

Instituto Politécnico do Porto
Instituto Superior de Engenharia

Departamento de Engenharia Electrotécnica
Engenharia Electrotécnica - Electrónica & Computadores
Automação e Sistemas

Laboratório de Sistemas



Gestor De Missões

Realizado Por:

José Eduardo Barbosa Leite Nunes nº1010615
Ricardo Filipe Pinto Moreira Guimarães nº1010896

Planeador

Esta parte do sistema terá como principal função o planeamento da missão a realizar pelo veículo autónomo. Isto será executado antes do veículo autónomo entrar em funcionamento, visto que normalmente existe um conhecimento prévio das tarefas necessárias a realizar para um correcto cumprimento dos objectivos da missão.

Para o nosso caso definimos que o planeador irá introduzir num ficheiro, com uma organização previamente estipulada, todos os dados referentes a missão que se pretende executar, sequência de funções que se podem chamar, valores das condições iniciais no caso de serem necessárias e condições de teste, referidas mais a frente. Assim sendo no caso de existir a necessidade de se alterar uma missão não será necessário compilar o código, bastando simplesmente alterar esse ficheiro.

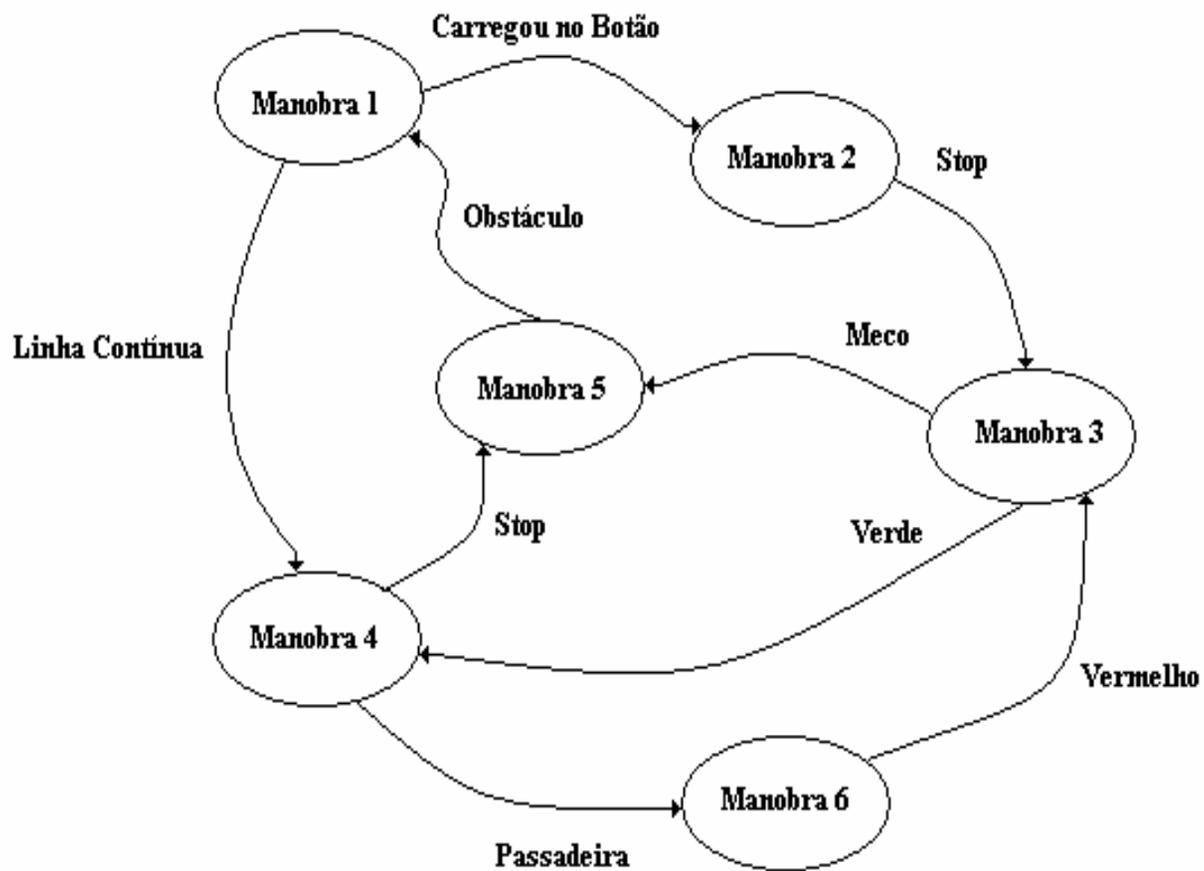
Missão

A missão será tudo o que o veículo autónomo irá realizar ou precisará de saber para completar a sua tarefa (os seus objectivos). A execução da missão vai exigir uma tarefa de pilotagem onde se articula as diferentes modalidades de funcionamento disponíveis para o mesmo.

Cada missão será constituída por um conjunto de manobras, as quais poderão ter um diverso conjunto de acções, transições ou ocorrências, que o fará transitar para uma manobra específica.

No nosso caso, e como pensado a missão será lida segundo algumas normas a partir do ficheiro que o planeador escreveu.

Grafo Direccional de uma Missão



Projectar o Gestor de Missões

Depois de definido de como seria lida a missão, vamos definir o modo de execução da mesma. Para isso foram criadas diversas estruturas, muitas delas encadeadas, contendo diversos apontadores e variáveis de estado, sendo o conteúdo destas preenchido aquando da leitura da missão.

Definido

Planeador – Parte do sistema que terá como principal função o planeamento da missão a realizar pelo veículo autónomo.

Missão – A missão será tudo o que o veículo autónomo irá realizar ou precisará de saber para completar a sua tarefa (os seus objectivos).

Ficheiro de Missões – Ficheiro com uma organização previamente estipulada, contendo todos os dados referentes a missão que se pretende executar, sequência de funções que se podem chamar, valores das condições iniciais no caso de serem necessárias e condições de teste (transições).

Manobra – Uma manobra é constituída por acções, que irão fazer com que o veículo autónomo execute uma função, contendo ainda transições para poder executar outras manobras.

Acções – As acções serão constituídas por todas as funções para o controlo do veículo autónomo, e concretização da missão.

Transições – São conjuntos de funções de teste de alguns parâmetros do veículo autónomo, para transitar de manobra para manobra ou funções de execução de parâmetros específicos (exemplo: se um veículo detecta que há pouca luminosidade executa função que acenderá as luzes do mesmo).

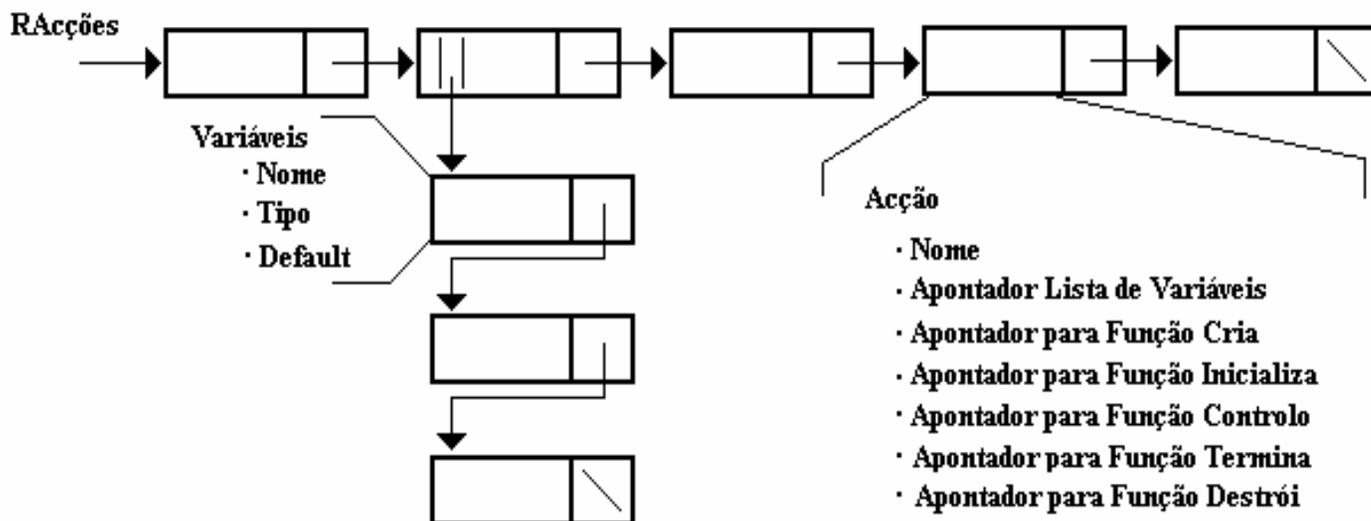
Registo das Funções Essenciais

A quando da leitura do ficheiro de missões, será necessário ter já dois registos, um com todas as acções disponíveis e outro com todas as transições que poderão existir, como o referido anteriormente. Deste modo, quando se ler num ficheiro uma determinada manobra com uma acção e suas transições, procurar-se-á nesses registos suas correspondências, não só para verificar a sua existência, como também serão utilizados para o preenchimento da estrutura do gestor de missões.

Registo das Acções

O registo de acções é essencialmente o registo de todas as funções para o controlo do veículo autónomo e concretização da missão numa lista, podendo conter acções generalistas (“recta”, “curva”, “túnel”...), ou mais específicas. Esse registo será utilizado, aquando da criação de uma missão, para pesquisa das acções existentes e cópia destas.

Cada acção conterá um diverso conjunto de apontadores para funções (para o controlo e realização da acção).



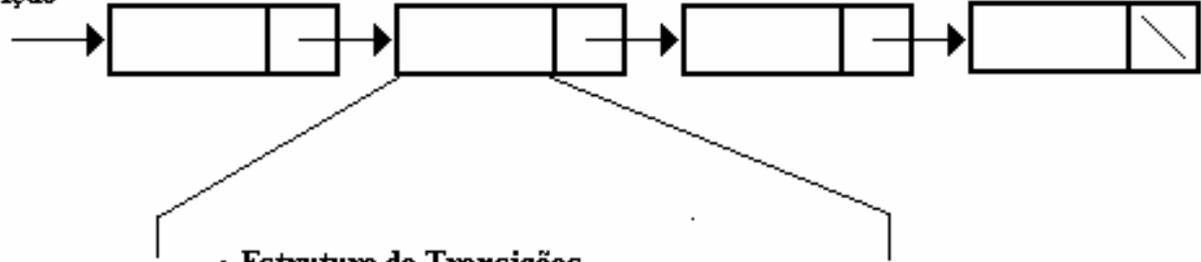
Registo das Transições

O registo das transições consiste no registo de todas as transições que possam ocorrer durante as manobras. Este registo de transições será também necessário, assim aquando da criação de uma missão, para pesquisa das transições pretendidas, para confirmação da sua existência e cópia das mesmas para uma manobra. Cada transição irá conter ainda um apontador para a função de teste da mesma, retornando se essa dada transição é verdadeira ou falsa. Conterá ainda uma variável, que dependendo do estado dela, e do resultado da transição poderá mandar executar uma função específica, que contém o seu apontador guardado na estrutura a registar.

Estados da variável “Executar”:

- 0 – Não Executará nada (Default)
- 1 – E valor retornado pela transição for verdadeiro executa aplicação
- 2 – E valor retornado pela transição for falso executa aplicação

RTransição



· **Estrutura de Transições**

- **Nome**
- **Apontador para Função Teste Transição**
- **Executar**
- **Apontador para Função Aplicar**

Gestor de Missões

O gestor de missões terá como principal objectivo a execução e controlo das missões a executar por cada veículo autónomo. Deste modo serão criadas diversas estruturas e listas, sendo algumas das entradas destas copiadas a partir do registo de acções e registo das transições, para o correcto teste e execução de todas as manobras do veículo autónomo, lidas segundo algumas normas a partir do ficheiro de missões.

Cada veículo autónomo para atingir o objectivo da missão poderá ter uma única sequência de manobras, como poderá ter várias sequências de manobras ou acções em paralelo, sem qualquer relação a funcionarem ao mesmo tempo (paralelismo).

Paralelismo

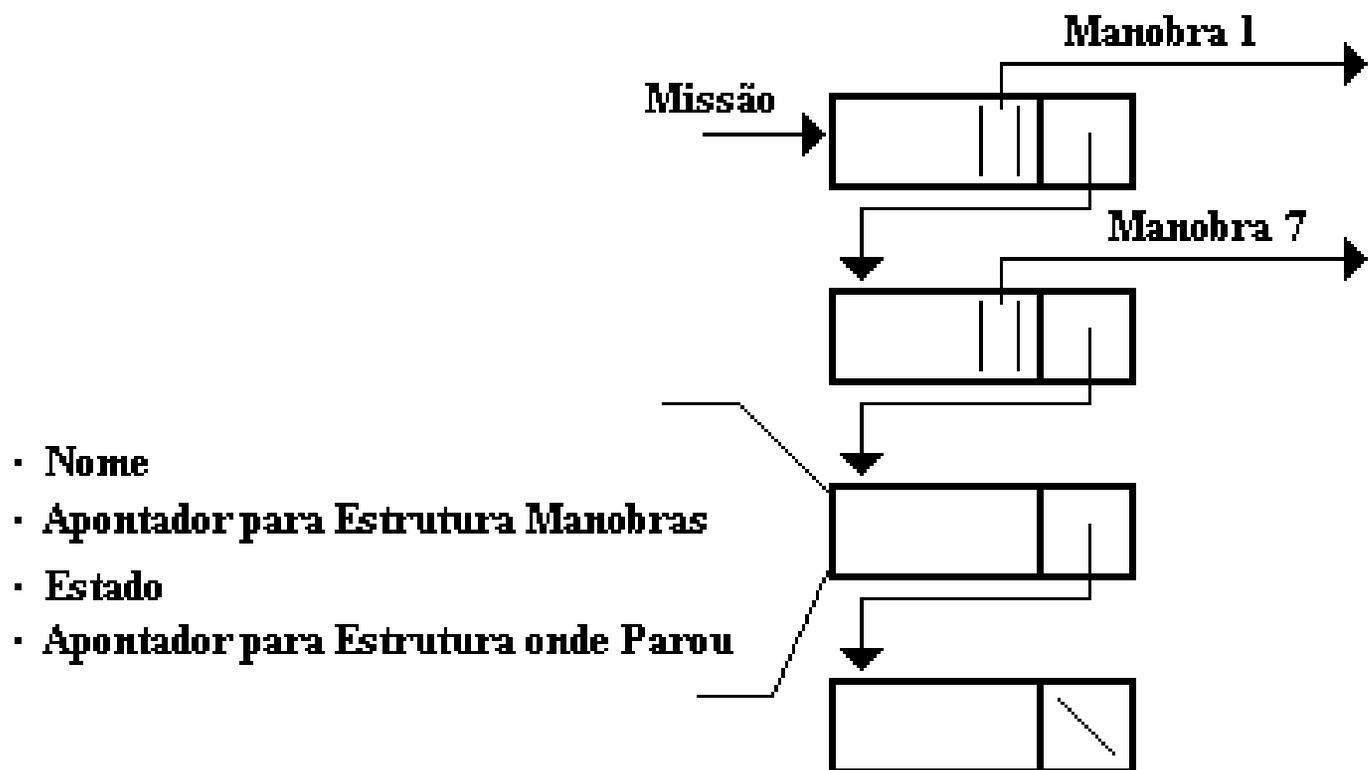
Ao existir vários conjuntos de sequências de manobras a correr em paralelo, podemos ter o controlo de diversos aspectos de um veículo autónomo ao mesmo tempo, podendo existir dois tipos de paralelismo, "paralelismo de alto nível" e "paralelismo de baixo nível".

No caso de paralelismo de alto nível poderemos ter como exemplo um robô industrial, que ao mesmo tempo que tem que percorrer um percurso sem parar, tem que apanhar com um braço robótico todo o material que encontrar nas plataformas ao seu lado direito e colocar nas plataformas do seu lado esquerdo. Deste modo temos um conjunto sequencial de manobras que fará o controlo da trajectória do mesmo, e um outro conjunto sequencial de manobras que sempre que encontrar ao seu alcance algum material irá transporta-lo, assim sendo temos dois conjuntos sequenciais de manobras sem relação aparente e a funcionarem ambas ao mesmo tempo em paralelo, para atingir um único objectivo, a missão.

No caso de paralelismo de baixo nível poderemos ter como exemplo um robô que aspira, em que este durante a sua locomoção, se detectar lixo terá que ligar o sistema de aspiração, assim sendo, depois de detectado o lixo este autómato continuará a sua locomoção (trajecto normal) e em paralelo terá o sistema de apanhar lixo ligado, tendo assim uma manobra que conterà duas acções em paralelo.

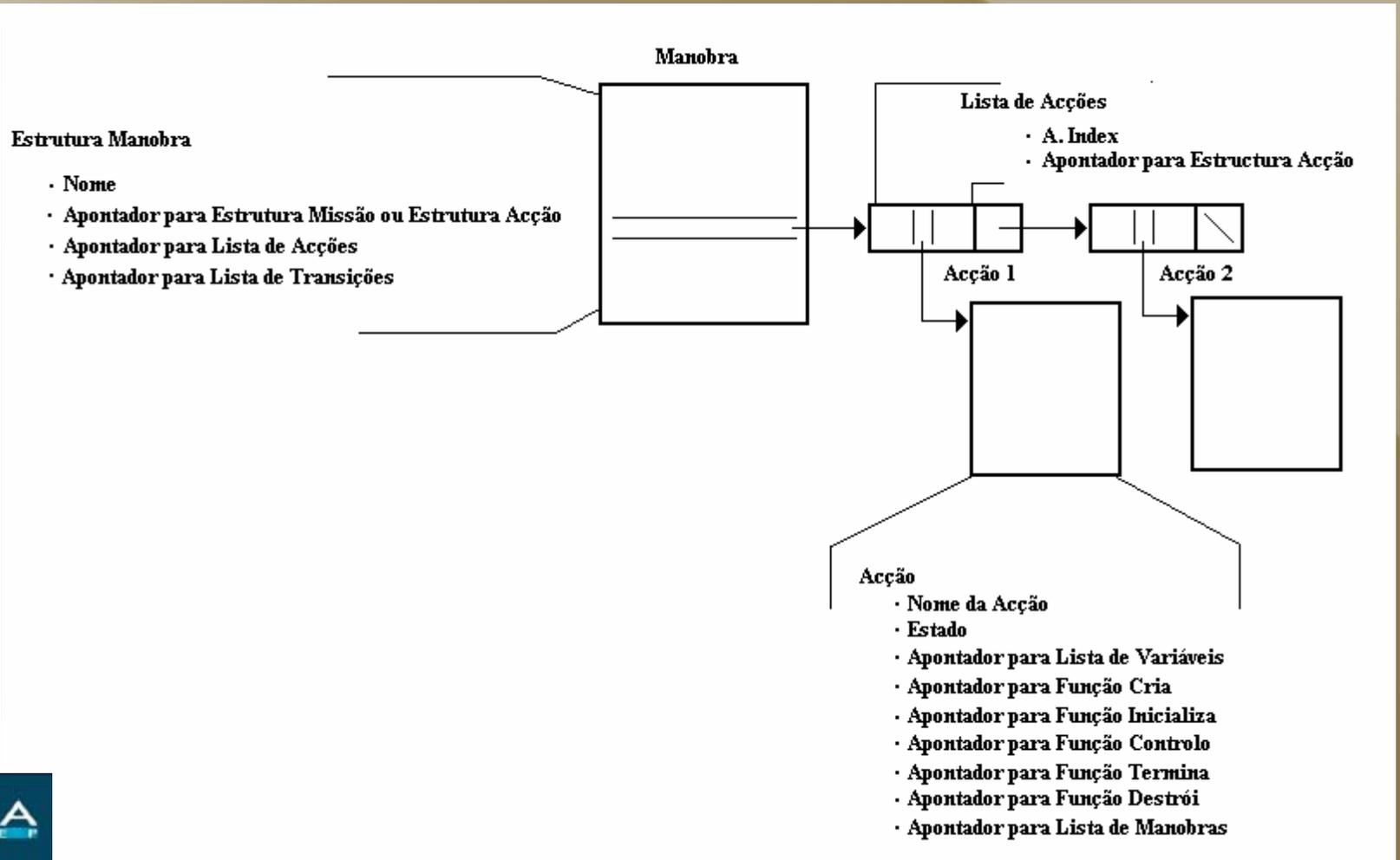
Paralelismo Alto Nível

Para a realização do paralelismo de alto nível, foi criada uma lista ligada a qual demos o nome de missão, sendo essa a nossa primeira lista do gestor de missões. Essa lista conterà um registo de todas as sequências de manobras que poderão estar a funcionar em paralelo.



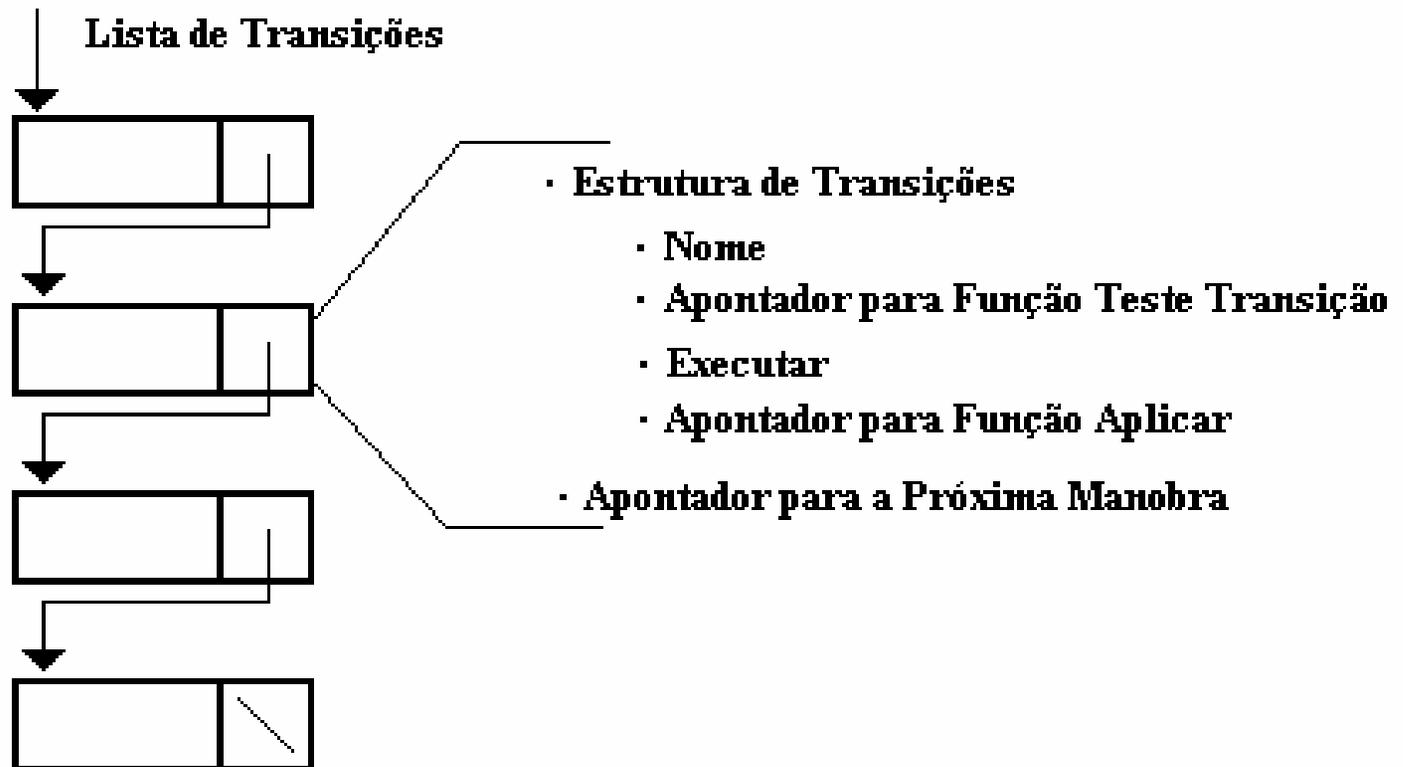
Paralelismo Baixo Nível

Para a realização do paralelismo de baixo nível o registo de cada manobra vai conter um diverso conjunto de acções, sendo essas guardadas numa lista na estrutura de manobra (lista de acções).



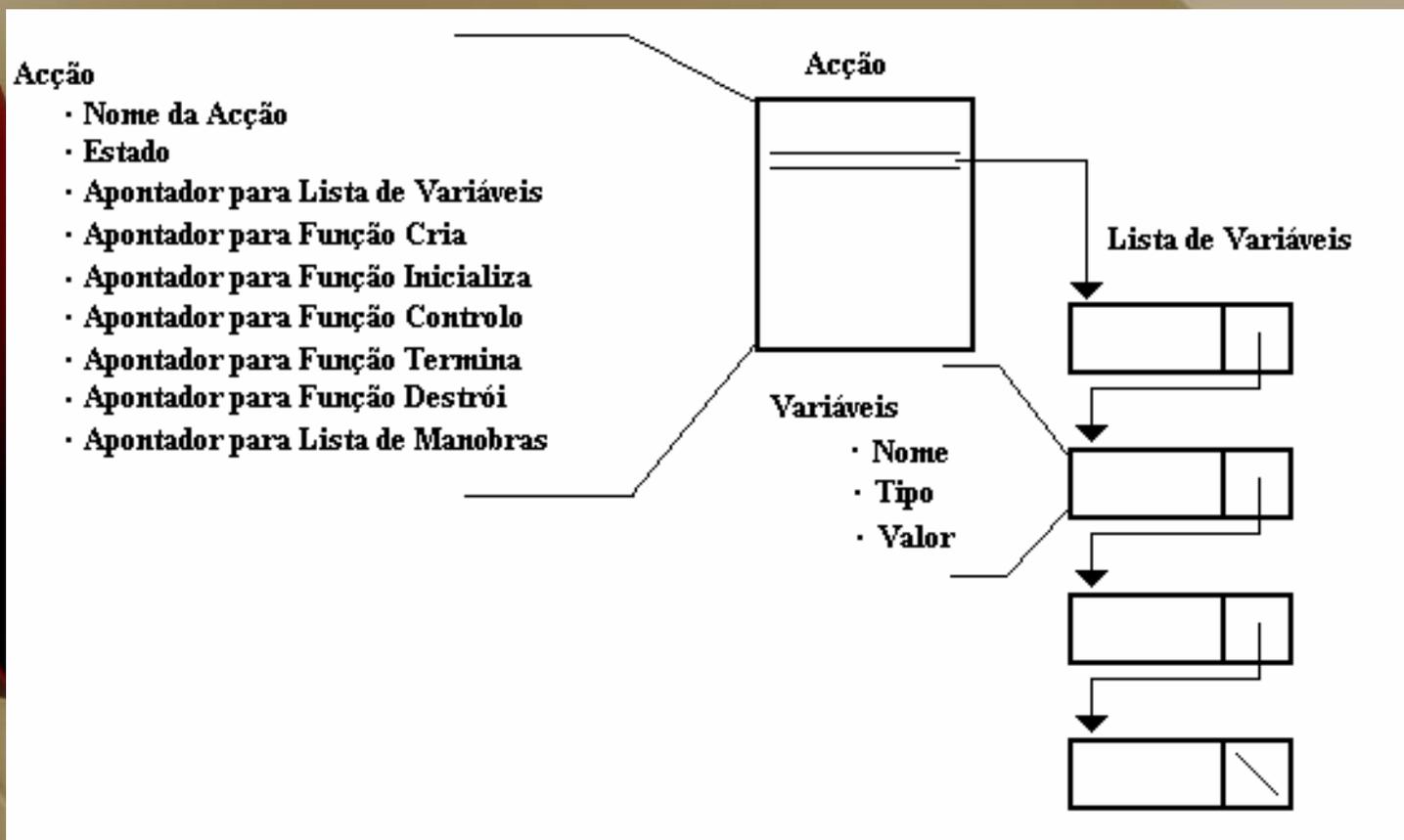
Lista de Transições

Na lista de transições, cada entrada contém uma cópia de uma transição registada no registo de transições, e poderá conter um apontador para uma manobra.



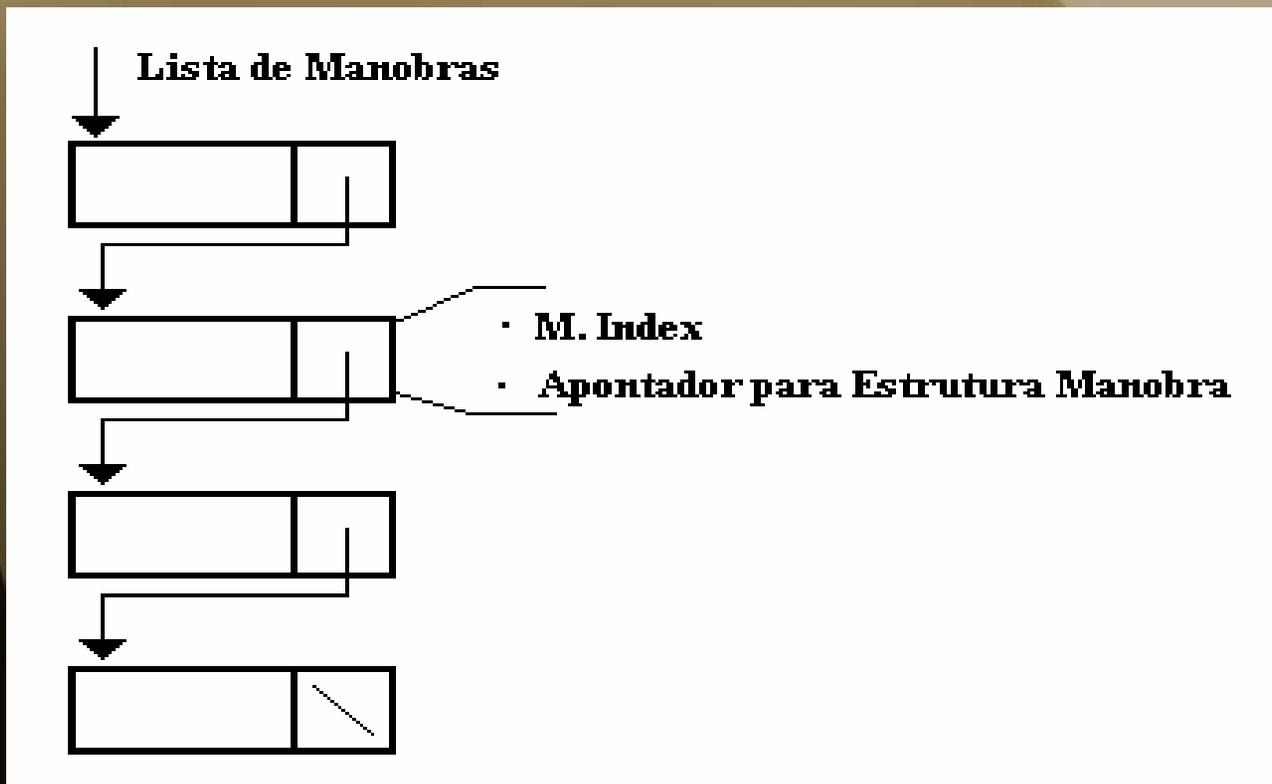
Estrutura Acção

Como visto anteriormente cada manobra pode ter várias acções a correrem em paralelo. Cada acção conterá apontadores para as funções necessárias para o comprimento da manobra, sendo esta uma copia de uma estrutura acção do registo de acções. Esta estrutura também é importante para a implementação de sub-manobras.



Lista de Manobras

Na lista de manobras podemos encontrar apontadores para as diversas estruturas de manobras que poderão ser executadas pela acção de uma manobra. Deste modo podemos criar sub-manobras dentro de cada manobra, sendo que essa sub-manobra é por si só uma manobra.



Ficheiro de Missões

Os ficheiros de missões conterão descritos todos os conjuntos de manobras, acções e transições para que uma missão possa ser carregada por software com as estruturas anteriormente descritas e executada. Um ficheiro de missões pode ainda ter uma chamada para outro ficheiro de missões, sendo que o ultimo contém mais manobras. Todos estes ficheiros de missões serão estruturados e escritos segundo algumas normas.

Tipos de Variáveis

Tipo	Num de bits	Intervalo	
		Início	Fim
Char	8	-128	127
unsigned char	8	0	255
signed char	8	-128	127
Int	16	-32.768	32.767
unsigned int	16	0	65.535
signed int	16	-32.768	32.767
short int	16	-32.768	32.767
unsigned short int	16	0	65.535
signed short int	16	-32.768	32.767
Long int	32	-2.147.483.648	2.147.483.647
signed long int	32	-2.147.483.648	2.147.483.647
unsigned long int	32	0	4.294.967.295
Float	32	3,4E-38	3.4E+38
Double	64	1,7E-308	1,7E+308
Long double	80	3,4E-4932	3,4E+4932

Deste modo no ficheiro para a descrição de variáveis utilizaremos a seguinte estrutura, em que no campo N indica qual a variável, TYPE conterà o tipo de variável (um dos anteriores), e no campo VAL será escrito qual o conteúdo que a variável deverá conter:

```
<VAR>
```

```
{N="x" TYPE="int" VAL="-3"}
```

```
{N="y" TYPE="unsigned int" VAL="2"}
```

```
</VAR>
```

Funções Manobra

Todas as manobras conterão acções e transições, para criar uma nova manobra, começaremos por lhe dar um nome (MMAN). Depois de termos indicado toas as acções e transições de uma manobra, concluiremos essa manobra com o comando </MMAN>.

```
<MMAN="nome_m">
```

```
</MMAN>
```

Funções Acção

As acções já estão registadas no registo de acções, deste modo aqui só necessitaremos de indicar o nome da acção (MAC), para posteriormente a procurarmos nesse registo. No caso de uma acção simplesmente implementar sub-manobras, poderá ser necessário indicar qual a sub-manobra que será iniciada primeiro, deste modo existirá um campo (INIT) no qual indicaremos o número da sub-manobra (a contagem das sub-manobras começará em 0).

No caso de não ser indicada nenhuma sub-manobra utilizar-se-á a sub-manobra zero (0).

```
<MAC="nome_acção" INIT="0">  
</MAC>
```

Funções Transição

Do mesmo modo que as acções se encontram registadas, as funções de transição também estão registadas no registo de transições, deste modo aqui só necessitaremos de indicar o nome da transição (TTRANS), para posteriormente a procurarmos nesse registo. Será ainda necessário indicar qual manobra seguinte a executar (JPM), no caso de a transição ser verdadeira. Posteriormente podemos indicar de quanto em quanto tempo queremos que a transição seja verificada (TIME), assim deste modo se as restantes funções de controlo da manobra demorarem muito tempo, a transição será executada num tempo mais curto que este.

```
<TTRANS="nome_t" JPM="manobra" TIME="em segundos">
```

Funções Ler Ficheiros

Como o dito inicialmente, um ficheiro missões pode ter uma chamada para outro ficheiro de missões, em que este pode simplesmente conter uma acção, uma manobra, ou um conjunto de transições. Deste modo terá que existir alguma atenção onde chamaremos esse ficheiro.

```
<LFILE="nome_ficheiro">
```

Normalmente os ficheiros conterão manobras, assim sendo antes da instrução de ler o ficheiro teremos que iniciar uma nova acção. Se o ficheiro contivesse uma acção teria que ser colocada a sua chamada depois da abertura de uma nova manobra, se por outro lado o ficheiro contivesse um conjunto de transições teríamos que ter o cuidado de o ler antes de fechar uma manobra.

Ficheiro .log

Durante a leitura de uma missão irá ser criado um ficheiro “.log” que vai conter os erros, não só de leitura do ficheiro, como também os erros a quando da criação de todas as estruturas.

Todos os erros a serem escritos serão iniciados por caracteres predefinidos, sendo que tudo que estiver escrito depois até ser encontra uma nova linha será referente a esse erro. Teremos então os seguintes caracteres de descrição de erros:

Teremos então os seguintes caracteres de descrição de erros:

Erro	Descrição
<LF>	Caracteres inválidos no ficheiro de missões (leitura)
<CM>	Criação na memória (espaço...)
<R>	Falta de registos das funções (cópia dos registos das funções)
<AV>	Impossível alterar variáveis (Manobras criadas)
<TJPM>	Manobra inválida para transitar (transições)

Referências:

- <http://www.cis.upenn.edu/group/mars/site/multimédia.htm#pictures>
- <http://www.swarms.org/>
- <http://www.cis.upenn.edu/mars>
- <http://www.grasp.upenn.edu/>