

Controlador da memória DDR SDRAM da plataforma ML401

Este controlador visa abstrair as complicações de acesso à memória do tipo DDR, tratando esta de forma semelhante a uma memória RAM comum. A plataforma ML401 possui dois chips de memória DDR SDRAM HYB25D256160BT(L)-7 ×16 ligados em paralelo de maneira a formar uma memória com palavras de 32 bits. A Figura 1 ilustra a interface do controlador e sua ligação com a memória, seguida de uma descrição.

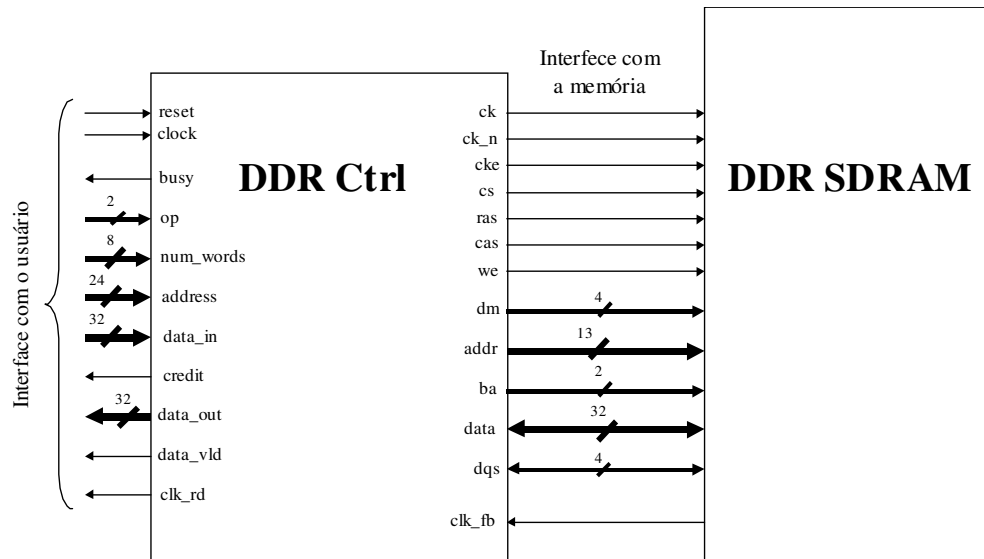


Figura 1 – Interface do controlador.

Descrição da interface com o usuário.

- *reset*: reset do controlador ativo em nível alto.
- *clock*: clock do controlador.
- *busy*: em nível alto indica que o controlador está ocupado executando a inicialização da memória, *refresh* ou leitura/escrita. Em nível baixo indica que o controlador está apto a realizar uma operação de leitura/escrita.
- *op*: operação a ser executada. 1: READ, 2: WRITE, 0: IDLE.
- *num_words*: número de palavras a serem lidas/escritas em uma operação (1 – 255).
- *address*: endereço inicial de leitura/escrita.
- *data_in*: palavra a ser escrita na memória.
- *credit*: em nível alto indica que o controlador está apto a receber palavras na entrada *data_in*.
- *data_out*: palavra lida da memória.
- *data_vld*: em nível alto indica que a palavra em *data_out* é válida.
- *clk_rd*: clock para a leitura das palavras na saída *data_out*.

Características do controlador:

- Abstrai a organização de matriz de palavras comum das memórias DDR
- Escrita: 1 palavra de 32 bits por ciclo
- Leitura: 2 palavras de 32 bits por ciclo
- Burst: até 255 palavra de 32 bits (leitura/escrita)
- Refresh a cada 7us

Escrevendo na memória

A Figura 2 ilustra uma escrita em *burst* de 4 palavras na memória a partir do endereço 0x100. As entradas em negrito devem ser geradas pelo usuário. As palavras a serem escritas são: 1, 2, 3 e 4.

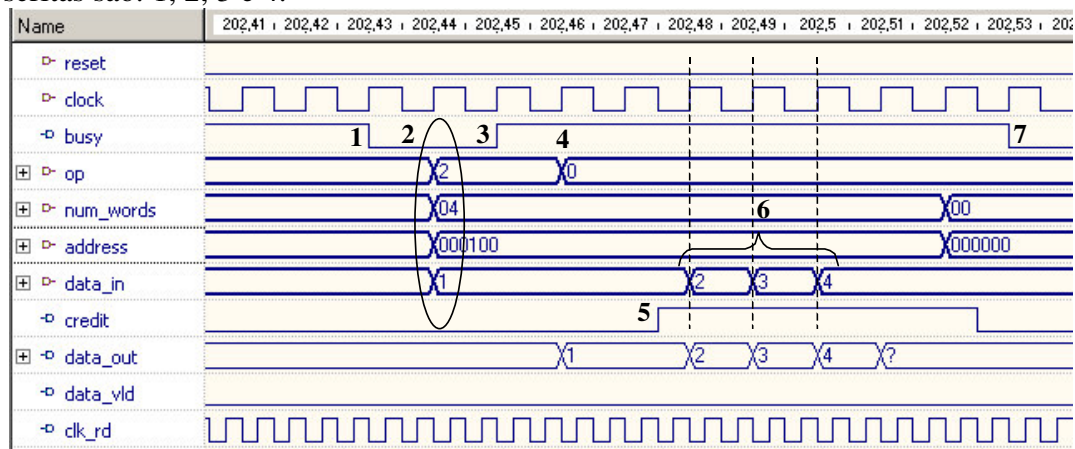


Figura 2 – Escrita na memória.

1. O controlador coloca a saída *busy* em nível baixo, indicando que está apto a realizar uma leitura ou escrita na memória.
2. Uma escrita é solicitada ao controlador através da entrada *op* (*op* = 2). O número de palavras a serem escritas (*num_words*) é 4 a partir do endereço (*address*) 0x100 e a primeira palavra a ser escrita (*data_in*) é 1.
3. O controlador indica através da saída *busy* que a solicitação foi atendida.
4. A entrada *op* deve voltar para *idle* (*op* = 0), pois a solicitação de escrita já foi atendida.
5. O controlador indica através da saída *credit* que na próxima borda de subida da entrada *clock* a próxima palavra a ser escrita na memória deve ser colocada na entrada *data_in*.
6. A cada borda de subida da entrada *clock* uma nova palavra a ser escrita na memória é colocada na entrada *data_in*.

7. O controlador indica através da saída *busy* que a operação está concluída e novas solicitações podem ser feitas.

O resultado da escrita é o seguinte:

MEM[0x100] = 1

MEM[0x101] = 2

MEM[0x102] = 3

MEM[0x104] = 4

Lendo da memória

A Figura 3 ilustra a leitura em *burst* de 4 palavras da memória a partir do endereço 0x100. As entradas em negrito devem ser geradas pelo usuário. As palavras a serem lidas correspondem às escritas na Figura 2, ou seja: 1, 2, 3 e 4.

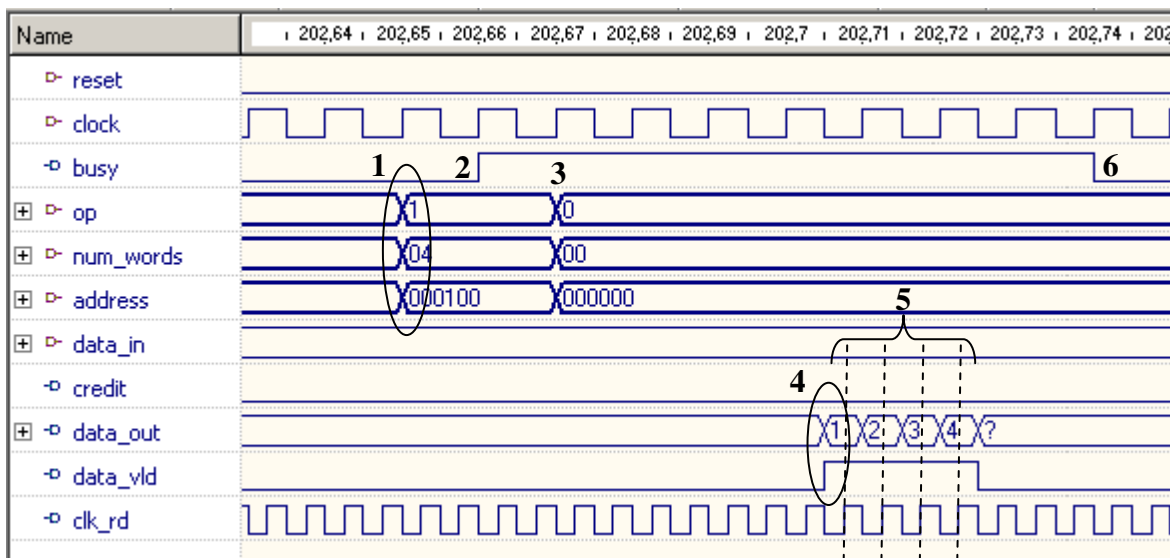


Figura 3 – Leitura da memória.

1. Uma leitura é solicitada ao controlador através da entrada *op* (*op* = 1). O número de palavras a serem lidas (*num_words*) é 4 a partir do endereço (*address*) 0x100 .
2. O controlador indica através da saída *busy* que a solicitação foi atendida.
3. A entrada *op* deve voltar para *idle* (*op* = 0), pois a solicitação de leitura já foi atendida.
4. A saída *data_vld* indica que a palavra na saída *data_out* é válida. A cada borda do *clock* (descida/subida) uma nova palavra estará válida na saída *data_out*.

5. As palavras em *data_out* devem ser lidas na borda de subida da saída *clk_rd*.
6. O controlador indica através da saída *busy* que a operação está concluída e novas solicitações podem ser feitas.