BIG IP Prototyping

Prototyping and Partitioning

Guri Kamelo
Nadav Rosenblat
INTEL

June 16, Year (2015)

The Daniel hotel , Herzliya ,Israel

Agenda

Introduction

The issue - Big IP

The solution - Partitioning

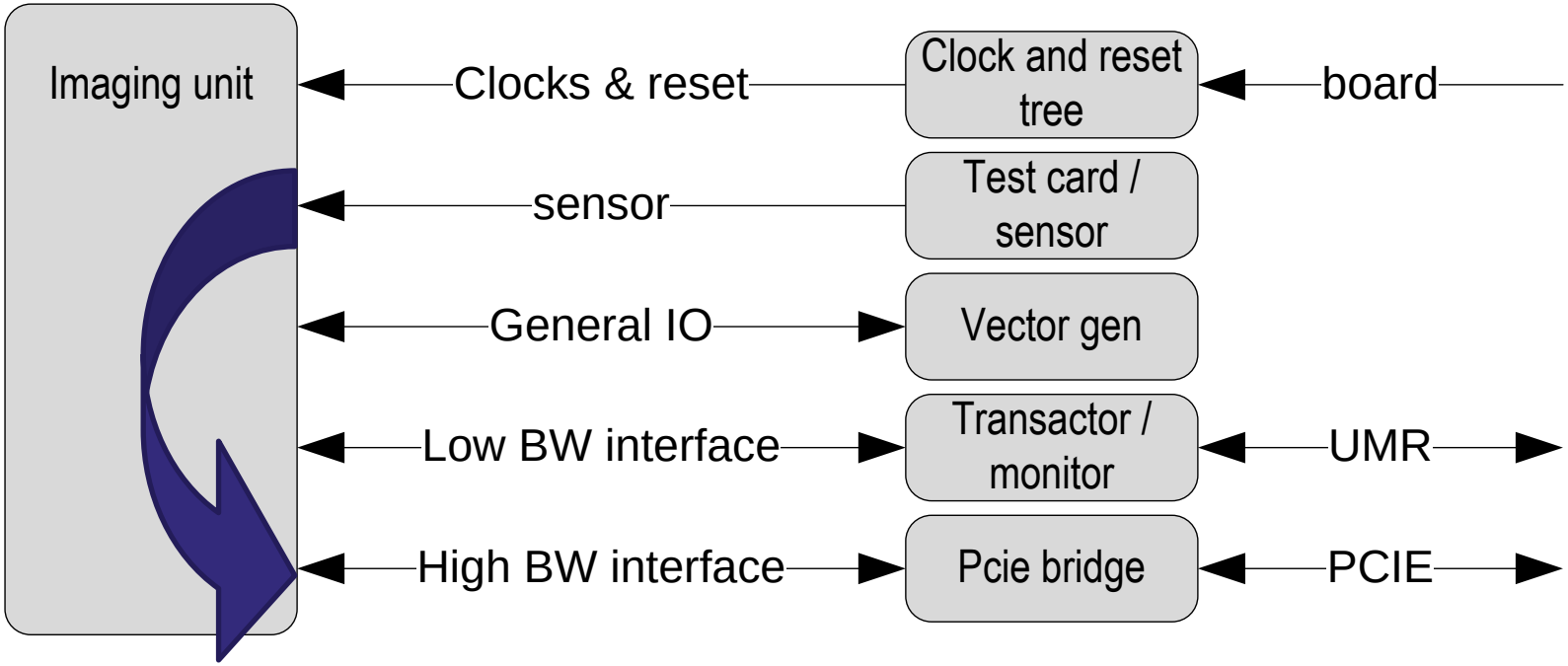
Who are we

- FPGA team under Validation group.
- Image processing IP .
- Located in INTEL HIFA .

Why prototyping

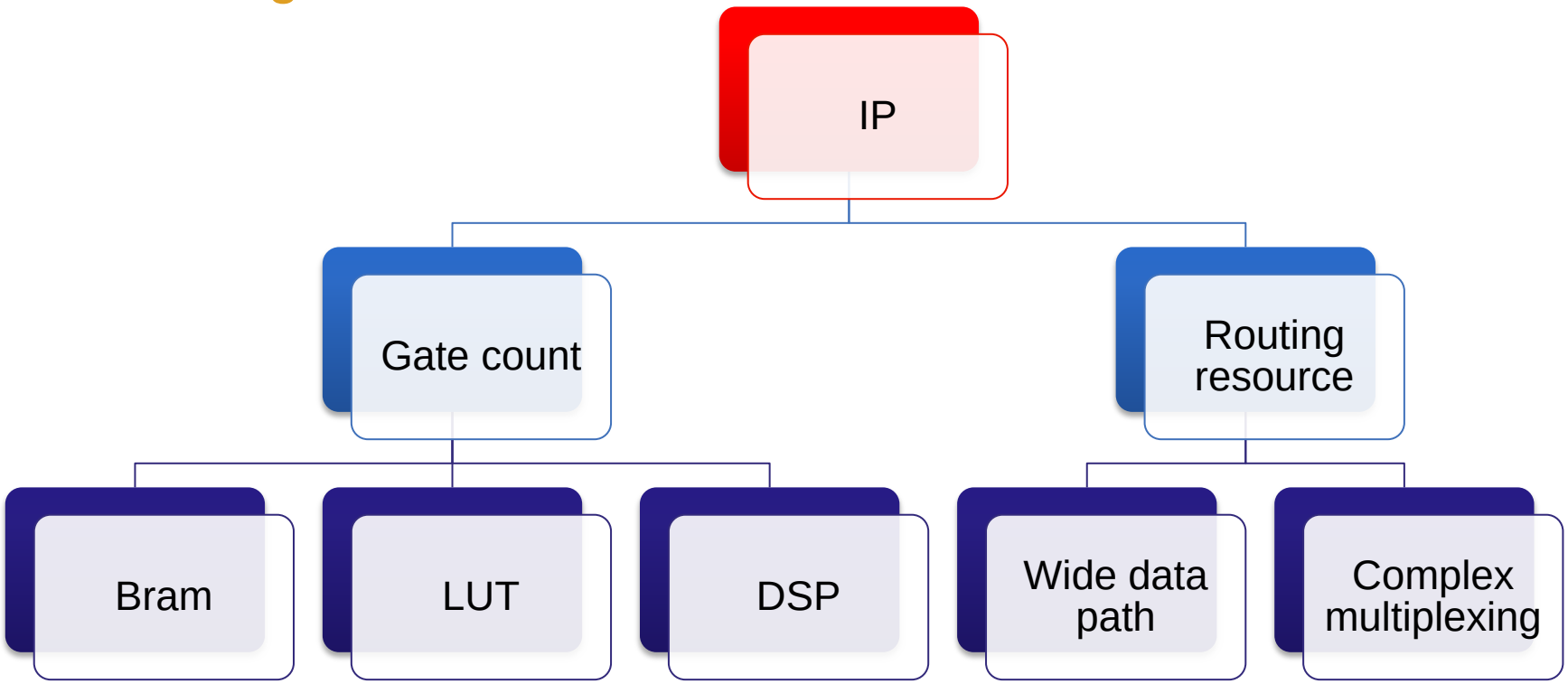
- Shift left
  - FW
  - Driver
  - Application
  - Low cost (v.s emulation)
- Find HW bugs

My IP
Design overview



Big IP

What is big ?



- Balance utilization
- Synthesis directive

- Memory modeling
- P&R strategy
- Location constraints

Partition

Avoid partition

- Limit IP clock .
- Complex flow .
- Cost

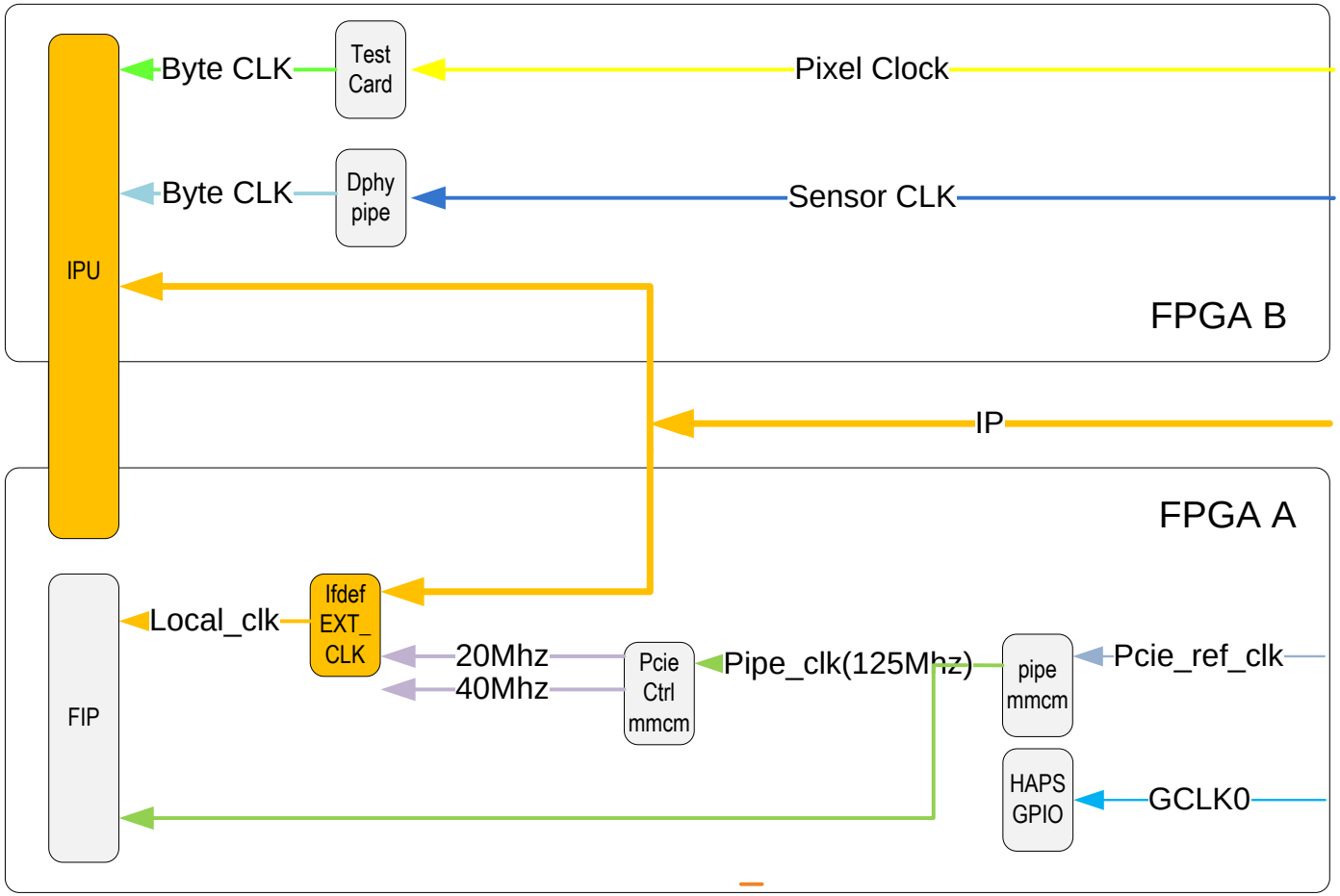
Partition

What should we worry about?

- Clock tree
- Reset tree
- Connection types and requirements (Direct, TDM etc.)
- IO limits

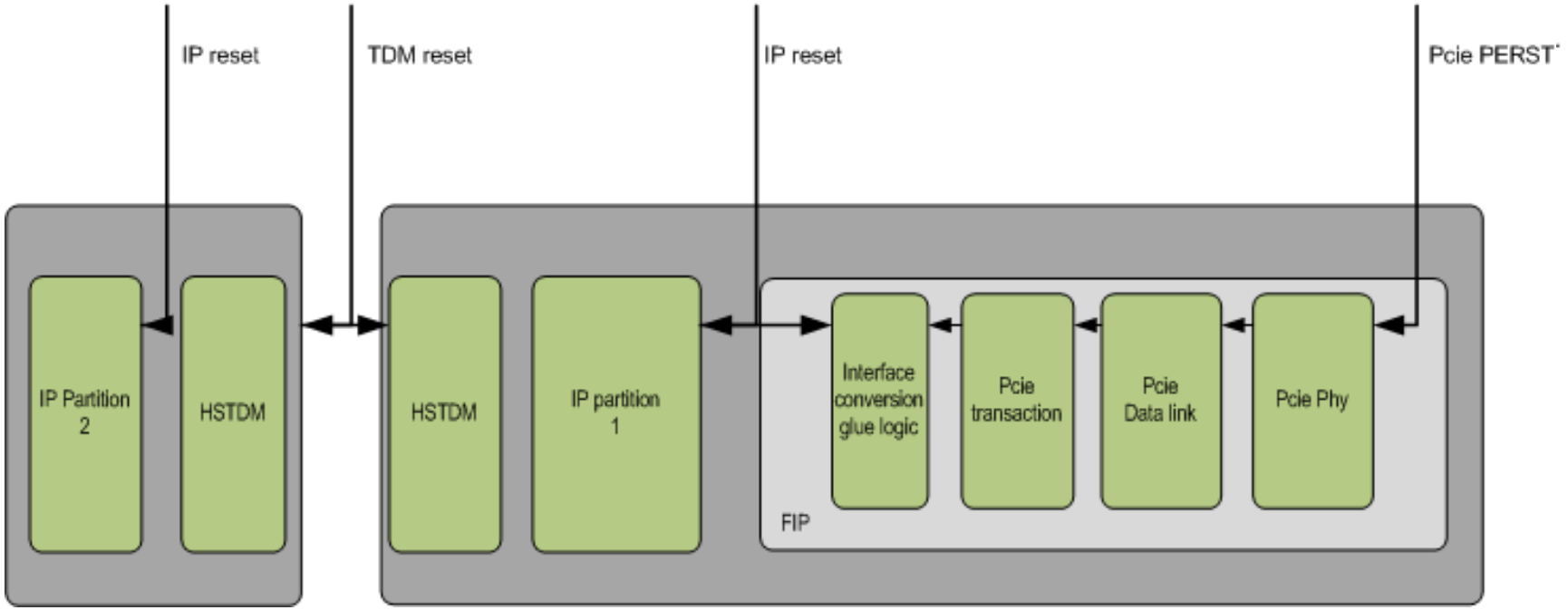
Partition

Clock tree - Generated clock stay inside



Partition

Reset tree

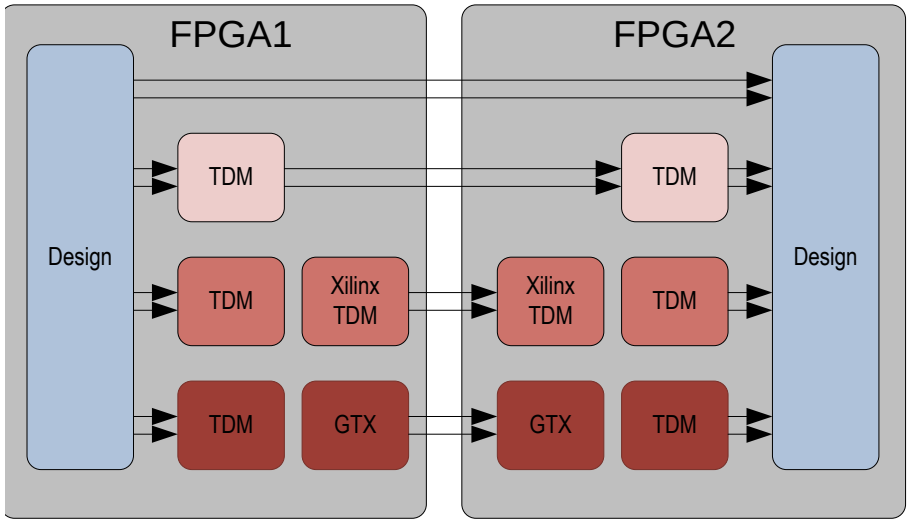


TDM reset → TDM train done → Ip reset → Pcie reset

Partition

Connection type

- Direct
- Single ended TDM
- High speed TDM
- GTX (not cycle accurate)



200Mhz

200Mhz

1.2Ghz(for 2 IO)

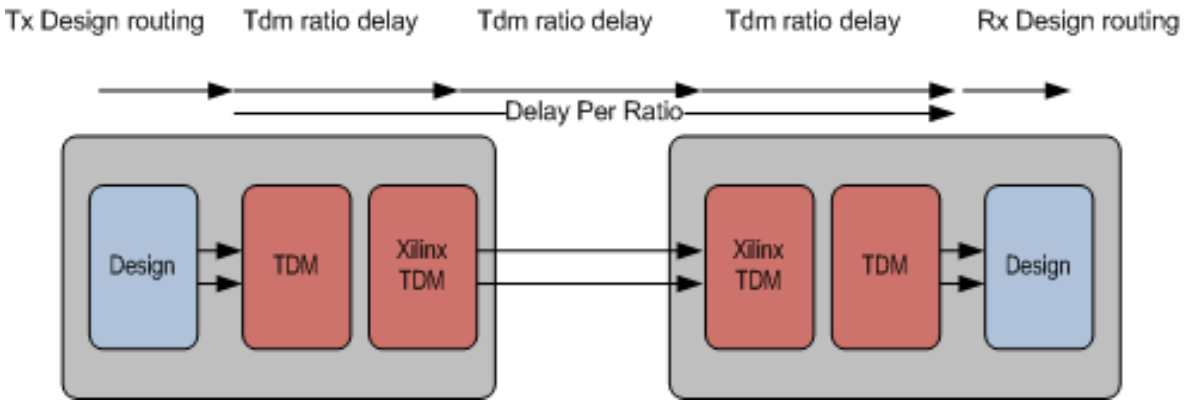
8Ghz

- Higher bandwidth
- Higher design complexity
- Harder to close timing

Partitioning

High speed TDM Latency (HAPS)

LVDS Frequency 1100Mhz	
TDM ratio [for 2 IO]	delay per Ratio[nSec]
8	34
16	50
24	58
32	68
40	76
48	82
56	90
64	98
72	104
80	112
88	118
96	126
104	134
112	140
120	148
128	156



$$Latency \sim (TDM\ ratio\ X\ LVDS\ clock\ frequncy) + \ route\ latency$$

Partitioning

LVDS TDM

$$DesignClock > Latency + FPGA\ internal\ routing$$

TDM ratio\ Design clock	1Mhz	5Mhz	6.3Mhz	10Mhz	20Mhz
8	966	166	116	66	16
16	950	150	100	50	0
24	942	142	92	42	-8
32	932	132	82	32	-18
40	924	124	74	24	-26
48	918	118	68	18	-32
56	910	110	60	10	-40
64	902	102	52	2	-48
72	896	96	46	-4	-54
80	888	88	38	-12	-62
88	882	82	32	-18	-68
96	874	74	24	-26	-76
104	866	66	16	-34	-84
112	860	60	10	-40	-90
120	852	52	2	-48	-98
128	844	44	-6	-56	-106

- Worst case Internal routing will determine max ratio
- Use FF to FF routing
- Using variable ratio to ease timing on specific signals

Partitioning

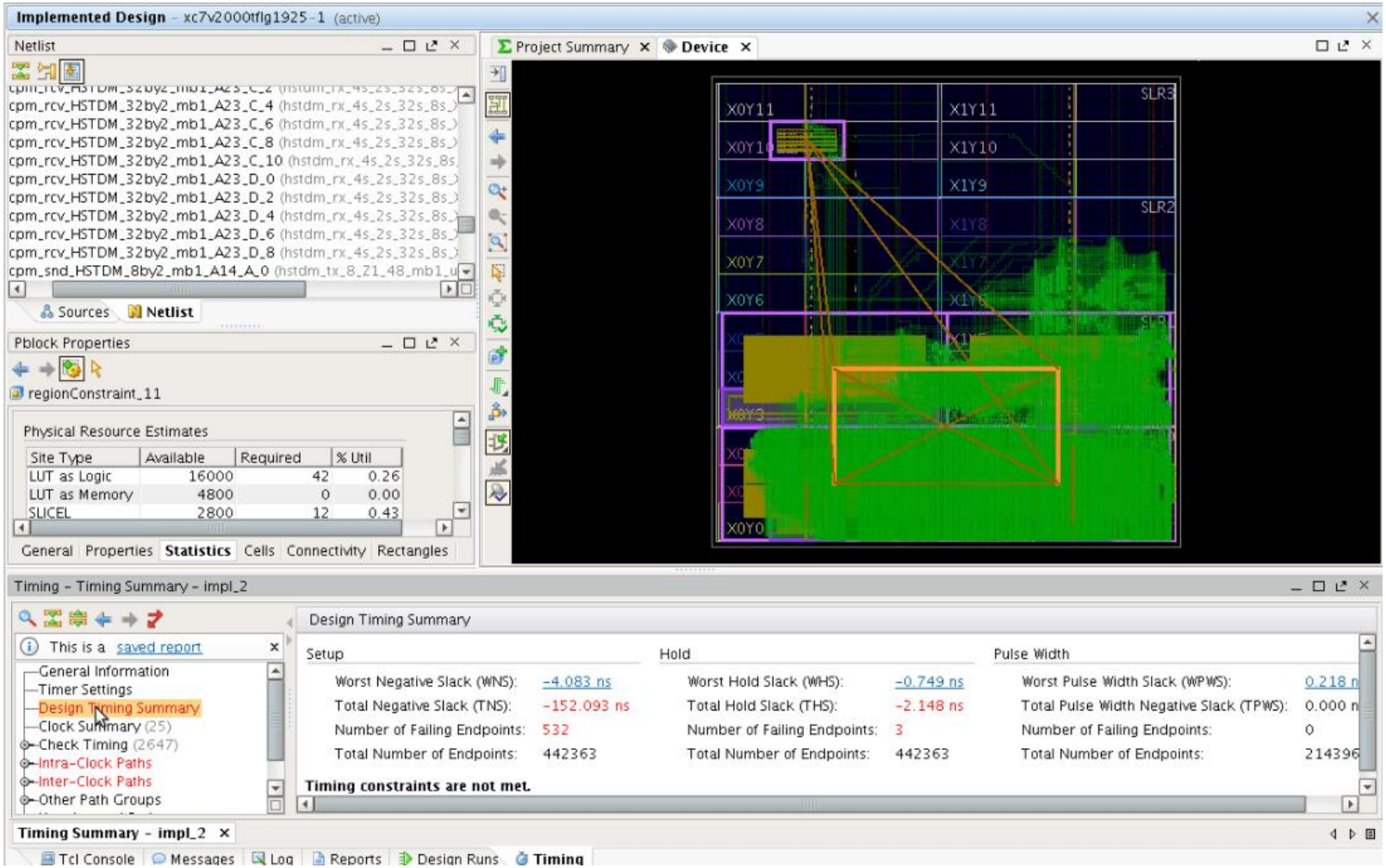
LVDS TDM

$$\frac{Num\ of\ Signals}{Num\ of\ IO\ per\ connector * TDM\ ratio} = Num\ of\ connector$$

TDM ratio 64:2				
Source	Destination	signals	Number of IO	Number of connectors
A	B	3128	98	3
A	C	4504	142	3
A	D	5526	174	4
A	A'	5526	174	4
A	B'	2286	72	2
Total A		20970	660	16

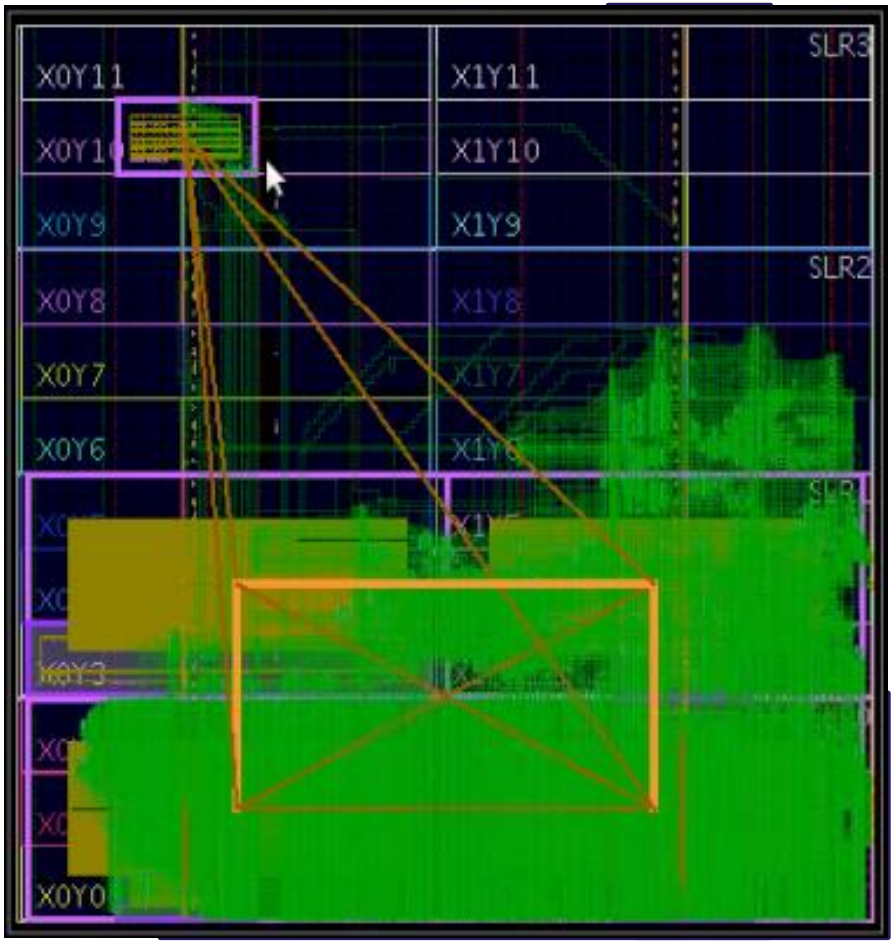
Partition

TDM P&R effect(example)



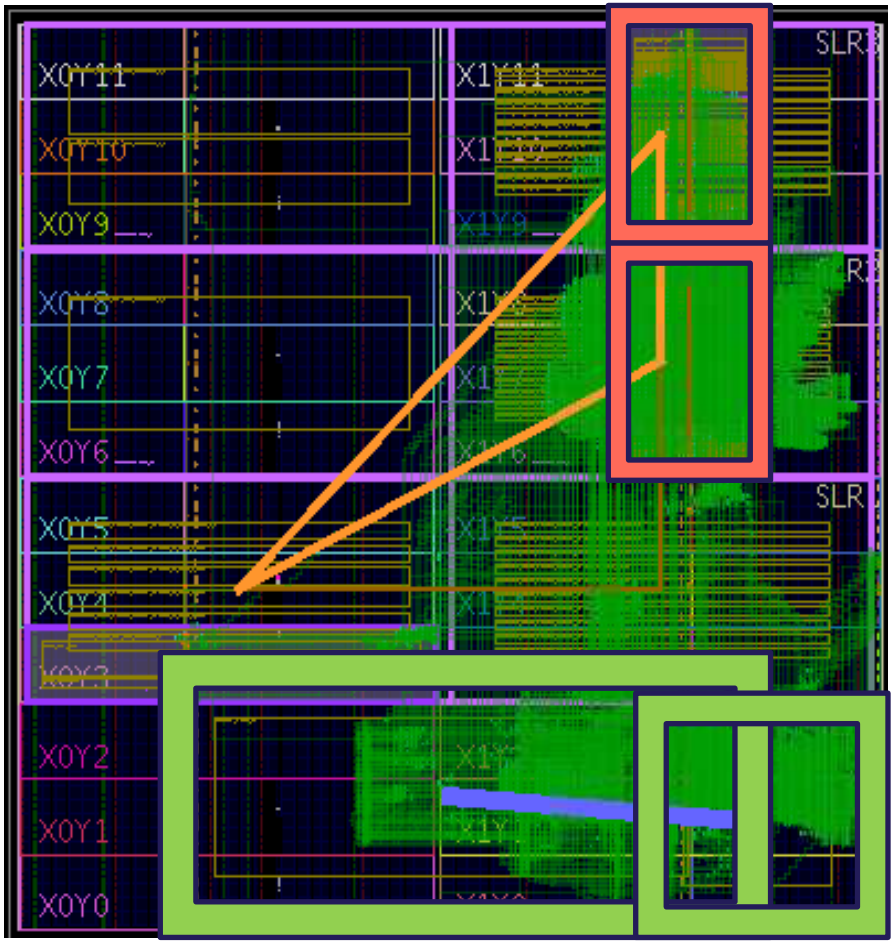
Partition

TDM P&R effect(example)



Partition

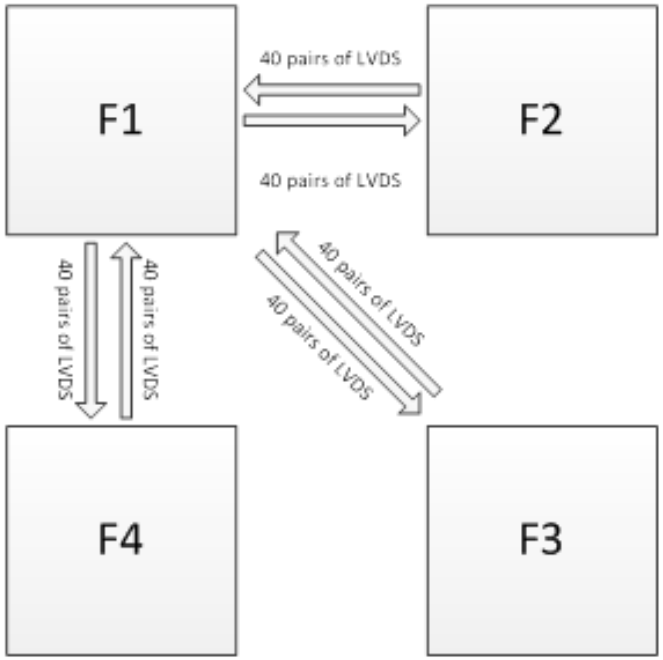
TDM P&R effect(example)



Partition

IO limited Partitioning

- Routed signals ~25k
- 40 LVDS pairs for FPGA interconnect

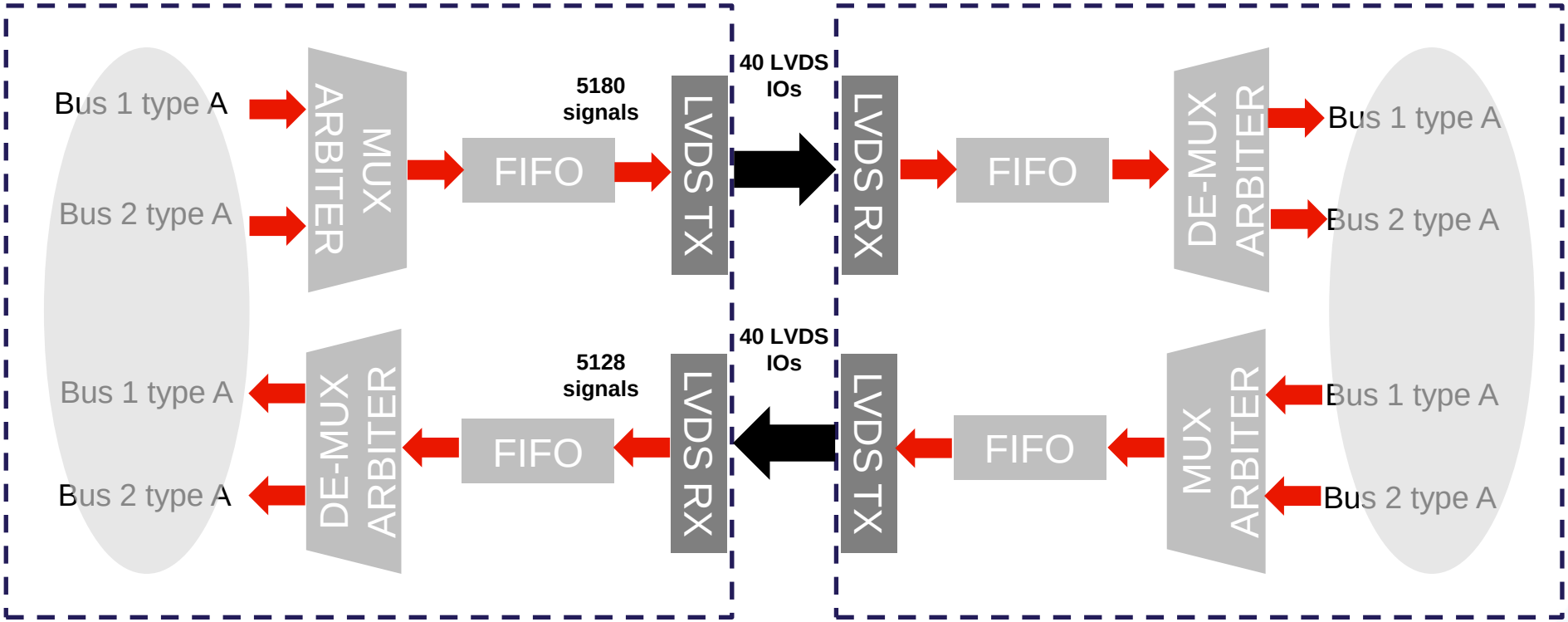


Star topology

Partition

IO limited partition : Example

- TDM ratio 128:2
- Design frequency 1M



Partition

What do you get on HAPS

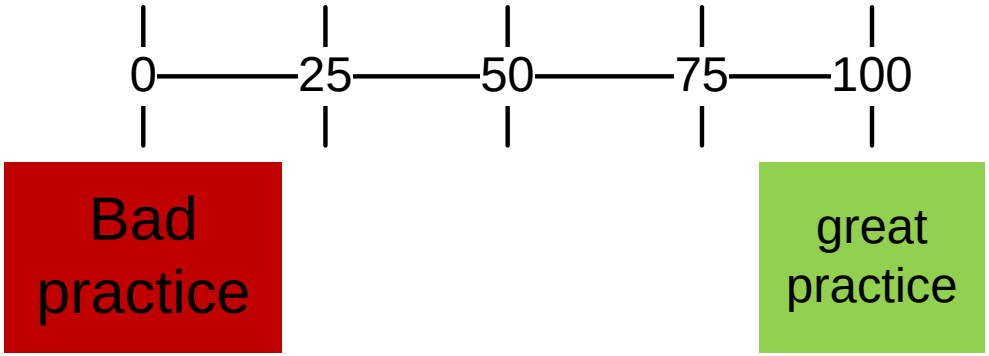
- Semi-auto partitioning – from my experience
- Full IP view before and after partitioning .
- Auto TDM insertion – TDM bist mode
- Backdoor access

Prototyping

Key principles

		Asic	prototype
✓	keep functional logic	f(x)	f(x)
✓	keep cycle accurate	1 clock	1 clock
✓	control IP interfaces	full control	partial
	IP bandwidth	based on SOC	partial
✓	Push button model	Not there yet	almost there

Supported on Protocompiler



Any question ?

Thank You