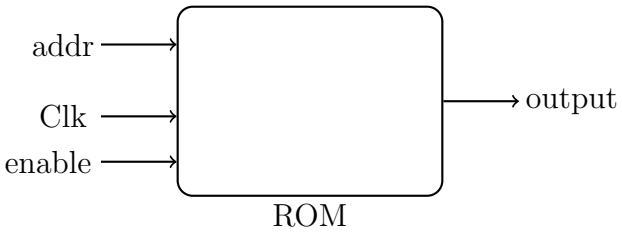
ROM (Instruction Memory)

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 16 bits, and the length of instruction is 14 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 0000000010011001 to address 0000000010100100 in the same sequence as they are written here.
Instructions:

01100000000010
10111010101101
10111010110000
11101111011111
01110000100111
01001111111110
10110001010111
10000000111011
10010111000101
01110101011101
11001101111011
00011110100110

Note that:

- the content of other locations in the ROM is zero.
- it should be implemented as a single cycle ROM.
- it works on falling edge 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 3" and attach your file "ROM_beh.vhdl".

Good Luck and May the Force be with you.