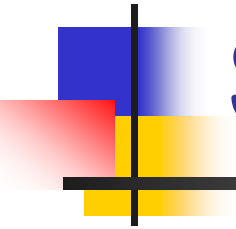
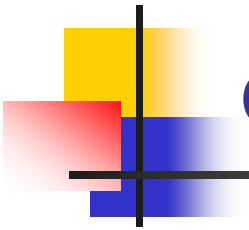


SOC lab Case study

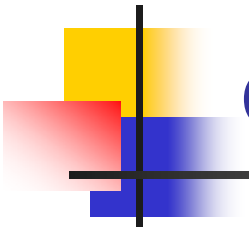
Joonsung Park

Case studies for Area and Speed optimization

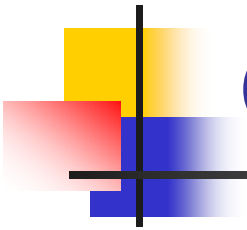


- Following slides summarized several ideas suggested by each group in last semester to give you initial guidance on area and speed optimization on estimation.
- These are just for reference. You can come up with more advanced idea based on these materials.

Case1-Optimizing the HW/SW communication



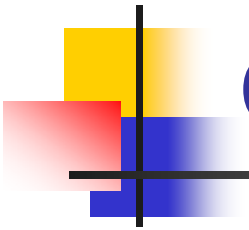
- It takes considerable time to communicate between hardware and software (ARM CPU).
- Using DMA scheme to reduce the memory access time and increase the CPU usage.
- You can examine the operation of Viterbi decoder to reduce the length of datapath to proper level.-This saves the area and memory access time considerably.



Case2-Software optimization

- Software part is not optimized to run on ARM processor
- We can examine the source code to remove unnecessary function calls – this improves the runtime efficiency.
- Moved some operations done in Hardware accelerator to software part. This reduces the size of Viterbi decoder.

Case3-System level approaches



- Reference source code is not optimized for hardware implementation. – Some memories are redundant. Pipelining can be used to reduce the size.
- Use system level approach to reduce the size of Viterbi Decoder – You can research several version of Viterbi algorithm optimized for hardware implementation.
- Where are you going to place the memory – inside or outside the Viterbi Decoder?

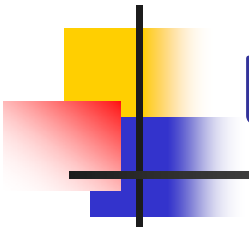
Comparison of each approach

	Case1	Case2	Case3
Power	384.6mW	284uW	2.5mW
Total Area	1.143mm^2	10.22mm^2	0.719mm^2
Worst case delay	4.79ns	5ns	14.3ns
Total cycle count	101,710,975	394,557,595	121,000,000

Area and Speed estimation

- Following is one example of area and speed estimation from structural level SystemC code.
  - Used a script to create a gate library containing elements of all different bit widths, based on templates.
  - Each gate SC_MODULE had area and delay values encoded.
  - Each gate SC_MODULE has embedded code to print area, calc ulate propagation delay, and display min and max integer value s.
  - Power estimation – Think of active power (Arithmetic Unit) and leakage power (Cache).
  - Using following equation to calculate the active power

    $P_{active} = 1/2 \cdot SF \cdot C_{switch-den} \cdot Area \cdot V^2 \cdot freq$
  - Assume active power occupied 80% total power
  - $P_{total} = P_{active}/0.8$



Lessons learned from last semester

- Following is the several quote from the students after the project is done. You might need to consider these before you start project.
  - Always verify feasibility of the specifications that marketing pro vides before agreeing to meet them.
  - Design trade offs – focus on the most critical path and reduce time spent fine tuning
  - It is important to discuss tradeoffs in high level architecture before beginning implementation
  - Cycle accurate hardware/software simulations with CoWare are slow
    - Do simulations as high level as possible-capturing behavior
    - Perform lower level simulations against the captured behavior
  - Start early!