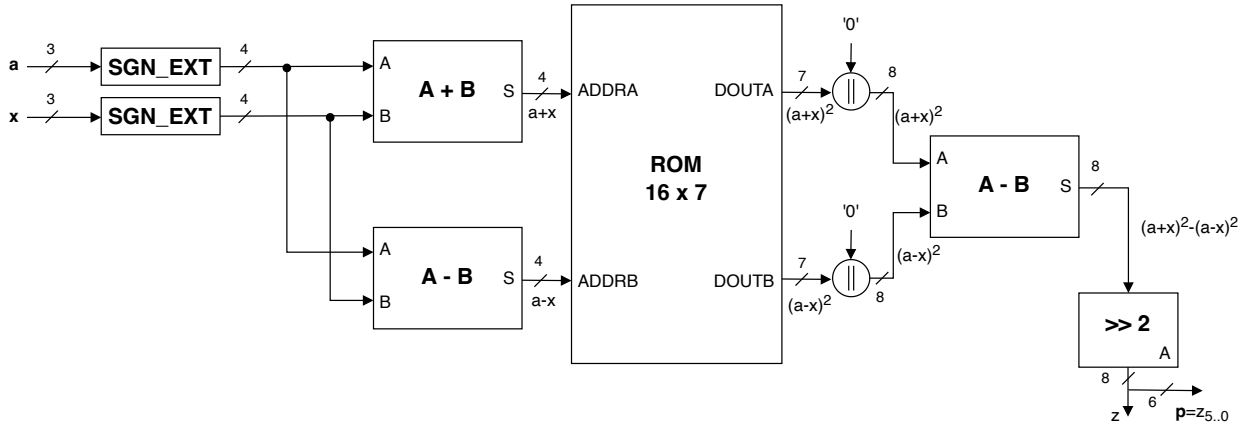
# Lab 2
# Implementing & Synthesizing Combinational Logic in VHDL

## Main Task: Signed Multiplication by Squaring

Multiplication by squaring takes advantage of the following dependence:

$$p = a \cdot x = ((a+x)^2 - (a-x)^2) / 4$$

Develop a Register-Transfer Level (RTL) VHDL description of the signed multiplication by squaring unit, MUL_by_SQR, based on the block diagram shown below.




In this block diagram, ROM is used to perform squaring of 4-bit signed integers. This functionality is accomplished by placing at each position i value $y=i^2$, where i is a value of a 4-bit signed integer represented by ADDRA and ADDRB. The content of ROM is described by the table given at the top of the next page. SGN_EXT is a sign extension from 3 to 4 bits. >>2 is an arithmetic shift right by 2 positions.
Develop a corresponding testbench that checks the correct operation of your circuit for all possible values of inputs **a** and **x**, treated as 3-bit signed integers.

**Deliverables:**

1. Synthesizable VHDL code of the MUL_by_SQR unit.
2. Corresponding testbench capable of verifying this code for correct operation using
    a. All combinations of inputs
    b. Expected output calculated in VHDL
    c. For incorrect results – inputs, expected output, and actual output displayed on the screen
3. Reports (saved as .txt) and schematic (saved as .pdf) obtained as a result of synthesis. Determine the number of slice LUTs and slice registers used by your circuit.
4. Post-synthesis model in VHDL generated using the Tcl command:
   write_vhdl -mode funcsim <output_folder>

| ADDR | ADDR treated as a signed number | DOUT decimal | DOUT binary |
|---|---|---|---|
| 0000=0 | 0 | 0 | 000 0000 |
| 0001=1 | 1 | 1 | 000 0001 |
| 0010=2 | 2 | 4 | 000 0100 |
| 0011=3 | 3 | 9 | 000 1001 |
| 0100=4 | 4 | 16 | 001 0000 |
| 0101=5 | 5 | 25 | 001 1001 |
| 0110=6 | 6 | 36 | 010 0100 |
| 0111=7 | 7 | 49 | 011 0001 |
| 1000=8 | -8 | 64 | 100 0000 |
| 1001=9 | -7 | 49 | 011 0001 |
| 1010=10 | -6 | 36 | 010 0100 |
| 1011=11 | -5 | 25 | 001 1001 |
| 1100=12 | -4 | 16 | 001 0000 |
| 1101=13 | -3 | 9 | 000 1001 |
| 1110=14 | -2 | 4 | 000 0100 |
| 1111=15 | -1 | 1 | 000 0001 |

5. Waveforms, in PDF, obtained by using post-synthesis functional simulation to verify results for at least 16 different values of inputs a and x. These waveforms should show at least the following signals in your circuit: a, x, p, and signals containing values of the following expressions a+x, a-x, $(a+x)^2$, $(a-x)^2$, $(a+x)^2-(a-x)^2$.
6. Waveforms, in PDF, illustrating the difference between
    a. Post-synthesis functional simulation
    b. Post-synthesis timing simulation.

## Bonus Task

What changes are necessary in your circuit, so it performs multiplication on two 3-bit unsigned integers (represented using std_logic_vectors) rather than on two 3-bit signed integers?
Modify block diagram first, and then repeat all stages of the implementation.

### Deliverables

The same as for the main task.

## Important Dates

| | **Friday Section** | **Monday Section** |
|---|---|---|
| **Hands-on Session and Introduction to the Experiment** | Friday<br>02/03/2023, 8:40-11:20 AM | Monday<br>02/06/2023, 9:00-11:40 AM |
| **Deliverables Due** | **Monday**<br>**02/13/2023, 11:59 PM** | **Monday**<br>**02/13/2023, 11:59 PM** |
| **Demo and Q&A** | **Wed-Fri**<br>**02/15-17/2023** | **Wed-Fri**<br>**02/15-17/2023** |