EEE 491 Lab Assignment
Lab-CTRL: Controller of the system

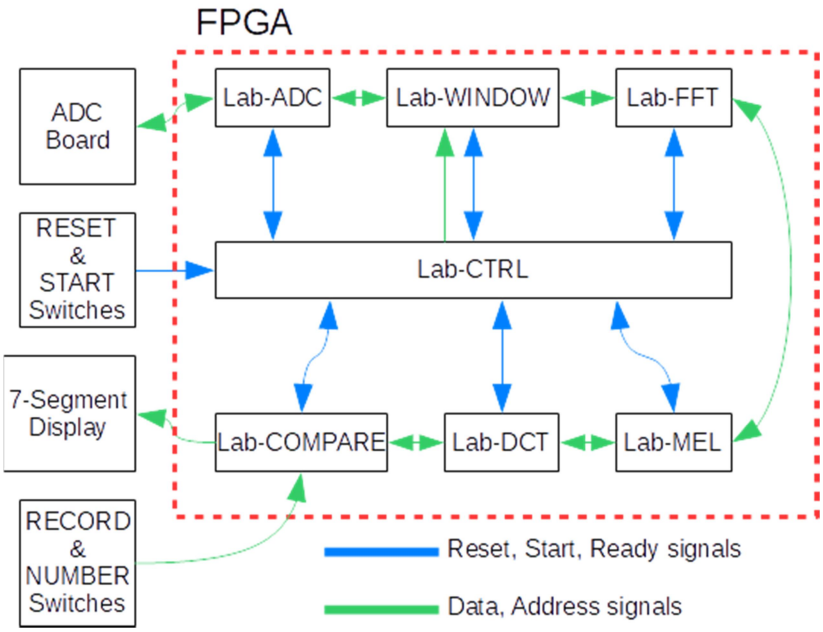
1. Technical Specifications/Requirements

The spoken number recognition project shall be composed of several building blocks (sub-systems) which are the lab assignments, as shown below.

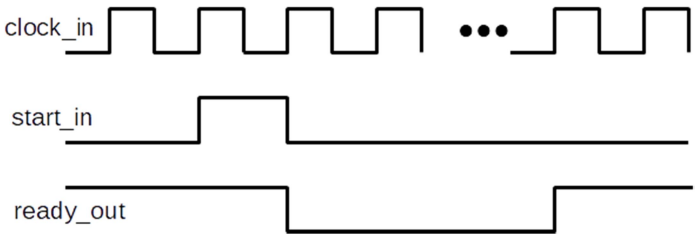
Each sub-system has an internal dual-port RAM that holds the processed data of the sub-system, which is read by the following sub-system. Lab-CTRL shall control the flow of these data along the sub-systems of the overall system by using the “start” and “ready” signals of each sub-system. Refer to waveforms below.

Lab-CTRL shall divide the speech signal (stored in Lab-ADC dual-port RAM) in to frames by generating frame start address to Lab-WINDOW. Note that frames must be overlapped by 50%. For each frame processing, Lab-CTRL shall send “start” signal to each sub-system and observe the “ready” signal from each sub-system. Lab-CTRL shall also control the Lab-DEBUG when attached to any of the sub-system output (not shown in the block diagram for clarity).

RESET and START switches on the Basys-3 shall be used. RESET switch signal shall be connected to the reset input of each sub-system on the FPGA. START signal is connected to Lab-CTRL, which shall start the overall system.



Each sub-system has a “start” control signal input and a “ready” status signal output. The start input control signal starts operation of the sub-system. The start input signal is logic-1 asserted at the rising edge of the clock as a single pulse for one clock-cycle duration (refer to waveform below). The ready output status signal indicates if the sub-system completed its operation and ready for the next start signal. The ready output signal is active high and asserted low just after the start signal is asserted and becomes active at the rising edge of the clock after the sub-system completes its operation (refer to waveform below).



Lab-CTRL in FPGA shall have the following input/output (IO) signals.

- “reset_in” input control signal, which resets all the registers in the design.
- ”clock_in” input clock signal of the 100MHz clock oscillator on the Basys-3 board.
- “start_adc_out” output control signal, which starts the Lab-ADC.
- “ready_adc_in” input status signal which indicates that Lab-ADC has completed its task.
- “frame_ addr_out” 14-bit address bus output signal to Lab-WINDOW for start address of a frame.
- “start_window_out” output control signal, which starts the Lab-WINDOW.
- “ready_window_in” input status signal which indicates that Lab-WINDOW has completed its task.
- “start_fft_out” output control signal, which starts the Lab-FFT.
- “ready_fft_in” input status signal which indicates that Lab-FFT has completed its task.
- “start_mel_out” output control signal, which starts the Lab-MEL.
- “ready_mel_in” input status signal which indicates that Lab-MEL has completed its task.
- “start_dct_out” output control signal, which starts the Lab-DCT.
- “ready_dct_in” input status signal which indicates that Lab-DCT has completed its task.
- “start_comp_out” output control signal, which starts the Lab-COMPARE.
- “ready_comp_in” input status signal which indicates that Lab-COMPARE has completed its task.
- “start_debug_out” output control signal, which starts the Lab-DEBUG.
- “ready_debug_in” input status signal which indicates that Lab-DEBUG has completed its task.
- “start_in” input control signal, which starts the Lab-CTRL. This signal is active high asserted.
- “ready_out” output status signal which indicates that Lab-CTRL has completed its task. This signal is active high and asserted low just after the start signal is asserted.

Write a test-bench in VHDL for the simulation of your design. Verify by using your test-bench simulation results that your design satisfies the technical specifications.

2. Demonstration (Checklist)

- Show that your design is implemented on FPGA.
- Present your test bench simulation results: inputs, outputs (waveforms). The 50% framing (frame addressing) and the sequence of operations must be shown clearly on the waveforms. Observe the input and output waveforms and verify the technical specifications listed above.

3. Guidance

Lab-CTRL is integrated with the other available sub-systems during the development of the project (system). On the Basys-3 board, use the start_in botton to start the operation and use a number of LEDs for ready_out signals of the sub-systems in order to observe the process sequence.

As you integrate progressively the other sub-systems, you shall use the Lab-DEBUG sub-system in order to observe the outputs of the integrated sub-system. Therefore, the Lab-CTRL design might be modified in order to use the Lab-DEBUG as the last integrated sub-system.

Keep versions of the Lab-CTRL as you modify the design.