LEON SVF, a hardware accelerator for LEON2-FT software tools

SESP 2008

Mauro CALENO (ESA/ESTEC) & Fréderic PASQUIER (Astrium Satellites)

9 October 2008

All the space you need

esa

EADS astrium

Contents of the presentation

- Near limit in LEON software simulators
- Rationale for hardware-based accelerator
- LEON SVF: concept, consortium and architecture
- Targeted performances
- Current status of study
- Lessons learned
- Activities to completion of contract
- Conclusions

Abbreviation note: SVF=Software Validation Facility

Near limit in LEON software simulators

- ERC32 SW-simulators emulate OBC@25MHz in real-time
  - Instruction set emulation
  - TSIM, SIMERC32, TargetSIM, ESOC-sim
  - Eg Galileo IOV, Herschel-Planck, Gaia, Bepi-Colombo
- LEON SW-simulators emulate OBC@40MHz in real-time, but
  - Flight OBC expected to clock Atmel AT697 ASIC at 80MHz
  - Simulation not fully representative, e.g. cache model
- Block-oriented dynamic translation SW-simulators:
  - Alternative emerging technique
  - Performance is boosted compared to instruction set emulation
  - Accuracy is reduced
  - Not-proven for real-time embedded application domain

Abbreviation note: OBC=On-Board Computer

Step forward: hardware-based accelerator

- Achieve real-time simulation of Leon OBC@80MHz
- Full accuracy of processor emulation by running LEON VHDL code on COTS FPGA PCI board
- Feasibility of hardware-in-the-loop emulators technology proven by SHAM product family:
  - Boards for the ASIC of ERC32, ERC32-SC, TSC21020, Ma3-1750, MAS281
  - Total of 50 boards deployed in XMM, ISO, Rosetta, Envisat, ...

Abbreviation note: SHAM=Simulation Handling Module; COTS=Commercial off-the-shelf; PCI=Peripheral Component Interconnect

LEON SVF Concept

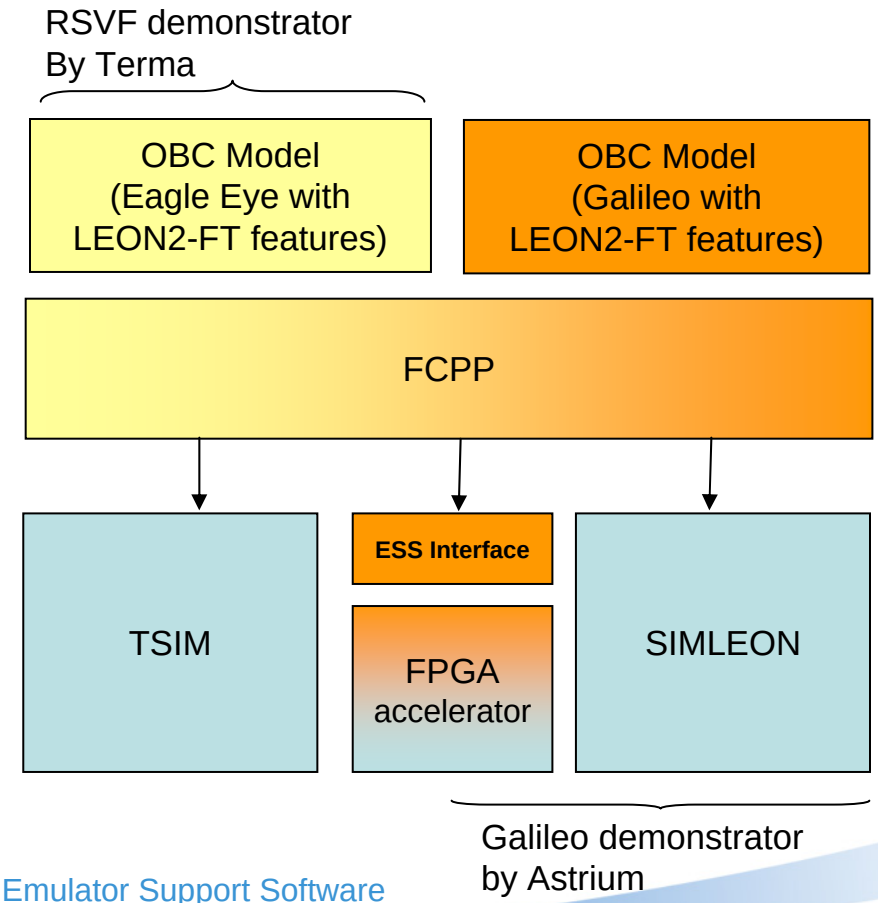
- Purpose:
  - Validation of non-instrumented flight software for Atmel 697 (Leon) with full control and observability at real-time for OBC@80MHz
  - Potentially operational simulators
- Demonstrate LEON SVF in two configurations:
  - Hardware configuration on COTS FPGA PCI board
  - Software configuration using SIMLEON and TSIM
  - Interchange between software and hardware configuration
  - Common interface hides SW or HW configuration (FCPP)
  - S/C and IO devices simulated by software on host
- Demonstrate LEON SVF in two representative use cases:
  - Galileo OBC Model on SIMTG infrastructure
  - Eagle Eye OBC model on VSRF/RSVF infrastructure

Abbreviation note: FCPP = Functional Component Plug-in Protocol; SIMTG= Simulation 3rd Generation; VRSF=Virtual Spacecraft; RSVF=Reference SVF

LeonSVF consortium

- Phase 1, concept study and prototyping
  - Astrium GmbH (D), Astrium SAS (F), SAAB (S), Terma (DK)
- Phase 2, design upgrade and production
  - Astrium SAS (F), Terma (DK)
- Expected finished by Q2 2009

LEON SVF Overview



Responsibilities

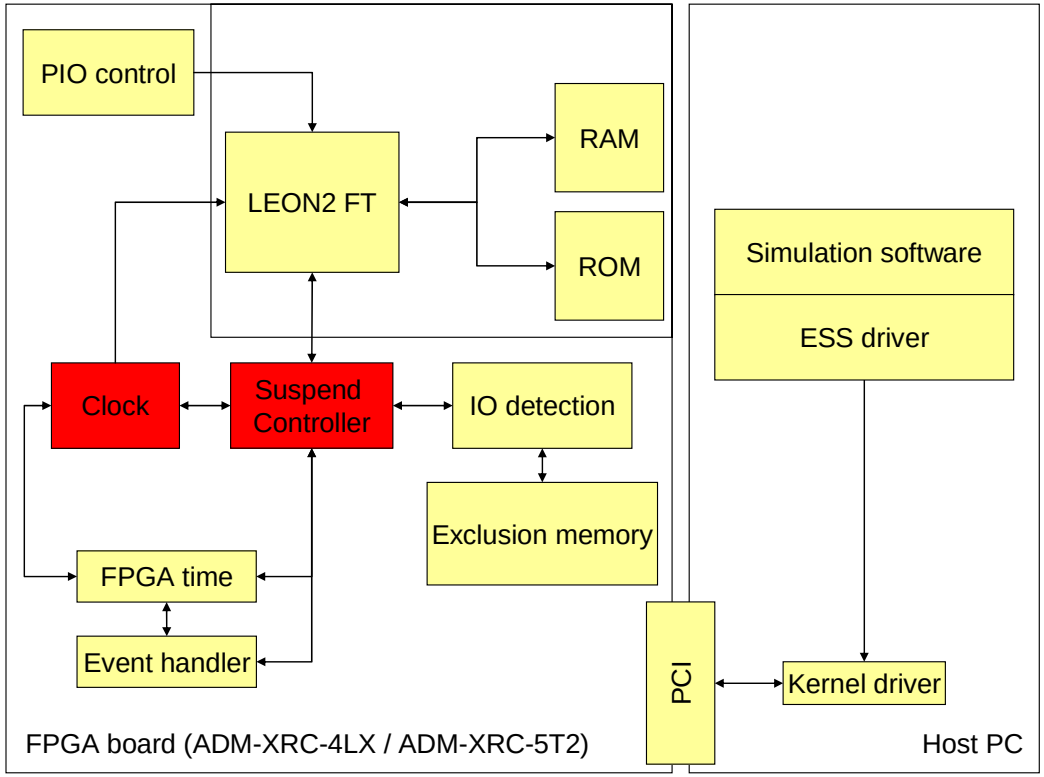
Orange: Astrium

Yellow: Terma

Grey: off the shelf

Abbreviation note: ESS: Emulator Support Software

LEON SVF with accelerator board



Key function Suspend Controller will suspend Leon while simulating I/O

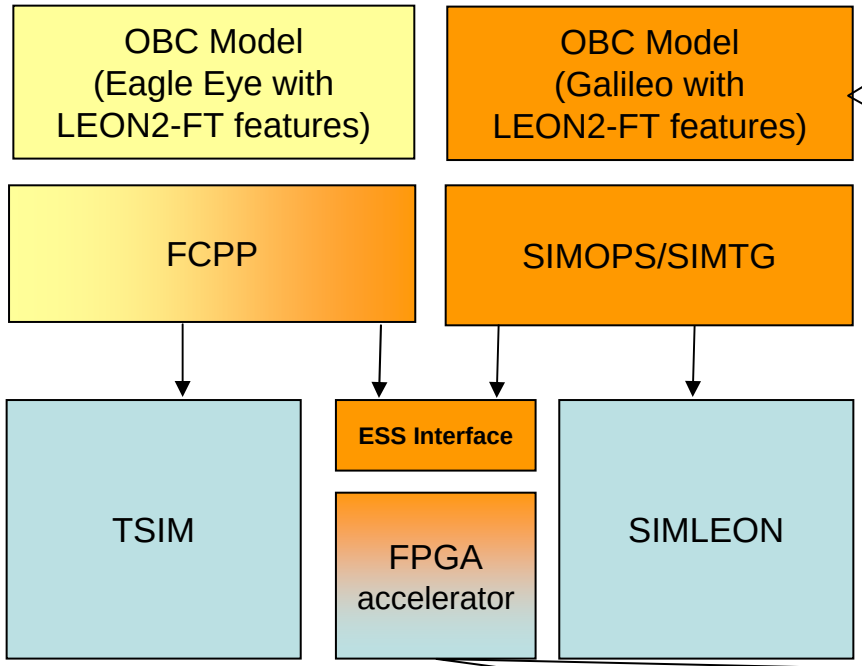
Performances

- ADM-XRC-4LX board (*) / Xilinx Virtex-4 LX80 FPGA
  - Leon2FT runs at 66MHz in FPGA
  - 5000 I/O per second introduce small overhead 10%
  - Emulate OBC@60MHz in real-time (measured)

- ADM-XRC-5T2 board / Xilinx Virtex-5 LX220 FPGA
  - Leon2FT expected to run at 90-100MHz
  - 5000 I/O per second introduce small overhead 10%
  - Emulate OBC@80MHz in real-time (expected)

(*) commercial board by Alpha Data Parallel Systems Ltd

Current status of study



Validated on Virtex 4 at 66MHz:

- Host simulates Galileo OBC at 60MHz
- Overall SVF speed is real-time
- 1 wait states in SRAM
- Realistic scenarios based on 5000 IO/s
- IO detection by the PC in 3.8us
- Simulation overhead measured 15us / IO
- Simulates COCOS by Saab with 1 x PWR, 1 x PWT, 1 x SPW, 1 x 1553, Mass memory
- Scenario: real time TM, 1553 messages, playback TM

Testing on Virtex 5 at 90MHz:

- FPGA-level testing in progress
- Leon clocked at 60MHz (target 90MHz)
- Preliminary measures show 60-100% of OBC@80MHz depending on cache pattern

Abbreviation note:
PWR Packet Wire Receiver, PWT=Packet Wire Transmitter, SPW=Spacewire

HW COTS: Lessons learned

- The SoW required to use a COTS FPGA board
- Pro's
  - Offload development to board supplier
  - Reduce development cost + time
- Con's
  - Constrained by market offer / adapt your requirements
  - More requirements → less offer
  - Quality depends on number of products deployed (flight hours)
  - Dependency from technical support of supplier
  - Long term availability
  - Ease of portability

Abbreviation note: SoW=Statement of Work

HW COTS: lessons learned, examples

- Board with Virtex-4
  - Maturity was good → smooth development of our application
  - SRAM memory bank broke
    - Waste of effort and time (fortunately we had 2 boards)
    - Technical support hasn't solved the problem yet
- Board with Virtex-5
  - We chose same supplier to re-use our design
  - Insufficient amount of SRAM → buy SRAM + DRAM and step-wise migration
  - Clocking constraints → add clocking areas (design complexity)
  - General lack of maturity
    - SRAM not operational → troubleshooting, stepped approach not possible
    - Dynamic memory controller does not compile → troubleshooting

Activities to completion of contract

- Finalisation of design on Virtex-5
- Full testing of Virtex 5 FPGA board
- Demonstration in system context:
  - Galileo OBC on Virtex-5 (Virtex-4 done)
  - EagleEye OBC on RSVF/VSRF
- Reference test cases generation on full software configuration (VSRF/RSVF)
- Hardware accelerator validation using reference test cases on VSRF/RSVF and Galileo on Virtex-5
- System simulator performance characterisation

Conclusion: performance & representativity

- Hardware-in-the-loop accelerator concept validated for Leon2-FT (accuracy, performance, operability)
- OBC@40MHz simulated in real-time on instruction set SW simulators
- OBC@60MHz simulated in real-time on Virtex-4
- OBC@80MHz targeted in real-time on Virtex-5
- Demonstration on Galileo and VSRF OBC simulators

Conclusion: functionality

Functions

- Run non-instrumented flight software for the AT697 Leon2FT
- Connect software simulations on host
- Non-intrusive control of flight software (RAM and I/O)
- Debug of Leon with gdb through DSU
- Interchange FPGA, TSIM and SIMLEON w/o modifications on the host simulation
- Save/restore mechanism

Miscellaneous

- Hardware accelerator 12K€ compares to SW licence 10K€
- Available Q2/09
- FPGA evolution → faster emulators

Conclusion: HW versus SW performance