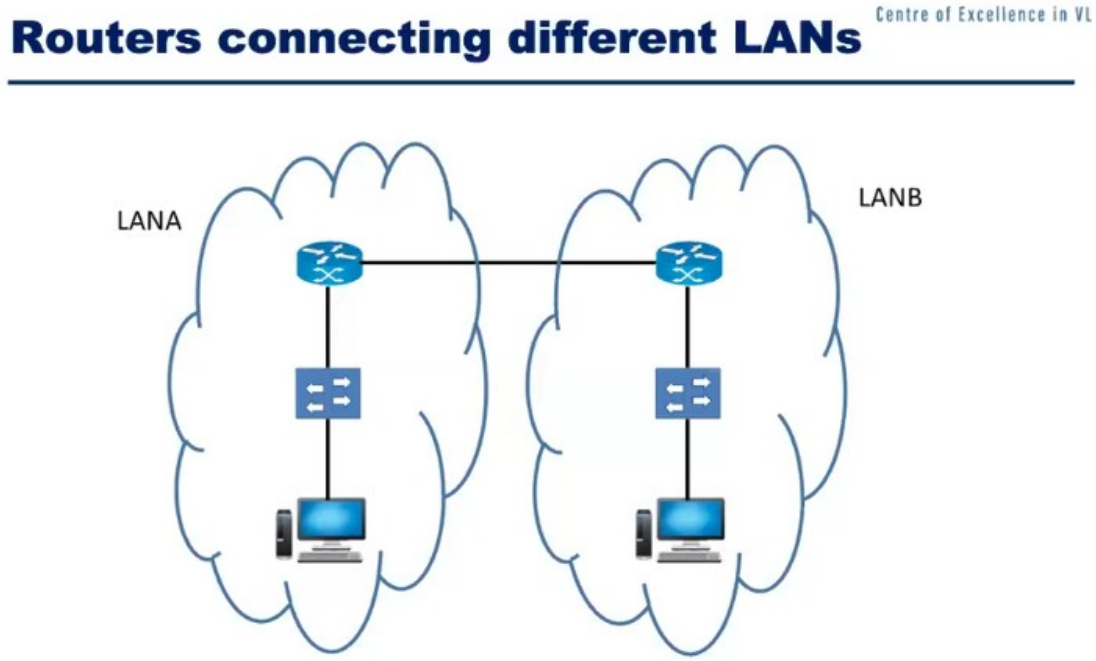
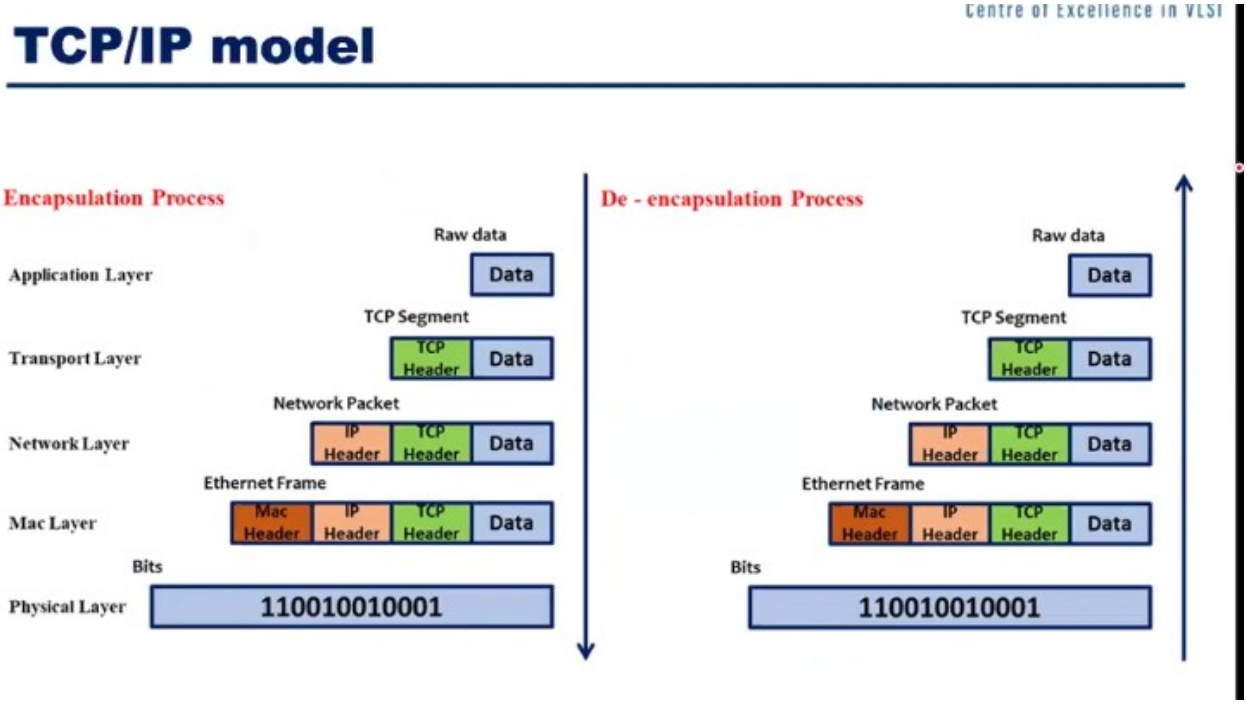
Router 1x3 Project:

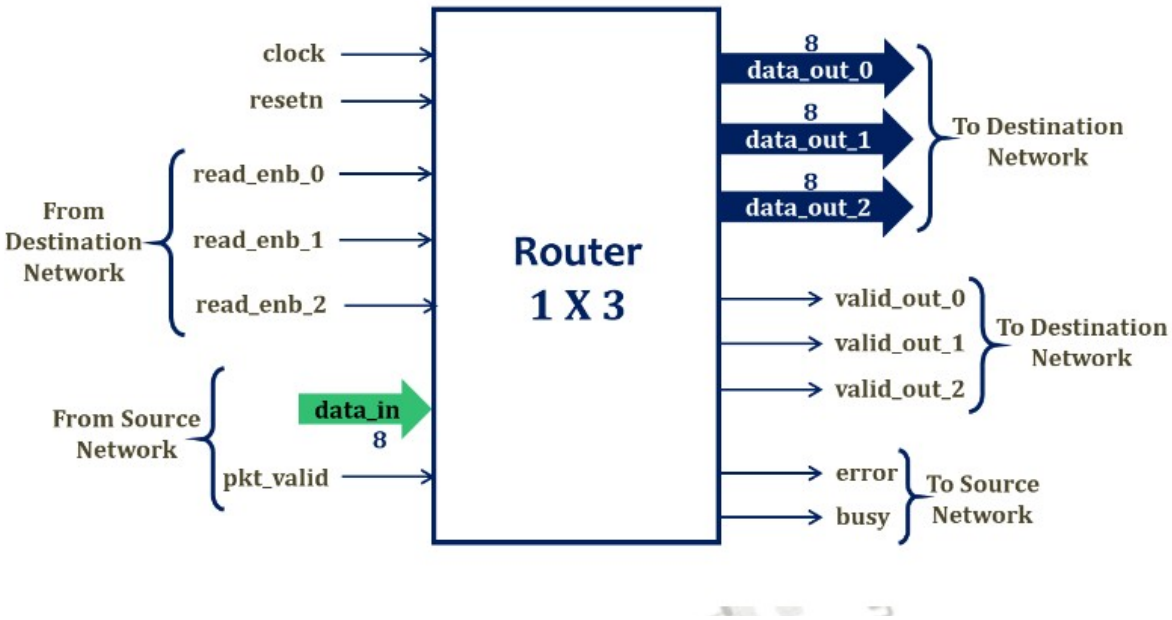
INTRODUCTION:

- A router is a networking device that forwards data packets between computer LANs.
- It follows TCP/IP layer 3 network layer protocol and does routing of te packets.





TOP BLOCK:



Interface:

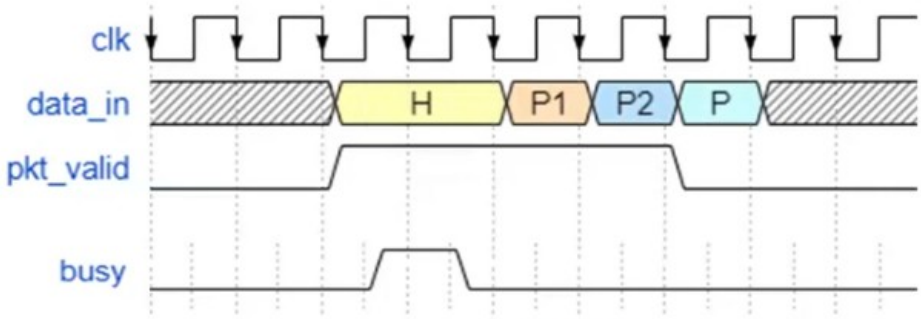
Busy: if all FIFOs are full then no new data will be accepted
Valid_out: indicates weather destination is ready to accept data
read_enb: if valid out is high, it indicates there is data available to read then read_enb can be made high, then data can be read by the destination.
If readenb is not made high for 30 clocks CLKs after valid_out is made high then a soft reset is triggered which will wipe out the entire FIFO.
error:

If destination is not active but data is continously read then it will be stored in FIFO blocks indicated ny data_out_0, data_out_1, data_out_2.
When valid_out is enabled, read_enb for particular FIFO will be high and the data stored in the FIFO is read by the destination.

pkt_valid: pkt_valid will be high from header to payload
Busy: if busy is high, it stops receiving new bytes and continues till the it is high

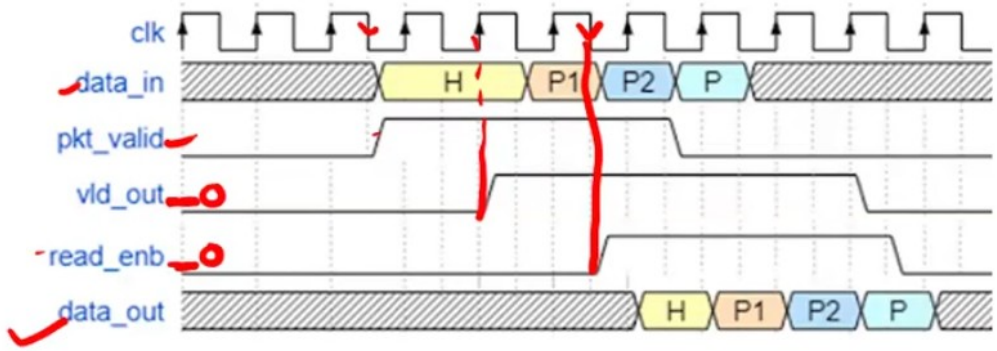
Router input protocol

Centre of Excelle

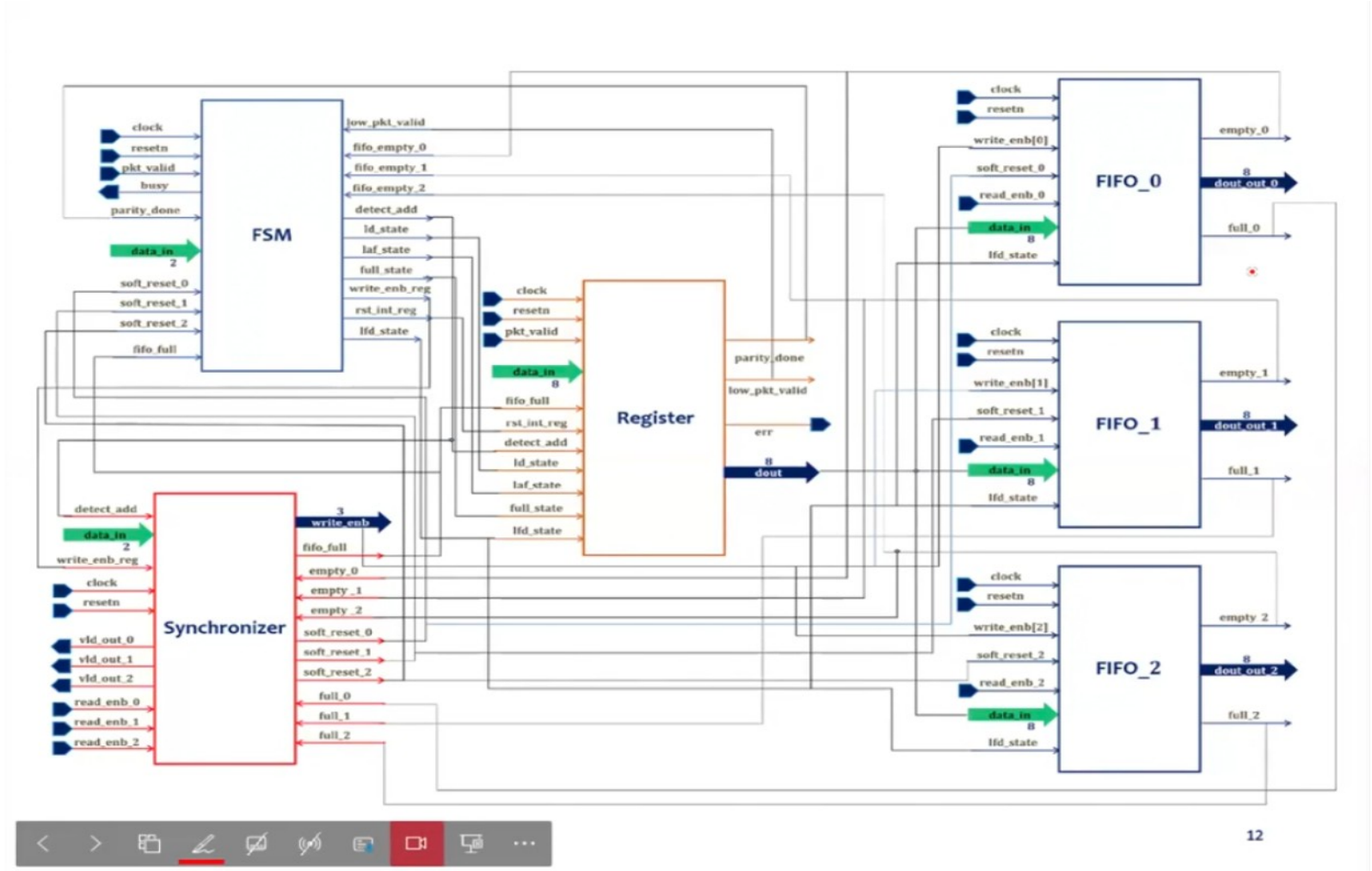


Router output protocol

Centre of Excellence

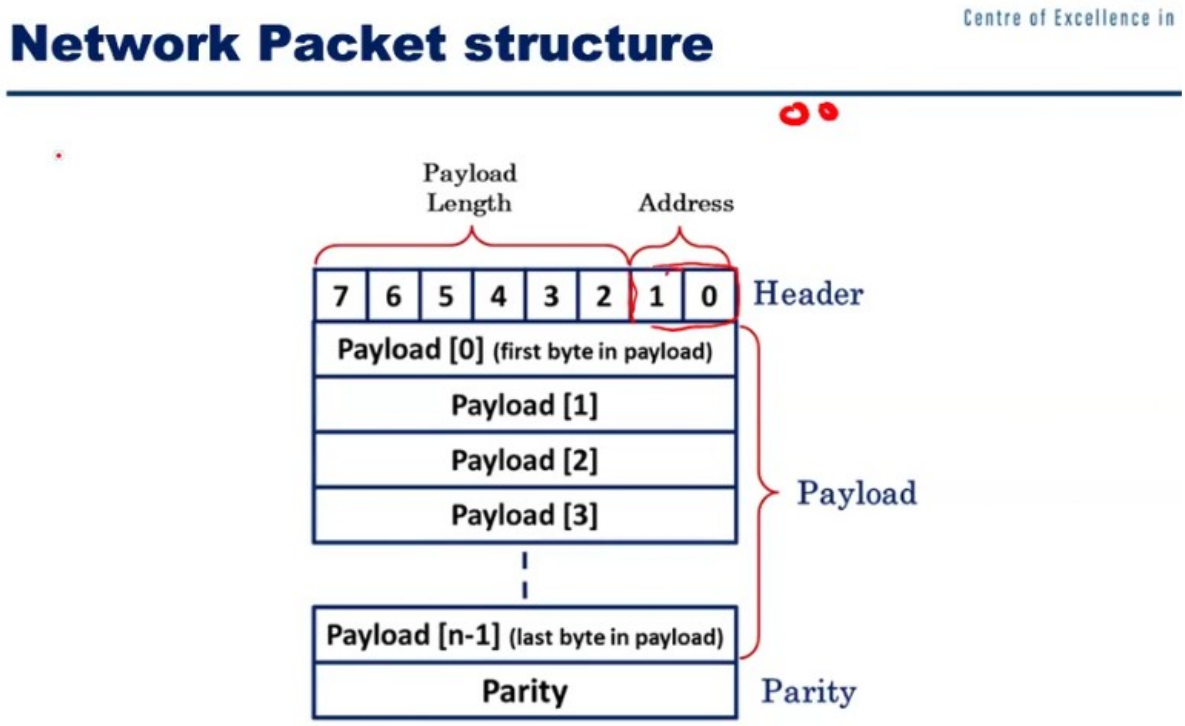


Various Blocks inside the Router:
1) Loadable Register: Data goes to register from source. If destination is accepting th data it will be sent to FIFO to store.
2) FIFOs : Acts as buffer between source and destination
3) Synchronizer : two imporant jobs i) to generate soft reset if read_enb is not made high after 30 clock clks of valid_out high.
ii) to make valid_out high once data is transfered from register to FIFO
4) FSM: Controls all the blocks as a controller


FSM
clock
resetn
pkt_valid
busy
parity_done
data_in
2
soft_reset_0
soft_reset_1
soft_reset_2
fifo_full
low_pkt_valid
fifo_empty_0
fifo_empty_1
fifo_empty_2
detect_add
ld_state
laf_state
full_state
write_enb_reg
rst_int_reg
lfd_state
Register
clock
resetn
pkt_valid
data_in
8
fifo_full
rst_int_reg
detect_add
ld_state
laf_state
full_state
lfd_state
parity_done
low_pkt_valid
err
8
dout
Synchronizer
detect_add
data_in
2
write_enb_reg
clock
resetn
vld_out_0
vld_out_1
vld_out_2
read_enb_0
read_enb_1
read_enb_2
3
write_enb
fifo_full
empty_0
empty_1
empty_2
soft_reset_0
soft_reset_1
soft_reset_2
full_0
full_1
full_2
FIFO_0
clock
resetn
write_enb[0]
soft_reset_0
read_enb_0
data_in
8
lfd_state
empty_0
8
dout_out_0
full_0
FIFO_1
clock
resetn
write_enb[1]
soft_reset_1
read_enb_1
data_in
8
lfd_state
empty_1
8
dout_out_1
full_1
FIFO_2
clock
resetn
write_enb[2]
soft_reset_2
read_enb_2
data_in
8
lfd_state
empty_2
8
dout_out_2
full_2

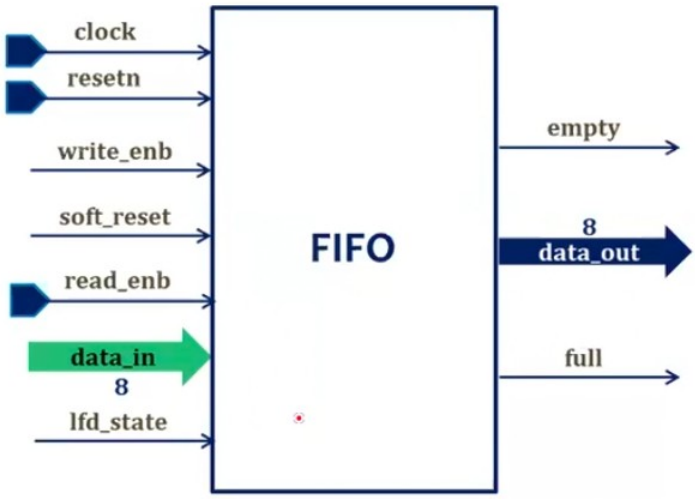
ROUTER PACKET STRUTURE:

- Packet Format: The Packet consists of 3 parts i.e Header, payload and parity.



Header: Destination Address,
Address
2-bit address,
00 --> D0
01 --> D1
10 --> D2
Payload length: 6-bits
Range of Payload 0 to 2^6
Parity: Calculate Packet Parity and internal parity (bit parity)

FIFO Block :



lfd_state: is used to identify the header byte. It will be high for header byte and low for remaining payload. lfd_state is generated by FSM.

soft_reset: Clears the FIFO contents, similar to reset
full: when FIFO is full, a full signal is sent to Synchronizer, when in return send a fifo_full signal to FSM and FSM will send the busy signal to source.
When busy is high no new data is accepted from source