Solutions to Assignment 4

Question 1

1. *chips per bank*: To create an 8-bit wide data bus using 4-bit chips we need $8/4 = 2$ chips per bank.

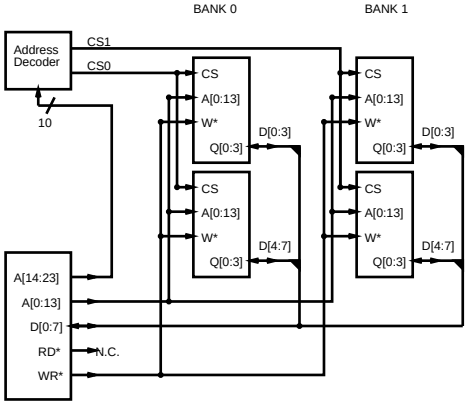
2. *number of banks*: Each bank provides 16 k words of 1 byte/word for a total of 16 k bytes per bank. To provide a total of 32k bytes we need $32/16 = 2$ banks.

3. *byte-select address lines*: To select from the 1 bytes available in parallel we need to use the $\log_2 1 = 0$ least-significant CPU address bits.

4. *word-select address lines:* To select one word in each bank we need $\log_2 16\text{k} = 14$ address bus bits (A0 to A13).

5. *bank-select address lines:* The remaining $(24 - 14 = 10)$ address lines (A14 to A23) are then used to select a bank.

The schematic of the memory system is as follows:



Question 2

One possible solution is:

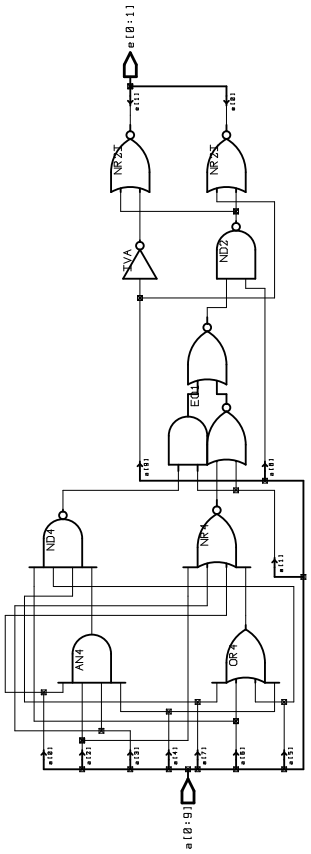
```
-- ELEC 379 Assignment 4, Question 2
-- memory decoder
-- Ed Casas, 98/2/22

library ieee ;
use ieee.std_logic_1164.all ;

entity decoder is
   port (
   a : in std_logic_vector (0 to 9) ;
   e : out std_logic_vector (0 to 1) ) ;
end decoder ;

architecture rtl of decoder is
begin
   e(0) <= '1' when a = "0000000010" else
           '1' when a = "1111111110" else
           '0' ;
   e(1) <= '1' when a = "0000000011" else
           '1' when a = "1111111111" else
           '0' ;
end rtl ;
```

The synthesized schematic for this design is:



Question 3

RAM Design

1. *chips per bank*: To create a 16-bit wide data bus using 8-bit chips we need $16/8 = 2$ chips per bank.

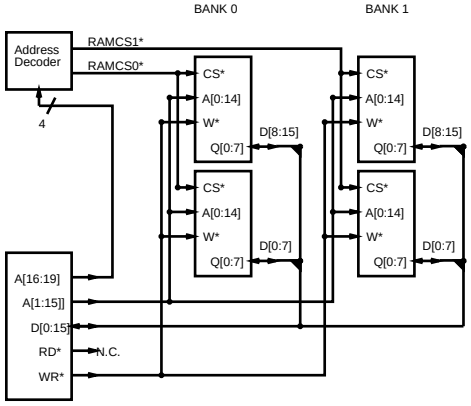
2. *number of banks*: Using a total of 4 chips and putting 2 chips/bank we can assemble $4/2 = 2$ banks. Since each bank provides 32 k words of 2 bytes/word the total is $32 \times 2 = 64$ k bytes per bank. This provides a total of 128k bytes of RAM.

3. *byte-select address lines*: To select from the 2 bytes available in parallel we need to use the $\log_2 2 = 1$ least-significant CPU address bit (A0).

4. *word-select address lines:* To select one word in each bank we need $\log_2 32\text{k} = 15$ address bus bits (A1 to A15).

5. *bank-select address lines:* The remaining $(20 - 16 = 4)$ address lines (A16 to A19) are then used to select a bank.

The schematic of the RAM design is:



EPROM Design

1. *chips per bank*: To create a 16-bit wide data bus using 4-bit chips we need $16/4 = 4$ chips per bank.

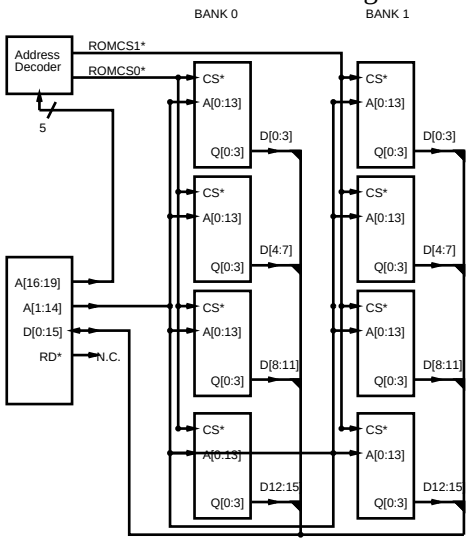
2. *number of banks*: Using a total of 8 chips and putting 4 chips/bank we can assemble $8/4 = 2$ banks. Since each bank provides 16 k words of 2 bytes/word the total is $16 \times 2 = 32$ k bytes per bank. This provides a total of 64k bytes of RAM.

3. *byte-select address lines*: To select from the 2 bytes available in parallel we need to use the $\log_2 2 = 1$ least-significant CPU address bit (A0, not available outside the CPU).

4. *word-select address lines:* To select one word in each bank we need $\log_2 16\text{k} = 14$ address bus bits (A1 to A14).

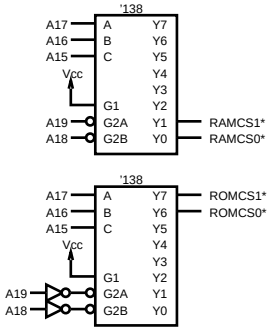
5. *bank-select address lines:* The remaining $(20 - 15 = 5)$ address lines (A16 to A19) are then used to select a bank.

The schematic of the EPROM design is:



Question 4

The first design uses one 3 to 8 decoder for each pair of enables. The enables are used to extend the number of address lines that can be decoded. Other solutions are possible.



The VHDL code for the decoders is similar to the one in the previous question:

```
-- ELEC 379 Assignment 4, Question 4
-- memory decoder
-- Ed Casas, 98/2/22

library ieee ;
use ieee.std_logic_1164.all ;

entity decoder is
   port (
   a : in std_logic_vector (4 downto 0) ;
   ramcs0_n, ramcs1_n : out std_logic ;
   romcs0_n, romcs1_n : out std_logic ) ;
end decoder ;

architecture rtl of decoder is
begin
   ramcs0_n <= '0' when a = "00000" else '1' ;
   ramcs1_n <= '0' when a = "00001" else '1' ;
   romcs0_n <= '0' when a = "11110" else '1' ;
   romcs1_n <= '0' when a = "11111" else '1' ;
end rtl ;
```

The synthesized schematic for this design is: