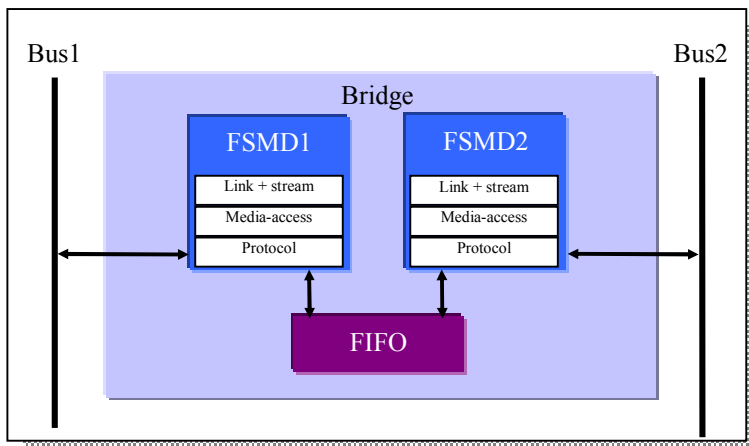
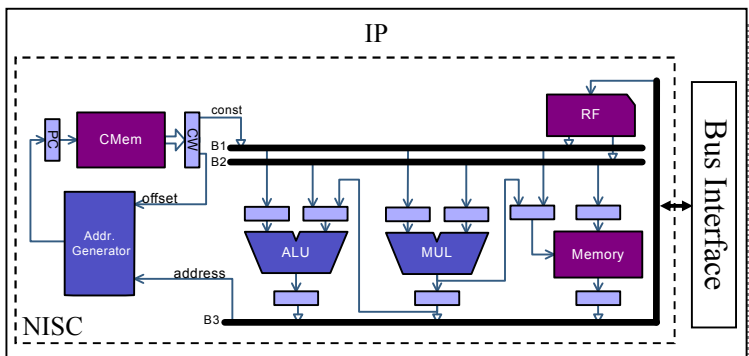
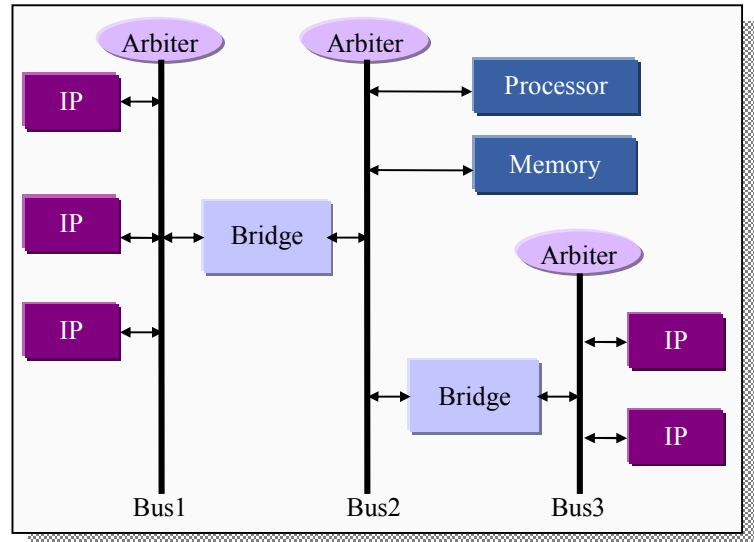


Embedded System Environment for Prototyping

ESE-Proto enables rapid FPGA prototyping of embedded applications

ESE-Proto

ESE-Proto is a tool for designing and prototyping embedded systems from C language. The target system architecture includes processors, custom IPs, memories, buses, arbiters and bridges. The tool provides automatic compilation of C code for custom IPs as well as generation of synthesizable RTL code for the entire system.



Features

NISC technology can be used for IP or soft custom processor design. In NISC methodology, the IP developer defines a custom datapath, and our NISC compiler uses this information for compilation of C code. The compiler outputs a stream of control words that are applied to datapath on every clock cycle.

Arbiter synthesizer automatically generates the RTL description of bus arbiters.

Bridge synthesizer automatically generates bridges that translate between any two bus protocols. The bridge receives data from one bus, stores it in a queue, and sends it out according to the proper network layers.

Bus interfaces (software and hardware) are automatically generated for all components.

Hard or soft core processors can be used for running operating systems and/or application.

Embedded System Environment for Prototyping

Design Flow

Inputs:

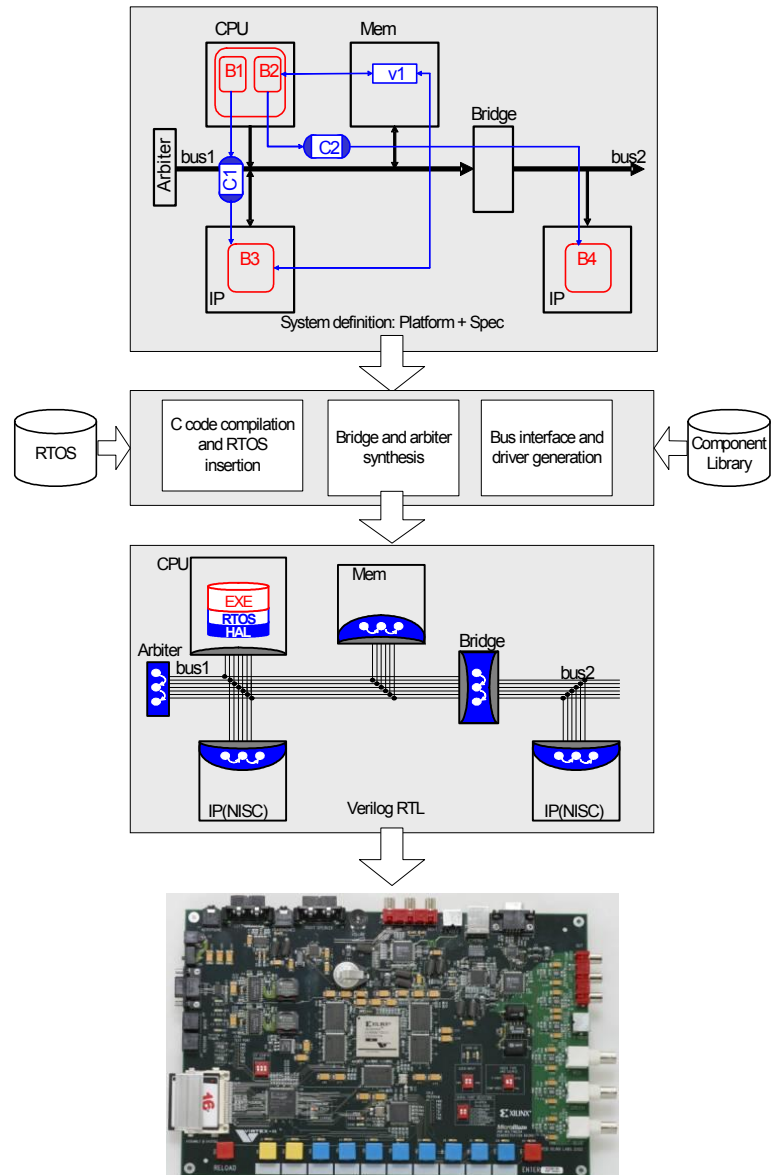
- **Platform description** represents the system architecture as a netlist of components.
- **Application C code** describes the behavior of each component in the system.
- **Architecture definition** for each IP or custom processor.

ESE-Proto functions:

- C code compilation for processors & IPs.
- Bridge and arbiter generation.
- Bus interface generation.
- Bus driver generation.
- RTOS insertion.
- RTL code generation for the entire system.

Outputs:

- **Complete synthesizable code** ready for FPGA download.
- **Executable binary** for processors.



Benefits

High productivity: Automatic compilation of C code for custom IPs, along with automatic generation of RTL code for bridges, and bus interfaces increase the productivity significantly.

Shorter time-to-market: Automatic synthesis of systems enables prototyping in few weeks.

Fast exploration: Large number of designs can be explored rapidly with FPGA-based emulation.

Better market penetration: Designs can be easily customized and re-prototyped.