Introduzione all'utilizzo di FPGA in esperimenti di Fisica

Introduzione alle Tecniche di Trigger e Data Acquisition in Esperimenti di Fisica

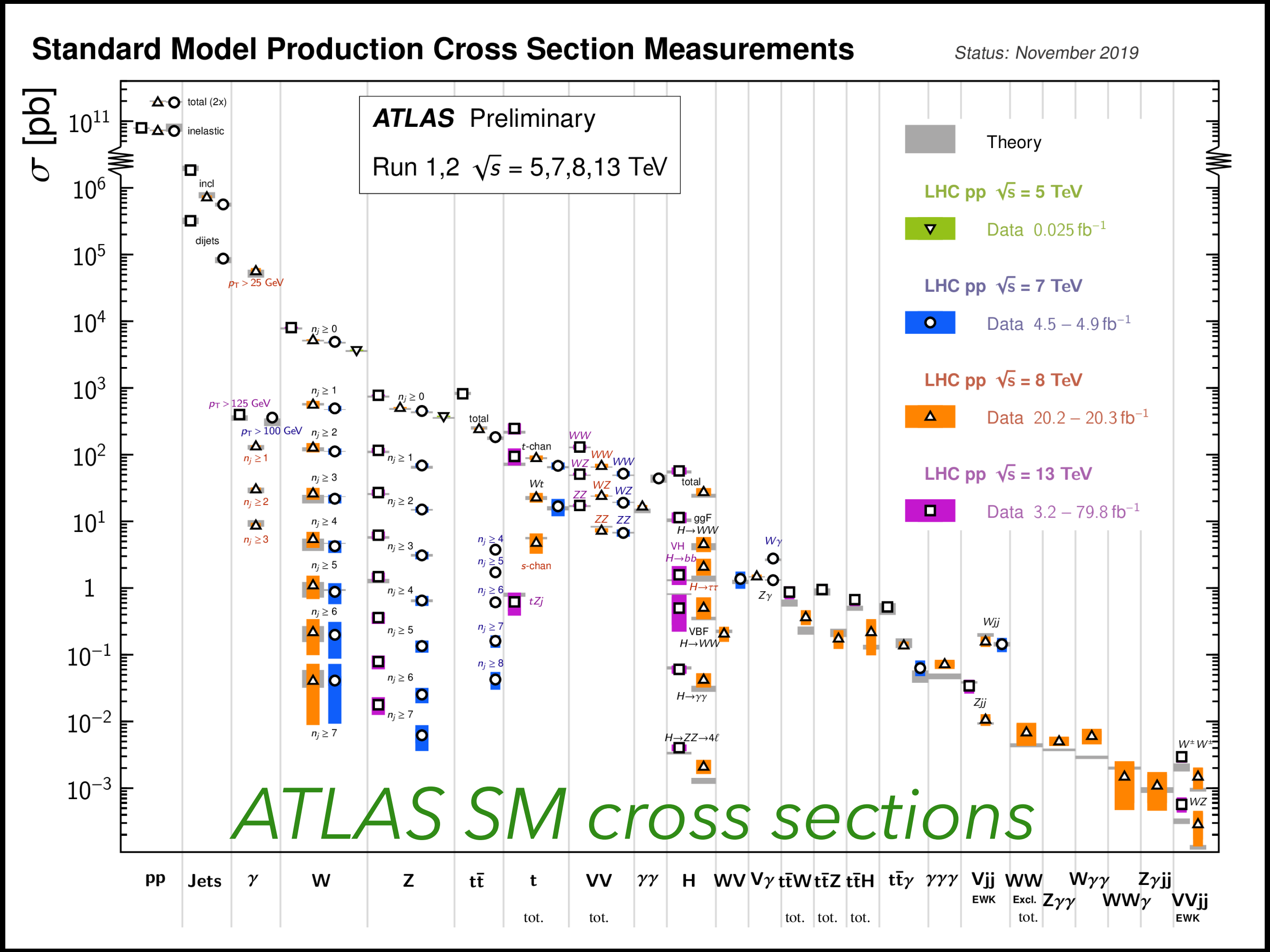
Napoli, 9 ottobre 2023

Riccardo Vari - INFN Roma

Trigger systems for High Energy Physics experiments

HEP event rate in a collider

- LHC nominal luminosity: 10^{34} cm^{-2} s^{-1}
- LHC proton-proton inelastic cross section: ~90 mb
- Cross section for most of the SM processes: ~1 fb



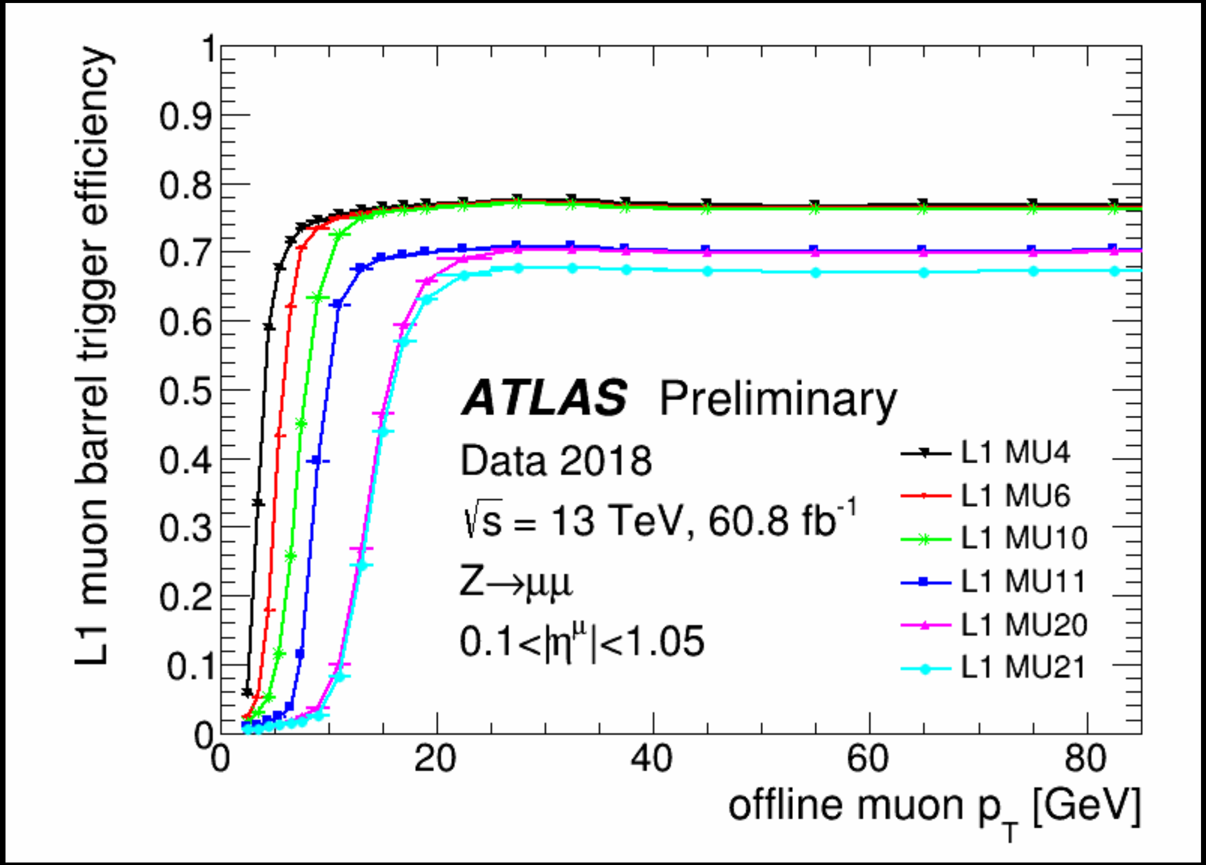
- LHC inelastic event rate at nominal luminosity: luminosity x cross section = 10^{34}x90x10^{-3}x10^{-24} = 900 million/s
- SM events rate: 10^{34}x1x10^{-15}x10^{-24} = 10^{-5} = 1 every 2.8 hours

HEP data rate

- LHC experiments:
  - Data size for a typical event up to ~MB per Bunch Crossing (ATLAS, CMS)
  - Total data rate before selection: ~TB/s
- Two possibilities:
  - Record all the produced event (if you can, then select and study them offline)
  - Record only the interesting events using a real-time system
- A trigger system allows to distinguish interesting signals from background, and to select rare events
- Trigger-able events in HEP: electrons, gamma, muons, jets, …
  - Trigger conditions: geometrical, energy or p_T cut, topological selection, …
- Data flow: Beam Crossing ➡ Detector ➡ Trigger ➡ Data Acquisition ➡ Storage

HEP Trigger Parameters

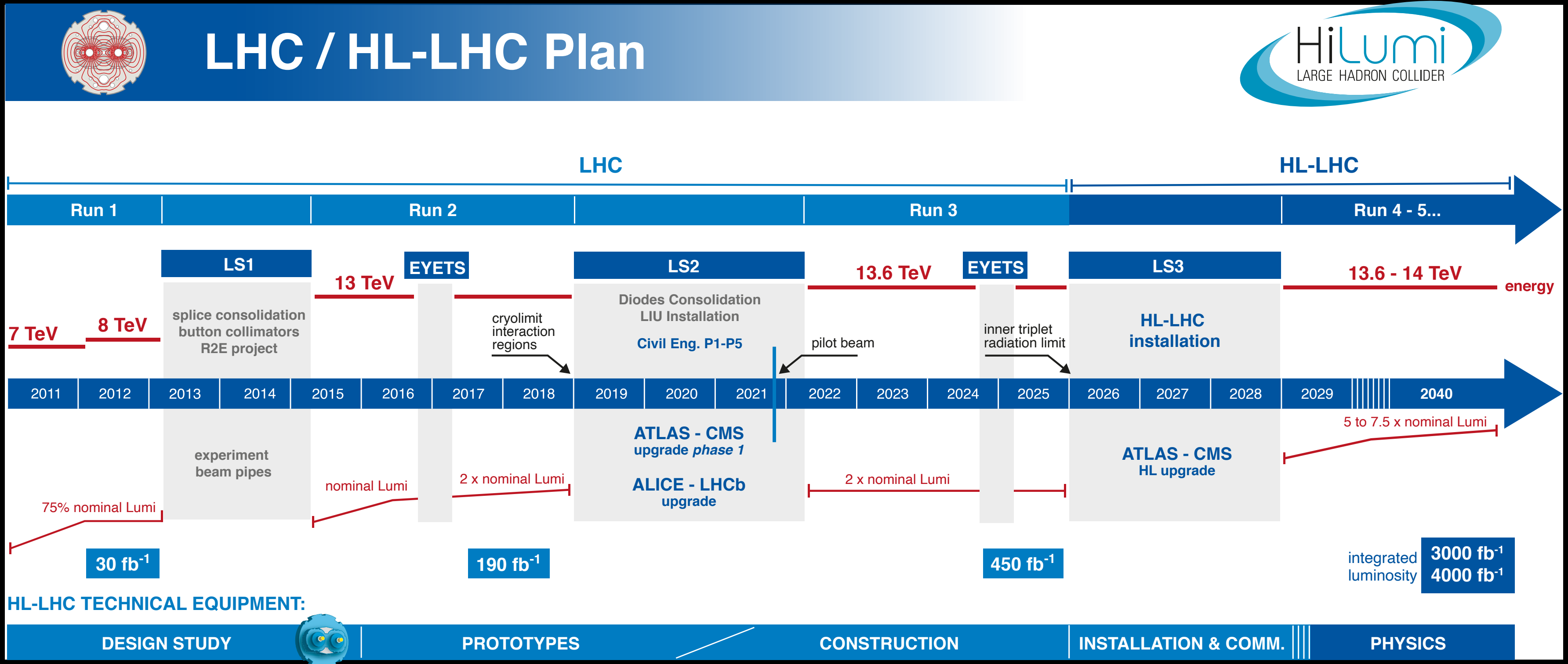
- Collision frequency: 40 MHz for LHC (multiple events originated at each collision)
- Event size: ~1 MB for ATLAS or CMS
- Data acquisition write bandwidth: ~1 kHz for ATLAS/CMS
- Latency: time needed to take the hardware trigger decision (few μs for ATLAS/CMS)
- Efficiency: % of accepted interesting events (turn-on curve)



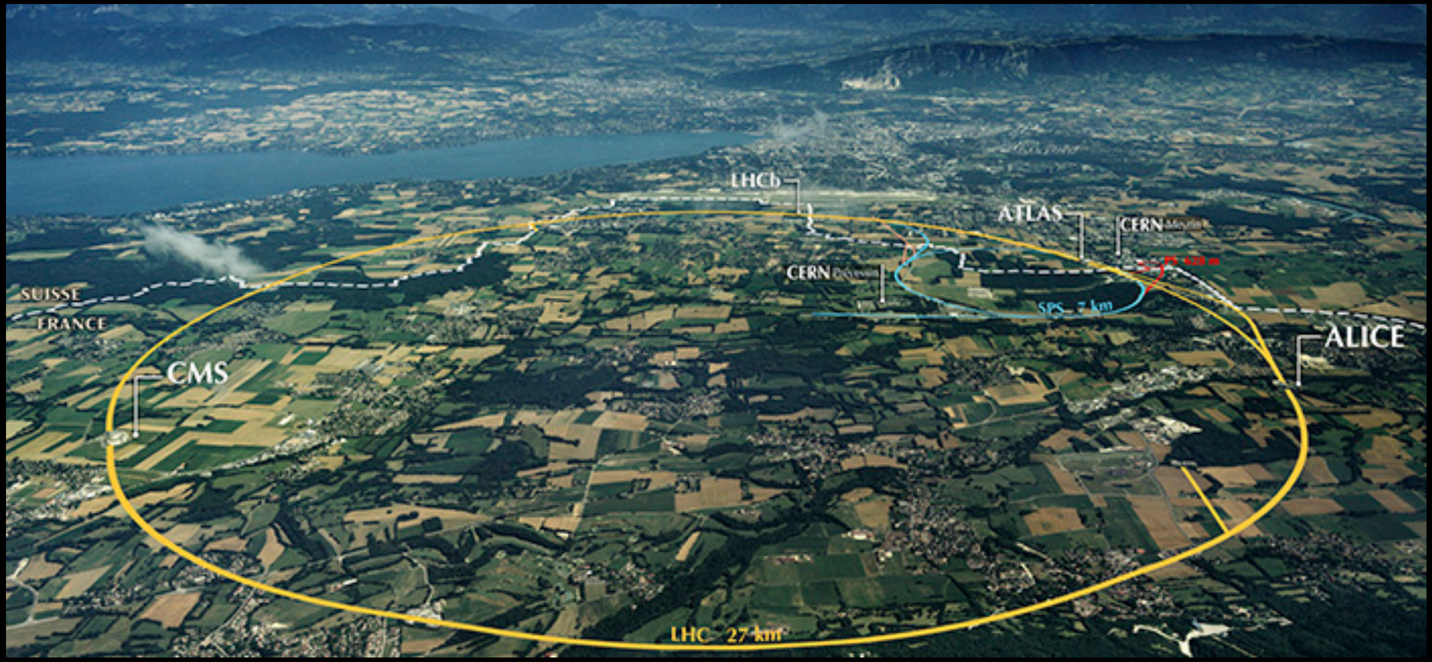
- Rejection: % of rejected not-interesting events
- Dead-time: % of time the trigger system cannot process events (for failures, out-of-memory states, …). Dead-time causes inefficiency
- Synchronisation: keeping track of the event identifier during the full triggering process. Loss of sync causes inefficiency

The Large Hadron Collider (LHC) at CERN

https://hilumilhc.web.cern.ch/content/hl-lhc-project



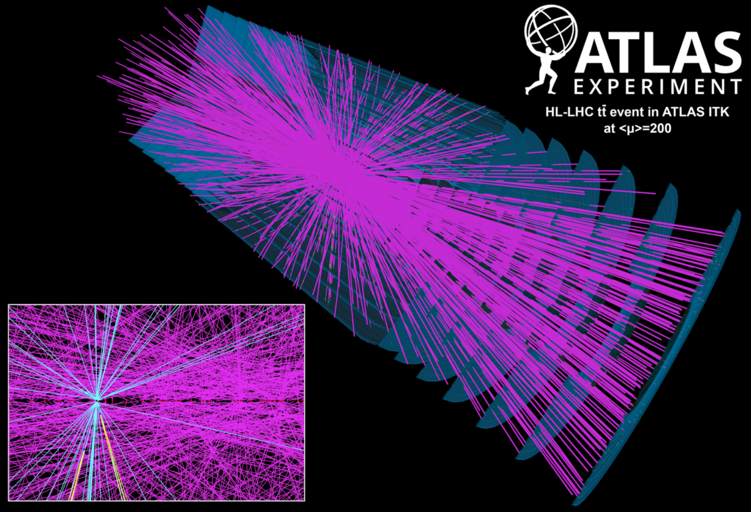
- 27 km p-p circular accelerator
- 2835 + 2835 bunches in the LHC ring
- 10^{11} protons/bunch
- 40 MHz collision rate
- Most of the interesting events are rare
- Readout channels: ~90 millions (ATLAS and CMS), ~1 million (LHCb and ALICE)



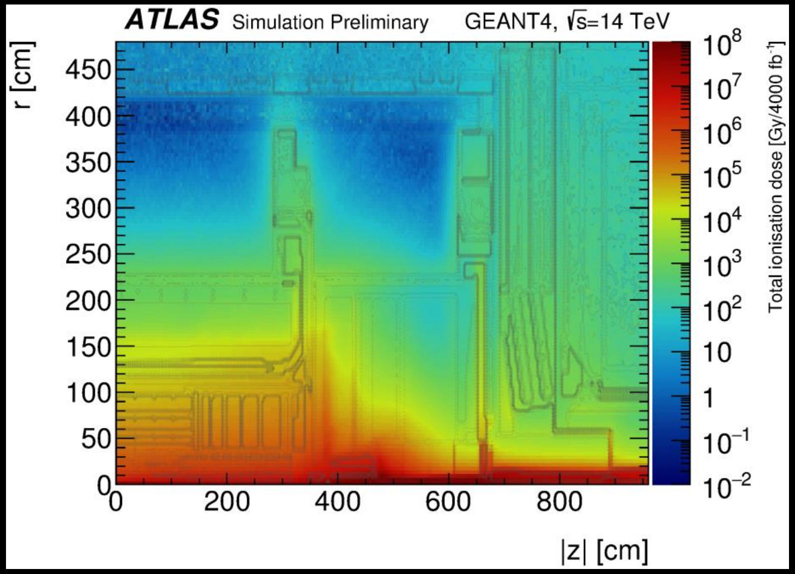
Increasing luminosity impact on LHC experiments

- High luminosity is needed to achieve physics goals
- The experiment has to stand the Run4 foreseen peak luminosity of 7.5 x 10^{34} cm^{-2} s^{-1}
  - high pile-up ~200 collisions/crossing for Run4
  - high radiation levels, up to ~10^{16} neq/cm^2, 10 MGy
- Requirements:
  - maintain good physics performances in the challenging environment
  - keep acceptable trigger rate for low p_T threshold
  - mitigate pile-up up to high η

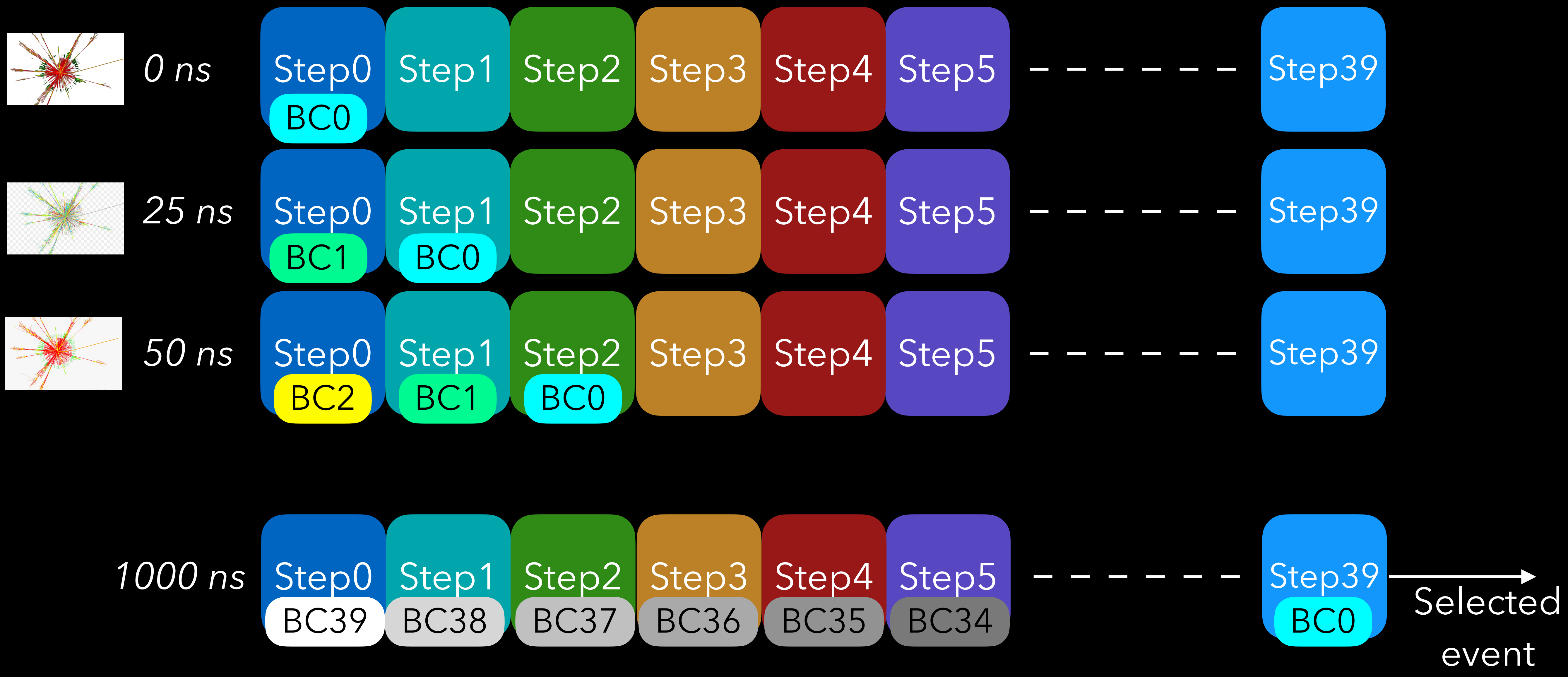
ATLAS pileup for Run4



ATLAS TID radiation levels for Run4



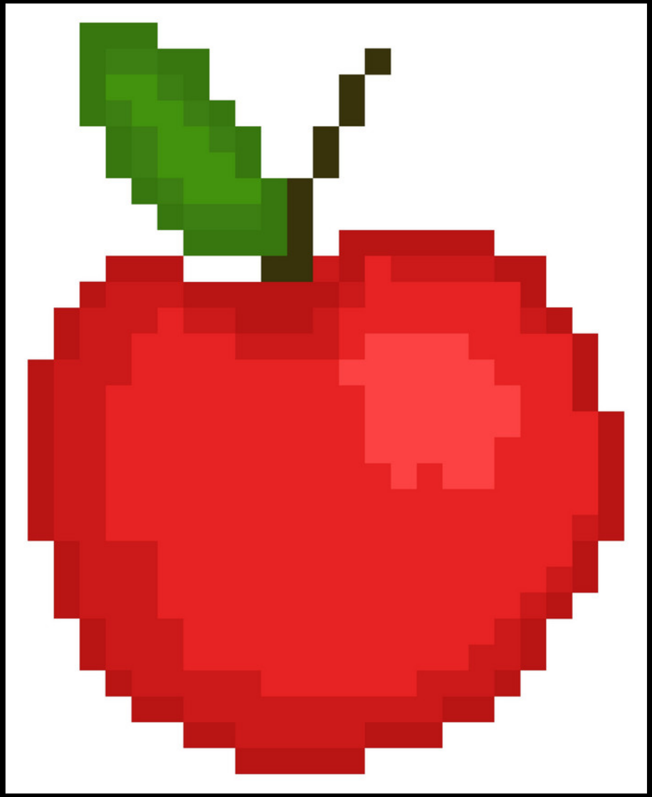
Pipeline processing in HEP trigger



- One collision every Bunch Crossing (25 ns)
- BCn represents the data of the nth Bunch Crossing
- Each step is a different operation made on the BC data to perform the trigger
- Selection process in this example takes 40 steps
- In this example trigger processing rate 40 MHz (1 μs trigger latency)

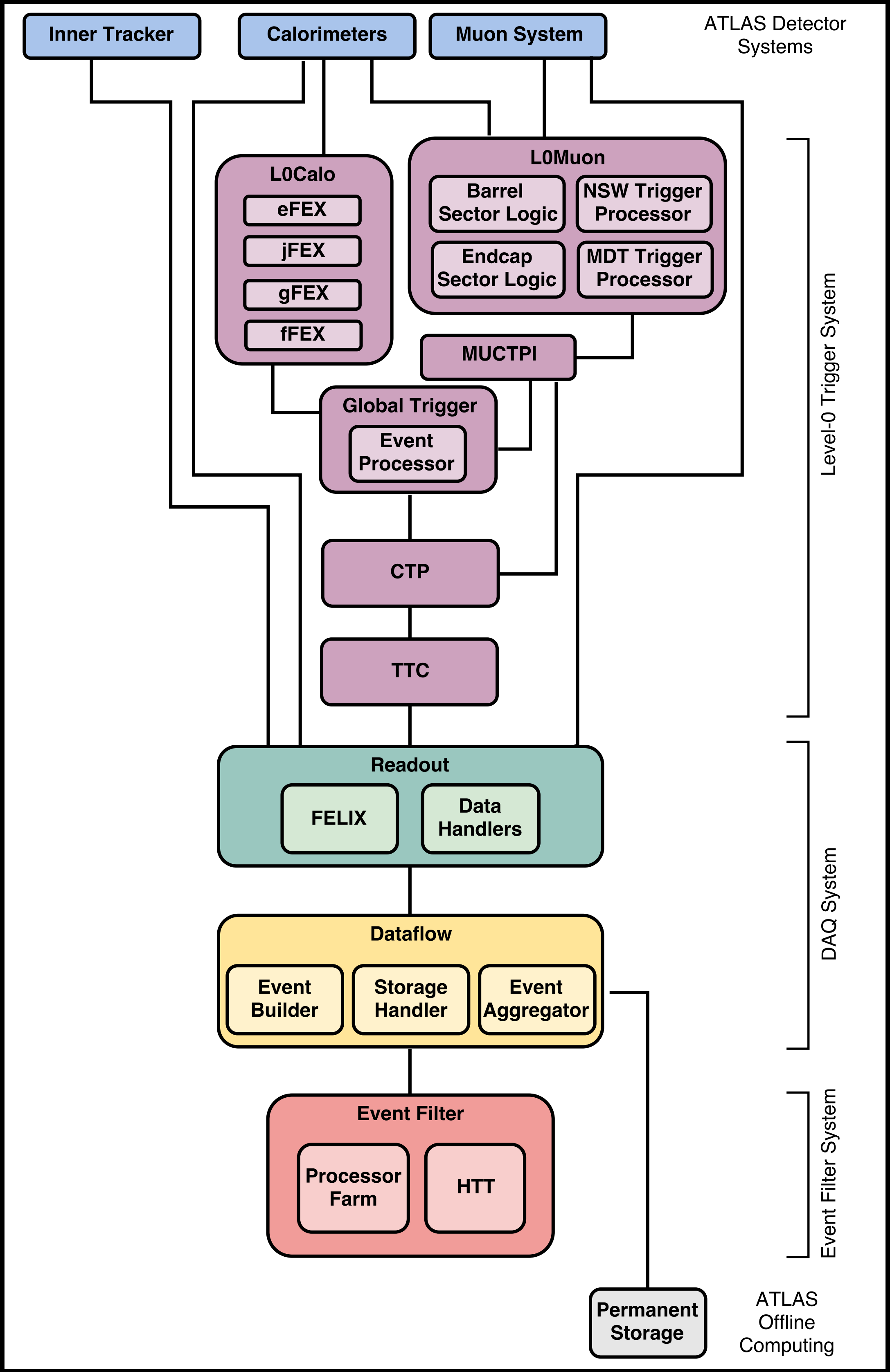
ATLAS Run3 trigger system

- The Level-1 trigger:
  - hardware system, custom-made electronics
  - Low granularity: works on a subset of information from the calorimeter and muon detectors
  - Low latency: decision ~2.5μs after the collision
  - Pipeline storage: the selected event is retrieved from pipelined buffers
  - Maximum output event rate: 100 kHz
- The High Level Trigger (HLT):
  - Software based trigger, farm of CPUs
  - Refines the selection of the Level-1 trigger
  - Detailed analysis of the detectors full granular information
  - Maximum output event rate: 1 kHz
  - The reconstructed events are passed to the data storage system for offline analysis



ATLAS Run4 Trigger and DAQ

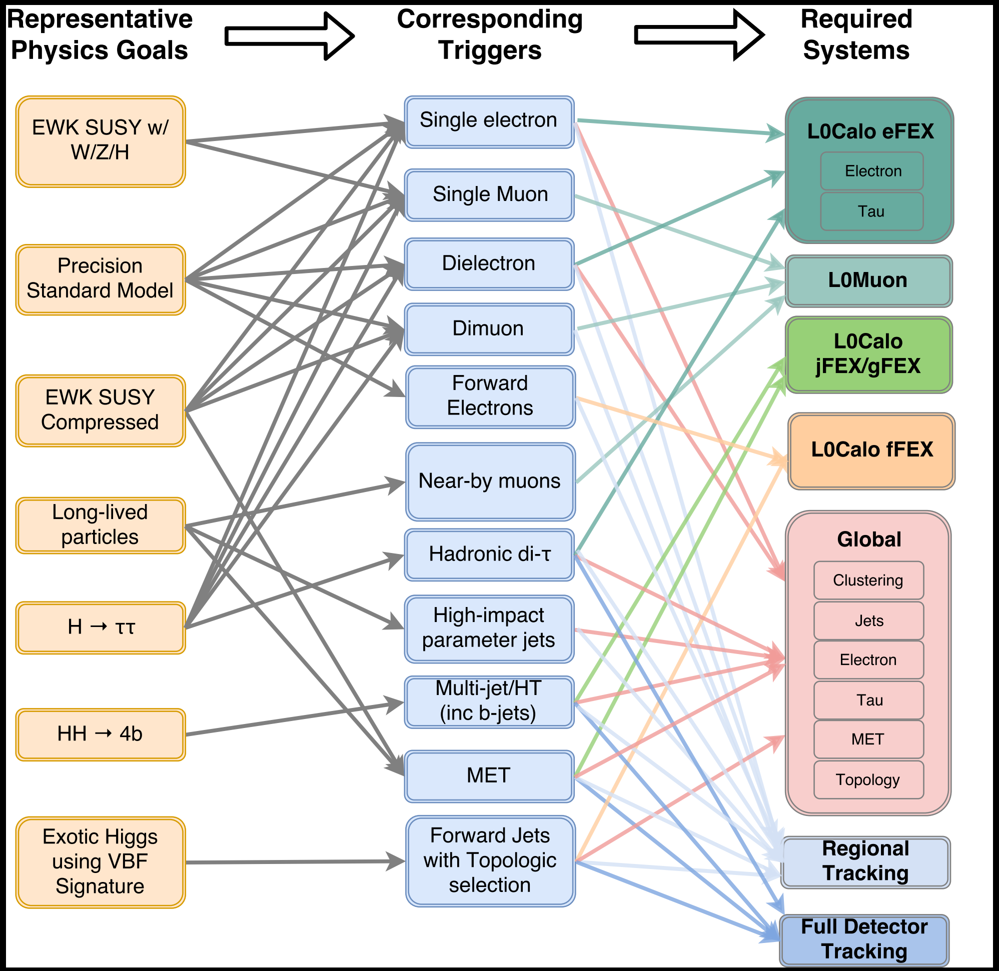
- Calorimeters and muons front-end full granularity readout at 40 MHz
- Level-0 hardware trigger with an output rate of 1 MHz, Level-0 readout latency is 10 μs
- Global event processor replaces the Run3 L1Topo and integrates topological functions with additional selection algorithms using information from muons and calorimeters
- Detectors readout based on FELIX system
- FPGA-based boards off-detector, on-detector where possible
- Possible hardware accelerator system for tracking at the Event Filter
- Goal of better e, γ, τ, jet identification and measurement, at hardware and software trigger levels and offline
- Event Filter output rate up to 10 kHz



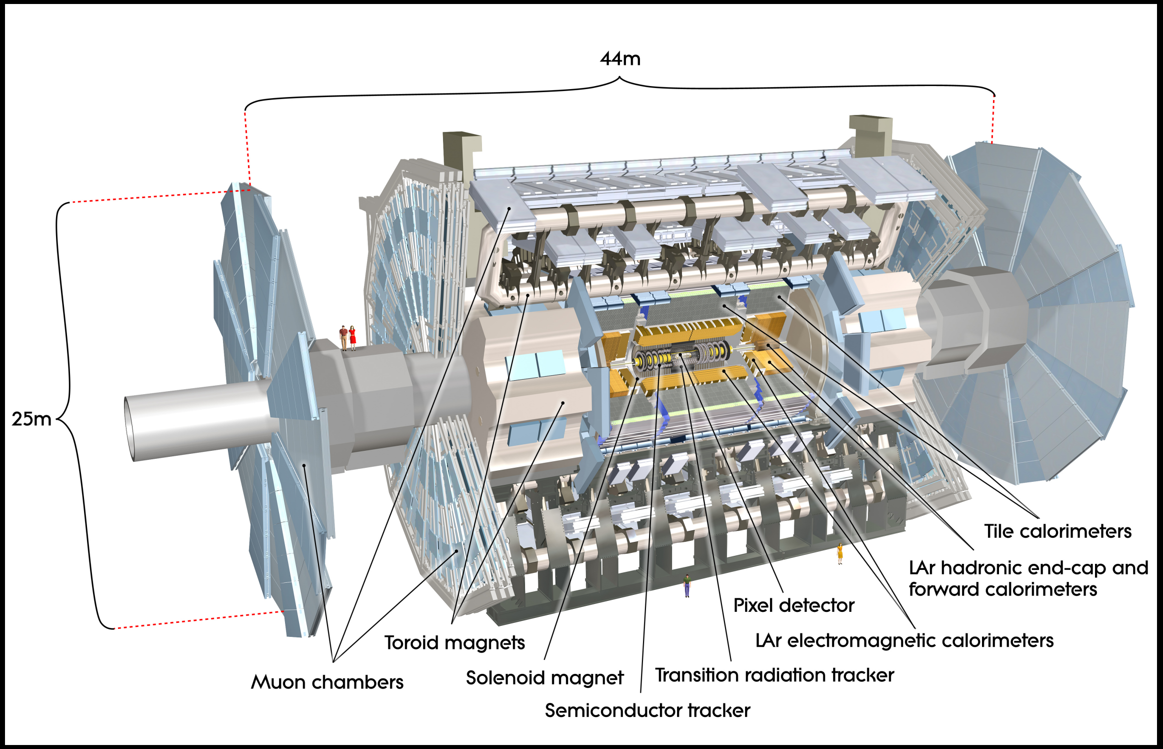
ATLAS Run4 TDAQ parameters

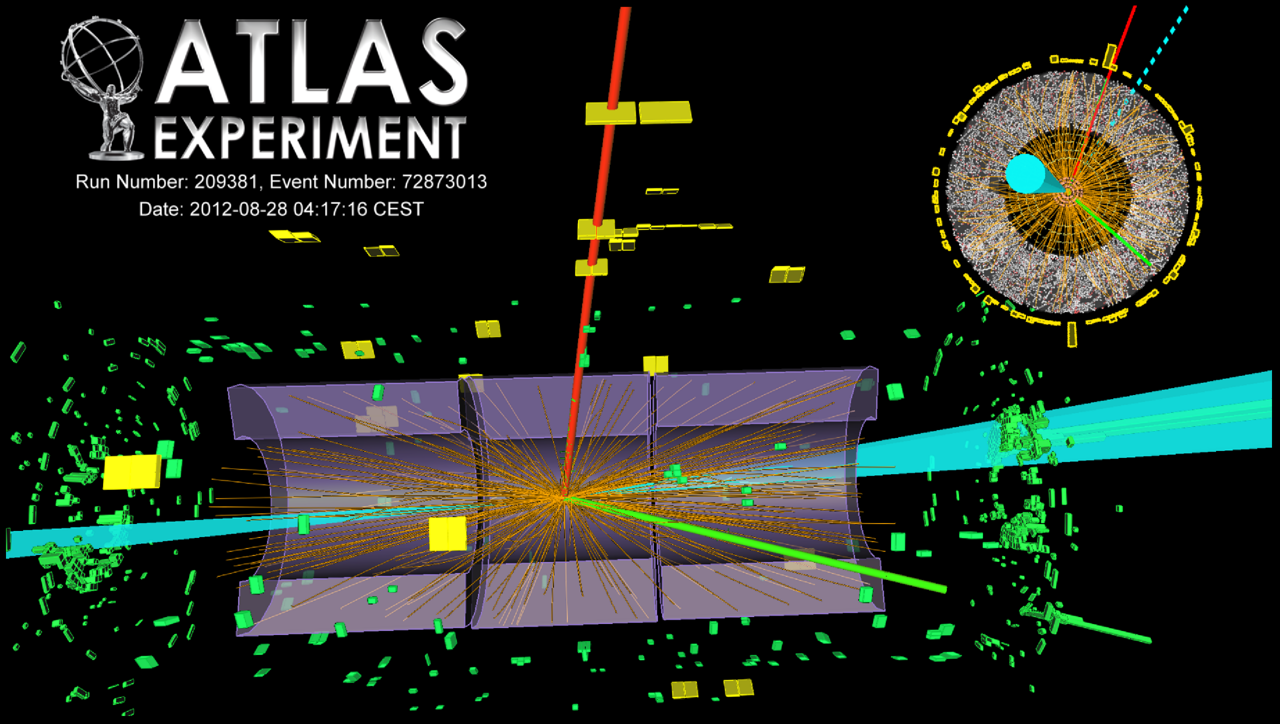
- About 100 million detector channels
- L0 output rate: 1 MHz
- L0 latency: 10 μs
- HLT output rate: 10 kHz
- Expected event size ~4.5 MB
- HLT out ~40 GB/s
- L0 hardware: ATCA FPGA-based boards, each board equipped with hundreds of ~10 Gb/s links, SoC FPGA for slow control and monitoring
- Readout hardware: new FELIX board with double PCIe bandwidth (PCIe gen4), 25 Gb/s links
- HLT hardware: CPU-based farm with thousands of servers, each one equipped with 25 Gb/s links
    - hardware accelerators (heterogeneous CPU/GPU/FPGA commodity processors) to reduce the number of servers and to reduce power consumption

Flow of the physics goals through the hardware systems

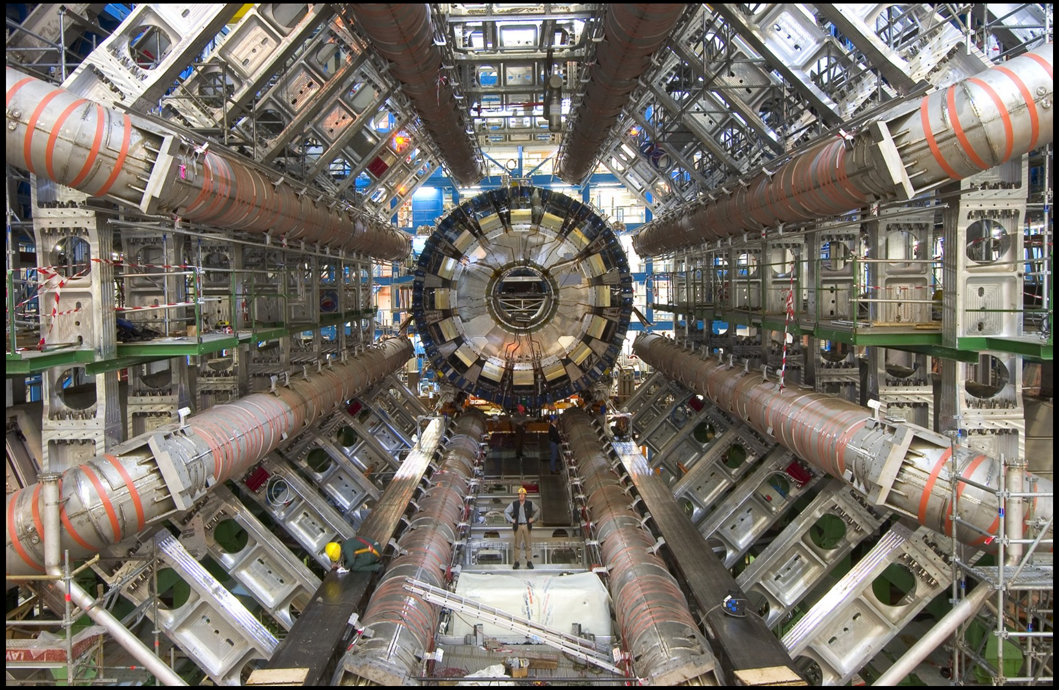


FELIX readout prototype board


44m
25m
Tile calorimeters
LAr hadronic end-cap and forward calorimeters
Pixel detector
LAr electromagnetic calorimeters
Toroid magnets
Muon chambers
Solenoid magnet
Transition radiation tracker
Semiconductor tracker


ATLAS EXPERIMENT
Run Number: 209381, Event Number: 72873013
Date: 2012-08-28 04:17:16 CEST

The First Level Muon Trigger in the Barrel region of the ATLAS experiment



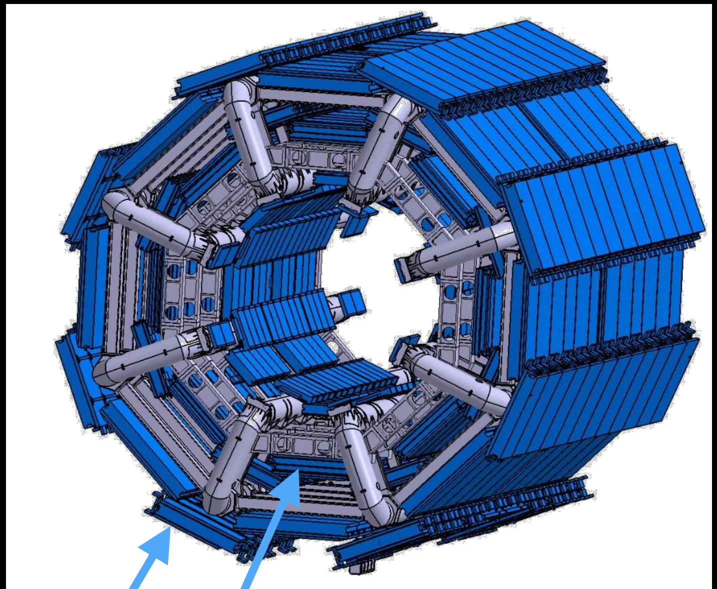
[ATLAS public web page]

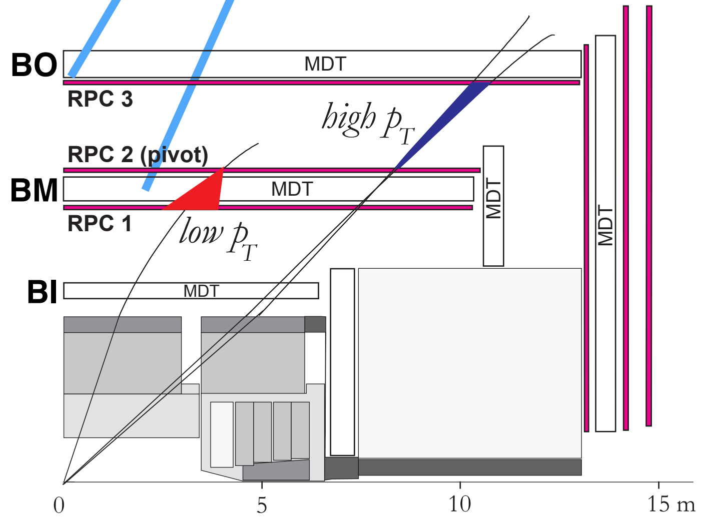
[ATLAS public results web page]

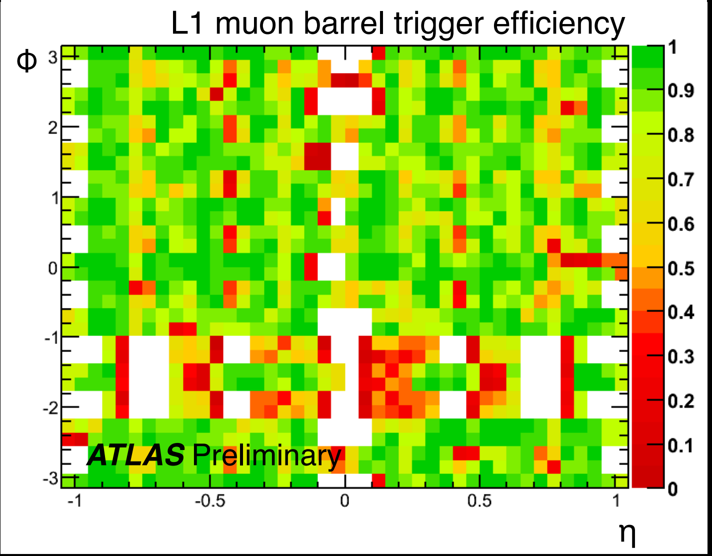
[ATLAS Phase-II TDAQ Upgrade TDR]

Run3 Level-1 muon trigger in the barrel region

- Muon detector: Resistive Plate Chamber (RPC). A hit signal is generated when a muon crosses the detector gas gap
- Every 25 ns the trigger system looks for muon hit coincidences of three concentric RPC stations
- Low-p_T trigger (< 10 GeV) makes use of the two BM stations
- High-p_T trigger (> 10 GeV) requires an additional confirmation on the BO station
- RPC detector coverage is limited to 74% because of the ATLAS mechanical support structures (toroid ribs in small sectors of BM, toroid feet in the lower part of the barrel)

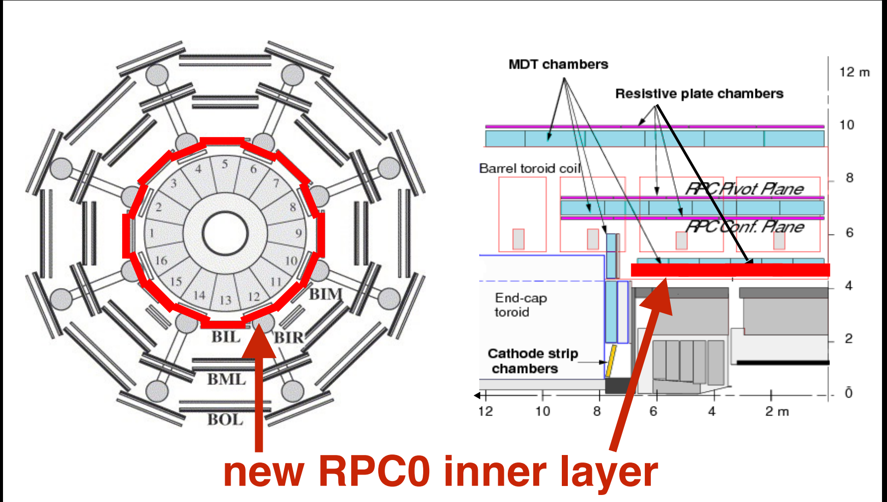


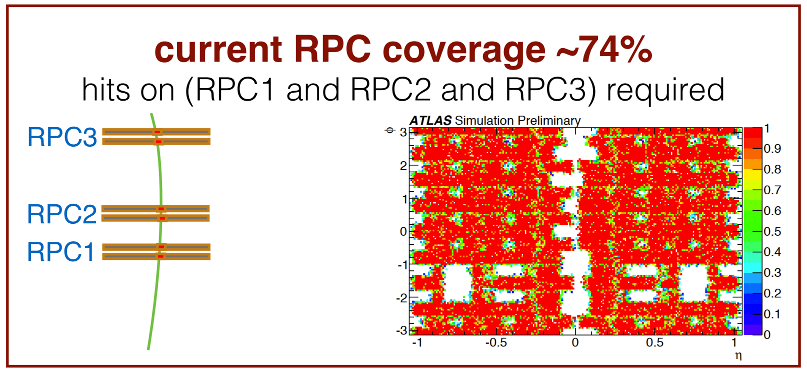


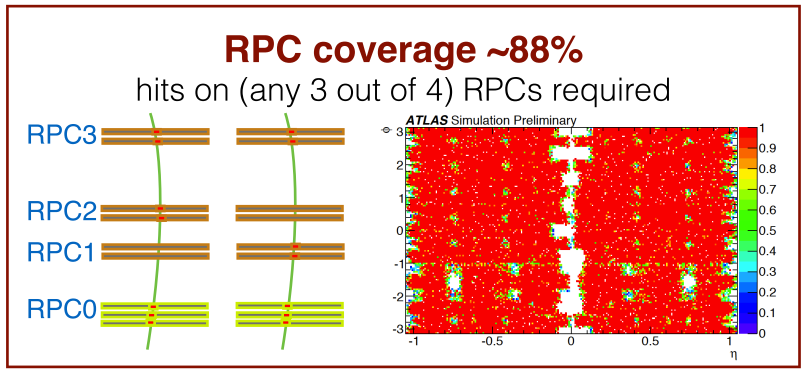


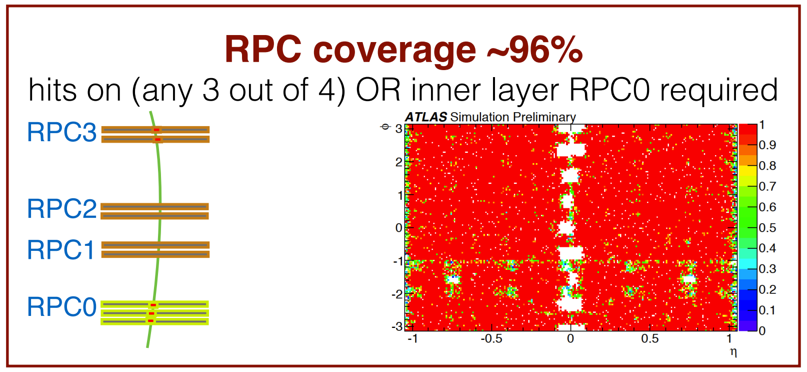
Run4 Level-0 Muon Trigger in the Barrel Region

- ATLAS Run4 upgrade: 2026-2028
- LHC Run-4: 2029-…
- Install additional RPC layer in the Barrel inner region to increase the current detector coverage
- Perform the trigger coincidence logic in 9 RPC layers
- Trigger coverage can increase from ~74% to ~96%



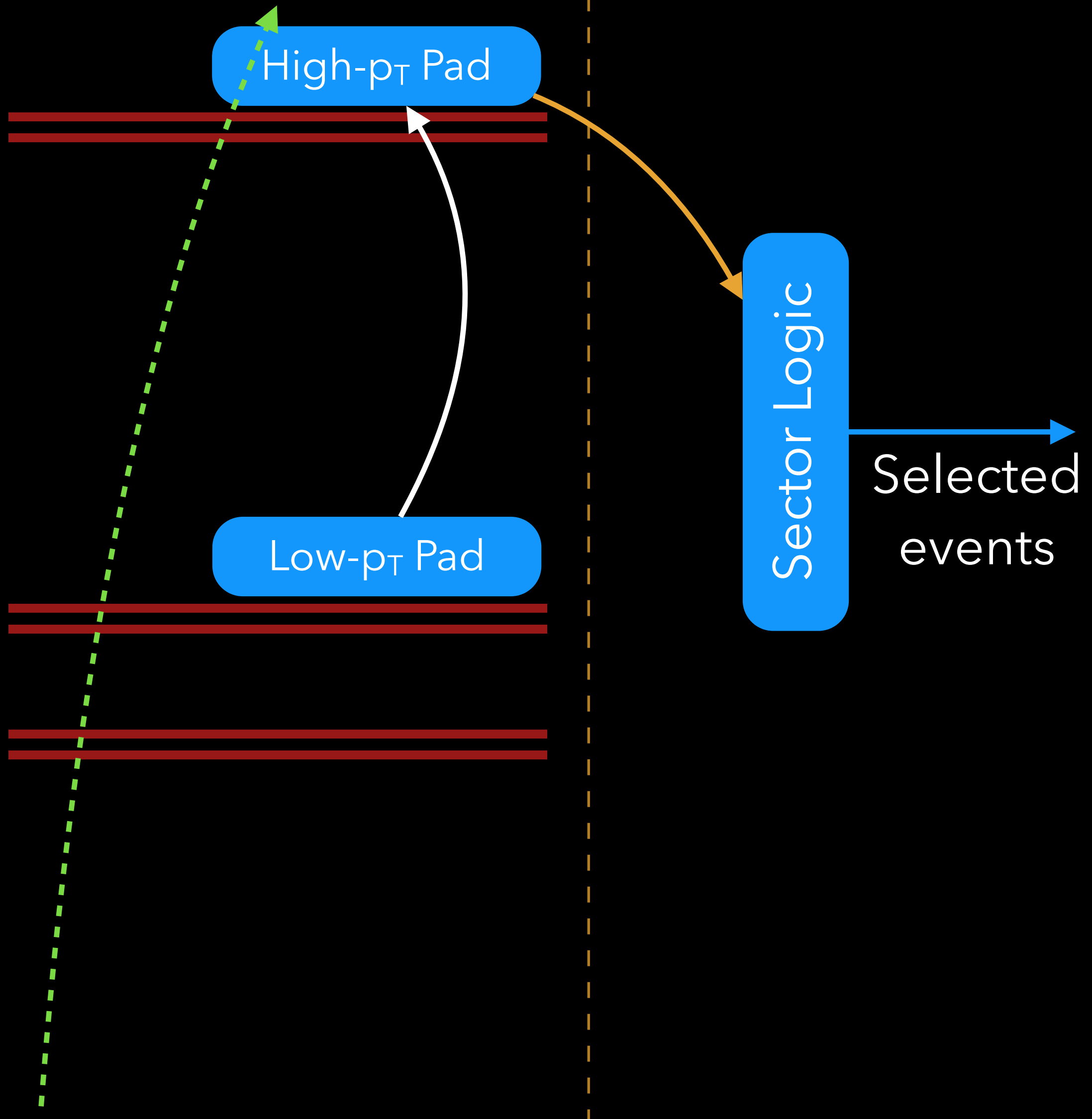




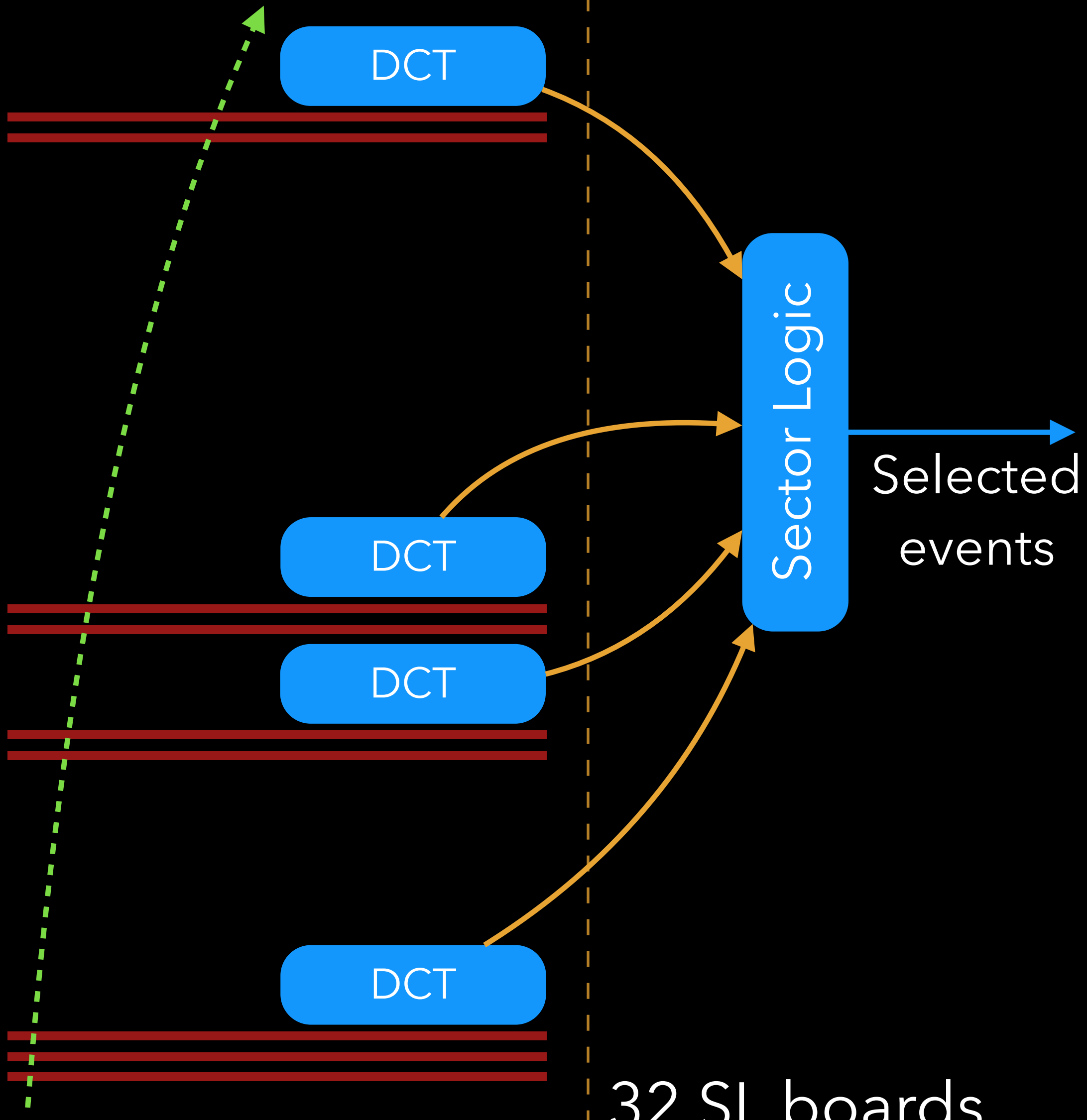


Barrel Muon trigger electronics upgrade

2008-2025



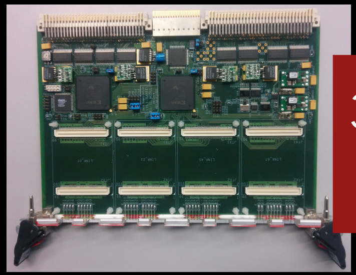
2029-2040



832 Pad boards

High complexity
4 ASICs, 1 FPGA

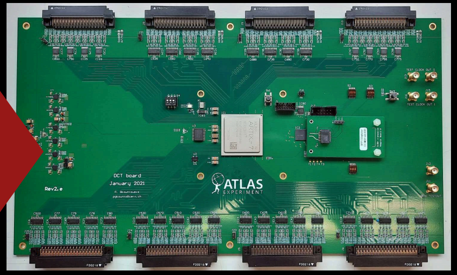
64 SL boards



Medium complexity
2 FPGAs

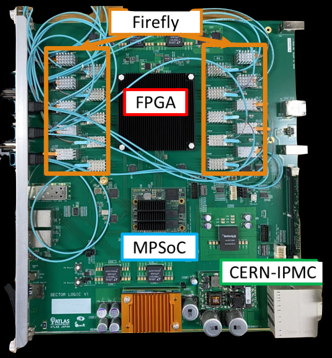
3328 ASICs + 896 FPGAs
1570 + 32 FPGAs

1570 DCT boards



Simple
1 FPGA

32 SL boards



High complexity
2 FPGAs

AI Level-0 muon trigger for ATLAS experiment

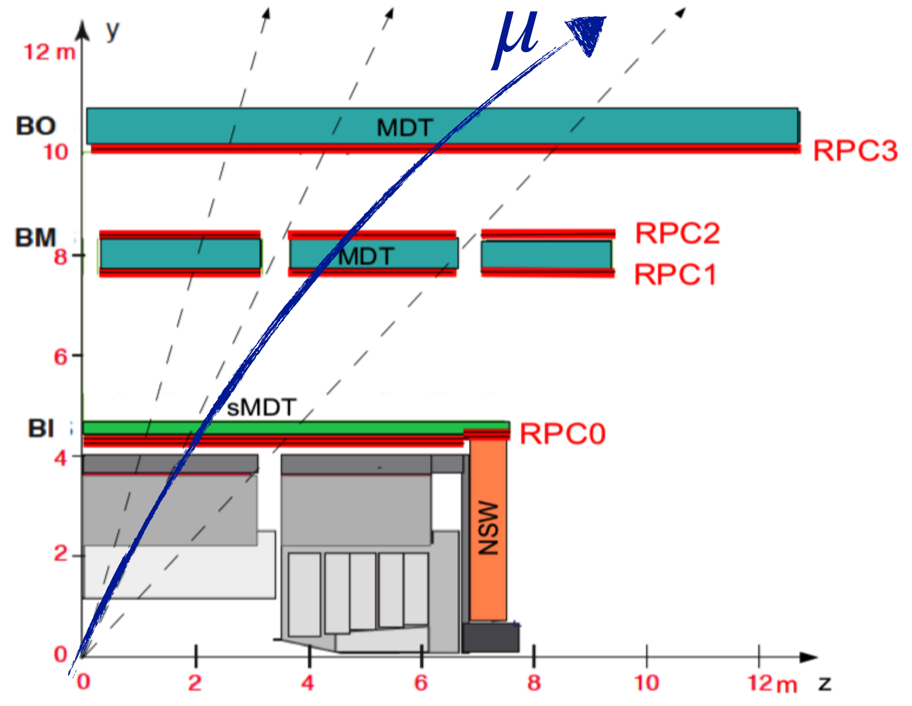
slide from S.Giagu

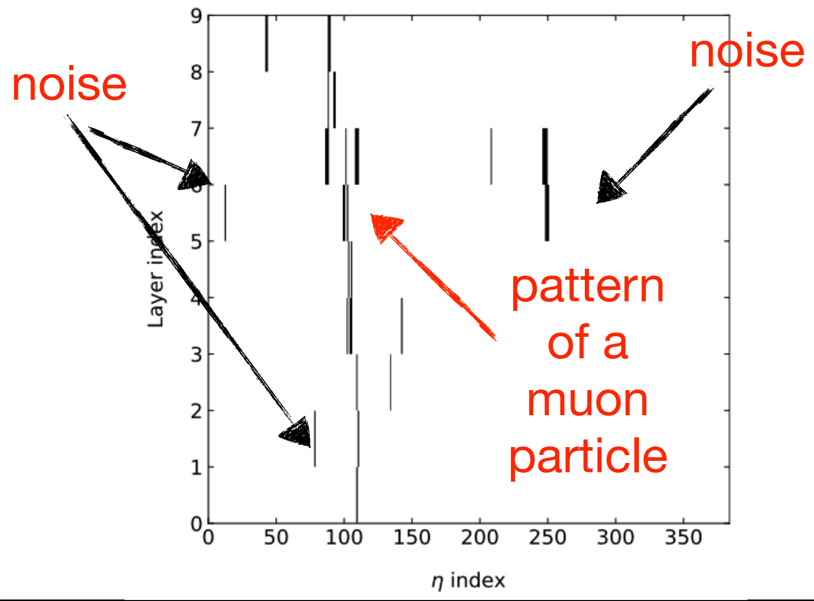
ULTRAFAST CNN ON FPGAS FOR THE ATLAS LEVEL-0 RPC MUON TRIGGER @HL-LHC

Goal: accurately reconstruct the momentum and angle of the muon track from the RPC detector hit information **in less than 400ns** (3 orders of magnitude faster than fastest AI models on CPUs and GPUs)

Latency and FPGA resource occupancy are in a trade-off relationship, while AI model performance strongly depends on the neural network scale

Strategy: muon identification and reconstruction via **Convolutional Neural Network** + multi-stage **AI model compression** and simplification based on **aggressive quantisation** and **knowledge transfer techniques** to avoid degradation of physics performances



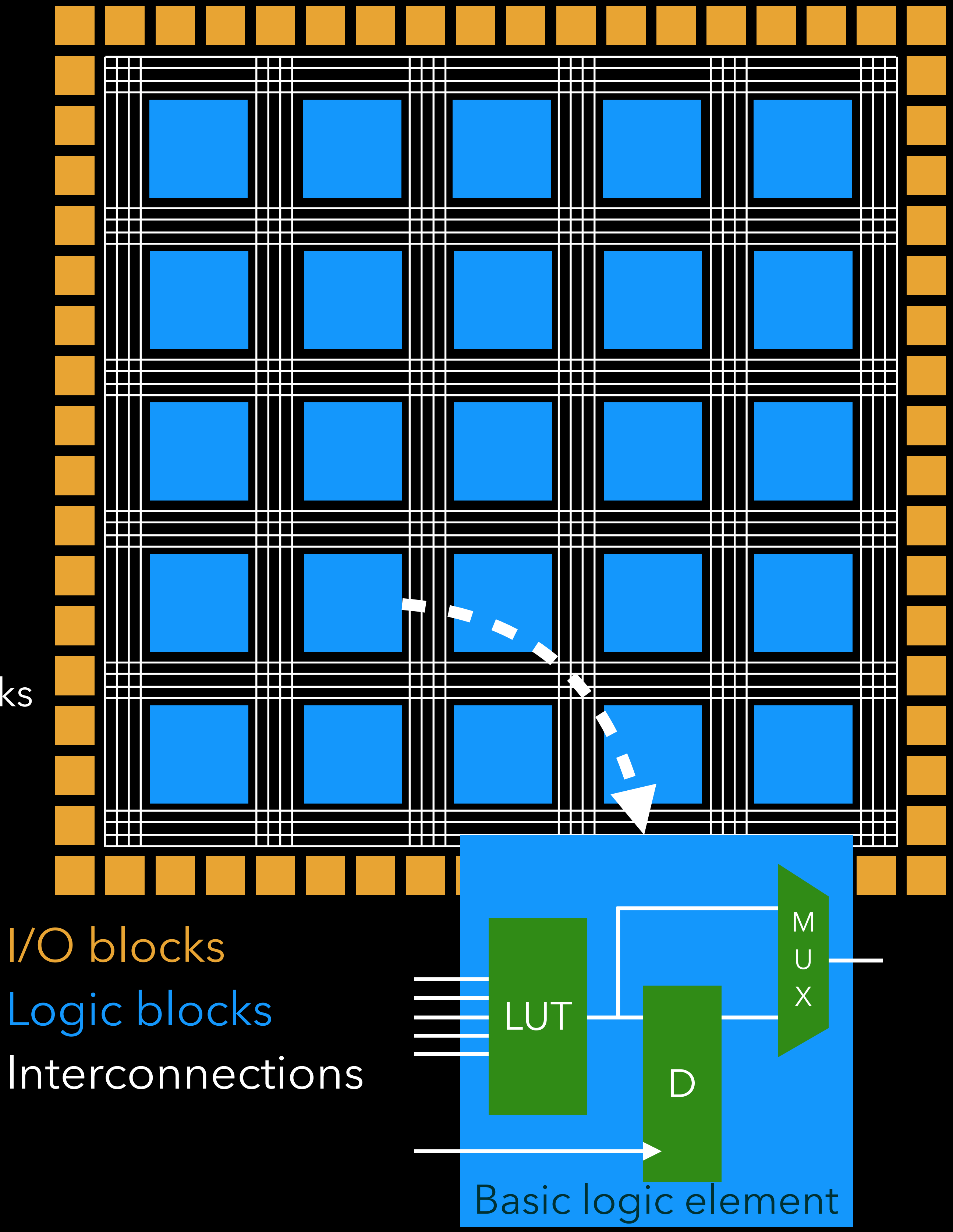


1

FPGAs

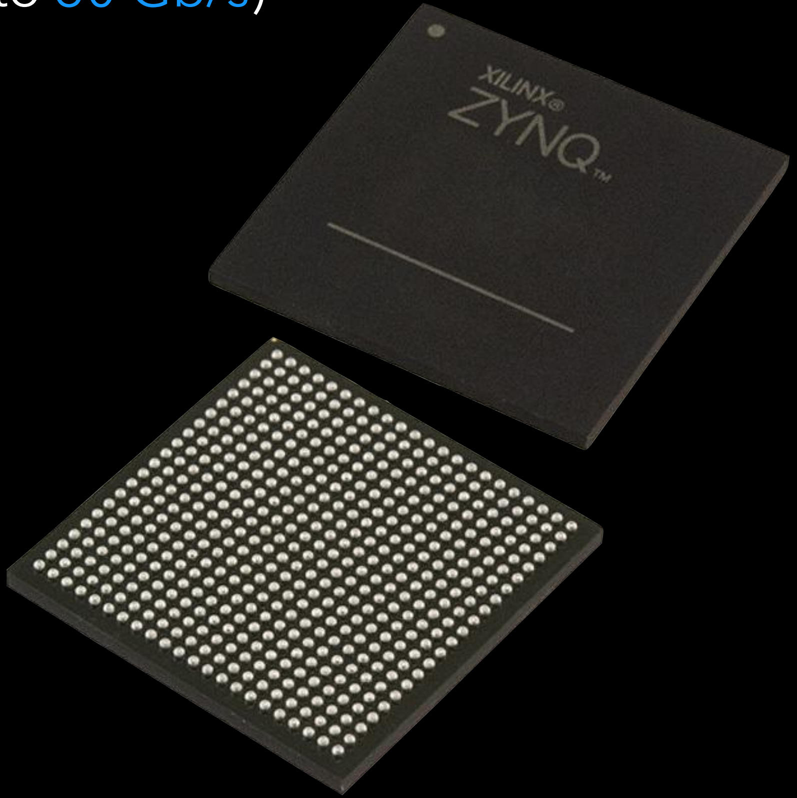
What is an FPGA?

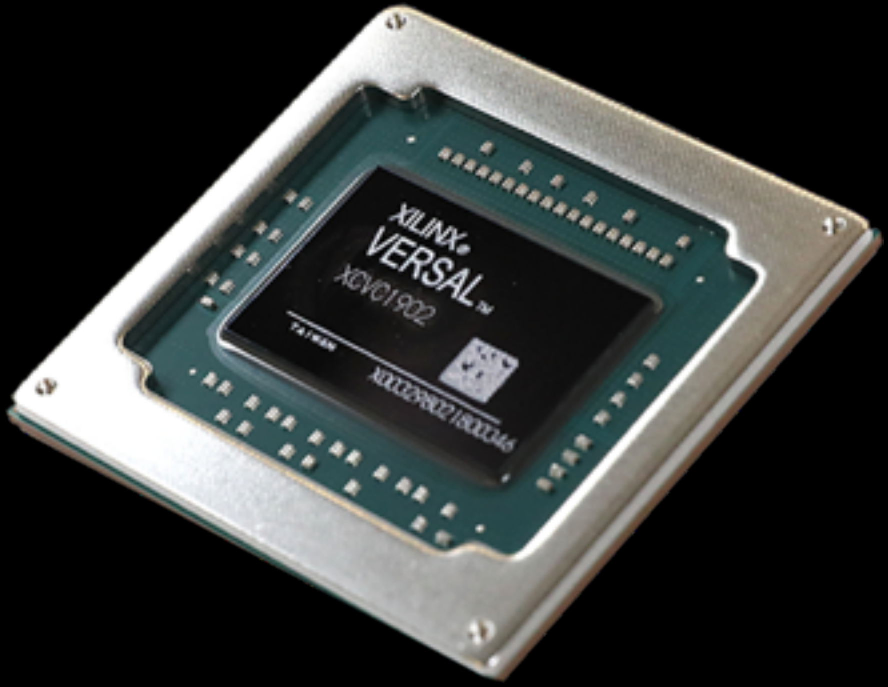
- Field Programmable Gate Array:
  - Field programmable: can be programmed by the user
  - Gate array: matrix of logic gates
- The internal logic is programmable:
  - Programmable I/O blocks
  - Matrix of user programmable logic blocks
  - Programmable network connecting the logic blocks
  - Clock trees
- One logic block is made of:
  - Combinatorial logic (LUT-RAM)
  - Sequential logic (flip-flops)
- Can contain several specific function blocks



What's inside an FPGA?

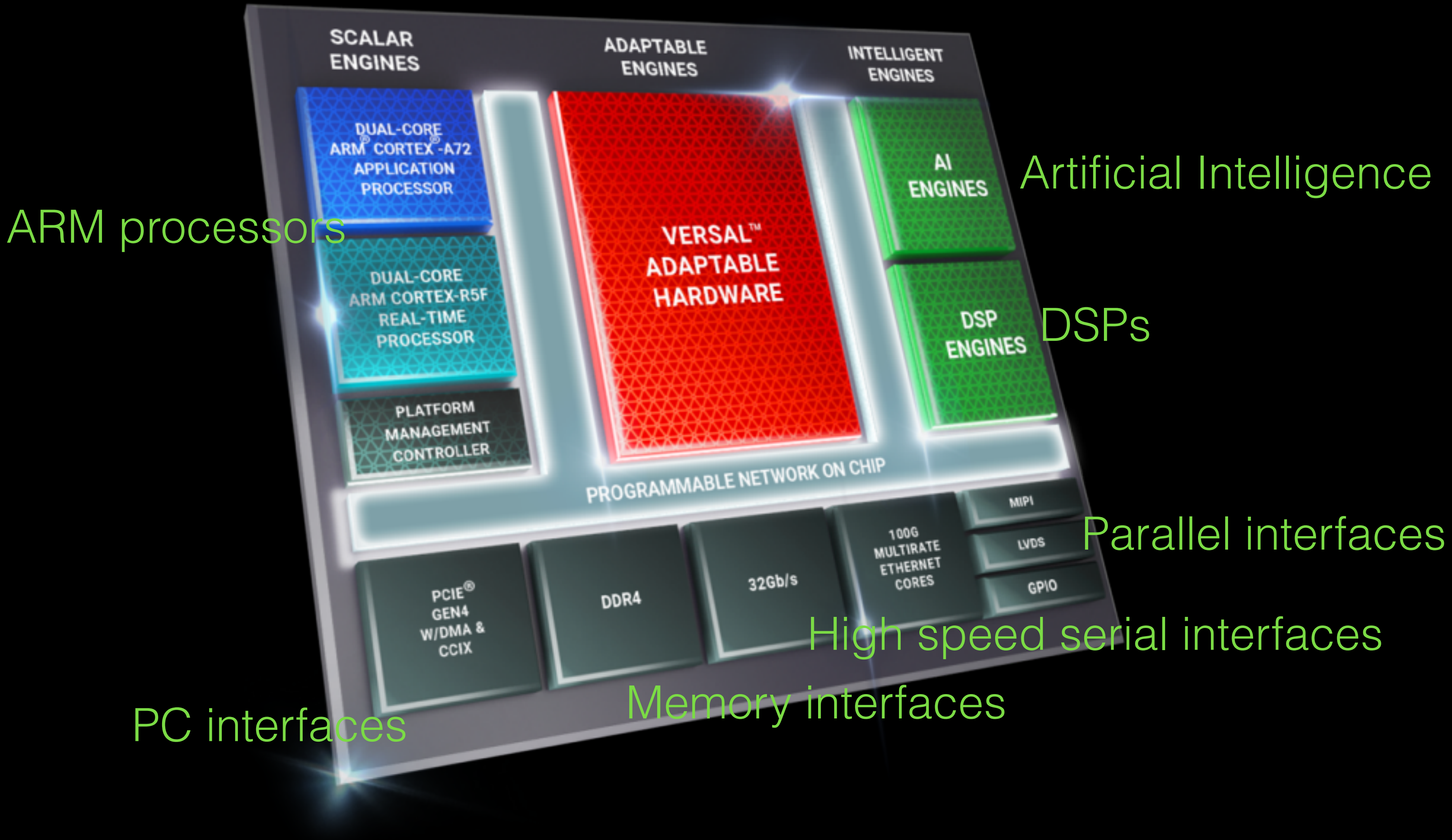
- I/O standard blocks (~100 to ~1000 IOs, input/output speed up to ~1Gb/s)
- I/O serialiser/deserialiser blocks (0 to ~100, transmission speed 1 Gb/s to 30 Gb/s)
- Programmable logic blocks (~10k to ~5M)
- DSP (Digital Signal Processing) blocks (~10 to ~15k)
- Clock multipliers and clock generators blocks (a few)
- RAM (~100 kB to ~500 MB)
- Memory interface controllers
- Ethernet interface
- PCI interface
- Analog functions (ADC)
- ARM processors (up to quad-core)
- Artificial Intelligence engines
- …





Example of a (complex) FPGA

AMD-Xilinx Versal ACAP (Adaptive Compute Acceleration Platform)


SCALAR ENGINES
ADAPTABLE ENGINES
INTELLIGENT ENGINES
DUAL-CORE ARM® CORTEX®-A72 APPLICATION PROCESSOR
DUAL-CORE ARM CORTEX-R5F REAL-TIME PROCESSOR
PLATFORM MANAGEMENT CONTROLLER
VERSAL™ ADAPTABLE HARDWARE
AI ENGINES
DSP ENGINES
PROGRAMMABLE NETWORK ON CHIP
PCIE® GEN4 W/DMA & CCIX
DDR4
32Gb/s
100G MULTIRATE ETHERNET CORES
MIPI
LVDS
GPIO
ARM processors
Artificial Intelligence
DSPs
Parallel interfaces
High speed serial interfaces
Memory interfaces
PC interfaces

FPGA technology

- Different possible ways to implement the FPGA programmable logic:
  - SRAM, antifuse, flash, EEPROM, …
  - SRAM technology is currently the most commonly used

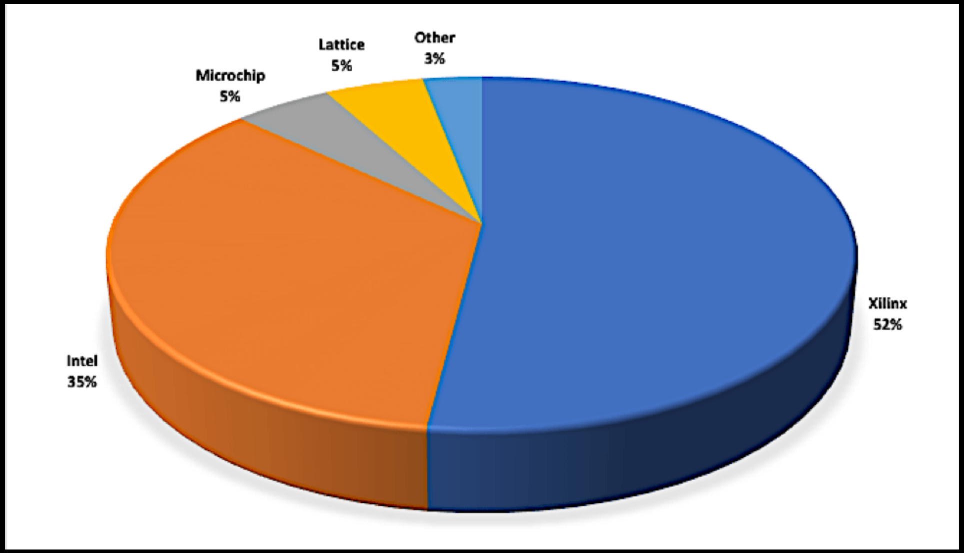
- Chip transistor size (CMOS technology):
  - FPGA (AMD-Xilinx): from 28 nm (lower cost) down to 7 nm (more expensive)
  - CPU: 10 nm Intel, 7nm AMD, 3nm Apple (TSMC manufacturing company)
  - GPU: 12 nm (Nvidia)

Why FPGA for HEP trigger?

- HEP experiments are using FPGA since many years
- Future hardware-based trigger systems are more and more FPGA based:
    - High number of I/O transceivers (>50 Gb/s), used to receive the incoming detector data (many channels)
    - High level of complexity, allow to perform complex algorithms in real time and low latency
    - High level of flexibility: peak finding, pattern recognition, track finding, clustering/energy summing, sorting, topological algorithms, control system, fast signal merging, neural networks and Artificial Intelligence based algorithms
    - Upgradable logic, very useful to change/upgrade the algorithms and to adapt to new physics requirements

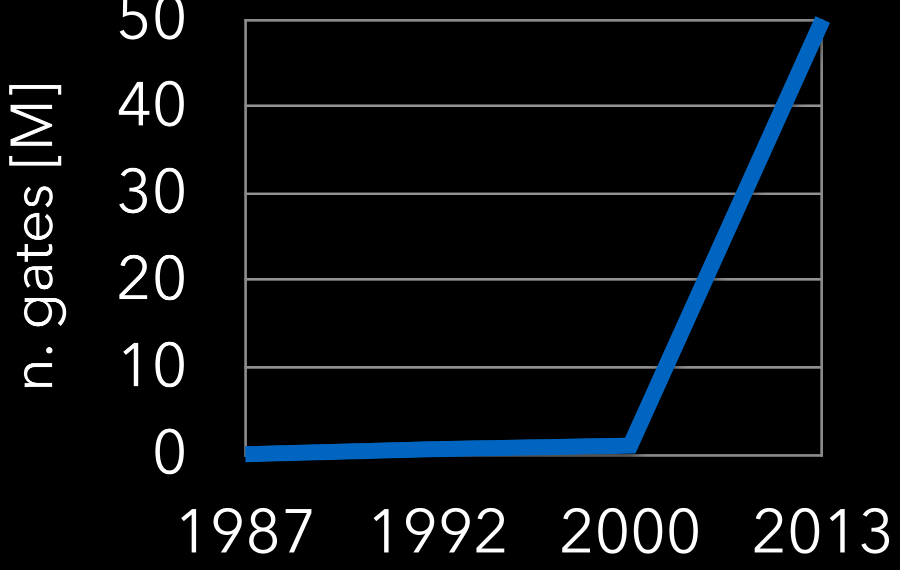
Main FPGA companies

FPGA market shares in 2019

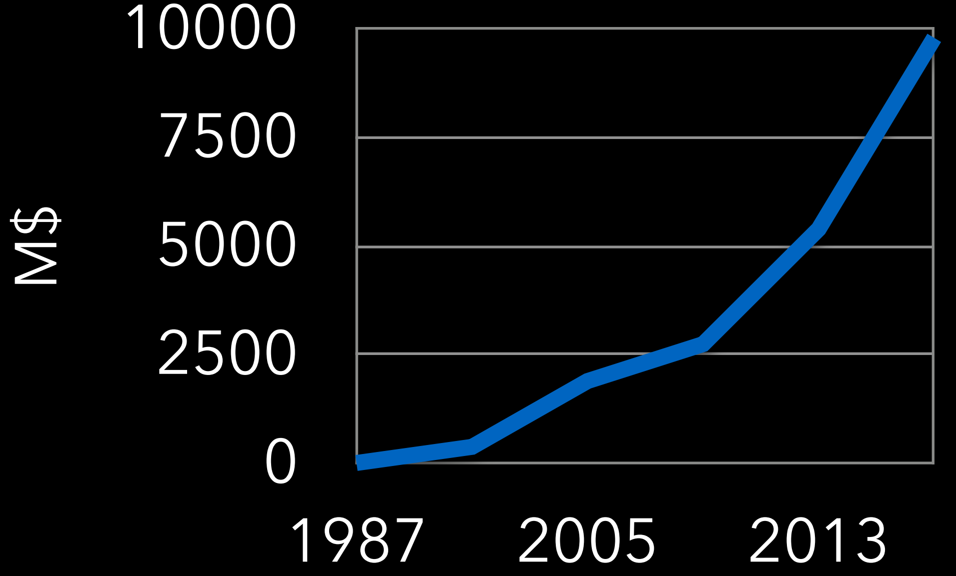


- AMD (SRAM technology, ex-Xilinx acquired by AMD in 2017 for 35 billion$)
- Intel (SRAM technology, ex-Altera acquired by Intel in 2015 for 17 billion$)
- Microchip (anti-fuse and flash, low power low cost)
- Lattice (SRAM and flash, low power mixed signals)

FPGA complexity



FPGA market



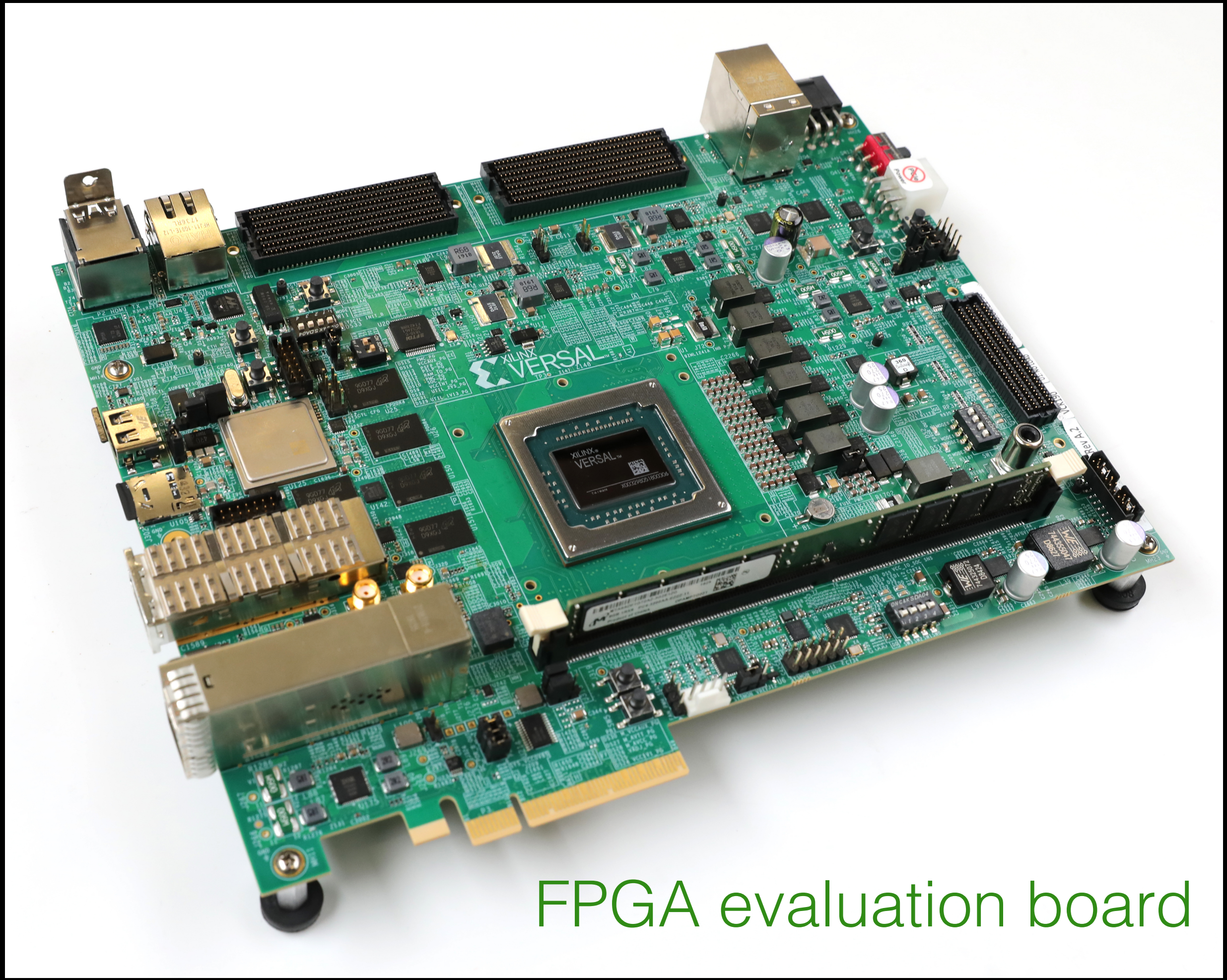
FPGA vs. ASIC

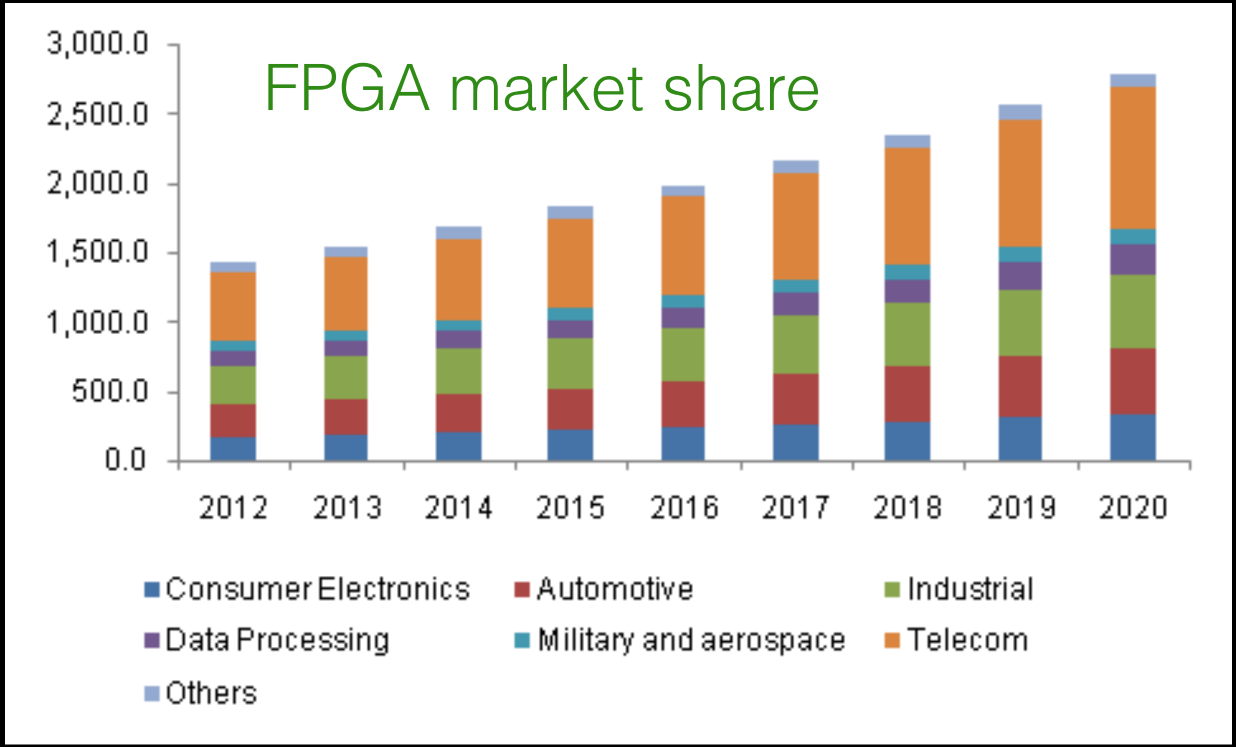
	FPGA	ASIC
Internal circuitry	Reconfigurable	Permanent
Learning curve	Easy 🙂	Difficult ☹️
Development time	Small 🙂	High ☹️
Power consumption	Less optimised ☹️	Very optimised 🙂
Maximum clock frequency	Lower ☹️	High 🙂
Analog logic	Limited	Not limited
Suitable for upgradable applications	Yes 🙂	No ☹️
Cost	Small for few devices	Small for many devices

- FPGA vs CPU:
  - CPU: software instructions are executed in sequence, parallelism achievable with multiple cores
  - FPGA: intrinsically parallel, many concurrent process run independently
    - Some FPGA models host multi core processors for maximum flexibility

FPGA possible applications

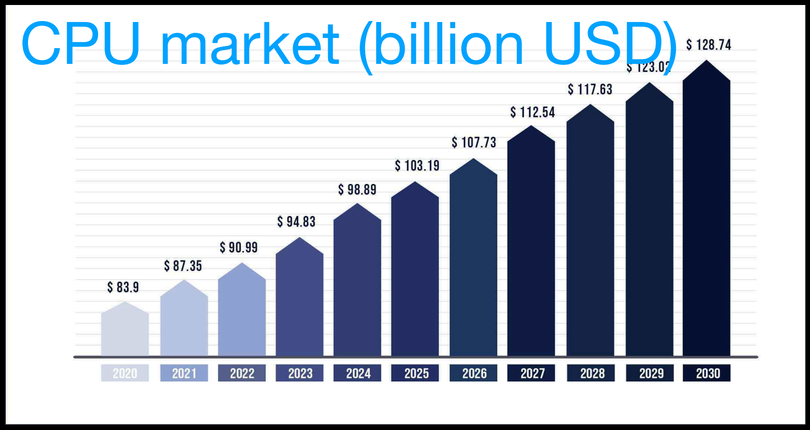
- Data Centres: video and image processing, speech recognition, cryptography, computational storage, database and data analytics, financial technology, high performance computing, network acceleration
- Communications: 5G wireless, wired and wireless communications
- Industry: aerospace, automotive, broadcast, consumer electronics, emulation, prototyping, medical, test and measurement
- Compute acceleration
- Artificial intelligence acceleration
- High Energy Physics
- …

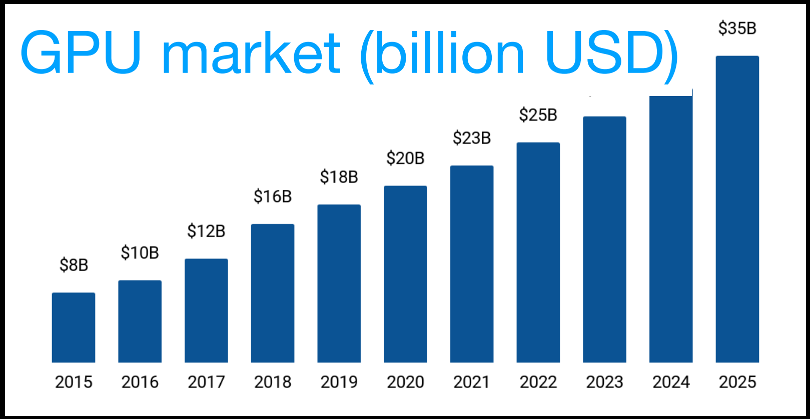


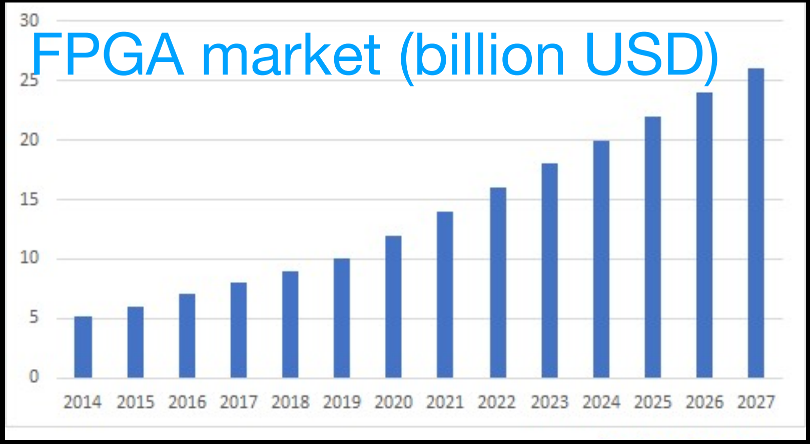


HEP and FPGA summary

- HEP trigger systems need to face increasing detector data bandwidths and challenging event selection requests (~Tb/s detector data down to ~Mb/s permanent storage)
- Moving complexity to off-detector electronics, keep detector front-end electronics as simple as possible, raw data sent to the off-detector trigger system
- First level trigger systems mostly (not always) based on FPGA boards:
    - high flexibility, high IO bandwidth
    - different algorithms can run on the same board, algorithms can adapt/evolve over time
- Higher level trigger systems mostly based on CPU farms with hardware accelerators (GPU or FPGA)
    - hardware accelerators allow to reduce the number of servers and power
    - high performance network needed for servers interconnectivity
    - FPGAs/GPUs hardware/software moving towards AI







Intel acquired Altera FPGA company in 2015 for 17 billion$

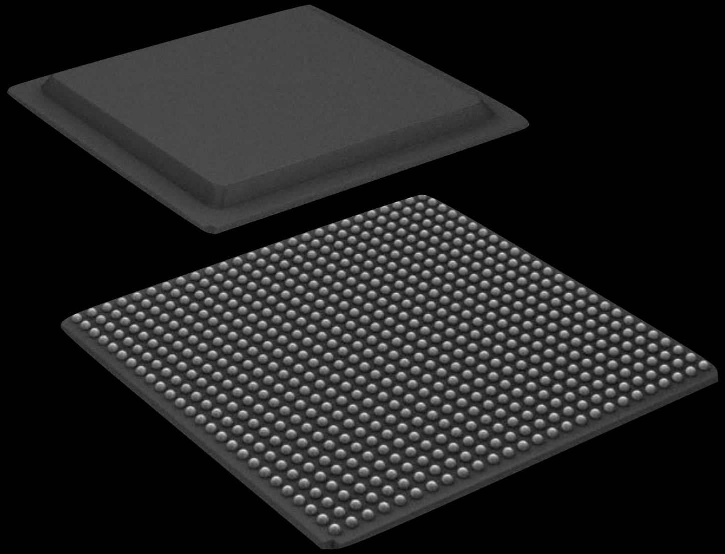
AMD acquired Xilinx FPGA company in 2017 for 35 billion$

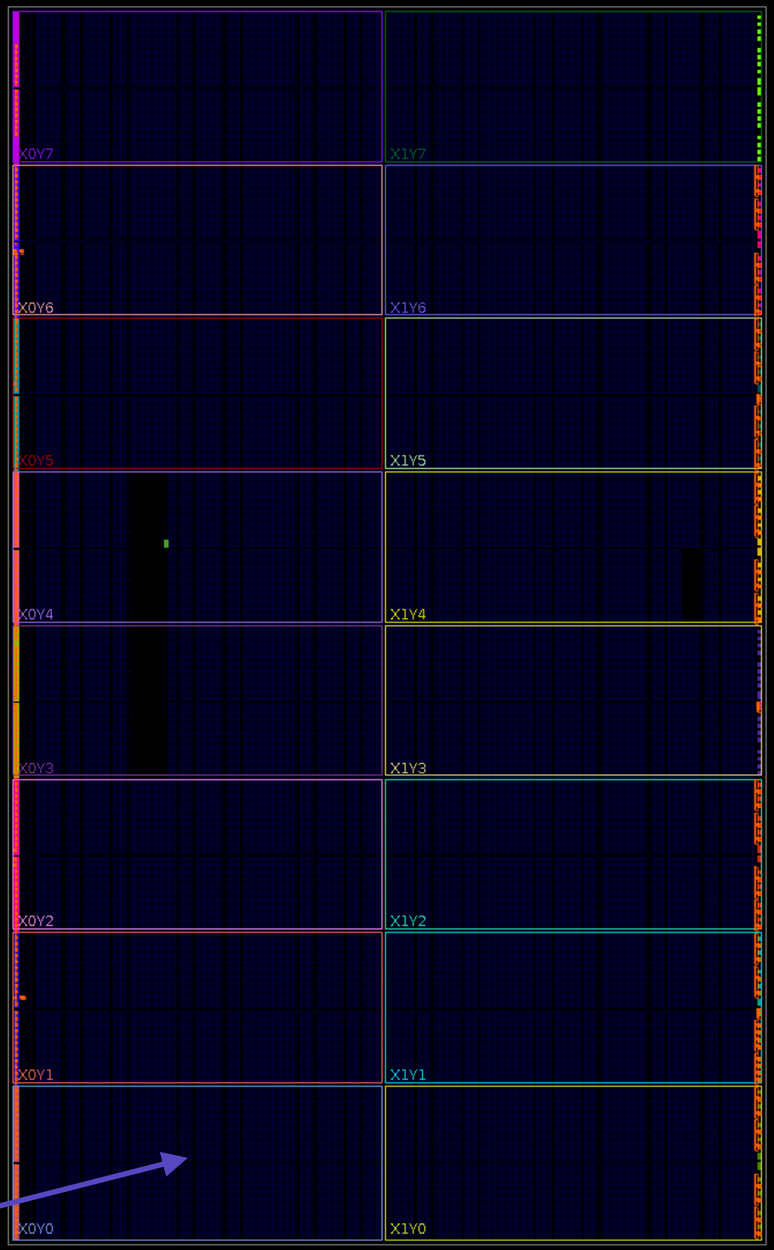
FPGA implementation workflow

example with AMD-Xilinx Vivado software

FPGA inside look

- Many different FPGA models on the market with programmable logic, some with processor cores and AI engines logic
- Let's look at the AMD-Xilinx Kintex-7 family, medium complexity FPGA family, no processors and no AI
- Package size: (2.3 x 2.3) cm^2 to (3.5 x 3.5) cm^2
- Medium cost FPGA (150€ to 5000€) depending on the device size, speed grade, working temperature range, voltage
- Biggest device has: ~500k logic cells (corresponding to millions of transistors), ~600k flip-flops, ~7MB distributed RAM, ~35 MB block RAM, ~2k DSP slices, 10 clock multipliers, 1 ADC, 1 PCI block module, 500 user I/O pins (max speed 1.3 Gb/s), 32 high speed transceiver (12.5 Gb/s)
- Higher performances FPGA families also available, with much more resources, and at a much higher cost (more than 50 k€)





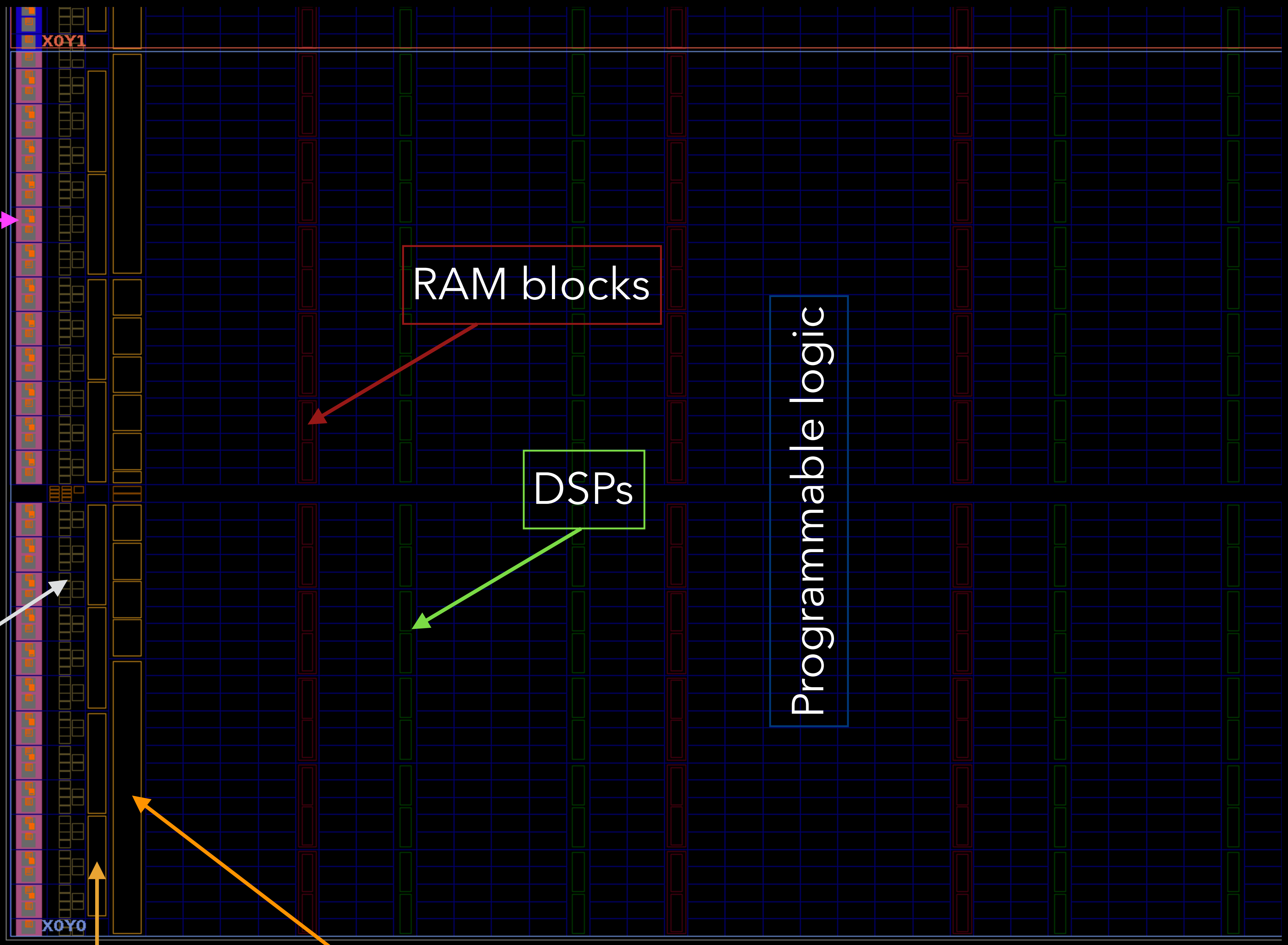
Programmable logic

Standard I/O ports:
Up to 500 I/O pins
Max speed 1.2 Gb/s

Fast I/O ports:
Up to 32 ports
Max speed 12.5 Gb/s

FPGA internal blocks

Configurable I/O ports:
- Input or output
- Voltage level
- Differential or unipolar



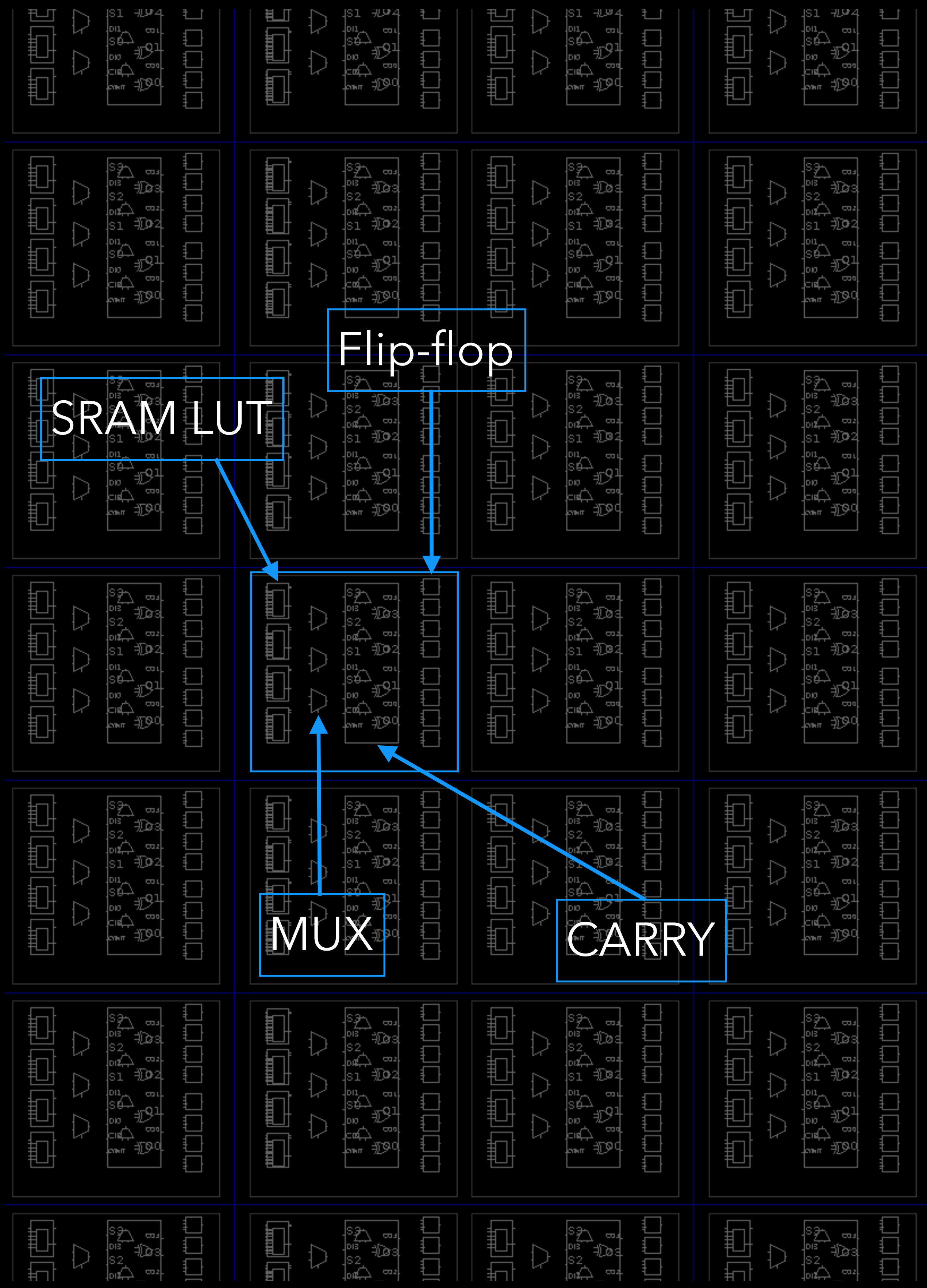
Configurable I/O logic:
- Input deserialiser
- Output serialiser
- Input/output delay

Input/output FIFOs

Clock frequency multipliers

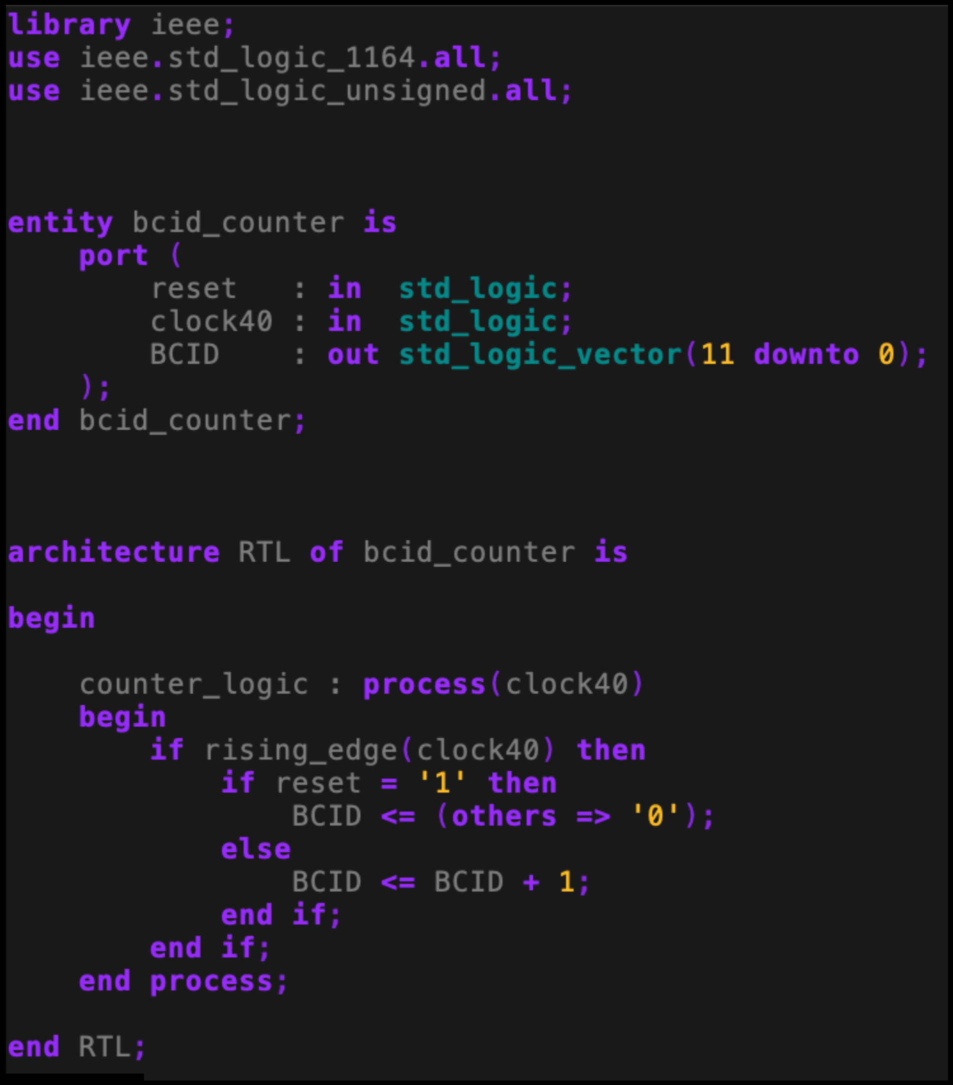
Programmable logic blocks

- A programmable logic block of a Xilinx Kintex-7 FPGA is made of:
  - 4 SRAM LUT (5-bit address)
  - 3 MUXes (2 in 1 out)
  - 1 CARRY logic block (useful to perform arithmetics operations like adders, …)
  - 8 flip-flops

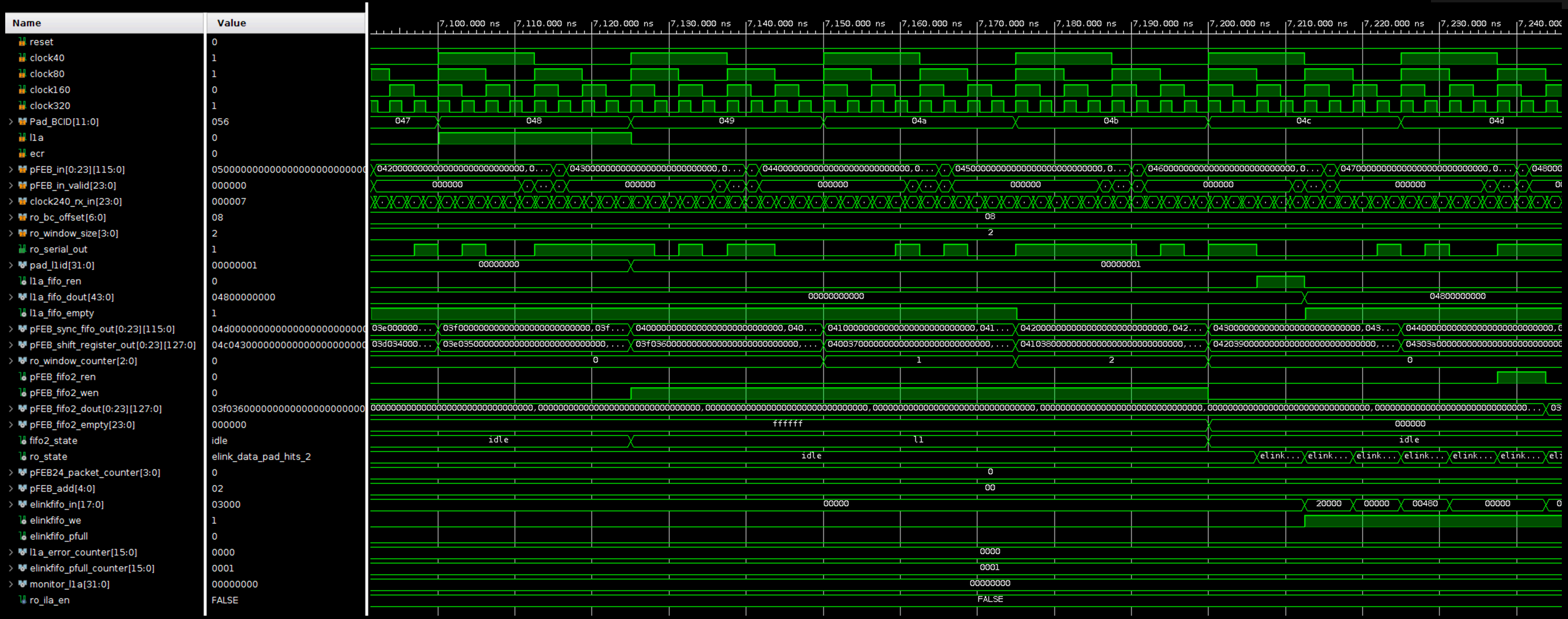


Hardware Description Languages

- Describe the behaviour of electronic circuits, most commonly digital logic
- Main languages used: VHDL and Verilog
- Simulator programs allow to verify the correctness of the HDL logic
- Usage of C++ or similar languages lately possible (in combination with VHDL/Verilog or not), useful for FPGAs with embedded processors, CPU interfaces and AI cores



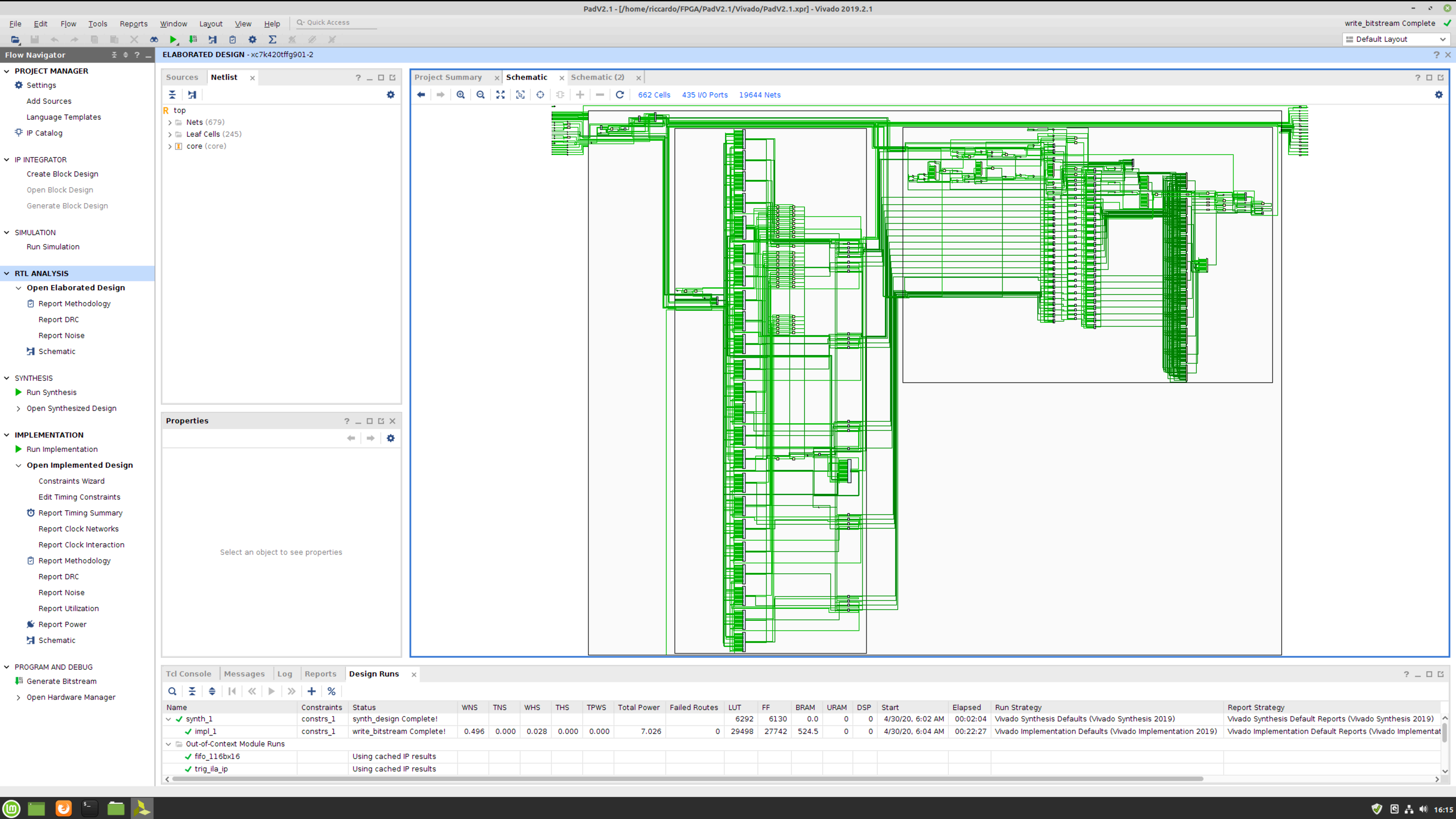
```vhdl
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;


entity bcid_counter is
    port (
        reset   : in  std_logic;
        clock40 : in  std_logic;
        BCID    : out std_logic_vector(11 downto 0);
    );
end bcid_counter;


architecture RTL of bcid_counter is

begin

    counter_logic : process(clock40)
    begin
        if rising_edge(clock40) then
            if reset = '1' then
                BCID <= (others => '0');
            else
                BCID <= BCID + 1;
            end if;
        end if;
    end process;

end RTL;
```



Name	Value
reset	0
clock40	1
clock80	1
clock160	0
clock320	1
Pad_BCID[11:0]	056
l1a	0
ecr	0
pFEB_in[0:23][115:0]	0500000000000000000000000000
pFEB_in_valid[23:0]	000000
clock240_rx_in[23:0]	000007
ro_bc_offset[6:0]	08
ro_window_size[3:0]	2
ro_serial_out	1
pad_l1id[31:0]	00000001
l1a_fifo_ren	0
l1a_fifo_dout[43:0]	04800000000
l1a_fifo_empty	1
pFEB_sync_fifo_out[0:23][115:0]	04d0000000000000000000000000
pFEB_shift_register_out[0:23][127:0]	04c0430000000000000000000000
ro_window_counter[2:0]	0
pFEB_fifo2_ren	0
pFEB_fifo2_wen	0
pFEB_fifo2_dout[0:23][127:0]	03f0360000000000000000000000
pFEB_fifo2_empty[23:0]	000000
fifo2_state	idle
ro_state	elink_data_pad_hits_2
pFEB24_packet_counter[3:0]	0
pFEB_add[4:0]	02
elinkfifo_in[17:0]	03000
elinkfifo_we	1
elinkfifo_pfull	0
l1a_error_counter[15:0]	0000
elinkfifo_pfull_counter[15:0]	0001
monitor_l1a[31:0]	00000000
ro_ila_en	FALSE

RTL analysis

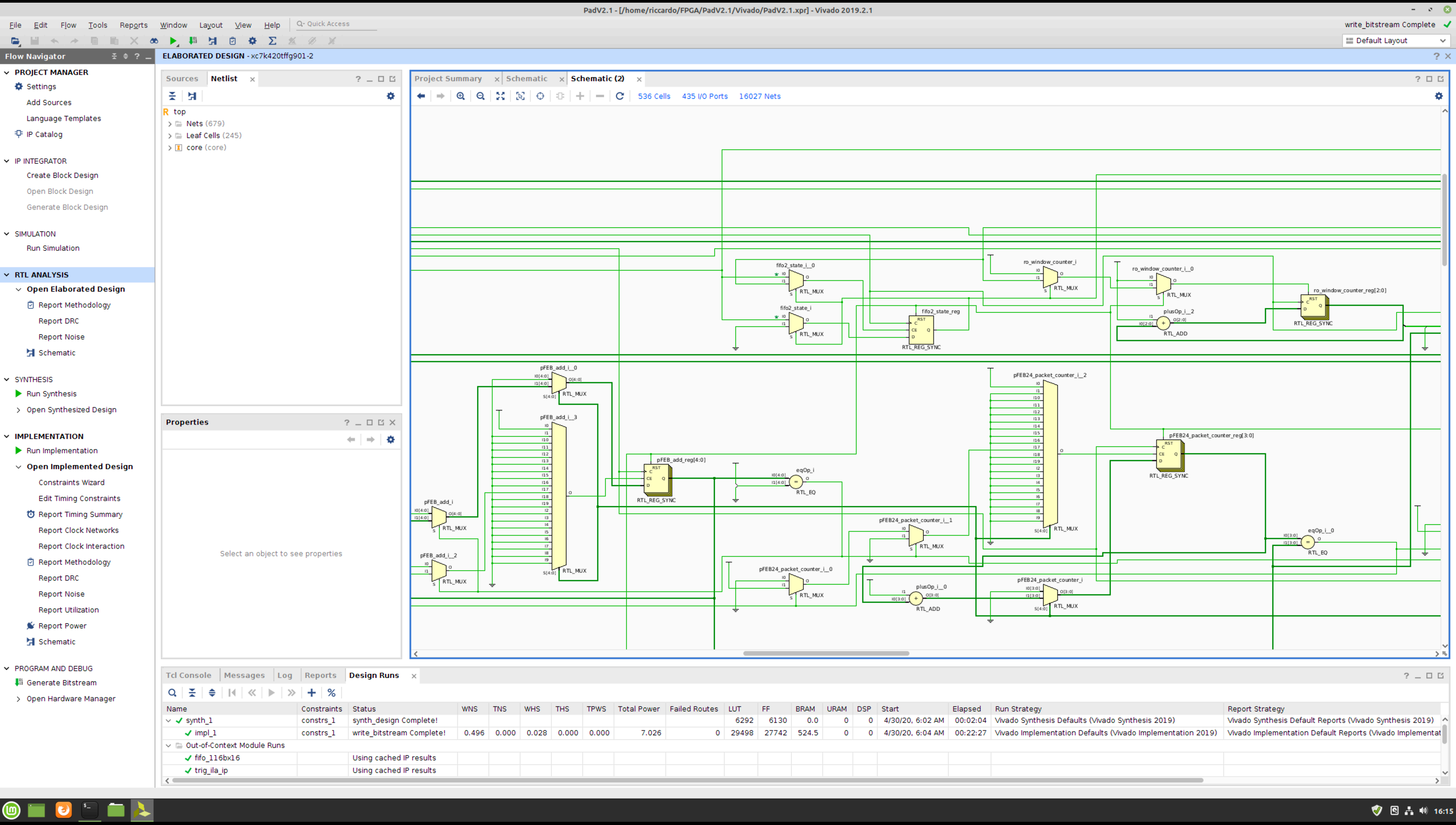
- Register Transfer Level analysis translates the HDL code into a digital circuit schema



Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy
✓ synth_1	constrs_1	synth_design Complete!								6292	6130	0.0	0	0	4/30/20, 6:02 AM	00:02:04	Vivado Synthesis Defaults (Vivado Synthesis 2019)	Vivado Synthesis Default Reports (Vivado Synthesis 2019)
✓ impl_1	constrs_1	write_bitstream Complete!	0.496	0.000	0.028	0.000	0.000	7.026	0	29498	27742	524.5	0	0	4/30/20, 6:04 AM	00:22:27	Vivado Implementation Defaults (Vivado Implementation 2019)	Vivado Implementation Default Reports (Vivado Implementat
Out-of-Context Module Runs																		
✓ fifo_116bx16		Using cached IP results																
✓ trig_ila_ip		Using cached IP results																

RTL analysis logic elements

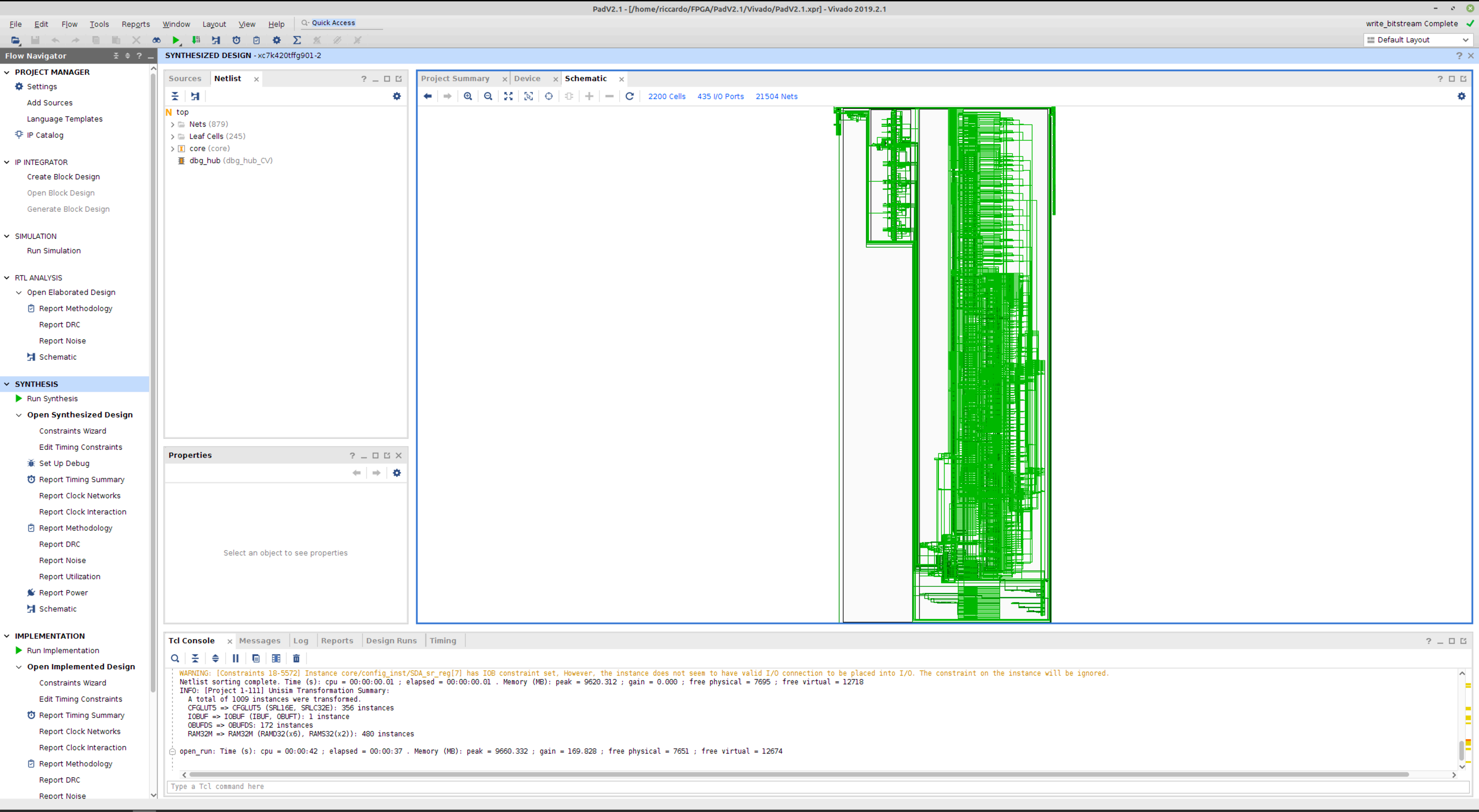
- Logic elements used are generic, independent from the FPGA chosen family



Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy
✓ synth_1	constrs_1	synth_design Complete!								6292	6130	0.0	0	0	4/30/20, 6:02 AM	00:02:04	Vivado Synthesis Defaults (Vivado Synthesis 2019)	Vivado Synthesis Default Reports (Vivado Synthesis 2019)
✓ impl_1	constrs_1	write_bitstream Complete!	0.496	0.000	0.028	0.000	0.000	7.026	0	29498	27742	524.5	0	0	4/30/20, 6:04 AM	00:22:27	Vivado Implementation Defaults (Vivado Implementation 2019)	Vivado Implementation Default Reports (Vivado Implementat
Out-of-Context Module Runs																		
✓ fifo_116bx16		Using cached IP results																
✓ trig_ila_ip		Using cached IP results																

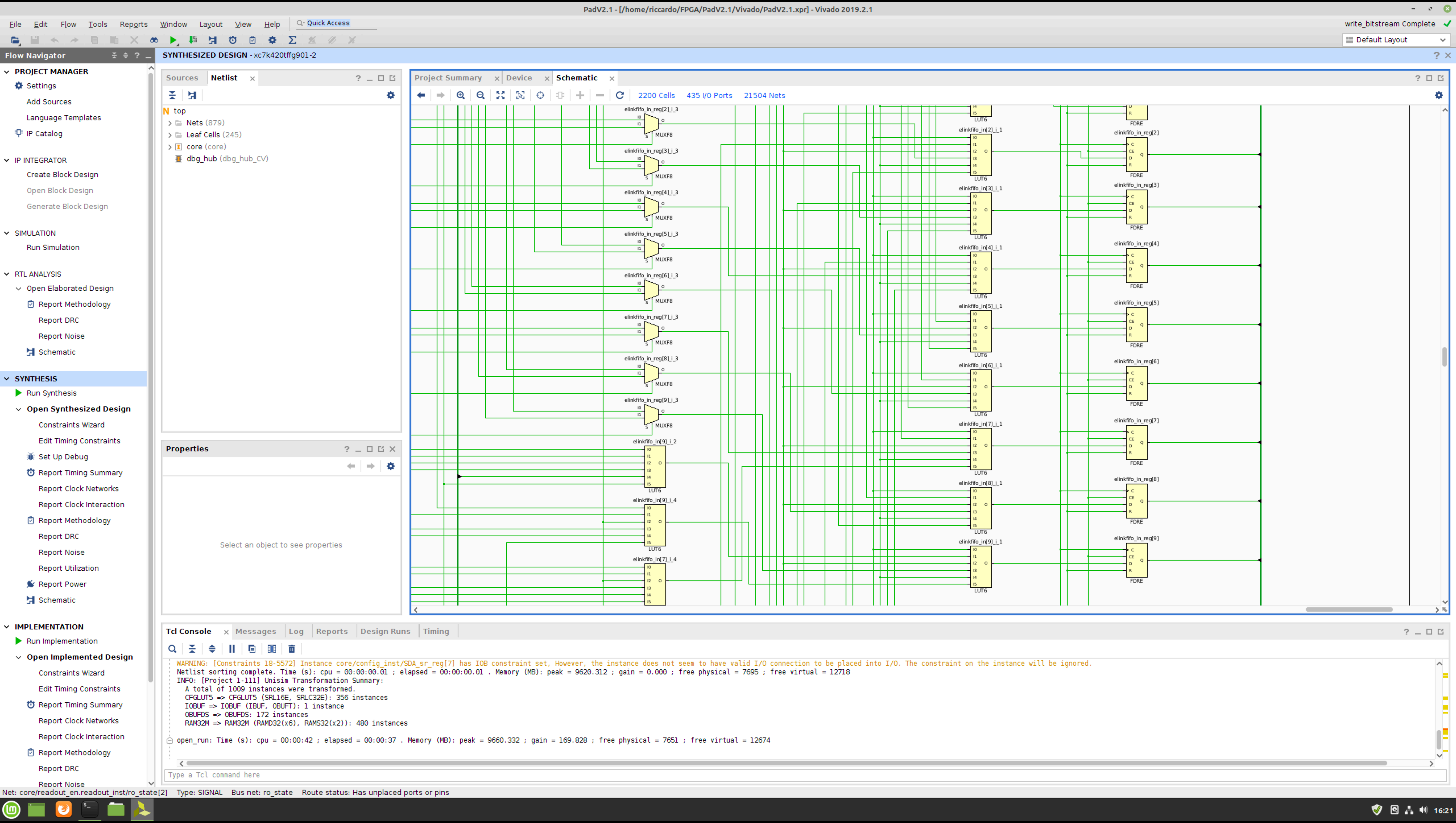
Synthesis

- Translates the analysed circuit into an equivalent circuit that uses the chosen FPGA logic blocks



```
WARNING: [Constraints 18-5572] Instance core/config_inst/SDA_sr_reg[7] has IOB constraint set, However, the instance does not seem to have valid I/O connection to be placed into I/O. The constraint on the instance will be ignored.
Netlist sorting complete. Time (s): cpu = 00:00:00.01 ; elapsed = 00:00:00.01 . Memory (MB): peak = 9620.312 ; gain = 0.000 ; free physical = 7695 ; free virtual = 12718
INFO: [Project 1-111] Unisim Transformation Summary:
  A total of 1009 instances were transformed.
  CFGLUT5 => CFGLUT5 (SRL16E, SRLC32E): 356 instances
  IOBUF => IOBUF (IBUF, OBUFT): 1 instance
  OBUFDS => OBUFDS: 172 instances
  RAM32M => RAM32M (RAMD32(x6), RAMS32(x2)): 480 instances

open_run: Time (s): cpu = 00:00:42 ; elapsed = 00:00:37 . Memory (MB): peak = 9660.332 ; gain = 169.828 ; free physical = 7651 ; free virtual = 12674
```

Synthesis logic blocks

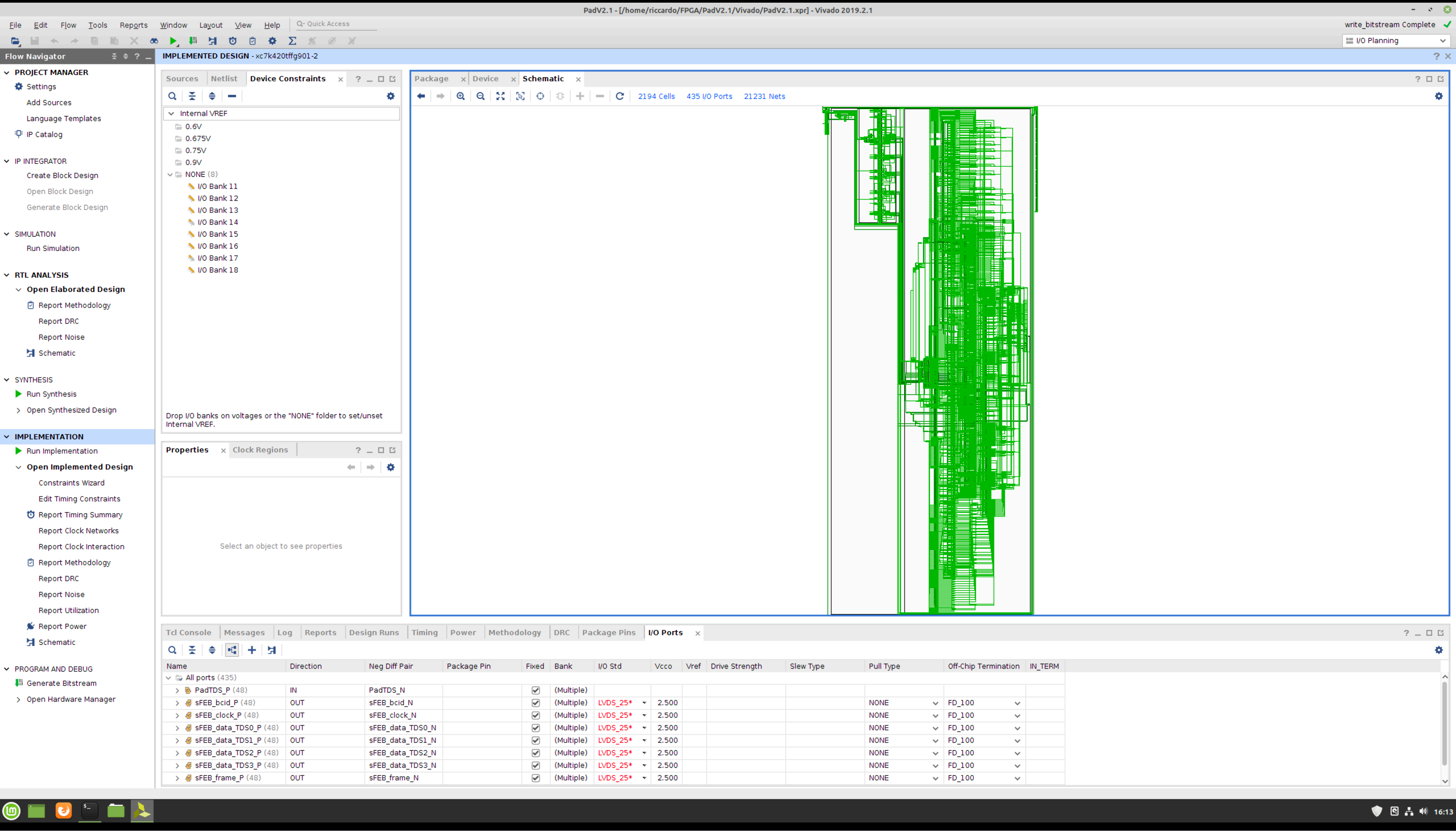


```
WARNING: [Constraints 18-5572] Instance core/config_inst/SDA_sr_reg[7] has IOB constraint set, However, the instance does not seem to have valid I/O connection to be placed into I/O. The constraint on the instance will be ignored.
Netlist sorting complete. Time (s): cpu = 00:00:00.01 ; elapsed = 00:00:00.01 . Memory (MB): peak = 9620.312 ; gain = 0.000 ; free physical = 7695 ; free virtual = 12718
INFO: [Project 1-111] Unisim Transformation Summary:
  A total of 1009 instances were transformed.
  CFGLUT5 => CFGLUT5 (SRL16E, SRLC32E): 356 instances
  IOBUF => IOBUF (IBUF, OBUFT): 1 instance
  OBUFDS => OBUFDS: 172 instances
  RAM32M => RAM32M (RAMD32(x6), RAMS32(x2)): 480 instances

open_run: Time (s): cpu = 00:00:42 ; elapsed = 00:00:37 . Memory (MB): peak = 9660.332 ; gain = 169.828 ; free physical = 7651 ; free virtual = 12674
```

Net: core/readout_en.readout_inst/ro_state[2]   Type: SIGNAL   Bus net: ro_state   Route status: Has unplaced ports or pins

Implementation

- Performs the final step: FPGA routing (logic block placement and interconnections) and checks the circuit timing. Logic can be changed/optimised during this process.



Internal VREF
- 0.6V
- 0.675V
- 0.75V
- 0.9V
- NONE (8)
  - I/O Bank 11
  - I/O Bank 12
  - I/O Bank 13
  - I/O Bank 14
  - I/O Bank 15
  - I/O Bank 16
  - I/O Bank 17
  - I/O Bank 18

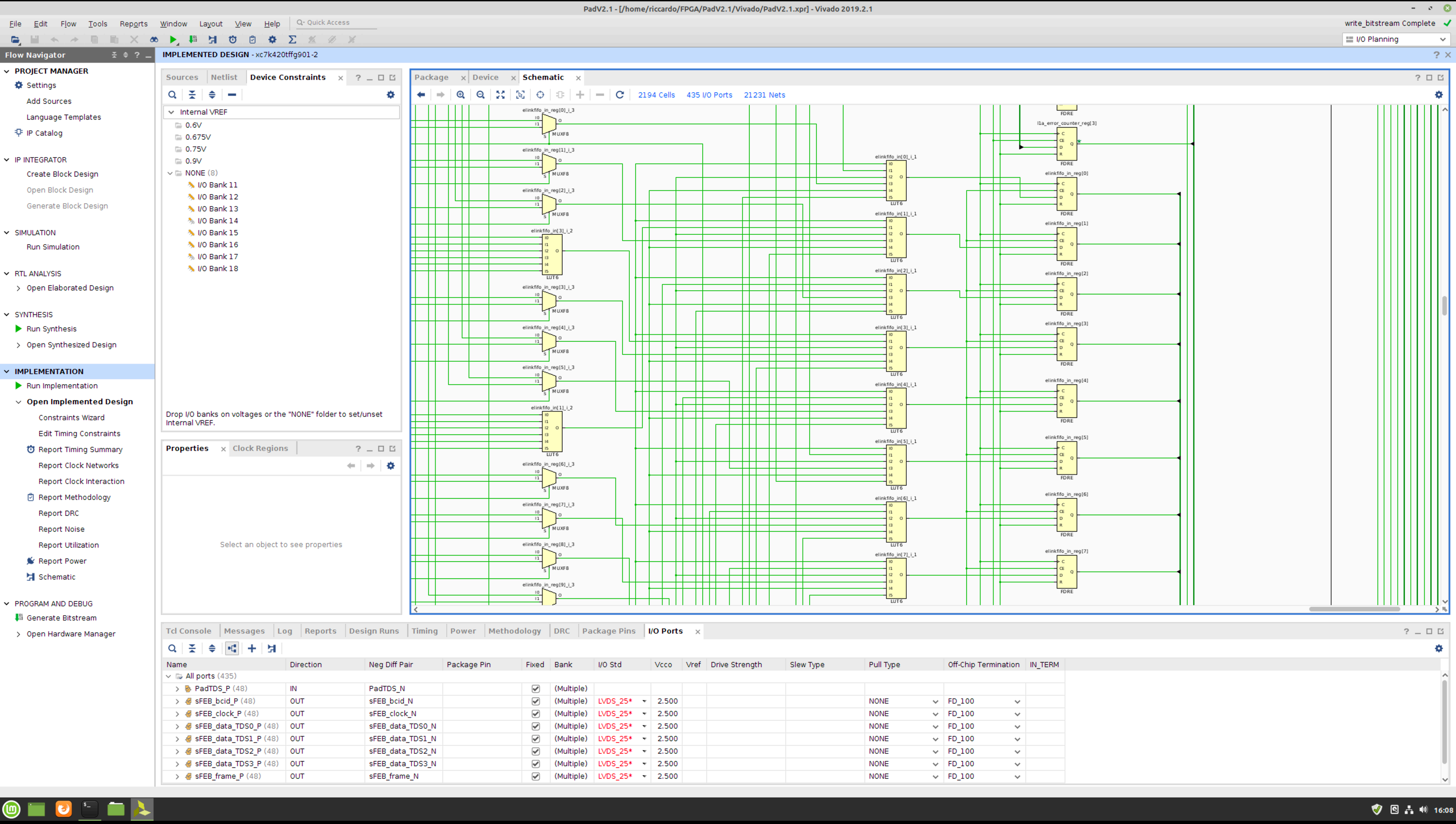
Drop I/O banks on voltages or the "NONE" folder to set/unset Internal VREF.

Select an object to see properties

Name	Direction	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength	Slew Type	Pull Type	Off-Chip Termination	IN_TERM
All ports (435)													
PadTDS_P (48)	IN	PadTDS_N		✓	(Multiple)								
sFEB_bcid_P (48)	OUT	sFEB_bcid_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_clock_P (48)	OUT	sFEB_clock_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS0_P (48)	OUT	sFEB_data_TDS0_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS1_P (48)	OUT	sFEB_data_TDS1_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS2_P (48)	OUT	sFEB_data_TDS2_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS3_P (48)	OUT	sFEB_data_TDS3_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_frame_P (48)	OUT	sFEB_frame_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	

Implementation result

- The circuit schema is changed if necessary, to achieve correct timing and taking into consideration the final cell placement.


PadV2.1 - [/home/riccardo/FPGA/PadV2.1/Vivado/PadV2.1.xpr] - Vivado 2019.2.1

write_bitstream Complete

IMPLEMENTED DESIGN - xc7k420tffg901-2

Internal VREF
- 0.6V
- 0.675V
- 0.75V
- 0.9V
- NONE (8)
  - I/O Bank 11
  - I/O Bank 12
  - I/O Bank 13
  - I/O Bank 14
  - I/O Bank 15
  - I/O Bank 16
  - I/O Bank 17
  - I/O Bank 18

Drop I/O banks on voltages or the "NONE" folder to set/unset Internal VREF.

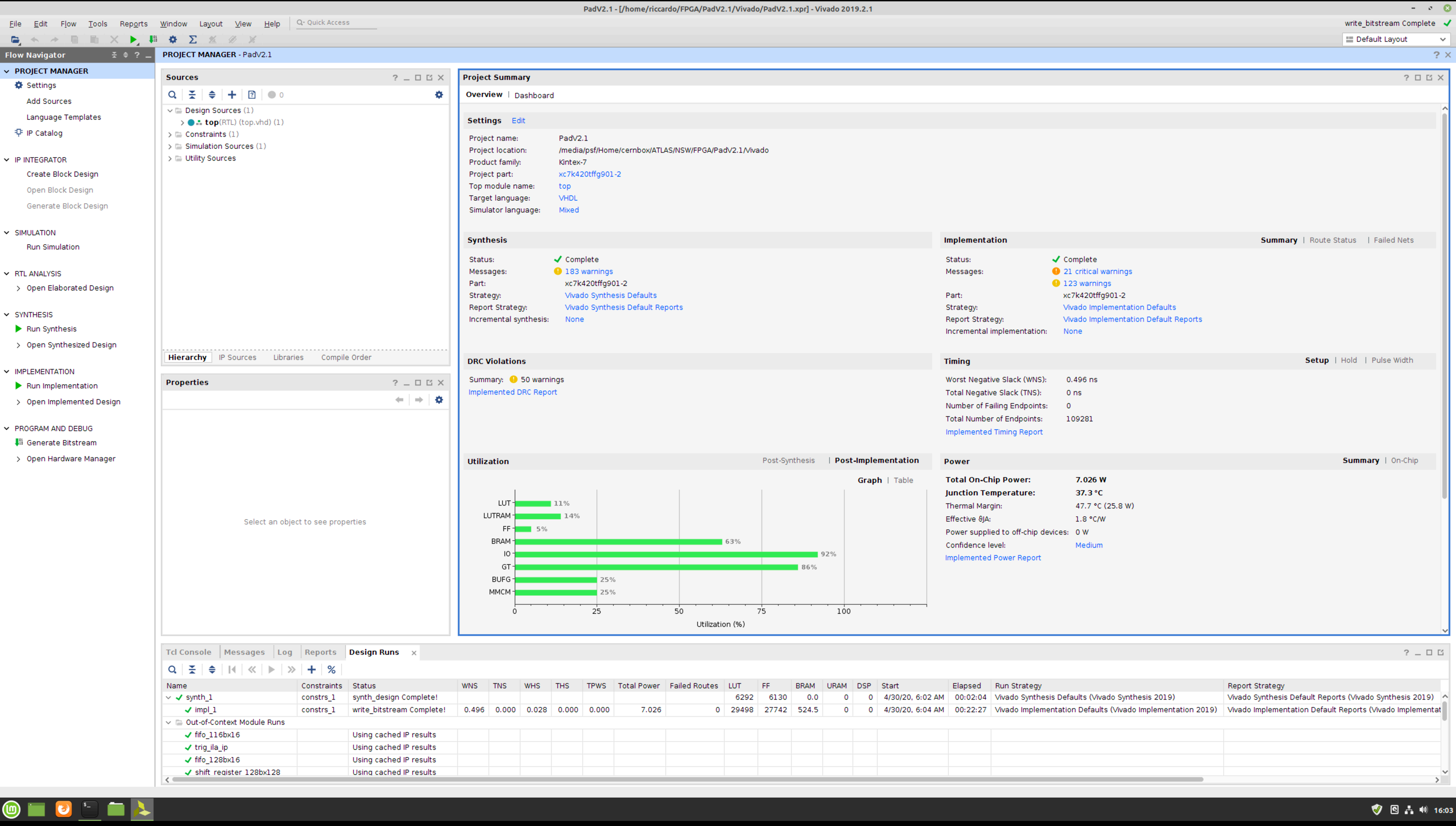
Select an object to see properties

2194 Cells 435 I/O Ports 21231 Nets

Name	Direction	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength	Slew Type	Pull Type	Off-Chip Termination	IN_TERM
All ports (435)													
PadTDS_P (48)	IN	PadTDS_N		✓	(Multiple)								
sFEB_bcid_P (48)	OUT	sFEB_bcid_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_clock_P (48)	OUT	sFEB_clock_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS0_P (48)	OUT	sFEB_data_TDS0_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS1_P (48)	OUT	sFEB_data_TDS1_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS2_P (48)	OUT	sFEB_data_TDS2_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS3_P (48)	OUT	sFEB_data_TDS3_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_frame_P (48)	OUT	sFEB_frame_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	

Implementation report

- A detailed report shows all implementation steps details and the FPGA resource usage (and the errors/warnings, if any) including timing results and power estimate.



PadV2.1 - [/home/riccardo/FPGA/PadV2.1/Vivado/PadV2.1.xpr] - Vivado 2019.2.1

write_bitstream Complete ✓

PROJECT MANAGER - PadV2.1

Project Summary

Overview | Dashboard

Settings Edit

Project name:	PadV2.1
Project location:	/media/psf/Home/cernbox/ATLAS/NSW/FPGA/PadV2.1/Vivado
Product family:	Kintex-7
Project part:	xc7k420tffg901-2
Top module name:	top
Target language:	VHDL
Simulator language:	Mixed

Synthesis

Status:	✓ Complete
Messages:	183 warnings
Part:	xc7k420tffg901-2
Strategy:	Vivado Synthesis Defaults
Report Strategy:	Vivado Synthesis Default Reports
Incremental synthesis:	None

Implementation (Summary | Route Status | Failed Nets)

Status:	✓ Complete
Messages:	21 critical warnings
	123 warnings
Part:	xc7k420tffg901-2
Strategy:	Vivado Implementation Defaults
Report Strategy:	Vivado Implementation Default Reports
Incremental implementation:	None

DRC Violations

Summary: 50 warnings

Implemented DRC Report

Timing (Setup | Hold | Pulse Width)

Worst Negative Slack (WNS):	0.496 ns
Total Negative Slack (TNS):	0 ns
Number of Failing Endpoints:	0
Total Number of Endpoints:	109281

Implemented Timing Report

Utilization (Post-Synthesis | Post-Implementation; Graph | Table)

Power (Summary | On-Chip)

Total On-Chip Power:	7.026 W
Junction Temperature:	37.3 °C
Thermal Margin:	47.7 °C (25.8 W)
Effective θJA:	1.8 °C/W
Power supplied to off-chip devices:	0 W
Confidence level:	Medium

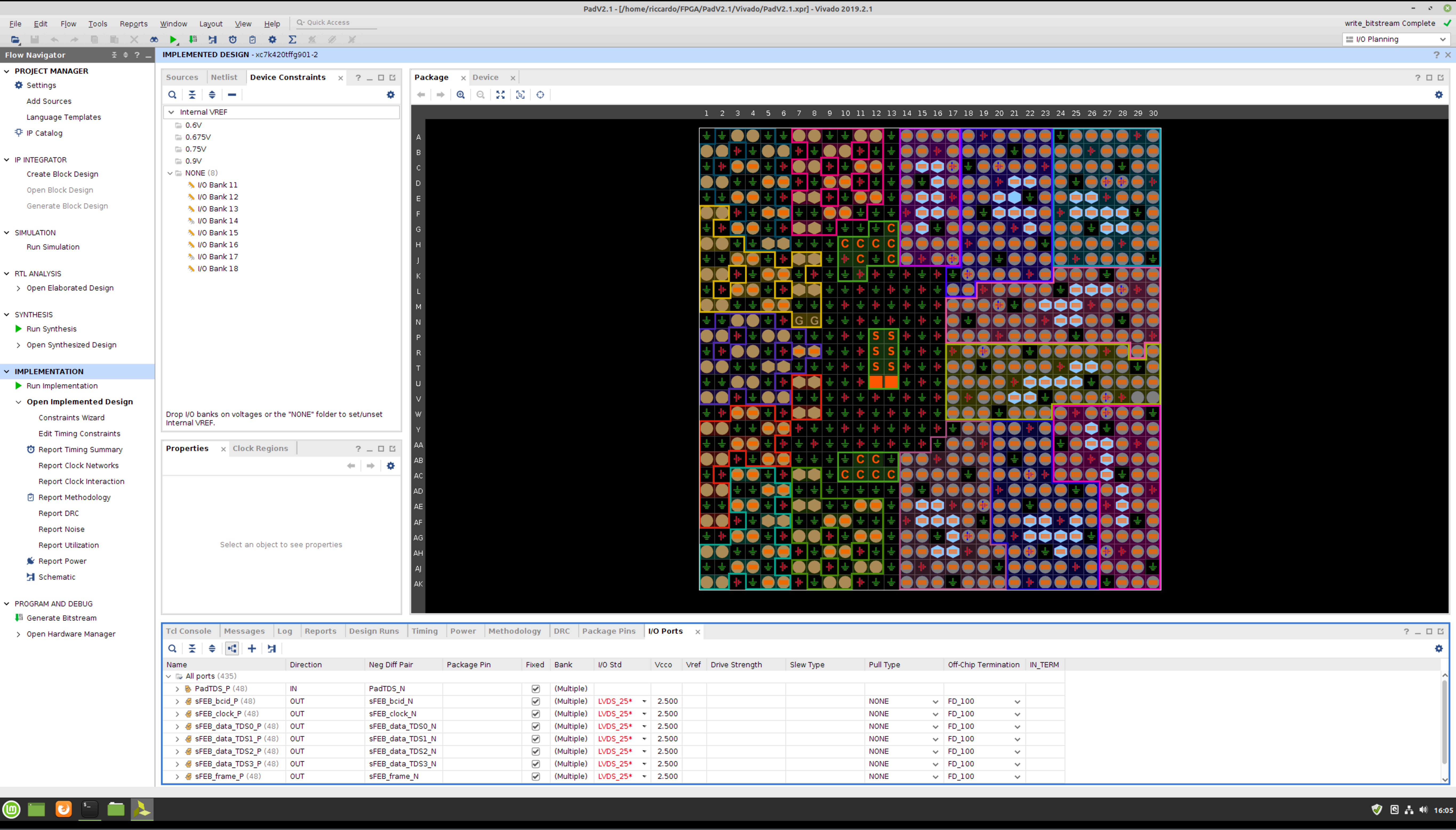
Implemented Power Report

Design Runs

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy
✓ synth_1	constrs_1	synth_design Complete!								6292	6130	0.0	0	0	4/30/20, 6:02 AM	00:02:04	Vivado Synthesis Defaults (Vivado Synthesis 2019)	Vivado Synthesis Default Reports (Vivado Synthesis 2019)
✓ impl_1	constrs_1	write_bitstream Complete!	0.496	0.000	0.028	0.000	0.000	7.026	0	29498	27742	524.5	0	0	4/30/20, 6:04 AM	00:22:27	Vivado Implementation Defaults (Vivado Implementation 2019)	Vivado Implementation Default Reports (Vivado Implementat
Out-of-Context Module Runs																		
✓ fifo_116bx16		Using cached IP results																
✓ trig_ila_ip		Using cached IP results																
✓ fifo_128bx16		Using cached IP results																
✓ shift_register_128bx128		Using cached IP results																

Implemented design I/O

- I/O signals can be automatically or manually assigned to the FPGA pins (and their voltage logic levels).



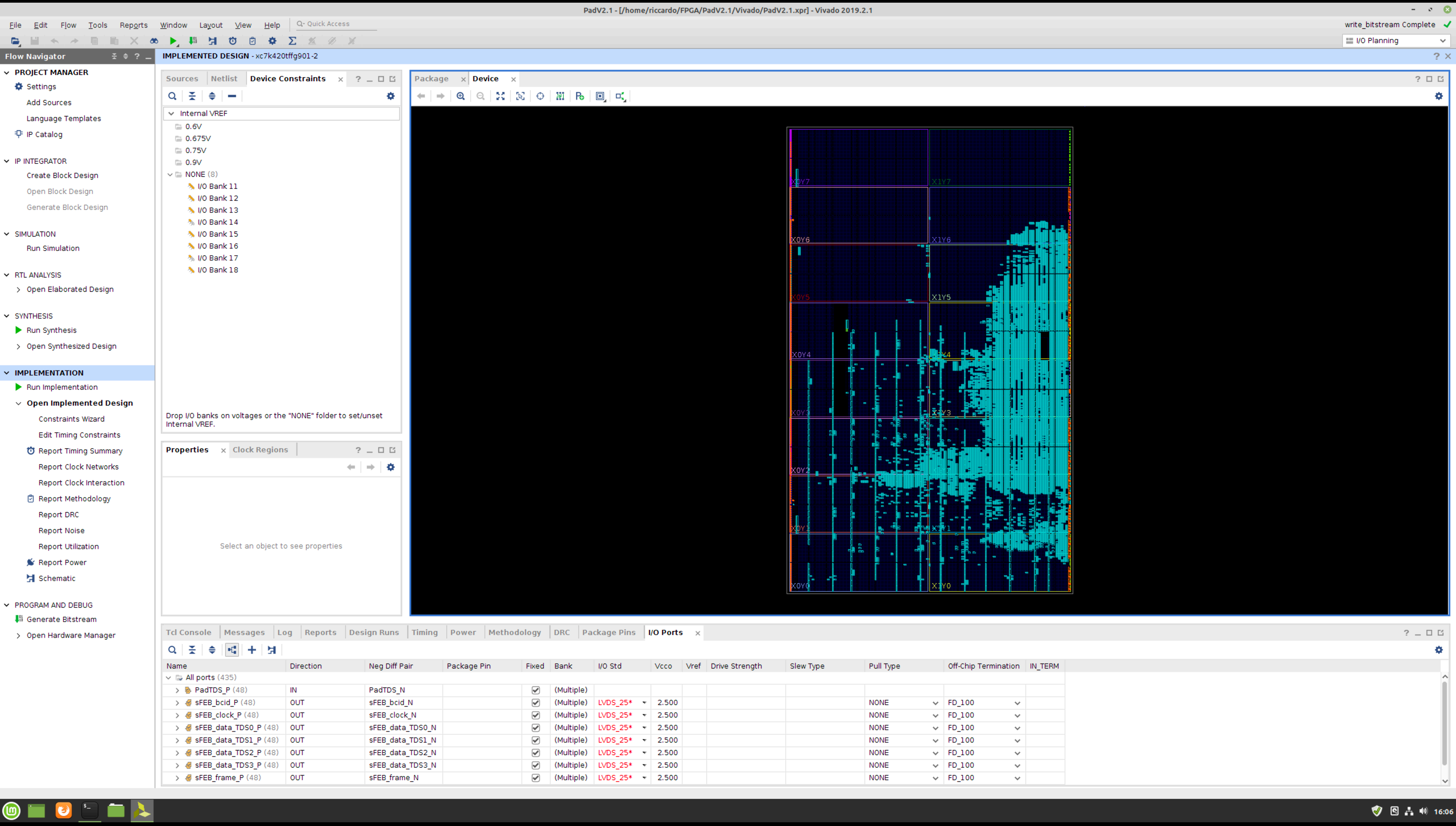
Drop I/O banks on voltages or the "NONE" folder to set/unset Internal VREF.

Select an object to see properties

Name	Direction	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength	Slew Type	Pull Type	Off-Chip Termination	IN_TERM
All ports (435)													
PadTDS_P (48)	IN	PadTDS_N		✓	(Multiple)								
sFEB_bcid_P (48)	OUT	sFEB_bcid_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_clock_P (48)	OUT	sFEB_clock_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS0_P (48)	OUT	sFEB_data_TDS0_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS1_P (48)	OUT	sFEB_data_TDS1_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS2_P (48)	OUT	sFEB_data_TDS2_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS3_P (48)	OUT	sFEB_data_TDS3_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_frame_P (48)	OUT	sFEB_frame_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	

Implemented design placement & routing view

- Detailed view of the FPGA logic cells used



PadV2.1 - [/home/riccardo/FPGA/PadV2.1/Vivado/PadV2.1.xpr] - Vivado 2019.2.1

File Edit Flow Tools Reports Window Layout View Help Quick Access

write_bitstream Complete ✓

I/O Planning

Flow Navigator

PROJECT MANAGER
- Settings
- Add Sources
- Language Templates
- IP Catalog

IP INTEGRATOR
- Create Block Design
- Open Block Design
- Generate Block Design

SIMULATION
- Run Simulation

RTL ANALYSIS
- Open Elaborated Design

SYNTHESIS
- Run Synthesis
- Open Synthesized Design

IMPLEMENTATION
- Run Implementation
- Open Implemented Design
  - Constraints Wizard
  - Edit Timing Constraints
  - Report Timing Summary
  - Report Clock Networks
  - Report Clock Interaction
  - Report Methodology
  - Report DRC
  - Report Noise
  - Report Utilization
  - Report Power
  - Schematic

PROGRAM AND DEBUG
- Generate Bitstream
- Open Hardware Manager

IMPLEMENTED DESIGN - xc7k420tffg901-2

Sources | Netlist | Device Constraints

- Internal VREF
  - 0.6V
  - 0.675V
  - 0.75V
  - 0.9V
  - NONE (8)
    - I/O Bank 11
    - I/O Bank 12
    - I/O Bank 13
    - I/O Bank 14
    - I/O Bank 15
    - I/O Bank 16
    - I/O Bank 17
    - I/O Bank 18

Drop I/O banks on voltages or the "NONE" folder to set/unset Internal VREF.

Properties | Clock Regions

Select an object to see properties

Package | Device

Tcl Console | Messages | Log | Reports | Design Runs | Timing | Power | Methodology | DRC | Package Pins | I/O Ports

Name	Direction	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength	Slew Type	Pull Type	Off-Chip Termination	IN_TERM
All ports (435)													
PadTDS_P (48)	IN	PadTDS_N		✓	(Multiple)								
sFEB_bcid_P (48)	OUT	sFEB_bcid_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_clock_P (48)	OUT	sFEB_clock_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS0_P (48)	OUT	sFEB_data_TDS0_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS1_P (48)	OUT	sFEB_data_TDS1_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS2_P (48)	OUT	sFEB_data_TDS2_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_data_TDS3_P (48)	OUT	sFEB_data_TDS3_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	
sFEB_frame_P (48)	OUT	sFEB_frame_N		✓	(Multiple)	LVDS_25*	2.500				NONE	FD_100	

16:06

Firmware file generation and FPGA programming

- If FPGA implementation was successful we can proceed with the firmware file generation
- Firmware file contains the FPGA programmable logic implementation represented by the used logic
- At power-up the FPGA does not have any logic, until the firmware file is uploaded into the FPGA configuration memory
- Firmware file can be uploaded from a computer via USB/JTAG or from an external flash memory at boot-up.

Fine!