Laboratory 4

(Due date: **002/003**: March 5th, **004**: March 6th, **005**: March 7th)

OBJECTIVES

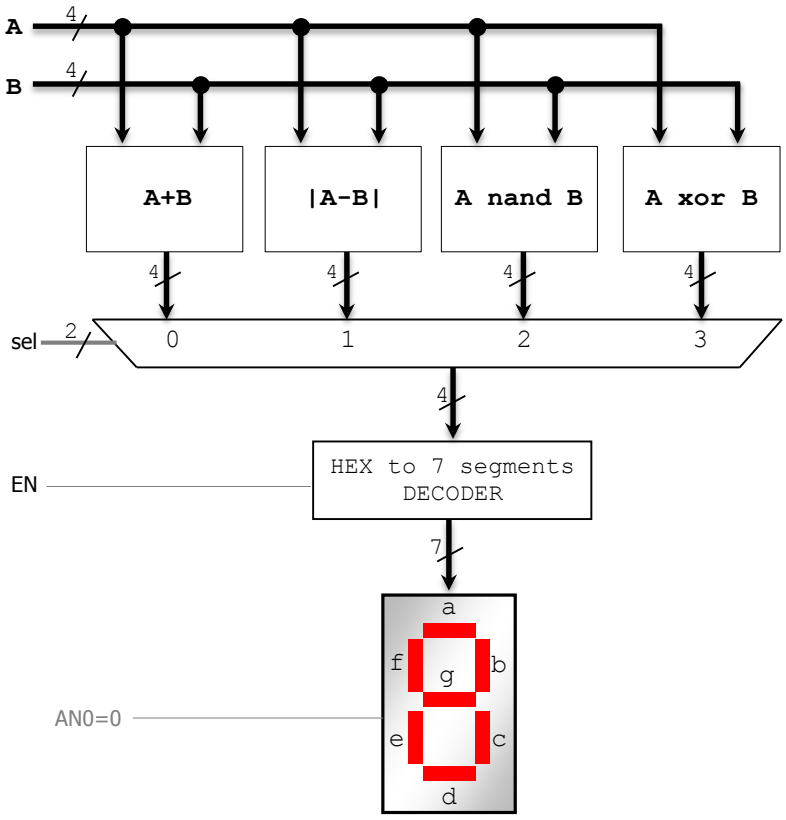
- ✓ Use the Concurrent Description and the Structural Description in VHDL.
- ✓ Implement Combinational circuits on an FPGA.

VHDL CODING

- ✓ Refer to the Tutorial: VHDL for FPGAs for a list of examples.

FIRST ACTIVITY: (100/100)

- **SIMPLE 4-BIT ARITHMETIC LOGIC UNIT (ALU):** This circuit selects between arithmetic (absolute value, addition) and logical (NAND, XOR) operations. Only one result (hexadecimal value) can be shown on the 7-segment display. This is selected by the input `sel(1..0)`.
- Input `EN`: If `EN=1` → result appears on the 7-segment display. If `EN=0` → all LEDs in the 7-segment display are off.
- Arithmetic operations: The 4-bit inputs `A` and `B` are treated as unsigned numbers.
  - ✓ `A+B`: If there is a carry out, ignore it.
  - ✓ `|A-B|`: 4-bit result, since `|A-B|` ∈ [0,15]. Tip: zero-extend the inputs to 5 bits and implement `A-B` (5-bit signed result). Then, implement `|A-B|`, where the 5-bit signed result is always positive. Finally, use the magnitude (4 LSBs) as the unsigned output.
- Logic Operations (`A xor B`, `A nand B`): These are bit-wise operations.
- **Nexys-4 DDR**: Each 7-segment display has active-low inputs (`CA-CG`) and an active-low enable `AN`. Make sure that only one 7-segment display is activated (e.g.: To use only the right-most 7-segment display, set `AN0=0`, `AN1-AN7=1`).



- ✓ Create a new VivadoE Project. Select the **XC7A100T-1CSG324 Artix-7 FPGA** device.
- ✓ Write the VHDL code for the given circuit.
  - **IMPORTANT**: For `A+B` and `|A-B|` circuits, you must use full adders and logic gates (as in Lab 2).
  - To implement the Bus MUX and decoder, it is strongly advised that you use the VHDL concurrent statements. To implement the top file, use the Structural Description: Create a separate file for the Arithmetic and Logic circuits, the 4-to-1 Bus MUX, and the Hex to 7-segment decoder.
- ✓ Write the VHDL testbench to properly test the circuit. Suggestion: Test 16 sets of A and B values. For each set, make `sel` vary from 0 to 3. Also, include at least one case where `EN=0` to verify that that the 7-segment display shows a 0.
- ✓ Perform Functional Simulation and Timing Simulation of your design. **Demonstrate this to your TA**.
- ✓ I/O Assignment: Create the XDC file. Nexys-4: Use `SW0` to `SW7` for the inputs A and B, `SW8` to `SW10` for the inputs `sel` and `EN`, `CA-CG` (7-segment display signals), and `AN7-AN0` (anode enable for each 7-segment display).
- ✓ Generate and download the bitstream on the FPGA and test. **Demonstrate this to your TA**.

- Submit (as a .zip file) all the generated files: VHDL code files, VHDL testbench, and XDC file to Moodle (an assignment will be created). DO NOT submit the whole Vivado Project.

TA signature: _____ **Date:** _____

Instructor: Daniel Llamocca
TAs: Jing Wu, Michael Lohrer, Saher Al-Khrayyef