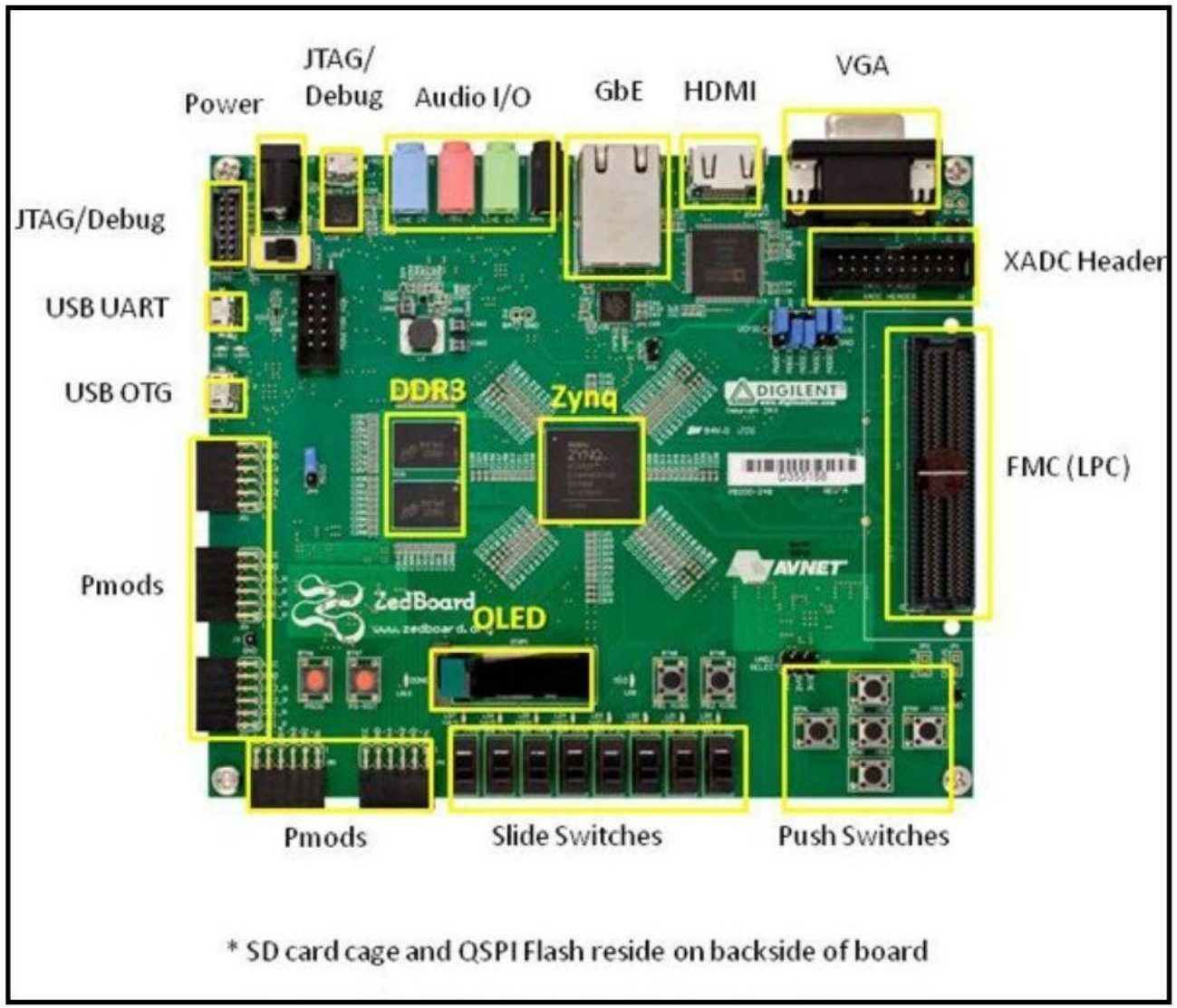
DCRUST - ECED

DIGITAL CIRCUIT IMPLEMENTATION ON ZEDBOARD

Aarti Kumari

Zedboard



FPGA:

- Xilinx Zynq-7000 SoC (Z-7020)
- Dual-core ARM Cortex-A9 processor
- Programmable logic (85K logic cells)

Memory:

- 512MB DDR3 RAM
- 256Mb Quad-SPI Flash
- 4GB SD card included (supports boot from SD)

Connectivity:

- Gigabit Ethernet
- USB OTG (micro-USB)
- USB UART
- HDMI output
- VGA output
- Audio In/Out (line-in, headphone out, mic)

Expansion:

- FMC (FPGA Mezzanine Card) connector
- 2 x Pmod connectors
- 1 x XADC header (analog input)

User Interface:

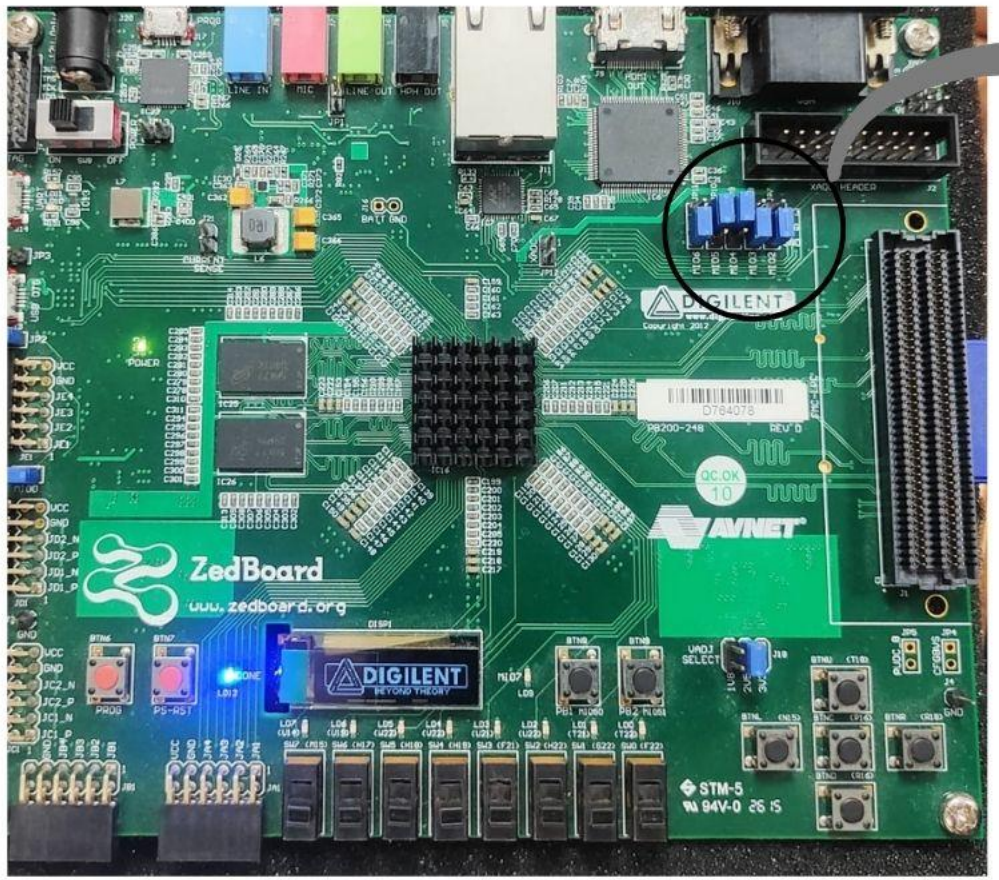
- Push buttons, slide switches
- LEDs (user and power)
- OLED display

Power:

- 12V input (AC adapter included)
- On-board power management

PIN Configuration

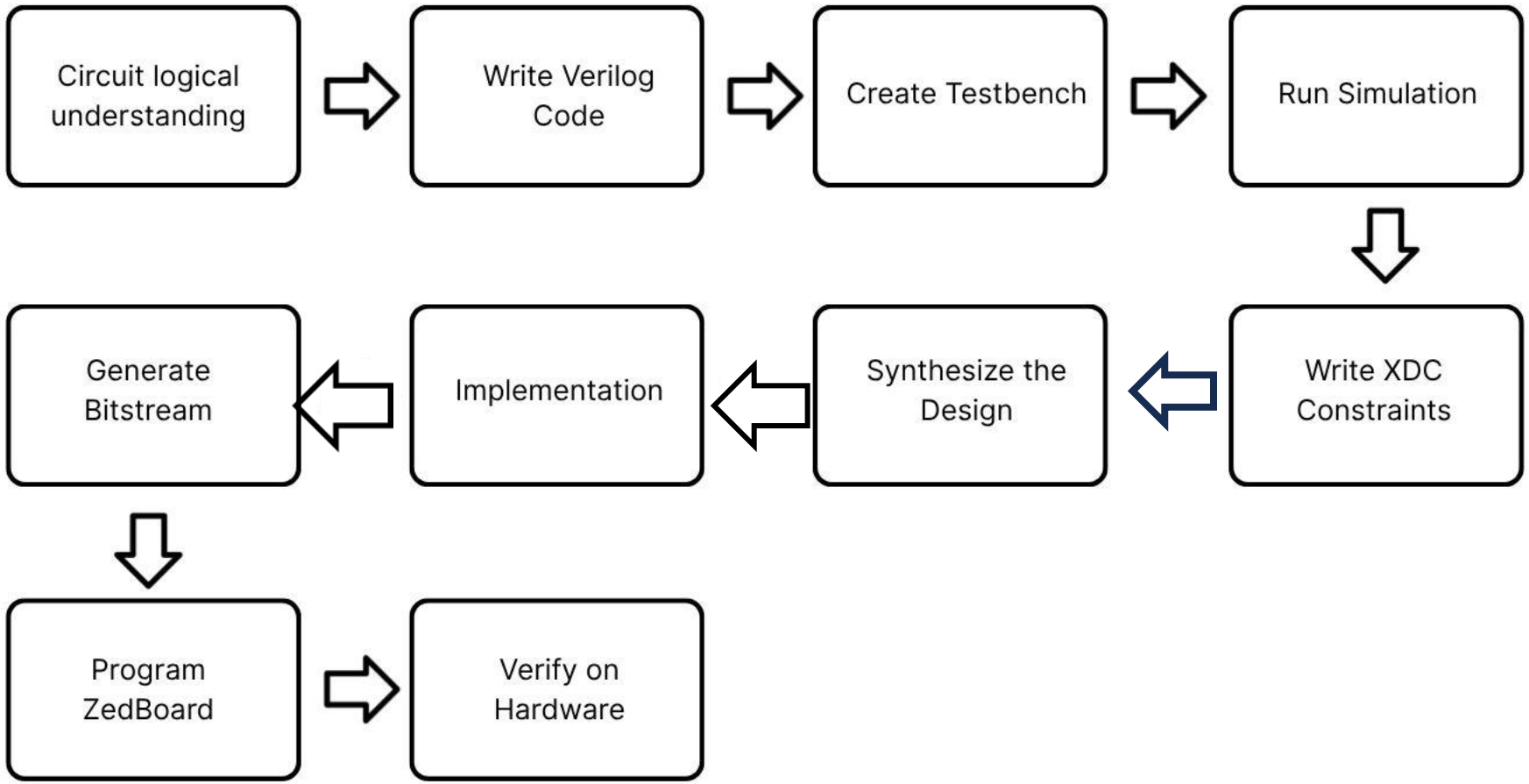
- Power on the ZedBoard with jumpers set and SD card inserted.
- Zynq's BootROM reads the SD card, loads BOOT.bin.
- FSBL runs, programs PL (FPGA bitstream), then hands off to U-Boot or Linux.
- System boots into Linux or your custom application.



SD CARD Boot mode:

Set JP 10 and JP 9 to 1, rest 0

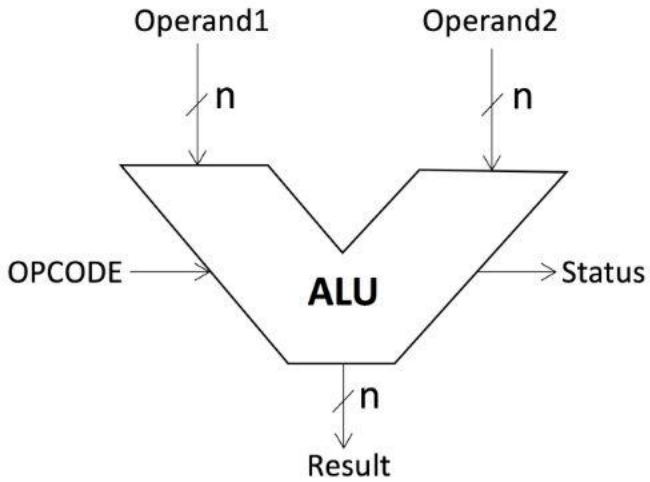
General Steps for FPGA-Based Circuit Implementation



ARITHMETIC LOGIC UNIT(ALU):

An ALU takes in:

- Two binary inputs: A and B (operands)
- One control input: OP (operation selector)
- And produces a single output: RESULT




```
module alu_zed (
    input [1:0] op,         // 2-bit operation selector
    input [3:0] a, b,       // 4-bit inputs from switches
    output reg [3:0] result
);
    always @(*) begin
        case (op)
            2'b00: result = a + b;
            2'b01: result = a - b;
            2'b10: result = a & b;
            2'b11: result = a | b;
            default: result = 0;
        endcase
    end
endmodule
```

OP Code	Operation
0	A + B
1	A - B
10	A & B
11	A | B

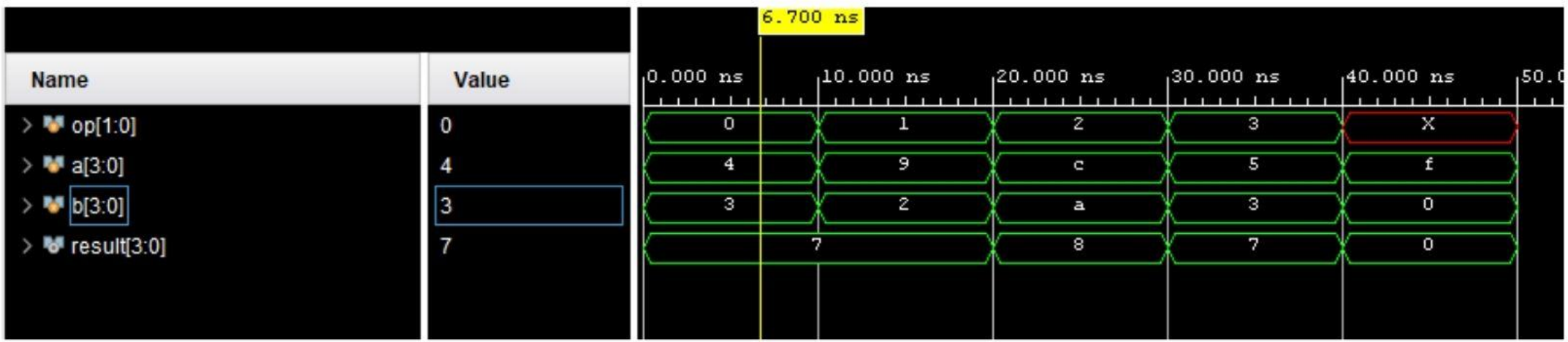
CONSTRAINT FILE

```
# INPUT SWITCHES a[3:0] - Connected to SW0-SW3
set_property PACKAGE_PIN F22 [get_ports {a[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[0]}]
set_property PACKAGE_PIN G22 [get_ports {a[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[1]}]
set_property PACKAGE_PIN H22 [get_ports {a[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[2]}]
set_property PACKAGE_PIN F21 [get_ports {a[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[3]}]

# INPUT SWITCHES b[3:0] - Connected to SW4-SW7
set_property PACKAGE_PIN H19 [get_ports {b[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[0]}]
set_property PACKAGE_PIN H18 [get_ports {b[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[1]}]
set_property PACKAGE_PIN H17 [get_ports {b[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[2]}]
set_property PACKAGE_PIN M15 [get_ports {b[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[3]}]

# BUTTONS for Operation Selection op[1:0]
set_property PACKAGE_PIN N15 [get_ports {op[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {op[0]}]
set_property PACKAGE_PIN P16 [get_ports {op[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {op[1]}]
```

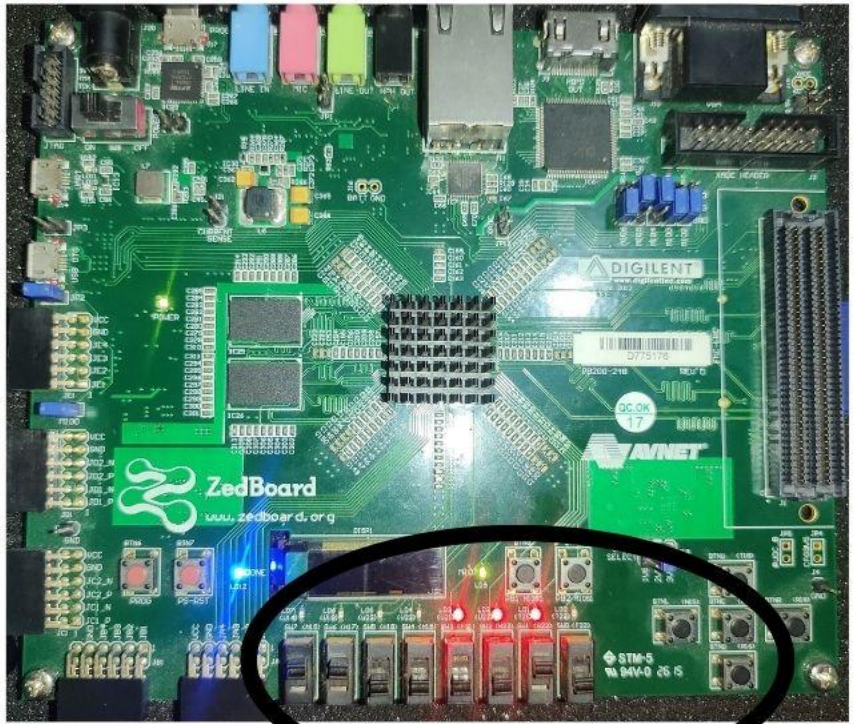
```
# LED OUTPUT result[3:0] → LD0-LD3
set_property PACKAGE_PIN T22 [get_ports {result[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {result[0]}]
set_property PACKAGE_PIN T21 [get_ports {result[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {result[1]}]
set_property PACKAGE_PIN U22 [get_ports {result[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {result[2]}]
set_property PACKAGE_PIN U21 [get_ports {result[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {result[3]}]
```

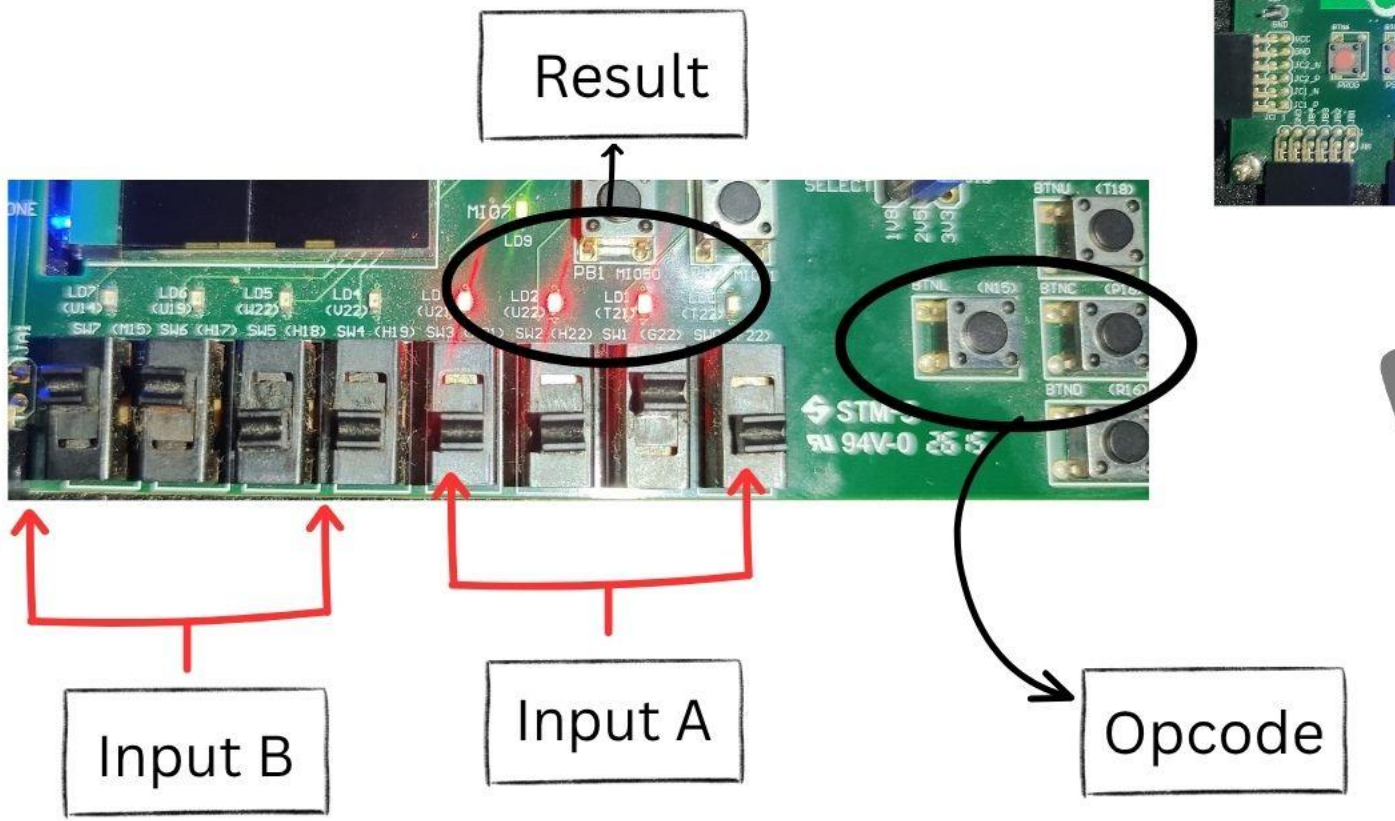
OUTPUT & WAVEFORM

Opcode	Operation	A	B	Result
00 (0)	ADD	0100 (4)	0011 (3)	0111 (7)
01 (1)	SUB	1001 (9)	0010 (2)	0111 (7)
10 (2)	AND	1100 (12)	1010 (10)	1000 (8)
11 (3)	OR	0101 (5)	0011 (3)	0111 (7)
xx (X)	DEFAULT	1111 (15)	0000 (0)	0000 (0)

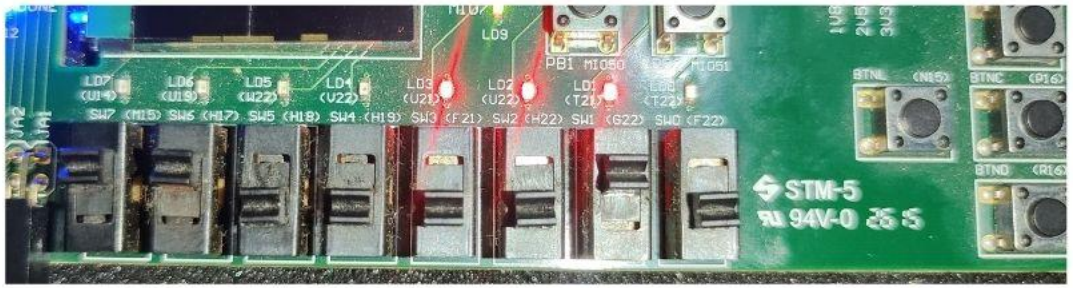


Opcode	Operation	A	B	Result
00 (0)	ADD	0100 (4)	0011 (3)	0111 (7)

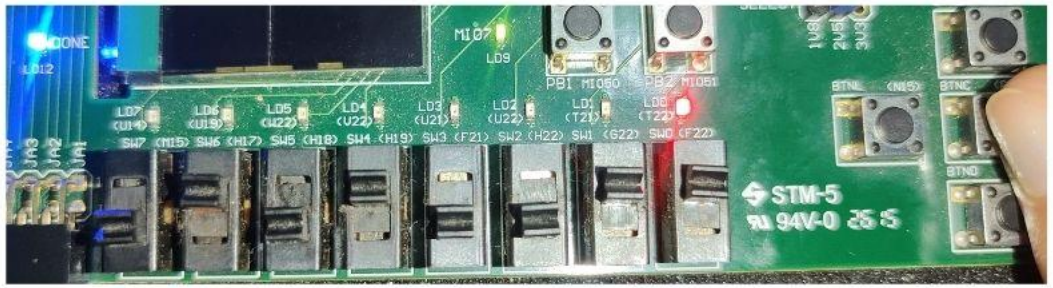




Verification on Zedboard

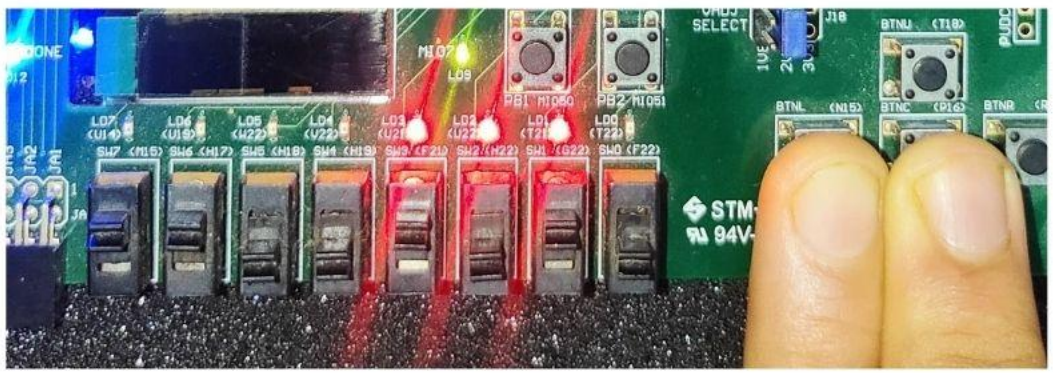


Test Case 1



Test Case 2

Test Case 3



Test Case 4