

# Digilent Basys3 FPGA Board

## Part 1

Ref: Digilent Basys3™ FPGA Board Reference Manual

黃稚存

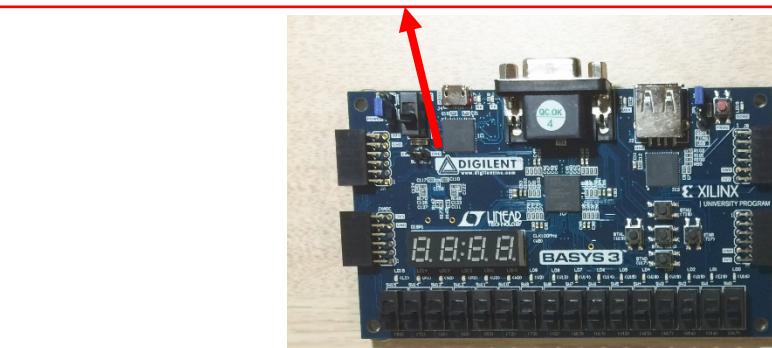


國立清華大學  
資訊工程學系

Lecture 03

# Unboxing

*Basys3 development board*



*Box*



*Pmod cable (optional)*

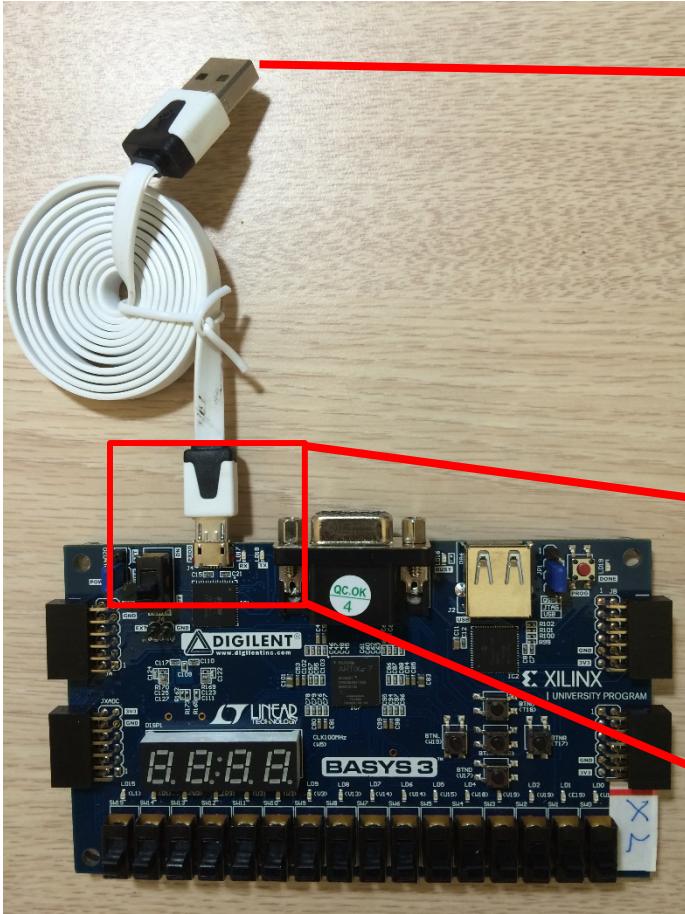


*PmodI2S  
audio amplifier*



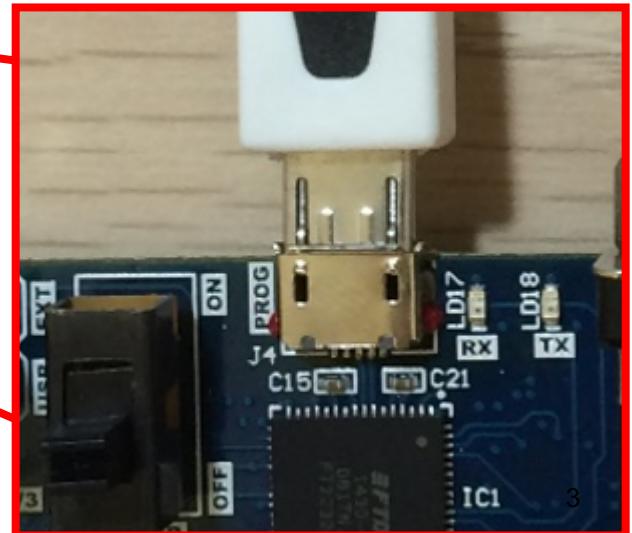
*USB-to-microUSB cable*

# USB Connection



Only 5V DC

Please mind the *direction* !!

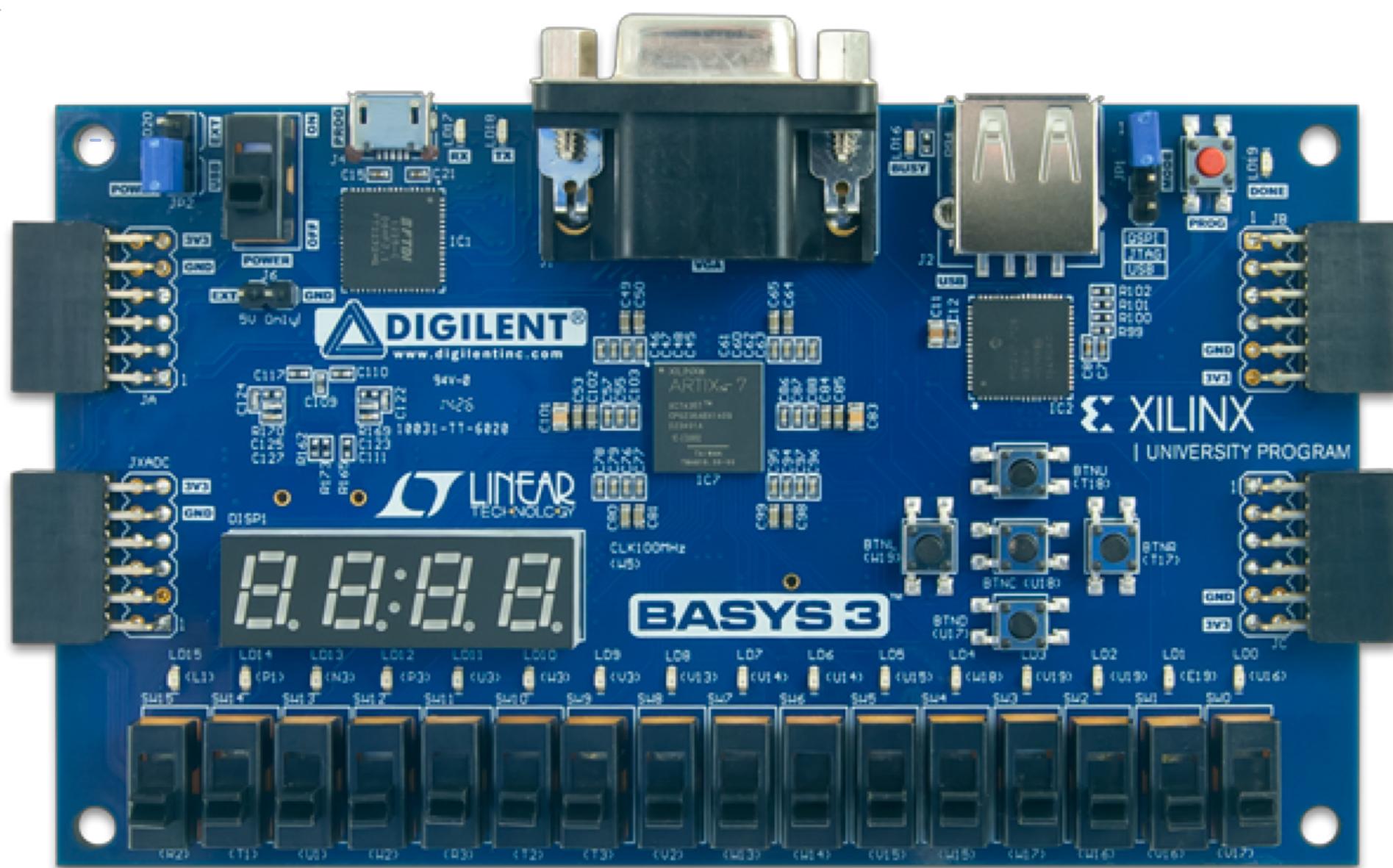


# Overview of Digilent Basys3

---



- Artix-7 FPGA (Field Programmable Gate Array) from [Xilinx](#)
  - ◆ Part number: XC7A35T-1CPG236C
- Low cost
- With USB, VGA, and other ports
- With switches, LEDs, and other I/O devices
- With uncommitted FPGA I/O pins to be expanded
  - ◆ Using Digilent Pmods
- Using Vivado Design Suite



# Artix-7 35T FPGA

---

- 33,280 logic cells in 5200 slices
  - ◆ Each slice contains four 6-input LUTs and 8 flip-flops
- 1,800 Kbits of fast block RAM
- Five clock management tiles
  - ◆ Each with a phase-locked loop (PLL)
- 90 DSP slices
- Internal clock speeds exceeding 450MHz
- On-chip analog-to-digital converter (XADC)

# Peripherals on Basys3

---

- 16 user switches
- 16 user LEDs
- 5 user pushbuttons
- 4-digit 7-segment display
- Three Pmod connectors
- Pmod for XADC signals
- 12-bit VGA output
- USB-UART bridge
- Serial flash
- Digilent USB-JTAG port for FPGA programming and communication
- USB HID host for mice, keyboards, and memory sticks

# Basys3 Demo Board

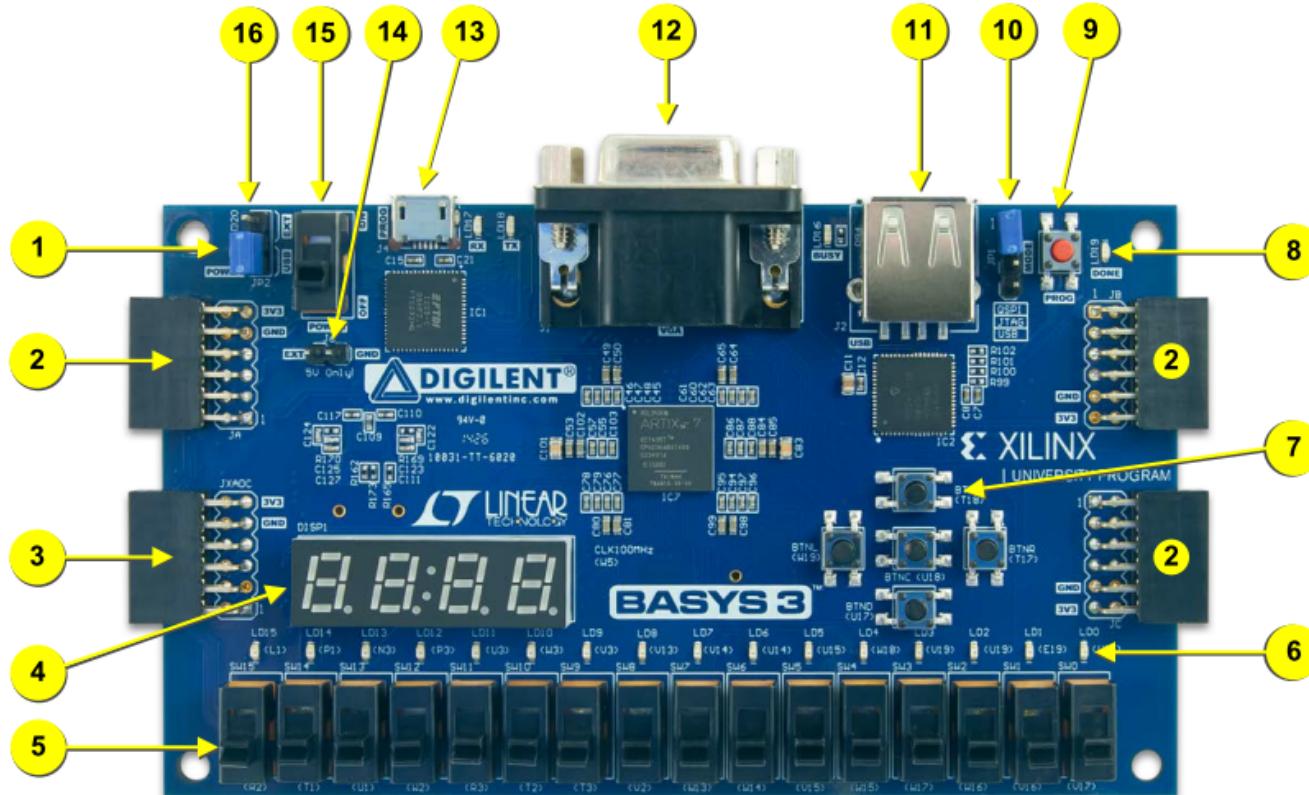


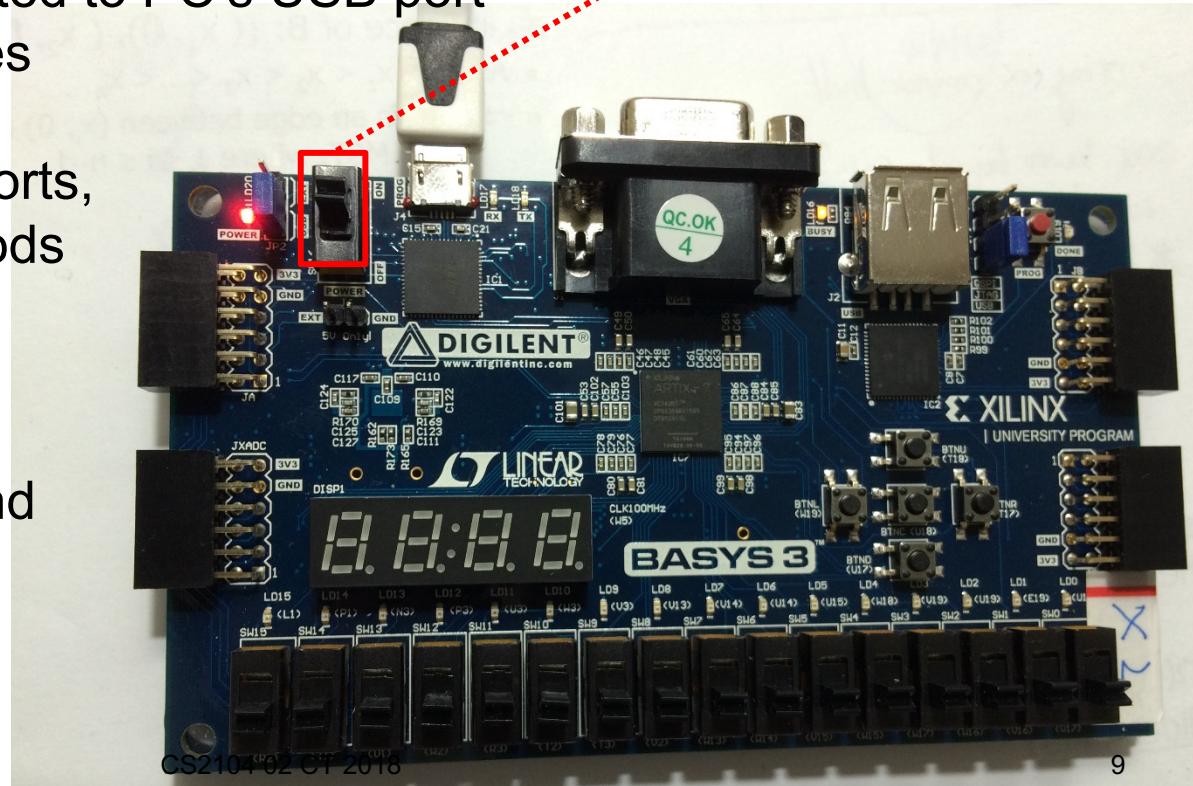
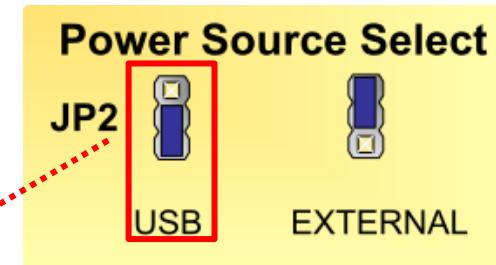
Figure 1. Basys3 FPGA board with callouts.

Callout	Component Description
1	Power good LED
2	Pmod connector(s)
3	Analog signal Pmod connector (XADC)
4	Four digit 7-segment display
5	Slide switches (16)
6	LEDs (16)
7	Pushbuttons (5)
8	FPGA programming done LED
9	FPGA configuration reset button
10	Programming mode jumper
11	USB host connector
12	VGA connector
13	Shared UART/ JTAG USB port
14	External power connector
15	Power Switch
16	Power Select Jumper

# Power Supplies

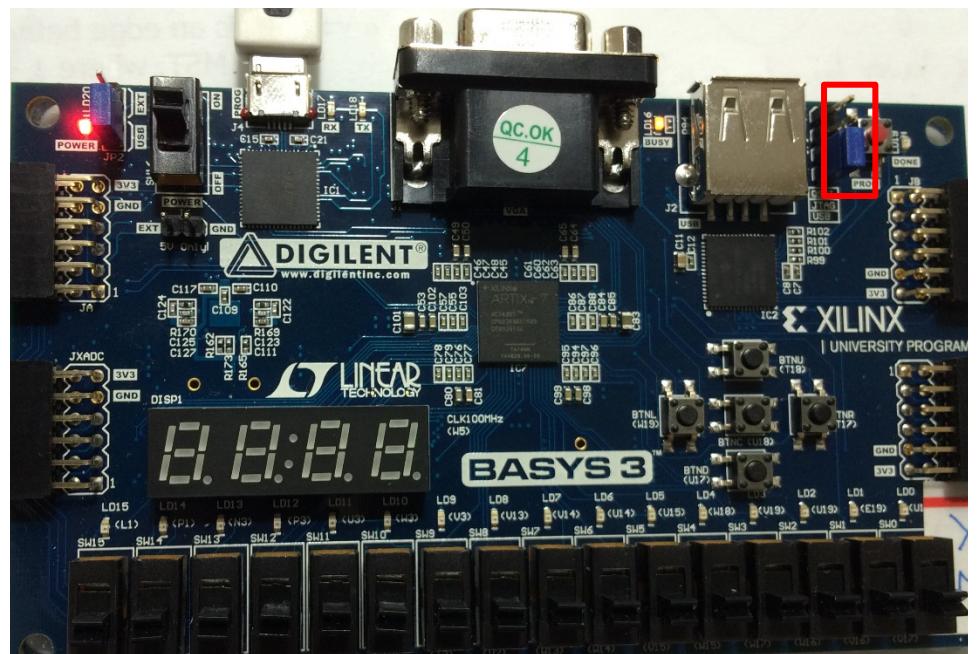
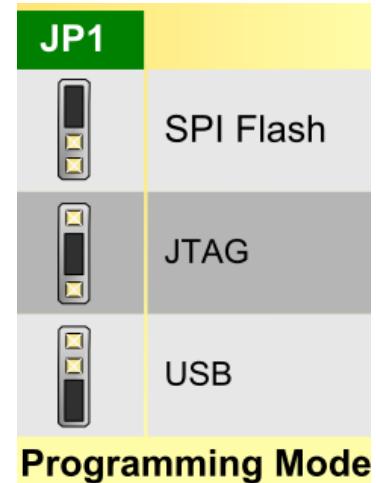
## To deliver 4.5-5.5V/1A (5W)

- The microUSB port
  - From a computer or battery pack
- A 5V external power supply
  - When with multiple peripheral boards
  - When not connected to PC's USB port
- Internal power supplies
  - 3.3V:  
FPGA I/O, USB ports,  
clocks, flash, Pmods
  - 1.0V:  
FPGA core
  - 1.8V:  
FPGA auxiliary and  
RAM



# FPGA Configuration

- FPGA must be configured (or programmed) to a specific hardware design, using “bitstreams” (in “Bit File”) via
  - ◆ USB to JTAG circuitry from a PC
  - ◆ Nonvolatile SPI Flash
  - ◆ USB memory stick



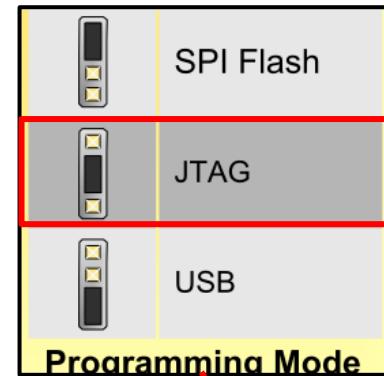
# Bitstreams

---

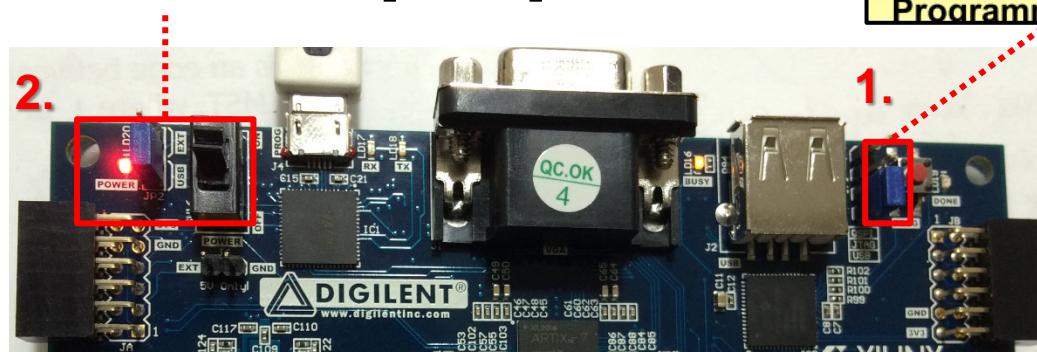
- FPGA configuration data
  - ◆ Stored in files \*.bit
  - ◆ Created using Vivado from Verilog designs
- 0 or 1 data stored in SRAM-based memory cells within FPGA
  - ◆ Defining FPGA's logic function and circuit connections
  - ◆ Remaining valid until power-off, reset, or reconfiguration
- Artix-7 35T bitstream has 17,536,096 bits
  - ◆ Long time to transfer
  - ◆ Compression can be enabled in Vivado
    - Up to 10x compression ratio

# Selection of Configuration Modes

Step 1:  
Change blue jumper to [ **JTAG** ] mode



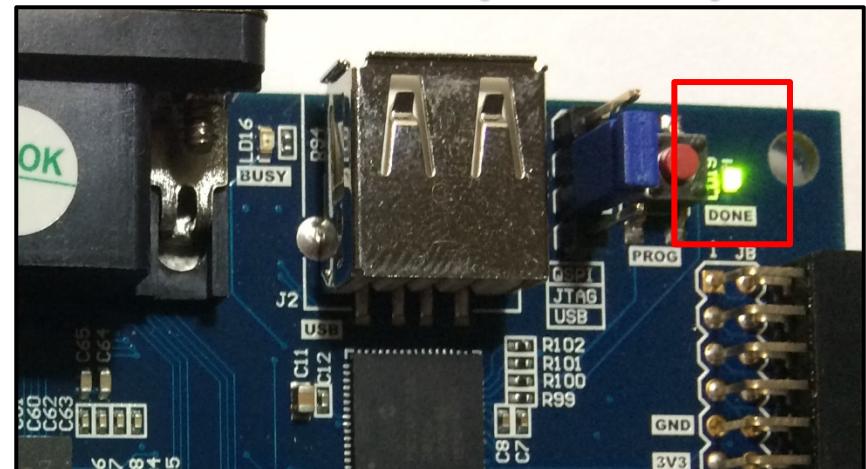
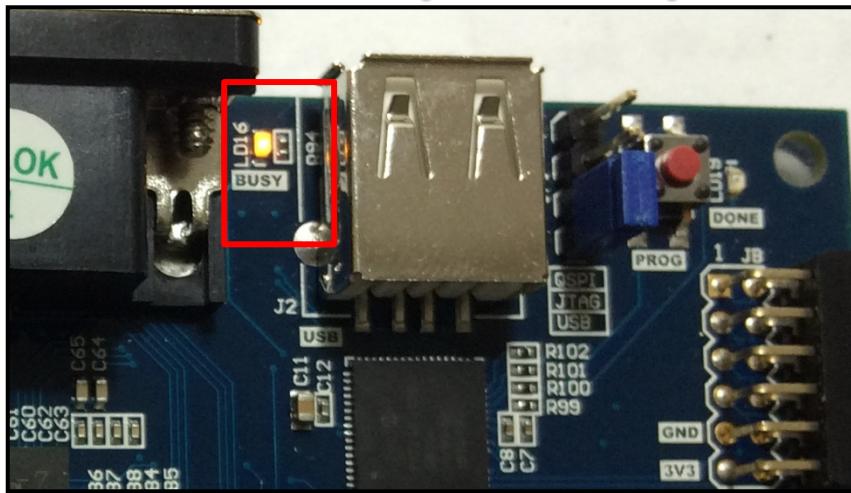
Step 2:  
Turn the power switch to [ **ON** ]



# Programing FPGA Device

- The LED indicates if the programming is done

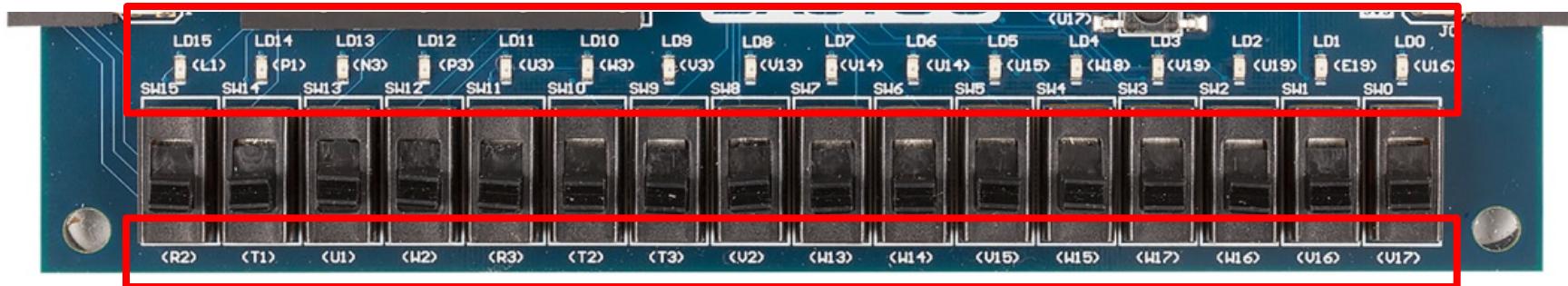
**BUSY (Yellow)** → **DONE (Green)**



# Basic I/O: Slide Switches & LEDs

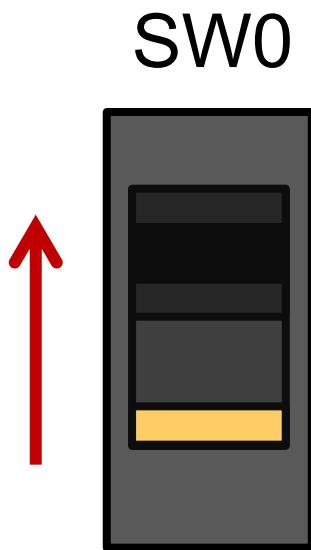
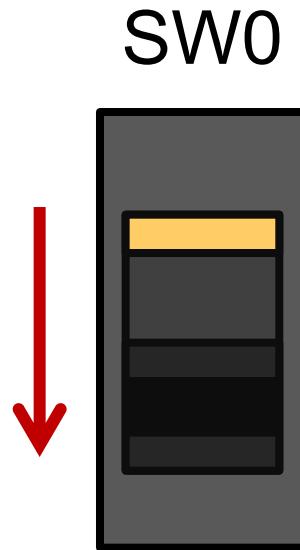
# 16 Slide Switches (1/2)

- Slide switches generate constant high or low signals depending on their physical positions



Symbol	SW15	SW14	SW13	SW12	SW11	SW10	SW9	SW8	SW7	SW6	SW5	SW4	SW3	SW2	SW1	SW0
FPGA Pin	R2	T1	U1	W2	R3	T2	T3	V2	W13	W14	V15	W15	W17	W16	V16	V17

# 16 Slide Switches (2/2)

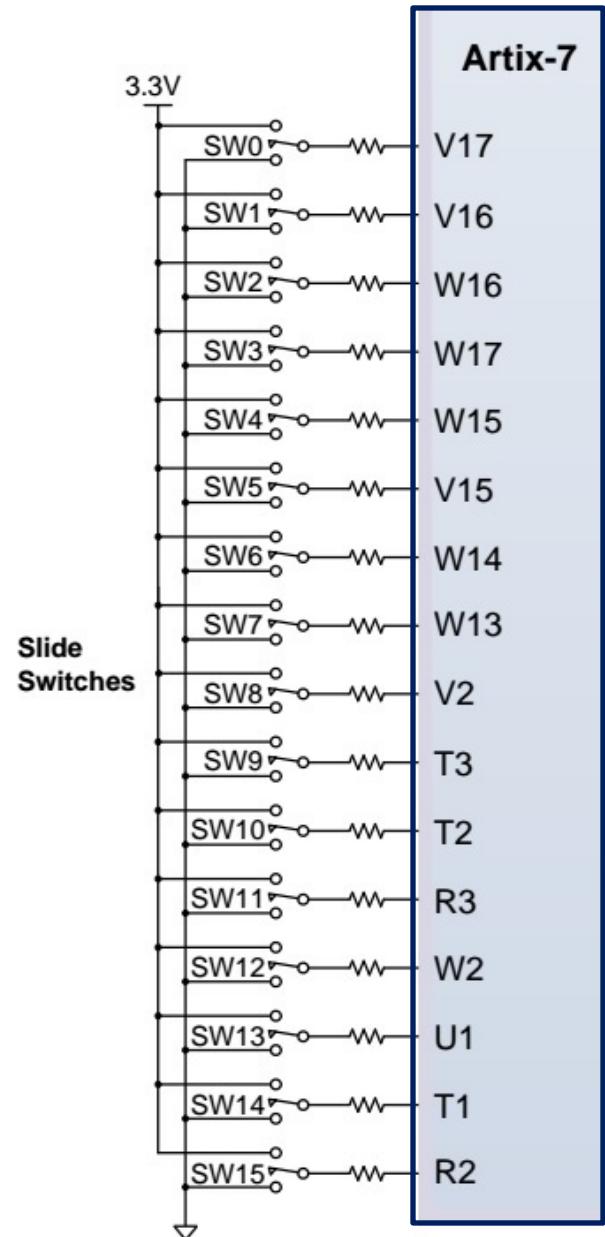


Switch down:

input value 0 into Artix-7

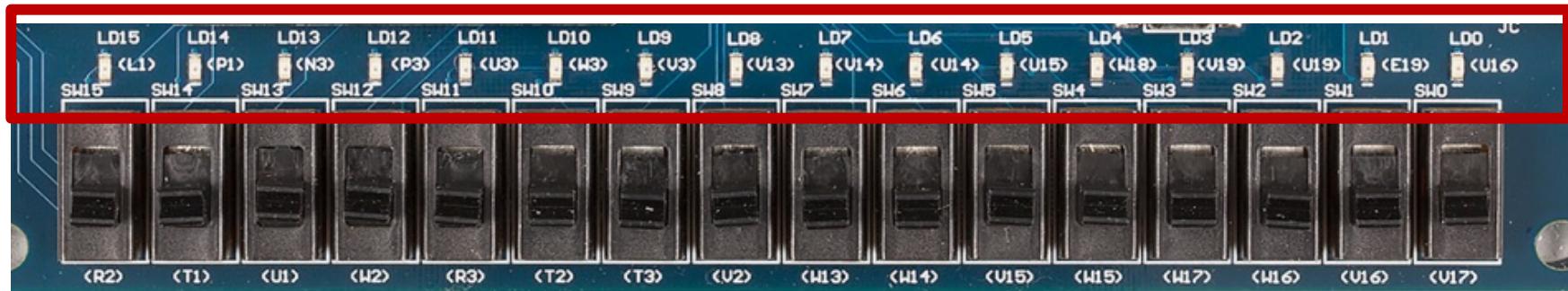
Switch up:

input value 1 into Artix-7



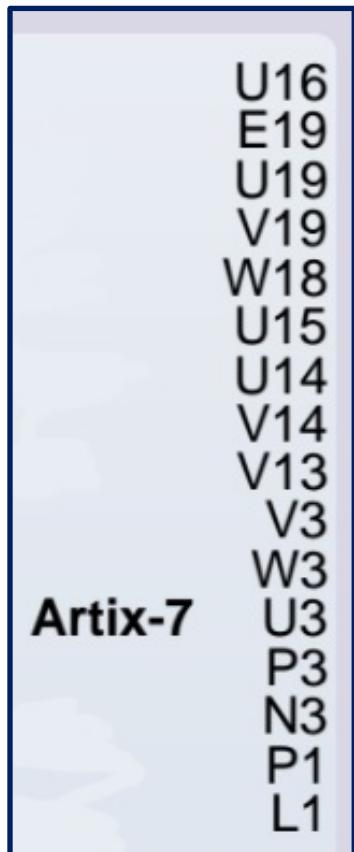
# 16 High-Efficiency LEDs (1/2)

- LEDs are anode-connected to the FPGA via 330 ohm resistors
  - Turned on when a logic high voltage is applied to their respective I/O pins.

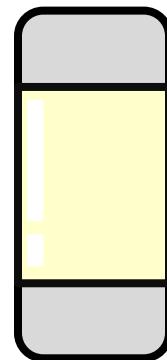


Symbol	LD15	LD14	LD13	LD12	LD11	LD10	LD9	LD8	LD7	LD6	LD5	LD4	LD3	LD2	LD1	LD0
FPGA pin	L1	P1	N3	P3	U3	W3	V3	V13	V14	U14	U15	W18	V19	U19	E19	U16

# 16 High-Efficiency LEDs (2/2)

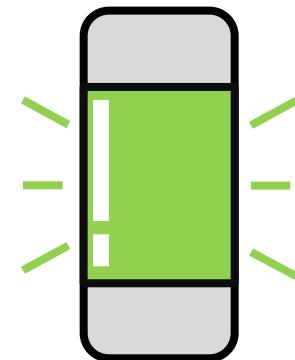


LD0



If input value = 0  
LED turn off

LD0



If input value = 1  
LED turn on

# Switches & LEDs Example

```
module led_sw_example(output [15:0]LED,  
                      input SW0);  
  
assign LED[15:0] = (SW0) ? 16'b1111_1111_1111_1111 : 16'b0000_0000_0000_0000;  
  
endmodule
```



Switch up  
Turn on all LEDs



Switch down  
Turn off all LEDs

# Oscillators/Clocks

# Clock Source

- The Basys3 board includes a single 100 MHz oscillator connected to pin W5
- Then different clocks can be generated by PLLs or self-designed clock dividers

Symbol	CLK100MHz
FPGA pin	W5



# Clock Divider Example ( $\text{clk}/2^3$ )

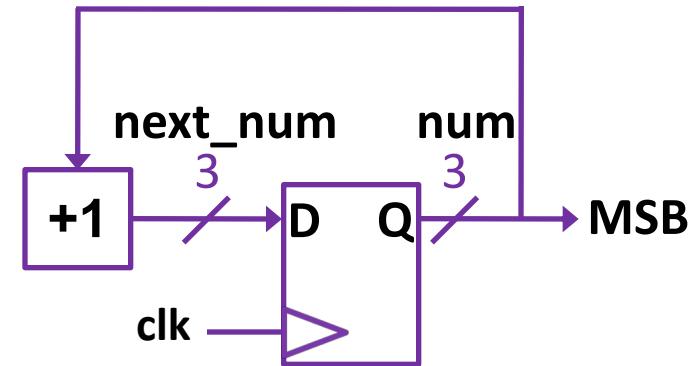
```
// 3-bit up counter as clock divider
module clock_divider (clk_div, clk);
input clk;
output clk_div;

reg [2:0] num;
wire [2:0] next_num;

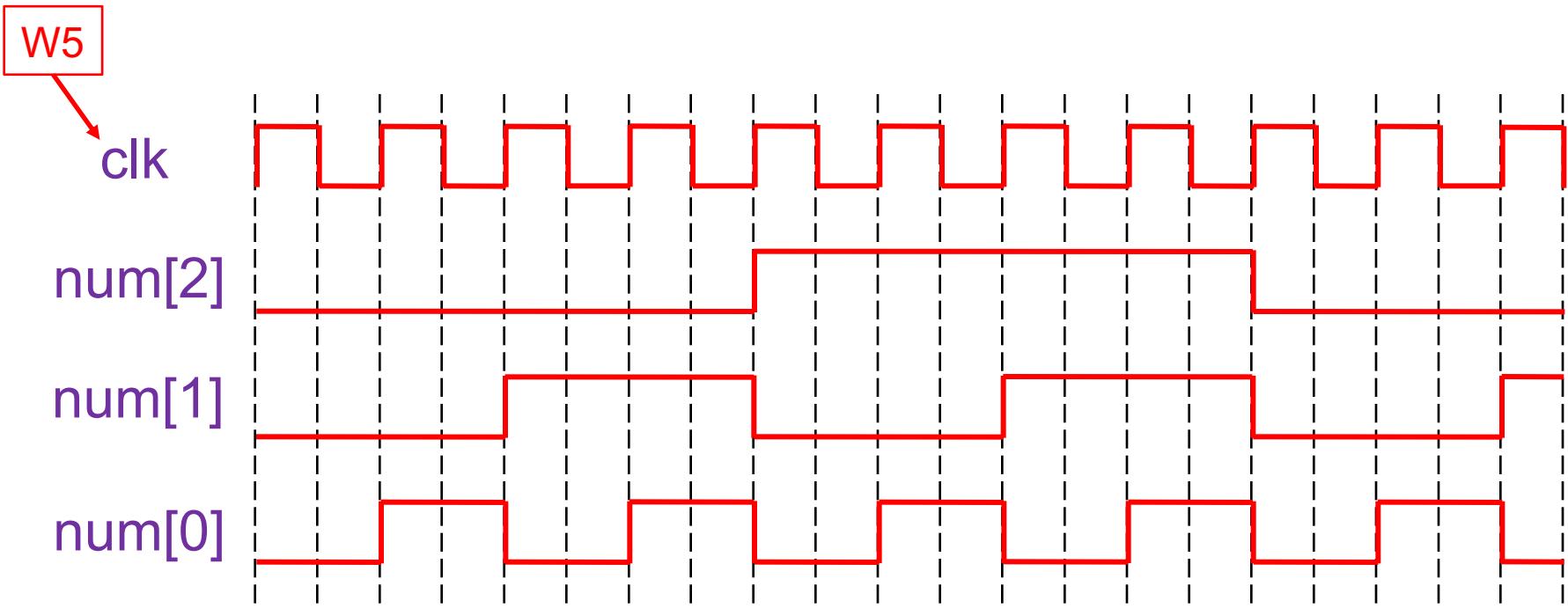
always @(posedge clk)
begin
    num <= next_num;
end

assign next_num = num + 1;
assign clk_div = num[2];

endmodule
```



# Waveforms



# Clock Divider vs. Multiplier

---

- Clock divider can be implemented by digital design
  - ◆ Using counters (divided by  $2^n$ )
  - ◆ Can you design a clock divider by arbitrary positive integer number?
    - E.g., divided by 3, 5, 10, 17, etc.
- Clock multiplier can be done by PLL (Phase Lock Loop)
  - ◆ Usually analog design
    - E.g., x2, x4, x1.5, etc.