

Projeto 2- PSI (Documento 3)

Este documento apresenta os esquemáticos de integração de módulos do Snake: do módulo topo completo, do controlador e do datapath. Os esquemas são apresentados com os seus submódulos com todos os seus portos. Todos os módulos individuais básicos já foram trabalhados de alguma forma pelo(a) aluno(a) (projetados, modificados ou simulados), portanto, cabe a ele(a) conectá-los corretamente e verificar o seu pleno funcionamento. Os esquemas são distribuídos e discutidos nas seções seguintes, sendo eventuais observações apresentadas.

Ao aluno(a), é recomendado que use o documento do projeto geral Snake para acompanhar o estudo dos esquemáticos.

1. Parte D

O módulo Snake_hw corresponde ao módulo-topo que integra todos os blocos do Snake. Como apresentada na Figura 1, a arquitetura da entity VHDL correspondente instancia 4 blocos. Os esquemáticos dos blocos de controle e datapath são detalhados nas figuras seguintes.

O Control_Snake apresentado na Figura 2 é o controlador composto das máquinas FSM_Main, FSM_Init, FSM_Food e FSM_Step. Observe que há um elemento de multiplexação dos sinais para a saída *dp_ctrl* que pode ser descrito como um modelo dataflow.

O Datapath apresentado na Figura 3 é o módulo contendo os tratamentos de dados e endereços da memória do Snake, sendo estes últimos registrados em registradores específicos ou na FIFO (Reg_bank). Observe que em relação ao módulo de datapath fornecido na aula 6 (fsmd_2), além da inclusão do módulo reg_bank, o sinal de seleção do mux de saída apresenta

2. Testbench

Para comprovar o funcionamento da entity topo Snake_hw, um testbench é fornecido para os alunos. No **Windows Explorer**, na pasta **Rede=> NEWSERVERLAB => psi3451 => aula_12_a 14**, uma pasta Teste poderá ser encontrada contendo o arquivo topo do testbench junto com os seus submódulos (em arquivos correspondentes). A estrutura do testbench topo, *snake_test_top* é composto de 3 submódulos: *snake_hw*, *snake_testbench* e *map_monitor*. O primeiro deve ser instanciado pelo aluno com o seu projeto; o segundo emula a ação do jogador pressionando os botões; por fim, o último é uma forma de verificar como os valores da memória devem ser varridas e passadas ao vídeo (não é relevante para verificar o funcionamento do jogo).

Em relação ao testbench, o(a) aluno(a) deverá observar com atenção as ações definidas em *snake_testbench*, pois estas definirão a sequência de passos do jogo e toda a interação que o "jogador" deverá ter com o jogo. É oportuno lembrar que caso o "jogador" não pressione nenhum botão, o jogo deveria prosseguir independentemente disto, avançando no sentido definido pelo último avanço da cobra. (continua)

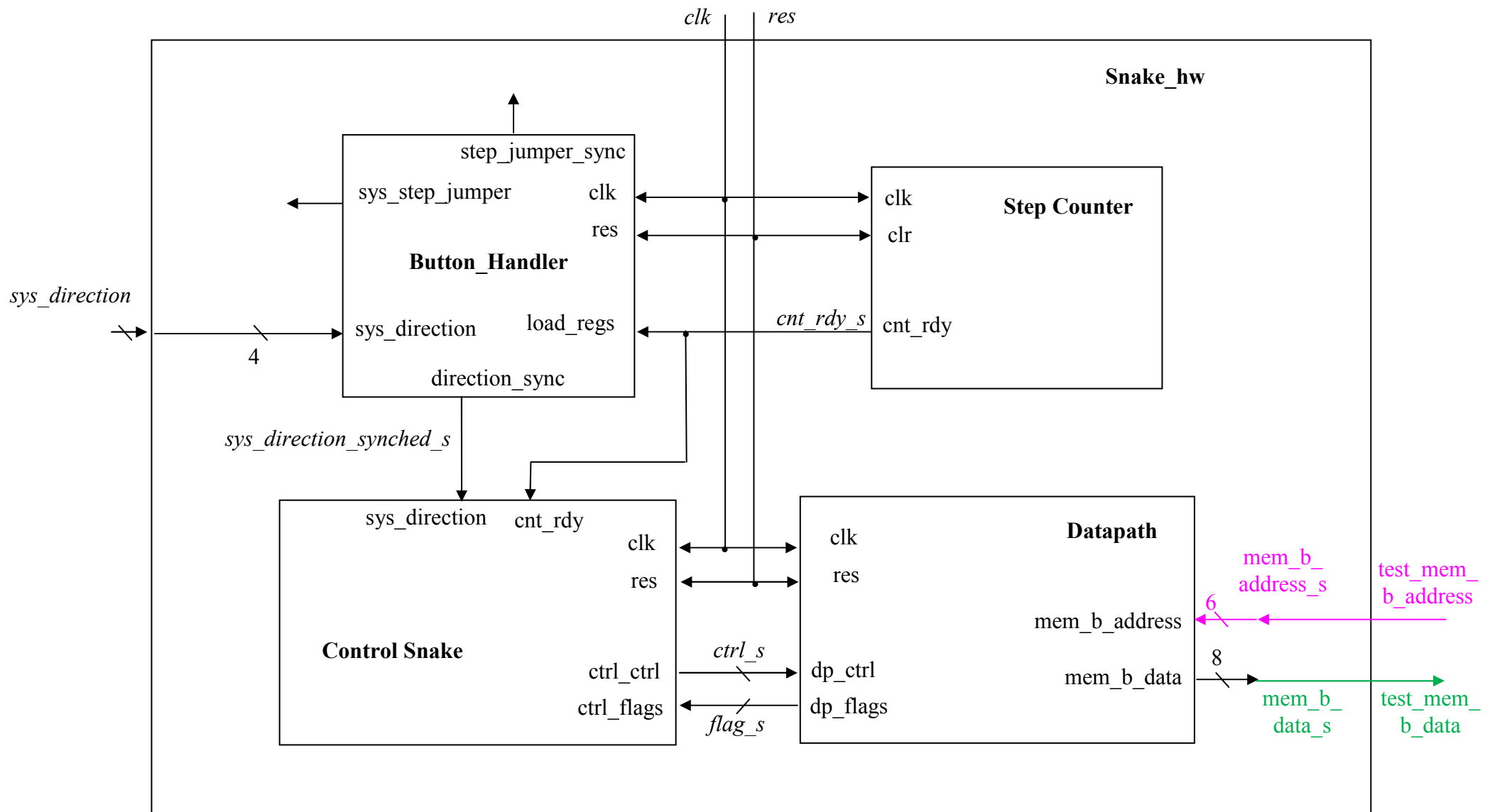


Figura 1. Esquema do módulo-topo

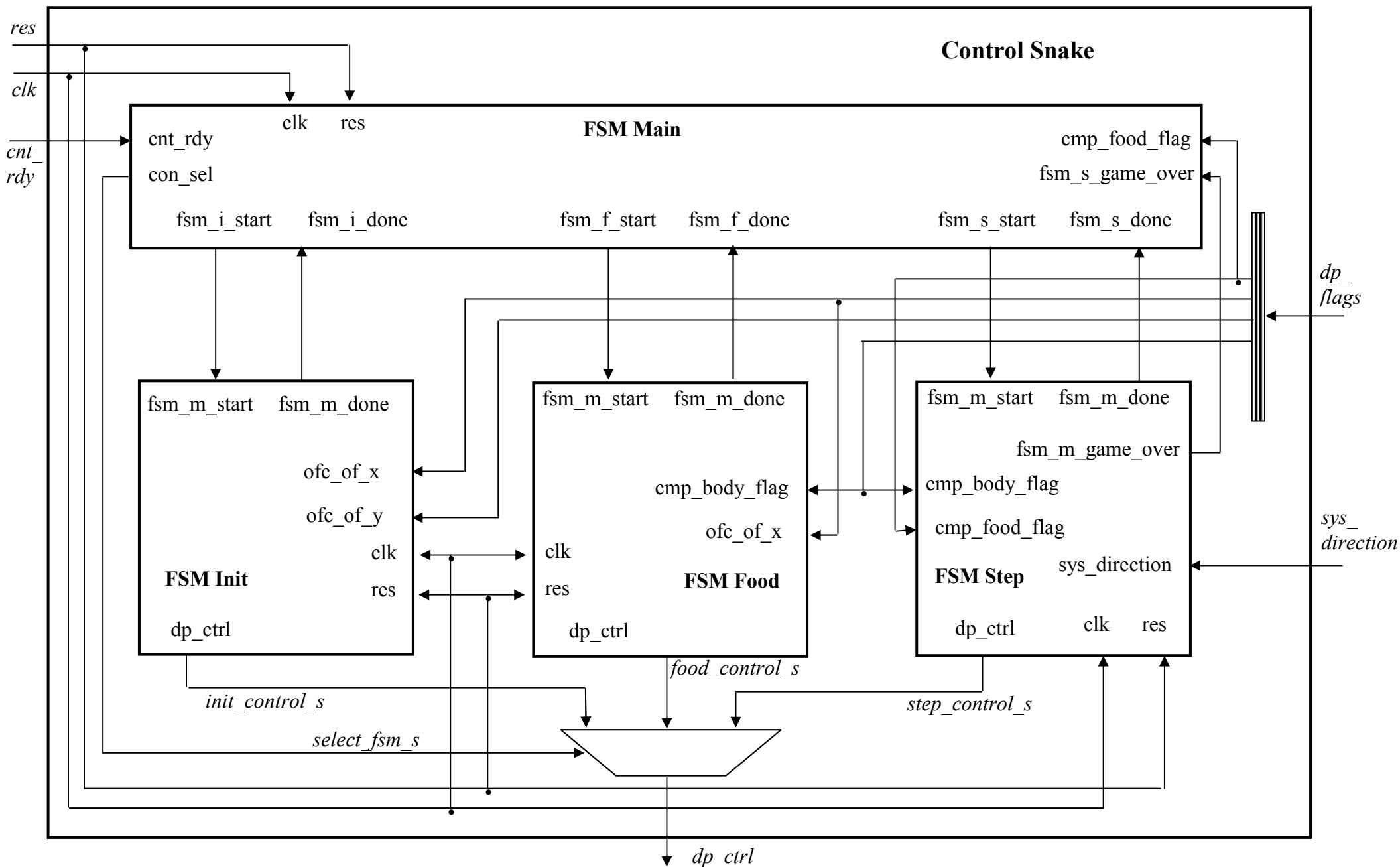


Figura 2. Controlador do Snake

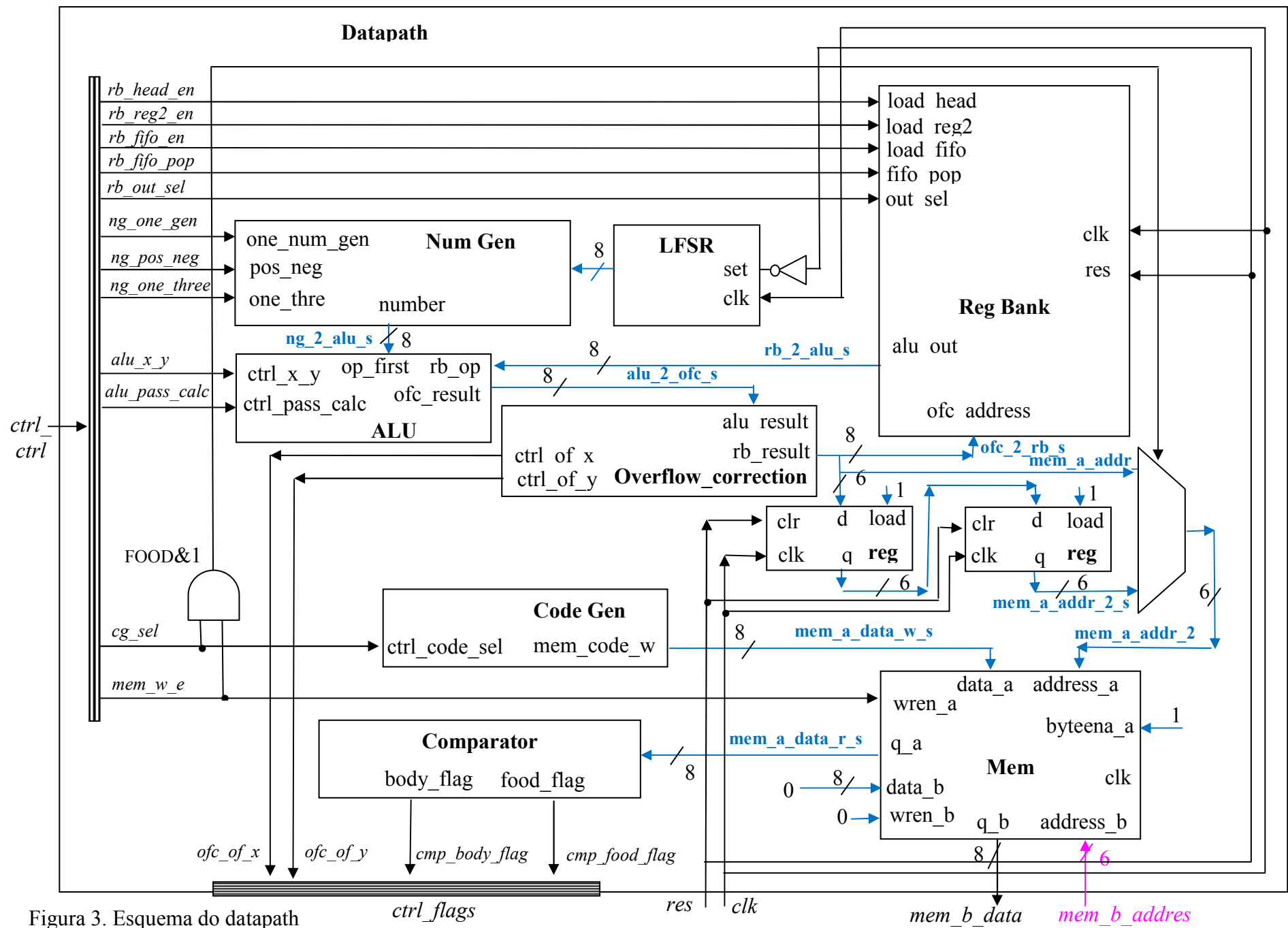


Figura 3. Esquema do datapath

O testbench pode ser utilizado nas duas formas já vistas durante o decorrer da disciplina:

- 1) Diretamente no Modelsim, na forma vista na aula 8 (a entity topo é o *snake_test_top*);
- 2) Pelo Quartus, na forma vista na aula 9 (a entity do projeto do Quartus é o *snake_hw*);

Para visualização no Wave, o(a) aluno(a) dever utilizar o script dado, de nome run_sim.txt. Este script pode ser usado dentro das duas formas acima:

- 1) Diretamente no Modelsim: siga as instruções dadas nas aulas 2 e 3 através do comando "do run_sim.txt"
- 2) Pelo Quartus: ao se ajustar os arquivos de testbench, selecione também a opção de se usar script para a simulação e selecione o arquivo run_sim.txt.

ATENÇÃO: Fica por conta do(a) aluno(a) ajustar a simulação de forma a mostrar que o jogo possa evolui corretamente em diversas condições.