

Projeto 2- PSI (Documento 2)

Este documento apresenta os módulos completos `button_handler`, `fsm_init`, `fsm_step` e `fsm_food`. Os arquivos VHDL correspondentes são fornecidos na pasta da disciplina em NEWSERVERLAB. Os alunos devem entender o funcionamento dos módulos por meio de estudo e simulação.

Para os alunos que trabalharem em dupla para o projeto, é obrigatória a apresentação dos testbenches destes módulos, indicando o correto funcionamento.

1. Parte C

Informações sobre a funcionalidade dos módulos descritos a seguir estão no documento descritivo do Snake, fornecido na aula 1.

1.1. Fsm_Food

O módulo de FSM denominado *FSM_Food* é aquele responsável pelo posicionamento do alimento no tabuleiro do jogo, como apresentado no documento sobre a funcionalidade do *Snake*. O seu diagrama de ASM é reapresentado na Figura 1. O aluno poderá reparar que, em relação ao modelo do primeiro documento descritivo do Snake, o estado adicional de *Food_OK* foi adicionado para que a máquina seja modelada como do tipo Moore.

O sinal de entrada *fsm_m_start*, originado da *FSM_Main* (já estudada anteriormente) é checado no estado *Ready*, indicando se a *FSM_Food* é colocada em operação. Na saída existe o *flag fsm_m_done* a ser enviado para *FSM_Main* indicando se a tarefa de posicionar o alimento no tabuleiro do *Snake* foi realizada ou não.

A transição entre os demais estados é realizada dependendo de dois sinais internos, oriundos do datapath: os *flags ofc_of_x* e *comp_body_flag*. O primeiro verifica no estado *Add_X* se ocorreu overflow na soma ao se calcular o endereço (da memória) da nova coluna x (para a atual linha y). Em caso positivo passa-se para o estado *Add_Y* para realizar o incremento da coordenada y (avança para a próxima linha). Por sua vez, *comp_body_flag* é verificado em *Add_X* e *Add_Y*, identificando se a posição do tabuleiro está livre. Se estiver o alimento é colocado no tabuleiro no estado *Food_OK*. Estes *flags* serão ainda abordados posteriormente.

Testbench:

Providencie um testbench nos moldes propostos na disciplina para uma verificação segura do módulo.

Se realizado em dupla, escolha uma das implementações realizadas no Projeto-1.

FSM_FOOD

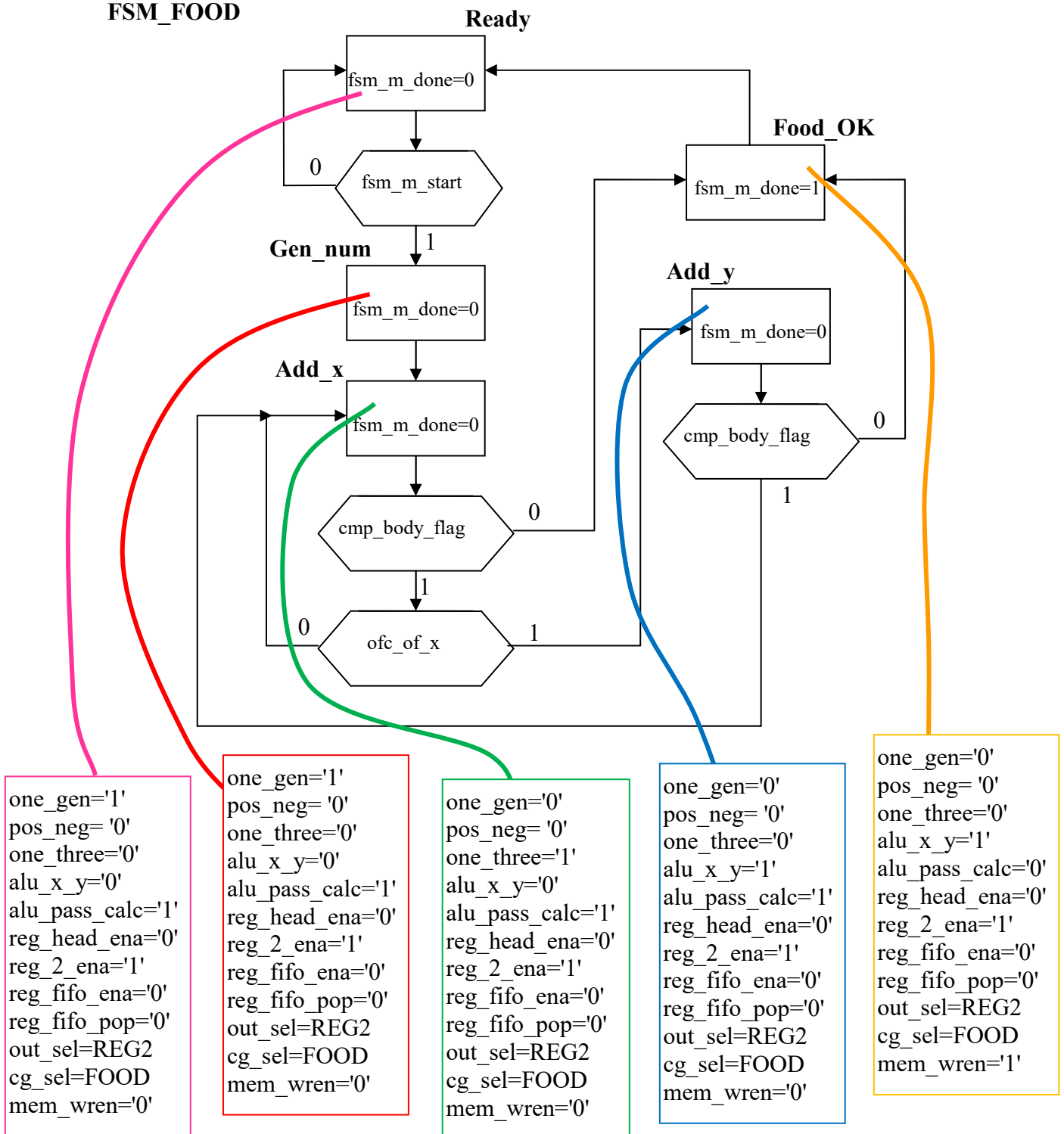


Figura 1. Diagrama da ASM (Fsm_Food)

1.2. Fsm_Step

O módulo Fsm_Step corresponde à máquina de estado de andamento e decisão do jogo. Pode-se observar pela Figura 2, que, diferente das outras formulações, ela corresponde a uma máquina de Mealy. Seguindo a ASM, vemos que ela realiza as seguintes ações em sequência:

- pelo estado New_Position, a próxima casa do tabuleiro (a posição correspondente de memória) é definida através da direção da cobra (ver seção do button handler da Semana 1).
- pelo estado Check, deve-se realizar a escrita da cabeça da cobra na nova posição, mas antes, verifica-se se nesta nova posição possuía: (a) uma parte da cobra; (b) um alimento; (c) um vazio. Caso (a), acaba o jogo; caso (b), segue para o próximo passo; caso (c) vai para o estado Pop_Write_Tail;
- o estado Pop_Write_Tail, a posição atual da cauda da cobra é resgatada (e eliminada) da fifo, e nesta posição de memória escreve-se um BLANK. Segue-se para o próximo passo.

Testbench:

Providencie um testbench nos moldes propostos na disciplina para uma verificação segura do módulo.

Se realizado em dupla, escolha uma das implementações realizadas no Projeto-1.

1.3. Pequenos Blocos do Datapath

De todos os blocos do datapath que são apresentados no documento de especificação do Snake, apenas três deles não foram tratados durante o semestre. São eles: code_gen, overflow_correction e comparator, cujas funcionalidades são apresentadas respectivamente nas seções 4.3.2, 4.3.6 e 4.3.7 do documento mencionado.

Os blocos são pequenos e simples e não vale a pena construir testbenches para eles. Os alunos devem realizar para cada bloco:

- ler as seções 4.3.2, 4.3.6 e 4.3.7 do documento de especificação do Snake e entender os objetivos de cada um dos blocos
- entender os blocos VHDL de cada um dos blocos
- fazer simulações à mão, mostrando a relação entrada-saída dos três blocos VHDL, de acordo com as entradas e saídas, com os respectivos tipos.

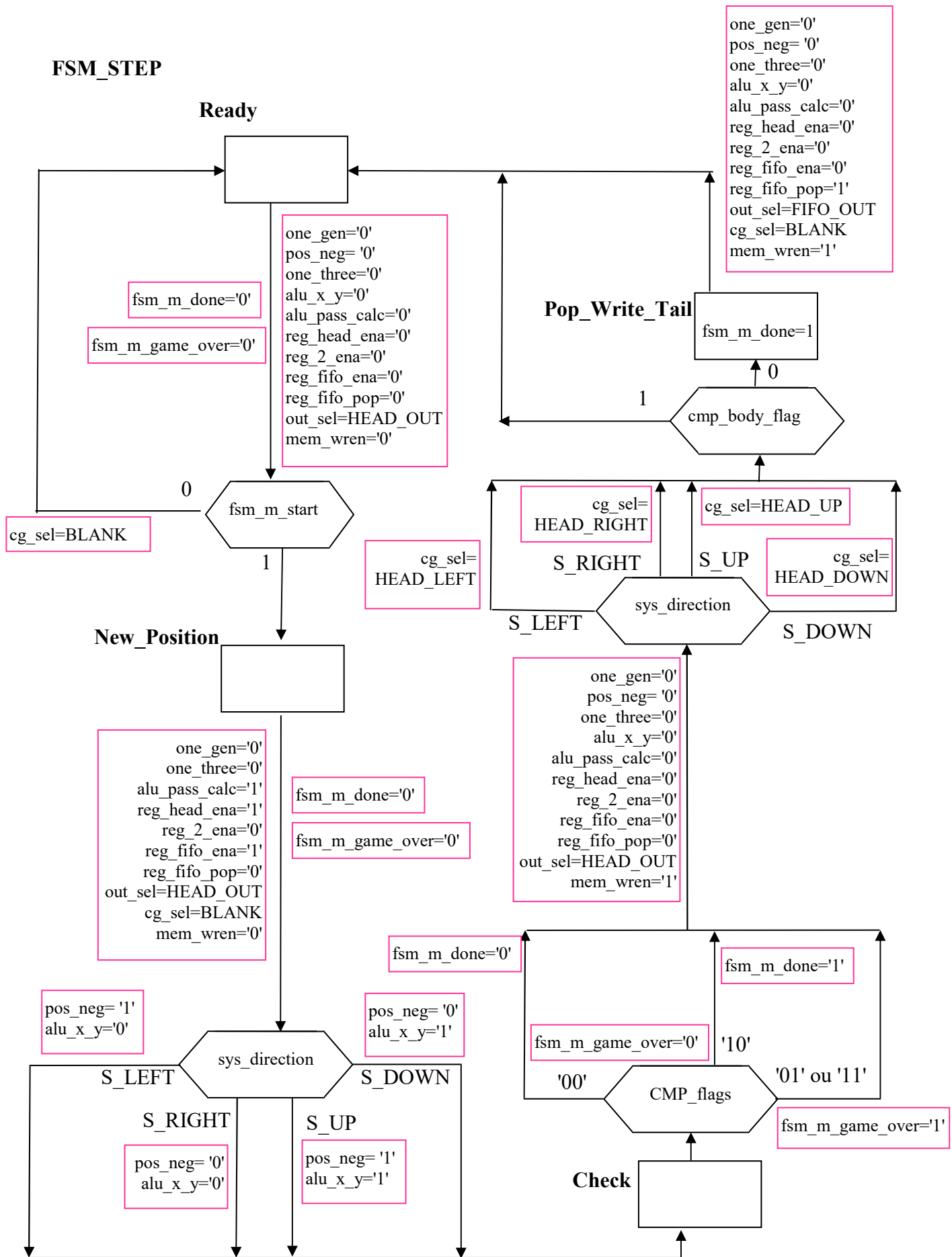


Figura 2. Diagrama da ASM (Fsm_Step)