IN2P3 FPGA System Planner

Journées VLSI 2016

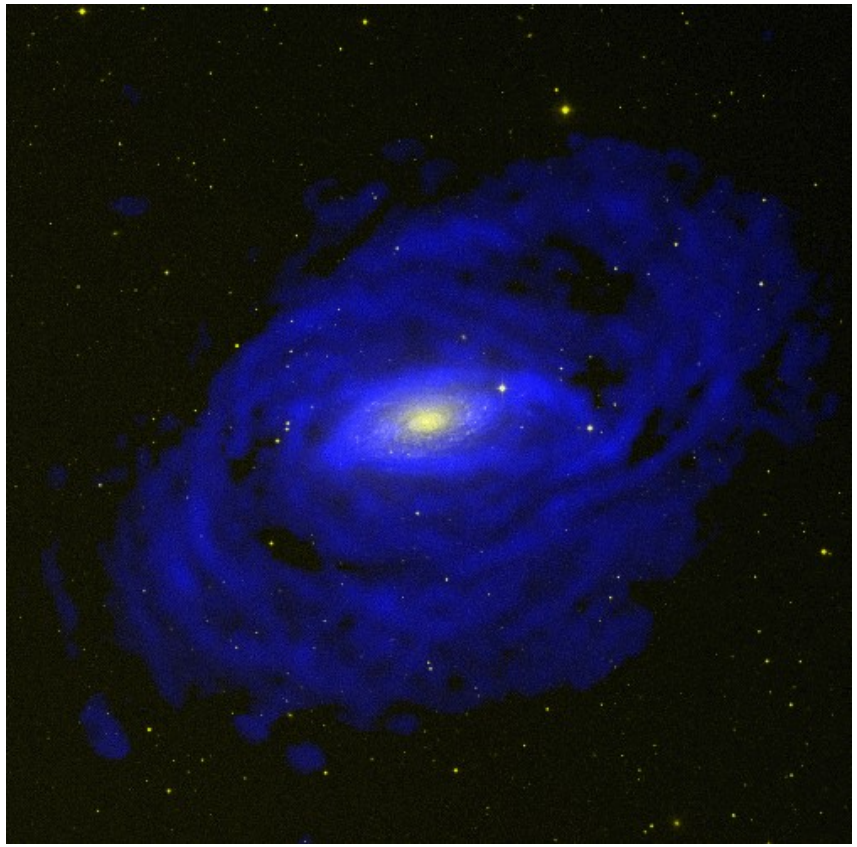
NEBULA DESIGN

Agenda

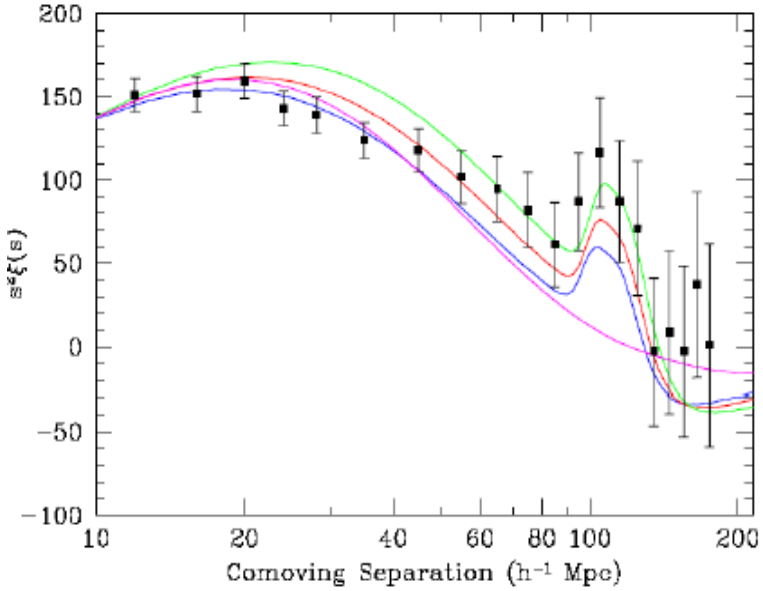
- PAON IV project
- NEBULA board
- IN2P3's old design methodology
- Overview of FSP
- IN2P3's methodology using FSP
- Improvements seen by using FSP

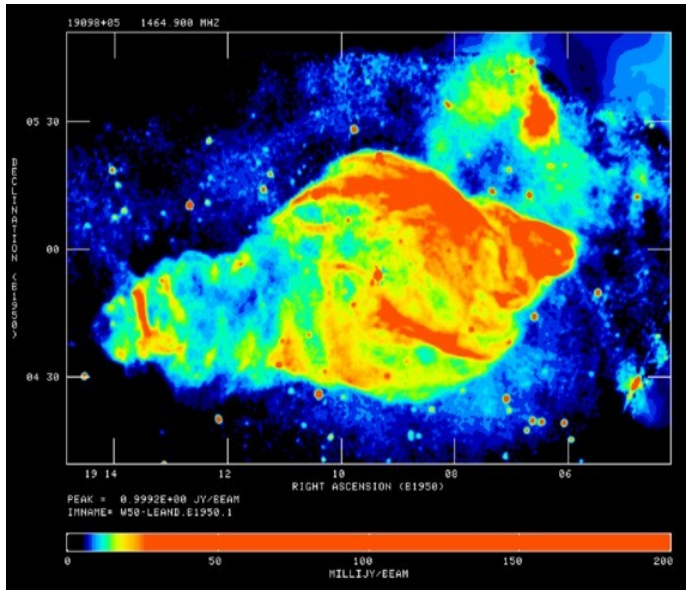
BAO project in radio

Fast acquisition system for 3D mapping of cosmological matter distribution in radio



BAO : Baryonic Acoustic Oscillations



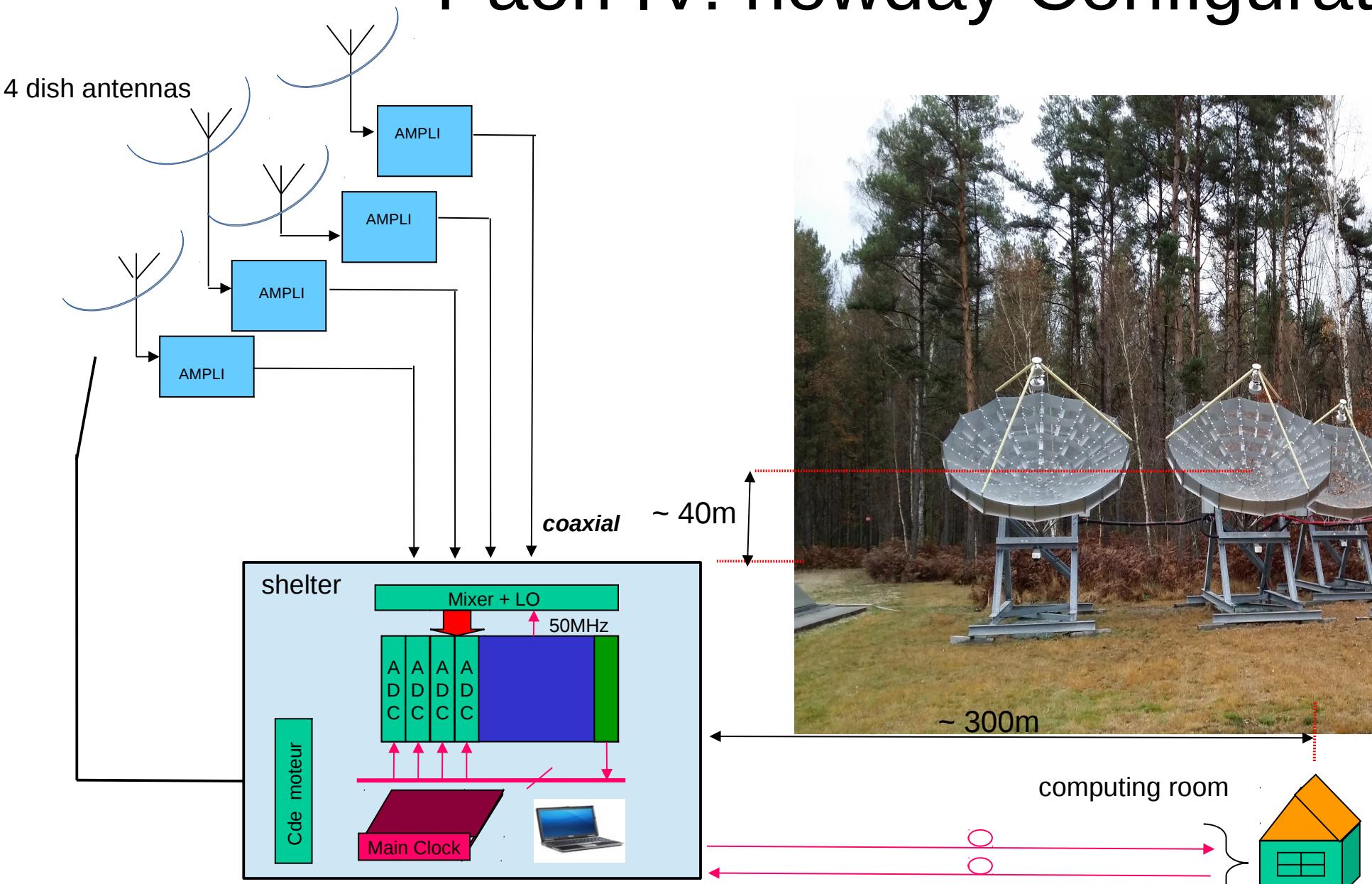


- Imprints left by the baryon-photon fluid (before recombination) in the distribution of ordinary (baryonic) matter
- Slight modulation of the distribution of matter, (and galaxies as tracers). Structure formation being mainly driven by CDM which dominates structure formation
- In Radio : Use 21 cm HI emission
- 3D HI mass distribution measurement through total 21 cm emission intensity mapping (No individual galaxy detection)
- Hyperfine transition (spin-orbit) of atomic hydrogen: $\nu \approx 1{,}420405$ GHz $\rightarrow \lambda \approx 21$ cm

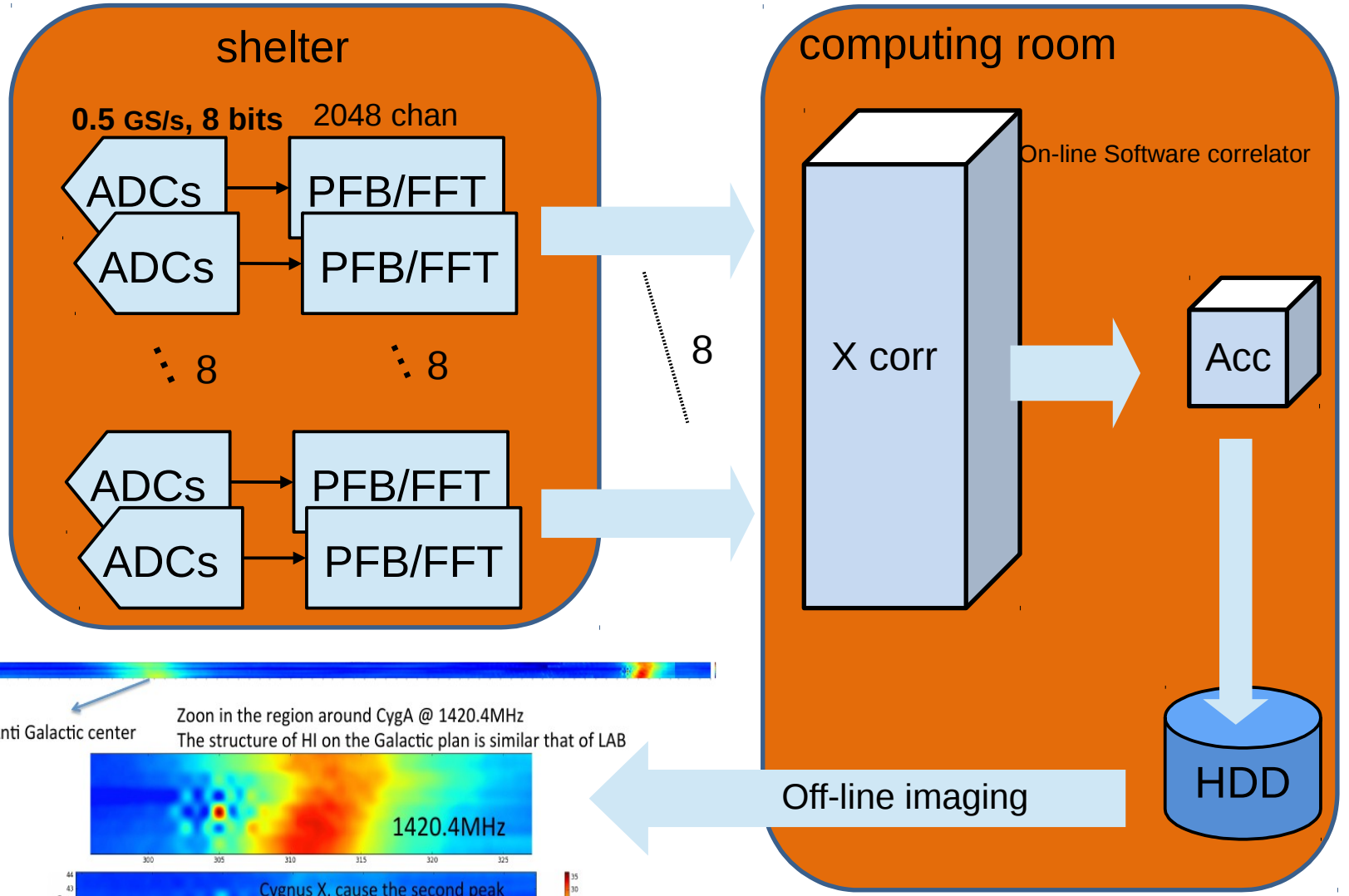
LABORATOIRE DE L'ACCÉLÉRATEUR LINÉAIRE

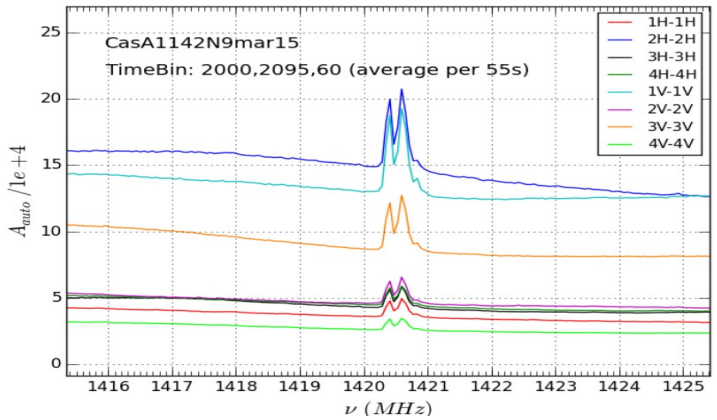
cnrs dépasser les frontières

Paon IV: nowday Configuration.

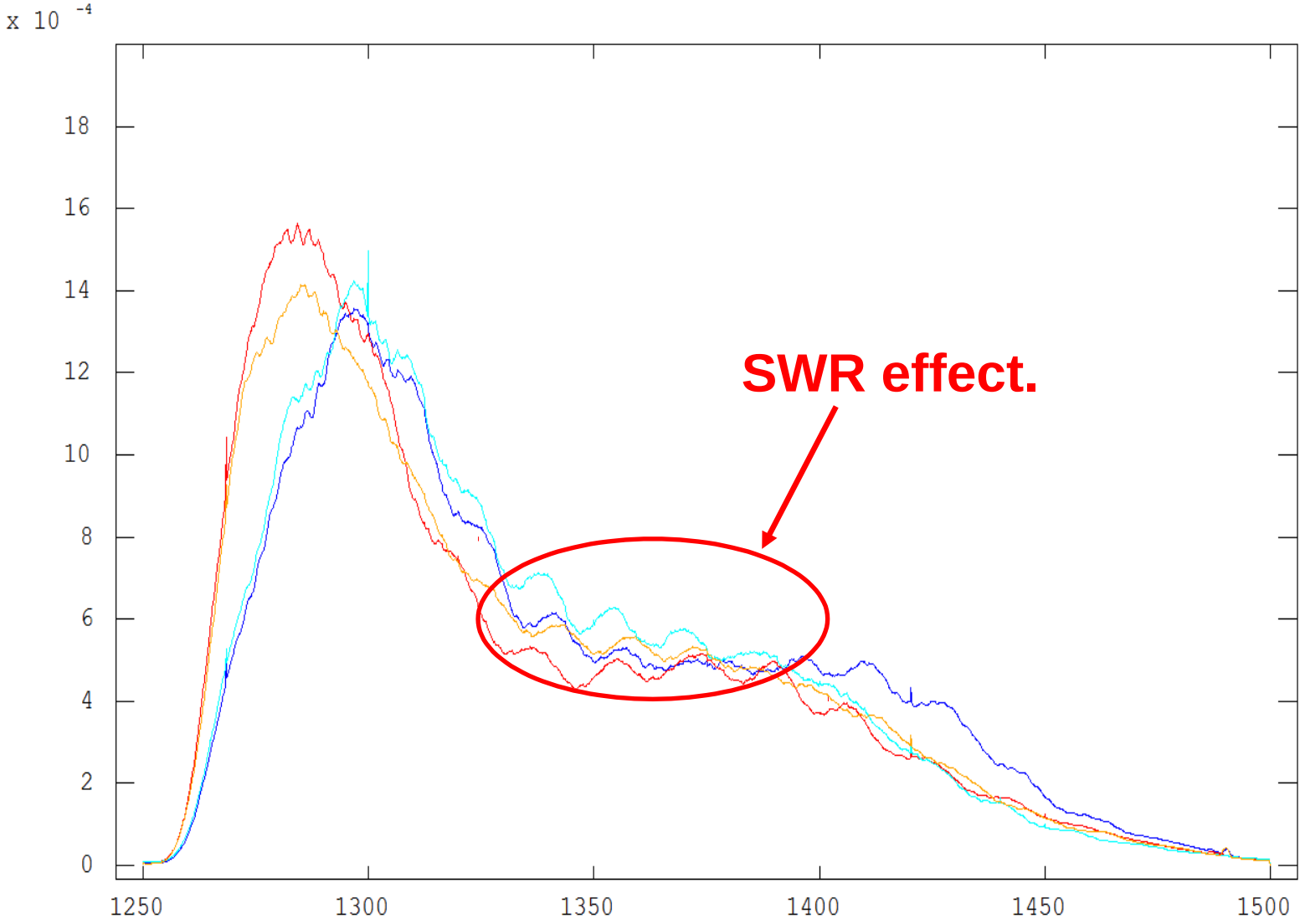


Data processing for off-line beam forming





Paon IV current analog limitation

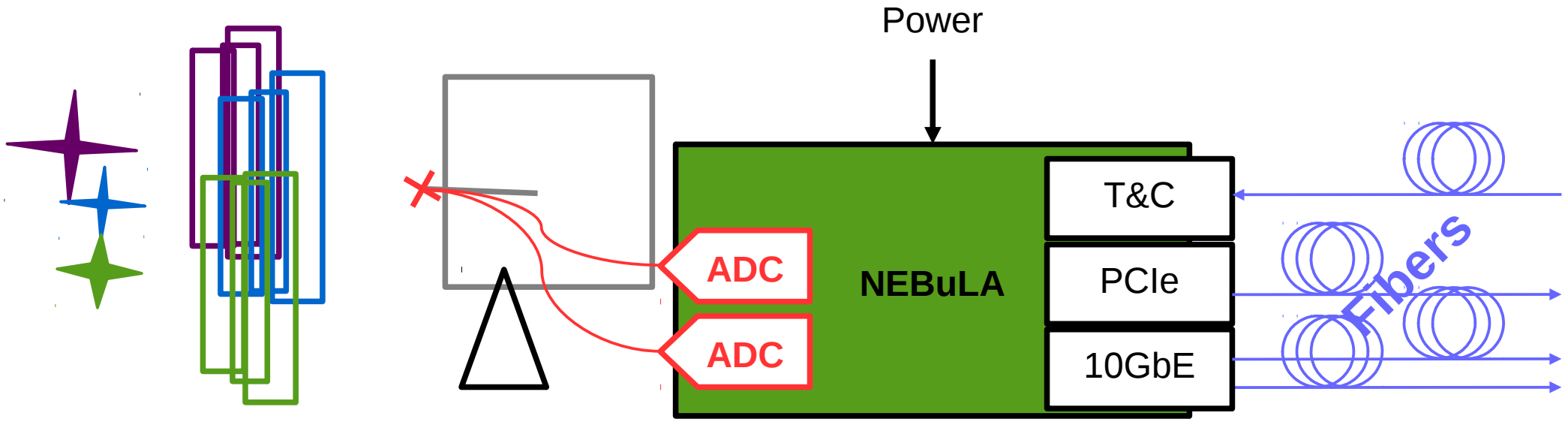


SWR depends of:

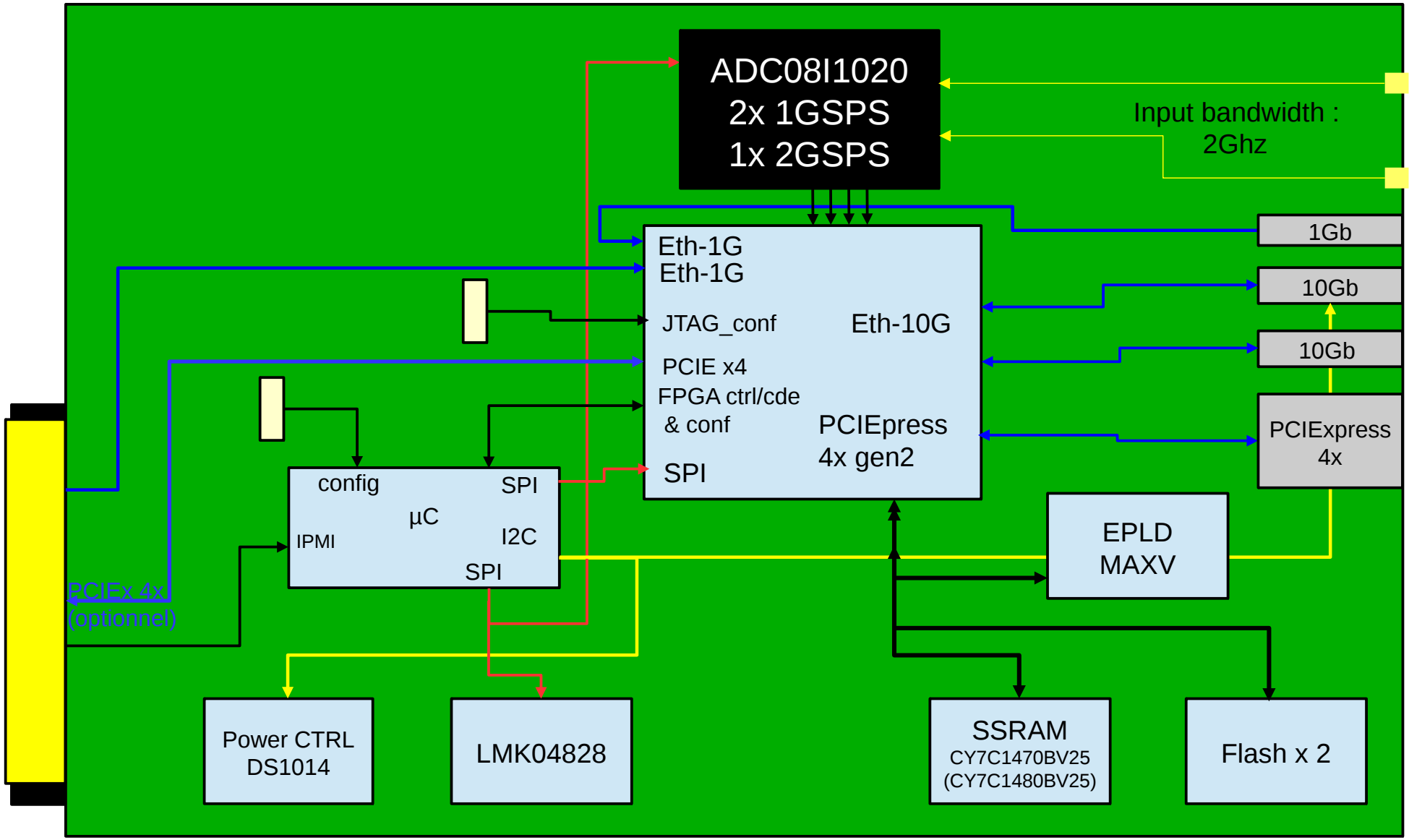
- long RF cables,
- adaptation mismatch,
- wide band acquisition.

NEBULA

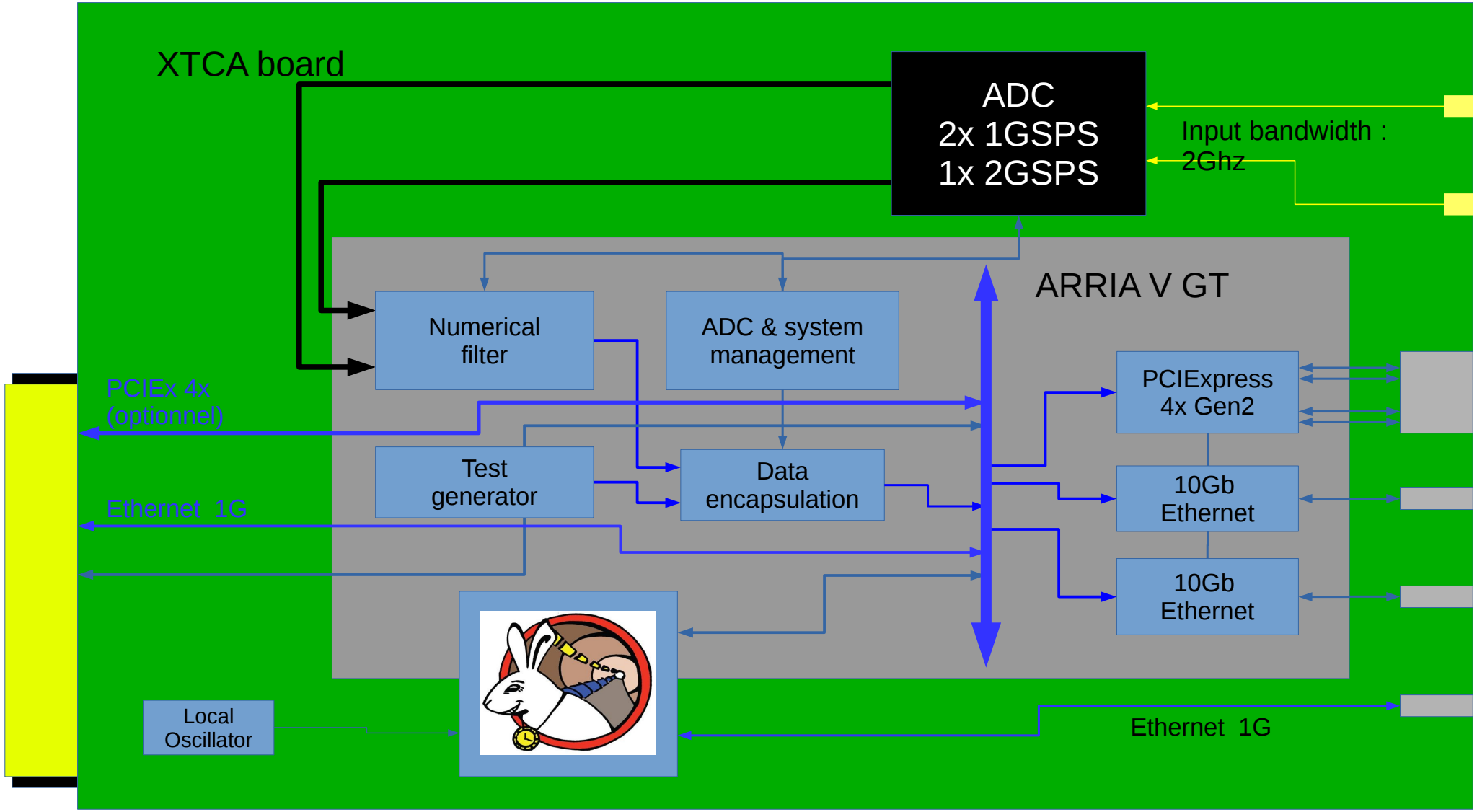
NumEriseur à Bande Large pour l'Astronomie



Nebula Design Architecture

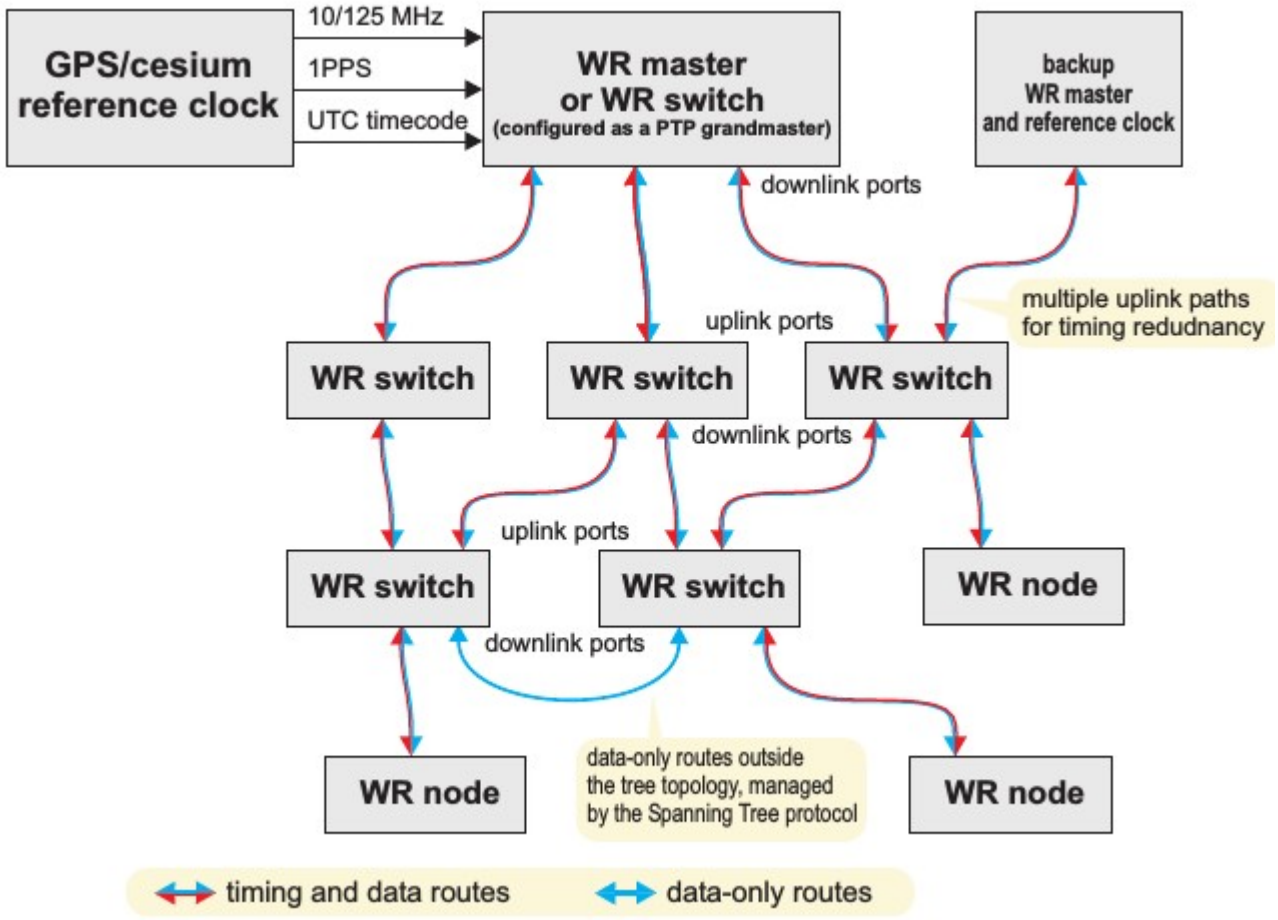


NEBULA FPGA content



White Rabbit

Network topology



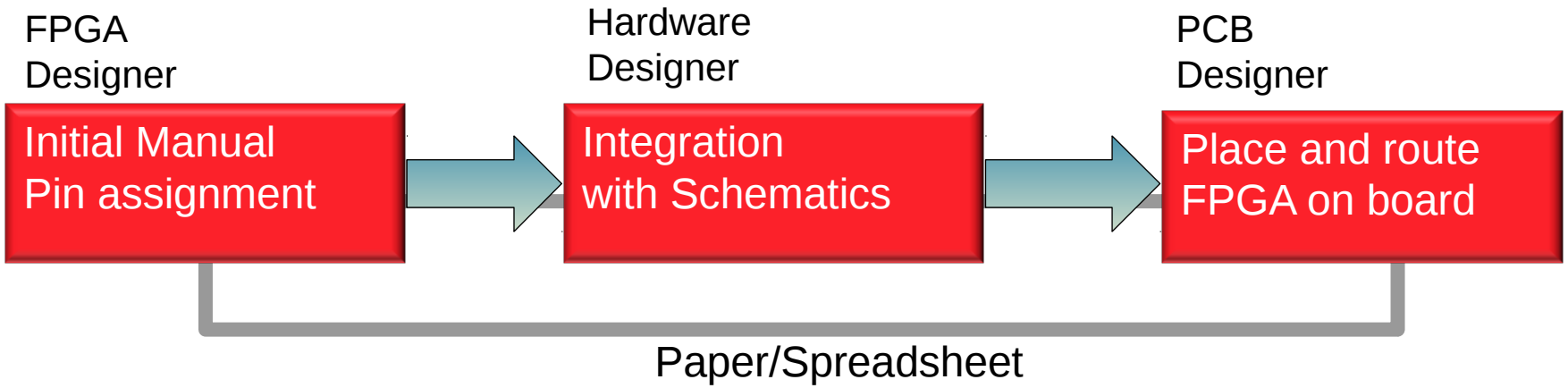
Extension of Ethernet

- Synchronous mode (Synch-E) – common clock for physical layer in entire network, allowing for precise time transfer.
- Deterministic routing latency

Technical concept

- Synchronous Ethernet
- Hardware -assisted PTP (IEEE1588) – Precision Time Protocol
- Packed preemption ans deterministic protocol

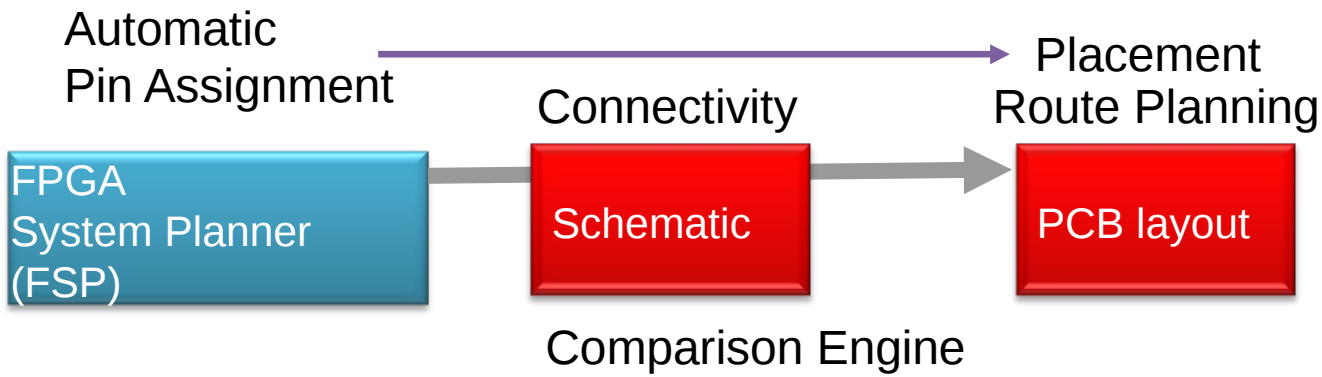
IN2P3's old design methodology

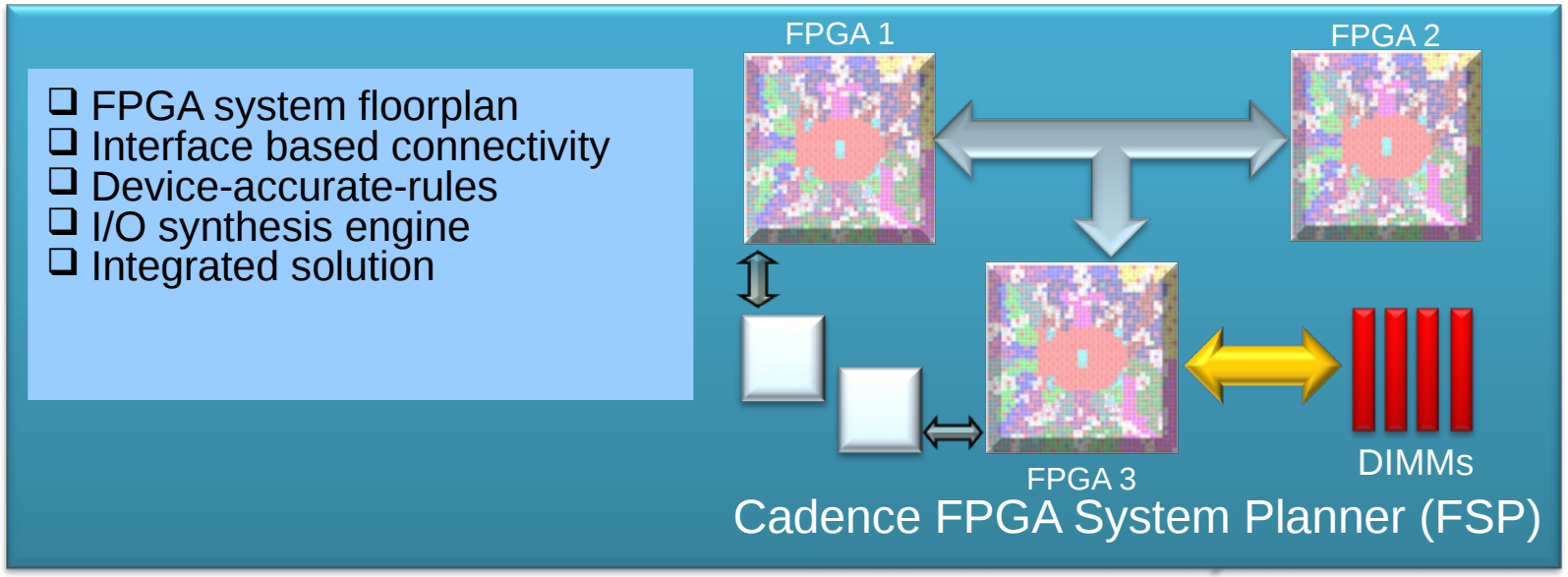


TOOLS	• Build Quartus FPGA in order to export .pin file	• Part developer librarian • Import .pin file • Schematic design tools	• PCB design tools
PAIN	Manual pin assignment ❑ one FPGA at a time ❑ Redo pin assignment with every ECO	❑ Place symbols and connect them ❑ Redo with each ECO	❑ No swapping possible

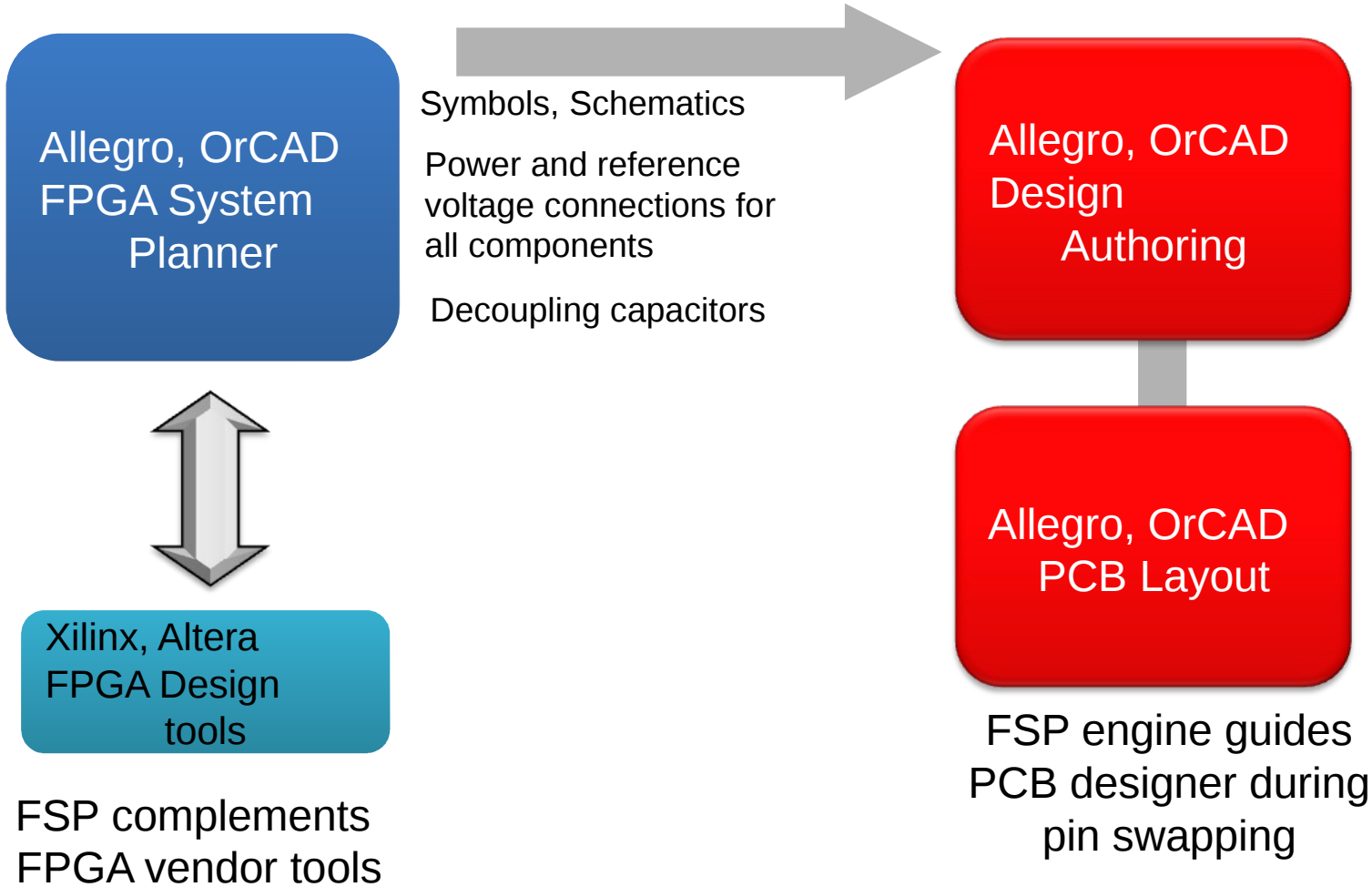
Overview of FSP :

Three Main Components of FPGA System Planner

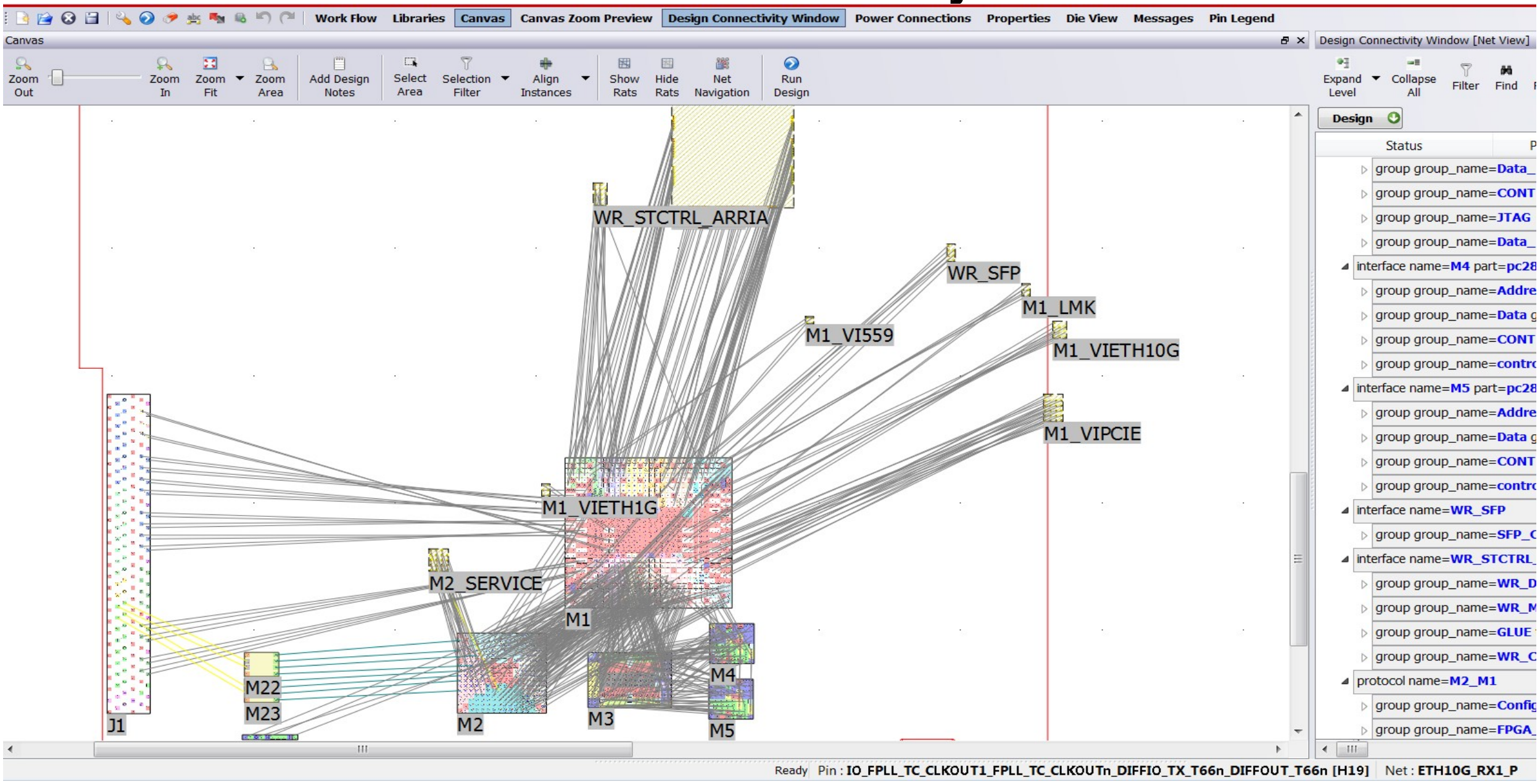




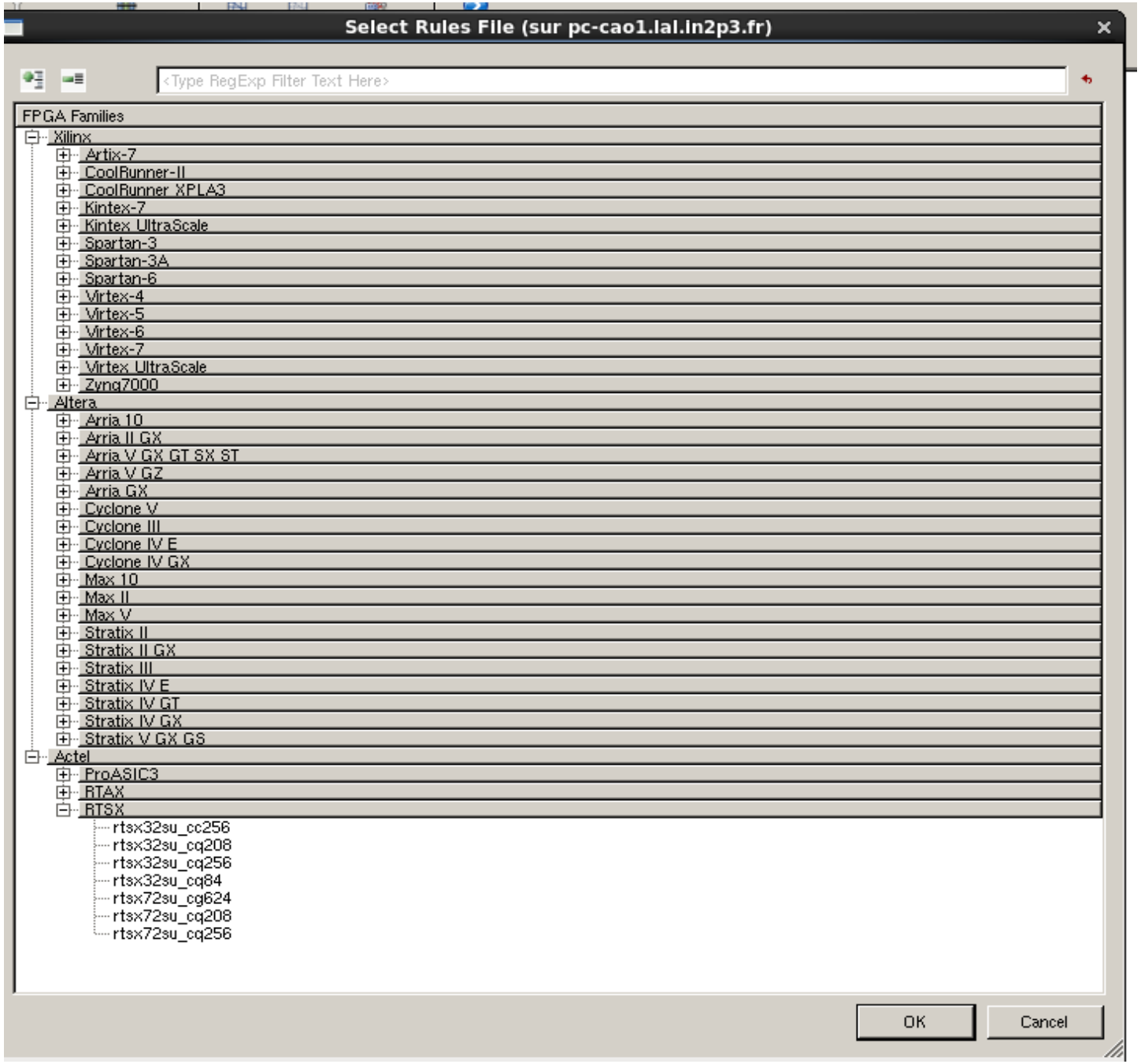
FSP Flow


Allegro, OrCAD FPGA System Planner
Symbols, Schematics
Power and reference voltage connections for all components
Decoupling capacitors
Allegro, OrCAD Design Authoring
Allegro, OrCAD PCB Layout
Xilinx, Altera FPGA Design tools
FSP complements FPGA vendor tools
FSP engine guides PCB designer during pin swapping

Nebula FSP Project



FPGA selection
FSP Project



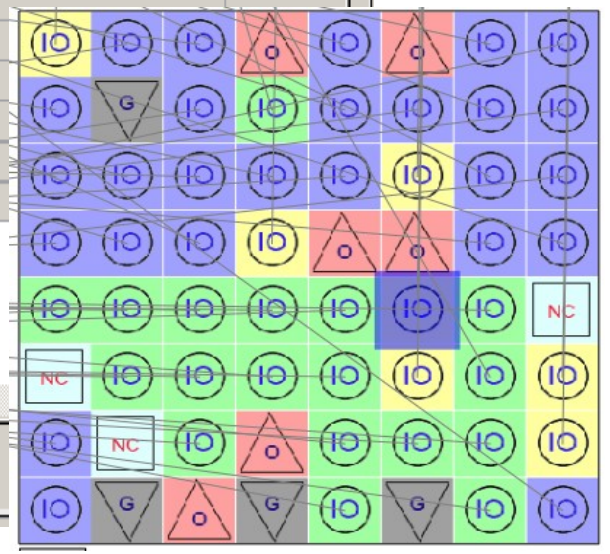
Component definition 1 FSP Project

Rules Viewer (pc28f00ap30bf) (sur laob.lal.in2p3.fr)

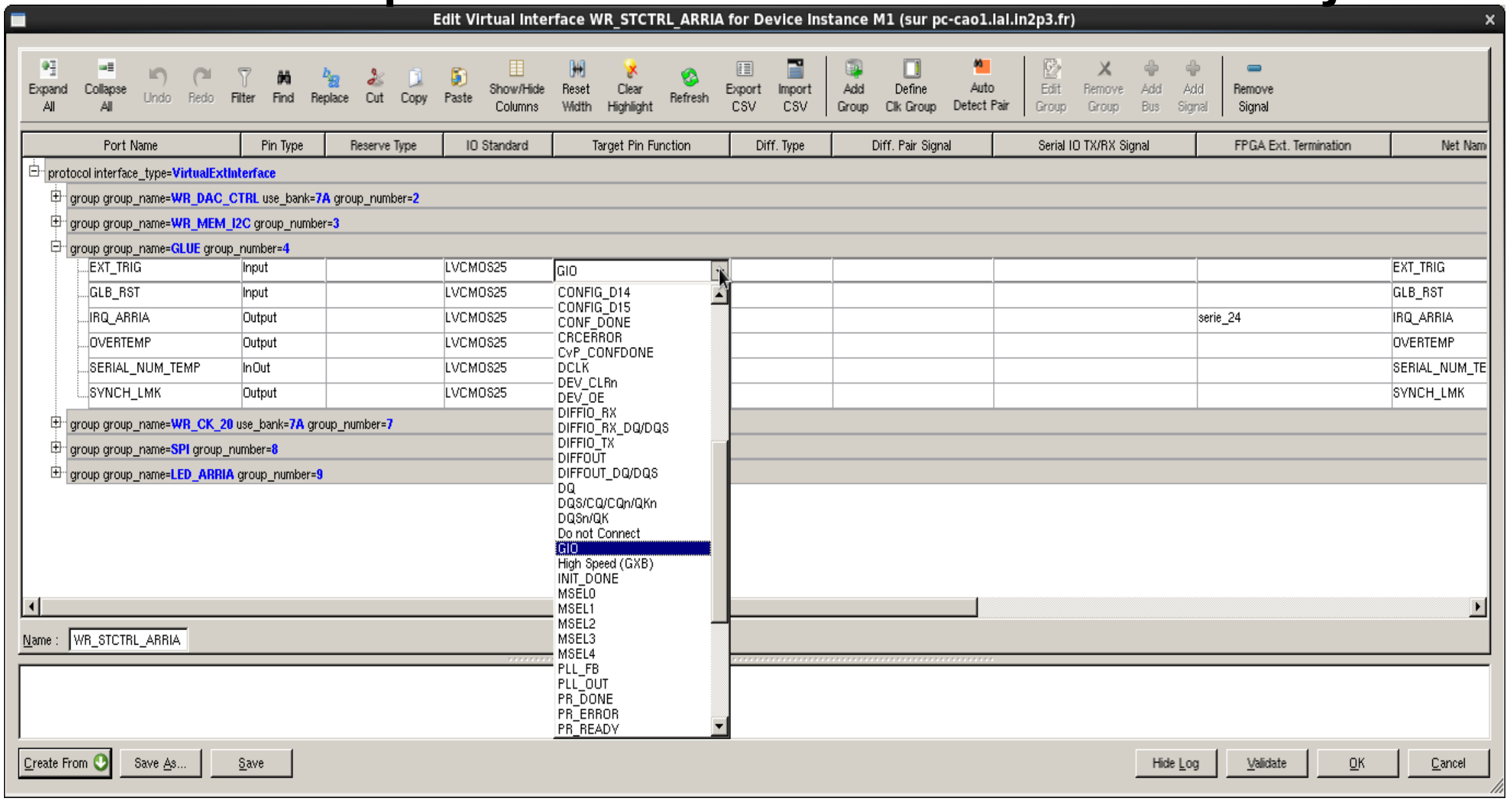
Expand All | Collapse All | Undo | Redo | Filter | Find | Replace | Cut | Copy | Paste | Show/Hide Columns | Reset Width | Clear Highlight | Refresh | Export CSV | Import CSV

Pin Name	Pin Number	Pin Type	Symbol Pin Name	Voltage Level	IO Standard	Target Pin Function	Diff. Type	Diff. Pair Pin	Serial IO TX/RX Pin	Clock Grou
lrf interface_type=NormalInterface target_family=ArriaVGX,MaxV unit=mm pin_size=1.00000 jedec_type=bga64h8_1000 instance_prefix=M pdescription=Parrallel%20NOR%20flash%20memory part_height=7000000 part_width=7000000 x_offset=-0 y_offset=7										
group group_name=power group_number=4 group_color=#ff0000 gdescription=vccq%20vccp%20vcc										
group group_name=gnd group_number=5 group_color=#000000 gdescription=gnd%20vss										
group group_name=Address group_constraint=same_bank group_number=6 group_color=#0000ff gdescription=lvcmos2.5v%20to%203.3v										
group group_name=Data group_constraint=same_bank group_number=7 group_color=#00ff00 gdescription=lvcmos%202.5v%20to%203.3v										
group group_name=CONTROL group_constraint=same_bank group_number=8 group_color=#ffff00 gdescription=lvcmos%202.5v%20to%203.3v										
ADD<1>	A1	Input	ADD<1>		LVCMOS25	GIO				
ADVN	F6	Input	ADVN		LVCMOS25	GIO				
CLK	E6	Input	CLK		LVCMOS25	GIO				
OEN	F8	Input	OEN		LVCMOS25	GIO				
RESETN	D4	Input	RESETN		LVCMOS25	GIO				
WEN	G8	Input	WEN		LVCMOS25	GIO				
WPN	C6	Input	WPN		LVCMOS25	GIO				
group group_name=NC group_number=9 group_color=#aaffff gdescription=NO%20CONNECT										
group group_name=control_2 group_number=10 group_color=#00ff00 gdescription=CONTROL_2										

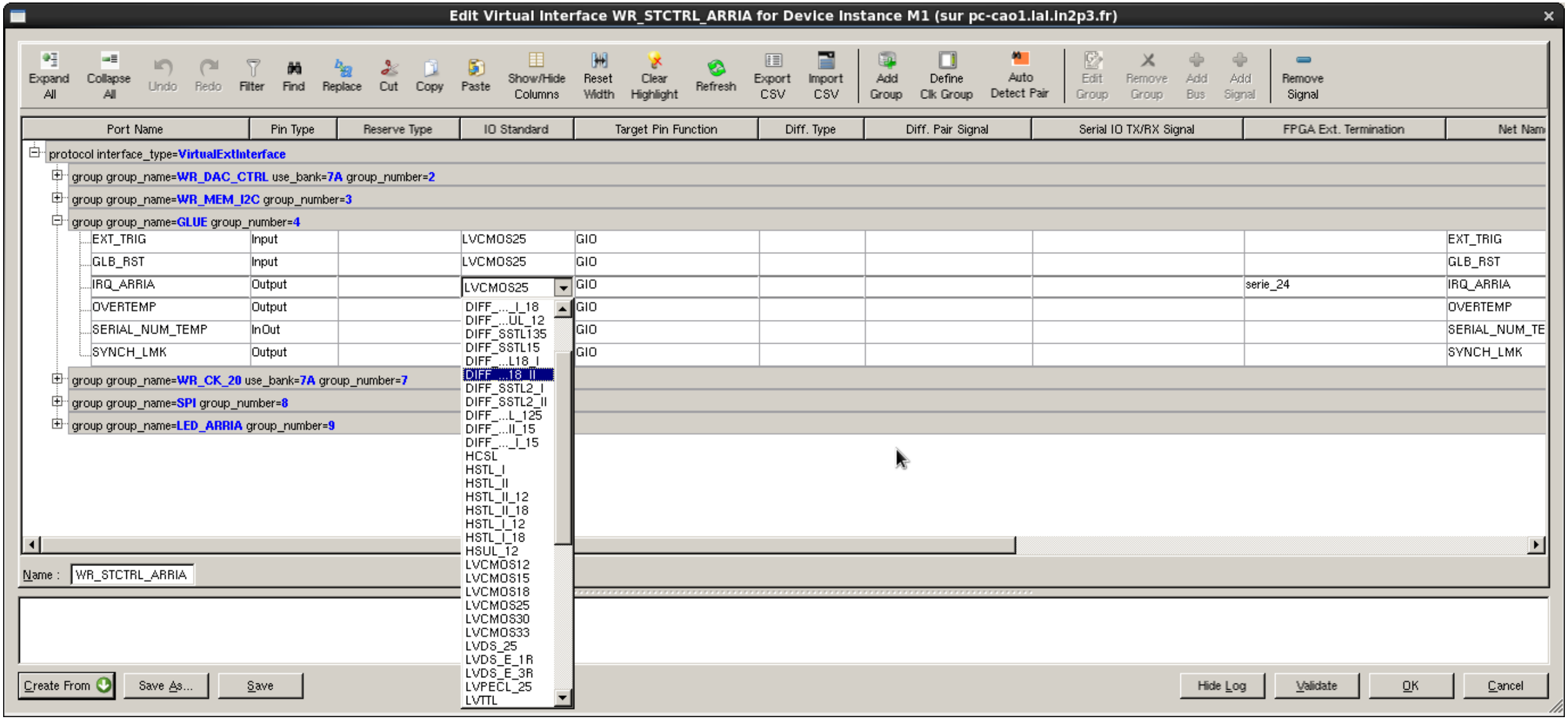
Show Package View



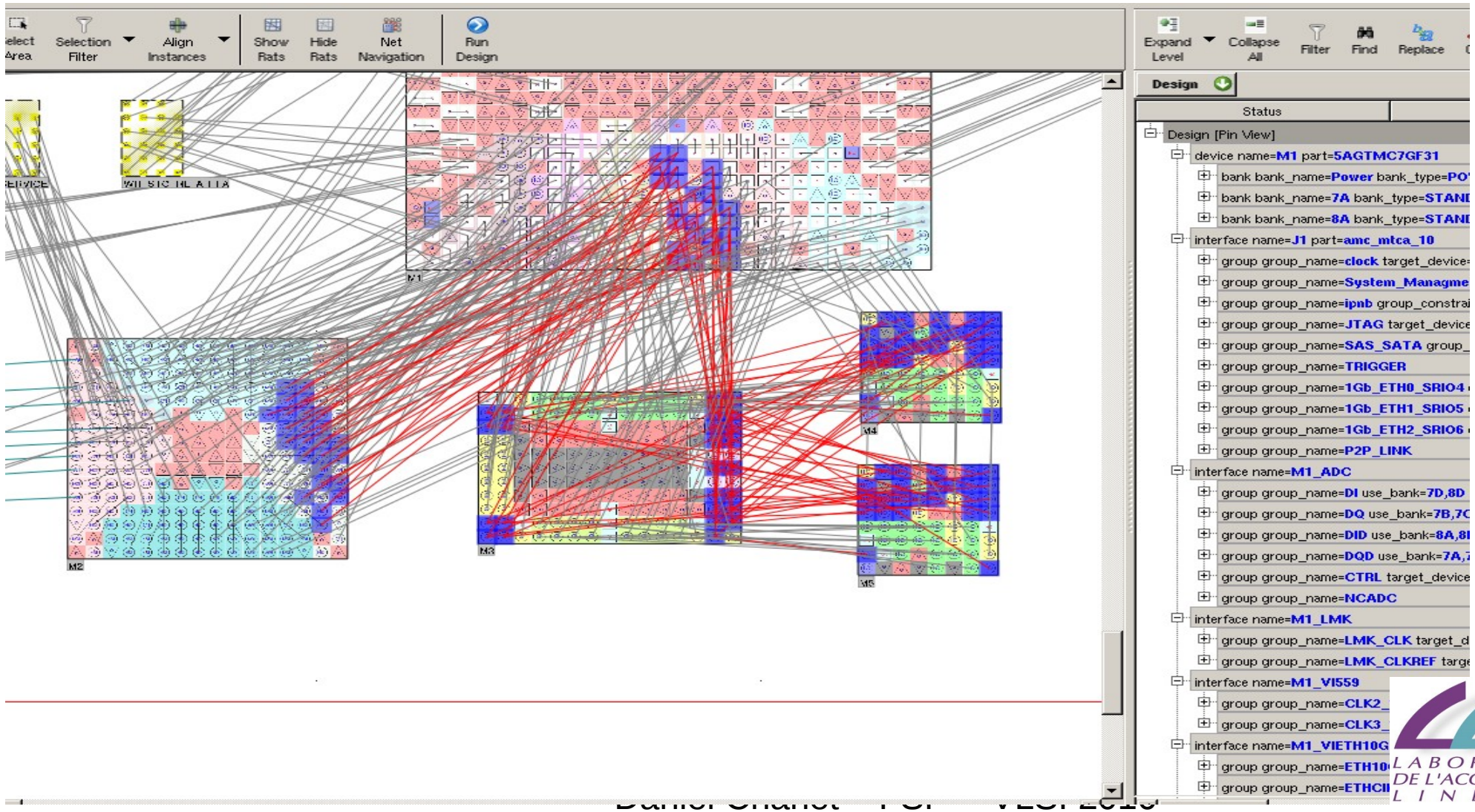
Component definition 2 FSP Project



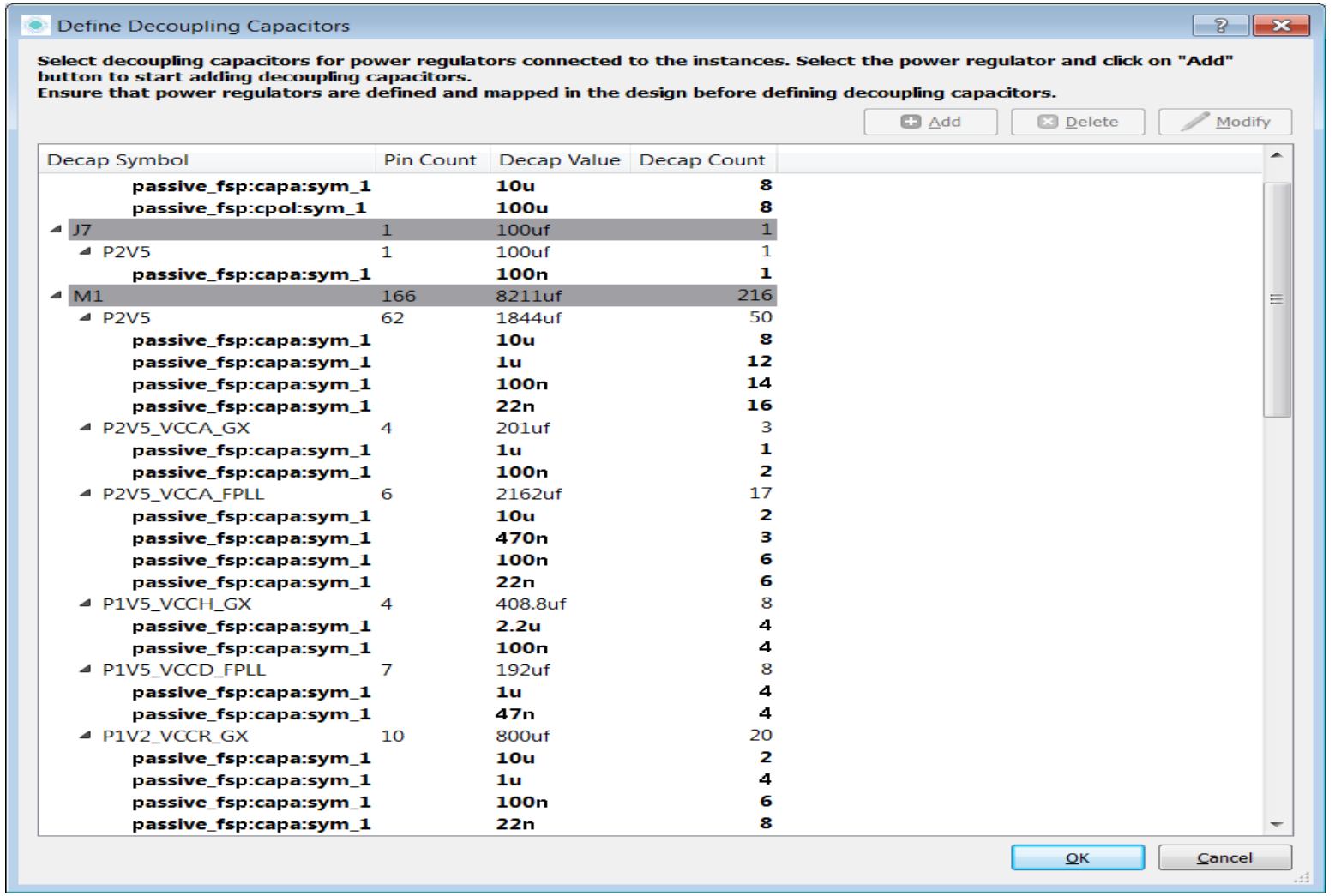
Component definition 3 FSP Project



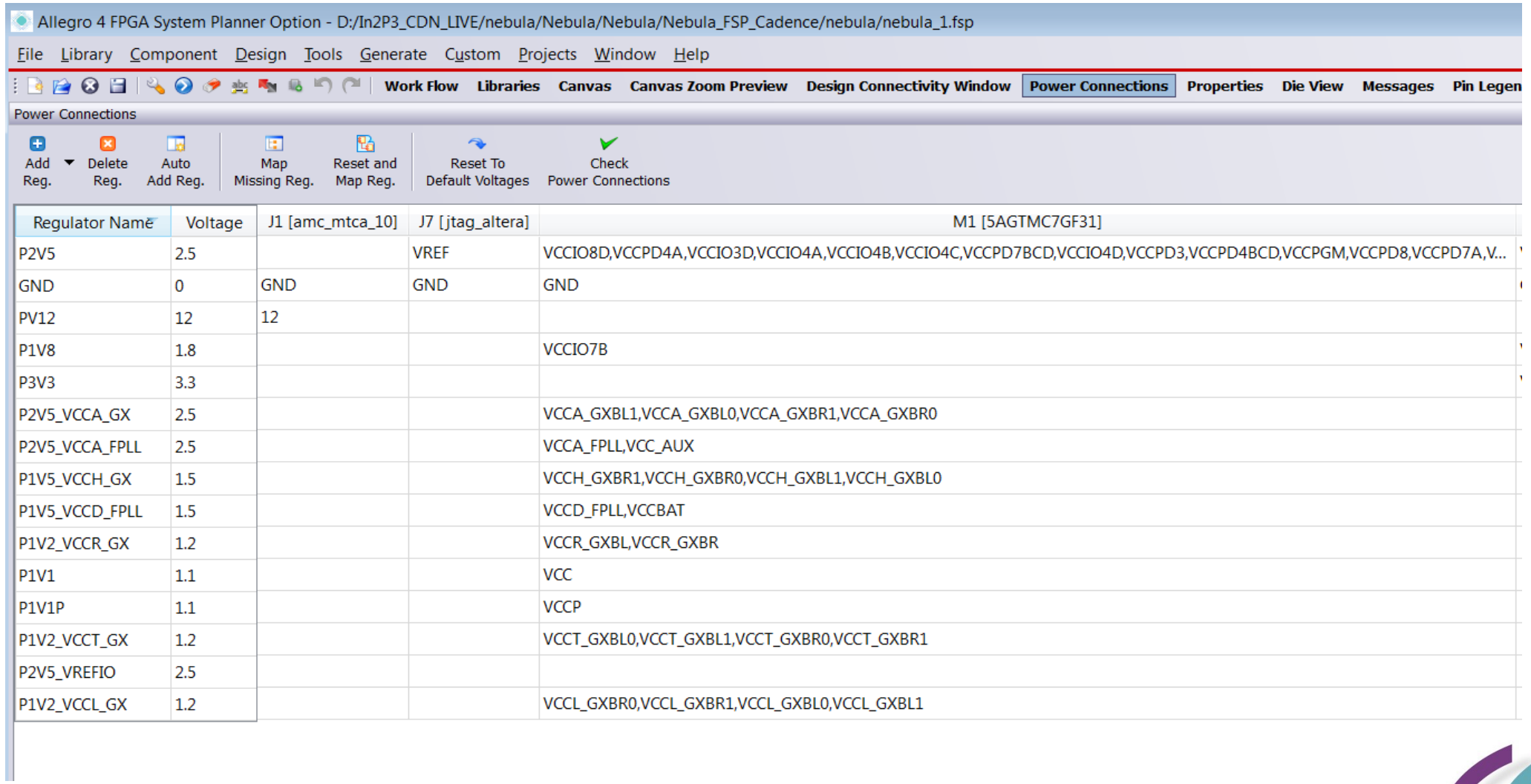
Address Bus Optimized and Assigned by FSP *(In Red the Bus Data_lsb)*



Decoupling assignment and associated to the chips

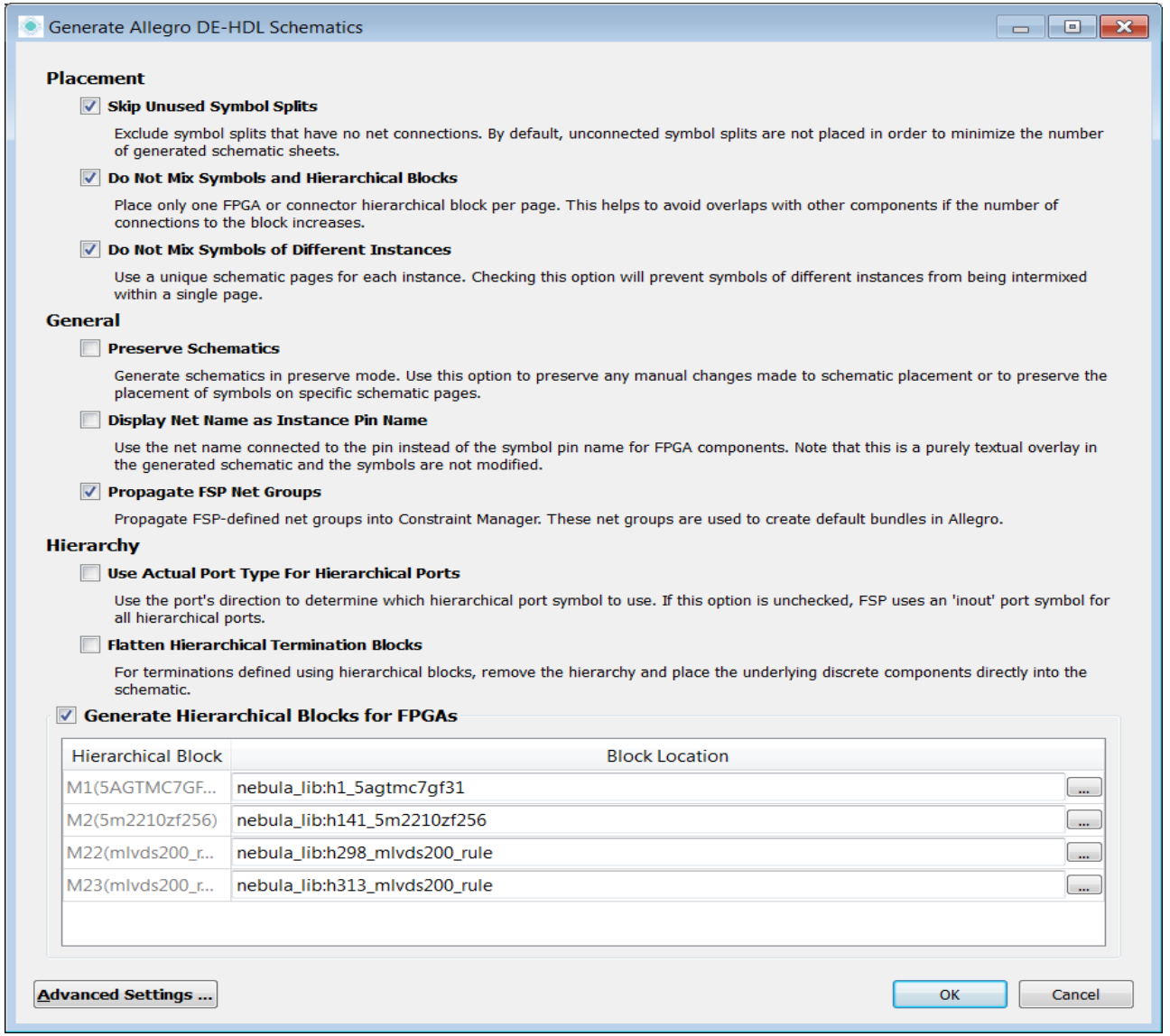


Power assignment

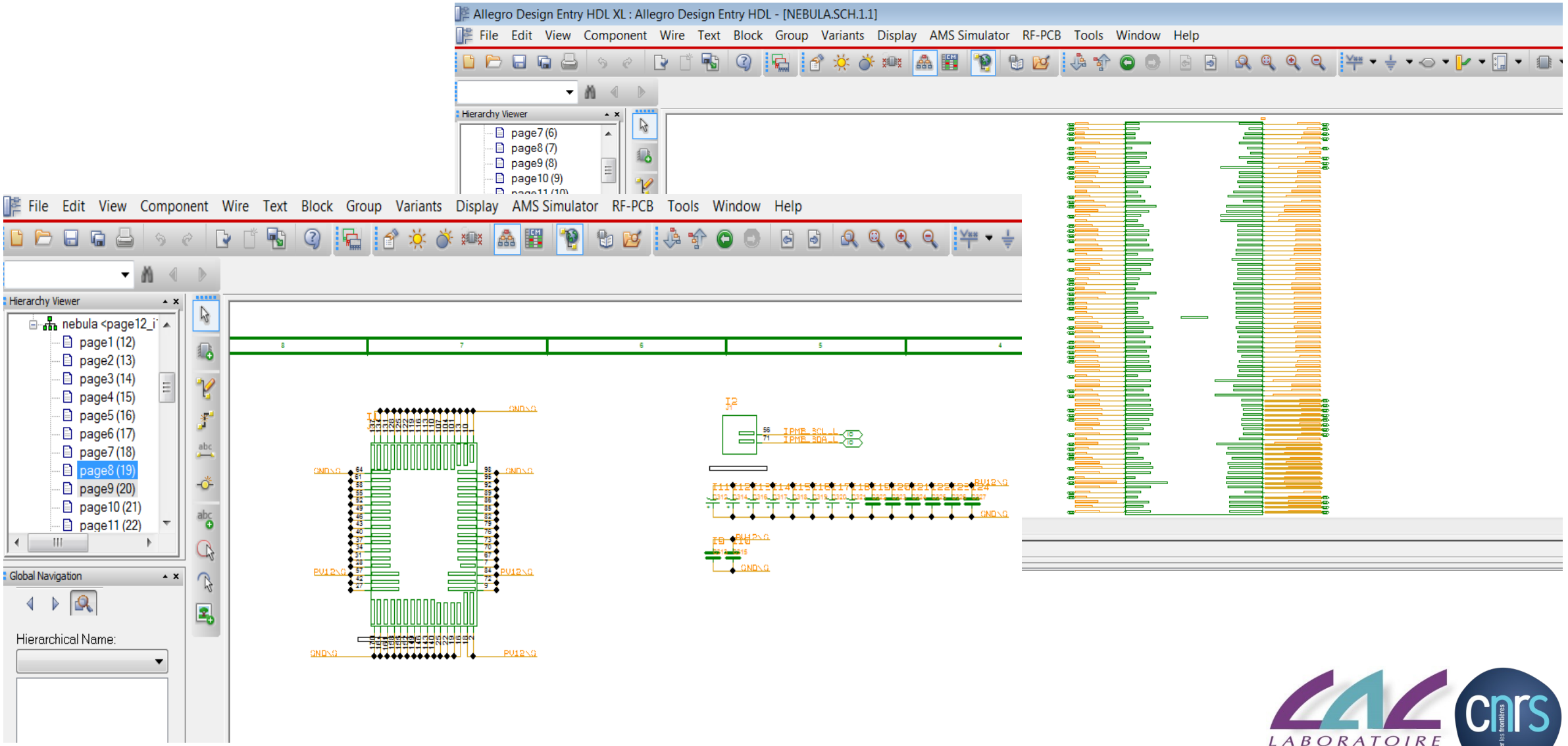


Regulator Name	Voltage	J1 [amc_mtca_10]	J7 [jtag_altera]	M1 [5AGTMC7GF31]
P2V5	2.5		VREF	VCCIO8D,VCCPD4A,VCCIO3D,VCCIO4A,VCCIO4B,VCCIO4C,VCCPD7BCD,VCCIO4D,VCCPD3,VCCPD4BCD,VCCPGM,VCCPD8,VCCPD7A,V...
GND	0	GND	GND	GND
PV12	12	12		
P1V8	1.8			VCCIO7B
P3V3	3.3			
P2V5_VCCA_GX	2.5			VCCA_GXBL1,VCCA_GXBL0,VCCA_GXBR1,VCCA_GXBR0
P2V5_VCCA_FPLL	2.5			VCCA_FPLL,VCC_AUX
P1V5_VCCH_GX	1.5			VCCH_GXBR1,VCCH_GXBR0,VCCH_GXBL1,VCCH_GXBL0
P1V5_VCCD_FPLL	1.5			VCCD_FPLL,VCCBAT
P1V2_VCCR_GX	1.2			VCCR_GXBL,VCCR_GXBR
P1V1	1.1			VCC
P1V1P	1.1			VCCP
P1V2_VCCT_GX	1.2			VCCT_GXBL0,VCCT_GXBL1,VCCT_GXBR0,VCCT_GXBR1
P2V5_VREFIO	2.5			
P1V2_VCCL_GX	1.2			VCCL_GXBR0,VCCL_GXBR1,VCCL_GXBL0,VCCL_GXBL1

Automatic Schematic Generation



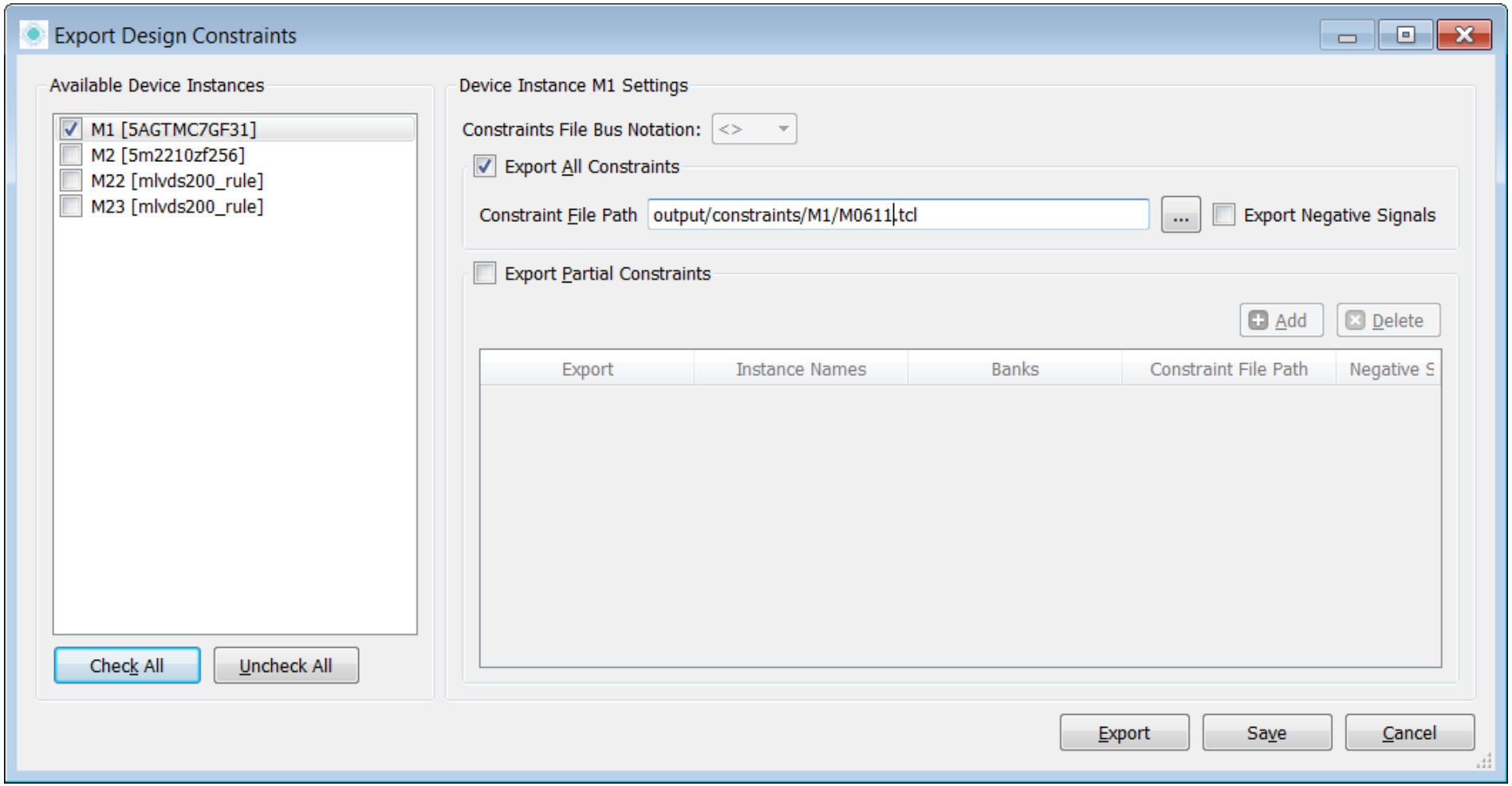
Output result from Generate design



LABORATOIRE DE L'ACCÉLÉRATEUR LINÉAIRE

cnrs

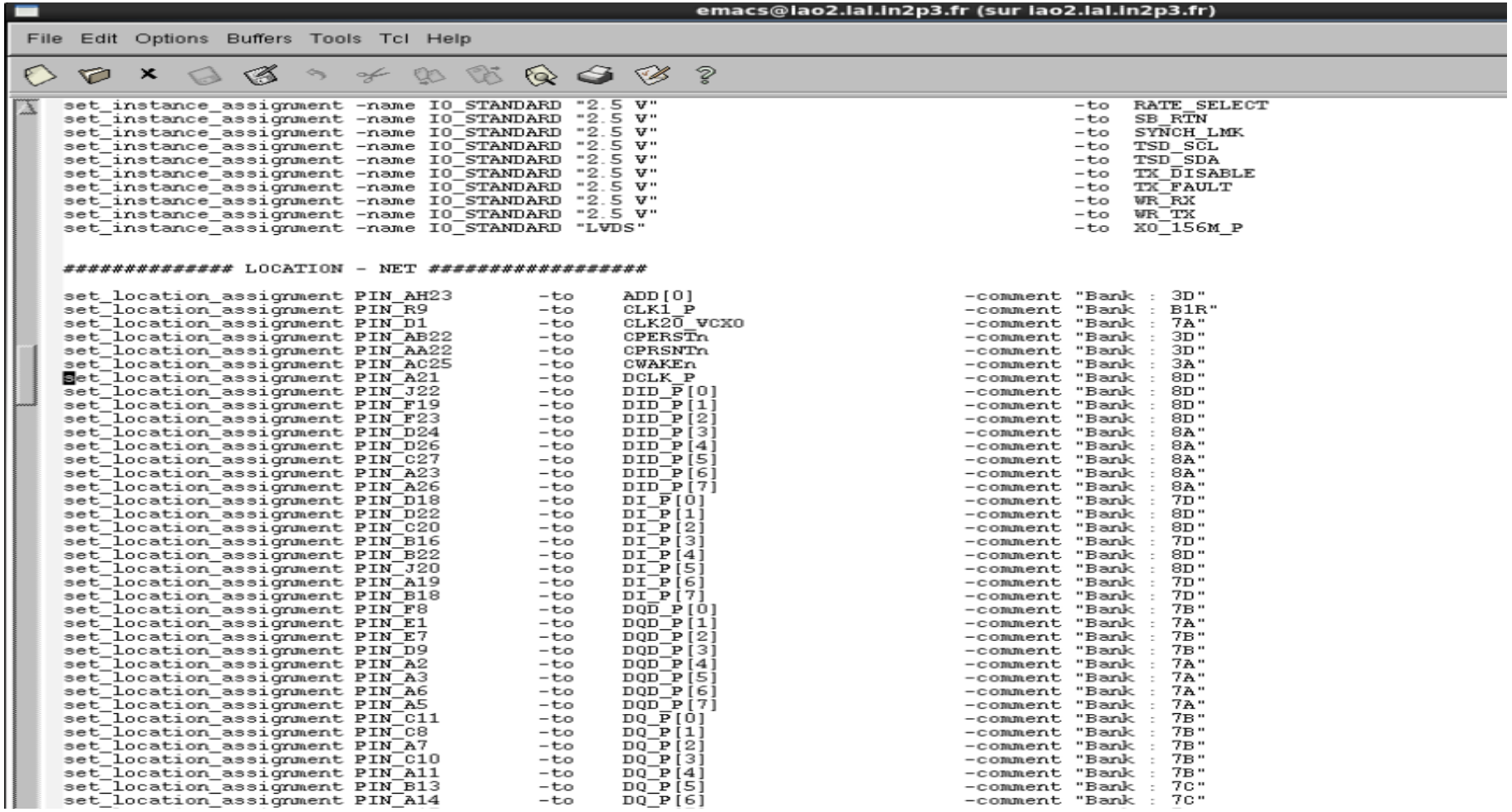
Automatic FPGA Pin Placement Constraint file for Quartus/Altera..



LABORATOIRE DE L'ACCÉLÉRATEUR LINÉAIRE

cnrs

Automatic FPGA Pin Placement Constraint file for Quartus/Altera..

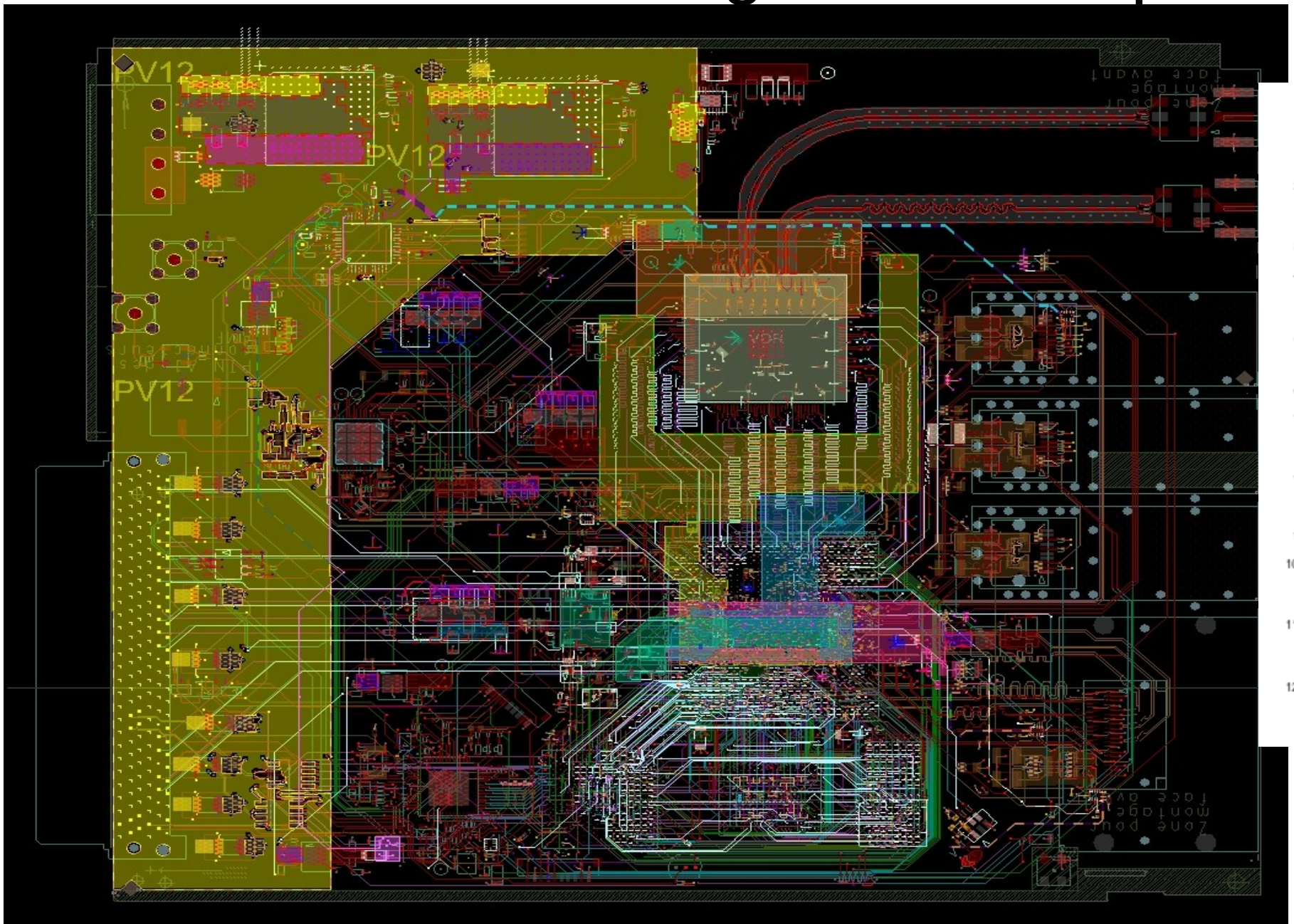


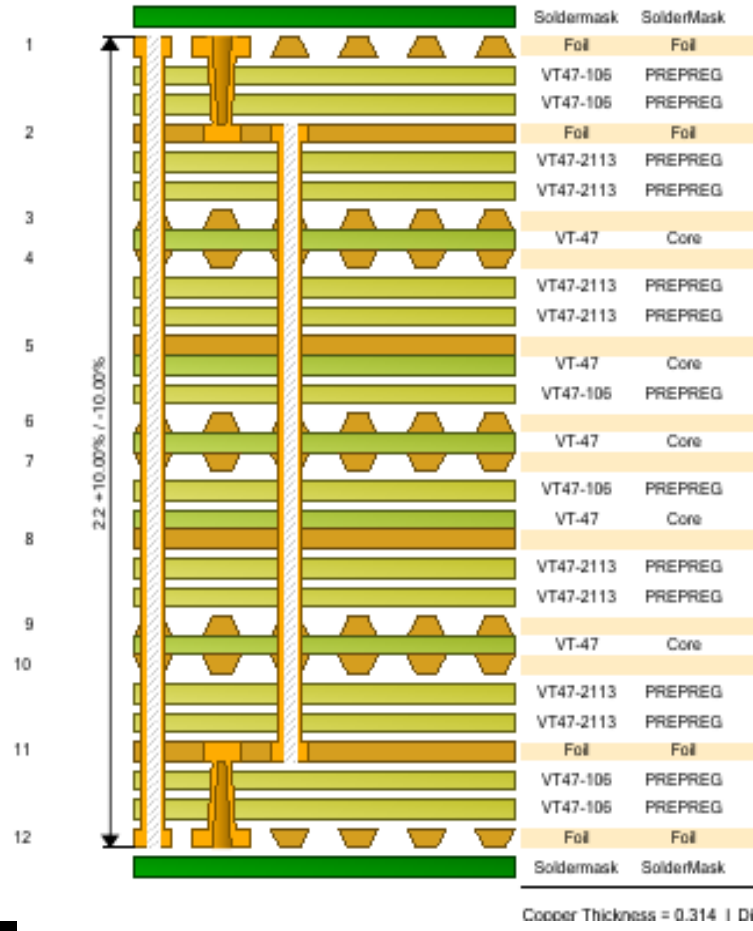

```
set_instance_assignment -name IO_STANDARD "2.5 V"                    -to  RATE_SELECT
set_instance_assignment -name IO_STANDARD "2.5 V"                    -to  SB_RTN
set_instance_assignment -name IO_STANDARD "2.5 V"                    -to  SYNCH_LMK
set_instance_assignment -name IO_STANDARD "2.5 V"                    -to  TSD_SCL
set_instance_assignment -name IO_STANDARD "2.5 V"                    -to  TSD_SDA
set_instance_assignment -name IO_STANDARD "2.5 V"                    -to  TX_DISABLE
set_instance_assignment -name IO_STANDARD "2.5 V"                    -to  TX_FAULT
set_instance_assignment -name IO_STANDARD "2.5 V"                    -to  WR_RX
set_instance_assignment -name IO_STANDARD "2.5 V"                    -to  WR_TX
set_instance_assignment -name IO_STANDARD "LVDS"                     -to  XO_156M_P


############### LOCATION - NET ##################

set_location_assignment PIN_AH23       -to     ADD[0]          -comment "Bank : 3D"
set_location_assignment PIN_R9         -to     CLK1_P          -comment "Bank : B1R"
set_location_assignment PIN_D1         -to     CLK20_VCXO      -comment "Bank : 7A"
set_location_assignment PIN_AB22       -to     CPERSTn         -comment "Bank : 3D"
set_location_assignment PIN_AA22       -to     CPRSNTn         -comment "Bank : 3D"
set_location_assignment PIN_AC25       -to     CWAKEn          -comment "Bank : 3A"
set_location_assignment PIN_A21        -to     DCLK_P          -comment "Bank : 8D"
set_location_assignment PIN_J22        -to     DID_P[0]        -comment "Bank : 8D"
set_location_assignment PIN_F19        -to     DID_P[1]        -comment "Bank : 8D"
set_location_assignment PIN_F23        -to     DID_P[2]        -comment "Bank : 8D"
set_location_assignment PIN_D24        -to     DID_P[3]        -comment "Bank : 8A"
set_location_assignment PIN_D26        -to     DID_P[4]        -comment "Bank : 8A"
set_location_assignment PIN_C27        -to     DID_P[5]        -comment "Bank : 8A"
set_location_assignment PIN_A23        -to     DID_P[6]        -comment "Bank : 8A"
set_location_assignment PIN_A26        -to     DID_P[7]        -comment "Bank : 8A"
set_location_assignment PIN_D18        -to     DI_P[0]         -comment "Bank : 7D"
set_location_assignment PIN_D22        -to     DI_P[1]         -comment "Bank : 8D"
set_location_assignment PIN_C20        -to     DI_P[2]         -comment "Bank : 8D"
set_location_assignment PIN_B16        -to     DI_P[3]         -comment "Bank : 7D"
set_location_assignment PIN_B22        -to     DI_P[4]         -comment "Bank : 8D"
set_location_assignment PIN_J20        -to     DI_P[5]         -comment "Bank : 8D"
set_location_assignment PIN_A19        -to     DI_P[6]         -comment "Bank : 7D"
set_location_assignment PIN_B18        -to     DI_P[7]         -comment "Bank : 7D"
set_location_assignment PIN_F8         -to     DQD_P[0]        -comment "Bank : 7B"
set_location_assignment PIN_E1         -to     DQD_P[1]        -comment "Bank : 7A"
set_location_assignment PIN_E7         -to     DQD_P[2]        -comment "Bank : 7B"
set_location_assignment PIN_D9         -to     DQD_P[3]        -comment "Bank : 7B"
set_location_assignment PIN_A2         -to     DQD_P[4]        -comment "Bank : 7A"
set_location_assignment PIN_A3         -to     DQD_P[5]        -comment "Bank : 7A"
set_location_assignment PIN_A6         -to     DQD_P[6]        -comment "Bank : 7A"
set_location_assignment PIN_A5         -to     DQD_P[7]        -comment "Bank : 7A"
set_location_assignment PIN_C11        -to     DQ_P[0]         -comment "Bank : 7B"
set_location_assignment PIN_C8         -to     DQ_P[1]         -comment "Bank : 7B"
set_location_assignment PIN_A7         -to     DQ_P[2]         -comment "Bank : 7B"
set_location_assignment PIN_C10        -to     DQ_P[3]         -comment "Bank : 7B"
set_location_assignment PIN_A11        -to     DQ_P[4]         -comment "Bank : 7B"
set_location_assignment PIN_B13        -to     DQ_P[5]         -comment "Bank : 7C"
set_location_assignment PIN_A14        -to     DQ_P[6]         -comment "Bank : 7C"
```


LABORATOIRE DE L'ACCÉLÉRATEUR LINÉAIRE

cnrs dépasser les frontières

PCB design in development

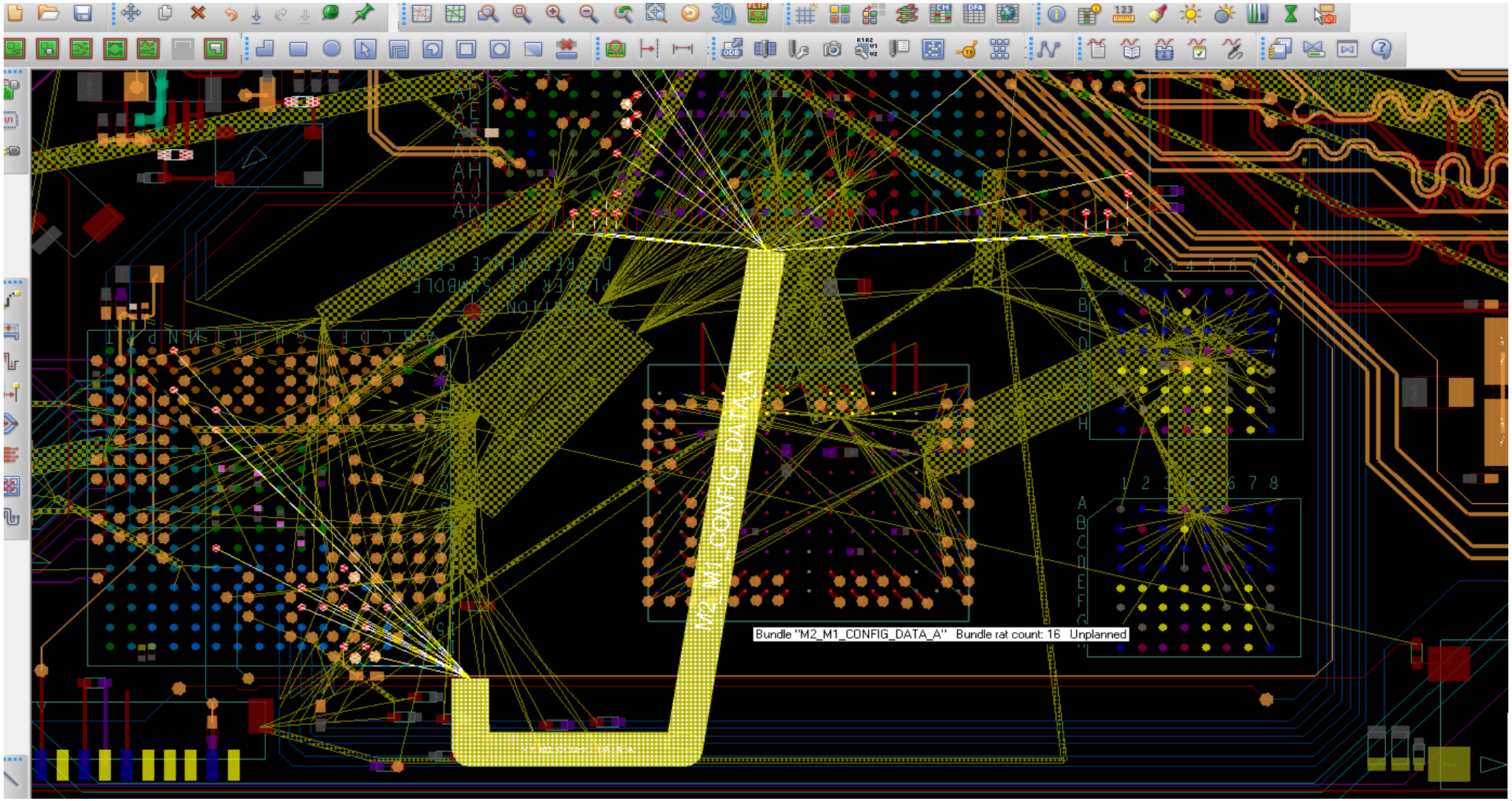




LABORATOIRE DE L'ACCÉLÉRATEUR LINÉAIRE

cnrs dépasser les frontières

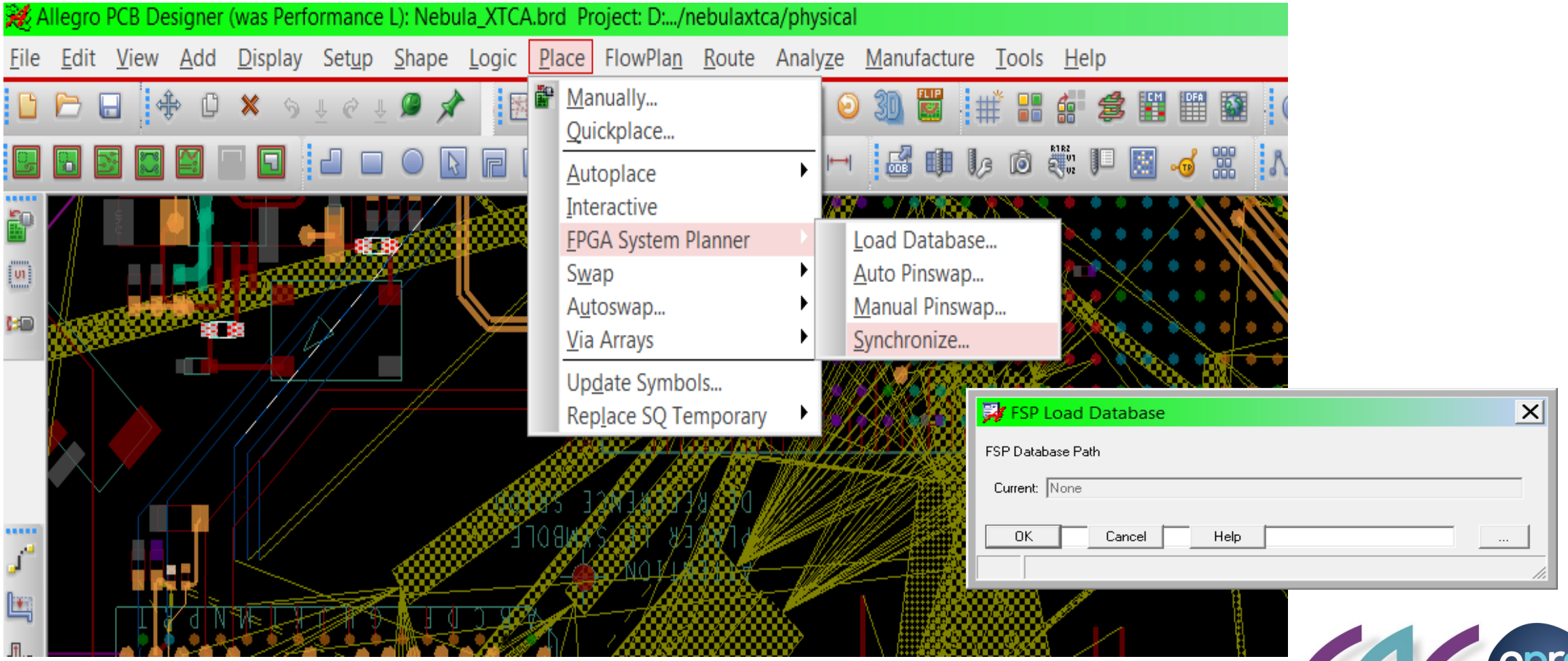
Bundle pin swap optimization
Scheduling and bus slide



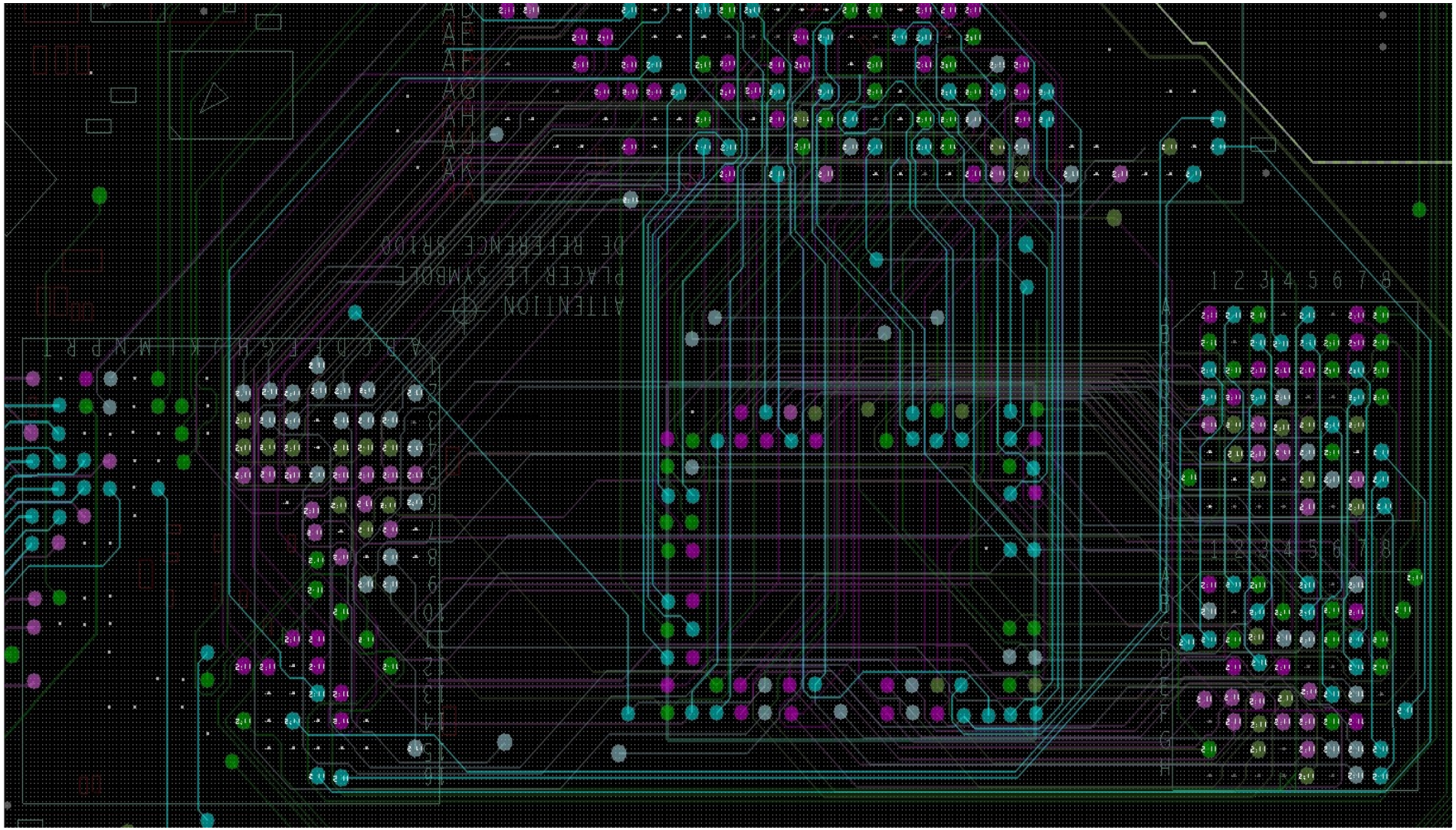
LABORATOIRE DE L'ACCÉLÉRATEUR LINÉAIRE

cnrs dépasser les frontières

Allegro PCB connected to FSP database
Automatic feedback when PCB is written

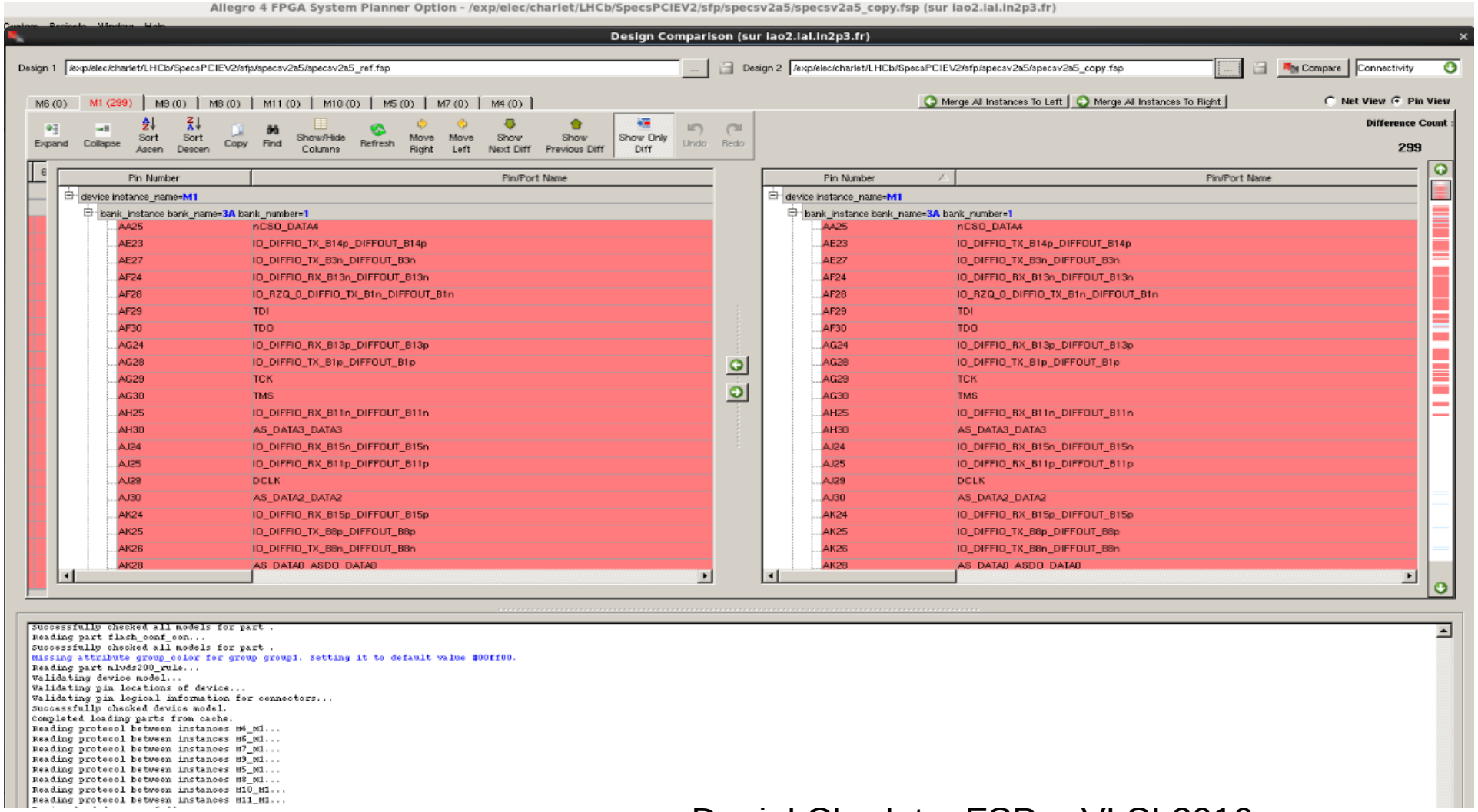


Allegro PCB connected to FSP database real time pin swapping.



LABORATOIRE DE L'ACCÉLÉRATEUR LINÉAIRE

PCB swapping back annotation Compare master FSP design to changes made in PCB



Improvements seen by using FSP

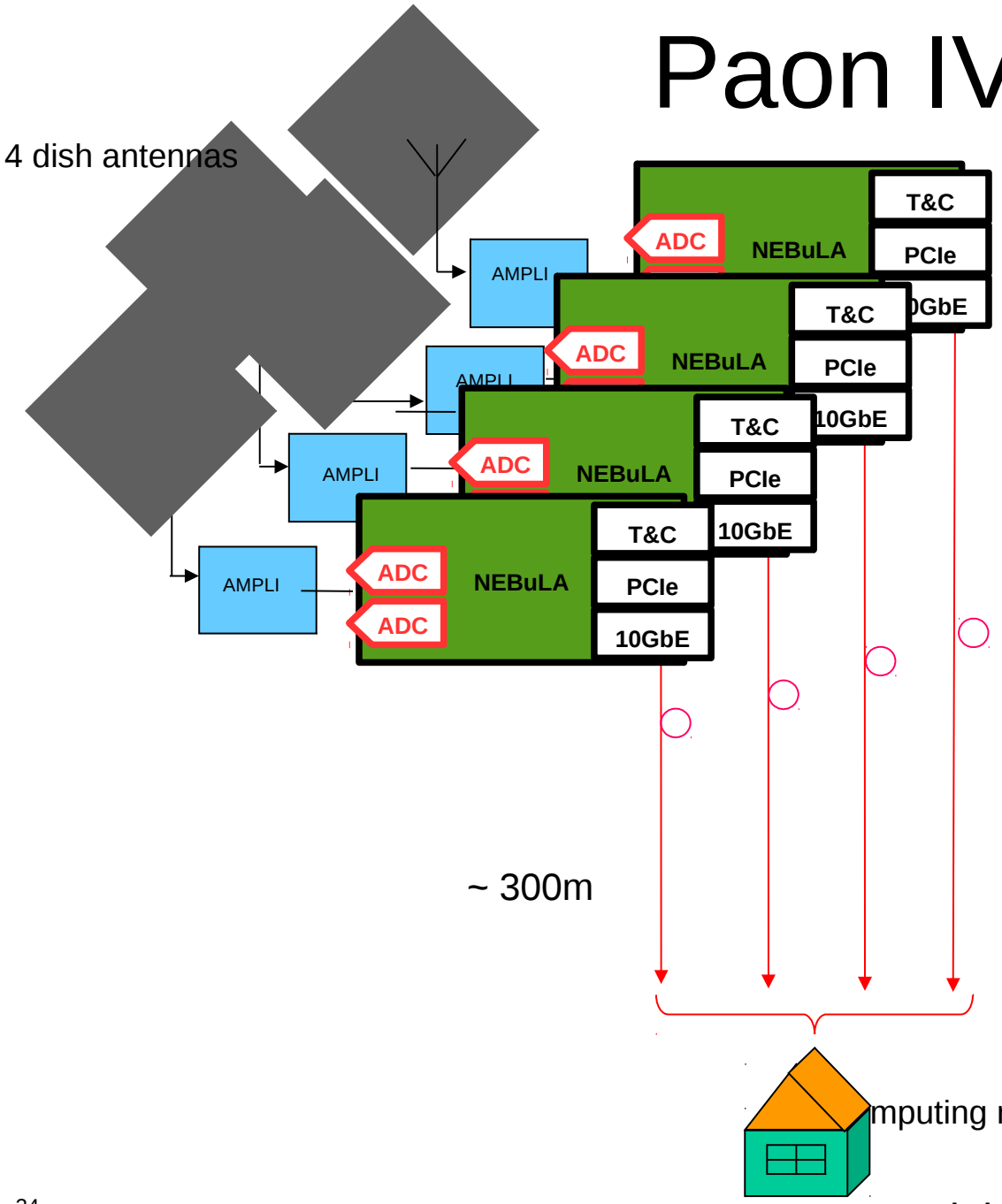
- ✓Board development before FPGA firmware development
- ✓Easy upgrade of FPGA during design
- ✓Exchange physical/logical during development
- ✓Easy swapping in place and route phase with back annotation
- ✓Easy decoupling definition
- ✓Easy power supply assignment
- ✓Save 50% time of previous manual method
- ✓Concurrent engineering between software architecture and PCB
- ✓Automatic schematic generation
- ✓Automatic FPGA pin placement constraint file for Quartus/Altera..


cnrs
dépasser les frontières
IN2P3
Les deux infinis


LABORATOIRE
DE L'ACCÉLÉRATEUR
LINÉAIRE


Station de
Radio astronomie
de Nançay

Paon IV: future Configuration.



IN2P3 FPGA System Planner

Journées VLSI 2016

NEBULA DESIGN

This is the agenda of my presentation
First I wll introduce the CNRS and IN2P3

Trouver la définition des BAO
Le lien avec les ondes gravitationnelles
Surtout lié a la recherche de matier/energie noire
L'ampleur de la collaboration
Le temps depuis

Trouver la définition des BAO
Le lien avec les ondes gravitationnelles
Surtout lié a la recherche de matier/energie noire
L'ampleur de la collaboration
Le temps depuis

The PAON IV demonstrator has a Classic structure for radio astronomy acquisition system

Dishes antenna equipped with low noise amplifier at the top and a long cable to the digitizer system

The Data transfer is performed by optical fiber to the computing room

PAON IV comprises 8 channels ,2 polarity by antenna

The bandwidth of the analog chain is 250MHz between 1.25Ghz and 1.5Ghz

The digitalization is performed at 500Mhz with 8bits of dynamics

The dish diameter is 5m with a beam antenna around 3°

After digitalization a fast Fourier transformation of 2048 point is made on-line by an FPGA. The FPGA format and send data to the computing room by optical fibers at 5Gb/s using home made protocol.

In the computing room, by PC farm, computation of the cross correlation at each frequency

Data accumulated and disk storage

Off line processing data to clean data and to performed maps at differents frequency .

A snapshot of the cyg A galaxy whit the arm of our own galaxy realize by the PAON IV detector

Due to bandwidth and also of the sensitivity of the sys tem we encounter SWR problem . This has the effect of modification of signal.

The extend of the signal modification is proportional of the mismatch adaptation, the bandwidth , the cable length.

Classically on radio astronomy there is a narrow bandwidth and there are not affected by this issue.

SWR Standing wave ratio

I will present you the NEBULA project and how we have designed it using FSP

The NEBULA board is base on the xTCA for physics standard but

It can also work in stand alone mode , only 12v power supply and optical link are mandatory to the board.

The earth of the system is an FPGA ARRIA V GX

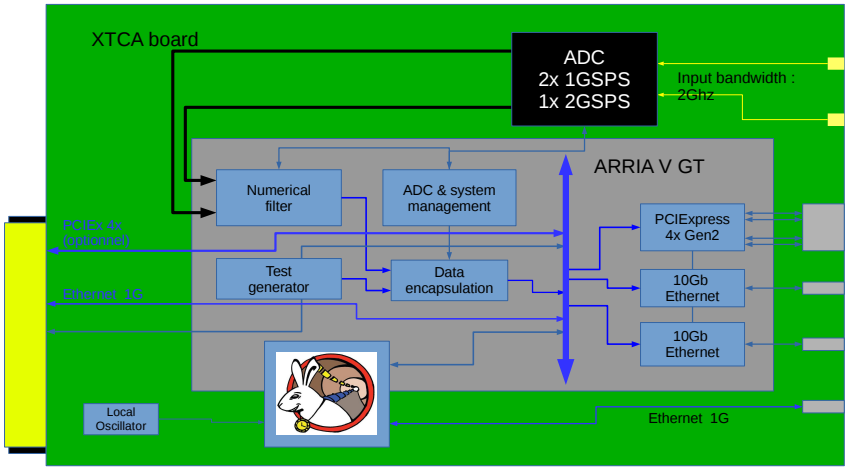
The board integer one double channel ADC at 1G simple on 8bits with the possibility to configure in one channel at 2G samples

For configuration synchronization and time tagge we have implemented the white-rabbit protocol developed by the CERN

that permit to synchronize a distributed system whit an accuracy under 20ps

The data rate transfer could reach 20Gb/s using 2 x 10Gb Ethernet link or one 4x Gen2 exiternal PCIEpress link

NEBULA FPGA content



- Common clock for the entire network

All network nodes use the same physical layer clock,
generated by the System Timing Master
Clock is encoded in the Ethernet carrier and recovered by
the PLL in the PHY.

- PTP IEEE1558
- Synchronizes local clock with the master clock by measuring
- and compensating the delay introduced by the link.

For electronic development the institute use cadence product since 25 years.

I will described the classical methodology used for a board study incorporating FPGA

First stage generation of FPGA pin assignment using EDA products in my case it's ALTERA and QUARTUS

Generation of the component with part developer to integrate in the schematics entry

You do manually your design whit CONCEPT schematic

Finally Placing and Layout with ALLEGRO

This classic methodology are further inconvenient:

Time consuming

Numerous possible human error at different stage mainly due to the iterative process.

At the first stage when we fix the FPGA pin assignment we have only a rough idea of the relative chips placement on the board, and no idea how the net will be routed.

To swap pin he is necessary to do an iterative process

The FSP Methodology

The main feature is it 's interface base connectivity (you define bus function like address, data, ctrl...)

It's a system floor-plan, the inter-connexion are guided by the relative placement of the components, the designer can also early define placement

The system has a accuracy component pin rules like llogic standard, functions

The resulting data base is linked to PCB designer tools (allegro, orcad)

FSP provide FPGA libraries and additional components like memories , connectors, ….

Designer can define it's own components

The design interconnection is under the control of the designer

The FSP flow
Generation of interconnection between components according to the relative components placement and I/O rules
Generation of powers and reference voltage for all components
Decoupling capacitor definition
Generation by FSP of a database compatible with ALLEGRO
Generation by FSP PCB design placement and HDL schematics .
Generation by FSP of FPGA pin file assignment
Import of the FSP schematic in top design
Import of the FSP allegro database In allegro
Layout of the design with pin swap capability under control of FSP engine
Export to FSP of the modify data-base for more accuracy check and validation by the designer
Export to FPGA tools the new pin file assignment

During all different stage no manually enter potentially source of error

The numerical part of the board have been designed using FSP.

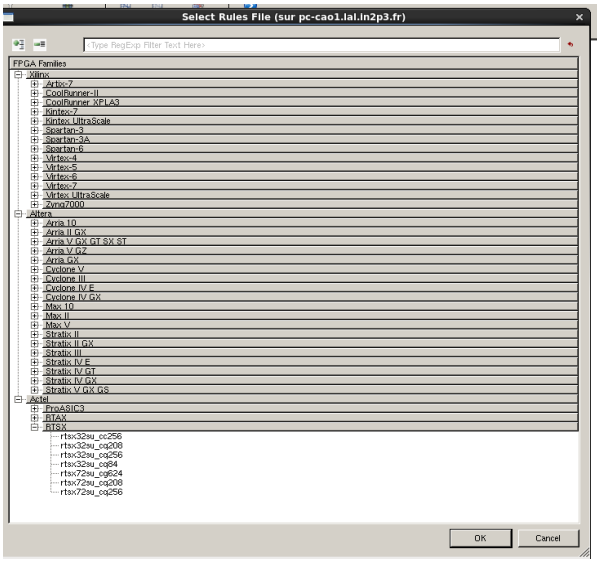
To achieve this FSP project I have use FPGA libraries delivered by Cadence, and also some others like connectors but we have to define nearly all the others

A snapshot of FSP board canvas with all the interconnection realize by FSP in dependence of the relative placement of the components and by taking into account of the pins rules.

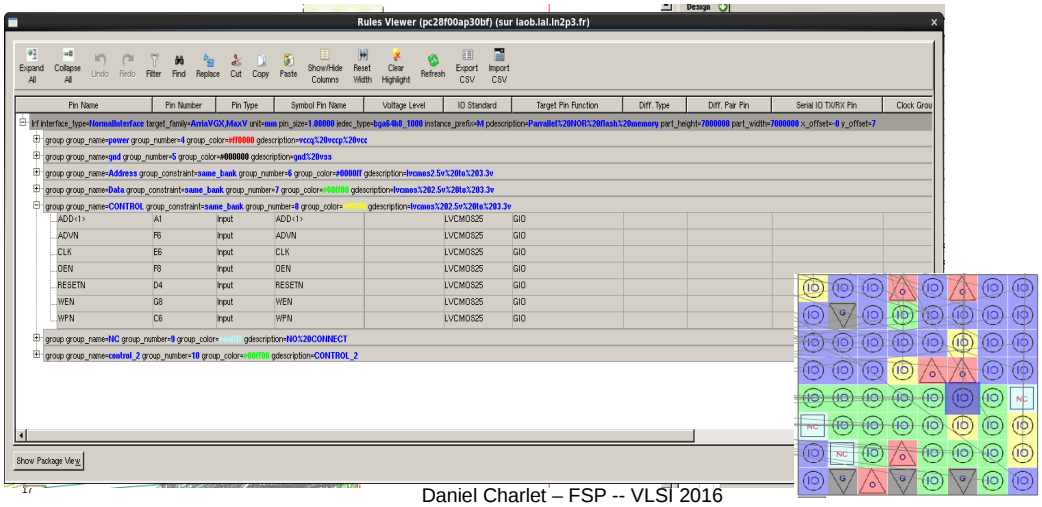
To succeed to properly interconnect all the net it's necessary to schedule the interface routing and fix the routing of some nets

Canvas toile

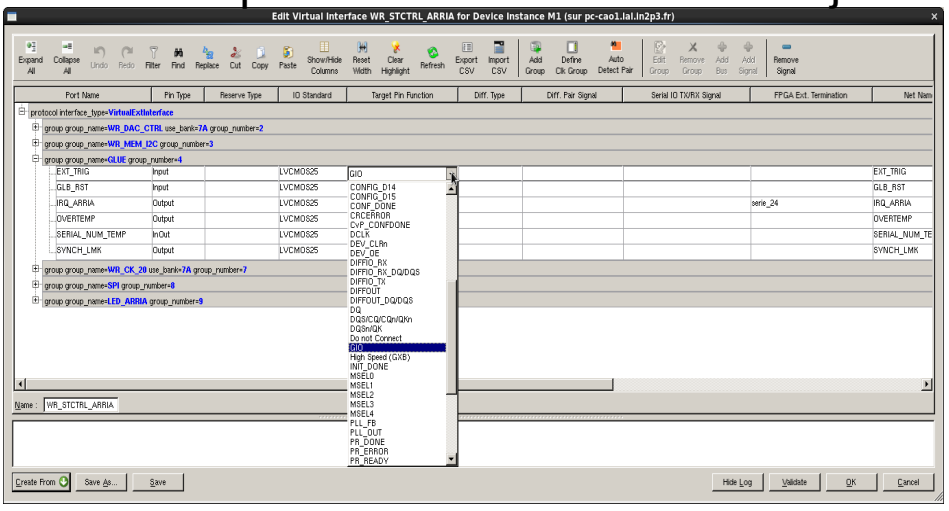
FPGA selection
FSP Project



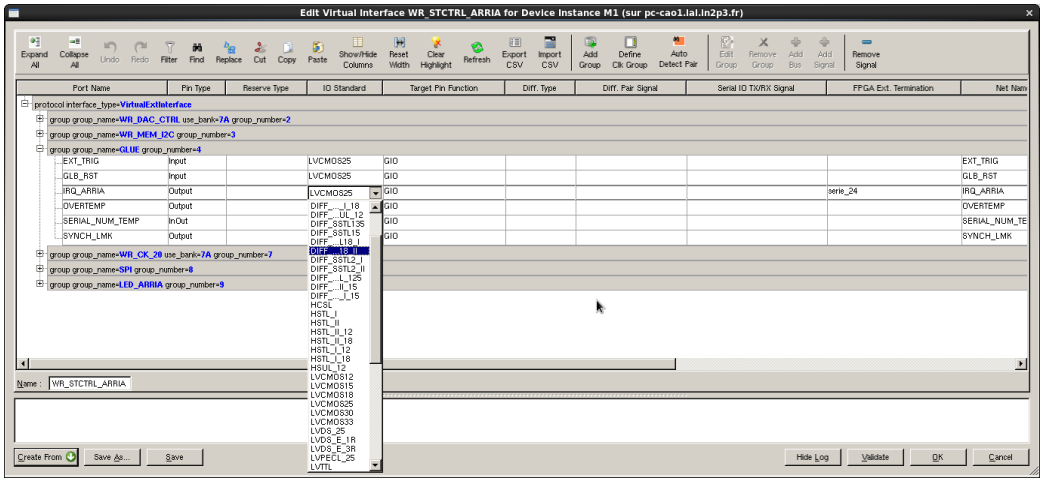
Component definition 1 FSP Project



Component definition 2 FSP Project



Component definition 3 FSP Project



A zoom of the connectivity where we can see the capability of the system to manage a daisy interconnection between 4 components

On the left side the net view windows thats allows to check the realize connectivity order by components

One of the very interesting feature is decoupling capacitor. You can define for each components and each power supply the decoupling capacitor .

Advantageously, In ALLEGRO when you place the capacitor there are grouped by component and by power supply .

In old development methodology you have to attach manually capacitor to component

Other functionality, the power assignment.
For each component yo can define power supply.

For big FPGA , up to 2 thousand pins, you have a huge numbers of powers pins (further hundred), as the system has the exact knowledge of all powers pins there is no risk to forgot any power pins.

The schematic generation.

Related to the design connectivity, FSP can generate HDL schematics. The main contribution of this sate is the automatic generation of HDL component as well as the writing of net interconnect. As there is no human intervention no error possible at this stage. The HDL schematic is the exact replica of the interface interconnect define by the designer

Furthermore it automatically add the capacitor on the schematics

This schematics can be easily imported in the top design.

A view of the resulting schematic This one could be modified by users and be preserved at each new iteration.

The other main contribution is the generation of the pin constrain for the FPGA. It can be made for all the FPGA of the project and at any stage of design.

A view of the generated files with I/O standard definition and the pin assignment

An Allegro view of the current board

As FSP is base on interface base connectivity, this concept is passed to ALLEGRO by the ability to define and use bundle. Each interface define at FSP level became a bundle in ALLEGRO.

The bundle shape can be modifies to take into account the layout possibility

During iterative process you need to synchronize FSP to ALLEGRO, this can be done easily in allegro using a set of command.

During the routing with ALLEGRO the system highlight the authorize pins and assign color code for the different interfaces. This pin swap capability is under control of FSP data base.

During this stage it's not mandatory to refer to this pin planner of the EDA tools to check if is it possible to swap those pins

At the end of the swap modification it's mandatory that the designer validate the modification with a full synchronize ALLEGRO to FSP data-base. At this level the system perform an extensive verification.

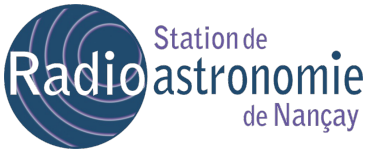
A new schematics generation and import in the top design is necessary to keep the synchronism everywhere.

Last step a new FPGA pins constraints need to be generate.

My conclusion
Main feature of fsp
Improvement on somme points

My design save time
Changemnent
Easy design reuse


cnrs
dépasser les frontières
IN2P3
Les deux infinis

LABORATOIRE
DE L'ACCÉLÉRATEUR
LINÉAIRE

Station de
Radio astronomie
de Nançay

The future acquisition system

It's Distribute acquisition system where digitalization is performed the most closer as possible of the LNA to decrease the cable length to limits the SWR effect

The new issue of this architecture is the synchronization of a distributed system.

With this architecture there is no limitation of distance between antennas

This architecture has been made possible by using new FPGA family ARRIA V. there are fast and low cost and integrate further transceivers up to 10Gb/s