EECS 222A: System-on-Chip Description and Modeling Lecture 8

Rainer Dömer

doemer@uci.edu

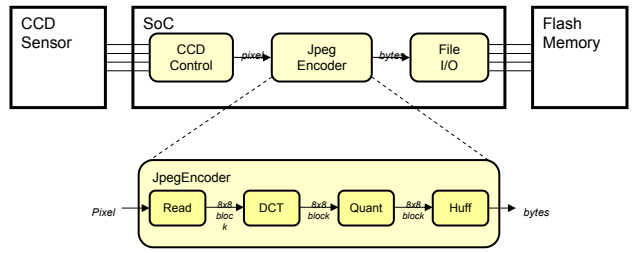
The Henry Samueli School of Engineering
Electrical Engineering and Computer Science
University of California, Irvine

Lecture 8: Overview

- Project Discussion: Digital Camera
  - Assignment 2: JPEG Application Structure
  - Assignment 3: Sequential SpecC Model
  - Assignment 4: Sequential SpecC Model in SCE
  - Assignment 5: Parallel SpecC Model for Synthesis
  - Assignment 6: Design Space Exploration
  - Discussion, Q&A
- Modeling with SystemC SLDL
  - SystemC Overview
  - Introduction to SystemC
    - Presentation by Stuart Swan, Cadence

Project Discussion: Assignment 2

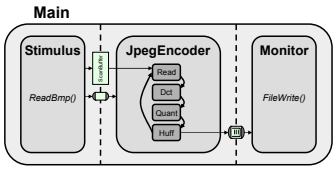
- Digital Camera Example
  - Component Model



  - Homework Assigment 2
    - Become familiar with JPEG Encoder Application
      - Study reference code: **`/home/doemer/EECS222A_F09/jpegencoder.tar.gz`**
      - Draw block diagram of files, functions, and key communication variables
      - Simplify code for a 116×96 pixel CCD (eliminate **`malloc`** calls!)

Project Discussion: Assignment 3

- Task
  - Convert JPEG Encoder Application to a proper SpecC Specification Model



- Deliverables
  - Email to **`doemer@uci.edu`** with subject "EECS222A Assignment 3"
    - Brief status description (in body of your email)
    - Source code **`jpegencoder.tar.gz`** (attachment)
- Due
  - Next week: October 30, 2009, 12pm (noon)

Project Discussion: Assignment 3

- Hint:
  - Use the **`sir_tree`** tool to validate your hierarchy
  - The final model should look like this:

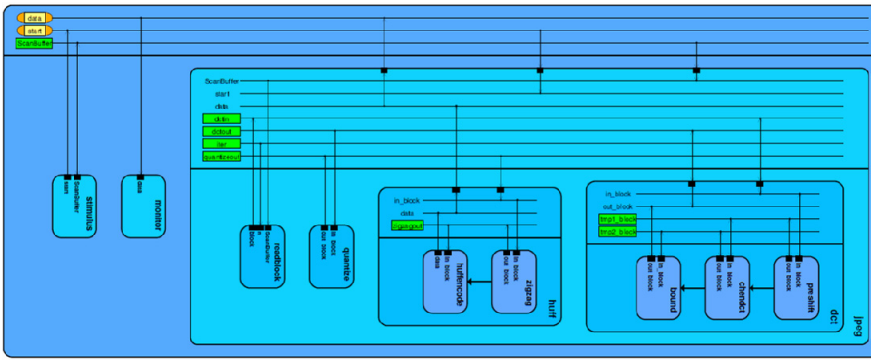
```
doemer@epsilon.eecs.uci.edu:3 > sir_tree -blt digicam.sir
B i o   behavior Main
B i o   |------ JpegEncoder jpeg
B i s   |       |------ Dct dct
B i l   |       |       |------ Bound bound
B i l   |       |       |------ ChenDct chendct
B i l   |       |       \------ Preshift preshift
B i s   |       |------ Huff huff
B i l   |       |       |------ Huffencode huffencode
B i l   |       |       \------ Zigzag zigzag
B i l   |       |------ Quantize quantize
B i l   |       \------ ReadBlock readblock
B i l   |------ Monitor monitor
B i l   |------ Stimulus stimulus
C i l   |------ c_queue data
C i l   \------ c_handshake start
```


Project Discussion: Assignment 4

3. Analyze your digital camera model in SCE
   - Setup
     - …continued from step 2 (previous page)
   - View the structural hierachy chart
     - Select the **`Main`** behavior in the behavior browser
     - Right-click **`->Chart`**
     - Double-click the chart to add further levels of hierarchy
     - Turn on connectivity **`View->Connectivity`**
     - **`Window->Print`**… to file **`"digicam.ps"`**
     - In your shell window, convert the PostScript file to PDF: **`ps2pdf digicam.ps`**
     - Check the PDF file: **`acroread digicam.pdf`**
- Deliverables
  - Hierarchy chart
    - **`"digicam.pdf"`**
- Due
  - by Friday, Nov 6, 2009, at noon
  - by email to **`doemer@uci.edu`** with subject "EECS222C HW4"

Project Discussion: Assignment 4

- Sequential Digicam Model
  - Screen shot of Hierarchy Chart in SCE
    - (rotated by 90 degrees)

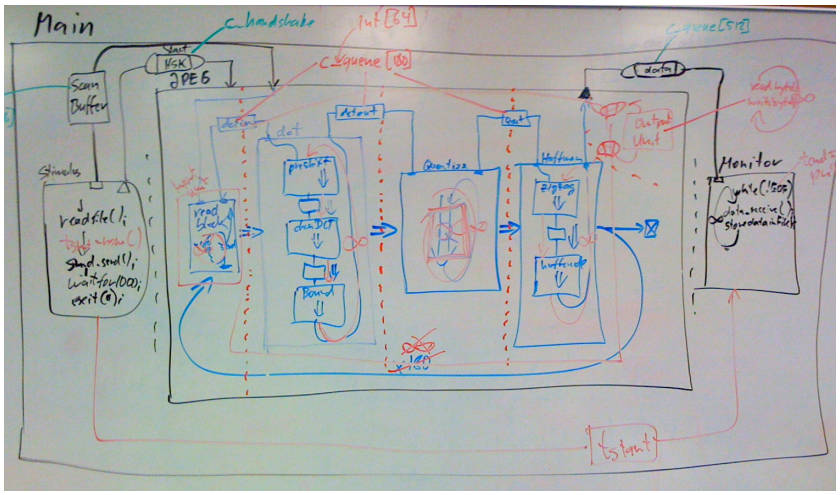


  - **`jpegencoder2.tar.gz`**

Project Discussion: Assignment 5

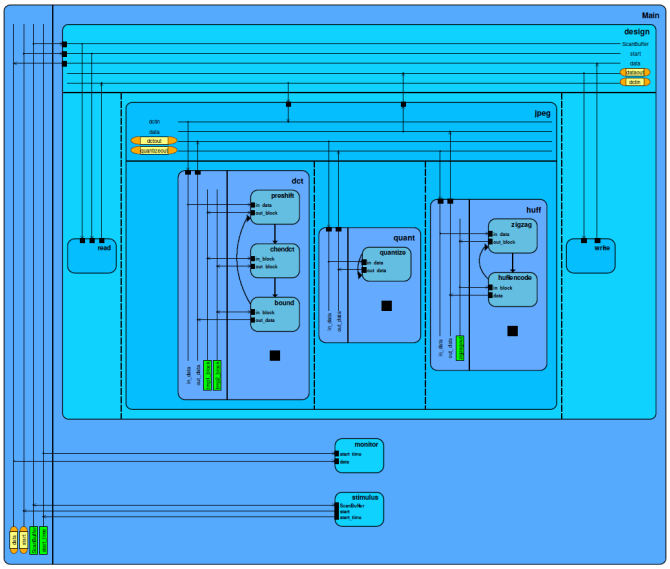
- Create a parallel/pipelined and synthesizable Digicam model
  - Start from previous model
    - **`/home/doemer/EECS222A_F09/jpegencoder2.tar.gz`**
  - Insert timing checks into the test bench
  - Add explicit I/O units
  - Parallelize/pipeline the **`JpegEncoder`** block
    - Modify **`Dct`**, **`Quantize`** and **`Huffman`** to infinitely work on continuous streams of data over typed queue channels
  - Validate functionality
  - Print hierarchy chart with connectivity
- Deliverables
  - Status description
  - Source code **`"digicam.tar.gz"`**
  - Hierarchy chart **`"digicam.pdf"`**
- Due
  - by Friday, Nov 13, 2009, at noon
  - by email to **`doemer@uci.edu`** with subject "EECS222C HW5"

Project Discussion: Assignment 5



Project Discussion: Assignment 5

- Pipelined Digicam Model
  - Explicit I/O Units
  - Explicit Pipelining
  - Communication via Queues
- `jpegencoder3.tar.gz`



Project Discussion: Assignment 6

- Design Space Exploration using SCE
  1. Profile, analyze, estimate the digicam model (**`jpegencoder3.tar.gz`**)
     - For a single ARM_7TDMI CPU at 100MHz
     - For metric of "Computation"
     - For blocks **`read`**, **`dct`**, **`quant`**, **`huff`**, **`write`**
     - Create a bar chart of the Computation Profile
       - **`"ARM100.pdf"`**
  2. Create a software-only reference model
     - Allocate and Map
       - 1 ARM7TDMI CPU at 100MHz as "ARM100" for JpegEncoder
       - 1 Custom HW_Standard block at 100Mhz as "InputUnit" for ReadBlock
       - 1 Custom HW_Standard block at 100Mhz as "OutputUnit" for WriteBlock
     - Estimate the allocation (Validation->Evaluate)
     - Perform Architecture Refinement
     - Perform Scheduling Refinement
       - Use Round-Robin scheduling policy on ARM100
     - When executed, the resulting model should encode our test picture in 39.252ms

Project Discussion: Assignment 6

- Design Space Exploration
  3. Architecture Exploration
     - Explore various other system architectures
       - Use up to 5 ARM_7TDMI processors
         - » Clock frequency may be 100, 150, or 200MHz
         - » Cost is assumed at $100, $150, $200, respectively
         - » Use only 50MHz AMBA AHB bus
       - Use up to 5 HW_Standard accellerator blocks
         - » Clock frequency may be 100, 200, 300, or 400MHz
         - » Cost is assumed at $200, $400, $600, $800, respectively
       - Vary the mapping of blocks in the DUT to CPUs and HW units
       - Vary the scheduling policy as needed
     - Example:
       - Use 1 HW100 for DCT
       - Use 1 ARM100 for everything else

Project Discussion: Assignment 6

- Design Space Exploration
  4. Scheduling Exploration
     - Explore various scheduling strategies for each selected CPU
     - Choose from
       - Static scheduling
         - » with varying execution order
       - Round-Robin scheduling
       - Priority-based scheduling
         - » with varying priorities
     - Example:
       - ARM100 scheduled with round-robin
       - ARM200 scheduled with priority-based scheduling
       - Static scheduling on HW blocks

Project Discussion: Assignment 6

- Design Space Exploration
  5. Deliverables
     - Bar chart of the software-only computation profile
       - **"ARM100.pdf"**
     - Text file with table of 3 "good" architectures
       - List the allocation and mapping for each block
       - Simulate to estimate the resulting encoding delay
       - Calculate the assumed cost of the architecture
- Due
  - by Friday, Nov 20, 2009, 2pm
  - by email to **doemer@uci.edu** with subject "EECS222A HW6"
  - bring a copy for discussion in class!

SystemC Overview

- Goals
  - Common C++ Modeling Platform
    - System Level Design
    - HW/SW Codesign
    - RTL
  - Seamless Co-Simulation of HW and SW
  - IP Reuse
  - Free licensing, Open Source
  - De-facto Standard
- Open SystemC Initiative (OSCI)
  - Consortium of many EDA companies
    - Synopsys, Cadence, CoWare, Frontier, ...
  - Open Community (very large!)

SystemC Overview

- Language
  - C++ class library (layered SW architecture)
  - Hierarchy of Modules connected by Ports
  - Communication via Interfaces and Channels
  - Discrete-Event Simulation
- Methodology
  - Untimed Model
  - Transaction-level Model
  - Bus-functional Model
  - Cycle-accurate Model

Introduction to SystemC

- Presentation by Stuart Swan, Cadence, 2002
  - Goals and Requirements
  - History and Organization
  - Versions, Contents, Coverage
  - Language Architecture
  - Modeling, Models of Computation, Examples
  - Communication Refinement
  - Outlook