Introduction to Keyboard on FPGA

Professor：Terng-Yin Hsu

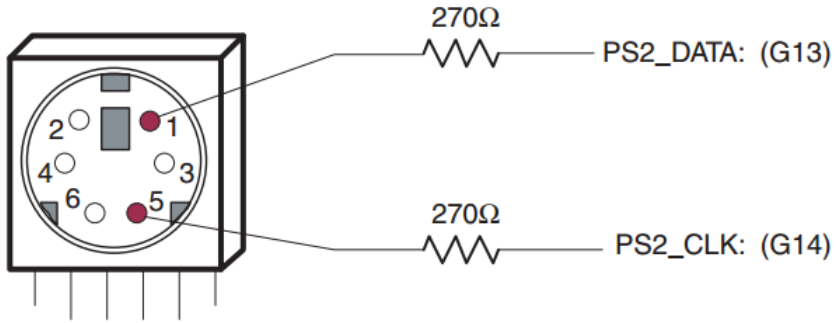
TA：Hsiao-Kai Liao

Outline

- Keyboard
  - Introduction
  - Working Principle
  - Code

Introduction

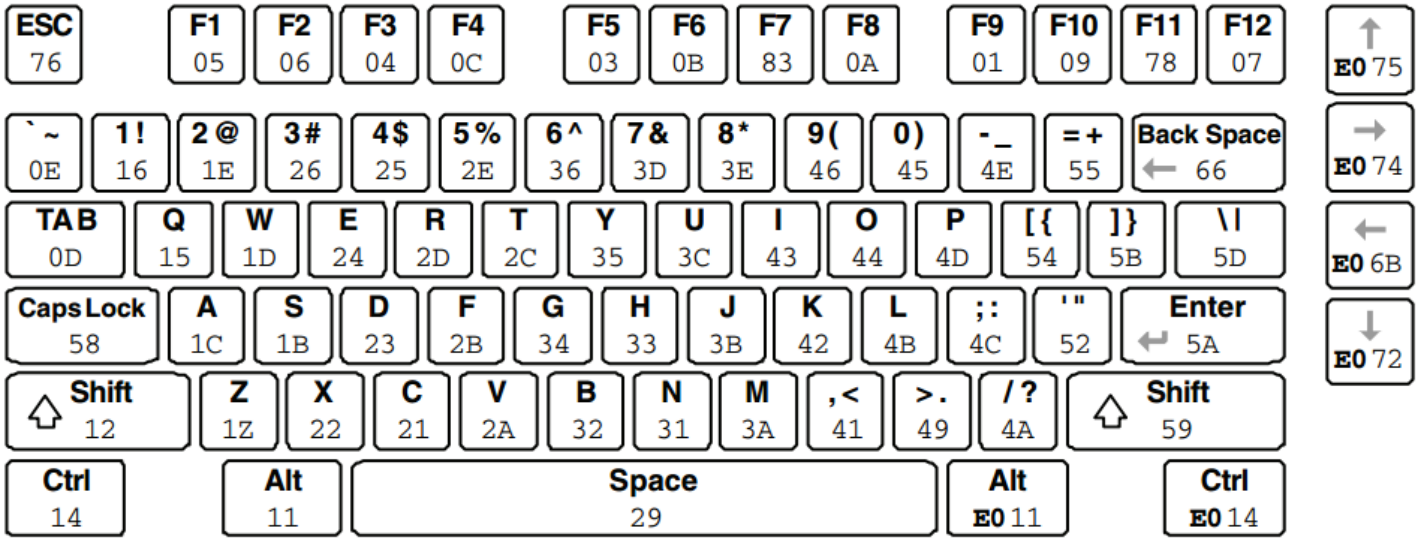
- The FPGA includes a 6-pin mini-DIN connector that can accommodate a PS2 mouse or PS2 keyboard connection.
- Both the mouse and keyboard use a two-wire serial bus (including clock and data) to communicate with a host device.
- If your keyboard is using USB to PS2 adapter, it may not work.



PS/2 DIN Pin	Signal	FPGA Pin
1	DATA (PS2_DATA)	G13
2	Reserved	G13
3	GND	GND
4	+5V	—
5	CLK (PS2_CLK)	G14
6	Reserved	G13

Working Principle(1/2)

- PS2-style keyboards use scan codes to communicate key press data
- Each key has a single, unique scan code.



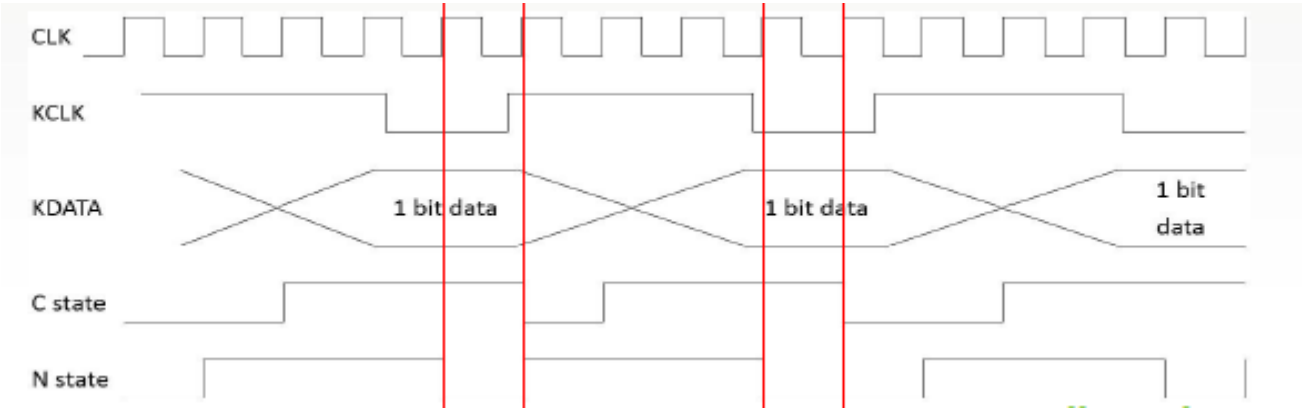
Working Principle(2/2)

- If the key is pressed and held
  - the scan code will be sent repeatedly once every 100ms
- If the key is released
  - a “F0” key-up code is sent, followed by the scan code of the released key.
- Both use 11-bit words that include a start, 8-bit data, odd parity bit and stop.

Code(CLK & KCLK)

```
always@(posedge CLK) begin
    C_state<=N_state;
    N_state<=KCLK;
end

always@(posedge CLK) begin
    case({C_state, N_state})
      2'b10:key_reg<={KDATA,key_reg[21:1]};
      default: key_reg<=key_reg;
    endcase
end
```



Code(Receive data)

- You have to count 11 times to receive the keyboard data
- When the key is released, you will receive “F0”

```
assign check = key_reg[1]^key_reg[2]^key_reg[3]^key_reg[4]^key_reg[5]^key_reg[6]^key_reg[7]^key_reg[8]^key_reg[9];
// parity pit

always@(posedge CLK) begin
    if(key_counter==4'd0&&check==1'b1)begin
        if(key_reg[8:1]==8'hXX) // XX :
            key_data<=8'd0;
        else
            key_data<=key_reg[19:12];
    end
    else
        key_data<=key_data;
end
```

Thank for your attention