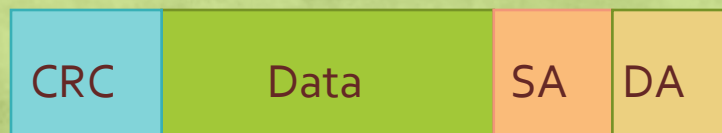
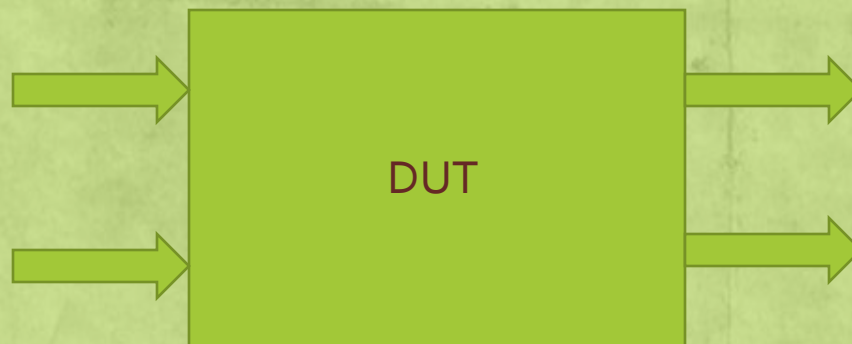


Exercise 2: Coding

- Create a simple design for a 2x2 port switch that we saw in first section?
- Design spec (same as in case study in first section)

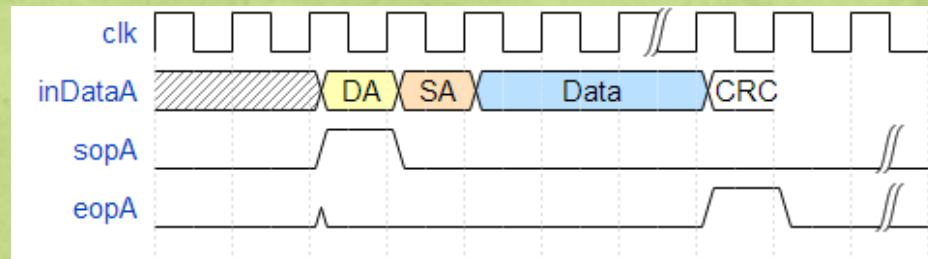


Packet Format

- 4 Bytes of Destination Address (DA)
- 4 Bytes of Source Address (SA)
- 4 Bytes of CRC
- Data Bytes - Minimum - 52 Bytes
Maximum - 1506 Bytes
- Total Size – Min – 64Byte and Max = 1518 Bytes

Design Spec

- 2 input and 2 output ports (A and B) – both on same clk
- Packets are of random sizes
- Packets can come back to back or with idle in between
- Both ports has a 32 bit data bus for packet data along with a start and end of packet pulse indication --→



- Assume port A and B has following address

```
parameter PORTA_ADDR = 01;  
parameter PORTB_ADDR = 02;
```

- Use a simple logic to decode Destination Addr (DA) and decide output port
- Store packet data in a FIFO and then send the data to that output port in the same format as received

Design Interface spec

- Use following top level module signals and inputs and outputs

```
module eth_sw_2x2 (  
    input clk,           //clk input  
    input rstN,          //active low reset  
    input [31:0] inDataA, //port A input data (32 bits per clk)  
    input sopA,          //port A input pulse indicating start of packet  
    input eopA,          //port A input pulse indicating end of packet  
    input [31:0] inDataB, //port B input data (32 bits per clk)  
    input sopB,          //port B input pulse indicating start of packet  
    input eopB,          //port B input pulse indicating end of packet  
    output [31:0] outDataA, //port A output data (32 bits per clk)  
    output sopA,         //port A output pulse indicating start of packet  
    output eopA,         //port A output pulse indicating end of packet  
    output [31:0] outDataB, //port B output data  
    output sopB,         //port B output pulse indicating start of packet  
    output eopB,         //port B output pulse indicating end of packet  
    output portAStall,    //stall indication to port A when switch cant accept packets  
    output portBStall     //stall indication to port B when switch cant accept packets  
);
```


Tools that can be used

- Once you have a sample code use following browser based Tool for compilation and eventually to simulate and verify – which you will learn by end of course
 - www.edaplayground.com
 - It is simple to use but in case you still need instructions refer to <http://eda-playground.readthedocs.org/en/latest/intro.html>
- Timing diagram:
 - <https://code.google.com/p/wavedrom/>

Discussions

- Make any other assumptions that you need.
- **In later exercises we will build a testbench around this and verify this same design !**
- And please discuss your solutions in the comments section with other students in the community!