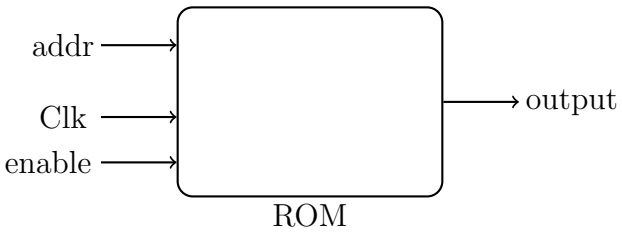
ROM (Instruction Memory)

Your task is to program the behavior of an entity called "ROM". This entity is declared in the attached file "ROM.vhdl" and has the following properties:

- Inputs: Clk and enable with type std_logic; these are clock and enable signals
- Inputs: addr with type std_logic_vector; the address of the instruction which is going to be read
- Outputs: output with type std_logic_vector; this is the instruction which is read



Do not change the file "ROM.vhdl".

The "ROM" entity shall behave according to the following:

It has three inputs which are address, clock and enable signals. The length of address is 14 bits, and the length of instruction is 16 which is equal to the length of the output.

You need to write the VHDL code of ROM and fill it with the sample list of instructions as below. Write the instructions (binary codes below) from address 00000000110010 to address 00000000111110 in the same sequence as they are written here.
Instructions:

1110100000000101
0101001101010111
0100100010111001
0100110011011110
0111001111110101
1000011110111110
0111101010110010
0100101100000010
0101100001100010
1111110010111110
1101110101101000
1101111011101110
0011110011111011

Note that:

- the content of other locations in the ROM is zero.
- it should be implemented as a single cycle ROM.
- it works on rising of the clock signal.
- when the ROM is disabled the output is zero. When it is enabled the instruction stored at the input address is output at the output.

This behavior has to be programmed in the attached file "ROM_beh.vhdl".

To turn in your solution, write an email to vhdl-dis+e384@tuwien.ac.at with Subject "Result Task 2" and attach your file "ROM_beh.vhdl".

Good Luck and May the Force be with you.