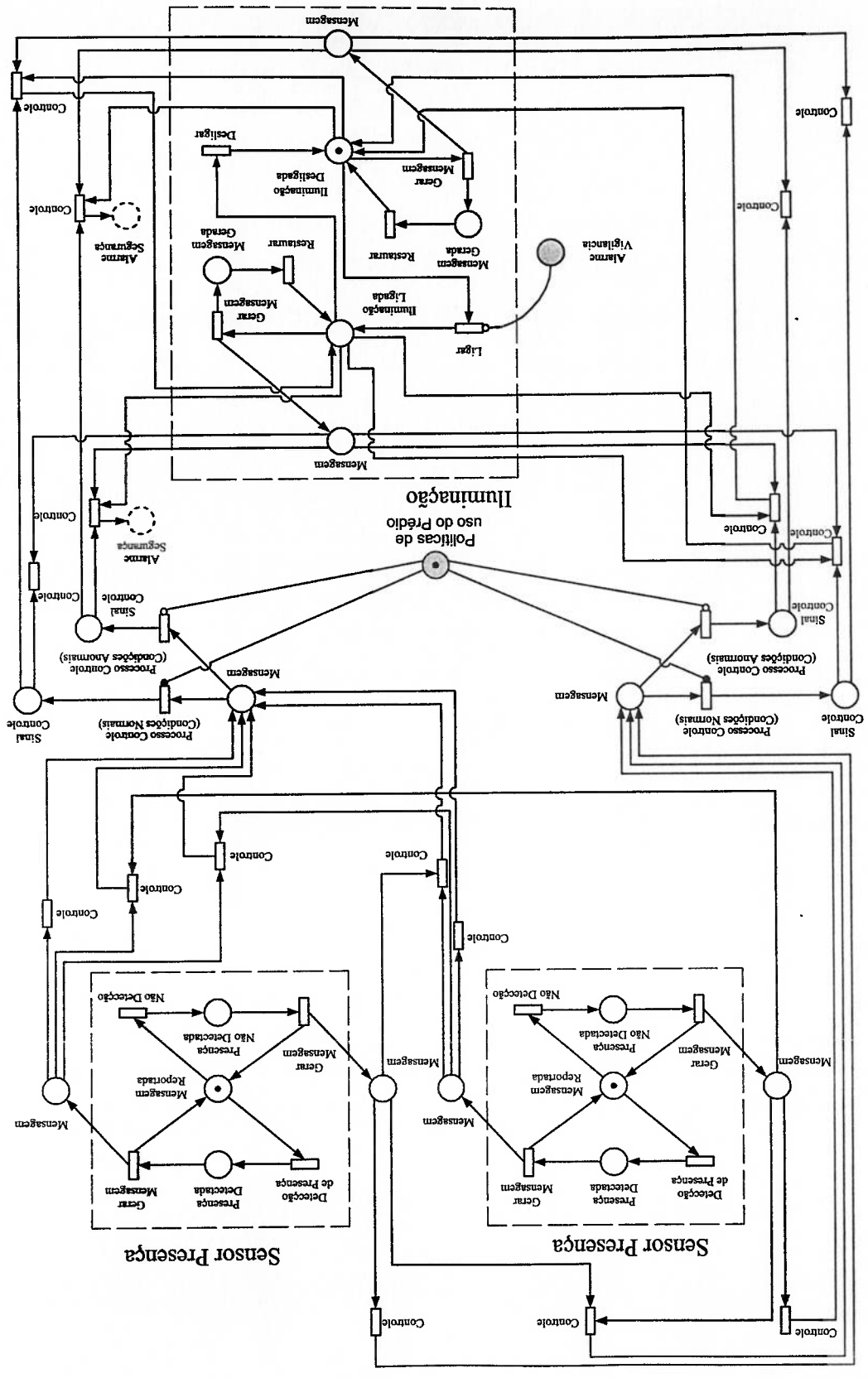


Figura 6.15: Modelo do Objeto Duplo Acesso Iluminação Presença Vigilância

Figura 6.16: Modelo do Objeto Duplo Presença Iluminação Interior Vigilância



O Encapsulamento permite isolar uma subrede em um MacroElemento. A seguir apresentam-se os modelos dos objetos controladores do prédio.

## 6.5 Encapsulamento dos Objetos em MacroElementos

Figura 6.17: Modelo do Objeto Acesso Vigilância

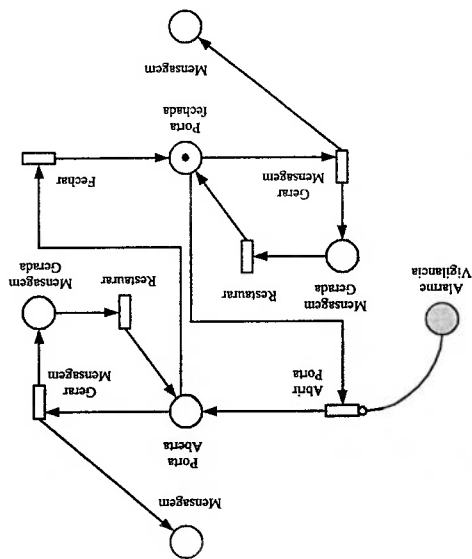




Figura 6.18: Encapsulamento do Objeto Presença Iluminação Exterior

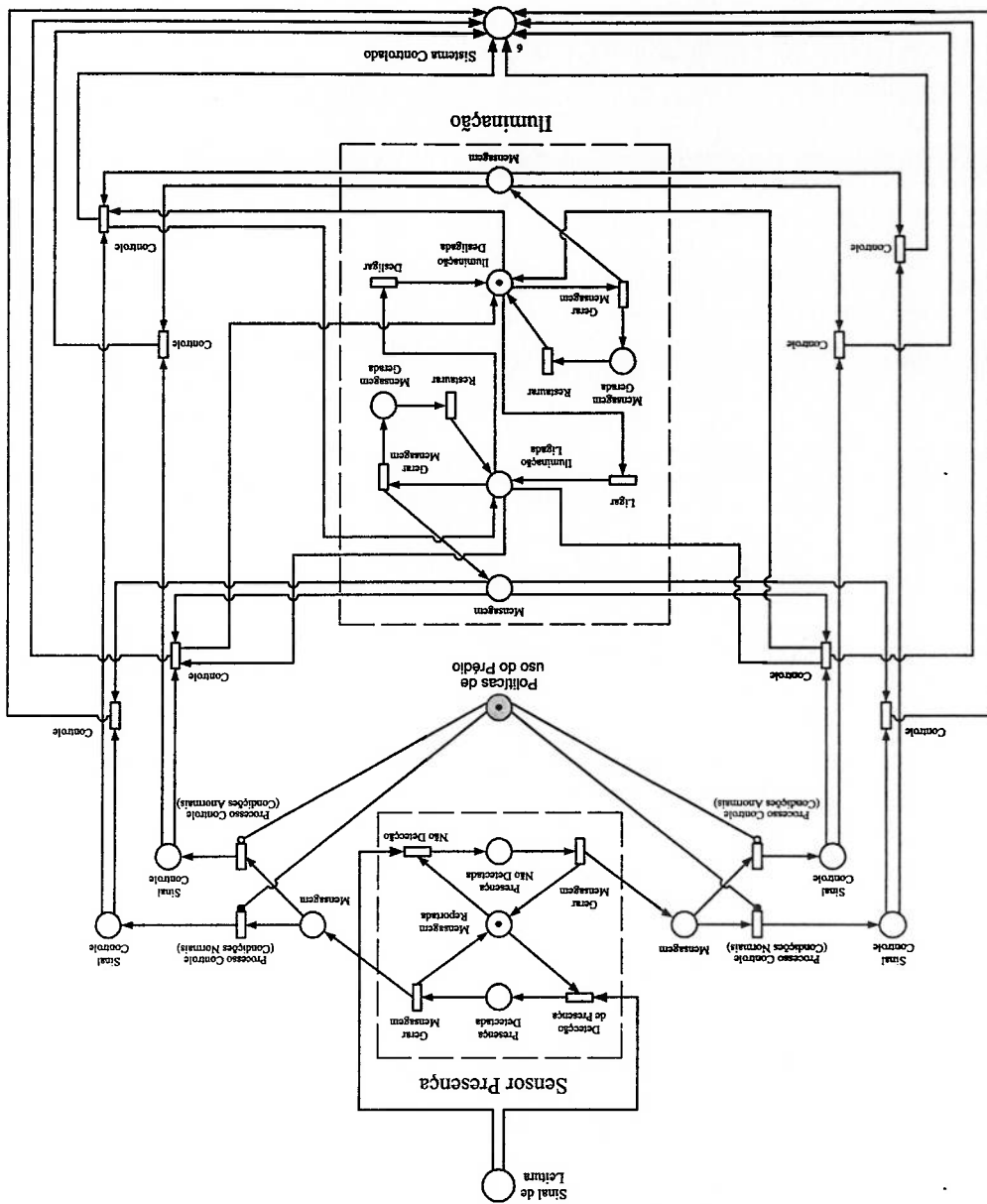


Figura 6.19: Encapsulamento do Objeto Presença Iluminação Interior Vigilância

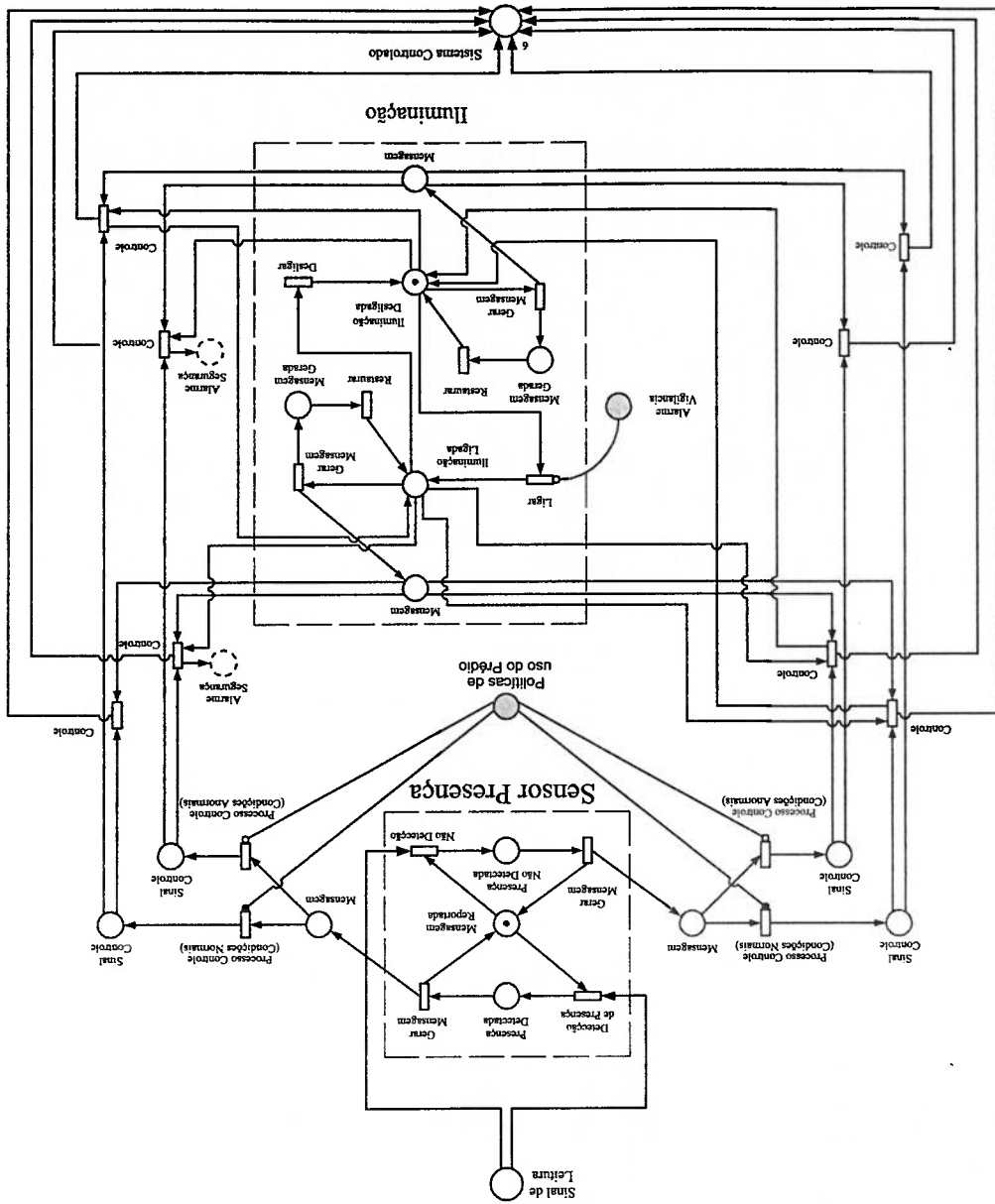


Figura 6.20: Encapsulamento do Objeto Presença Iluminação Acesso Vigilância

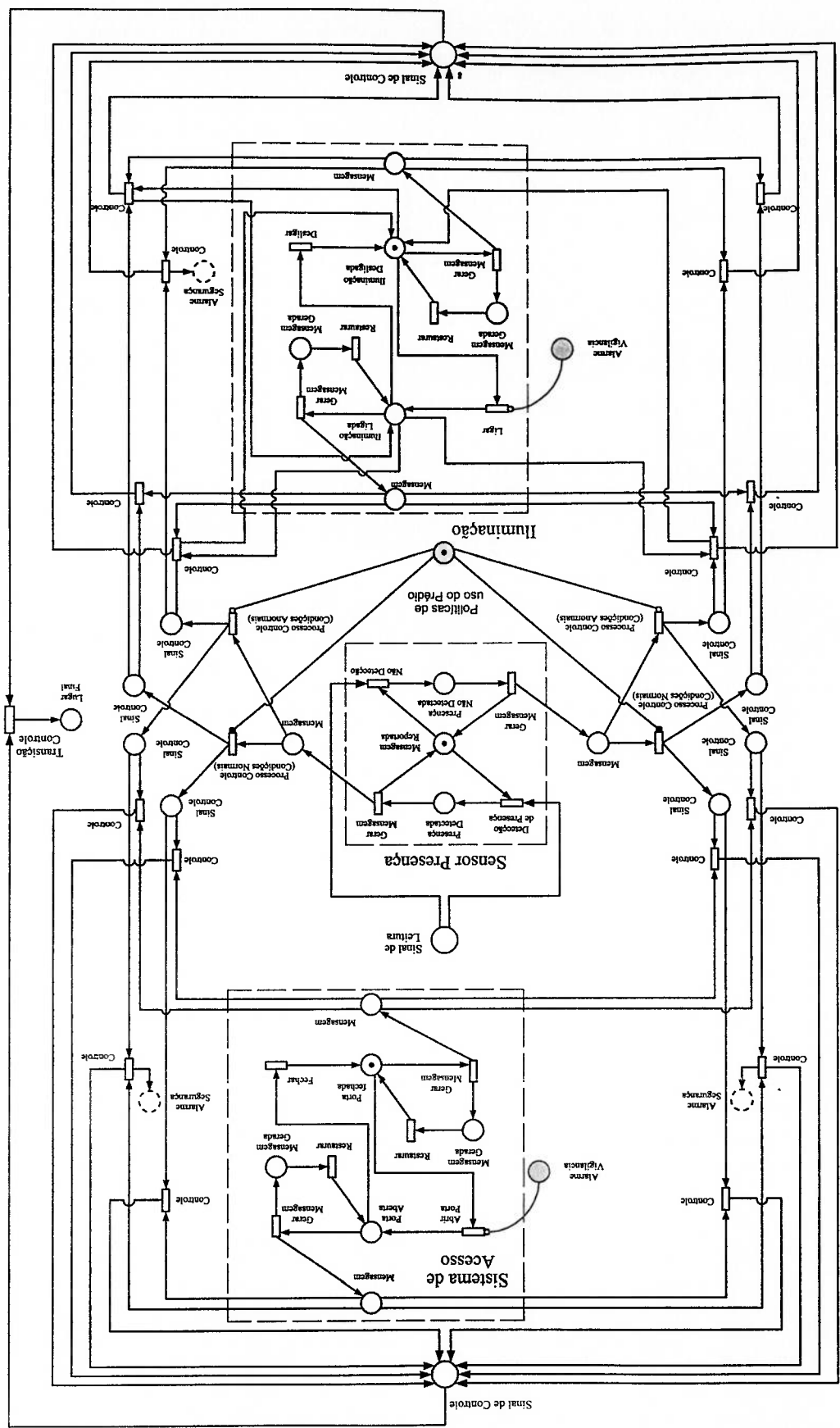
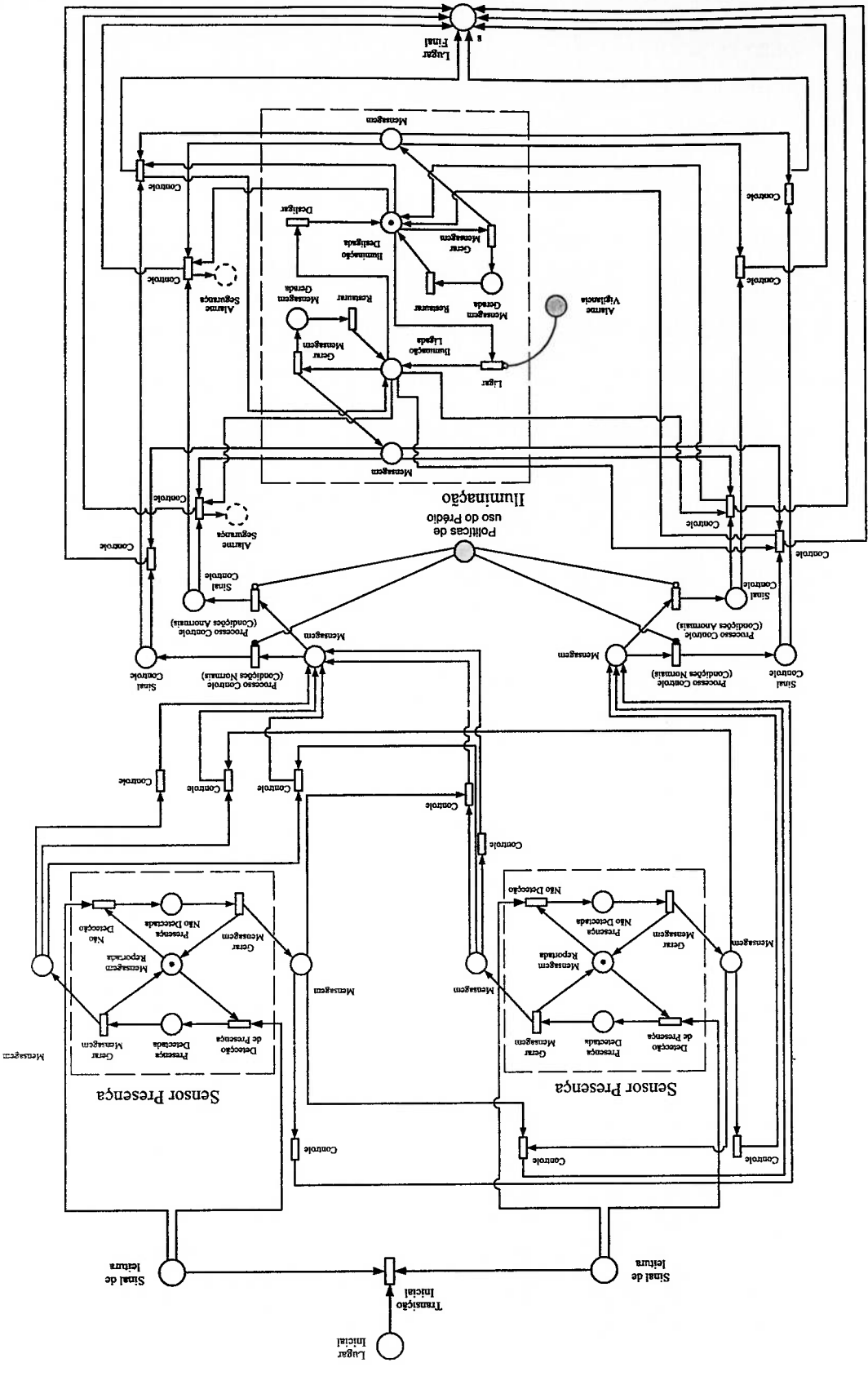


Figura 6.21: Encapsulamento do Objeto Duplo Presença Iluminação Vigilância



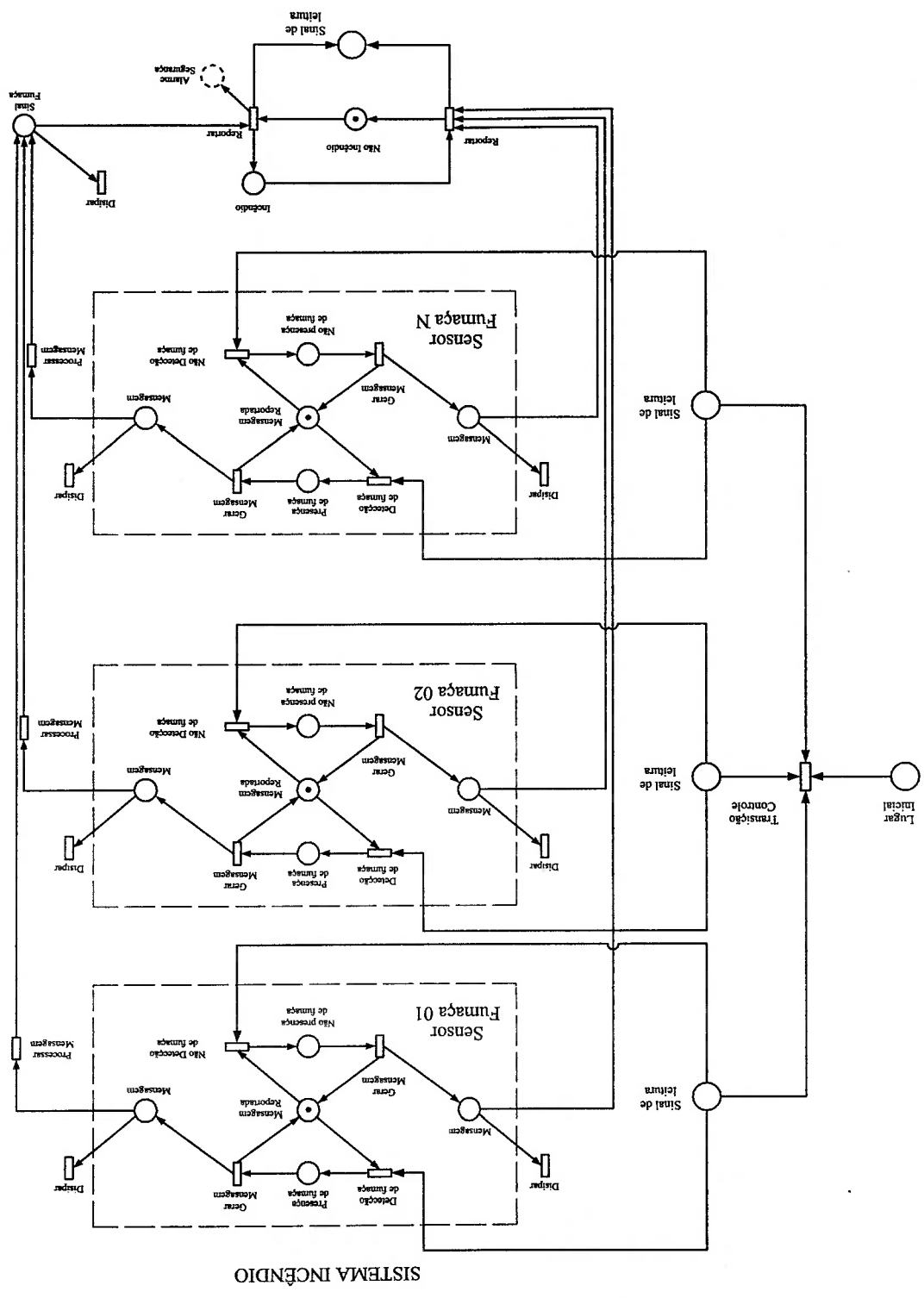


Figura 6.22: Encapsulamento do Objeto de Detecção de Incêndios

Figura 6.23: Encapsulamento do Objeto Duplo Presença Iluminação Exterior

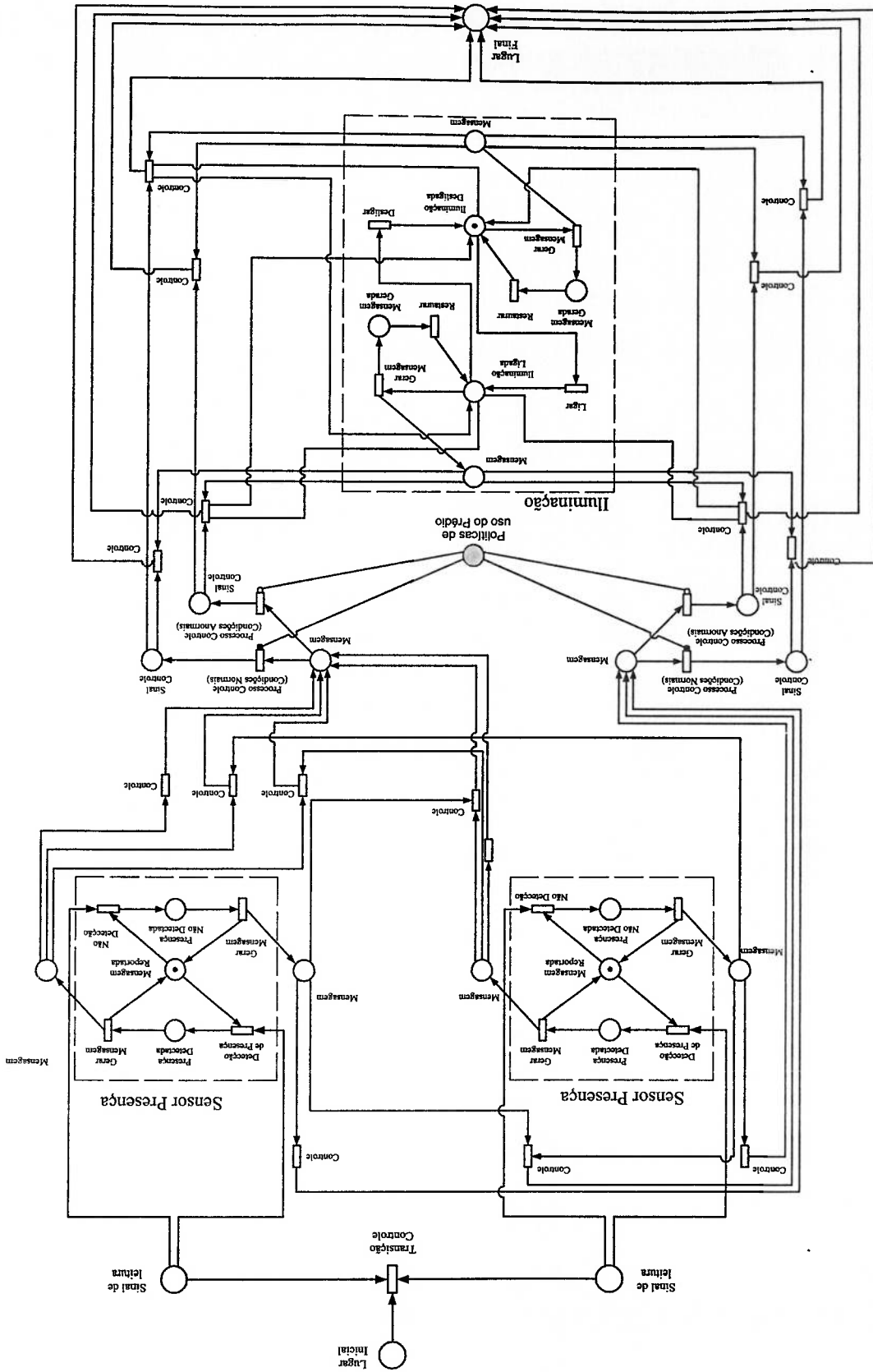


Figura 6.24: Encapsulamento do Objeto Duplo Acesso Iluminação Presença Vi-  
gilância

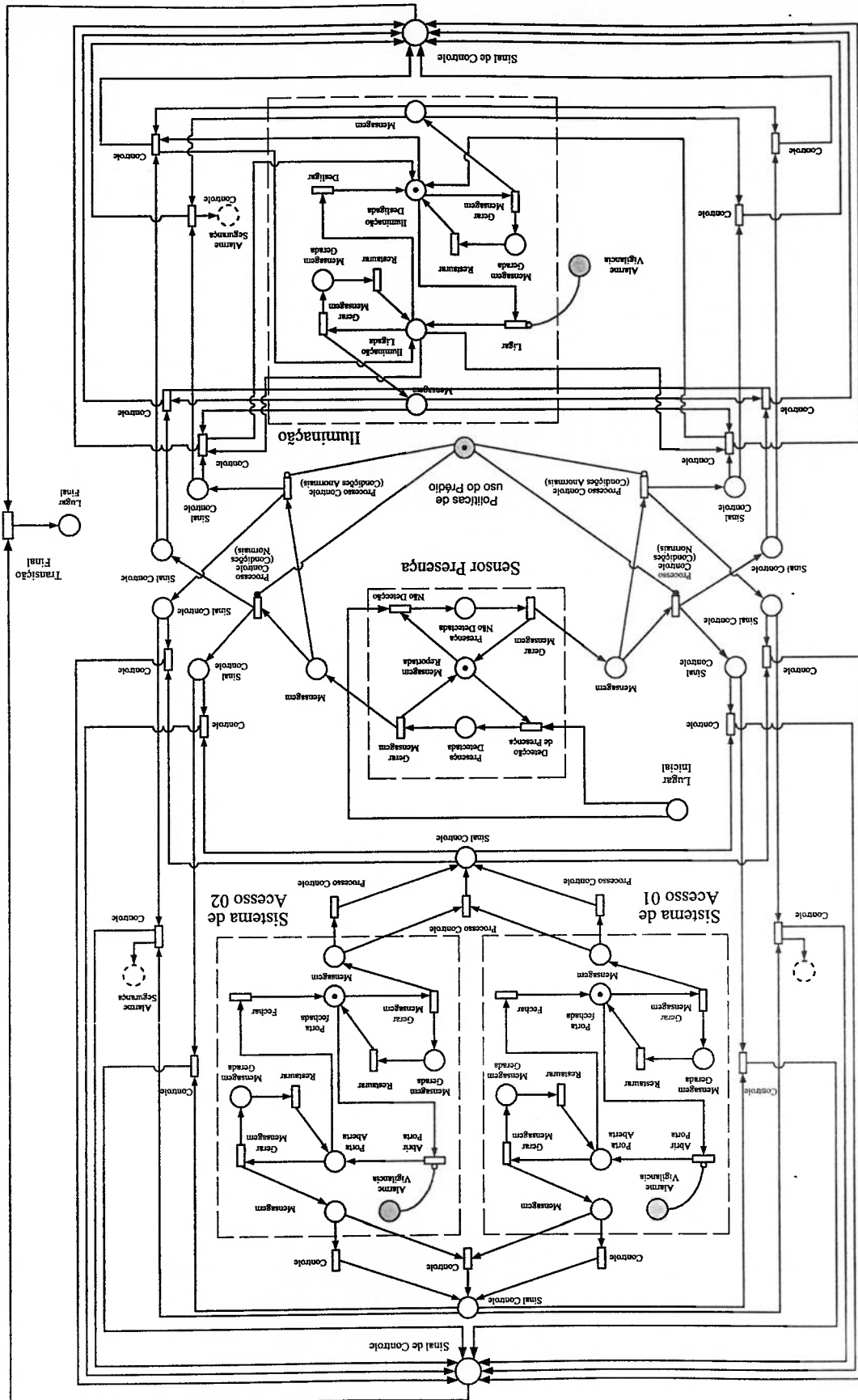
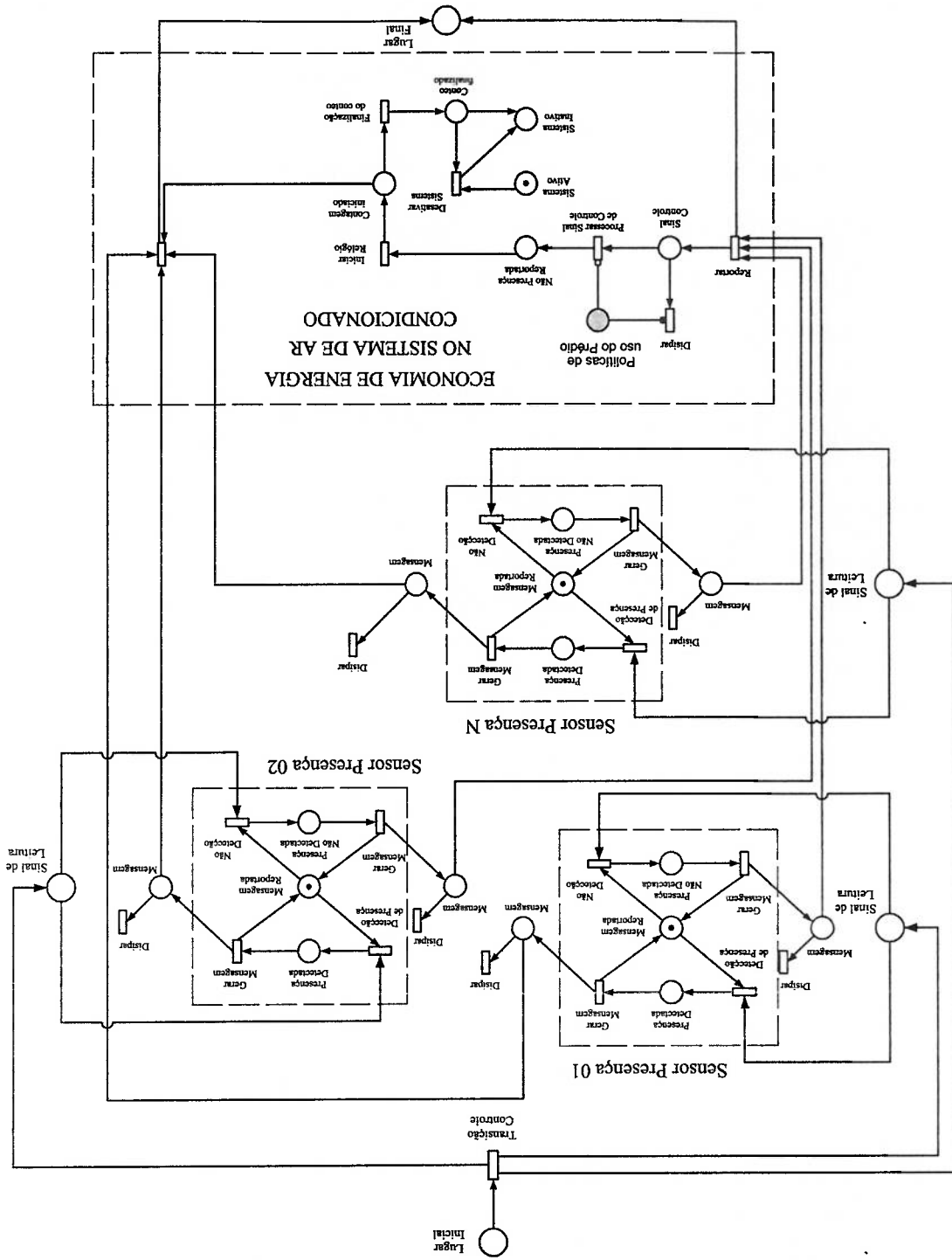


Figura 6.25: Encapsulamento do Objeto Economia de Energia do HVAC





## 6.6 Árvore de Hierarquia

Constrói-se com base nos Objetos Controladores definidos para o prédio, procurando estabelecer a hierarquia do sistema de controle. A seguir são determinadas as instâncias dos Objetos Controladores necessárias em cada andar.

A tabela 6.2 apresenta a hierarquia das instâncias dos Objetos Controladores e do andar terreo descrito nas plantas A4-01, A4-05, A4-08 e A4-011. A figura 6.27 ilustra o sistema de comunicação dos MacroElementos.

A tabela 6.3 apresenta a hierarquia das instâncias dos Objetos Controladores do andar Subsolo 1 descrito nas plantas A4-03, A4-05, A4-08 e A4-011. A figura 6.6 ilustra o sistema de comunicação dos MacroElementos.

A tabela 6.5 apresenta a hierarquia das instâncias dos Objetos Controladores e do andar Subsolo 2 descrito nas plantas A4-03, A4-07, A4-010 e A4-13. A figura 6.6 ilustra o sistema de comunicação dos MacroElementos.

Figura 6.26: Encapsulamento do Objeto Acesso

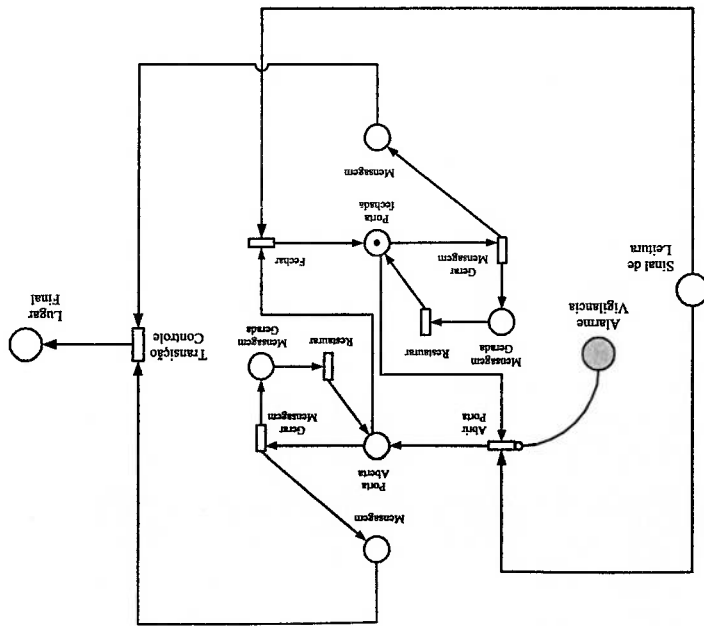


Tabela 6.2: Objetos Controladores do Andar Terreo

Elemento	Tipo de Objeto	Referencia
MB <sub>1</sub>	Sistema Acesso Vigilância	T_SA.02
MB <sub>2</sub>	Presença Iluminação Acesso Vigilância	T_PLA.01
MB <sub>3</sub>	Presença Iluminação Interior Vigilância	T_PLI.01
MB <sub>4</sub>	Sistema HVAC	T_HVAC.01
MB <sub>5</sub>	Presença Iluminação Interior Vigilância	T_PLI.02
MB <sub>6</sub>	Presença Iluminação Interior Vigilância	T_PLI.03
MB <sub>7</sub>	Presença Iluminação Acesso(2)Vigilância	T_PLA(2).01
MB <sub>8</sub>	Presença Iluminação Exterior	T_PLE.01
MB <sub>9</sub>	Presença(2) Iluminação Exterior	T_SP(2).L.03
MB <sub>10</sub>	Presença Iluminação Exterior	T_PLE.02
MB <sub>11</sub>	Sistema Acesso Vigilância	T_SA.02
MB <sub>12</sub>	Sistema Acesso Vigilância	T_SA.03
MB <sub>13</sub>	Presença Iluminação Exterior	T_PLE.03
MB <sub>14</sub>	Sistema Acesso Vigilância	T_SA.03
MB <sub>15</sub>	Sistema HVAC	T_HVAC.02
MB <sub>16</sub>	Presença Iluminação Exterior	T_PLE.04
MB <sub>17</sub>	Presença Iluminação Exterior	T_PLE.05
MB <sub>18</sub>	Sistema Detecção Fumaga	T_DF.01

Tabela 6.3: Objetos Controladores do Subsolo 1

Elemento	Tipo de Objeto	Referencia
MB <sub>1</sub>	Sistema HVAC	S1_HVAC.01
MB <sub>2</sub>	Presença(2) Iluminação Interior Vigilância	S1_SP(2).L1.03
MB <sub>3</sub>	Sistema Detecção Fumaga	S1_DF.01
MB <sub>4</sub>	Presença Iluminação Interior Vigilância	S1_PLI.01
MB <sub>5</sub>	Sistema HVAC	S1_HVAC.02
MB <sub>6</sub>	Presença Iluminação Interior Vigilância	S1_PLI.02
MB <sub>7</sub>	Presença Iluminação Interior Vigilância	S1_PLI.03
MB <sub>8</sub>	Presença Iluminação Acesso Vigilância	S1_PLA.01
MB <sub>9</sub>	Presença Iluminação Acesso(2) Vigilância	S1_PLA(2).01

Figura 6.28: Sistema de Comunicação dos Objetos Controladores Subsolo 1

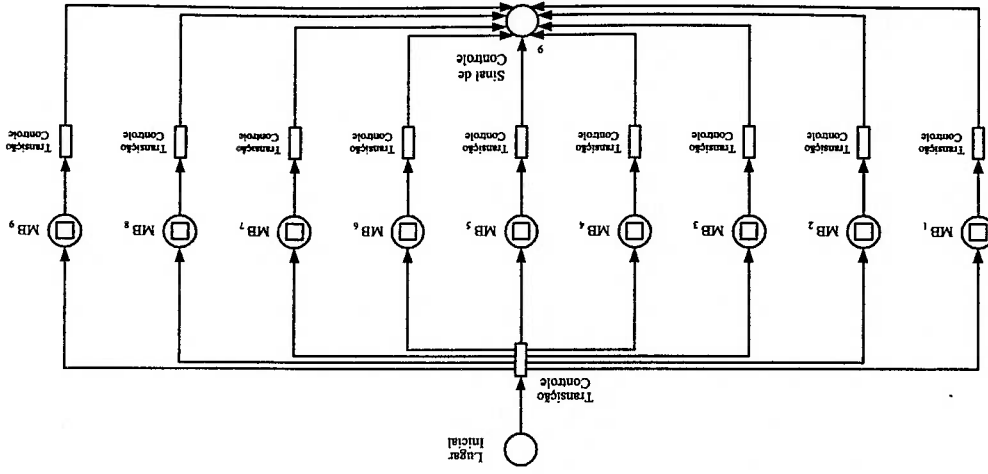


Figura 6.27: Sistema de Comunicação dos Objetos Controladores do Andar Terreo

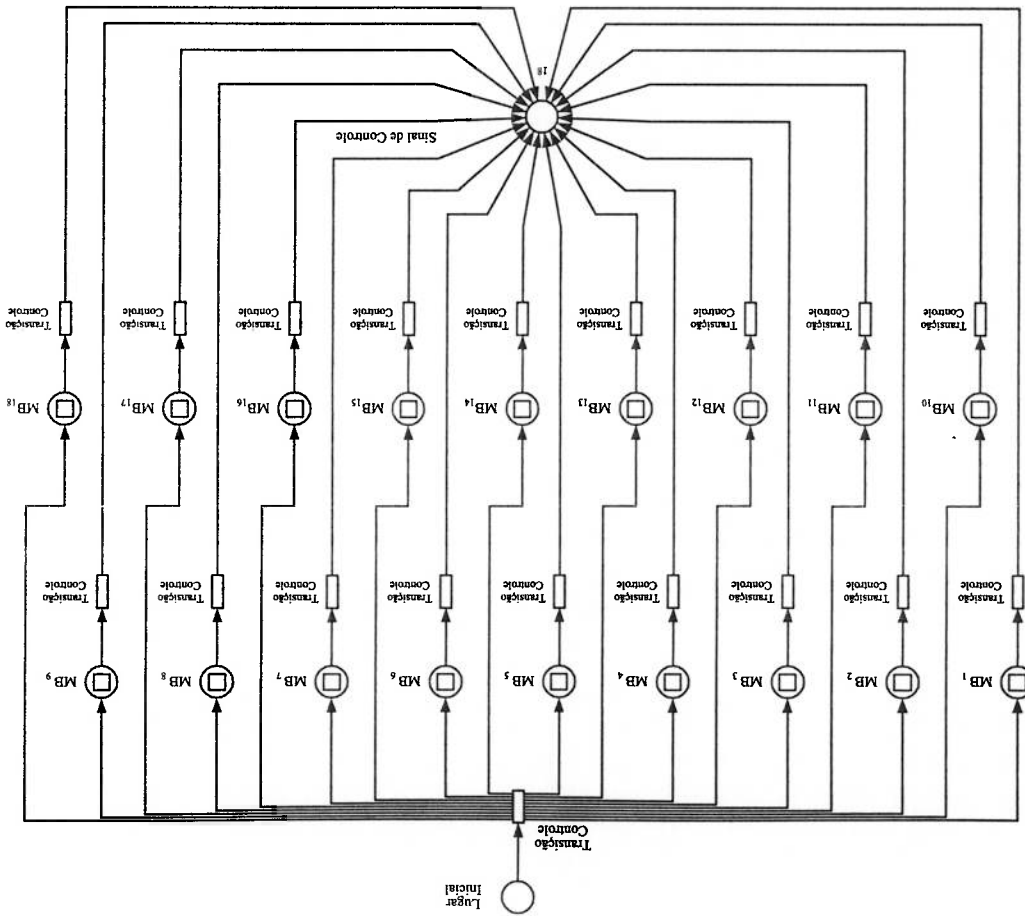
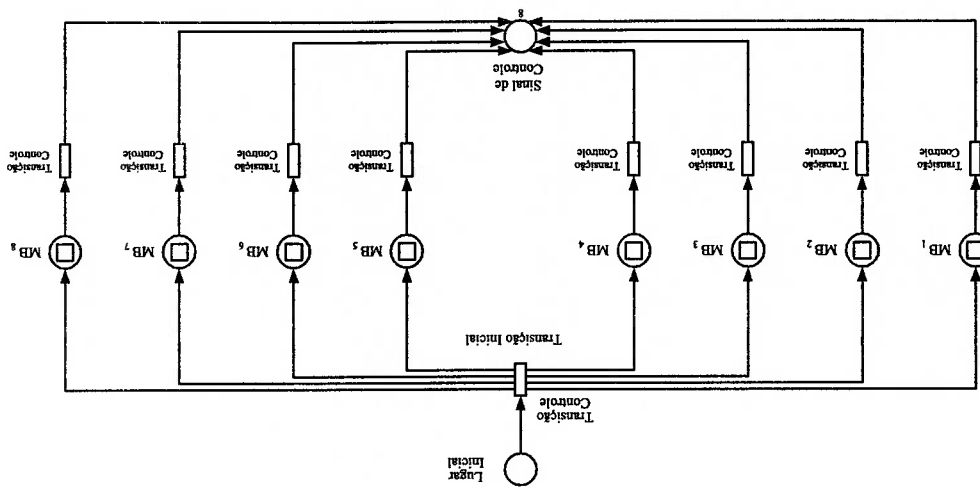


Tabela 6.4: Sistema de Comunicação dos Objetos Controladores Subsolo 2

Elemento	Tipo de Objeto	Referencia
MB <sub>1</sub>	Presença Iluminação Acesso Vigilância	S2.PLA.01
MB <sub>2</sub>	Presença Iluminação Interior Vigilância	S2.PL.I.01
MB <sub>3</sub>	Presença Iluminação Acesso Vigilância	S2.PLA.02
MB <sub>4</sub>	Sistema HVAC	S2.HVAC.01
MB <sub>5</sub>	Sistema Detecção Fumaça	S2.DF.01
MB <sub>6</sub>	Presença Iluminação Interior Vigilância	S2.PL.I.02
MB <sub>7</sub>	Presença Iluminação Interior Vigilância	S2.PL.I.03
MB <sub>8</sub>	Sistema Acesso Vigilância	S2.SA.02

Tabela 6.5: Objetos Controladores do Subsolo 2



Assim a árvore de hierarquia da rede *GENEsys* que modela o controle do prédio automatizado objeto de estudo neste trabalho apresenta-se na figura 6.29. O nó inicial da rede se apresenta na figura 6.30

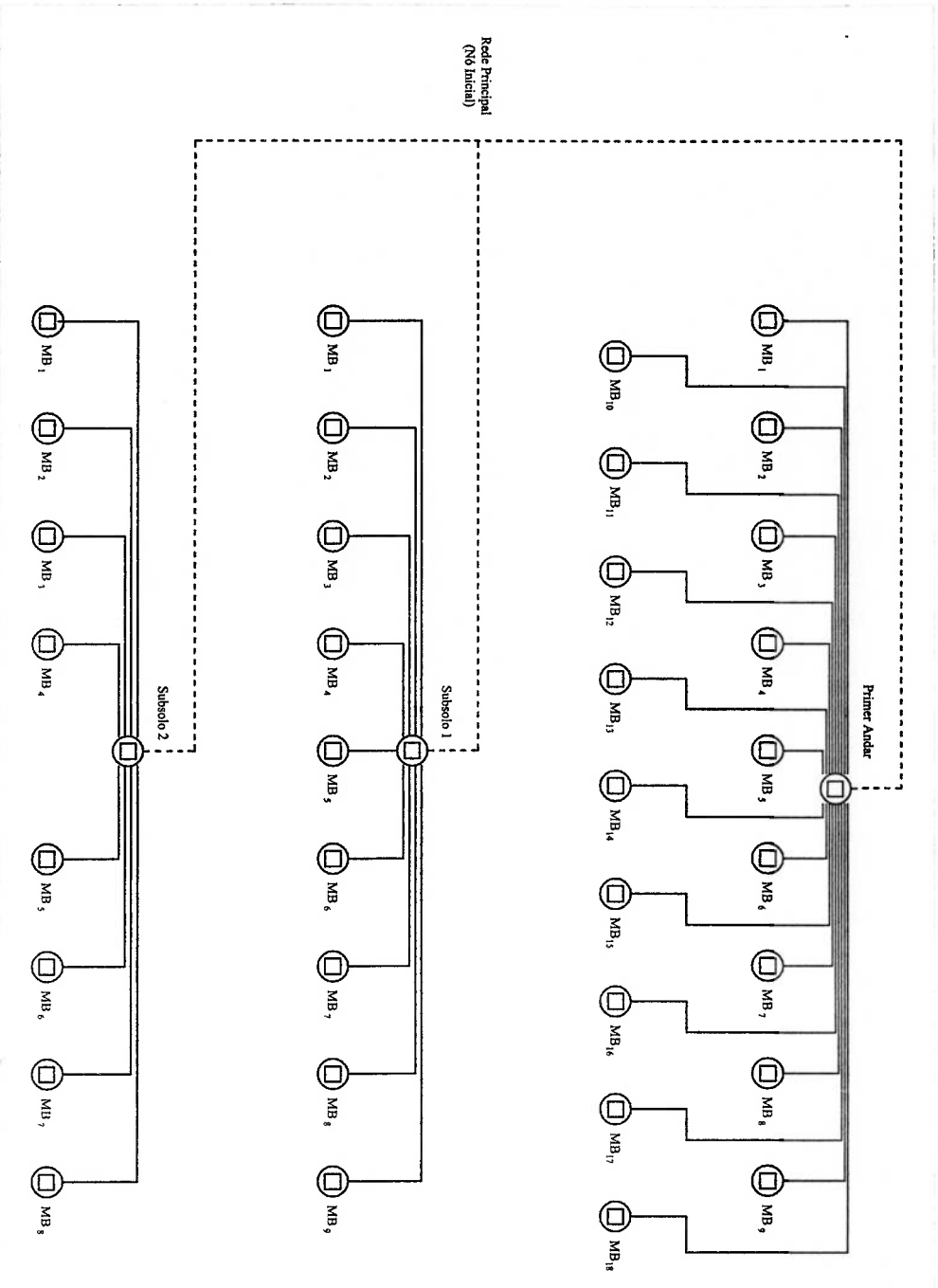
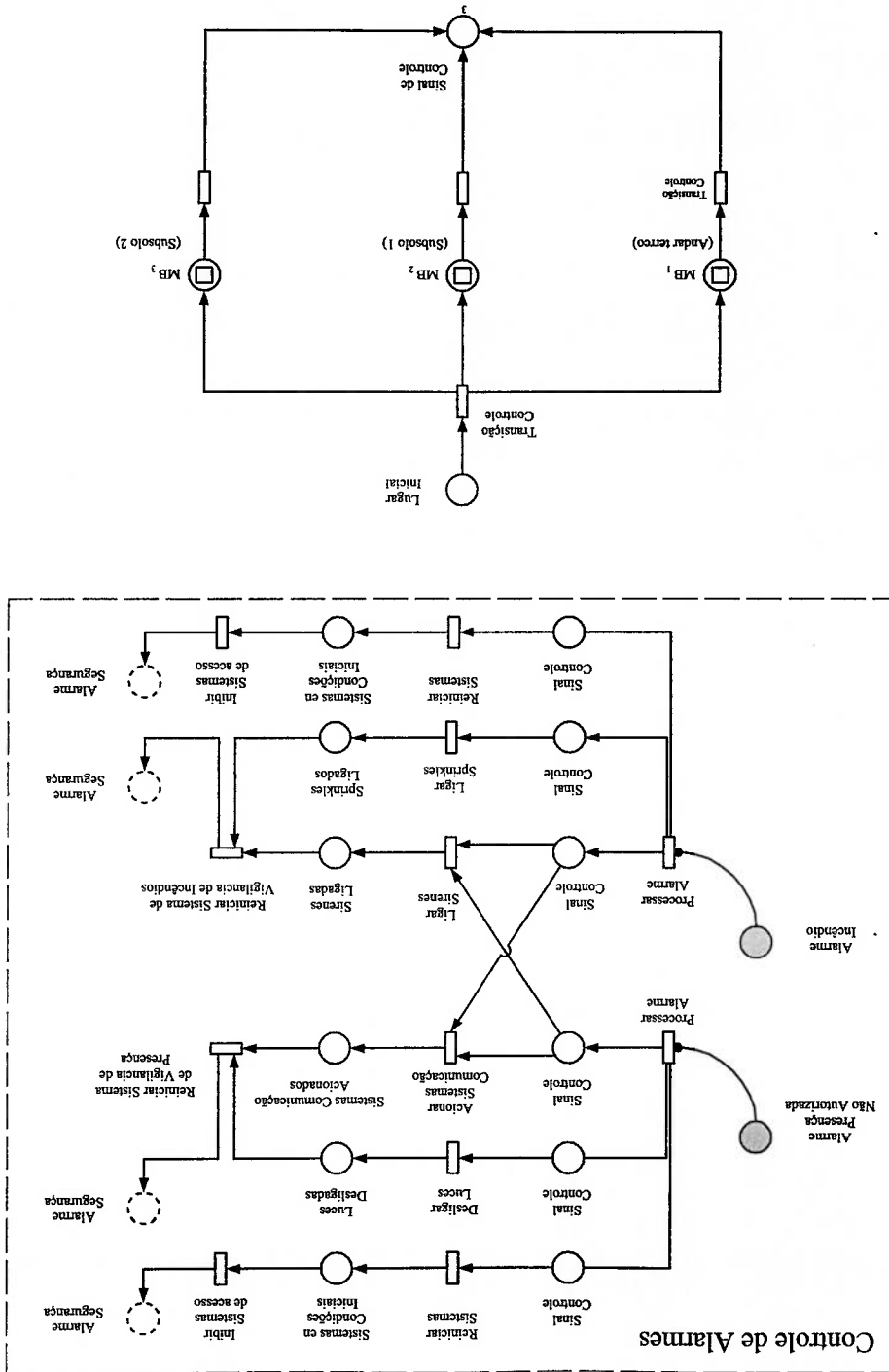


Figura 6.29: Árvore de Hierarquia do Estado de Caso

Figura 6.30: Nó Inicial do Modelo de Controle



## 7. CONCLUSÕES E TRABALHOS

### FUTUROS

#### 7.1 Conclusões

Neste trabalho foi apresentado um tipo de Rede de Petri estendida que permite a modelagem Orientada a Objetos de sistemas a eventos discretos. A rede utilizou o refinamento e a composição hierárquicos como um método para romper a complexidade na construção de modelos de grande porte.

Nesta abordagem, foi possível utilizar as redes como Objetos, permitindo a reutilização e o intercâmbio de mensagens entre redes. Assim, a construção de bibliotecas de modelos para o projeto de sistemas foi viável, com o consequente aumento na produtividade e a confiabilidade do processo de modelagem.

O trabalho apresentado foi dirigido para fusão do *Petri Net Inside Objects* e do *Object Inside Petri Nets*, não obstante, a rede não é considerada ainda de alto nível, permitindo entrar tanto a representação algébrica e a utilização dos métodos de análise das redes já consagrados.

Um jogador de marcas incorporando os algoritmos para a execução da rede foi apresentado, permitindo o uso da rede *GHENesys* para a construção e execução de modelos de sistemas a eventos discretos. A utilização de métodos de refinamento hierárquico, intercâmbio de mensagens e uso de redes como objetos são algumas das características desenvolvidas.

No estudo de caso em questão foi utilizado o conceito de *Elemento Controlador*, onde as regras heurísticas originadas de política de uso de um prédio são modeladas mediante o intercâmbio de mensagens entre objetos. O sistema supervisorio, assim construído, permitiu obter o comportamento desejado da planta (o prédio).

## 7.2 Trabalhos Futuros

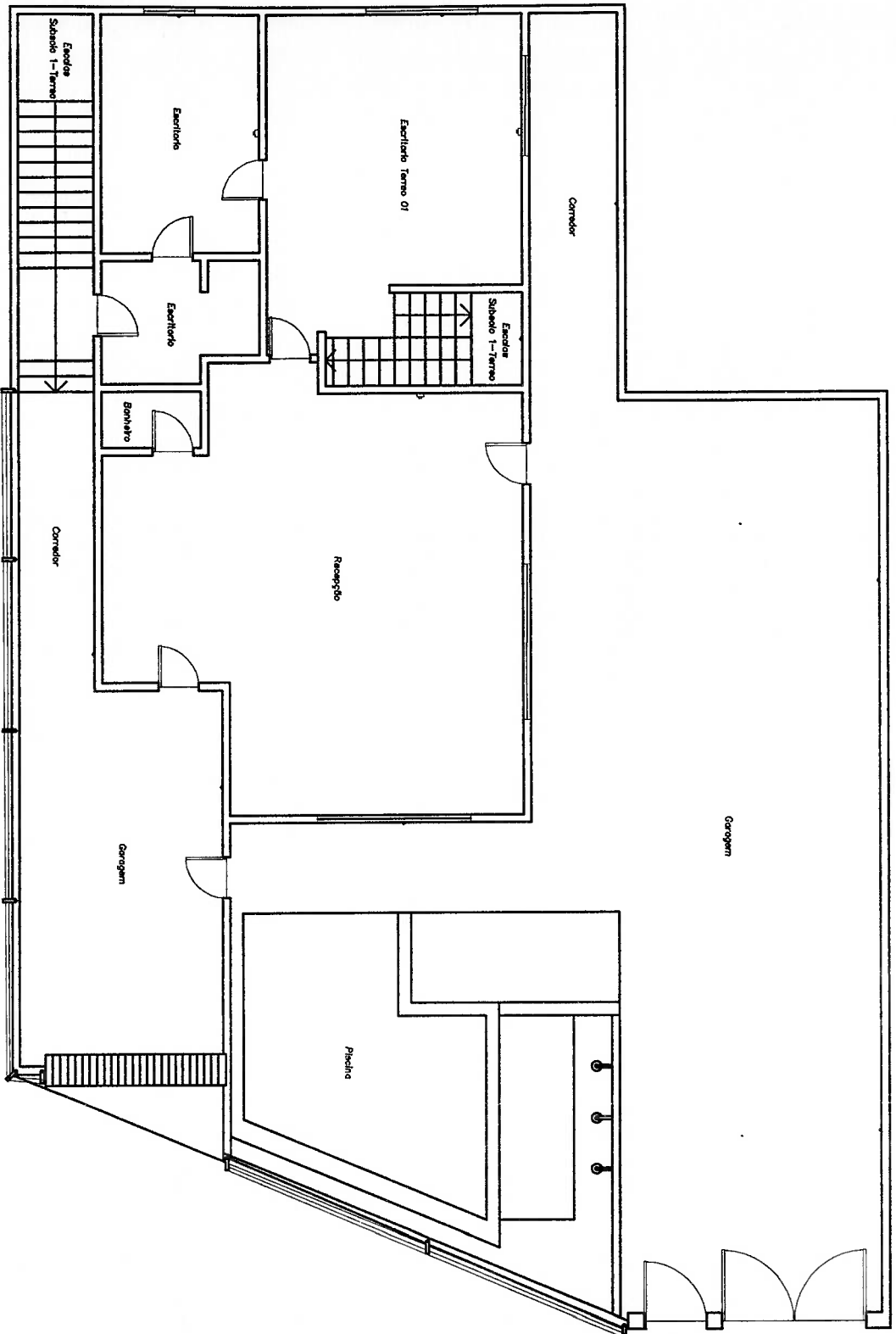
Em trabalhos futuros procurase-a mostrar que a rede *GHENeSys* pode ser transformada em uma rede de alto nível, incluindo as marcas no diagrama de classes. Se o conjunto de cores for pouco sofisticado teremos uma rede ainda simples para proceder à análise de invariantes dispensando a transformação em uma rede básica de alto nível (*Basic High Level Net*) e daí para uma rede ordinária. Neste caso é preciso tratar o problema da criação e destruição do objeto marca. Com isto poderia-se tratar a síntese de sistemas supervisórios com uma representação formal mais simples de ser interpretada.


O fato da rede apresentada permitir a composição hierárquica, abre algumas questões sobre a utilização do conceito de hierarquia na modelagem de sistemas a eventos discretos que poderiam ser tratados no futuro, tais como: a conveniência de implementar sistemas que permitam a avaliação do estado de nós na rede com prioridade sobre os demais, a reestruturação do sistema de mensagens apresentadas para que permita desmarcar elementos *Pseudobox* e a utilização de diferentes métodos de avaliação da estrutura hierárquica do projeto, além do apresentado nesta proposta (*Deep First*).

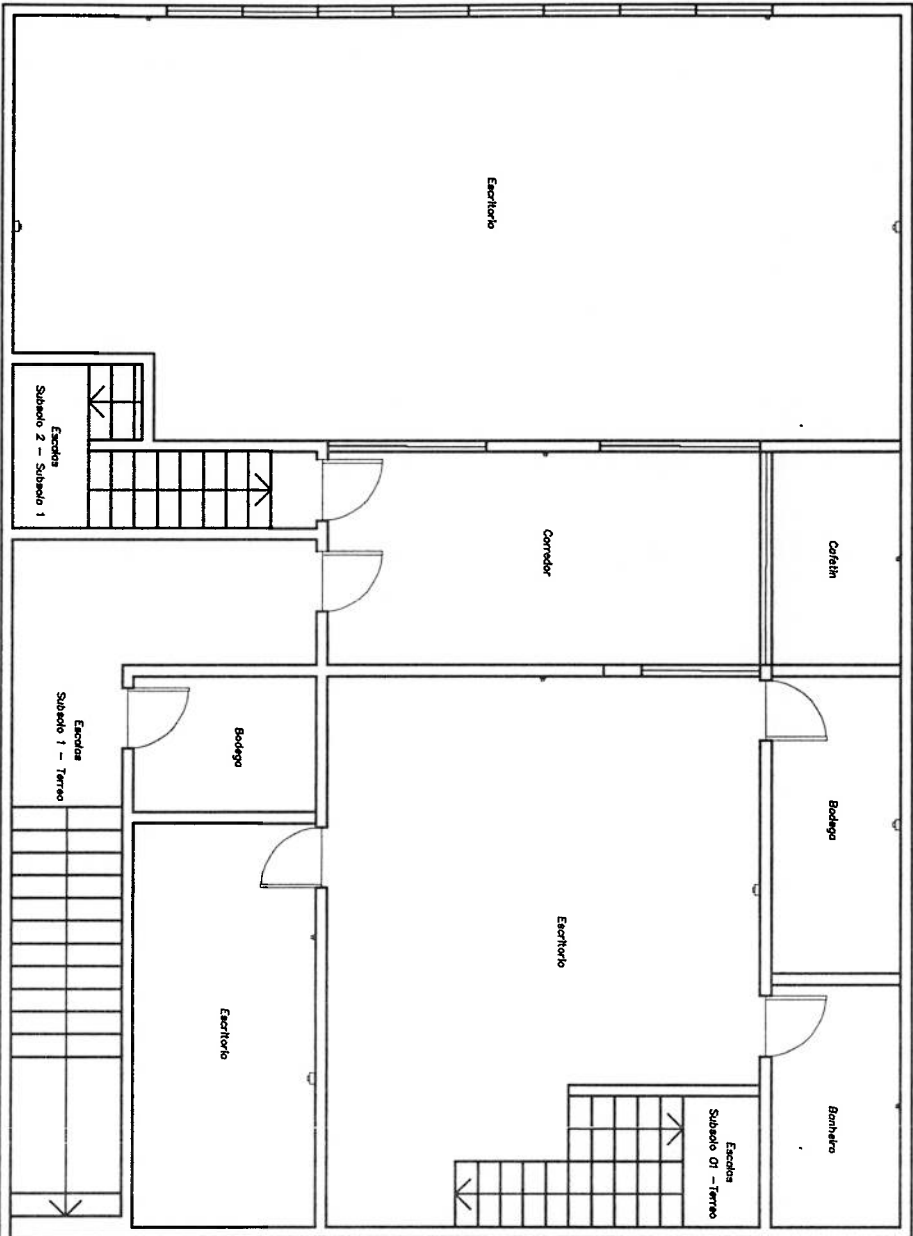
O jogador de marcas apresentado entra no processo de uso prático, com o qual o ciclo de vida do software entra na fase de correção de *Bugs* inevitáveis num software de certa dimensão. Além disso, alguns módulos secundários podem ser implementados.



PLANTAS



 <b>ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO</b>	
<b>PLANTA ANDAR TERREO</b>	
PROJETO: PEDRO L. ANGEL, RESITREDO REVISOR: PROF. DR. JOSÉ RENAULO SILVA	DESENHO: PEDRO L. ANGEL, RESITREDO SUPERVISOR: PROF. DR. JOSÉ RENAULO SILVA
PL.: A4-01	ESCALA: 1:125
DATA: 19-FEV-94	



**ESCOLA POLITECNICA DA UNIVERSIDADE DE SAO PAULO**

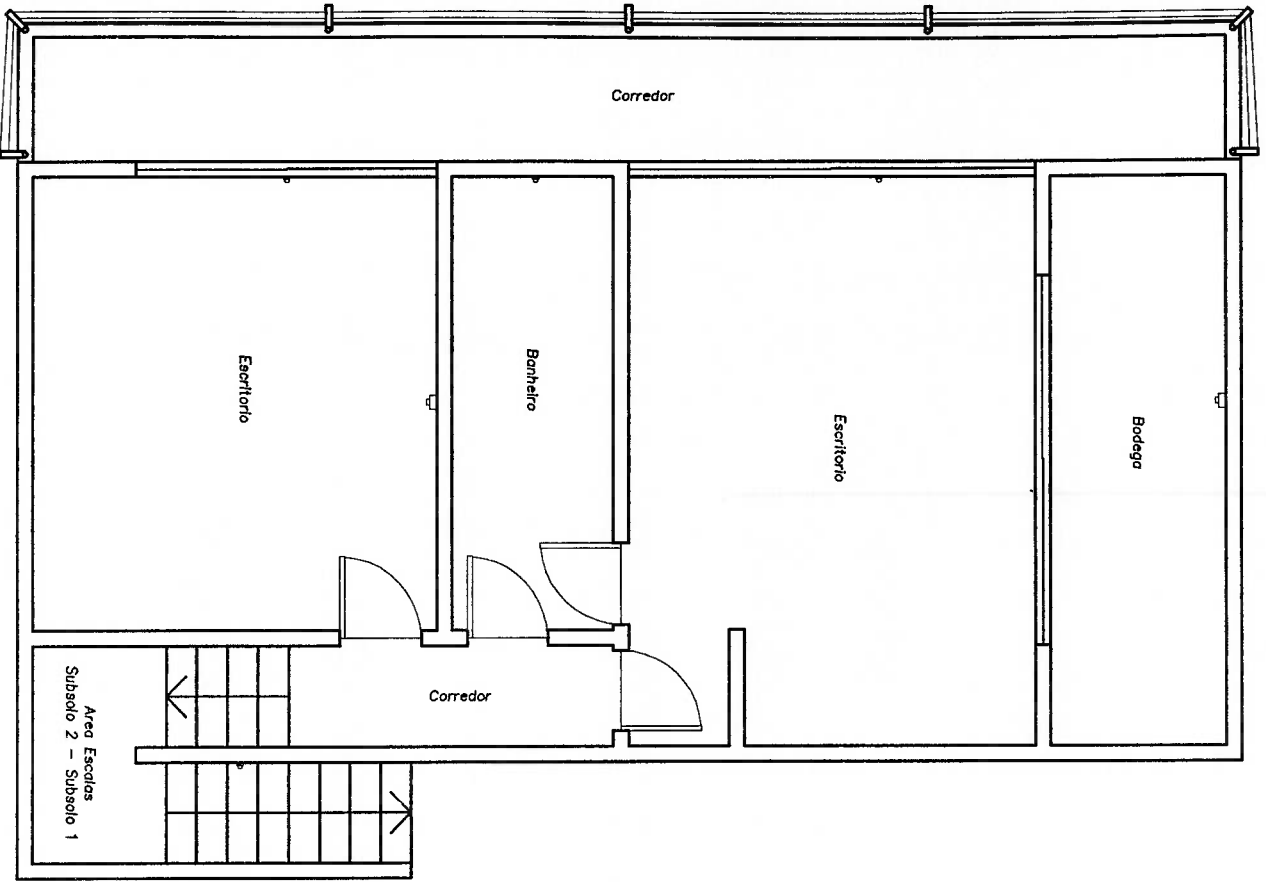
**PLANTA ANDAR SUBSUELO 1**


Pl.: 44-02

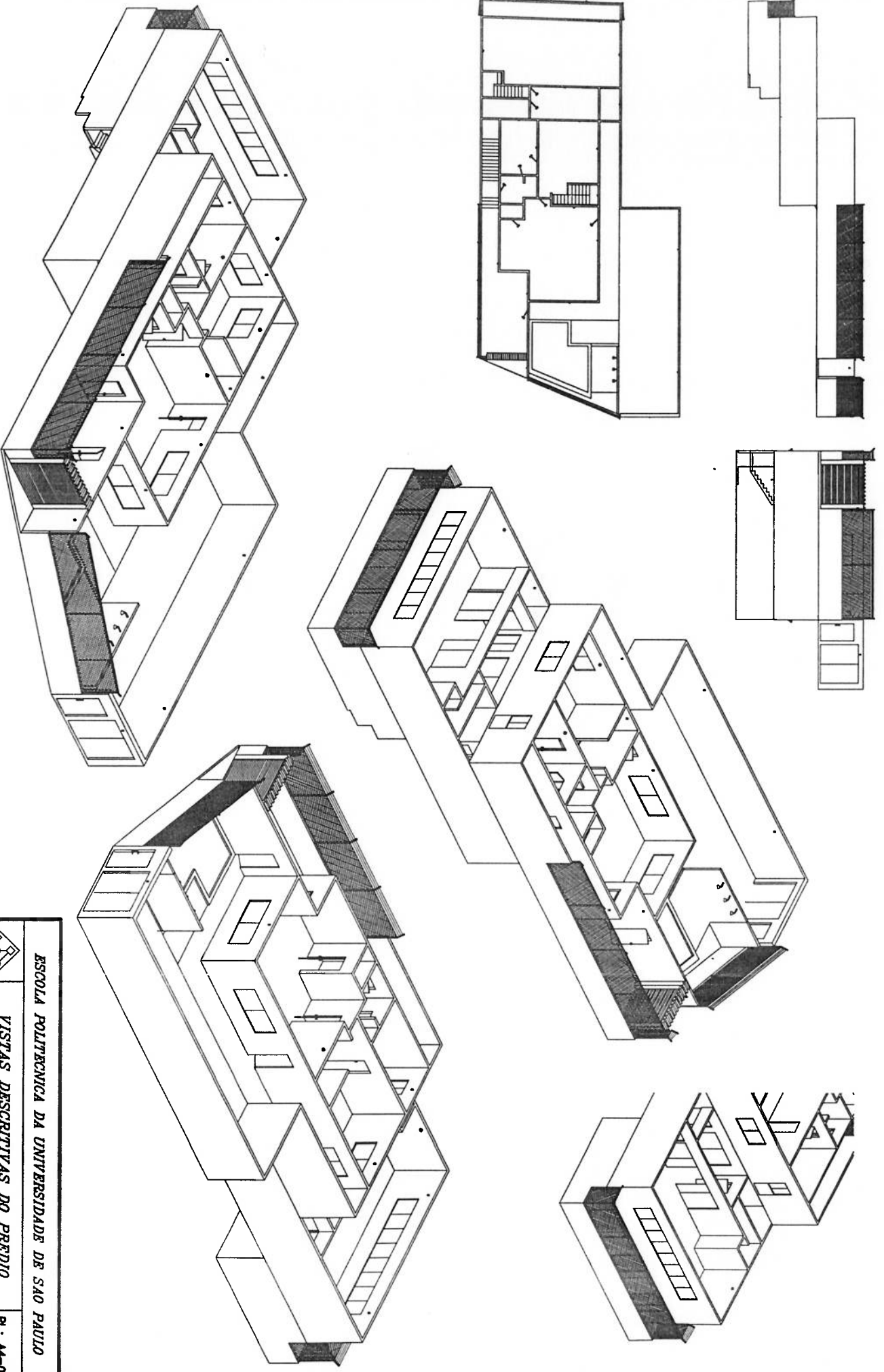


ARQUIVADO

PROJETO: PEDRO L. ANGEL RESTREPO	DESENHO: PEDRO L. ANGEL RESTREPO	ESCALA: 1:100
REVISOR: PROF. DR. JOSE RENALDO SILVA	SUPERVISOR: PROF. DR. JOSE RENALDO SILVA	DATA: 18-Fev-04



			
<b>ESCOLA POLITECNICA DA UNIVERSIDADE DE SAO PAULO</b>			
<b>PLANTA ANDAR SUBSUELO 2</b>		<b>PL.: 44-03</b>	
PROFESSOR PEDRO L. ANGEL RESTREPO	DESENHADOR PEDRO L. ANGEL RESTREPO	ESCALA: 1:75	DATA: 13-Fev-04
REVISOR Prof. Dr. JOSE RENALDO SILVA	SUPERVISOR Prof. Dr. JOSE RENALDO SILVA		

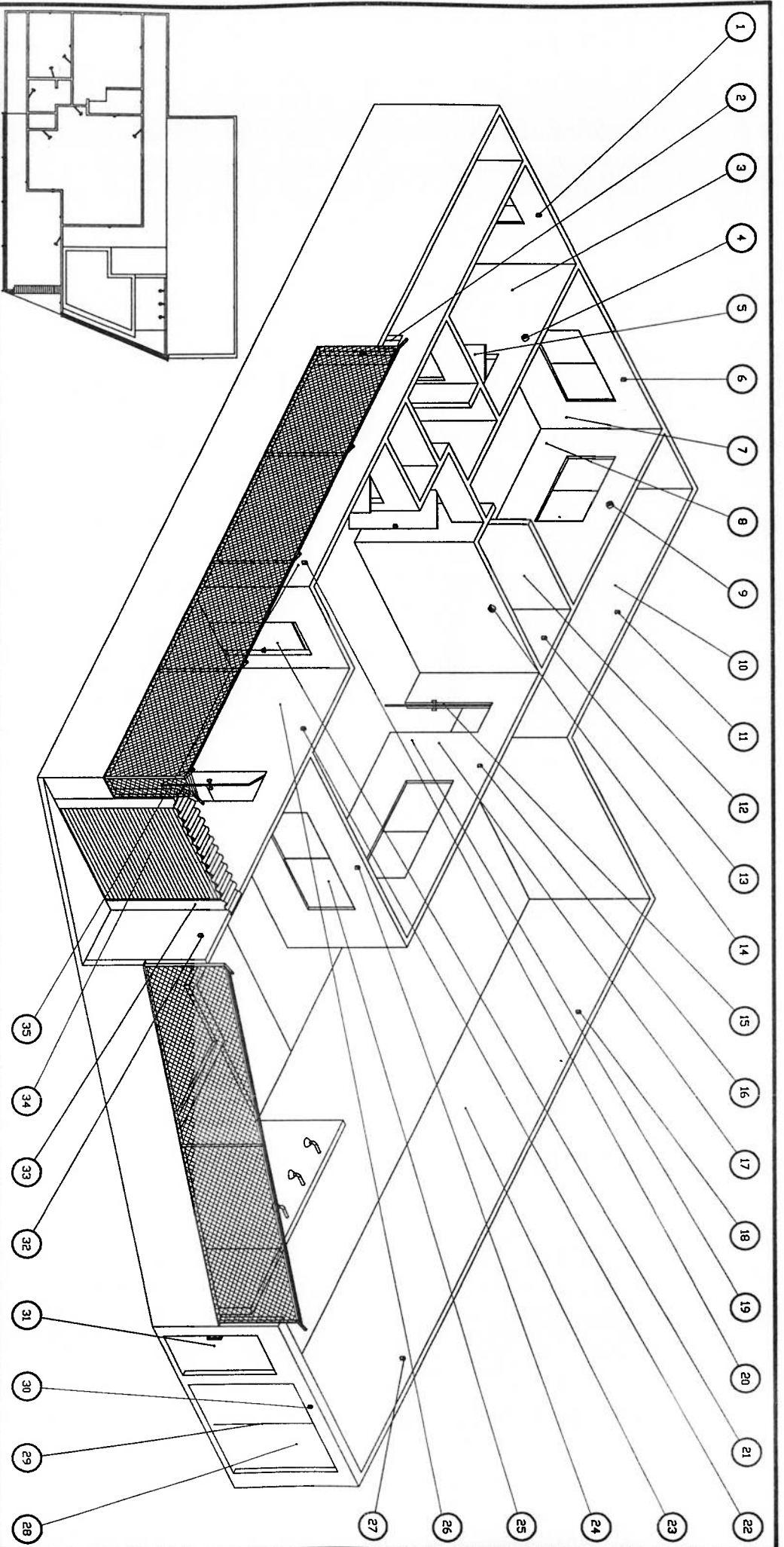


ESCOLA POLITECNICA DA UNIVERSIDADE DE SAO PAULO

VISTAS DESCRITIVAS DO PREDIO

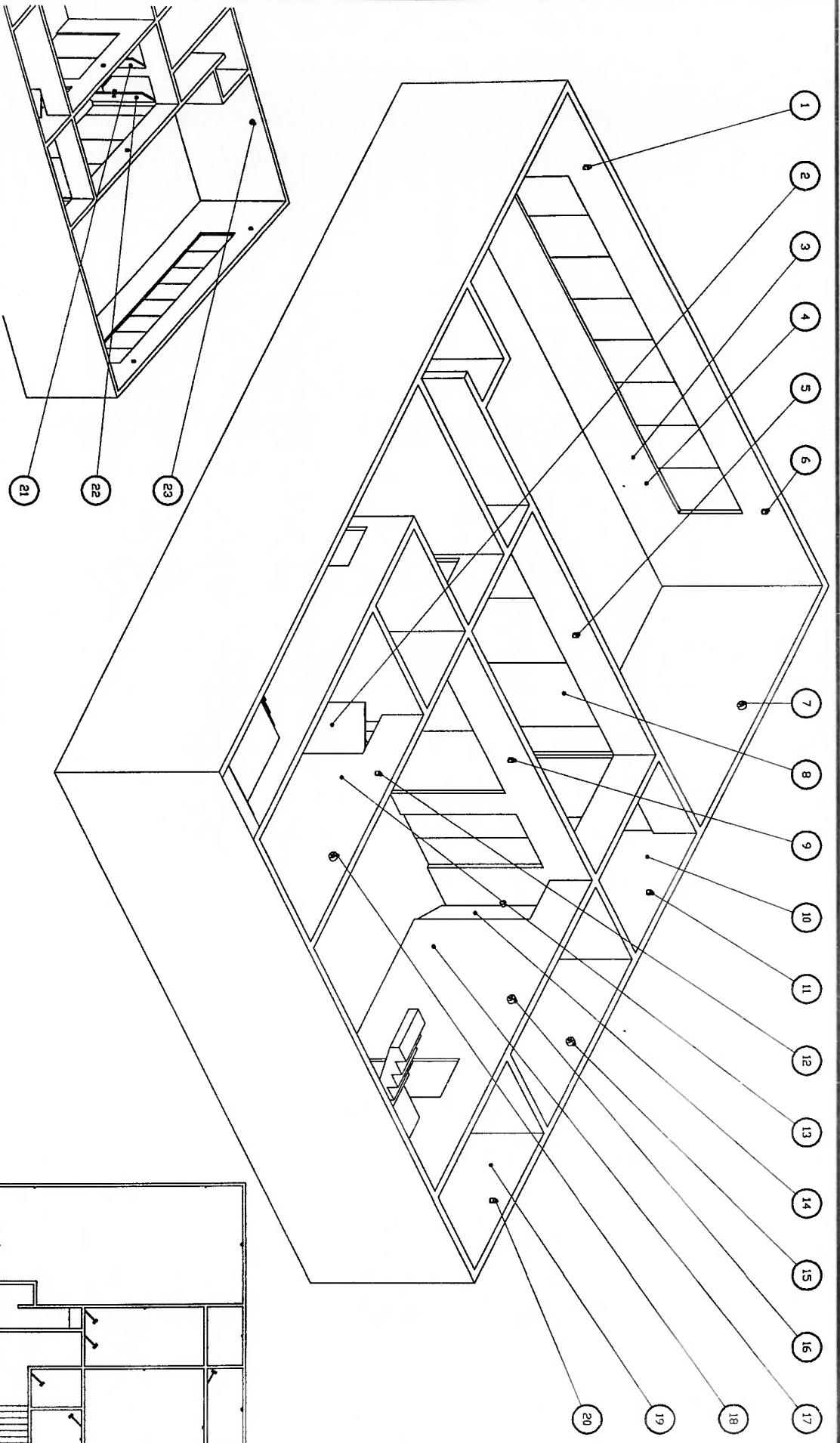
	PROJETO: PEDRO L. ANGELO RESTEMBO	REVISÃO: PEDRO L. ANGELO RESTEMBO	ESCALA:
COORDENADOR: PROF. DR. JOSÉ RENALDO SILVA	COORDENADOR: PROF. DR. JOSÉ RENALDO SILVA	COORDENADOR: PROF. DR. JOSÉ RENALDO SILVA	DATA: 19-10-64

PL: A4-04

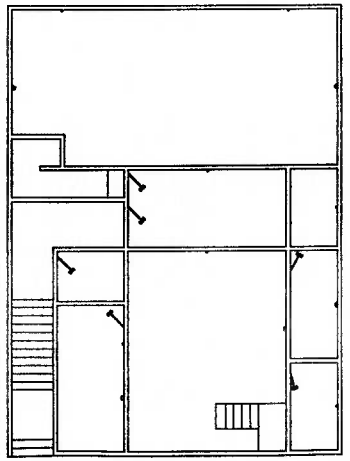



25	Sistema de Luminosidade	T.SL-11	21	Sistema de Acesso	T.SA-04	7	Sistema de Luminosidade	T.SL-02
34	Sensor de Acesso	T.SA-07	20	Sensor de Presença	T.SP-07	8	Sensor de Presença	T.SP-02
33	Sistema de Luminosidade	T.SL-10	19	HVAC - Infravermelho	T.HVAC-02	5	Sistema de Acesso	T.SA-02
32	Sensor de Presença	T.SP-11	18	Sensor de Presença	T.SP-08	4	Sensor de Presença	T.SP-01
31	Sistema de Acesso	T.SA-06	17	Sistema de Luminosidade	T.SL-05	3	Sistema de Luminosidade	T.SL-01
30	Sensor de Presença	T.SP-11	16	Sensor de Presença	T.SP-06	2	Sistema de Acesso	T.SA-01
29	Sistema de Acesso	T.SA-05	15	Sistema de Acesso	T.SA-03	1	Sensor de Presença	T.SP-01
28	Sistema de Luminosidade	T.SL-09	14	Sensor de Fumog	T.SF-03	Plano de Emergência		
27	Sensor de Presença	T.SP-10	13	Sensor de Presença	T.SP-04			
26	Sistema de Luminosidade	T.SL-08	12	Sistema de Luminosidade	T.SL-04			
25	Sistema de Luminosidade	T.SL-07	11	Sensor de Presença	T.SP-03			
24	Sensor de Presença	T.SP-09	10	Sistema de Luminosidade	T.SL-03			
23	Sistema de Luminosidade	T.SL-08	9	Sensor de Fumog	T.SF-02			
22	Sensor de Presença	T.SP-08	8	HVAC - Infravermelho	T.HVAC-01			

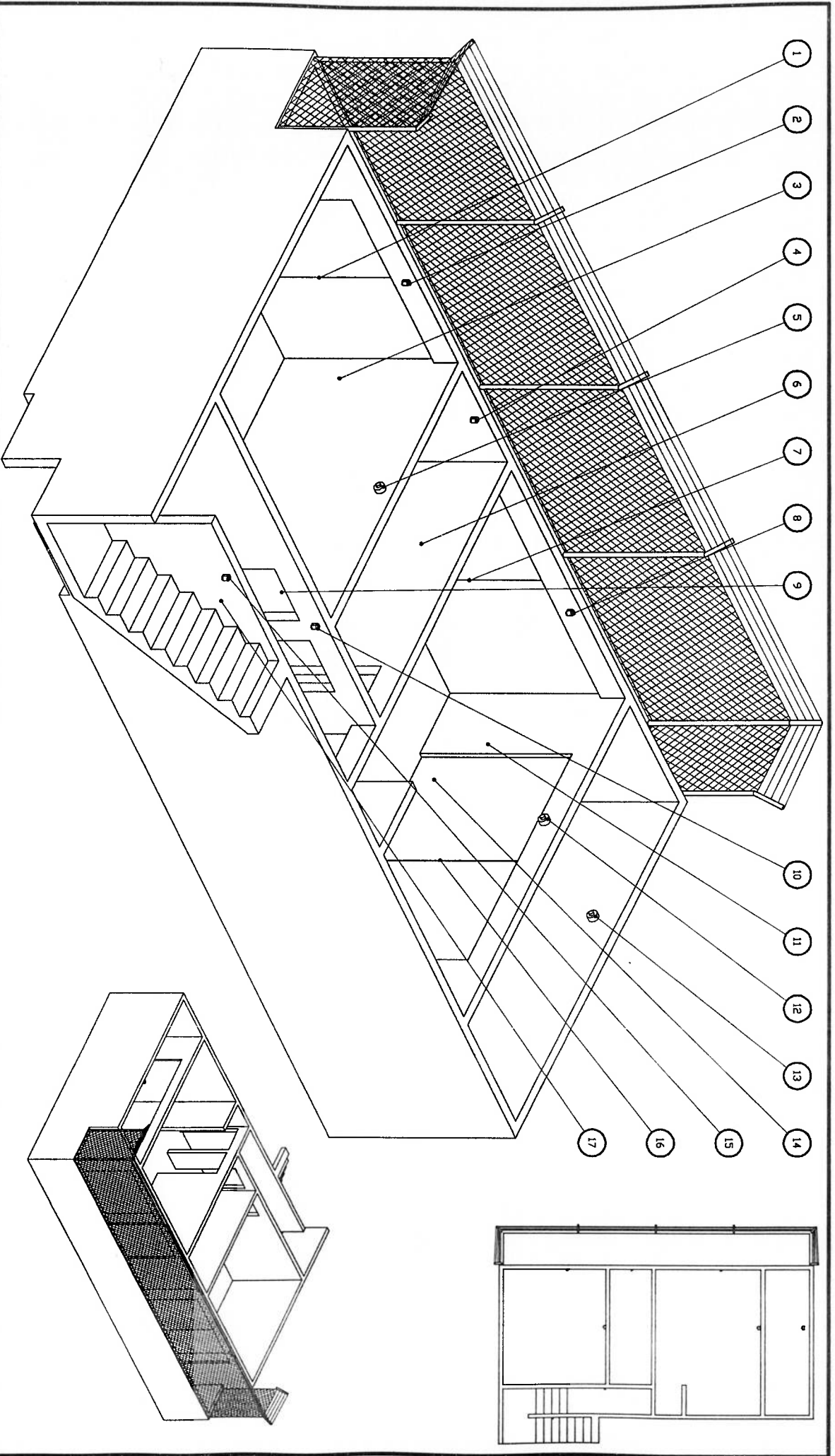

**ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO**  
**OBJETOS ANDAR TERREIRO**  
 PRODUTO PEDRO L. ANGEL RESTREPO  
 REVISÃO Prof. Dr. JOSÉ RENALDO SILVA  
 DESENHO PEDRO L. ANGEL RESTREPO  
 SUPERVISÃO Prof. Dr. JOSÉ RENALDO SILVA  
 PL: A4-05  
 ESCALA  
 DATA: 19-Fev-04



23	Sensor de Fumog	SI_SF-05	13	Sistema de Luminolidade	SI_SL-07	3	Sistema de Luminolidade	SI_SL-01
22	Sistema de Aquec	SI_SA-04	12	Sensor de Presso	SI_SP-06	2	Sistema de Aquec	SI_SA-01
21	Sistema de Presso	SI_SP-03	11	Sensor de Presso	SI_SP-05	1	Sensor de Presso	SI_SP-01
20	Sensor de Presso	SI_SP-07	10	Sistema de Luminolidade	SI_SL-03	<b>Objetos</b> Descrição:		
19	Sistema de Luminolidade	SI_SL-06	9	Sensor de Presso	SI_SP-04	<b>ESCOLA POLITECNICA DA UNIVERSIDADE DE SAO PAULO</b> <b>OBJETOS SUBSOLO 1</b>		
18	Sensor de Fumog	SI_SF-04	8	Sistema de Luminolidade	SI_SL-02	PL: A4-06 ESCALA:		
17	Sistema de Luminolidade	SI_SL-05	7	Sensor de Fumog	SI_SF-01	DATA: 18-11-04		
16	Sensor de Fumog	SI_SF-03	6	Sensor de Presso	SI_SP-03	DESIGNADO POR: DR. JOSE RENALDO SILVA		
15	Sensor de Fumog	SI_SF-02	5	Sensor de Presso	SI_SP-02	DESIGNADO POR: DR. JOSE RENALDO SILVA		
14	Sistema de Aquec	SI_SA-02	4	HVAC - Resfriador	SI_HVAC-01	DESIGNADO POR: DR. JOSE RENALDO SILVA		




**ESCOLA POLITECNICA DA UNIVERSIDADE DE SAO PAULO**  
**OBJETOS SUBSOLO 1**  
 PL: A4-06  
 ESCALA:



17	Sistema de Iluminação	S2_Sl-05	9	Sistema de Iluminação	S2_Sl-03
16	Sistema de Acesso	S2_SA-03	8	Sistema de Acesso	S2_SA-03
15	Sistema de Preencho	S2_SP-05	7	Sistema de Preencho	S2_SA-02
14	Sistema HVAC	S2_HVAC-01	6	Sistema de Iluminação	S2_Sl-02
13	Sistema de Forno	S2_SF-03	5	Sistema de Forno	S2_SF-01
12	Sistema de Forno	S2_SF-02	4	Sistema de Preencho	S2_SP-02
11	Sistema de Iluminação	S2_Sl-04	3	Sistema de Iluminação	S2_Sl-01
10	Sistema de Preencho	S2_SP-04	2	Sistema de Preencho	S2_SP-01

1	Sistema de Acesso	S2_SA-01
---	-------------------	----------

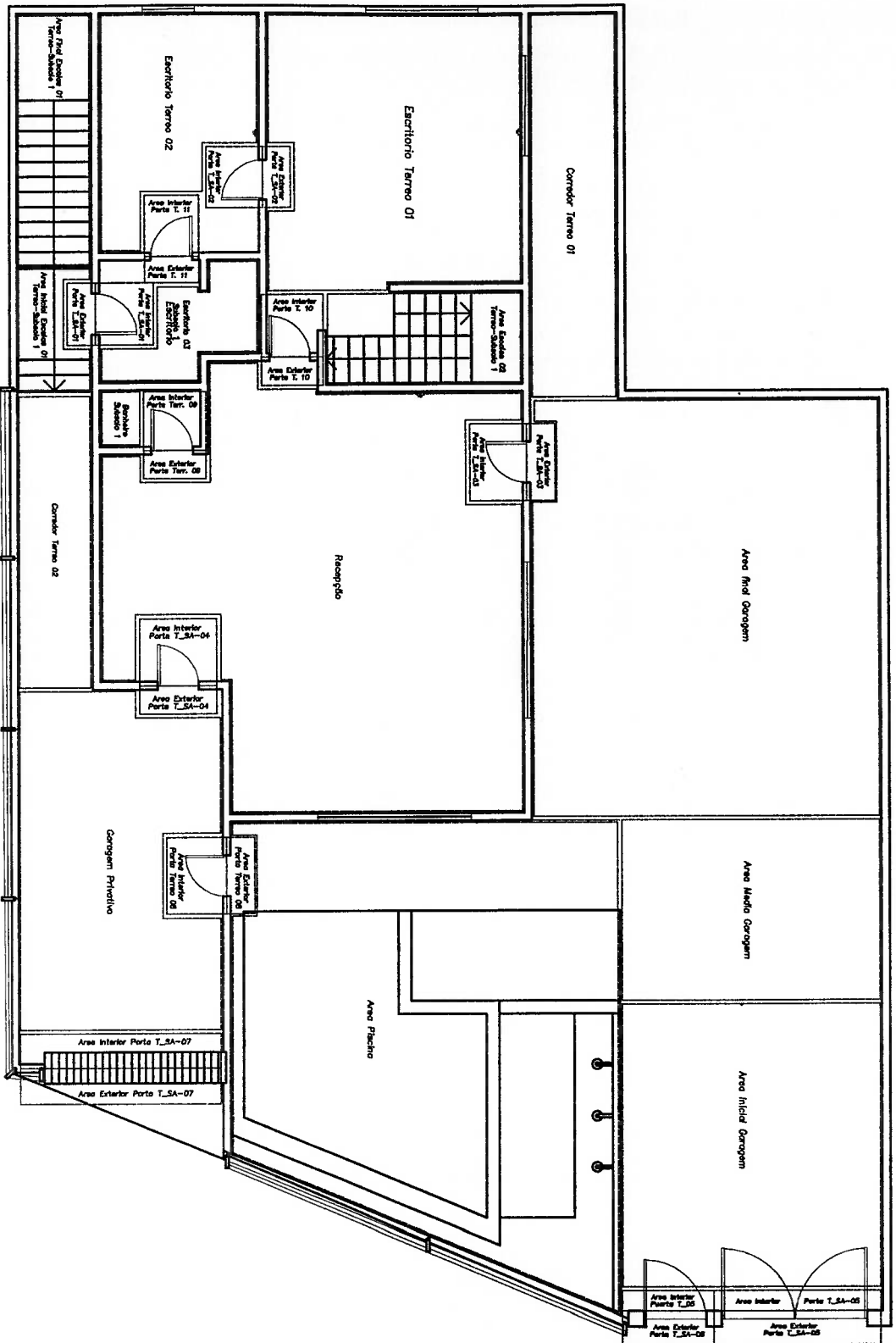
**ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO**

**OBJETOS SUBSOLO 2**

PL: A4-07

PROFESSOR RESPONSÁVEL: **PROF. DR. JOSÉ RENALDO SILVA**  
 COORDENADOR: **PROF. DR. JOSÉ RENALDO SILVA**  
 DATA: 19/04/04





**ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO**

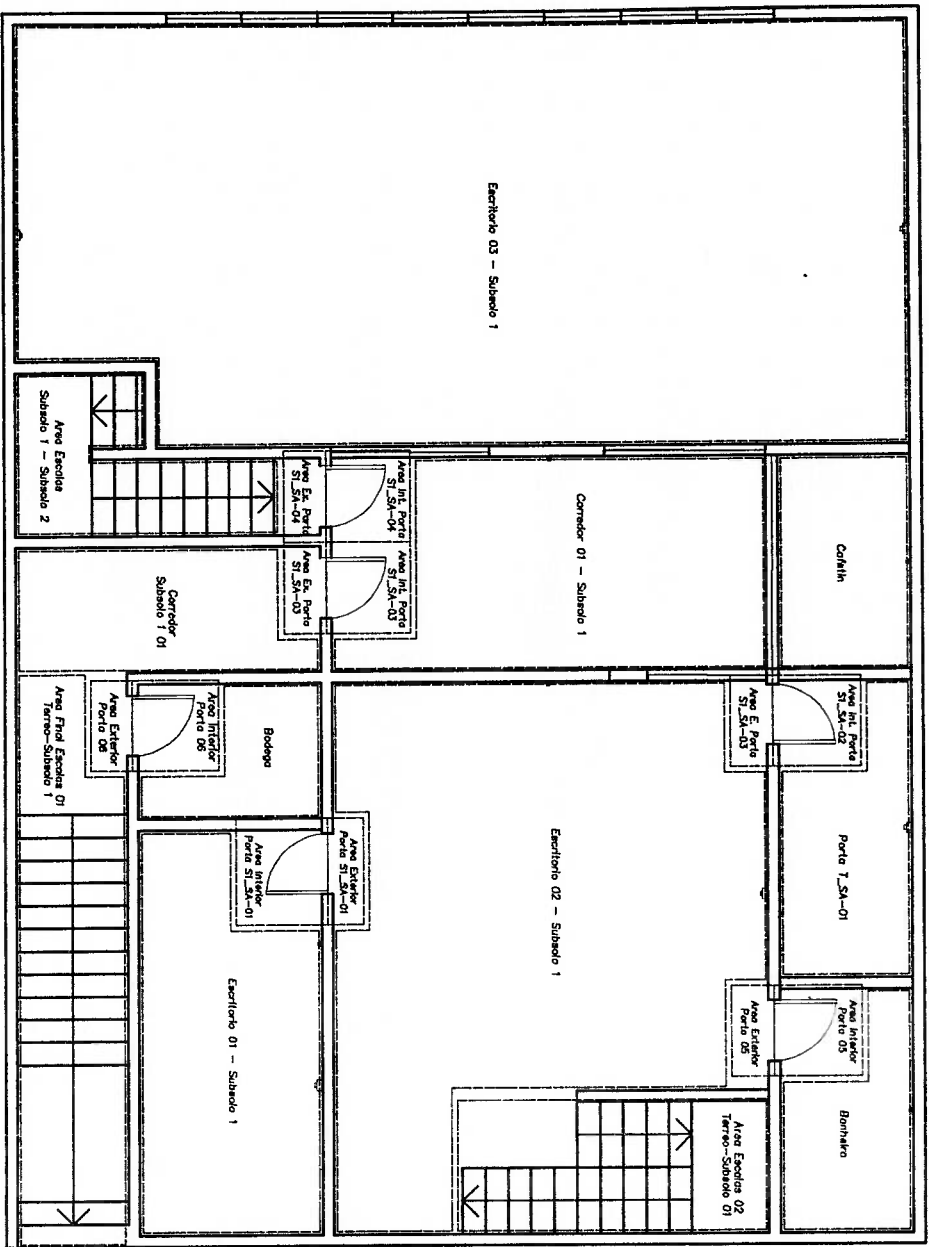
**ÁREAS ANDAR TERREO**

PL.: A4-08




BRUNO LAB  
ARQUITETURA

PROJETA: PEDRO L. ANGEL RESTREPO  
DESENHA: PEDRO L. ANGEL RESTREPO  
SUPERVISOR: PROF. DR. JOSÉ RENVALDO SILVA  
DATA: 19-FEV-04

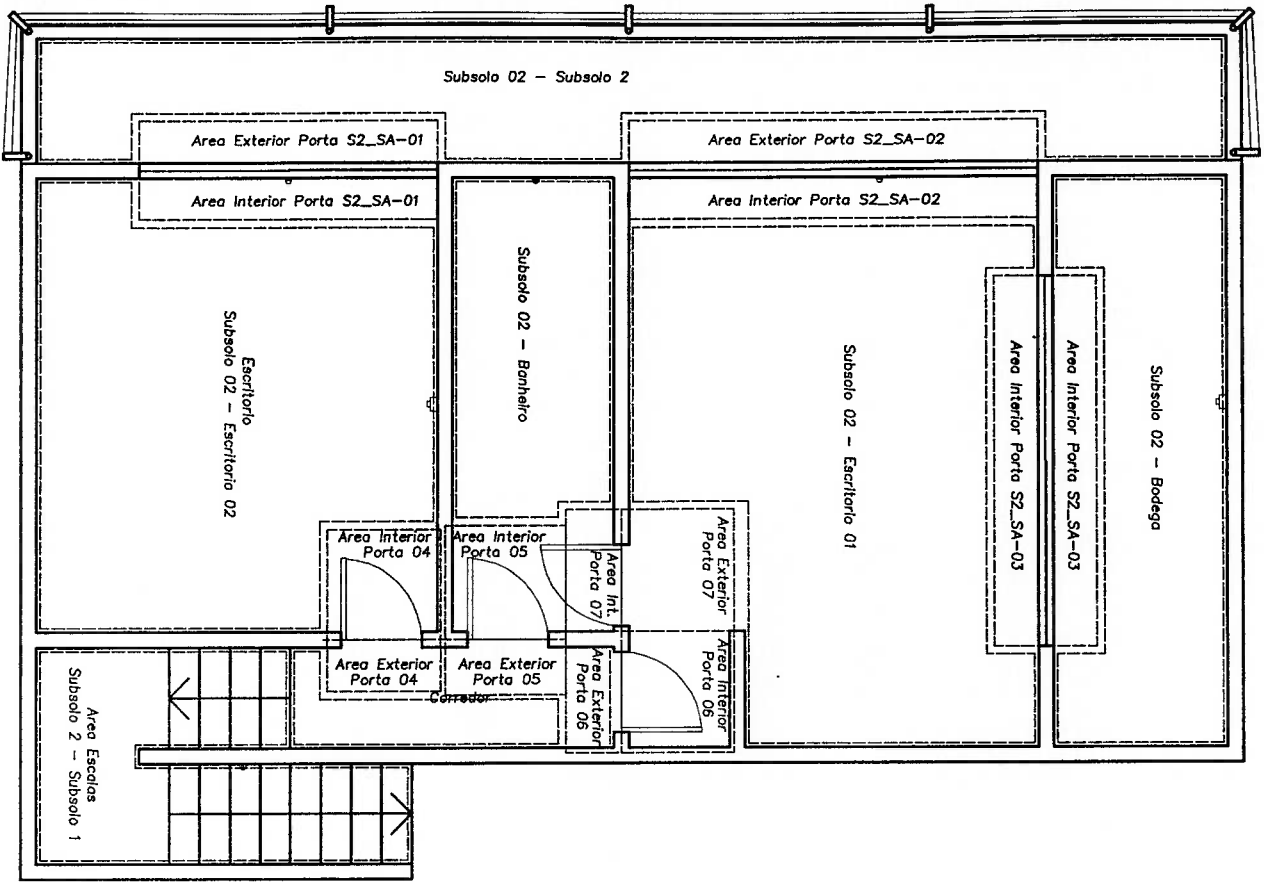



**ESCOLA POLITECNICA DA UNIVERSIDADE DE SAO PAULO**

**AREAS SUBSUELO 1**

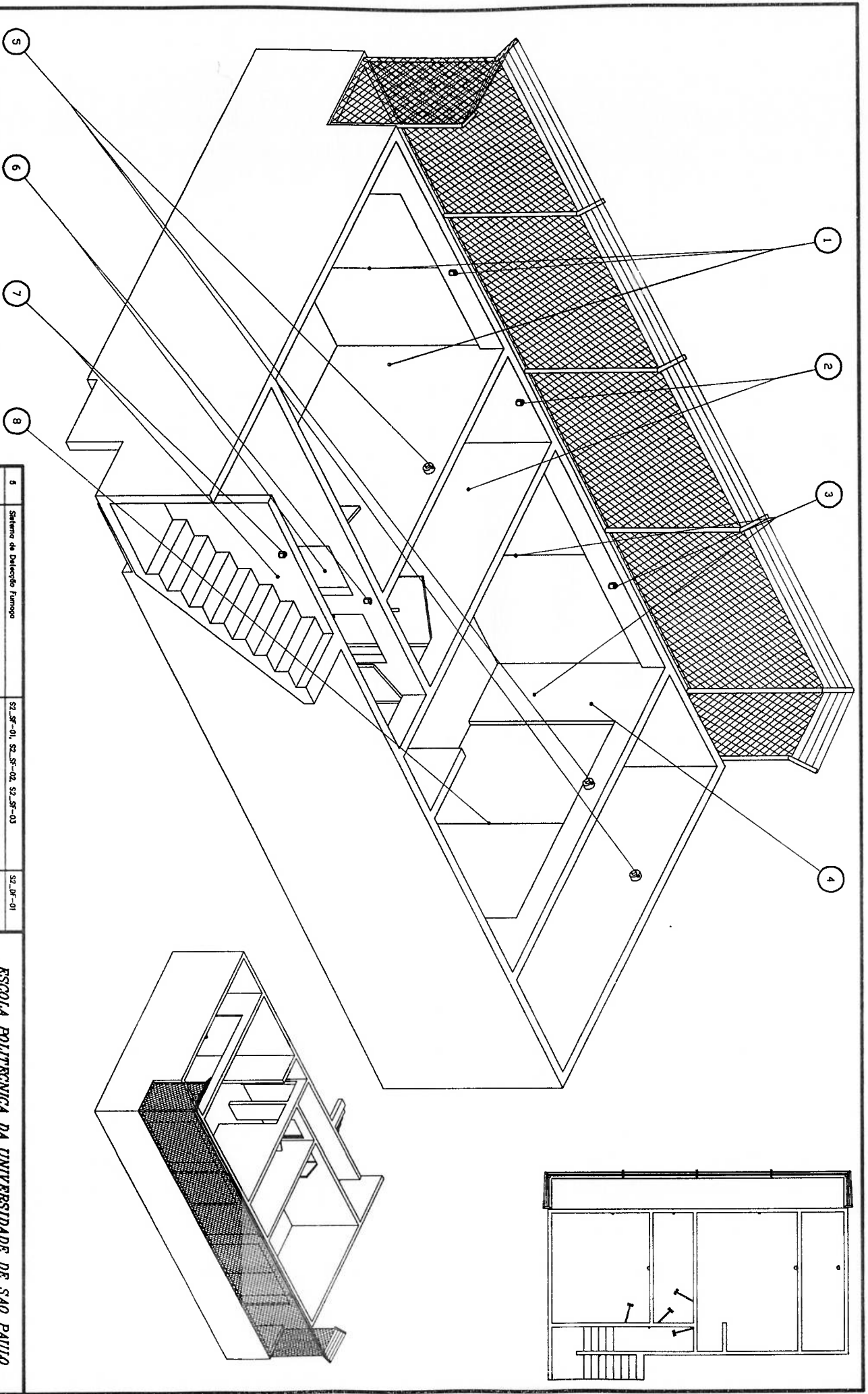
	PROJETA	DESIGNO	ESCALA
	PEDRO L. ANGEL RESTREPO RENSON Prof. Dr. JOSÉ RENVALDO SILVA	PEDRO L. ANGEL RESTREPO SUPERVISOR Prof. Dr. JOSÉ RENVALDO SILVA	1:100 DATA: 15-7-04

PL.: A4-09



 <b>ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO</b>	
<b>AREAS SUBSUELO 2</b>	PL.: A4-10
PROJETO: PEDRO L. ANGELO RESTREPO REVISOR: Prof. Dr. JOSÉ RENALDO SILVA	DESENHO: PEDRO L. ANGELO RESTREPO SUPERVISOR: Prof. Dr. JOSÉ RENALDO SILVA
ESCALA: 1:75 DATA: 19-Fev-04	





6	Sistema Asas	S2.SA-03	S2.SA-01	6	Sistema de Detecção Fumogo	S2.SF-01, S2.SF-02, S2.SF-03	S2.DF-01	 <p><b>ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO</b></p> <p><b>OBJETOS ACRESCADOS SUBSÓLO 2</b></p> <p>PL: 44-13</p>
7	Sistema Freamço-Limboado de Início	S2.SP-03, S2.SI-06	S2.PI-03	3	Sistema Freamço-Limboado-Acesso	S2.SA-02, S2.SF-03, S2.SI-04	S2.PI-02	
8	Sistema Freamço-Limboado de Início	S2.SI-03, S2.SP-04	S2.PI-02	2	Sistema Freamço-Limboado-Início	S2.SP-02, S2.SI-02	S2.PI-01	
6	Sistema Freamço-Limboado de Início	S2.SI-03, S2.SP-04	S2.PI-02	1	Sistema Freamço-Limboado-Acesso	S2.SA-01, S2.SF-01, S2.SI-01	S2.PI-01	
				4	Sistema HVAC	S2.HVAC-01	S2.HVAC-01	<p>PROFESSOR PEDRO L. ANGELO RISTENBO</p> <p>ORÇAMENTO Prof. Dr. JOSÉ RENALDO SILVA</p> <p>DESENHO Prof. Dr. JOSÉ RENALDO SILVA</p> <p>DATA: 19-F-04</p>
				5	Sistema HVAC	S2.HVAC-01	S2.HVAC-01	

## APÊNDICES

### A.1 Critérios para a Seleção do Compilador

Os compiladores, aspectos avaliados e as notas associadas para a seleção do compilador para a construção do sistema computacional *GHENESYS* são apresentadas na tabela A.2. O compilador selecionado foi o Delphi.

Tabela A.1: Critérios e Escala de Avaliação

AGENTE	FATOR	Escala da Calificación
Compilador	Nível de Dificuldade	0-5
	Bibliotecas Gráficas	0-5
	Suporte	0-5
Produto	Estabilidade	0-5
	Portabilidade	0-10
	Consumo de Hardware	0-5
	Desempenho	0-5
	Manutenção	0-5
	Ampliação Posterior	0-5
	Interação com o Hardware	0-5
	Desenvolvedor	Conhecimento da Ferramenta

É preciso considerar que aspectos como custos do compilador e do hardware, não foram avaliados pois o laboratório de automatização (DLAB) tem disponível o hardware e software necessários.

Tabela A.2: Escolha do Ambiente de desenvolvimento para a Construção da Ferramenta Computacional GHENESYS

SOFTWARE FATOR	C		C++		Visual C++		JAVA		Visual Basic		Visual Fortran		Delphi		Power Builder		
Nível de Dificuldade	1		3		3		3		4		3		4		4		4
Bibliotecas Gráficas	0		2		3		5		3		3		4		4		4
Suporte	3		4		3		5		4		2		3		2		2
Estabilidade	4		3		1		5		3		4		5		3		3
Portabilidade	8		6		0		9		0		0		7		5		5
Consumo de Hard- ware	5		5		3		1		2		3		4		4		4
Desempenho	10		10		8		3		4		9		8		6		6
Manutenção	2		3		3		4		3		5		4		4		4
Ampliação Posterior	2		2		3		4		3		3		4		3		3
Interação com o Hard- ware	5		5		5		2		4		5		4		3		3
Conhecimento da Ferramenta	2		2		2		2		5		5		1		2		2
Total	42		45		34		43		35		42		48		40		40

## Referências Bibliográficas

- [Anisimov, 1994] Anisimov N.; Kovalenko A.; Postupalski P. Compositional Petri Net Environment. *IEEE Symposium on Emerging Technologies*. 1994
- [Akimaru, 1991] Akimaru, H. Intelligent Buildings: Myth, Reality, or Wishful Thinking?. *IEEE Communications Magazine*. U.S.A. 1991
- [Basten, 2001] Basten, T.; Van der Aalst, W. M. P. Inheritance of Dynamic Behaviour: Development of a Groupware Editor, in *Lecture Notes in Computer Science 2001, Advances in Petri Nets*. Springer-Verlag, 2001.
- [Bastide, 1995] Bastide R. Approaches in unifying Petri Nets and the Object-Orientation Approach. *Proceedings of the Application and Theory of Petri Nets 1995*. Workshop on Object-Oriented Programming and Models of Concurrency. Torino, Italy, June 1995.
- [Booch, 1994] Booch, G. Object Oriented Analysis and Design with Applications, Benjamin-Cummings, 1994.
- [Booch, 1994] Booch, G. Object-Oriented Analysis and Design. Addison-Wesley Publishing Menlo Park, California, 1994. 589p.
- [Brandin, 1992] Brandin B. A.; Wonham W. M. The Supervisory Control of timed Discrete Event System. *Proceedings of the 31st Conference on decision and control*. p. 3357-3362. Tucson Arizona 1992.
- [Brandin, 1992a] Brandin B. A.; Wonham W. M.; Benhabib B. Manufacturing Cell Supervisory Control A timed Discrete Event System. *Proceedings of the 1992 International Conference on Robotics and Automation*. p. 931-936, Nice France, 1992.
- [Brauer, 1990] Brauer W.; Gold R. Vogler W. A survey of Behaviour and Equivalence Preserving Refinements of Petri Nets. *Advances in Petri Nets 1990, LNCS 483*. p. 1-46. Springer-Verlag, 1990.
- [Bourdon, 2002] Bourdon S. E.; Lawford M.; Wonham W. M. Robust Nonblocking Supervisory Control of Discrete-Event Systems. *Proceedings Of the American Control Conference Anchorage*, p. 730-735. Alaska 2002.



- [Burdea, 1994] Burdea, G.; Coiffet P. **Virtual Reality Technology**. John Wiley and Sons Inc, New York. 1994. 400p.
- [Basten, 2001] Basten, T., Van der Aalst, W. M. P. **Inheritance of Dynamic Behaviour: Development of a Groupware Editor. Lecture Notes in Computer Science 2001. Advances in Petri Nets**, Springer-Verlag, 2001.
- [Cassandras, 1993] Cassandras, C.G. **Discret Event Systems: Modeling and Performance Analysis**. Aksen Associates Incorporated Publishers, Burr Ride, Illinois, USA. 1993. 790p.
- [Calkin, 1998] Calkin, D.W.; Parkin, R.M.; Safaric, R.; Czarniecki, C.A. **Visualization, simulation and control of a robotic system using Internet technology, Proceedings of the 5th International Workshop on Advanced Motion Control**. Coimbra, Portugal June-July 1998.
- [Cerdá, 1989] Instituto Cerdá - Área de Telecomunicaciones, **Prédios y Areas Inteligentes - Definición de un concepto emergente**. Idefons Cerdá, Ira Edición, Octubre 1989, Barcelona, España.
- [Cortés, 2001] Cortés L.A.; Eles P.; Peng Z.; Hierarchical Modeling and Verification of Embedded Systems. **Proceedings Euromicro Symposium on Digital Systems Design, 2001, 2001**.
- [Crock, 1998] Crock, S. The use of simulation and virtual reality in the design and operation of airport terminals. **Proceedings of International Conference on Simulation '98**, York, U.K. October, 1998.
- [Fabian, 1995] Fabian M.; Lennartson B. Applying Supervisory Control to Discrete Event Systems Modeled by Object Oriented Principles. **Proceedings ETTA'95**, Paris, France, Oct 1995.
- [Finley, 1991] Finley, Jr.; M. R., Karakura, A.; Nbogni, R. Survey of Intelligent Building Concept. **IEEE Communications Magazine**. p. 18-23, 1991.
- [Freund, 2001] Freund, E.; Rossmann, J. Multimedia and Virtual Reality Techniques for the Control of ERA, the First Free Flying Robot in Space. **Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation**. p. 1921-1926 vol.2, Seoul, Korea. May, 2001.

- [Frost, 2000] Frost, P.; Warren, P. Virtual Reality Used in a Collaborative Architectural Design Process. *Proceedings IEEE International Conference on Information Visualization, 2000.* p. 568-573. USA, 2000.
- [Forte, 2001] Forte, M.; Pietroni, E.; Rufa, C.; Bizzarro, A.; Tilia S. DVR-Pompei Project: Reconstructions of Archaeological Contexts in Desktop-OpenGl Environments. *Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage, 2001*, p. 307-314. ACM Press, New York, NY, USA, 2001.
- [Fujie, 1991] Fujie, S.; Mikami, Y.; Construction Aspects of Intelligent Buildings, *IEEE Communications Magazine*, p. 50-57. 1991.
- [Genrich, 1981] Genrich, H. J.; Lautenbach, K. System modelling with high-level Petri nets. *Theoretical Computer Science*, p. 109-136, 1981.
- [Giraud, 2003] Giraud C.; Valk R. Petri Net for System Engineering. Springer, Germany 2003.
- [Gina, 1991] Gina A.; DiCesare F. Supervisory Design Using Petri Nets. *Proceedings of the 30th Conference on Decision and Control* p. 92-97. Brighton, England, 1991.
- [Gina, 1994] Gina A.; DiCesare F.; Blocking and Controllability of Petri Nets in Supervisory Control. *IEEE Transactions on Automatic Control* Vol. 39 No. 4, p. 818-823. April 1994.
- [Holloway, 1994] Holloway, L. E.; Krogh, H.; Controlled Petri Nets: a Tutorial Survey, *Lecture Notes in Control and Information Sciences* 1994, 11th International Conference on Analysis and Optimization of Systems, p. 158-168. 1994.
- [Hix, 1999] Hix, D.; Swan, J.E.; II; Gabbard, J.L.; McGee, M.; Durbin, J.; King, T.; User-centered design and evaluation of a real-time battlefield visualization virtual environment, *Proceedings Virtual Reality, 1999.* , p. 96-103, March, 1999.
- [Jaworski, 1997] Jaworski, J. *JAVA Guia de Desarrollo*. Prentice Hall, Madrid. 1997. 718p.

- [Jensen, 1997] Jensen, K. Coloured Petri Nets; Basic Concepts, Analysis Methods and Practical Use EATCS Monographs on Theoretical Computer Science, Springer, 1992.
- [Lakos, 1991] Lakos C.A., Keen C.D. Simulation with Object-Oriented Petri Nets. Australian Software Engineering Conference, Sydney, Australia 1991.
- [Lakos, 1995] Lakos C.A. The Object Orientation of Object Petri Nets/ Department of Computer Sciences, University of Tasmania. 1995.
- [Li, 2002] Li, Fung-Chun.; Angelier, J.; Deffontaines, B.; Hu Jyr-Ching; Hsu Shih-Hao; Lee, Chin-Hui; Huang, Chia-Hui; Chen, Cheng-Hung. A virtual Reality Application for Distance Learning of Taiwan Stream Erosion in Geosciences. Proceedings. International Conference on Computers in Education, 2002. P. 1156-1160 vol.2, Dec. 2002.
- [Lind, 1996] Lind M., Status and Challenges of Intelligent Plant Control. Proceedings Of the 4th IFAC Symposium on Dynamics and Control of Chemical Reactors, Distillation Columns and Batch Processes. Helsinki, P. 23-41. 1996.
- [Malhotra, 1993] Malhotra M., Trivedi K. S. A Methodology For Formal Expression of Hierarchy in Model Solution, Proceedings 5th International Workshop on Petri Nets and Performance Models, Toulouse, pages 258-267. October 1993
- [Marca, 1988] Marca D. A.; McGowan C. L.; SADT : Structured Analysis and Design Techniques, McGraw-Hill, New York, 1988.
- [Modjeska, 2000] Modjeska D.; Waterworth J.; Effects of Desktop Design on User Navigation and Search Performance, Proceedings IEEE International Conference on Information Visualisation, IV2000. London, England, 2000.
- [Moody, 2000] Moody J. O.; Antsaklis P. J. Petri net supervisors for DES with uncontrollable and unobservable transitions, IEEE Trans. Automatic Control. Vol. 45, pp. 462-476, Mar. 2000.
- [Murata, 1989] Murata, T. Petri Nets: Properties, Analysis and Applications, Proceedings of the IEEE, Vol. 77, No. 4, p. 541, 1989.

- [Navinchandra, 2002] Navinchandra K. P.; Simon P.C; Terrace, F. Evaluating the use of Virtual Reality as a tool for briefing clients in architecture. **Proceedings of the Sixth Conference on Information Visualisation IV'02**. p. 657-664. London, England, 2002.
- [Peters, 1993] Peters L.; Schultz R. The Application of Petri-Nets in Object-Oriented Enterprise Simulations. **Proceedings of the 27th Annual Hawaii Conference 1993**. Hawaii, 1993.
- [Pilgrim, 2001] Pilgrim M.; Bouchlaghem D.; Loveday D.; Holmes M. A Mixed Reality System for Building Form and Data Representation. **Proceedings of Fifth International Conference on Information Visualisation, 2001**. London, England, 2001.
- [Pilgrim, 2000] Pilgrim M.; Bouchlaghem D.; Loveday D. Holmes M. Abstract Data Visualisation in the Built Environment. **Proceedings. IEEE International Conference on Information Visualization** p. 126 - 134, Julio 2000.
- [Peterson, 1981] Peterson, J.L. Petri Net Theory and the Modeling of Systems. Englewood Cliffs, N.J. Prentice-Hall Inc. 1981. 290p.
- [Ramadge, 1989] Ramadge P. J. G.; Wonham W. M, The Control of Discrete Event Systems, **Proceedings of the IEEE Communications Magazine**. January 1989, Pages 81-97.
- [Ramos, 1998] Ramos, R.B.; Silva, J.R. A Formal Model for Integrated Complex Dynamic Systems, **IMS'98, Workshop on Intelligent Manufacturing Systems**. Gramado (Brasil), December, 1998.
- [Ramos, 1999] Ramos, R.B.; Silva, J.R. An Integron-Based Architecture for the Modeling of Complex Dynamic Systems, **SBAI'99, Brazilian Symposium of Automation**. São Paulo, September, 1999.
- [Ramos, 2001] Ramos, R.B.; Silva, J.R. Integron Based Control Architecture for AUV's, **11th International Offshore and Polar Engineering Conference**. Stavanger, Norway, June, 2001.
- [Rumbaugh, 1991] Rumbaugh, J.; Blaha, M.; Premerlani, W.; Eddy, F.; Lorenzen, W.; **Object-Oriented Modeling and Design**. Prentice-Hall, 1991.

- [Satava, 1998] Satava, R.M.; Jones, S.B.; Current and Future Applications of Virtual Reality for Medicine, **Proceedings of the IEEE**. Volume: 86 Issue: 3, P. 484-489, March 1998.
- [Satava, 1997] Satava, R.M.; Jones, S.B.; Virtual Reality for Medical Applications, Information Technology Applications in Biomedicine. **Proceedings of the IEEE Engineering in Medicine and Biology Society Region 8 International Conference 1997. ITAB '97**, P. 19-20, Sept. 1997.
- [Schiel, 1990] Schiel U. Net Hierarchies **Proceedings of the International Conference on Databases, Parallel Architectures and Their Applications**. Los Alamitos, CA, USA: IEEE Comput. Soc. Press, 1990.
- [Shelbourn, 1998] Shelbourn, M.; Aouad, G.; Hoxley, M. An Integrated Virtual Reality and Case-Based Reasoning Model for Training in Building Pathology. **Proceedings of the IEEE Conference on Information Visualization**, 1998 p. 44-49, July 1998.
- [Silva, 1996] Silva, J.R.; Ramos, R. L.; Miyagi, P. E. Supervisory Control Of Integrated Building Systems: A Balanced Approach, **Balanced Automation II : Implementation Challenges for Antropocentric Manufacturing**, ed.London, Chapman and Hall, 1996.
- [Silva, 2003a] Silva, J.R.; Poli M.; Restrepo, P.A. Towards a Modeling Discipline for Building and Residence Automation. **Annals of 17th International Conference in Mechanical Engineering**, Sao Paulo, 2003.
- [Stipidis, 1998] Stipidis, E.; Li Shuming; Powner E.T. Intelligent Building Systems: System Integration using ATM, **IEEE Conference** p. 349-358. 1998.
- [Tait, 1992] Tait A.; Desktop virtual reality. **Proceedings of the IEEE Colloquium on Using Virtual World**, pp. 5/1-5/5, London, UK, 1992.
- [Tianmiao, 2001] Tianmiao W.; Da L.; Lei H.; Hongbo, Lv; Zingang W.; Zesheng T. A Simulation and Training Systems of Robot Assisted Surgery Based on Virtual Reality. **Proceedings of the International Workshop on Medical Imaging and Augmented Reality (MIAR '01)**. Shatin, N.T., Hong Kong, June 2001.
- [Valette, 1979] Valette R. Analysis of Petri Nets by Stepwise Refinement, **Journal of Computer and System Sciences**. Vol. 18, p 35-46, Fevereiro 1979.

- [Whyte, 1999] Whyte, J.; Bouchlaghem, D.; Thorpe, T. Visualization and Information: a Building Design Perspective. **Proceedings. 1999 IEEE International Conference on Information Visualization** p. 104-109. Nice France, July 1999.
- [Whyte, 1998] Whyte, J.; Bouchlaghem, D.; Thorpe, T. Visualising Residential Development Using Desktop Virtual Reality. **Proceedings. 1998 IEEE International Conference on Information Visualization** p. 40-43. July 1998.
- [Wong, 1995] Wong R. Advanced Object-Oriented techniques for Modeling Robotic Systems, **Proceedings. 1995 IEEE International Conference on Robotics and Automation** p. 1099-1104, 1995.
- [Zapf, 1999] Zapf M.; Heini A. Techniques for Integrating Petri-Nets and Object Oriented Concepts. **Working papers in Information Systems**, Agosto 1999.
- [Zhou, 1995] Zhou M.; Wang D. T. Design of Petri Net Objects and their application in command and Control Systems. **Proceedings IEEE International Conference on Systems, Man, and Cybernetic**, Vol. 5, p. 3463-3468, Vancouver, Canada, 1995.
- [Zuzuki, 1983] Zuzuki I.; Murata T. A method for Stepwise Refinement and Abstraction of Petri Nets. **Journal of Computer and Systems Sciences**. Vol. 27, p. 51-76, Agosto 1983.