Laboratory 5

Schematic description. Combinatorial logic circuits.

5.1 Objectives

The implementation methodology of digital systems using XILINX ISE was covered in details in the previous class. For this laboratory, the proposed objectives are the following:

- Exercising the implementation of digital systems on Xilinx FPGAs, starting from schematic description;
- Using the oscilloscope and logic analyser to investigate digital systems.

5.2 Implementation of combinatorial logic systems starting from schematic description

5.2.1 Logic function with two inputs

Implement a combinatorial logic circuit described by the equation:

$F1(A, B) = \sum(0, 1, 3)$

Follow the following steps:

- Complete the truth table.
- Draw the V-K diagram for the function, minimize it and deduce the minimal structure of logic gates that implements the desired functionality.
- Create a new project in Xilinx ISE and name it **F1**. Draw the schematic with logic gates. Save the file as **F1.sch**.
- Create the constraints file for the project. and save it as **F1.ucf**. Connect inputs A and B to SW1 (L14) respectively SW0 (L13) switches. Assign the output of the system to LD0 (F12) LED.

- Generate the configuration file **F1.bit** and download it into FPGA.
- Trigger the inputs by changing the position of the switches and observe the output (LED). Fill in the truth table for $F1$ and compare the experimental results with the ones obtained by analytical calculus.

5.2.2 Logic function with three inputs

Implement a combinatorial logic function described by the equation:

$F2(A, B, C) = \sum(1, 3, 4, 7)$

Follow the steps described in the previous paragraph. Create a new project in Xilinx ISE and name it **F2**. Draw the schematic with logic gates. Save the schematic as **F2.sch**. Create the constraints file for the project and save it as **F2.ucf**. Connect inputs A, B and C to SW2 (H18), SW1 (L14) respectively SW0 (L13) switches. Assign the output of the system to LD0 (F12) LED.

Trigger the inputs by changing the position of the switches and observe the output (LED). Fill in the truth table for $F2$ and compare the experimental results with the ones obtained by analytical calculus.

5.3 Using the oscilloscope to investigate combinatorial logic functions

The oscilloscope is an instrument which let us view the waveform of periodic signals. In order to be able to use an analog oscilloscope to investigate digital circuits we need the input and output signals to be periodical.

We'll make the following modifications to the project:

- Design and implement in FPGA a stimulus generator, which triggers the inputs of the investigated combinatorial circuits and ensures driving the circuit to all possible states. For combinatorial circuits with N inputs, the stimulus generator can be implemented as a synchronous N-bit counter. Figure 5.1 shows the schematic drawing of a stimulus generator like the one mentioned above.

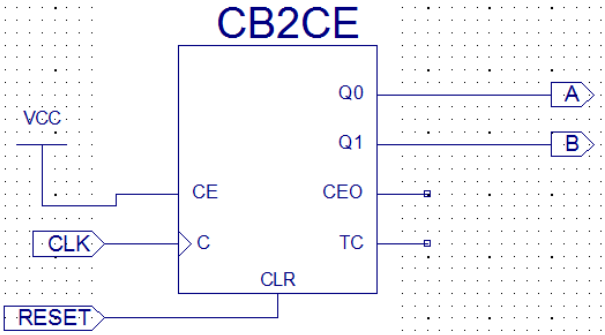

Figure 5.1 Periodical stimulus generator (**genSignals2.sch**).

A periodical stimulus generator can be implemented as a synchronous counter with 2 or more bits.

- Design of a system which instantiates the combinatorial circuit and the stimulus generator. The inputs and outputs of this circuit are going to be connected to the I/O connectors of the Spartan-3E board. The oscilloscope probes can access this way both the input and output signals of the tested combinatorial circuit. Figure 5.2 shows the graphical representation of the structure of the project having the instances of the two blocks.

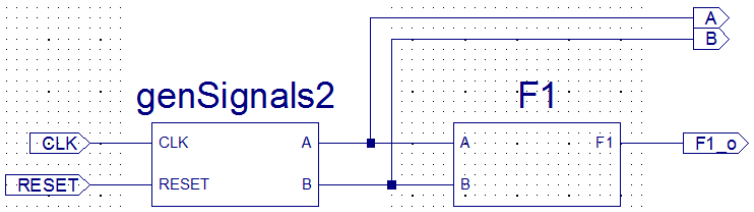

Figure 5.2 Analysed combinatorial circuit, connected to the counter-based stimulus generator (**F1test.sch**).

5.3.1 Logic function with two inputs

Investigate the same function:

$F1(A, B) = \sum(0, 1, 3)$

Based on **F1.sch**, create a symbol and save it as **F1.sym**.

Create a new schematic file and name it **F1test.sch**. Place an instance of the stimulus generator (**genSignal2.sym**) and an instance of function **F1** (**F1.sym**). Set **F1test** as the top module of the design. Create a constraint file and save it as (**F1test.ucf**). The content of the constraint file is the following:

```
# clock signal for the stimulus generator
NET "CLK"       LOC = "C9";                 # 50 MHz
# asynchronous reset
NET "RESET"     LOC = "K17" | PULLDOWN;     # BTN South

# input and outputs
NET "A"         LOC = "D7";     # J4 IO9
NET "B"         LOC = "C7";     # J4 IO10
NET "F1_o"      LOC = "E8";     # J4 IO12
```

Rerun the implementation process and ensure the **F1test.bit** was re-generated. Connect the oscilloscope probes (3 channels) to the proper I/O connector pins and view the waveforms. (CH1 = A[J4-IO9], CH3 = B[J4-IO10], CH4 = F1[J4-IO12]).

Which signal you think would proper for triggering the oscilloscope's time-base? The triggering source of the oscilloscope should have only one rising edge in each period. This condition is satisfied by the most significant bit of the counter used as the stimulus generator for out test case. So, in this case we should trigger the oscilloscope with the signal associated to port B.

Switch the oscilloscope to digital mode (push **ANALOG/DIGITAL** button and make sure the corresponding LED turns blue).

The image you should see on the oscilloscope is similar to the one shown in 5.3. You can observe the waveforms of the two inputs A and B (all 4 combinations) and the waveform of the output signal. The output is logic high when the inputs are triggered with one of the combinations that match the minterms of the function (0, 1 and 3).

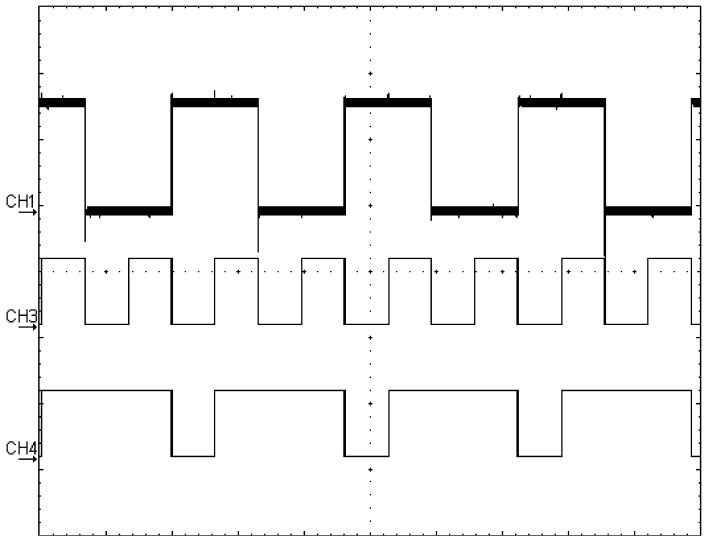

Figure 5.3 Oscilloscope image, investigating F1 (CH1 - A, CH3 - B, CH4 - F1).

5.3.2 Logic function with three inputs

Investigate the 3 input logical function in the same manner as described in the previous paragraph.

$F2(A, B, C) = \sum(1, 3, 4, 7)$

The constraint file saved as **F2test.ucf** will contain an additional line for input C:

```
NET "C"         LOC = "F8";     # J4 IO11
```

Use all four available channels of the oscilloscope to view the waveforms of the input and output. (CH1 = A[J4-IO9], CH2 = B[J4-IO10], CH3 = C[J4-IO11], CH4 = F2[J4-IO12]). You should get an oscilloscope image similar to the one shown is figure 5.4. Make a correlation between the the waveforms and the truth table of the function.

5.4 Using logic analyser to investigate combinatorial logic circuits

The logic analyser is an instrument with which one can visualize several digital signals in the same time. In contrast to the oscilloscope, the logic analyser can display non-periodical signals also. The logic analyser doesn't show the real waveform of a signal, but only the logic level of it.

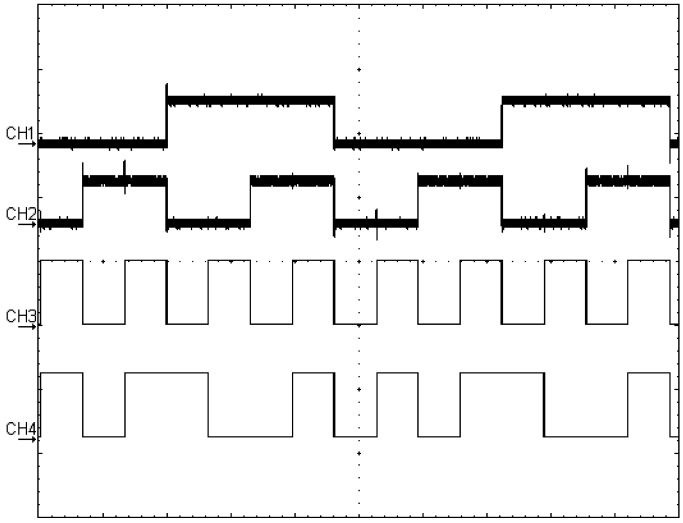

Figure 5.4 Oscilloscope image, investigation of F2 (CH1 - A, CH2 - B, CH3 - C, CH4 - F2).

5.4.1 Logic function with two inputs

Investigate the same function:

$F1(A, B) = \sum(0, 1, 3)$

To investigate the combinatorial circuit associated to $F1$ use the same bit-stream (**F1test.bit**) that we used for the experiment with the oscilloscope.

Connect the probes of the logic analyser to the proper pins of the I/O connector, as described in table 5.1. Start the application and analyser view the waveforms.

Table 5.1

Mapping the logic analyser's channels to the inputs and output of function F1.

Port	FPGA Pin	Board Connector	Channel, Connector logic analyser
A	D7	J4-I09	0, black/white
B	C7	J4-I010	1, brown/white
F1	E8	J4-I012	3, orange/white

Connect the ground channel of the analyser (black/grey) to one of the GND pins of the Spartan-3E board.

Create a new project for the logic analyser in which you activate and place just the channels you are going to use (0, 1 and 2). The mapping of the channels is suggested in table 5.1. Save the project file as **f1.LPF**.

Set the triggering source of the analyser as the rising edge of the signal A.

The image you should obtain is similar to the one shown in figure 5.5. You can observe the waveforms of the A and B input (all 4 possible combinations) as well as the waveform of the output F1. The output is logic high when the inputs are triggered with one of the combinations that match the minterms of the function (0, 1 and 3).

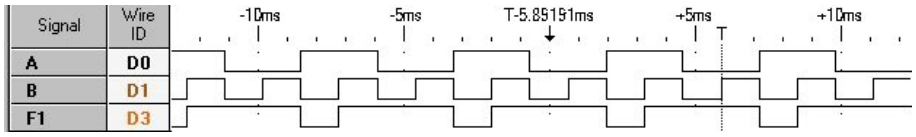

Figure 5.5 Waveforms of the logic analyser while investigating F1.

5.4.2 Logic function with three inputs

Investigate the 3 input logical function in the same manner as described in the previous paragraph.

$F2(A, B, C) = \sum(1, 3, 4, 7)$

The project file saved as **f2.LPF** will contains an additional line corresponding for input C, as shown in table 5.2.

Table 5.2

Mapping of the logic analyser's channels to the inputs and output of function F2.

Port	FPGA Pin	Board Connector	Channel, Connector Logic analyser
A	D7	J4-I09	0, black/white
B	C7	J4-I010	1, brown/white
C	F8	J4-I011	2, red/white
F1	E8	J4-I012	3, orange/white

Use 4 channels of the logic analyser to visualize the 3 inputs and output of the F2 function. You should obtain an image similar with the one shown in figure 5.6. Make a correlation between the the waveforms and the truth table of the function F2.

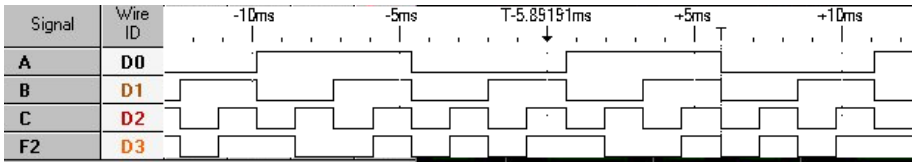

Figure 5.6 Waveforms of the logic analyser while investigating F2.