



Flash Memory Summit

Connecting your ASIC or FPGA design to an OMI attached Memory

Bruno Mesnet, IBM Client Engineering
OpenCAPI - OMI enablement

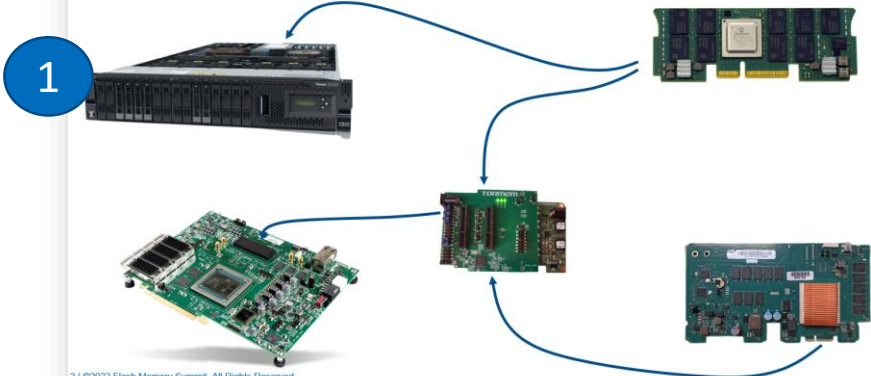


Just Simple and Open

Overview

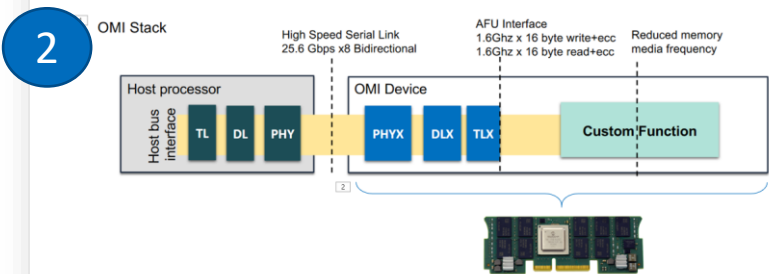


OMI DDIMM in its environment



3 | ©2022 Flash Memory Summit. All Rights Reserved.

OMI chain block diagram



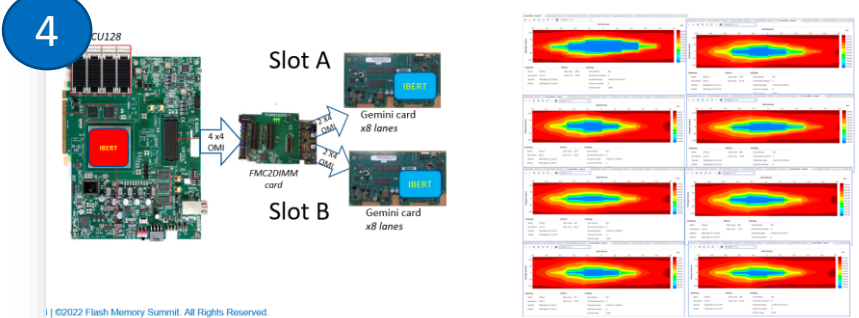
5 | ©2022 Flash Memory Summit. All Rights Reserved.



Checking OMI serial links integrity



- OMI link Xilinx standalone IBERT between host and device via Tormem adapter

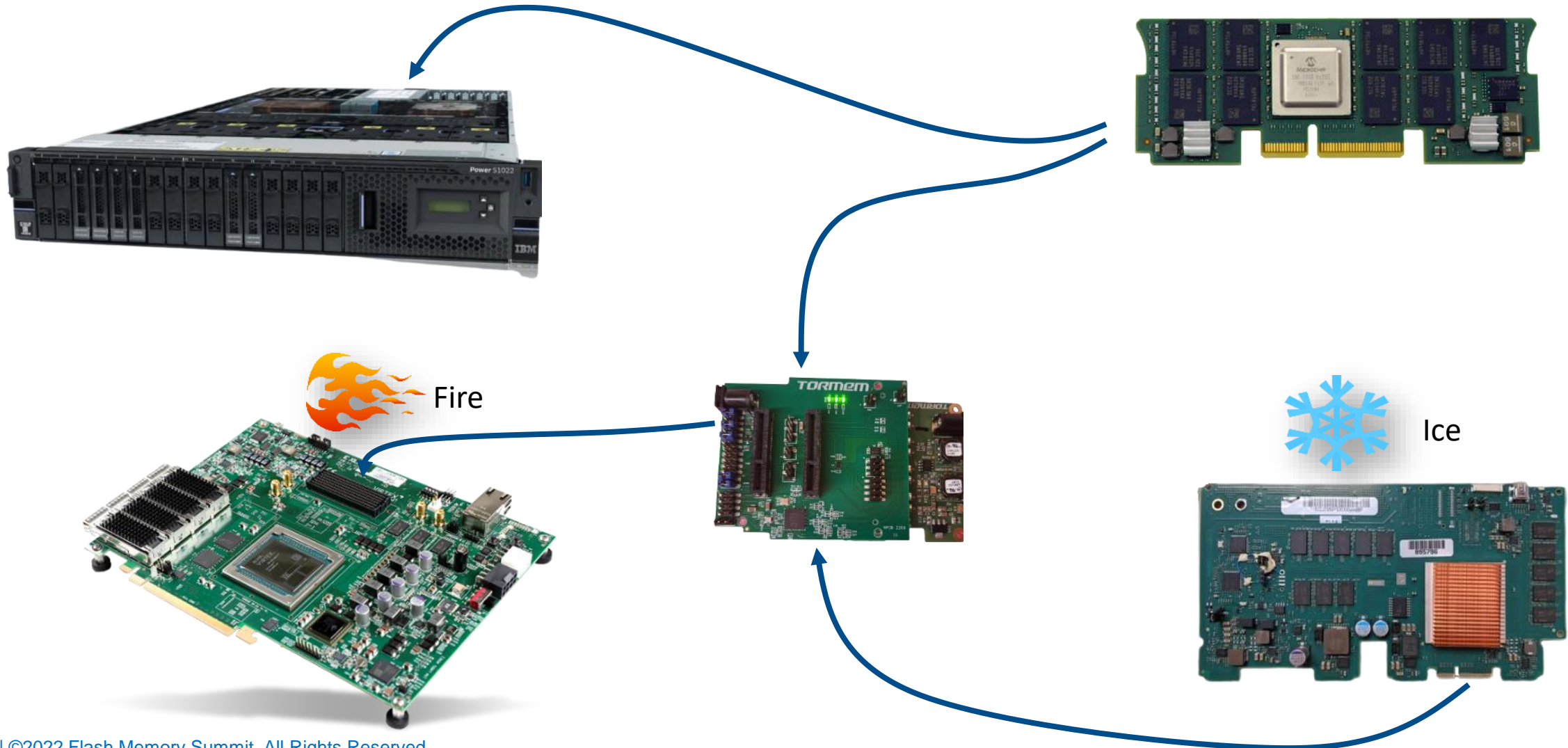


1 | ©2022 Flash Memory Summit. All Rights Reserved.

OMI DDIMM in its environment

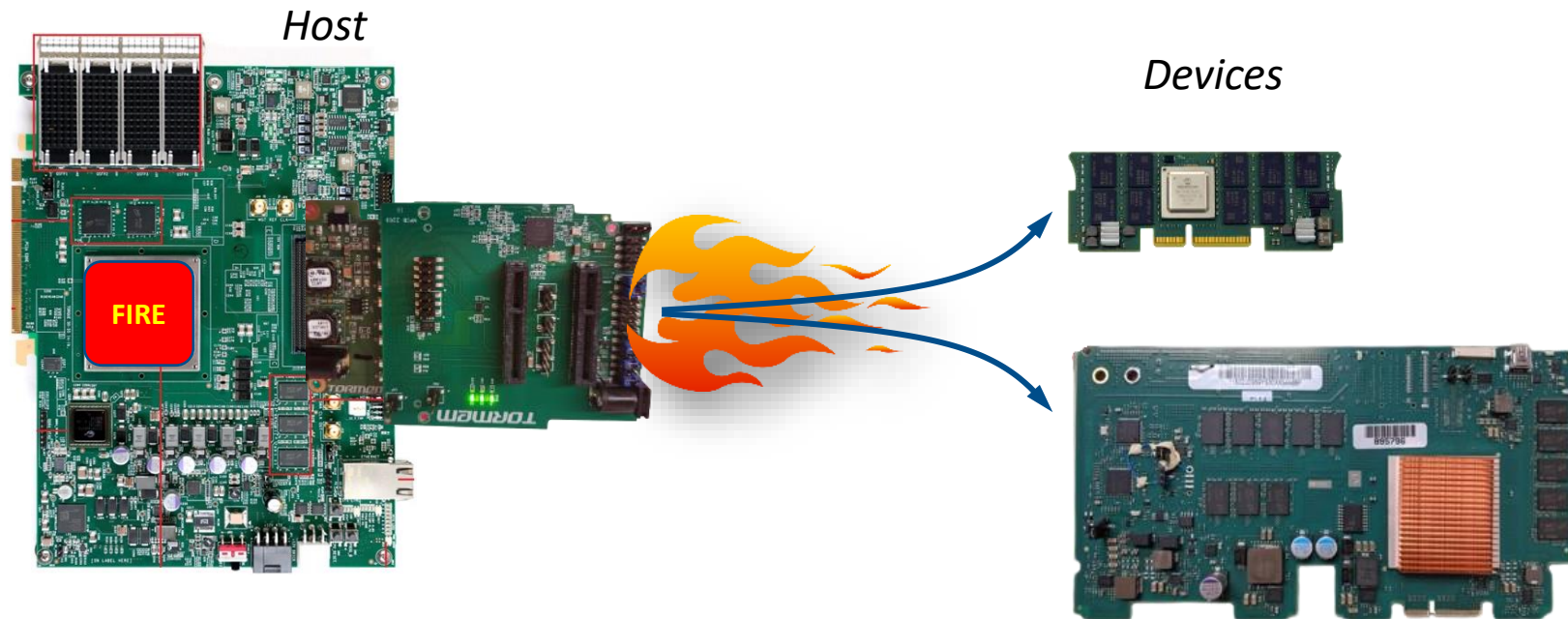


Flash Memory Summit

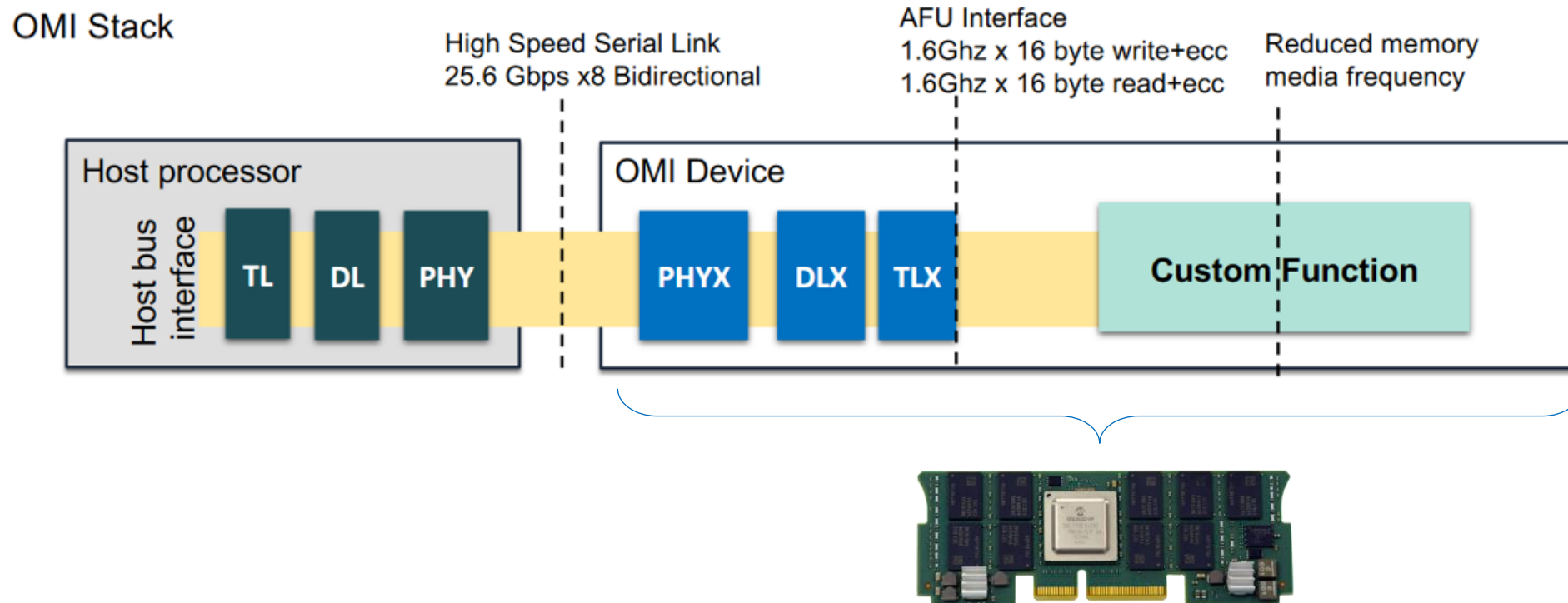


Fire implementation on VCU128 + OMI DDIMM

- check OMI link training
- Frames generators and integrity checkers available
- I2C control to read and write registers on host and device



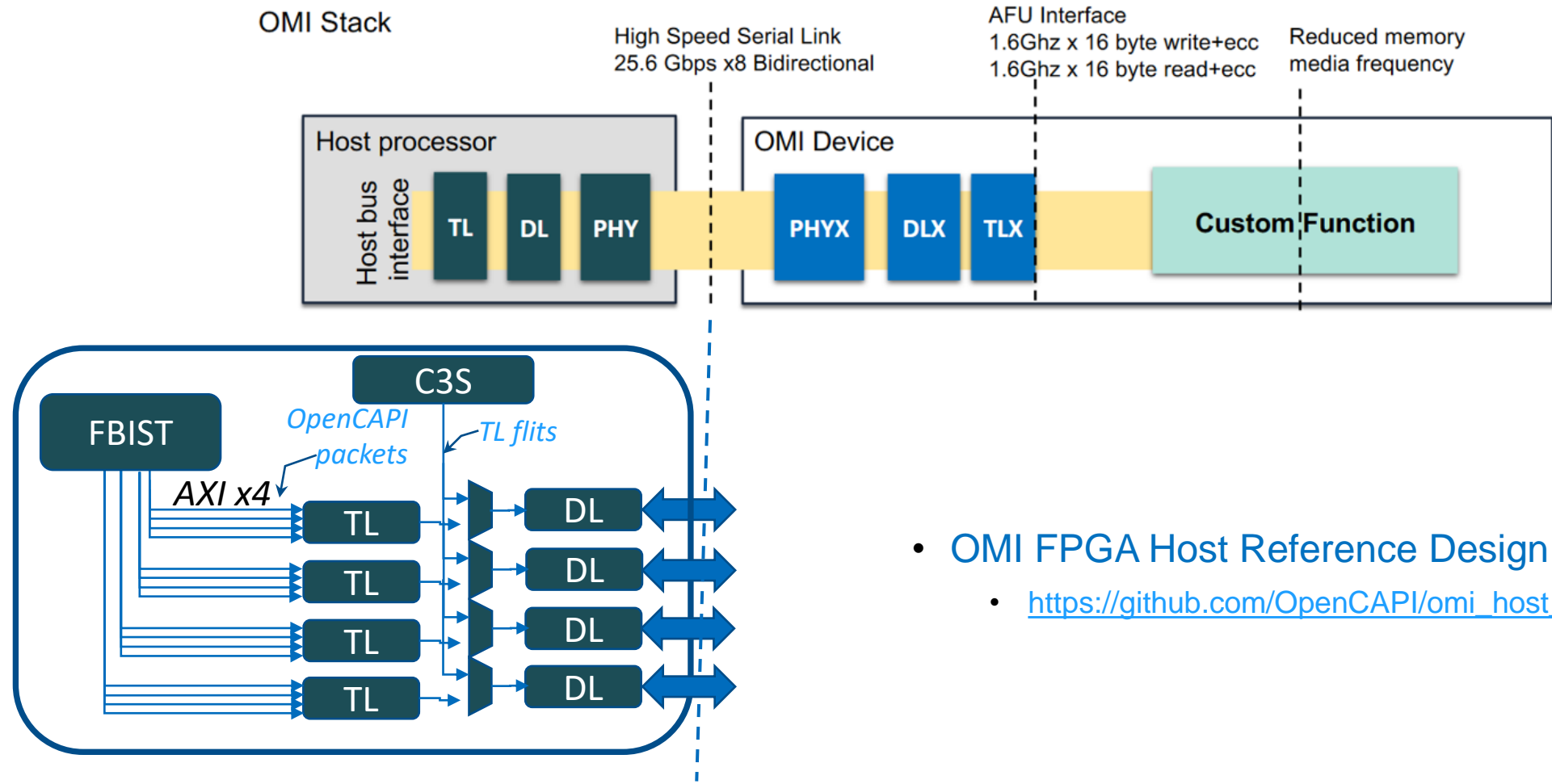
OMI chain block diagram



Host FPGA Block diagram



Flash Memory Summit



- OMI FPGA Host Reference Design
 - https://github.com/OpenCAPI/omi_host_fire

Table 5-1. DL frame format showing CRC and “bad data flit” coverage

Bytes(63:0)	
DL content	TL command/response/32-, 8-byte data content
Data flit 0	
Data flit 1	
Data flit 2	
Data flit 3	
Data flit 4	
Data flit 5	
Data flit 6	
Data flit 7	
DL content	TL command/response/32-, 8-byte data content
Data flit 0	
Data flit 1	
DL content	TL command/response/32-, 8-byte data content
DL content	TL command/response/32-, 8-byte data content

- **Control flits.** The control flit contains TL command/response content and DL content. The DL content contains several DL-generated subfields including the CRC that covers the control flit and any preceding data flits. There are fields in the DL content that are generated by the TL.
- **Data flits.** There are 0 to 8 data flits between each control flit.

C3S generator – full control



Flash Memory Summit

TL flit

```

programConfigRead
/*
 * Command Flit
 * Tl Template = 0, Slots 7 - 4
 *
 *      3      2      1      0
 *      10987654321098765432109876543210
 * 0x0F -----
 * 0x0E -----T-----D      T = TL Template | D = DRL
 * 0x0D -----
 * 0x0C -----
 * 0x0B -----
 * 0x0A -----
 * 0x09 -----
 * 0x08 -----
 * 0x07 -----
 * 0x06 PPP-CCCCCCCCCCCCAAAAA      P = pL | C = CAPPTag | A = PA
 * 0x05 AAAAAAAAAAAAAAAAAAAAAA      A = PA
 * 0x04 AAAAAAAAAAAAAAAAAAAAAA      A = PA
 * 0x03 -----O-----      O = Opcode (0b11100000) => E0=RD
 * 0x02 -----
 * 0x01 -----
 * 0x00 -----
 */
    
```

Write fire 010100000000000E	0000000000000000	- DRL=0 for read
Write fire 0101000000000006	0000000040000000	- Data - PL = 0b010
Write fire 0101000000000005	0000000000000000	- Data - PA[31:20]
Write fire 0101000000000004	000000000026C000	- Data - PA[19:0]
Write fire 0101000000000003	0000000000E00000	- Data - RD Opcode
Write fire 0101000000020000	0000000080000001	- INSTR - Next instr=1 - repeat 0
Write fire 0101000000020100	0000000000000100	- INSTR - Last instr
Read fire 0101000000030000	0000000000000008	- CNTL - START
Write fire 0101000000030000	0000000000000009	- CNTL - STOP
Read fire 0101000000030000	000000000000000A	- Data - 0x26C~64/4=B
Read fire 010100000001020B	0000000000000000	

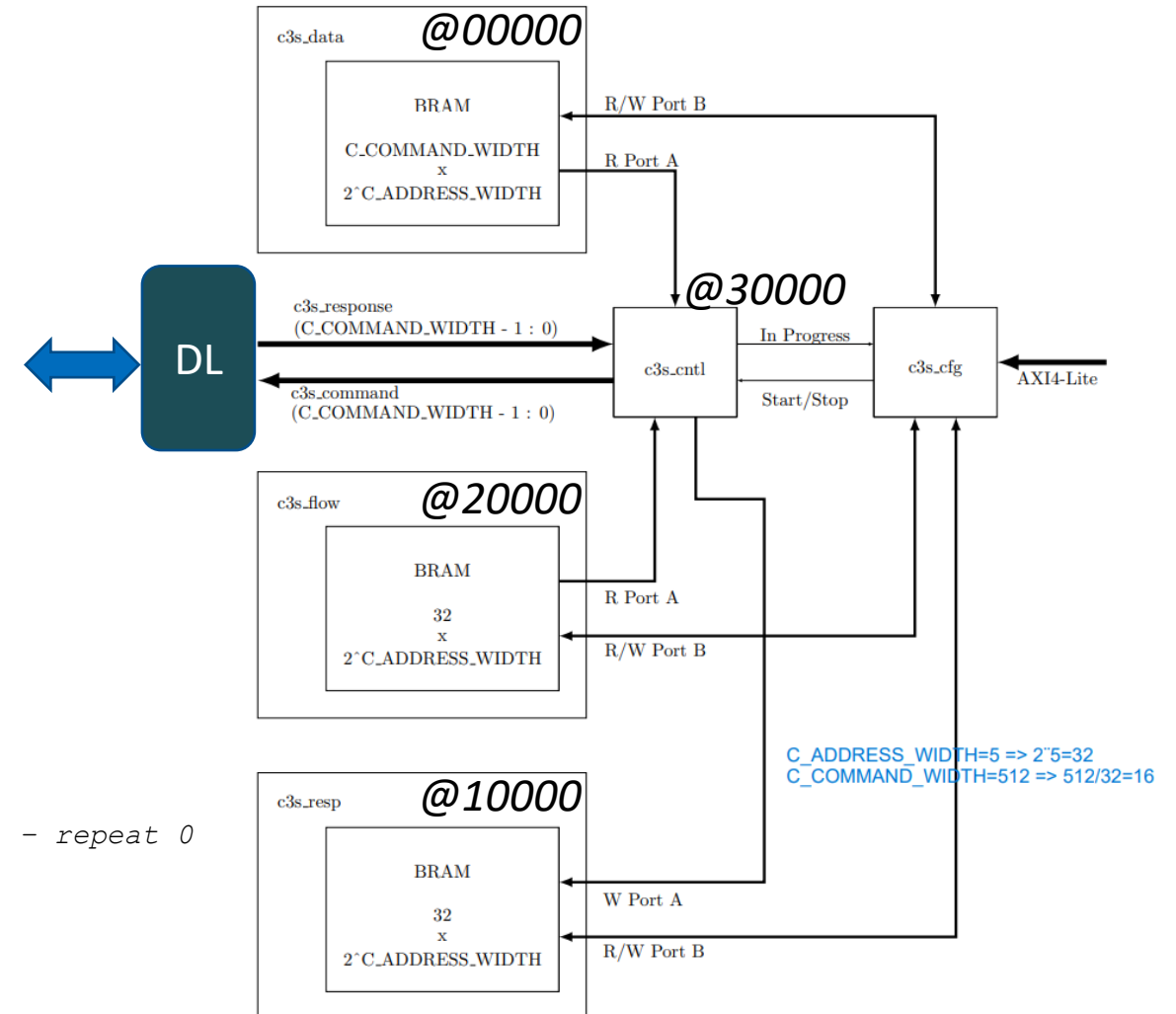
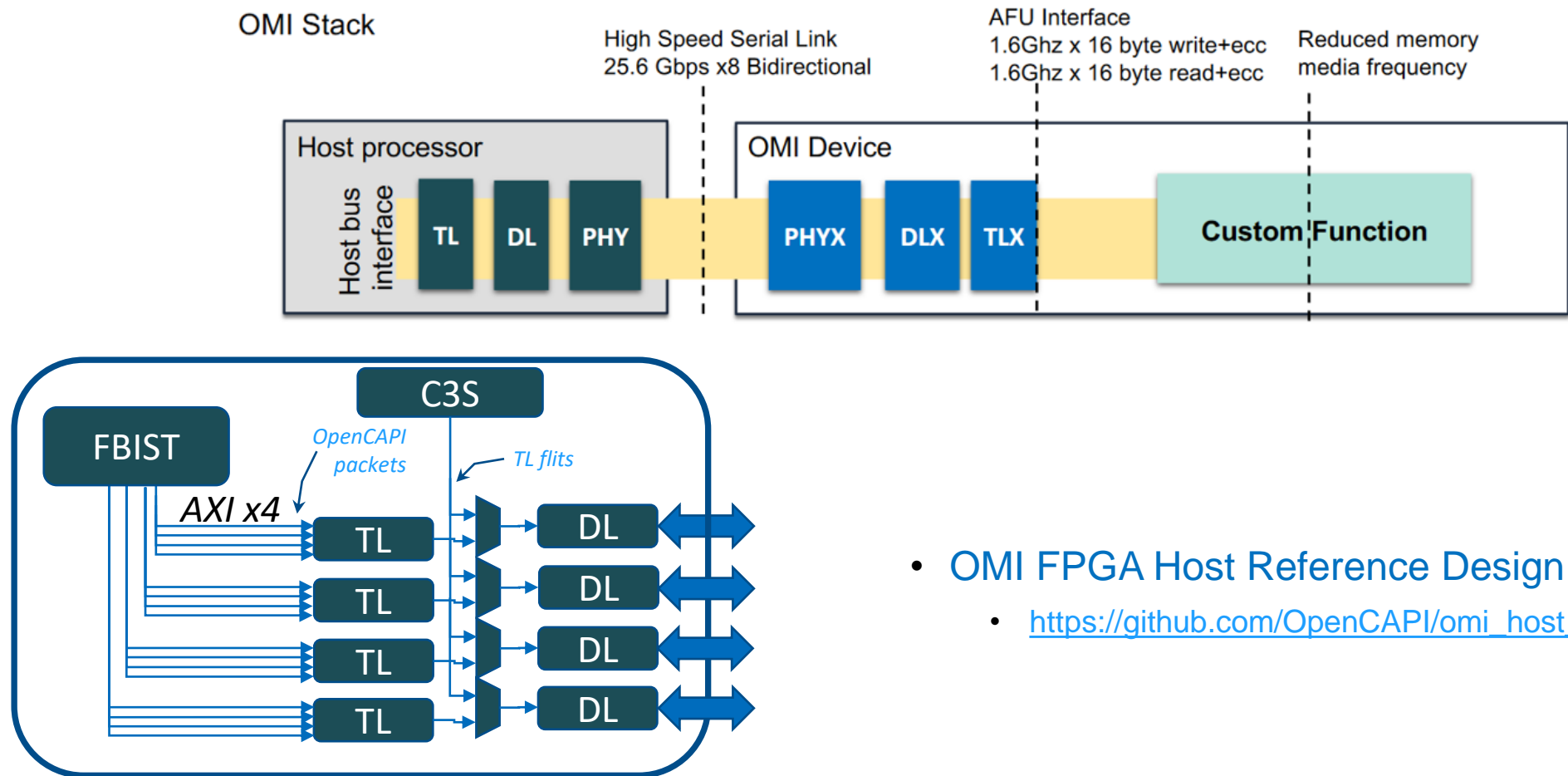


Figure 6: Overview of c3s_rlm.vhdl

Host FPGA Block diagram



Flash Memory Summit



- OMI FPGA Host Reference Design
 - https://github.com/OpenCAPI/omi_host_fire

FBIST generator



Flash Memory Summit

Command generators

Sequence of commands (cfg, arb, spc)

Each Cmd generator has 8 engines

Each engine is programmed with a single command type

Address generator

Adds address + tag

Each Add generator has 8 engines

Each engine is programmed with a single address type

Data generator

WR data based on previous address

```
echo "FBIST_POOL_0_ENGINE_0 - 0x01=@incrementing @" - 0x05=64B RD
write fire 0102000000000000 00000000000000000105
```

```
echo "-- start address for Pool 0 Engine x"
echo "FBIST POOL 0 ENGINE 0 ADDRESS START LOW"
putscom fire 010200000000000002C 0000000000000000
echo "FBIST POOL 0 ENGINE 0 ADDRESS START HIGH"
putscom fire 0102000000000000030 0000000000000000
```

```
echo "FBIST Data Pattern => repeating 0xA"
putscom fire 0102000000000000094 0000000033333333
```

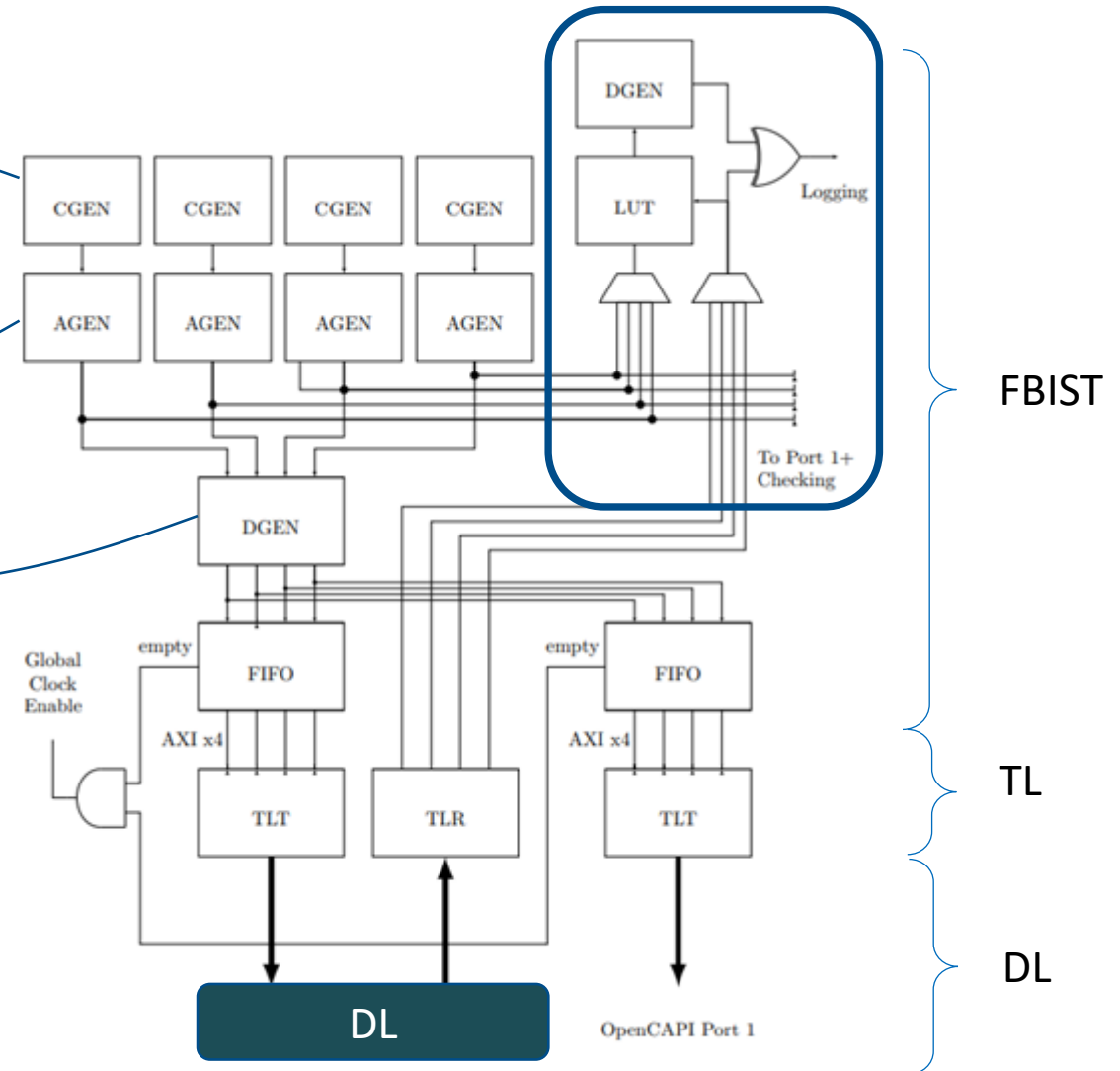
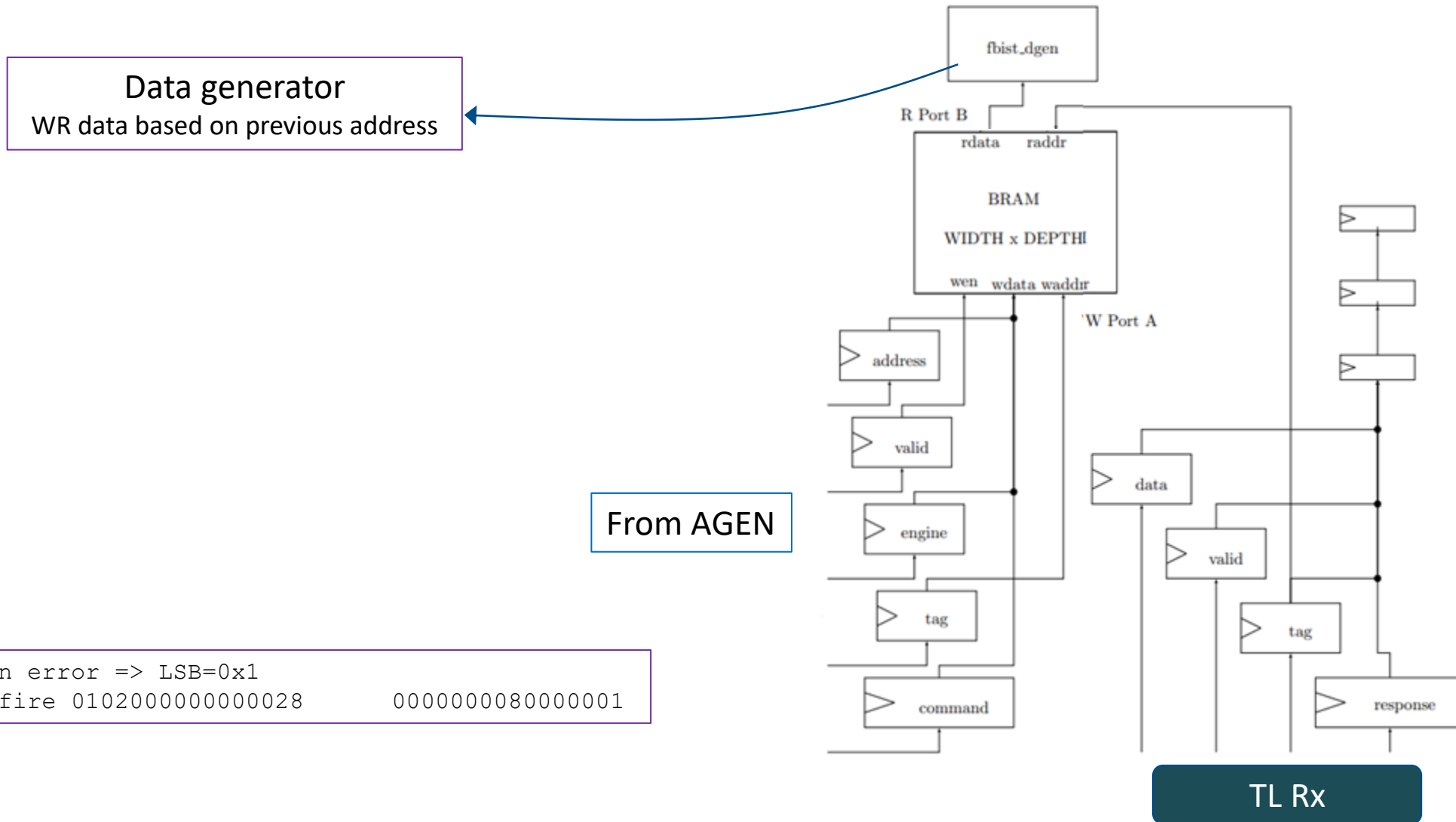


Figure 8: Overview of fbist_rlm.vhdl

FBIST checker



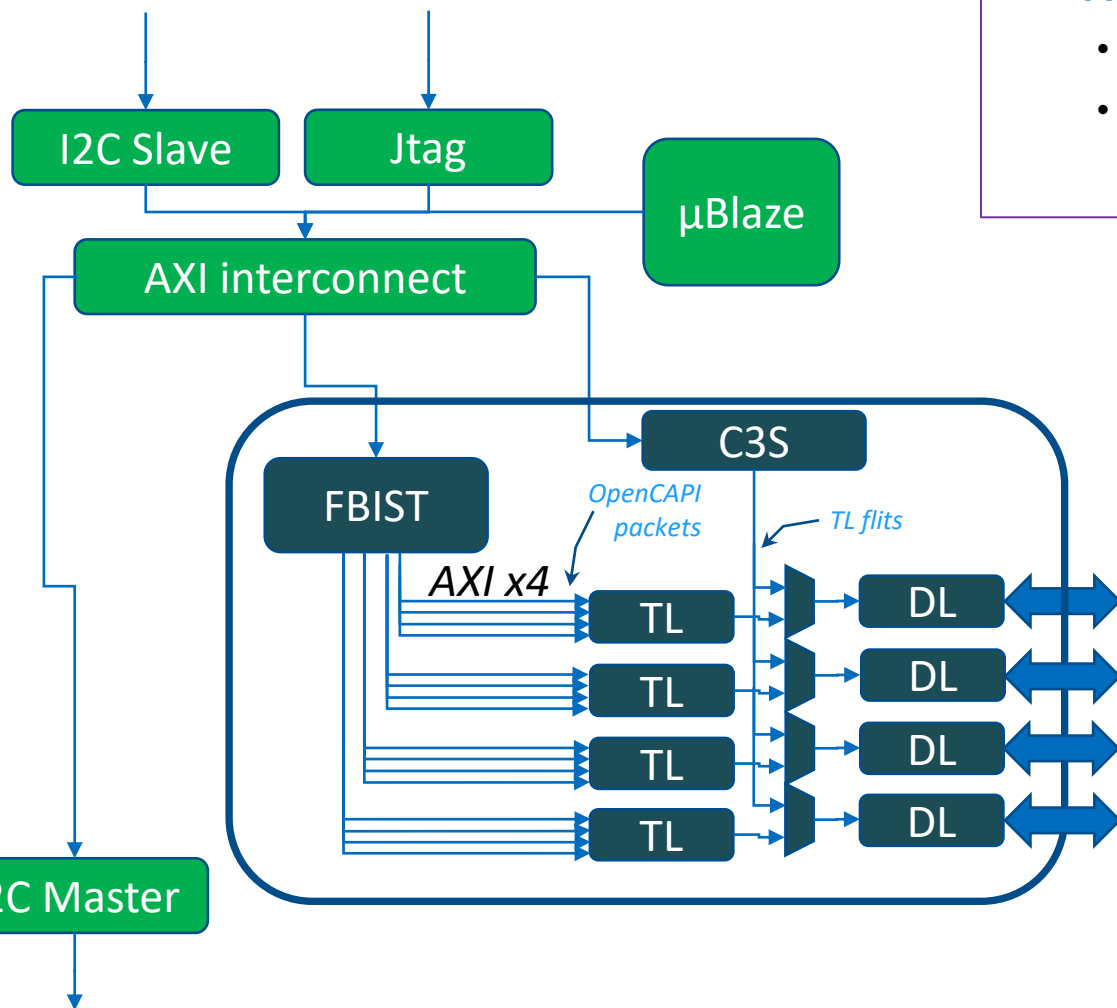
Flash Memory Summit



Memories full access



Flash Memory Summit



- Access to:
 - Host / Fire
 - Device / OMI DDIMM / Gemini

- Several types of access:
 - I2C
 - Jtag
 - inband registers access

Macro	Address Start	Address End	Size	Protocol	Interface
DDIMM 0 Memory	0x0000_0000_0000_0000	0x0000_03FF_FFFF_FFFF	4 TB	AXI3	M1
DDIMM 1 Memory	0x0000_0400_0000_0000	0x0000_07FF_FFFF_FFFF	4 TB	AXI3	M4
DDIMM 2 Memory	0x0000_0800_0000_0000	0x0000_0BFF_FFFF_FFFF	4 TB	AXI3	M7
DDIMM 3 Memory	0x0000_0C00_0000_0000	0x0000_0FFF_FFFF_FFFF	4 TB	AXI3	M10
DDIMM 0 CFG	0x0001_0000_0000_0000	0x0001_0000_7FFF_FFFF	2 GB	AXI3	M3
DDIMM 1 CFG	0x0001_0400_0000_0000	0x0001_0400_7FFF_FFFF	2 GB	AXI3	M6
DDIMM 2 CFG	0x0001_0800_0000_0000	0x0001_0800_7FFF_FFFF	2 GB	AXI3	M9
DDIMM 3 CFG	0x0001_0C00_0000_0000	0x0001_0C00_7FFF_FFFF	2 GB	AXI3	M12
DDIMM 0 MMIO	0x0001_0001_0000_0000	0x0001_0001_7FFF_FFFF	2 GB	AXI3	M2
DDIMM 1 MMIO	0x0001_0401_0000_0000	0x0001_0401_7FFF_FFFF	2 GB	AXI3	M5
DDIMM 2 MMIO	0x0001_0801_0000_0000	0x0001_0801_7FFF_FFFF	2 GB	AXI3	M8
DDIMM 3 MMIO	0x0001_0C01_0000_0000	0x0001_0C01_7FFF_FFFF	2 GB	AXI3	M11
FML	0x0100_0000_0000_0000	0x0100_0000_00FF_FFFF	16 MB	AXI4-Lite	M0
DDIMM 0 C3S	0x0101_0000_0000_0000	0x0101_0000_00FF_FFFF	16 MB	AXI4-Lite	M1
DDIMM 1 C3S	0x0101_0400_0000_0000	0x0101_0400_00FF_FFFF	16 MB	AXI4-Lite	M9
DDIMM 2 C3S	0x0101_0800_0000_0000	0x0101_0800_00FF_FFFF	16 MB	AXI4-Lite	M11
DDIMM 3 C3S	0x0101_0C00_0000_0000	0x0101_0C00_00FF_FFFF	16 MB	AXI4-Lite	M13
DDIMM 0 FBIST	0x0102_0000_0000_0000	0x0102_0000_00FF_FFFF	16 MB	AXI4-Lite	M2
DDIMM 1 FBIST	0x0102_0400_0000_0000	0x0102_0400_00FF_FFFF	16 MB	AXI4-Lite	M10
DDIMM 2 FBIST	0x0102_0800_0000_0000	0x0102_0800_00FF_FFFF	16 MB	AXI4-Lite	M12
DDIMM 3 FBIST	0x0102_0C00_0000_0000	0x0102_0C00_00FF_FFFF	16 MB	AXI4-Lite	M14
AXI IIC	0x0103_0000_0000_0000	0x0103_0000_00FF_FFFF	16 MB	AXI4-Lite	M3
DDIMM 0 Host Config	0x0104_0000_0000_0000	0x0104_0000_00FF_FFFF	16 MB	AXI4-Lite	M4
DDIMM 1 Host Config	0x0104_0400_0000_0000	0x0104_0400_00FF_FFFF	16 MB	AXI4-Lite	M6
DDIMM 2 Host Config	0x0104_0800_0000_0000	0x0104_0800_00FF_FFFF	16 MB	AXI4-Lite	M7
DDIMM 3 Host Config	0x0104_0C00_0000_0000	0x0104_0C00_00FF_FFFF	16 MB	AXI4-Lite	M8
System Monitor	0x0105_0000_0000_0000	0x0105_0000_00FF_FFFF	16 MB	AXI4-Lite	M5
ISDIMM	0x0300_0000_0000_0000	0x0300_001F_FFFF_FFFF	128 GB	AXI3	M22

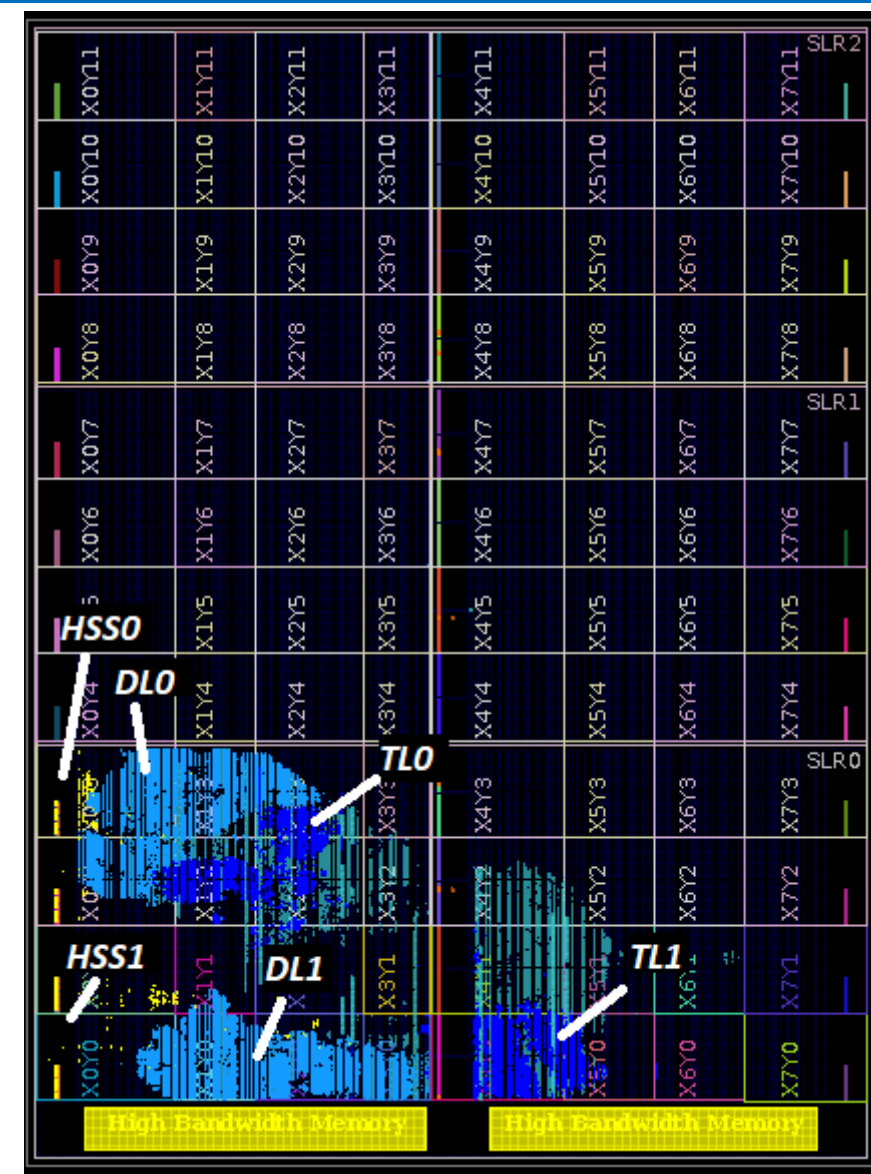
DL – TL FPGA implementation



Flash Memory Summit

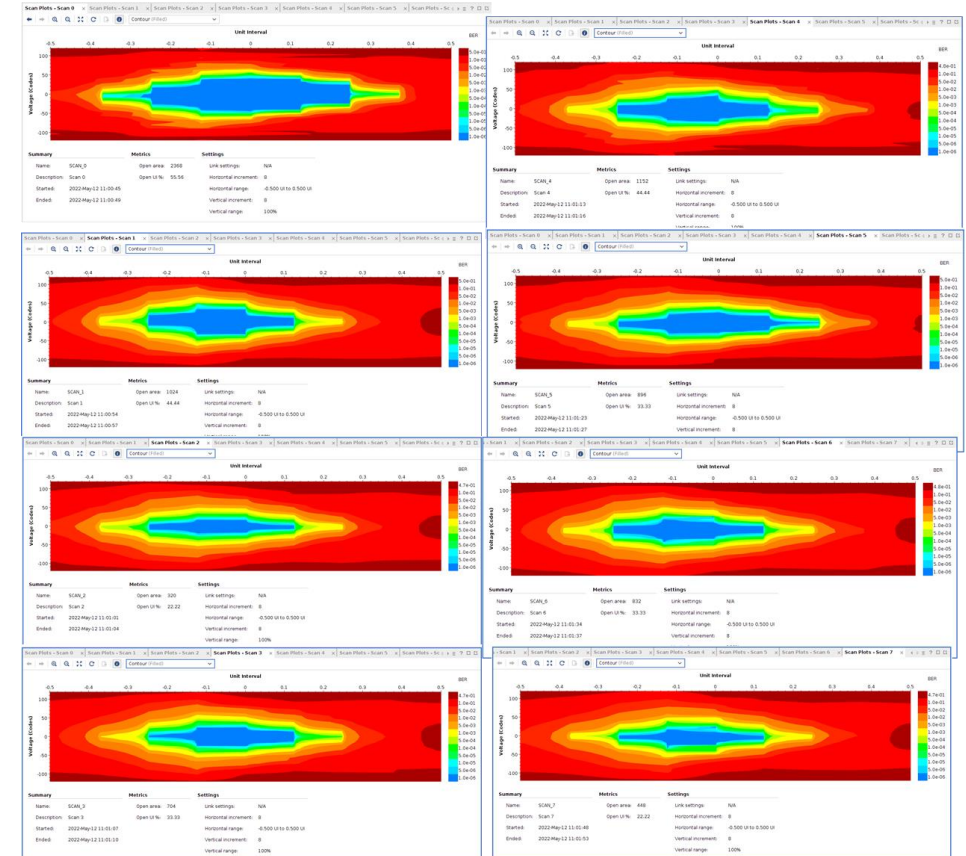
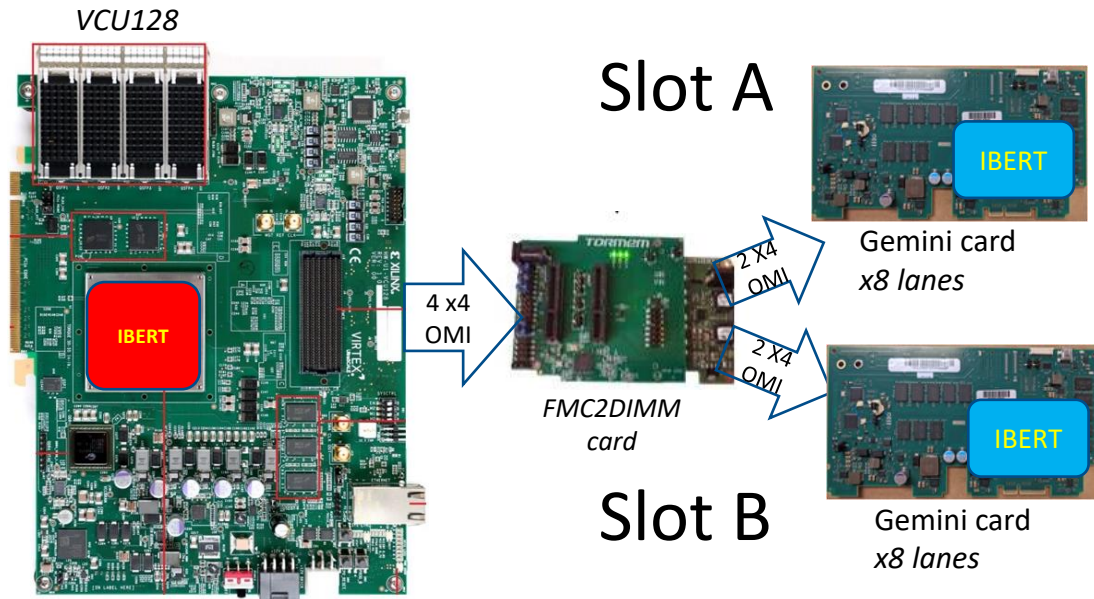
hss0		dl		tl	
Primitive Statistics		Primitive Statistics		Primitive Statistics	
Primitive type	Count	Primitive type	Count	Primitive type	Count
ADVANCED/GT	10	BLOCKRAM/BRAM	8	BLOCKRAM/BRAM	17
CLB/CARRY	8	CLB/CARRY	38	CLB/CARRY	16
CLB/LUT	261	CLB/LUT	23199	CLB/LUT	8401
CLOCK/BUFFER	1	CLB/MUXF	69	CLB/LUTRAM	2448
I_O/INPUT_BUFFER	2	REGISTER/SDR	13484	CLB/MUXF	98
REGISTER/SDR	426			REGISTER/SDR	6941

*VU37P contains 2,852 k System logic cells
(2,607k CLB Flip-Flops / 1,304K CLB LUTs)*



Checking OMI serial links integrity

- OMI link Xilinx standalone IBERT between host and device via Tormem adapter





Just Simple and Open

Questions



Flash Memory Summit