

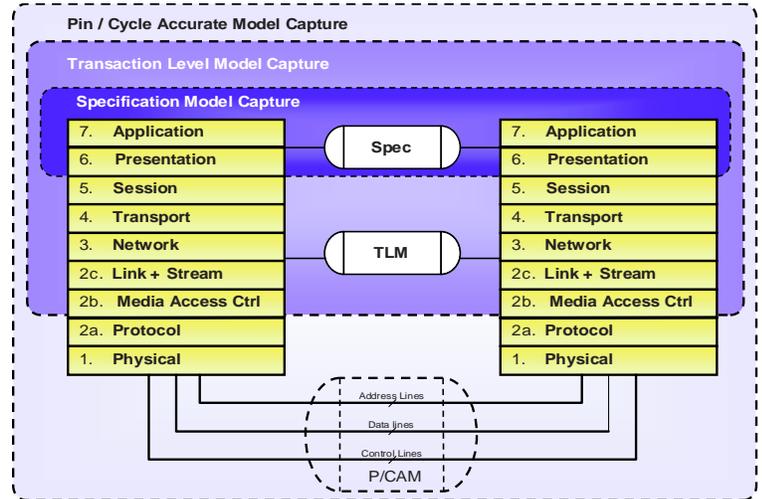


Embedded Systems Environment Front End

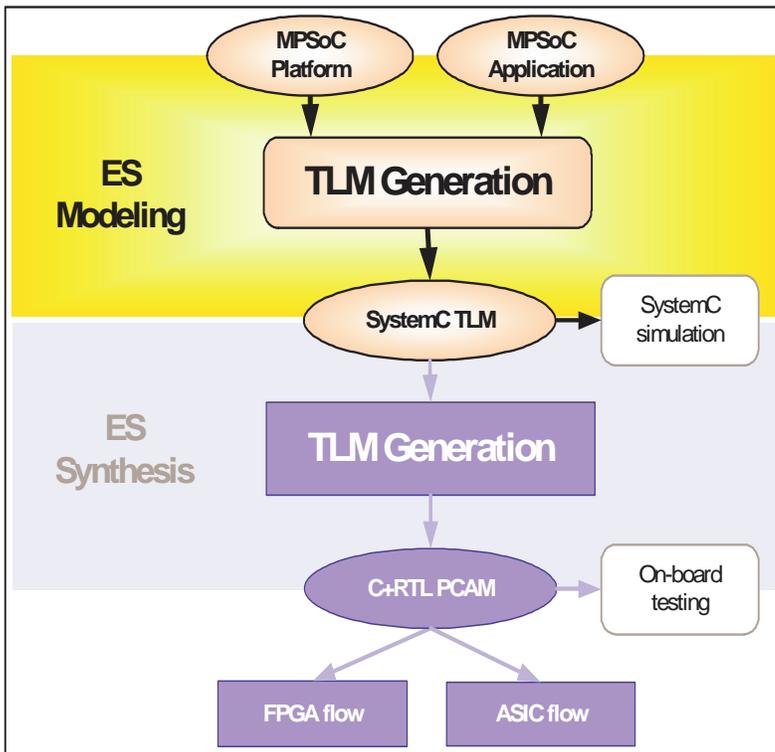
ESE Technology

Today's embedded system designs are complex heterogeneous platforms consisting of multiple standard and custom processors, each with possibly different interface, communicating over a network of busses and bridges.

System design tools are needed for prototyping such platforms with FPGA or ASIC technologies. ESE incorporates more than 15 years of research and development in system languages, synthesis and verification to provide automatic system prototyping from TLM.



ESE Front End helps application developers by automatically generating Transaction Level Models (TLMs) from application code and graphical platform specification. Application developers can easily map their code to different platforms and get fast TLMs. They can quickly evaluate the performance of their application and explore the computation and communication design space.



ESE Features

Graphical entry of platform as a netlist of processors, memories, busses and bridges.

Graphical entry of application as C processes communicating over channels.

Graphical mapping of processes to processors and channels to busses.

Automatic generation of SystemC TLMs for functional validation.

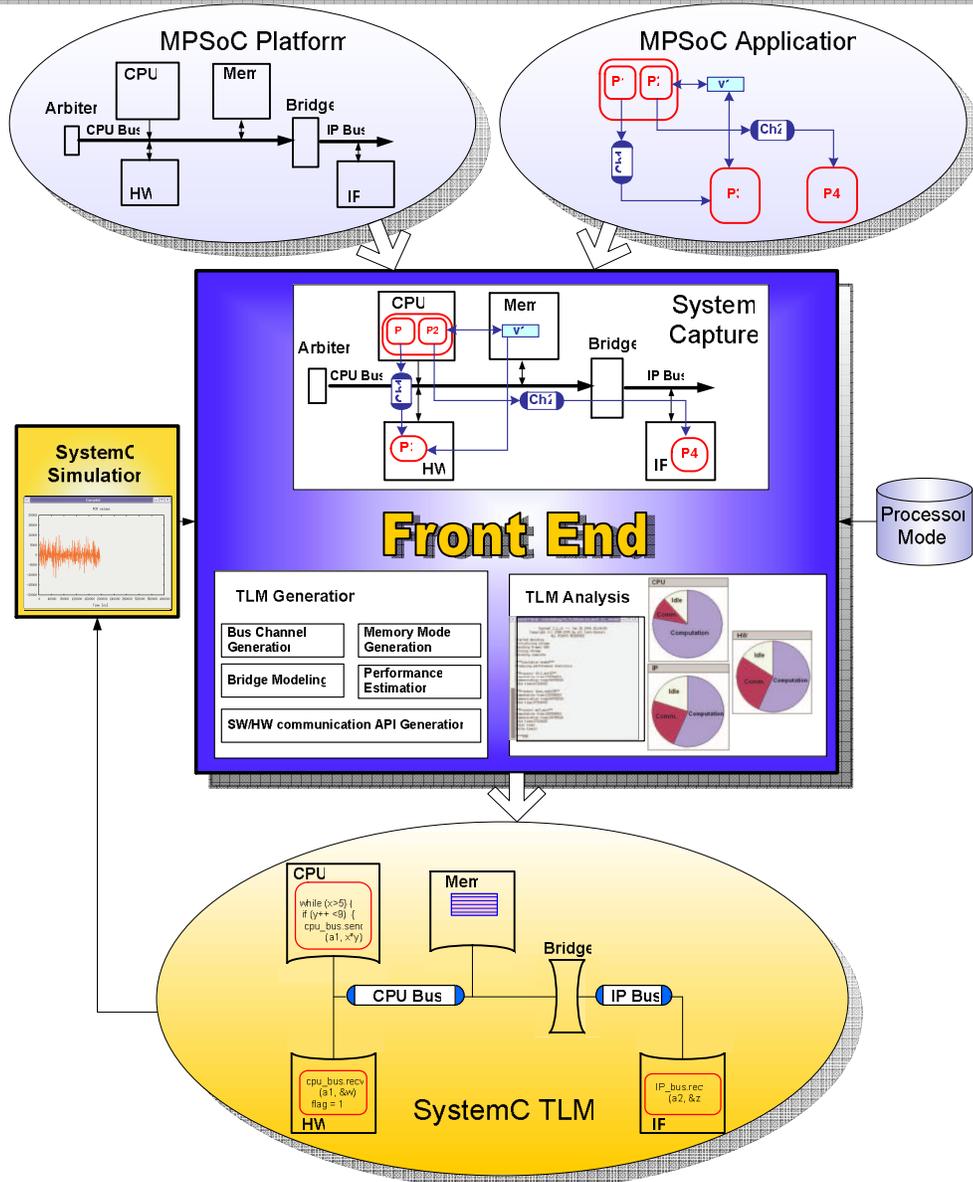
Performance annotation of generated TLMs for platform evaluation.



Embedded Systems Environment Front End

Designer Advantages

- | | | |
|---|---|--|
| Graphical application and platform input | → | Freedom from system level design languages |
| Automatic TLM generation for design decisions | → | Easier design space exploration |
| TLM simulation | → | 1000x faster simulation than RTL/ISS |
| TLM annotation | → | Early validation of design constraints |



Management Benefits

- | | | |
|---|---|---|
| Models are automatically generated | → | 1000x productivity gain, shorter time to market |
| Design decisions and models can be exchanged | → | Simplified globally-distributed collaboration |
| Designs can be easily modified and prototyped | → | Better market penetration through customization |
| Models and design decisions can be reused | → | Easier derivatives and version management |