Assignment 7

Due November 12, 1996

Design a DRAM controller to interface a bank of 1 Mbit DRAMs to a simple CPU. Your controller need only deal with read and refresh cycles.

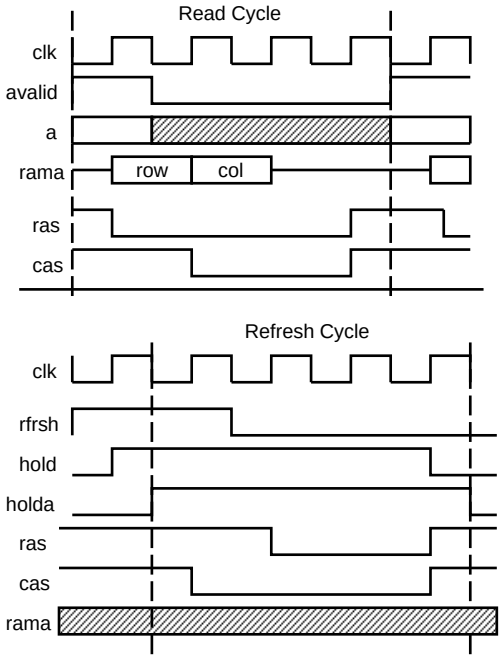
The controller's inputs are:

- CLK: A clock input. The clock has a 50% duty cycle and each bus cycle starts on the falling edge of the clock. There are 4 clock cycles per bus cycle. Each read or refresh cycle should take 4 clock cycles.
- A: A 20-bit address bus from the CPU. This address is valid only during the first clock cycle of a bus cycle so your circuit must latch it.
- AVALID: An 'address valid' signal from the CPU. This signal is only asserted during the first clock cycle of a bus cycle.
- HOLDA: A hold acknowledge signal from the CPU. This signal is asserted when the CPU has relinquished control of the bus and will not attempt to access the DRAM. This signal is asserted at the start of a bus cycle and stays active until the start of the bus cycle after HOLD is released.
- RFRSH: A refresh request input from a external timer circuit. This signal is synchronized so that it will go active at the start of the last clock cycle of a bus cycle if a memory refresh cycle is required. It will stay active for two clock periods.

The controller's outputs are:

- HOLD: A hold request to the CPU. It must be asserted at least half a clock period before the end of a bus cycle for the CPU to release the bus during the next clock cycle.
- RAMA: The multiplexed 10-bit address supplied to the DRAMs. The DRAMs have zero ns setup times and a hold times of less than one clock period for both row and column addresses.
- CAS and RAS: CAS* and RAS* strobes. Their behaviour must agree with the timing diagram shown below. Note that these signals are active-low and that the refresh cycle uses CAS*-before-RAS* refresh.

The timing diagram below shows the desired behaviour of the DRAM controller signals for RAM read and refresh cycles.



Write a synthesizeable VHDL description of a circuit that implements this DRAM controller. Test your design with the supplied test bench. Synthesize your code using the 'class' target library. Submit your VHDL code, the testbench output and the schematic electronically using the keys 7vhd, 7ps and 7log respectively.

Use the following VHDL entity:

```
entity dramc is
   port ( clk, rfrsh, avalid, holda : in bit ;
      a : in bit_vector (19 downto 0) ;
      hold, ras, cas : out bit ;
      rama : out bit_vector (9 downto 0) ) ;
end ;
```