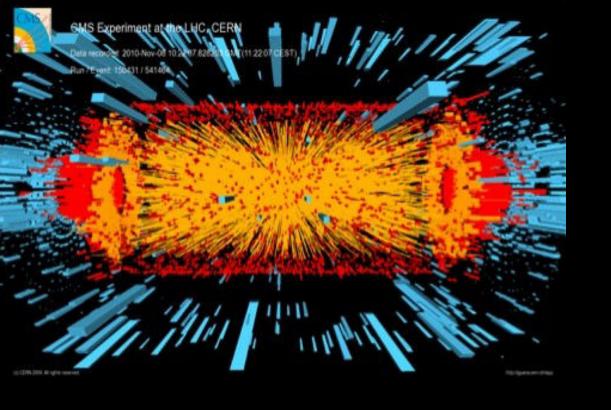


Managed by Fermi Research Alliance, LLC for the U.S. Department of Energy Office of Science

#### CMS L1 Silicon-based Tracking Trigger Evaluation of Virtex-7 and Kintex UltraScale FPGA

Irene Degl'Innocenti Italian Summer Students Final Reports September 25<sup>th</sup> 2015 Supervisor: Ted Liu



# CMS Experiment at the LHC CERN

CMS L1 Tracking Trigger: Will need to reconstruct charged particle trajectories for every beam crossing.

#### Few numbers:

- 40 million beam crossings per second, one every 25 ns
- Bandwidth required to transfer up to 100 Tb/s

# The challenges

#### Compute-intensive

 Massively parallel computation involving very large number of processing elements;

#### Communication-intensive

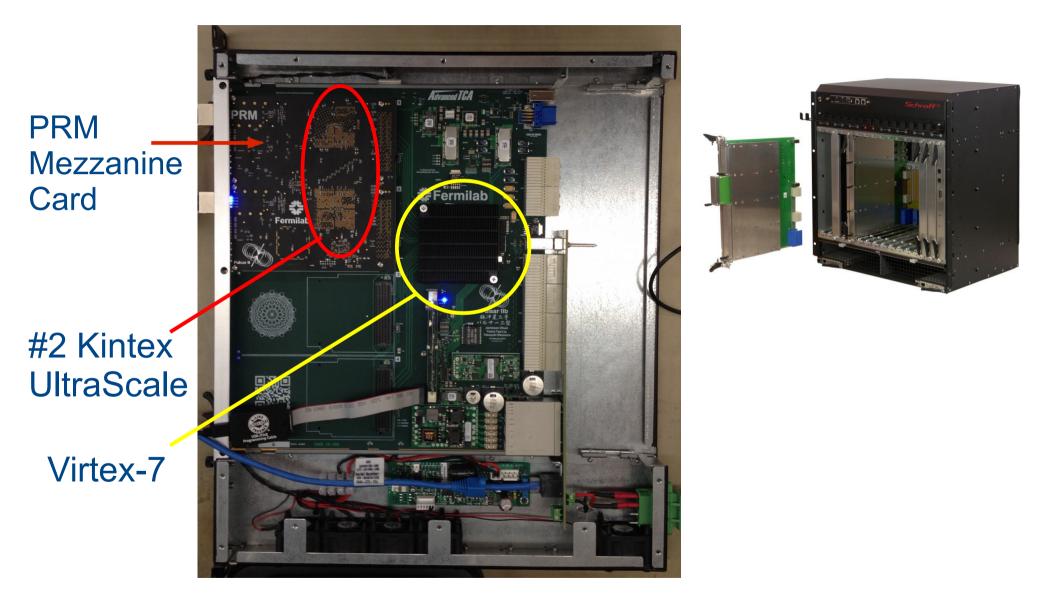
- High-speed transfer of data among processing elements;
- Data-intensive

3

- High-speed manipulation of very large quantities of data.



#### **Pulsar IIb**





#### **XILINX FPGAs**

#### Focus on:

- Data transfer evaluation: IBERT Test
   ---> High speed data transfer & serial link
- BlockRAM features study
   ---> Huge data storage and quick access



# **GTH Transceivers**



Irene Degl'Innocenti | CMS L1 Silicon-based Tracking Trigger: Evaluation of Virtex 7 and Kintex UltraScale FPGA

#### **GTH Transceivers**

VC709 board provides acces to 22 GTH Transceivers:

- #8 PCI Express x8 endpoint edge connector
- #10 FMC HPC connector
- #4 SFP+ connectors

GTH are grouped by four in Quads; Through SFP connectors we have access to **Quad 113**.



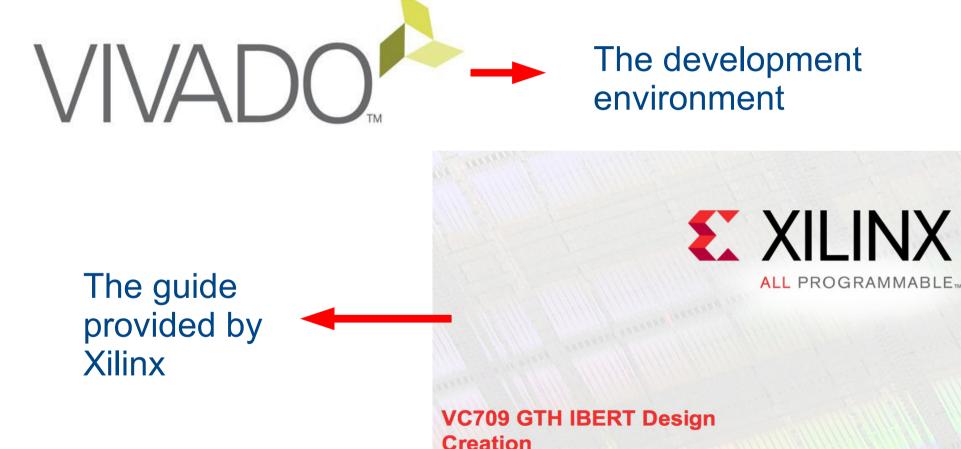
# **Integrated Bit Error Ratio Test**

- Integrated Bit Error Ratio Test (IBERT) core for 7 series FPGA GTX transceivers is designed for evaluating and monitoring the GTX transceivers.
- This core includes pattern generators and checkers that are implemented in FPGA logic, and access to ports and the dynamic reconfiguration port attributes of the GTX transceivers.

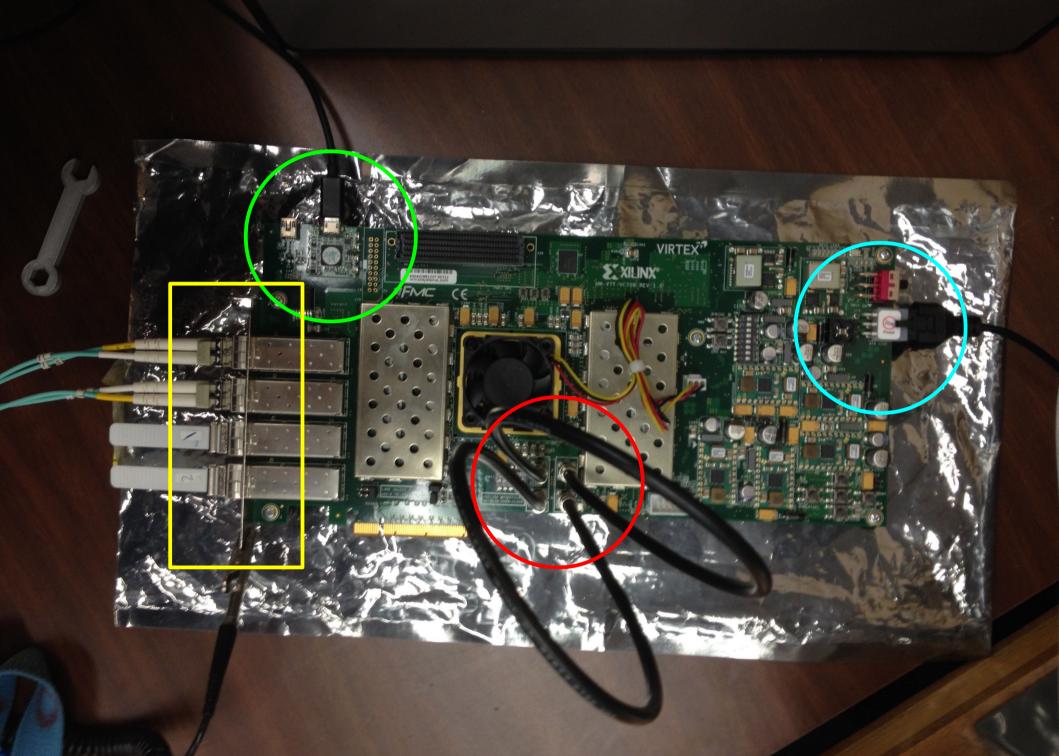
Communication logic is also included to allow the design to be run time accessible through JTAG.



#### How to program VC709: IBERT test







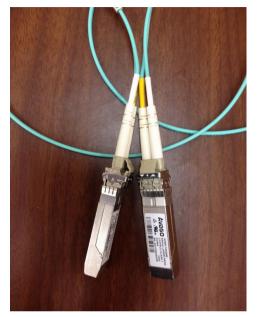
#### **Connections:** optical cables and loopback



Prismian O.C. 2F 50/125 OM2 BB



AFL Telecommunications 1-800-AFL\_FIBER 50/125



AMPHENOL CABLES LSZH 04/13 0942M



ELPEUS TECHNOLOGY SFPP LB



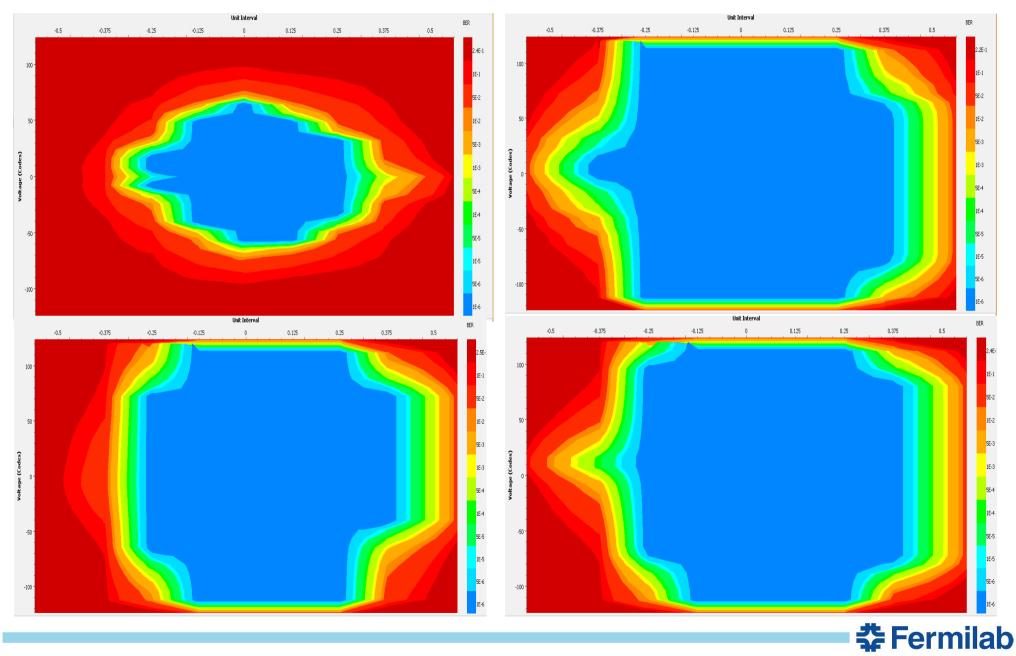
#### **Serial I/O Links Monitor**

Ser	rial I/O Links														пēх
0	ame	тх	RX	Status	Bits	Errors	BER	BERT Reset	TX Pattern	RX Pattern	TX Pre-Cursor	TX Post-Cursor	TX Diff Swing	- 1	DFE Enabled
		MGT_X1Y13/TX MGT_X1Y14/TX	MGT_X1Y12/RX MGT_X1Y13/RX MGT_X1Y15/RX MGT_X1Y14/RX	10.313 Gbps 10.313 Gbps	2.882 2.887	0E0 0E0	3.326E-12 3.47E-12 3.464E-12 3.46E-12	Reset Reset	PRBS 7-bit	<ul><li>▼ PRBS 7-bit</li><li>▼ PRBS 7-bit</li></ul>	<ul> <li>▼ 0.00 dB (00000)</li> <li>▼ 0.00 dB (00000)</li> <li>▼ 0.00 dB (00000)</li> <li>▼ 0.00 dB (00000)</li> </ul>	<ul> <li>▼ 0.00 dB (00000)</li> <li>▼ 0.00 dB (00000)</li> <li>▼ 0.00 dB (00000)</li> <li>▼ 0.00 dB (00000)</li> </ul>	<ul> <li>269 mV (0000)</li> </ul>	v v	
	Tcl Console 🔵 🗩 Mess	sages 🔏 Seri	ial I/O Links	🦲 Serial I/O S	cans	21 문									۲

list design

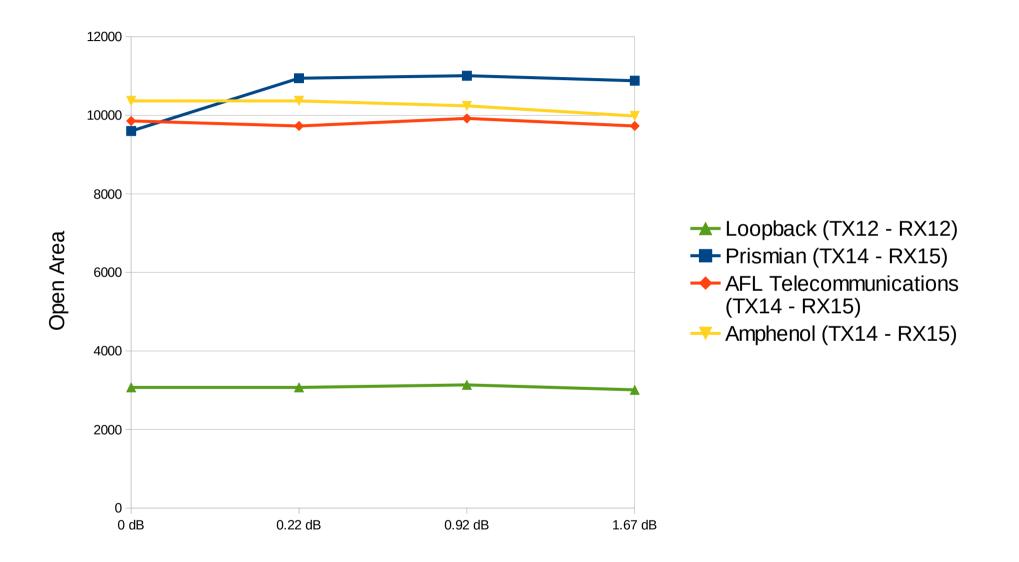


# The statistical eye



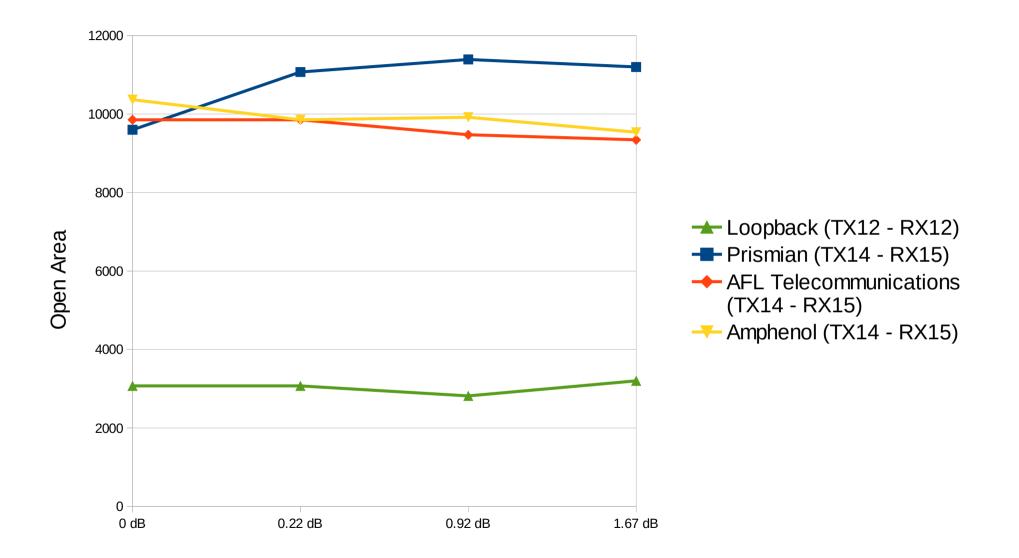
09/25/2015

#### **TX Pre Cursor**



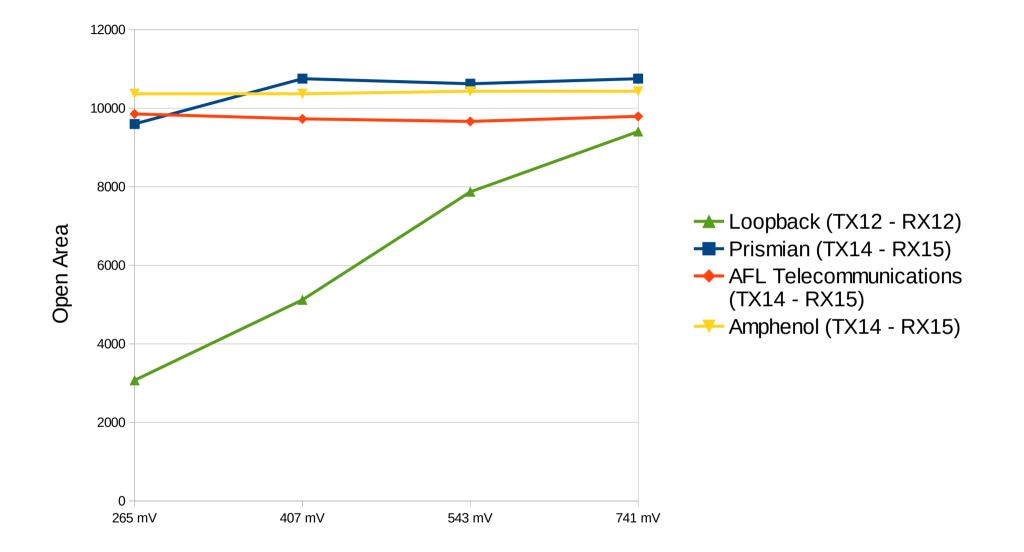


#### **TX Post Cursor**





#### **TX Differential Swing**





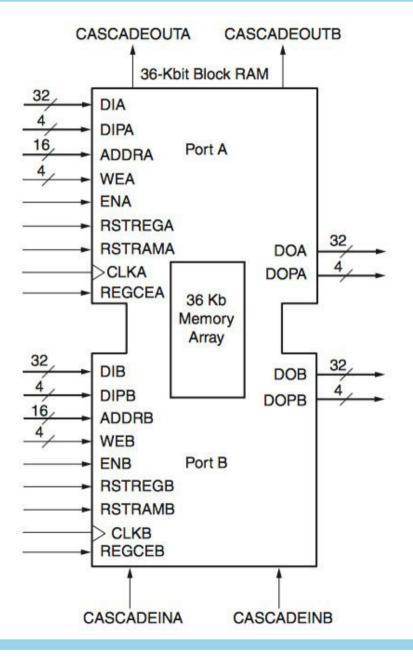
# **BlockRAM**



17 Irene Degl'Innocenti | CMS L1 Silicon-based Tracking Trigger: Evaluation of Virtex 7 and Kintex UltraScale FPGA

09/25/2015

# **Block RAM in Virtex-7 FPGA**



They are used for:

- $\rightarrow$  data storage or buffering
- $\rightarrow$  state machines or FIFO buffers
- $\rightarrow$  shift registers, LUT, or ROMs.

Main features:

- stores up to 36 Kbits
- two independent 18 Kb RAMs or single 36Kb RAM
- Write and Read are synchronous operations;
- the two ports are symmetrical and totally independent.



#### **Virtex 7 - XC7VX690T**

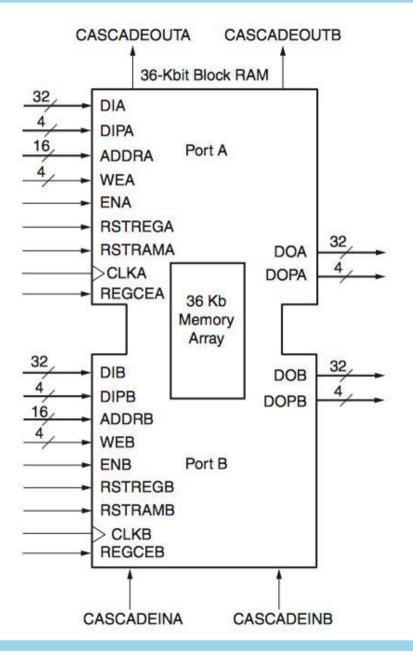
#### # Columns per device: 15

#### # 36Kb Block RAM blocks per column: 100





# **Possible configurations: TDP**



**True Dual Port Mode:** 

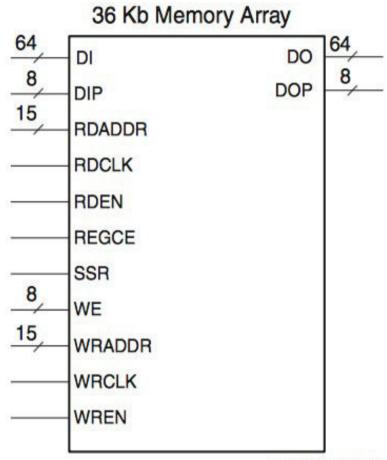
- → Ports A and B are completely indipendent, they only share data.
- $\rightarrow$  Clocks can be different
- → Data can be written to and be read from both ports
- $\rightarrow$  Maximum Bit Width: 36

ATTENTION: avoid conflict!

Accessing the same memory location from both ports could produce invalid output.



# **Possible configurations: SDP**



UG473\_c1\_06\_011414

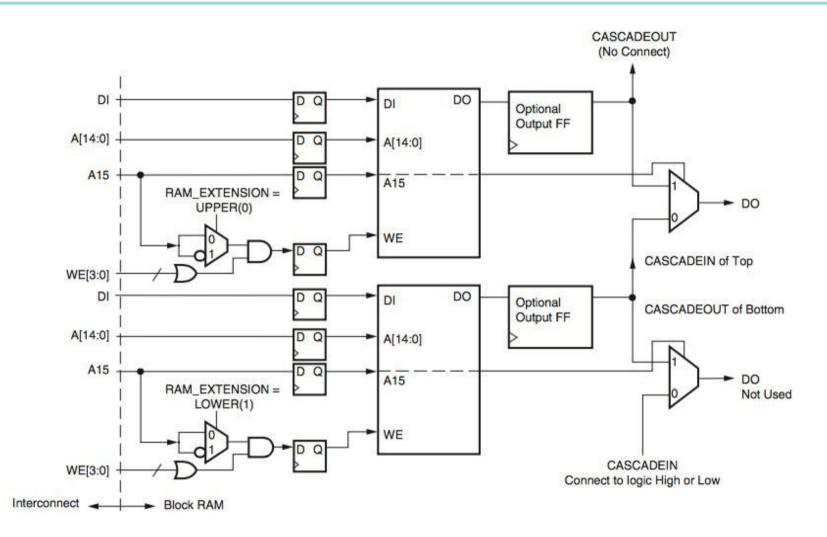
Simple Dual Port Mode:

- $\rightarrow$  Port A is designated as the READ port
- $\rightarrow$  Port B is designated as the WRITE port
- → Maximum Bit Width: 72
- $\rightarrow$  One port width is fixed x64 or x72

Collision: when the read and write port access the same data location at the same time.



# **Possible configurations: Cascadable**



Two adiacent 32K x 1 RAMs can be combined to form one 64K x 1 RAM



#### Write Modes

- WRITE\_FIRST: outputs the newly written data onto the output bus.
- READ\_FIRST: outputs the previuosly store data while the new one is being written.
- NO\_CHANGE: maintains the output previuosly generated by a read operation.

 $\rightarrow$  Byte-Wide Write Enable: allows writing 8 bit portions of incoming data



# **Waveform Simulation**

#### VHDL Entity bRAM created using the MACRO



Parameters:

- Block RAM size: 18Kb 36Kb
- Optional output registers: enabled not enabled
- Data Width: 16 bits 32 bits
- Write Mode: WRITE\_FIRST READ\_FIRST
- Clock frequency: 200 Mhz 400 Mhz



# **Example: port A R/W, port B R/O**

BRAM size 36Kb, data width 32 bits, output register enabled, WRITE FIRST, clock 200 MHz

tb_b	RAMgen_behav.wcfg											
<u>کا</u>												
	Name	Value		80 ns		85 ns	87.500	ns 90 ns	92.600	ns  95 ns		100 ns
<u>_</u> +	🗓 dka_cyde	29	<u> </u>					8			2	
Q-	🐚 dkb_cyde	29	1	6 X	1	7 )	1	8	1	y e	2	• X
	U testing	TRUE										
4	🗓 docka_s	0										
14	<sup>™</sup> dockb_s	0										
<b>N N N</b>	🖽 📲 addra_s[9:0]	000000000	00	0000000	1				00000	00010		
	∎¶addrb_s[9:0]	000000000	00	00000000	1				00000	00010		
	🖭 📲 dina_s[31:0]	a2222222		1111111								
2	🖽 📲 dinb_s[31:0]	b1111111										
4	🕼 wea_s	0										
r.	🕼 web_s	0										
-	🕼 rsta_s	0										
	₩ <mark>rstb_s</mark>	0										
	🖬 📲 douta_s[31:0]	i			alll	1111						
<u>zı</u>	∎ 📲 doutb_s[31:0]	0000000			alll	1111						
1000	🕼 CkPer_a	5000 ps										
	🕼 CkPer_b	5000 ps										
	🕼 TestLen	1000001										



# **Example: port A R/W, port B R/O**

BRAM size 36Kb, data width 32 bits, output register **not** enabled, WRITE\_FIRST, clock 200 MHz

Untitled 2									
<u>→</u>								87.600 n	<mark>.s</mark>
💾 Name	Value		85.0 ns	85.5 ns	86.0 ns	86.5 ns	87.0 ns	87. <mark>500 ns</mark> 87.5 ns	88.0 ns
🔍 🗓 dka_cyde	18				7				
🔍 🖟 dkb_cyde	18				7				
🔍 🗤 testing	TRUE					TRUE			
k locka_s	1								
dockb_s	1								
	002	001	X				002		
▶ ፹₩ addrb_s[9:0]	002	001	X				002		
🖆 🖽 📲 dina_s[31:0]	a2222222	allllll	¥			a	222222		
ڬ 🖪 📲 dinb_s[31:0]						υυυυυυυ			
📲 🧏 wea_s	1								
web_s	0								
I Ista_s	0								
ug rsub_s	0								
douta_s[31:0]	a2222222				111111				
21 🖽 📲 doutb_s[31:0]	a2222222			al	111111				
CkPer_a	5000 ps					5000 ps			
lie CkPer_b	5000 ps					5000 ps			
1 TestLen	75					75			



# **Example: port A R/W, port B R/O**

# BRAM size 36Kb, data width 32 bits, output register enabled, **READ\_FIRST**, clock 200 MHz

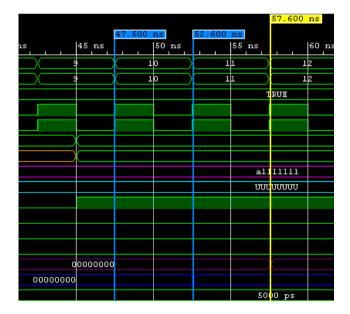
Untitled 1						
<b>≥</b> ]						97.600 ns
💾 Name	Value	75 ns  80 ns	85 ns	87.500 ns  90 ns	92.600 ns  95 ns	100 ns
 Q+ □	20		17		19 ×	
Q− ¼ dkb_cyde	20		17			
U testing	TRUE		+			RUE
U docka_s	1					
line clockb s	1					
ddra_s[9:0]	002	001	X		002	
▶ 🖬 📲 addrb_s[9:0]	002	001	X		002	
1 III IIII IIII IIIIIIIIIIIIIIIIIIIIII	a2222222	alllill	X			a2222222
	UUUUUUUU				ឃា	ບບບບບ
Wea_s	1					
₩ web_s ₩ rsta_s	0					
La rsta_s	0					
→ <b>a</b> → douta_s[31:0]	a2222222	allIll	1		0000000	a22
1 s[31:0]	a2222222	all1111				
🔢 🖟 CkPer_a	5000 ps				50	00 ps
🕼 CkPer_b	5000 ps					00 ps
1ª TestLen	75					75



#### **Observations**

- The clockedge-to-output delay is modeled 0.1 ns
- When WE\_A is low, port A outputs the previuosly store data while the new one is being written.
- When WE\_B is low, port B outputs the newly written data onto the output bus.

Name	Value			190 ns		195 ns	197.500	<mark>ns</mark>  200 ns	202.600	<mark>) ns</mark>  205 ns	207.600 I2	n 21
🕼 dka_cycle	50	37	<u> </u>	8		9		0	4		42	
We dkb_cyde	50	37		38	л <u> </u>			0	4		42	
W testing	TRUE	<u> </u>	<u> </u>		<u> </u>	Ĕ′	<u> </u>	°				
1 docka_s	0											
u dockb_s	0											
🖽 📲 addra_s[9:0]	0000000011		00000	00001	-				00000	00010		
	0000000011		00000	00001					00000	00010		
⊞	ааааааа											
🖽 📲 dinb_s[31:0]	bbbbbbbb		b111	1111								
14 wea_s	0											
₩ web_s	0											
🕼 rsta_s	0											
₩ rstb_s	0											
🛨 📲 douta_s[31:0]	b2222222				Ы11111	1			a222	2222		
	0000000				Ь111111	1						
🕼 CkPer_a	5000 ps											





## **Block RAM in Kintex UltraScale**

Main changes from 7 Series FPGA:

- SDP memory supports NO\_CHANGE mode
- New data cascading scheme (more than x2 block RAMs)
- Address enable added
- Dynamic Sleep mode preserving data content



## **Waveform Simulation**

VHDL Entity bRAM created using the IP block RAM generator



Different modes:

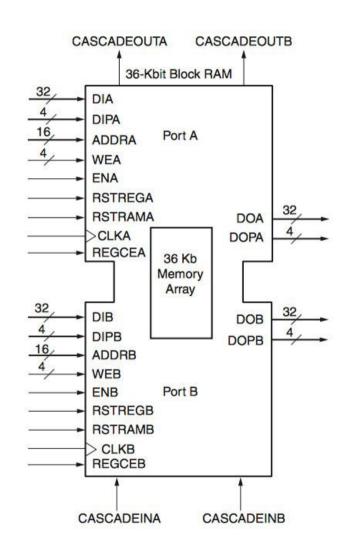
- TDP
- CASCADE Single Port
- SDP + ECC



# **TDP (True Dual Port) Mode**

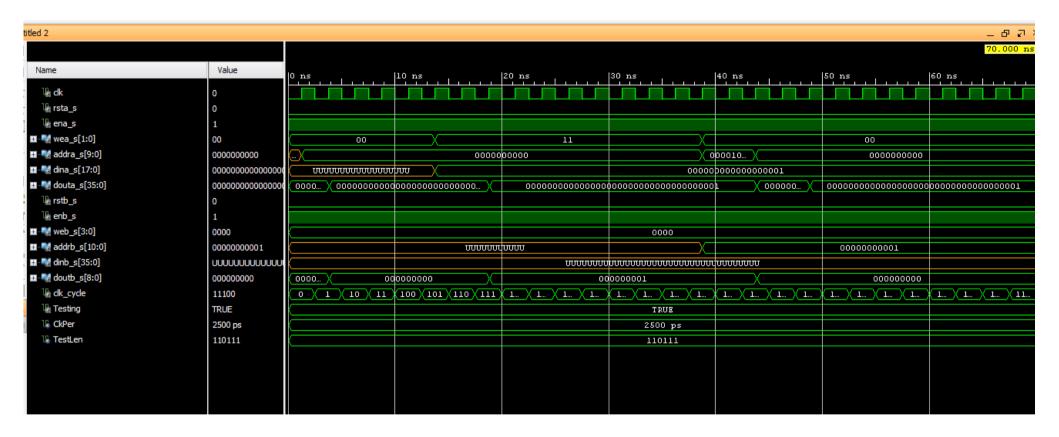
	Port A	Port B
Write Width	18	36
Read Width	36	9
Write Depth	1024	512
Read Depth	512	2048
Operating Mode	WRITE_FIRST	READ_FIRST
Output Register	Enabled	Enabled
Reset Priority	Clock Enable	Set-Reset
Address Width	10	11

# 36K BRAMs1Read Latency2 clock cycles





## **TDP (True Dual Port) Mode**

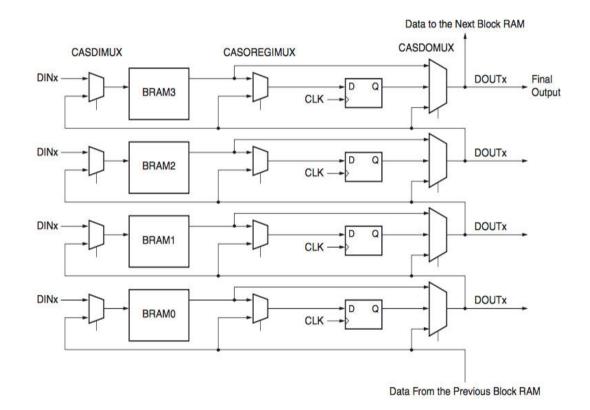




#### **Cascade Mode – Single Port A**

	Port A
Write Width	36
Read Width	36
Write Depth	4096
Read Depth	4096
Operating Mode	WRITE_FIRST
Output Register	Enabled
Reset Priority	Set - Reset
Address Width	12

# 36K BRAMs	4
Read Latency	2 clock cycles





#### **Cascade Mode – Single Port A**

itled 3							
Name	Value	1	.0 ns	20 ns	30 ns	40 ns	50 ns
14 dock	U						
¼a dock_s	1						
₩aen_s	0						
₩ rst_s	0						
	0			1	Х	0	
∎₩addr_s[11:0]	001		001	χ	002	χ	001
⊞¶in_s[35:0]	<del></del>	0	χ	00000002		fff	ffffff
	00000002	0C	. / 00000001 / 0000	00002 X 00000000		00000002	
₩ dk_cyde	23	3	4 5 6 7	8 9 10 11	12 13 14 15	16 17 18 19	20 21 22
U Testing	TRUE				TRUE		
1 CkPer	2500 ps				2500 ps		
1 TestLen	110111				110111		



# SDP (Simple Dual Port) Mode + ECC

	Port A (W)	Port B (R)
Write Width	64	~
Read Width	~	64
Write Depth	512	~
Read Depth	~	512
Operating Mode	WRITE_FIRST	~
Output Register	~	Enabled
Reset Priority	~	~
Address Width	9	9

36 Kb Memory Array 64 DI DO 8 DOP DIP 15 RDADDR RDCLK RDEN REGCE SSR 8 WE 15 WRADDR WRCLK WREN UG473 c1 06 011414



35 Irene Degl'Innocenti | CMS L1 Silicon-based Tracking Trigger: Evaluation of Virtex 7 and Kintex UltraScale FPGA

1

2 clock cycles

#36K BRAMs

**Read Latency** 

# **SDP (Simple Dual Port) Mode**

SDP_BRAM_func_synth.wcfg								
								128
Name	Value	0 ns	20 ns _	40 ns	60 ns	80 ns	100 ns	120 ns
₩ wdock_s	1							
14 rdock_s	1						<b>FITTER</b>	
⊞ 📲 we_s[0:0]	1					1		
⊞ 📲 waddr_s[8:0]	000	◊	001	X 00:	z (	03 X	00	0
⊞ <b>%</b> din_s[63:0]	000000000000000000000000000000000000000	00000	00000000001	00000000	poooooz X		000000000000000000000000000000000000000	
⊞₩ raddr_s[8:0]	003		ບບບ	X	φ01 X	002		003
🖽 📲 dout_s[63:0]	000000000000000000000000000000000000000	000	0000000000000	X 0000	000000000001 X 40	00000040000002	00000	00000000 <mark>003</mark>
🍓 sbiterr_s	0							
🖫 dbiterr_s	0							
🖪 📲 rdaddrecc_s[8:0]	003		XXXX		001	002 🛛 🔪		003
₩ wdk_cyde	51	0/1/2/3/4/5/6/7	89~~~~~					
₩ rdk_cyde	110011							
We Testing	TRUE				TRUE			
1 CkPer	2500 ps				2500 ps			
1 TestLen	110111				110111			
U injectsbiterr_s	0							
¼ injectdbiterr_s	0							



# **ECC – Hamming code error correction**

Built-in Hamming code error correction option makes possible to detect up to two-bit errors or correct one-bit errors.

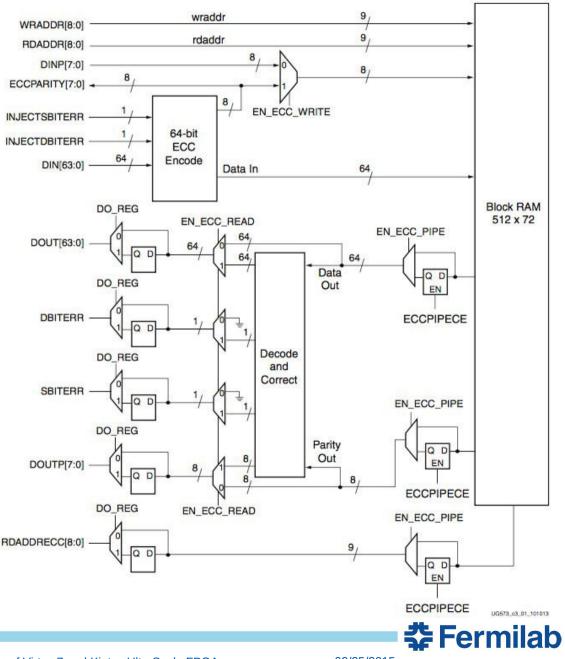
- Simple Dual-Port memory
- 64 word width RAM
- Encoder / Decoder



# **Top-level view of the ECC Architecture**

#### Modes:

- Standard mode
- Decoder-only mode
- Encoder-only mode
- SBITERR and DBITERR status bits
- Pipeline mode to improve maximum frequency
   ---->(more latency)



# Summing up

Technical evaluation of Xilinx FPGAs features can help in the future development of Pulsar II b to better achieve the high performance required:

- GTH tranceivers → High band width
- Block RAMs  $\rightarrow$  Huge storage data



# THANK YOU FOR YOUR ATTENTION!



40 Irene Degl'Innocenti | CMS L1 Silicon-based Tracking Trigger: Evaluation of Virtex 7 and Kintex UltraScale FPGA