

Contents

Week 4	1
How to put something on the board ?	1
Find a way to interact with the JTAGG interface	2
Implementing a simple IP core	2
Chain 1	2
Chain 2	2
Implement a trivial top design	2
Notes	3
Questions	3

Week 4

Objectives : Start working with the FPGA and create a simple IP core to light up some leds

How to put something on the board ?

1. Synthesis : `yosys -s <your_script_name>.script`

Where the script looks something like this :

```
read -sv test_button_light.v
synth_ecp5 -top button_led -json test_button_light.json
```

2. Place and route : `nextpnr-ecp5 --timing-allow-fail --85k --package CABGA381 --json ./test_button_light.json --lpf fpga.lpf --textcfg test_button_light.config`

Where the lpf file looks like this :

```
# Buttons

# LOCATE COMP "buttons[0]" SITE "D19";
# IOBUF PORT "buttons[0]" PULLMODE=UP IO_TYPE=LVC MOS18;
# LOCATE COMP "buttons[1]" SITE "D17";
# IOBUF PORT "buttons[1]" PULLMODE=UP IO_TYPE=LVC MOS18;
# LOCATE COMP "buttons[2]" SITE "J16";
# IOBUF PORT "buttons[2]" PULLMODE=UP IO_TYPE=LVC MOS18;

LOCATE COMP "buttons[0]" SITE "M18";
IOBUF PORT "buttons[0]" PULLMODE=UP IO_TYPE=LVC MOS18;
LOCATE COMP "buttons[1]" SITE "M17";
IOBUF PORT "buttons[1]" PULLMODE=UP IO_TYPE=LVC MOS18;
LOCATE COMP "buttons[2]" SITE "L18";
IOBUF PORT "buttons[2]" PULLMODE=UP IO_TYPE=LVC MOS18;

# LED

LOCATE COMP "led[0]" SITE "A7";
IOBUF PORT "led[0]" IO_TYPE=LVC MOS18;
LOCATE COMP "led[1]" SITE "C7";
IOBUF PORT "led[1]" IO_TYPE=LVC MOS18;
LOCATE COMP "led[2]" SITE "E6";
IOBUF PORT "led[2]" IO_TYPE=LVC MOS18;

LOCATE COMP "RGB_Column[0]" SITE "B6";
IOBUF PORT "RGB_Column[0]" IO_TYPE=LVC MOS18;
LOCATE COMP "RGB_Column[1]" SITE "J3";
```

```
IOBUF PORT "RGB_Column[1]" IO_TYPE=LVCMS18;
LOCATE COMP "RGB_Column[2]" SITE "K3";
IOBUF PORT "RGB_Column[2]" IO_TYPE=LVCMS18;
LOCATE COMP "RGB_Column[3]" SITE "B17";
```

3. Create the bit file : `ecppack --compress --freq 62.0 --input test_button_light.config --bit test_button_light.bit`

4. Load the bit file to the FPGA : `openFPGALoader test_button_light.bit`

As an example I create a very simple design that uses the swithes and the button to light a specific LED

Find a way to interact with the JTAGG interface

For now try to use the openocd to communicate but it didn't work or at least i have no result on the borad
Some tries but nothing concluent

Implementing a simple IP core

ipcore.v The Ip core has two chains one for instruction 1 and one for instruction 2

Chain 1

chain1.v

The first chain is composed of two registers :

1. A 9 bits shift register that is shifted when data arrives on JTDI when instrction 1 is selected
2. A register of 9 bits that hold the value of the shift register when update state is reached, this register is also link to the outputs LEDS of the IPcore that ideally change the color of the 3 first leds of the selected colum

Chain 2

chain2.v

Very similar than the first one just the registers are 4 bits wide and are use ideally to select the column

Create a tb for this ipcore

Implement a trivial top design

top.v

Link the JTAGG interface and the ipcore in a single design

The tb for this module is lighting the leds : **red,yellow,green** of the third column of the led array

To do that it does:

1. Write the **8'h32** in the instruction to select the first chain
2. Send the data **9'b011001101** to this chain
3. *Update the leds*
4. Write the **8'h38** in the instruction to select the second chain
5. Send the data **4'b0011** to this chain to select the 3rd column
6. **Update the column***

In the idea this **top.v** could be used for the FPGA by simply changing the instantiation of the JTAGG component and not includ the **JTAGG.v** and the **jtag_tap.v**.

New instantiation (The inputs and outputs that should be link to the external pins automatically):

```
JTAGG JTAGG(
    .JTD02(s_JTD02),
    .JTD01(s_JTD01),
    .JTDI(s_JTDI),
    .JTCK(s_JTCK),
    .JRTI2(s_JRTI2),
    .JRTI1(s_JRTI1),
    .JSHIFT(s_JSHIFT),
    .JUPDATE(s_JUPDATE),
    .JRSTN(s_JRSTN),
    .JCE2(s_JCE2),
    .JCE1(s_JCE1)
);
```

Notes

To keep the LEDS in a consistent state I can simply iterate through all columns fast enough so that it does not blink for human eyes

The pins of the JTAG interface are can't be mapped if I try I receive an error saying it does not exist in the CABGA381 package.

And in the yosys files we can find this :

```
(* blackbox *) (* keep *)
module JTAGG(
    (* iopad_external_pin *)
    input TCK,
    (* iopad_external_pin *)
    input TMS,
    (* iopad_external_pin *)
    input TDI,
    input JTD02, JTD01,
    (* iopad_external_pin *)
    output TD0,
    output JTDI, JTCK, JRTI2, JRTI1,
    output JSHIFT, JUPDATE, JRSTN, JCE2, JCE1
);
parameter ER1 = "ENABLED";
parameter ER2 = "ENABLED";
endmodule
```

And on a [project](#) they instantiate the JTAGG interface by ommiting the inputs/outputs that have the (* iopad_external_pin *)

So my question is does the yosys will automatically links those wires to the actual pins ? If you don't know I'll try when I will be able to connect to the pins

Questions

- I hav been stucked on trying to manually set the JTAG pins, do you have any idee of how can do that ?