CEG5010: Conclusion

What have we learnt?

- **FPGA**
  - When to use them
  - Low volume, upgradability, easy re-engineering
  - How to take advantage of the architecture
    - Runtime reconfiguration
    - Use special features: RAM, SRL, DLL, flip-flop rich

- **Computer Arithmetic**
  - Distributed arithmetic
  - Cordic
  - STAM

- **Computer Architecture**
  - Finding parallelism via pipelining and parallel processing
  - Customization for a given problem: module generators, runtime specialization
  - Bus interfaces can be a bottleneck

Example application where many of these ideas are used (Andraka software radio)

Demonstration

- **23/4/2008 (WED) 10:30 a.m. in lab**
- Informal demo to give me a chance to ask questions about your project
  - Show understanding of the problem and design decisions made
Report

- Due same day
- 4 page report include previous work, full description of system, system performance, resource utilization, memory map etc (like a full conference paper)
- Appendix: Host program, VHDL listing, simulation graphs
- Submit hard and soft copy to Matthew