ESE Front End

D. Gajski, R. Doemer

(With contribution from A. Gerstlauer, J. Peng, D. Shin)

Center for Embedded Computer Systems
University of California, Irvine
Technology Advantages

- No basic change in design methodology required
  - ESE methodology follows present manual design process
- Productivity gain of more than 1000X demonstrated
  - Designers do not write models
- Simple change management: 1-day change
  - No rework for new design decisions
- High error-reduction: Automation + verification
  - Error-prone tasks are automated
- Simplified globally-distributed design
  - Fast exchange of design decisions and easy impact estimates
- Benefit through derivatives designs
  - No need for complete redesign
- Better market penetration through customization
- Shorter Time-to-Market through automation
ES Environment

ESE Front – End
System Capture + Platform Development

ESE Back – End
SW Development + HW Development

Application Tools: Compilers/Debuggers
Commercial Tools: FPGA, ASIC

Decision User Interface (DUI)
- Create
- Select
- Partition
- Map
- Compile
- Replace

Validation User Interface (VUI)
- Compiler
- Debugger
- Stimulate
- Verify
- TIMED
- CYCLE ACCURATE
- Compile
- Check
- Simulate
- Verify
Platform Architecture

Components:
- Processors
- Memories
- IPs
- Custom HW
- Buses
- Bridges

Diagram:
- CPU
- Mem
- Arbiter
- HW
- Bridge
- IP
System Definition

Computation
- Behaviors (in C)
  (processes/functions)
System Definition = (Partial) Platform + (Partial) Spec

Communication
- Channels (in C)
- Variables (in C)
Output: Application Model
Output: Transaction-Level Model (TLM)
Output: Pin-Accurate Model (PAM)
System Modifications

TLM is generated/upgraded automatically with changes in Spec or Platform, including:

- Software changes
- Hardware changes
- Communication changes
Output: Modified TLM
Output: Modified PAM

CPU

B1
B2
B5
OS

HAL

Mem

Bridge

Arbiter

HW

B6
B3

IP

B4
Example: MP3 Decoder

- Functional block diagram (major blocks only)

```
mp3
  |                      |                |
  | HuffDec              | 2 granules     |
  |                      |                |
  | AliasRed            | IMDCT          |
  | Left channel        | FilterCore     |
  |                      |                |
  | AliasRed            | IMDCT          |
  | Right channel       | FilterCore     |
  |                      |                |
  |                      | PCM            |
  |                      | pcm            |
```

- Timing constraints
  - 38 frames per second
  - Frame delay < 26.12ms
**ESE: System Definition**

- Allocate and connect system components
- Edit processes (C code) inside components
- Insert communication channels and variables
- Run transaction-level model generator

---

Copyright ©2006, CECS
ESE: TLM Simulation

- Run simulation on automatically-generated TLM
Computation Analysis

- View computation time of processes
  - *filtercore* is the most computation-intensive

- Look for parallelism in process hierarchy
  - Left and right *filtercore* processes can run in parallel
    → Use two identical custom HWs
ESE: System Modification (1)

- Allocate new components from database
- Create bus ports for PEs
- Connect PE ports to busses
ESE: System Modification (2)

- Move processes to newly allocated PEs
- Inter-PE channels are inserted automatically
- Run transaction-level model generator
**ESE: TLM Simulation**

- Run simulation on automatically-generated TLM
- Display over-the-time bus activity
Bus Contention Analysis

- Bus priority: ARM > HW1 > HW2

Usage Distribution for mainBus

Waiting time when ARM uses bus

Waiting time when HW2 uses bus
ESE: System Modification (3)

- Remove components and connections
- Change connectivity
- Run transaction-level model generator
• Run simulation on automatically-generated TLM
• Display over-the-time bus activity
Design Summary

Frame Delay (ms)

Deadline: 26.12

mainBus Utilization

mainBus Contention
ESE Advantages

- Platform and application can be easily captured using GUI
- TL models are automatically generated
- ESE allows concurrent development of platform SW, HW and application code
- ESE allows easy upgrade of platform
- ESE simplifies reuse of legacy application SW and RTL HW code