Virtual Prototyping in SpaceFibre System-on-Chip Design

Ilya Korobkov, Junior Researcher,

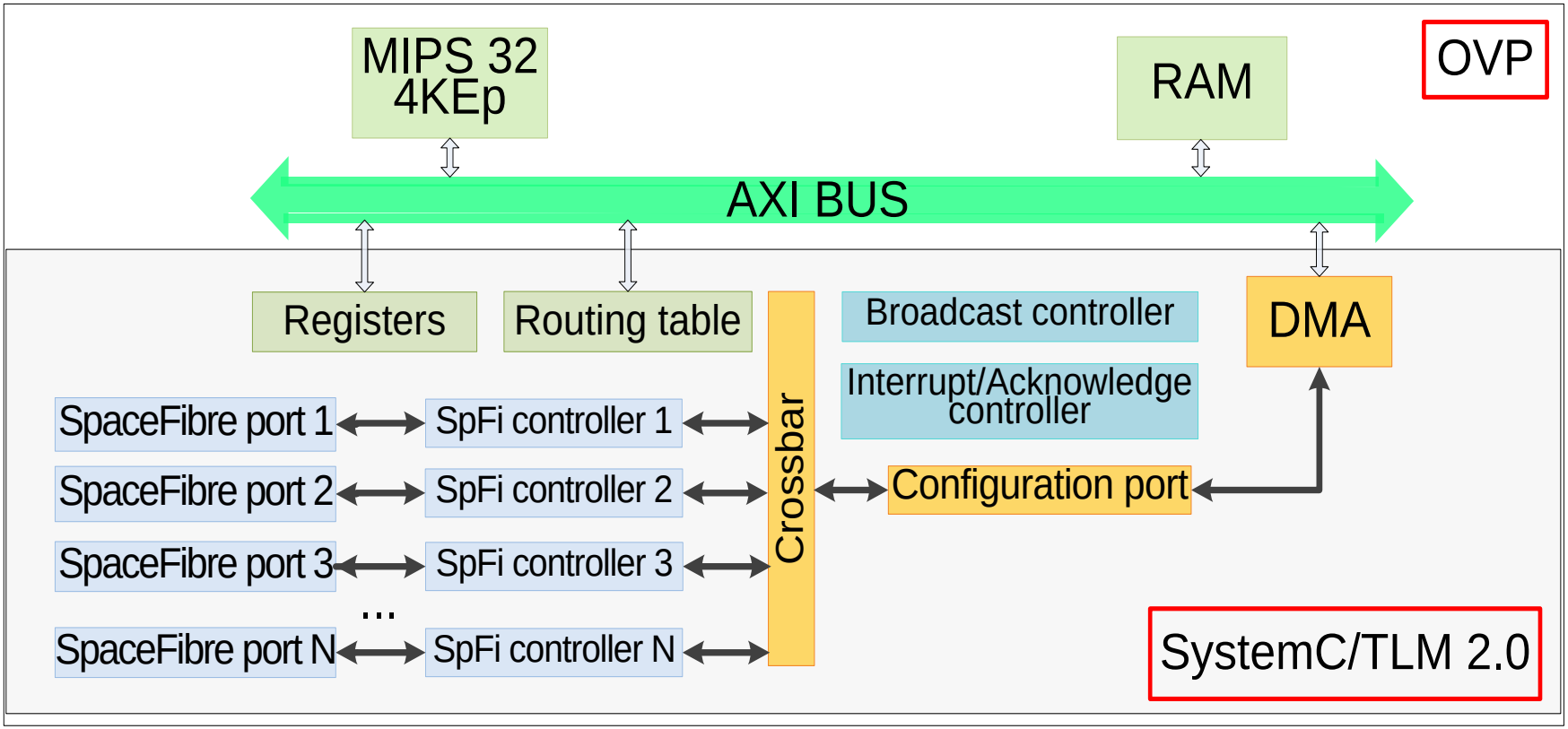
Saint-Petersburg State University of Aerospace Instrumentation

accellera SYSTEMS INITIATIVE

SUAI

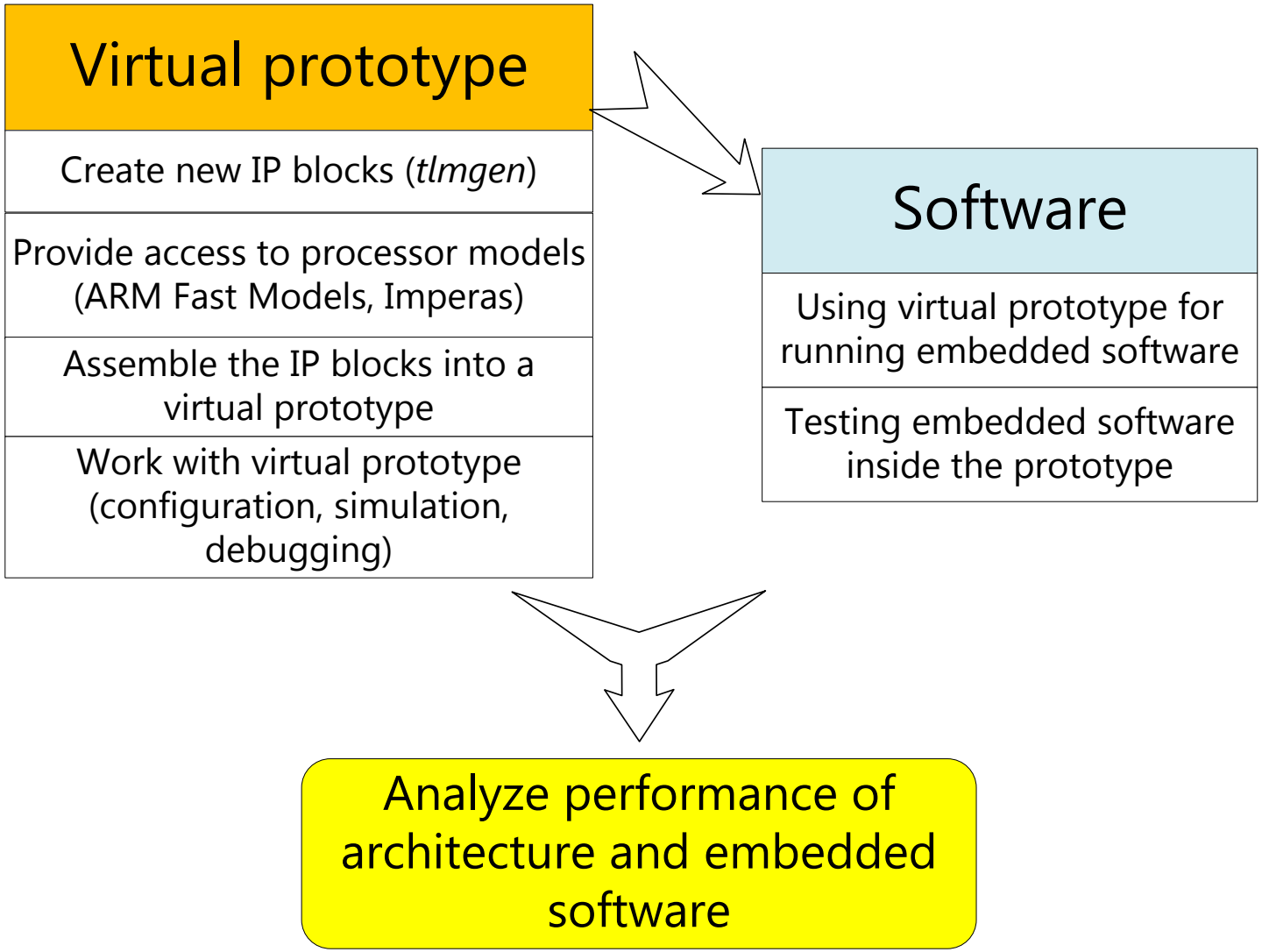
2015 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE

Network Memory Controller



*SpaceFibre – abridgement SpFi

VSP: Cadence Virtual System Platform



tlmgen tool

- Generates TLM module templates with memory mapped registers
- Helps to avoid common errors as naming inconsistencies, register overlapping, and illegal register accesses by the software
- Speeds up development time

There are two forms of input data:

- simple register definition language (simpleRDL)
- IP-XACT XML

RDFfile – Register Description File

It is input data for *tlmgen* on simpleRDL language

It consists of:

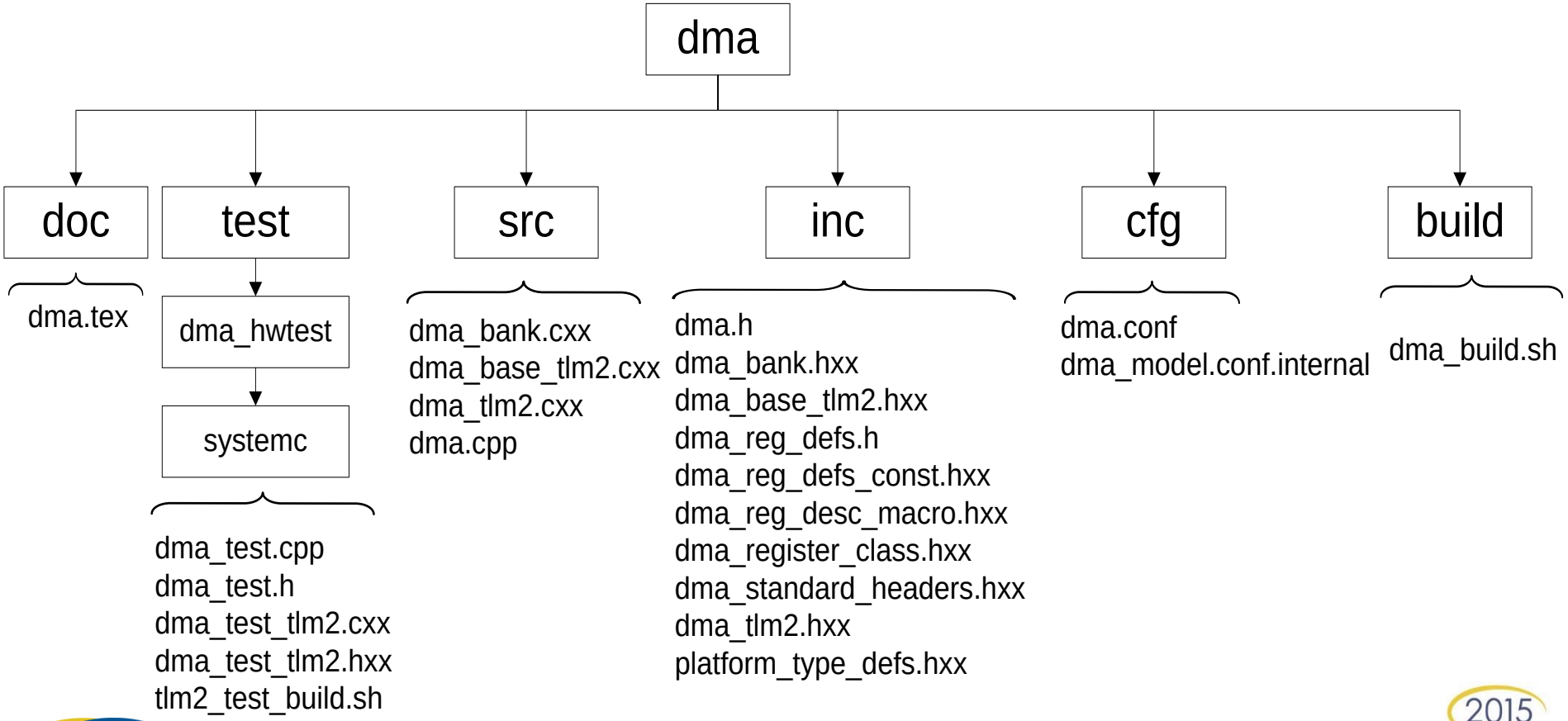
- description of registers: fields and register banks
- ports specification
- parameter specification

```
REG DMA_AREA_SIZE_R4 {
    /* Register description */
    ACCESS = rw;
    REGWIDTH = 32;
    RESET = 0x0000;

    FIELD { ACCESS = rw; } DMA_AREA_SIZE_R4 [31:0];
}
REG SpF_PORT_MODE_VC_PARAMS {
    /* Register description */
    ACCESS = rw;
    REGWIDTH = 32;
    RESET = 0x0000;

    FIELD { ACCESS = rw; } VC_LNUM [7:0];
    FIELD { ACCESS = rw; } VC_THROUGHPUT [15:8];
    FIELD { ACCESS = rw; } VC_PRIORITY [18:16];
    FIELD { ACCESS = rw; } VC_WORK_EN [19:19];
    FIELD { ACCESS = rw; } CREDIT_OVERFLOW [20:20];
    FIELD { ACCESS = rw; } DATA_OVERFLOW [21:21];
}
```

```
REGISTER_BANK dmaRegs {
    /*Register bank definition*/
    SIZE = 0x4dc;
    //Name of the busport generated
    BUSPORTNAME = cpu_access_target;
    //buswidth of the cpu_access_target
    BUSPORTWIDTH = 32;

    REG DMA_REGIONS_12_STATUS   0x10 ;
    REG DMA_REGIONS_34_STATUS   0x14 ;
    REG DMA_START_ADDR_R1   0x28 ;
    REG DMA_AREA_SIZE_R1    0x4c ;
    REG DMA_START_ADDR_R2   0x70 ;
    REG DMA_AREA_SIZE_R2    0x94 ;
    REG DMA_START_ADDR_R3   0xb8 ;
    REG DMA_AREA_SIZE_R3    0xdc ;
    REG DMA_START_ADDR_R4   0x100 ;
    REG DMA_AREA_SIZE_R4    0x124 ;
    REG SpF_PORT_MODE_VC_PARAMS 0x1e4 ;
    REG SpF_PORT_MODE_VC_TSLOTS_L   0x214 ;
    REG SpF_PORT_MODE_VC_TSLOTS_H   0x244 ;
    REG SpF_PORT_STATUS_VC1 0x294 ;
    REG INCORRECT_VC_INFO   0x414 ;
}
```

Result of *tlmgen* using

TLM-2.0 module template was generated for "DMA"

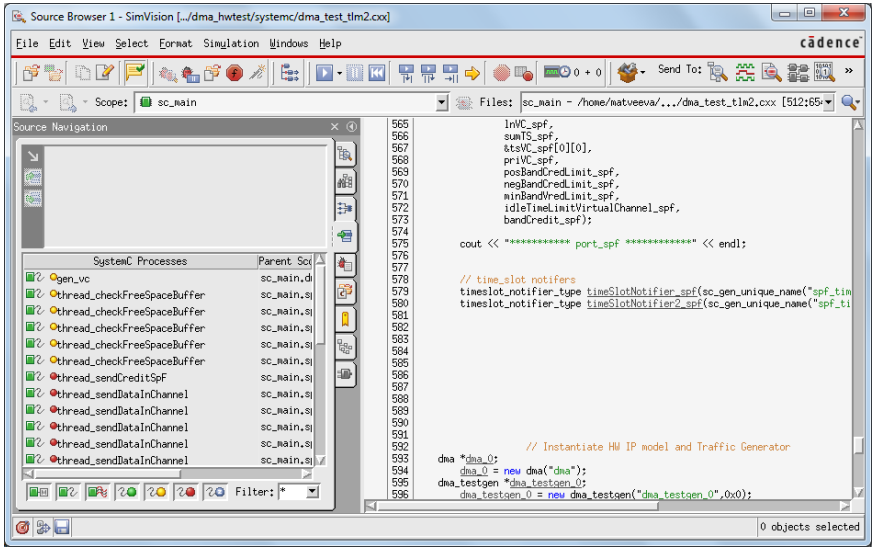


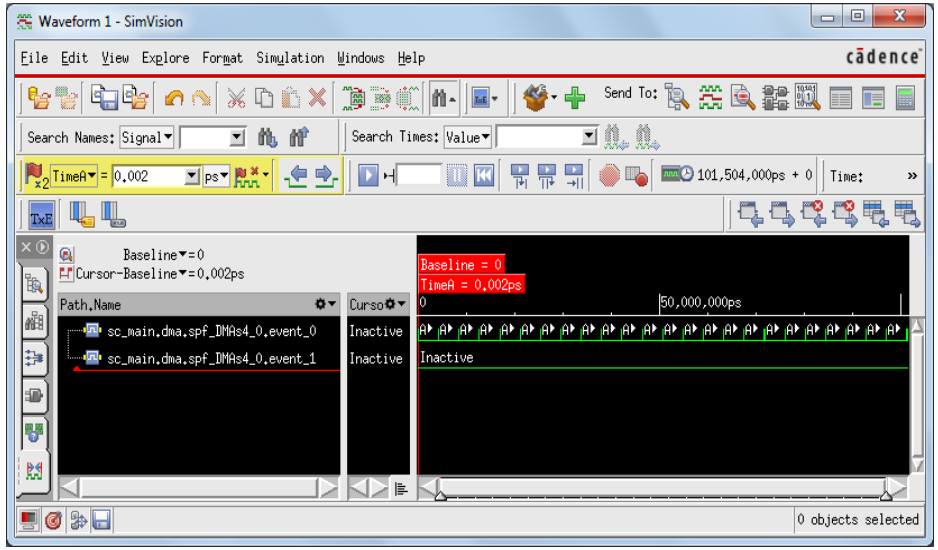
Benefits of *tlmgen*

- Input data format is simple and clear
- You can add new sockets, classes, threads and processes to template
- It provides classes for register description
- It decreases time for IP block development
- It helps to avoid common errors as naming inconsistencies, register overlapping

SimVision for debugging code

- Breakpoints can be created in objects, functions, processes
- Step-by-step simulation and debugging
- Waveforms
- Viewer of the variables, data members, call stack, processes
- …





Performance Analyzing (1/3)

- Static information about your SystemC design
- Simulation run-time information (dynamic)
- More than 10 performance metrics: throughput, TLM response status, read/write accesses, bytes transferred …

It is useful for finding performance bottlenecks!

```
Statistics for initiator: spf_frameGeneratorContr_0.txDataGenerator_1.socket_dg_crb_data_i_0

start time          0 s
end time            1 ms
delta time          1 ms

total read transactions          0
total write transactions        57
total bytes read                 0
total bytes written          58368

Transaction Responses:
TLM_OK_RESPONSE                         56
TLM_INCOMPLETE_RESPONSE                  0
TLM_GENERIC_ERROR_RESPONSE               0
TLM_ADDRESS_ERROR_RESPONSE               0
TLM_COMMAND_ERROR_RESPONSE               0
TLM_BURST_ERROR_RESPONSE                 0
TLM_BYTE_ENABLE_ERROR_RESPONSE           0

throughput                58.37 MB/sec
```

Performance Analyzing(2/3)

Resource usage

CPU and Memory

```
*** Resource and Simulation Information ***
ncsim: 14.10-p001: (c) Copyright 1995-2014 Cadence Design Systems, Inc.
Linux Localhost1 2.6.32-71.el6.x86_64 #1 SMP Fri May 20 03:51:51 BST 2011 x8
        model name:Intel(R) Xeon(R) CPU X7560 @ 2.27GHz
        cache size:24576 KB
Resource usage (from beginning of simulation to the time of this output):
Memory Usage - 28.0M program + 89.7M data + 1.0M profile = 118.7M total
CPU Usage - 8.5s system + 3.6s user = 12.1s total (1.5% cpu)

Profiled Intervals:
                    #start           #stop
[1]                   0 NS  to       101504 NS
```

/home/matveeva/DMA_TerminalNodeV6/test/dma_hwtest/systemc/ncprof.out - matveeva@194.226.199.26 - Editor - ...

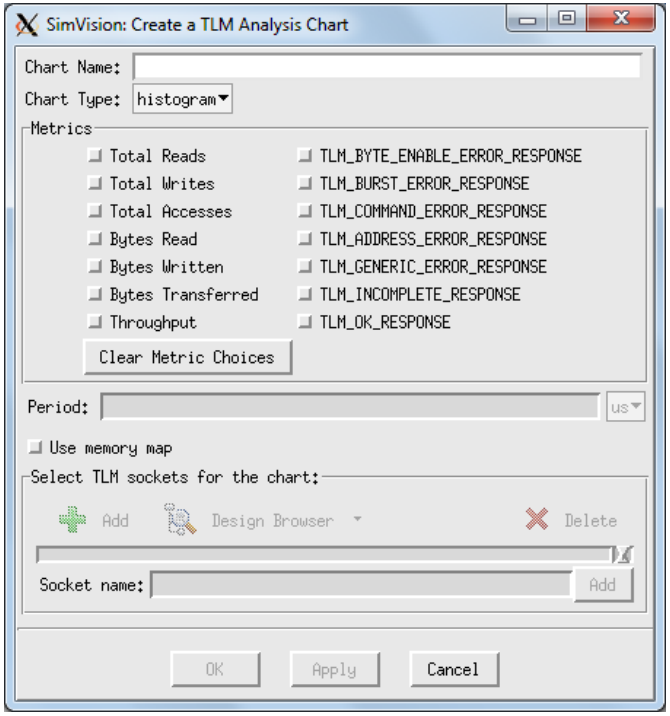
```
Most Active Modules (behavioral)
-------------------------------------------------------------
%hits #hits  #inst  name

-------------------------------------------------------------
SystemC Summary Counts (329 hits total)
-------------------------------------------------------------
%hits #hits  #inst  name
 99.7   328 [     ] SystemC process execution (5.5% of total)
                     23.1     76 scUserProcess ([tx_data_generator_type]spf_frameGeneratorCc
                     22.8     75 scUserProcess ([tx_data_generator_type]spf_frameGeneratorCc
                     16.7     55 scUserProcess ([tx_data_generator_type]spf_frameGeneratorCc
                     14.9     49 scUserProcess ([tx_data_generator_type]spf_frameGeneratorCc
                     14.3     47 scUserProcess ([tx_data_generator_type]spf_frameGeneratorCc
                      3.6     12 scUserProcess ([selector_4_type]spf_port_0.port_output_0.se
                      2.7      9 scOther (scKernel)
                      0.3      1 scUserProcess ([creditbuffer_type]spf_port_0.port_output_0.
                      0.3      1 scUserProcess ([dma_selector_4_type]dma.spf_DMAs4_0.thread_
                      0.3      1 scUserProcess ([dma]dma.b2nb_thread_2)
                      0.3      1 scUserProcess ([creditbuffer_type]spf_port_0.port_input_0.c
                      0.3      1 scUserProcess ([creditbuffer_type]spf_port_0.port_input_0.c
  0.3     1 [     ] SystemC shm event (0.0% of total)
                      inf     1 scOther (scKernel)

-------------------------------------------------------------
Stream Type Summary Counts (5962 hits total)
-------------------------------------------------------------
%hits #hits  #inst  name
 88.6  5280 [     ] Outside engine
```

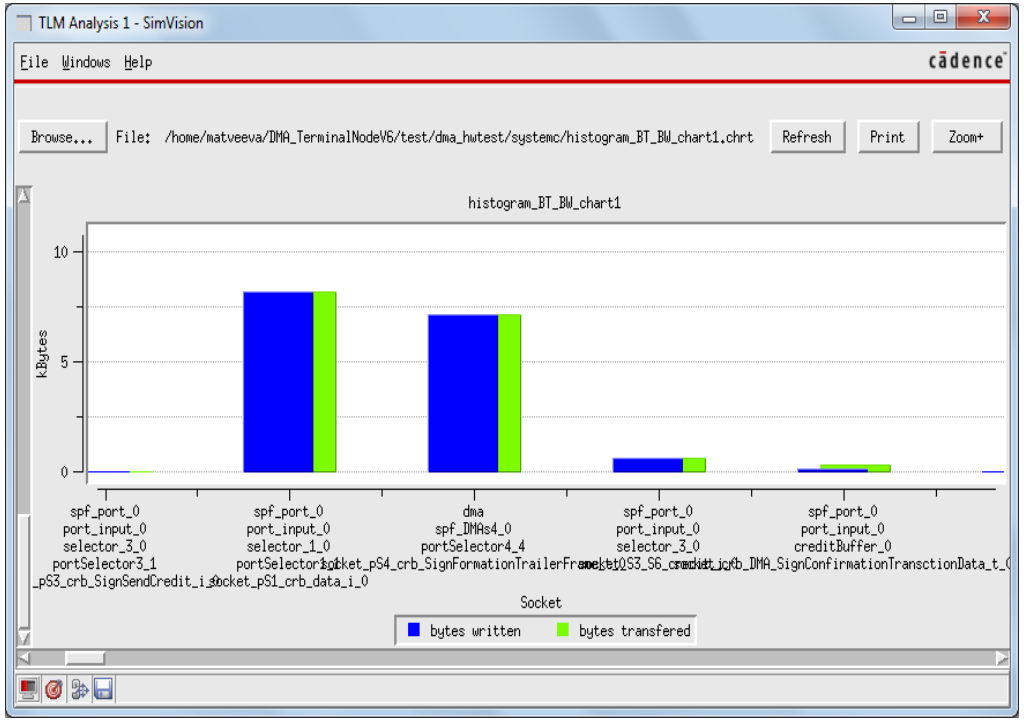
Change file encoding

Run-time distribution of resources between processes

Performance Analyzing(3/3)

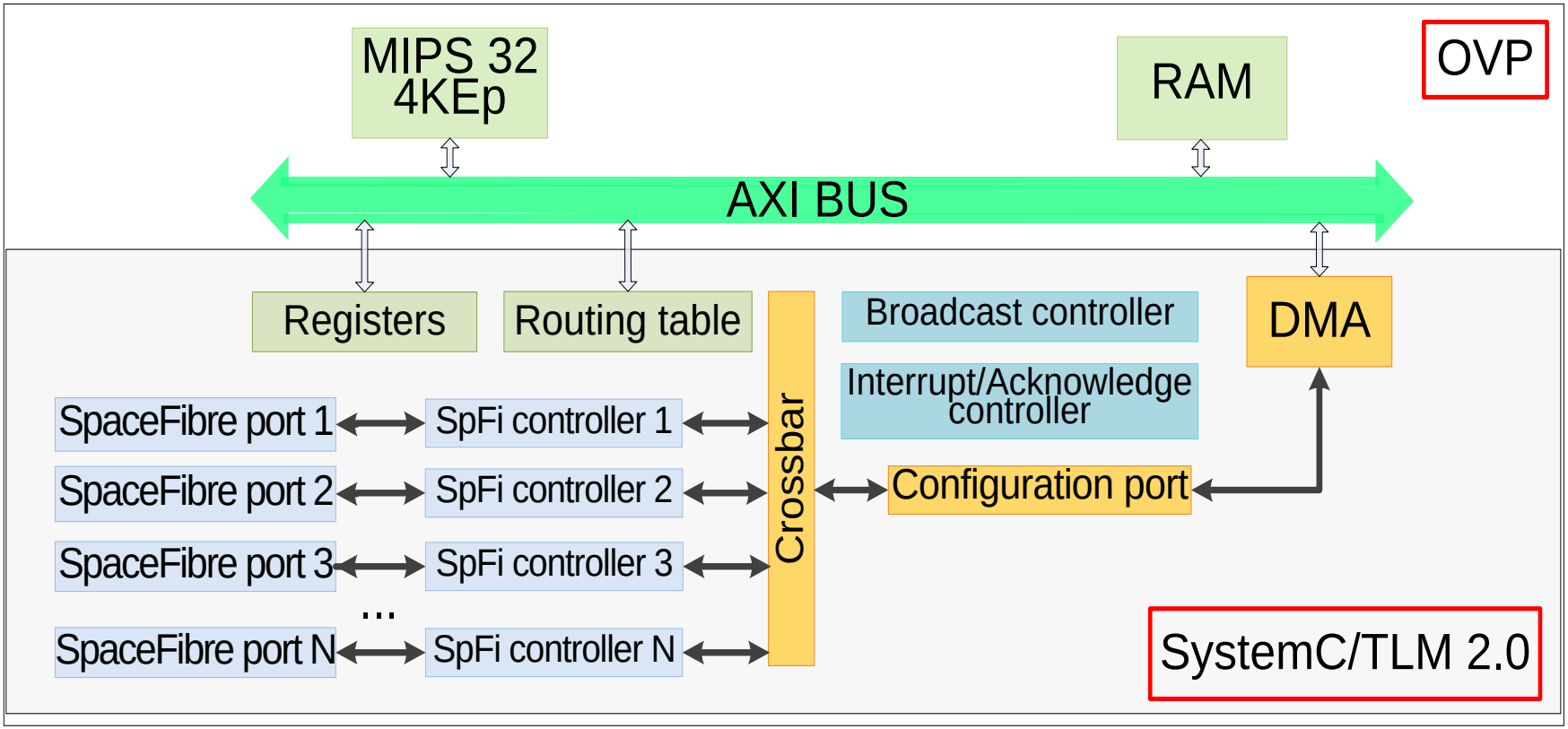
Set chart options to show simulation results





Plots with required parameters

Network Memory Controller



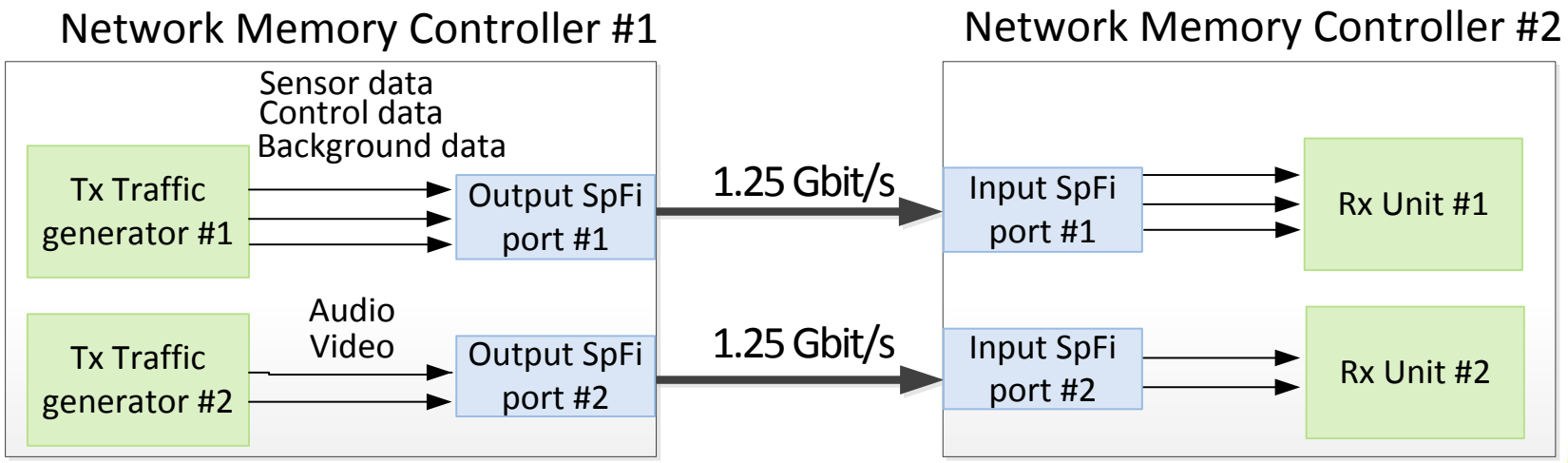
First results:

Performance Testing of SpaceFibre ports for Streaming Data

Streaming via SpaceFibre port

To test performance of SpFi ports for streaming we used traffics:

- Video: SVGA RGB 60Hz, 1.42Mbytes, accepted latency < 16 ms
- Audio: Kodec G.711, 160 bytes, 0.05 packet/ms, accepted latency < 100 ms
- Sensor data: 1Kbytes, 50 packet/ms, accepted latency < 5 ms
- Control data: 260 bytes, 0.02 packet/ms, accepted latency < 0.1 ms
- Background: 1Kbytes, 25 packet/ms, accepted latency – not defined

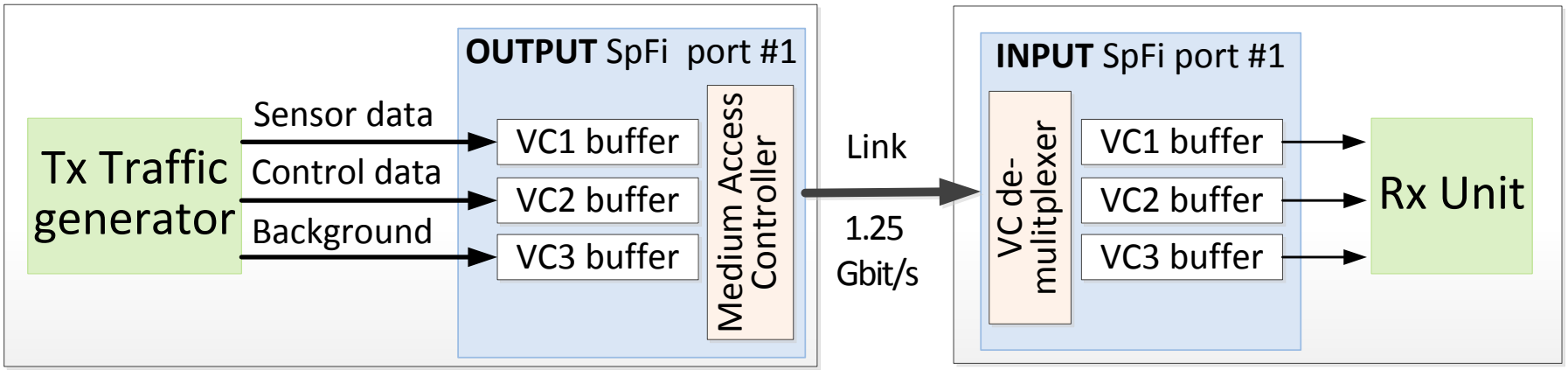


SpaceFibre port model

Port consists of input and output interfaces

Port includes:

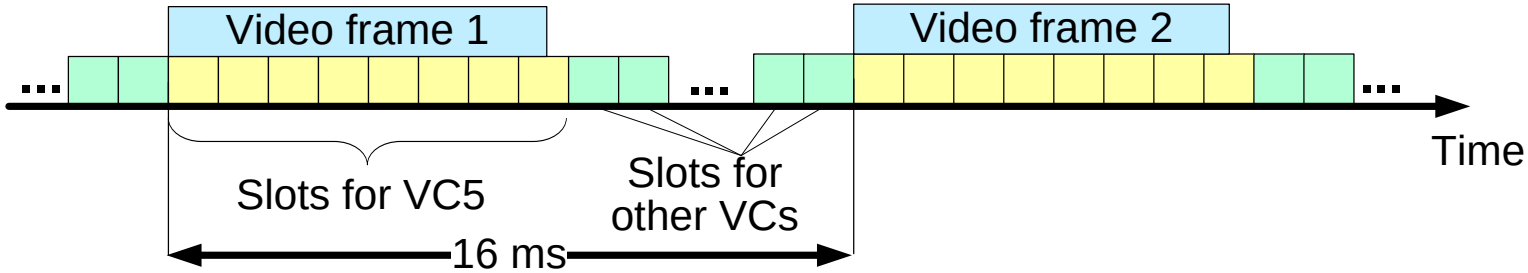
- Virtual Channels (VC) with 1Kbytes buffers for data storage
- Medium Access Controller multiplexes output data and provides QoS
- VC de-multiplexer distributes input data between VCs



What is SpaceFibre QoS?

Medium Access Controller provides QoS:

✓Sheduled: time is separated into max 256 time-slots during which VC can be scheduled to send data



The current time-slot is indicated by broadcast time values

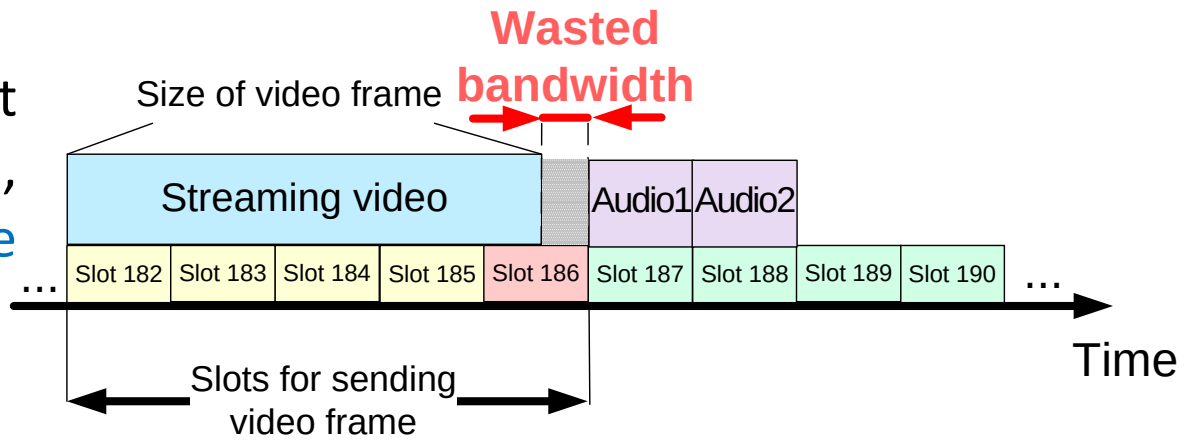
✓Priority: 16 priorities. VC with high priority will be entitled to send data first

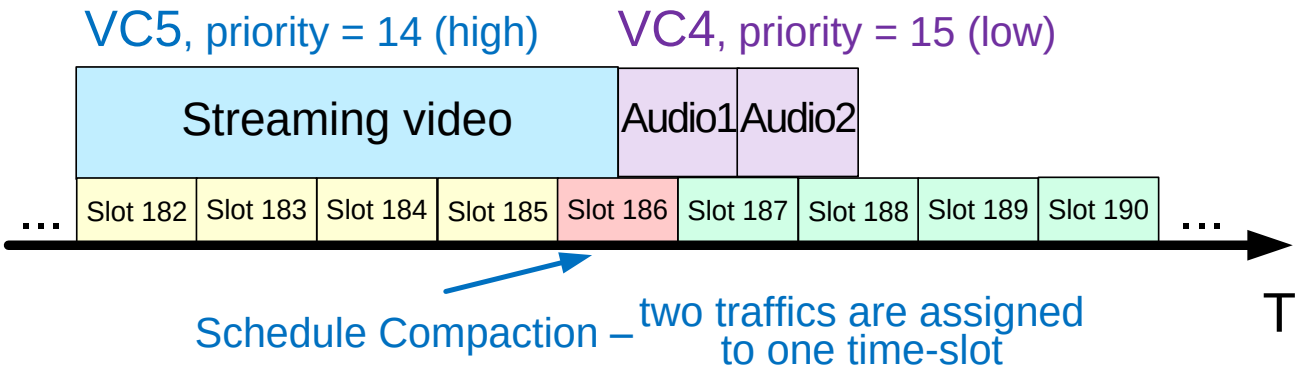
✓Bandwidth Reserved: determines the limitations of the link utilization by VC

Problem: SpaceFibre limits amount of priorities and time-slots

Schedule Compaction: Priority + Scheduled QoS

If VC scheduled in the time-slot does not have data to send, other VCs are permitted to use the wasted bandwidth



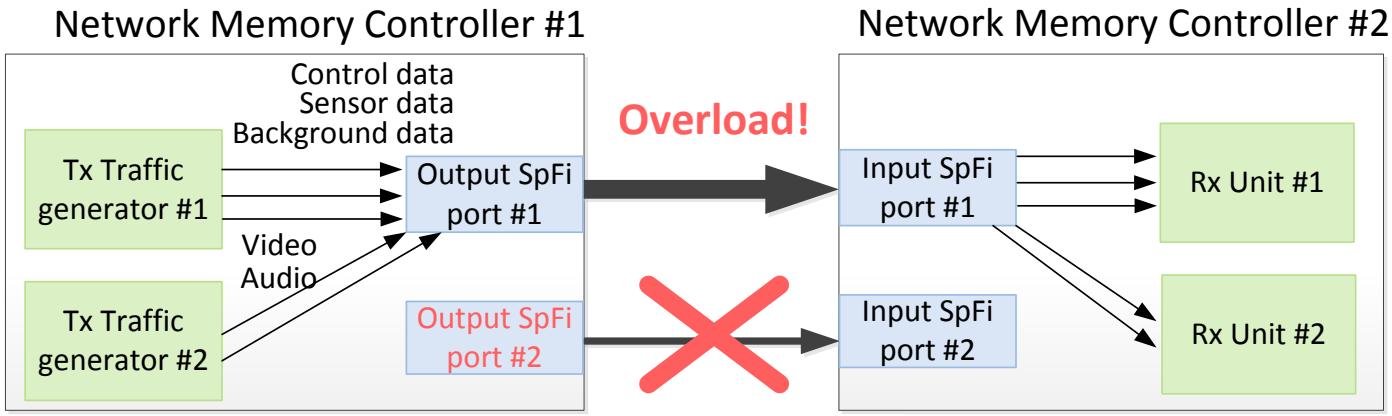


In general case, VC shall compete with others for sending data based on the priority

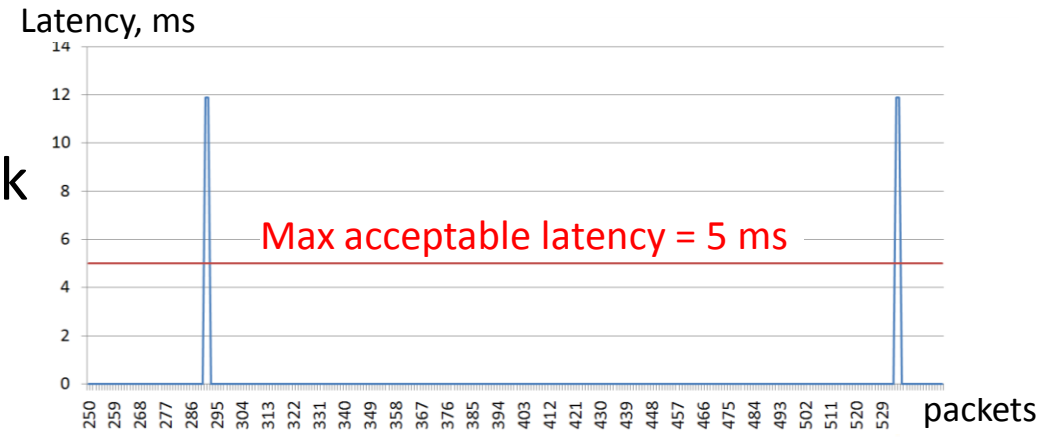
The Rule: VC5 must have **higher priority** than VC4 to guarantee that VC5 will be entitled to finish sending video frames. Then VC4 can send audio data

SpFi port performance testing (1/4)

Situation: port #2 was disabled. All data was passed through the port #1



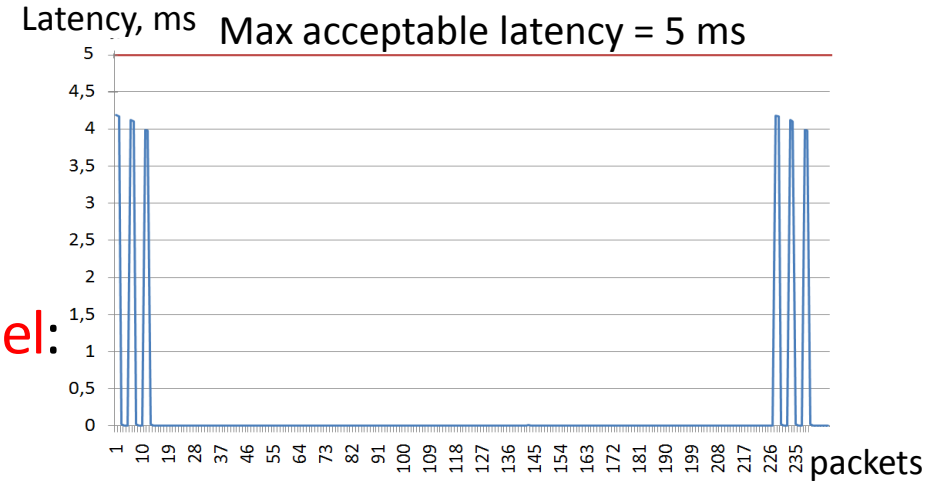
Result: sensor data were delayed for too much time, because the link was occupied by a video traffic. Sensor latency – unacceptable



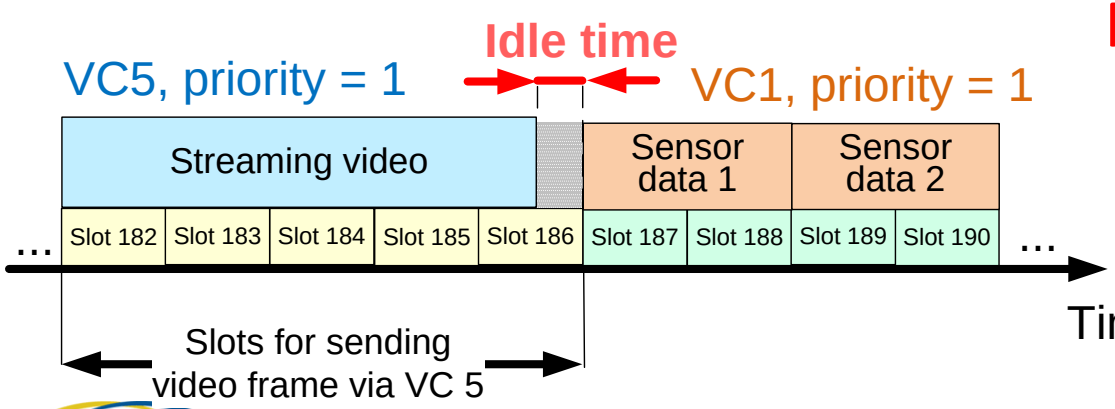
SpFi port performance testing (2/4)

SpaceFibre Solution: use Schedule Compaction to get delivery latency determinism

Result: all data were delivered with acceptable latencies



But there is problem – idle virtual channel: not transferred max possible amount of sensor data



Reason: **video VC priority** is not higher than **sensor VC priority** –> schedule compaction doesn't work –> **idle time**

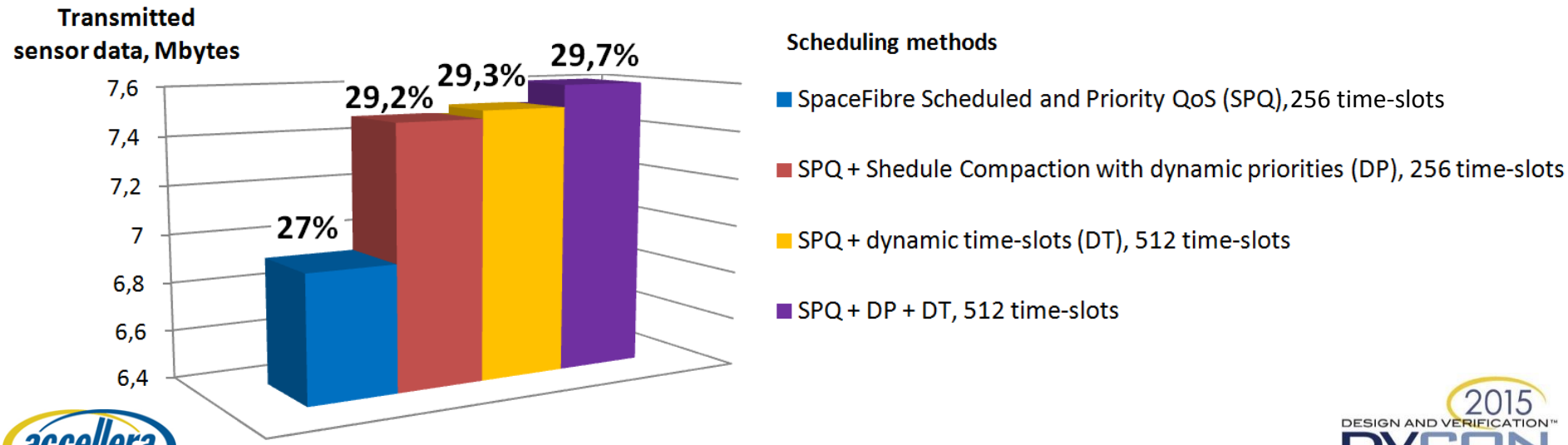
SpFi port performance testing (3/4)

Our solution: use Schedule Compaction with **A**daptive **D**ata **S**treaming **S**ervice

ADSS can dynamically change VC priorities and time value that defines the current time-slot. It removes the limitations of priorities and time-slots.

Result: 1) no idle virtual channel 2) more time-slots for scheduling: from 256 to 512

The increment of transmitted sensor data is up to 2.7%



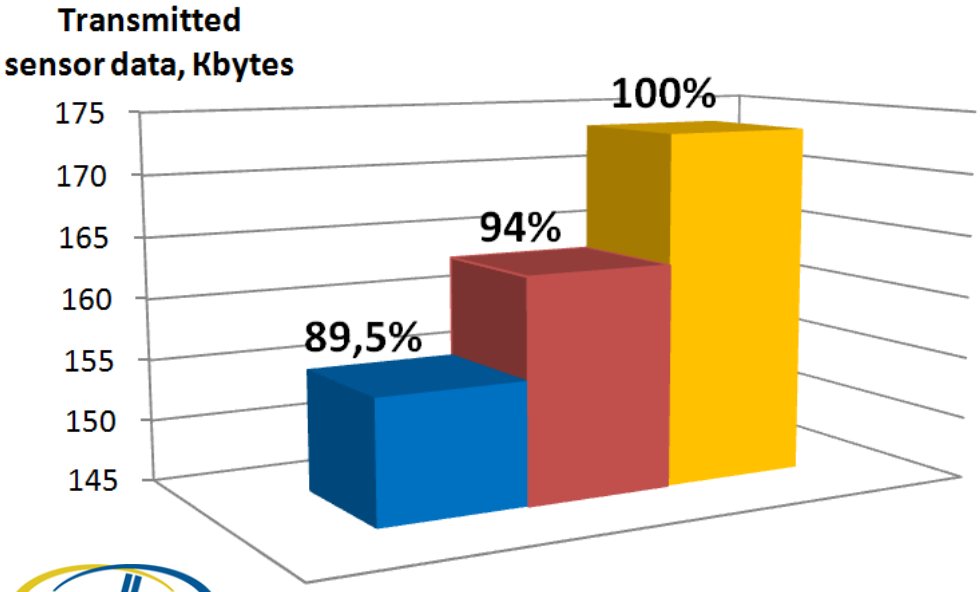
SpFi port performance testing(4/4)

The increment by 2.7%: is it a *small result*?

It's good result for spacecraft, because the reliability was improved!

Result depends on traffic and schedule:

- Packet size: 1.33 packet/ms
- Packet rate: 260 bytes



The increment of transmitted sensor data is up to 10.5%

Scheduling methods

- SpaceFibre Scheduled and Priority QoS (SPQ), 256 time-slots
- SPQ + Schedule Compaction with dynamic priorities, 256 time-slots
- SPQ + dynamic time-slots, 512 time-slots

accellera SYSTEMS INITIATIVE

2015 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE

Conclusion

SystemC/TLM 2.0 languages

OVP technology and Cadence VSP Software

They allowed to perform simulation of the Network Memory Controller project and to analyze performance of SpaceFibre ports

Thank you for attention!

Questions?

Feel free to contact with me

e-mail: ilya.korobkov@guap.ru

accellera SYSTEMS INITIATIVE

2015 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE