

CSCE 611:  
Timing Simulation and FPGA-Based Verification



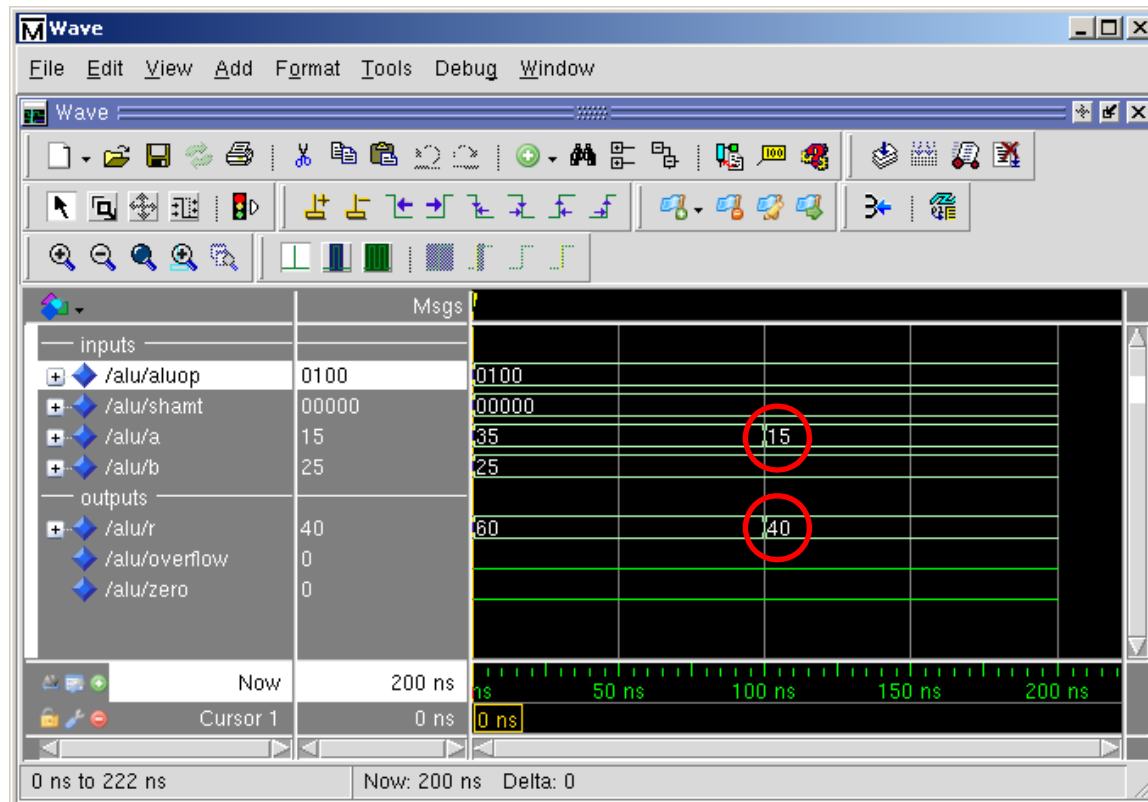
Instructor: Jason D. Bakos



UNIVERSITY OF  
SOUTH CAROLINA.

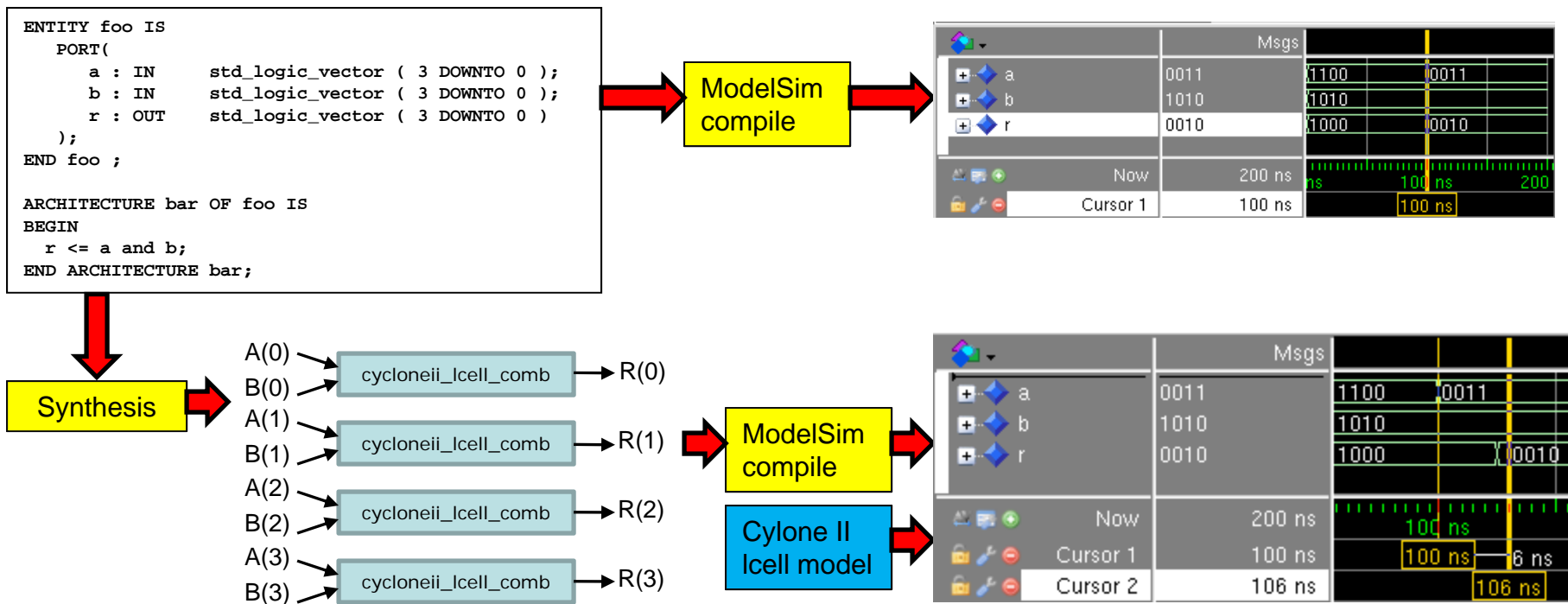
# Timing Simulation

- To this point we've done behavior simulation
  - Test the functionality of the design



# Timing Simulation

- What about logic latencies?
- Solution: post synthesis simulation (back annotation)

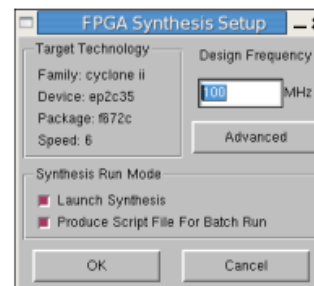
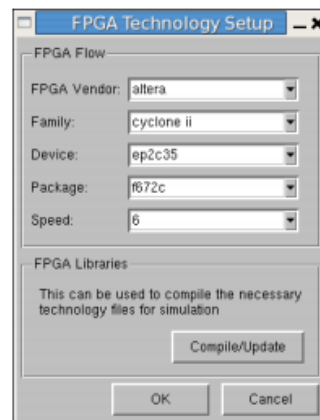


---

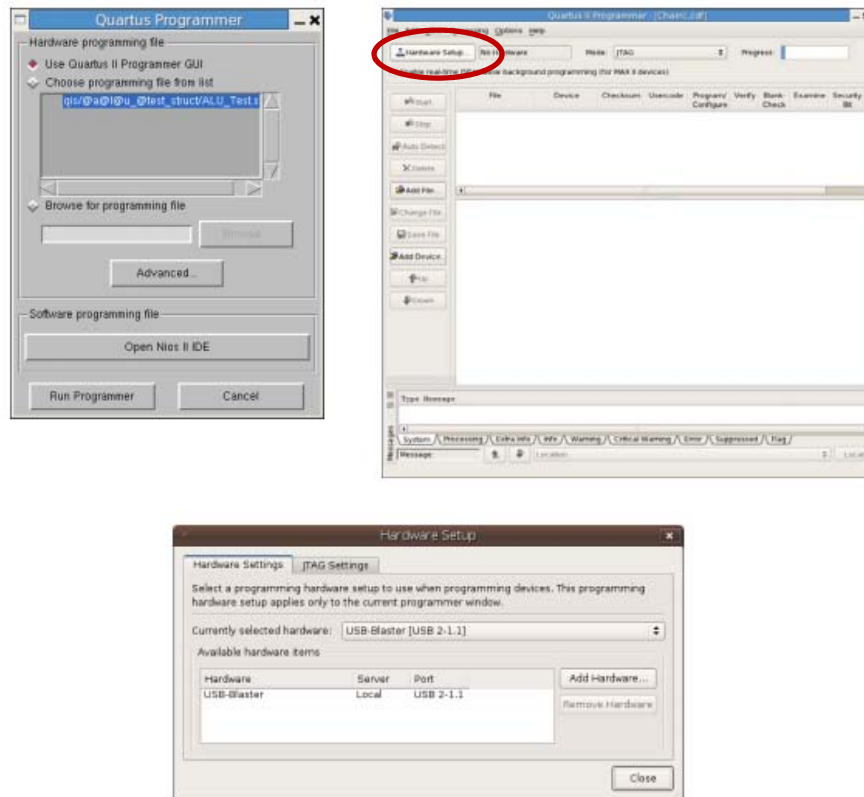
# Synthesis and Place-and-Route

---

- To implement your ALU on the FPGA, we provide a “wrapper” that connects the inputs and output of your ALU to test pins on the FPGA
  1. Open CSELib
  2. Copy ALU\_Test component to your ALU design library
  3. Make some minor modifications to this file (see tutorial 14)
  4. Use the “Quartus II Synthesis” task to synthesize ALU\_Test
  5. Set up the FPGA technology:

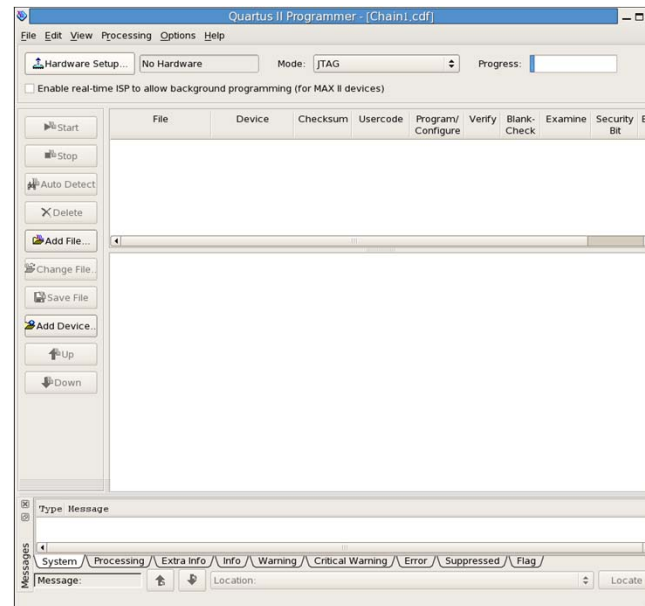
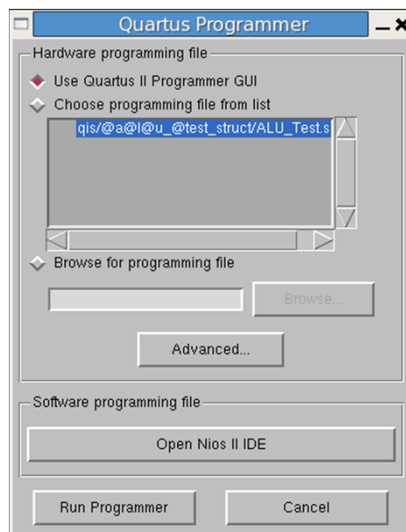


- 
6. Next, run the "Quartus II Place and Route" task on ALU\_Test
  7. Next, run the "Quartus II Programmer" task:

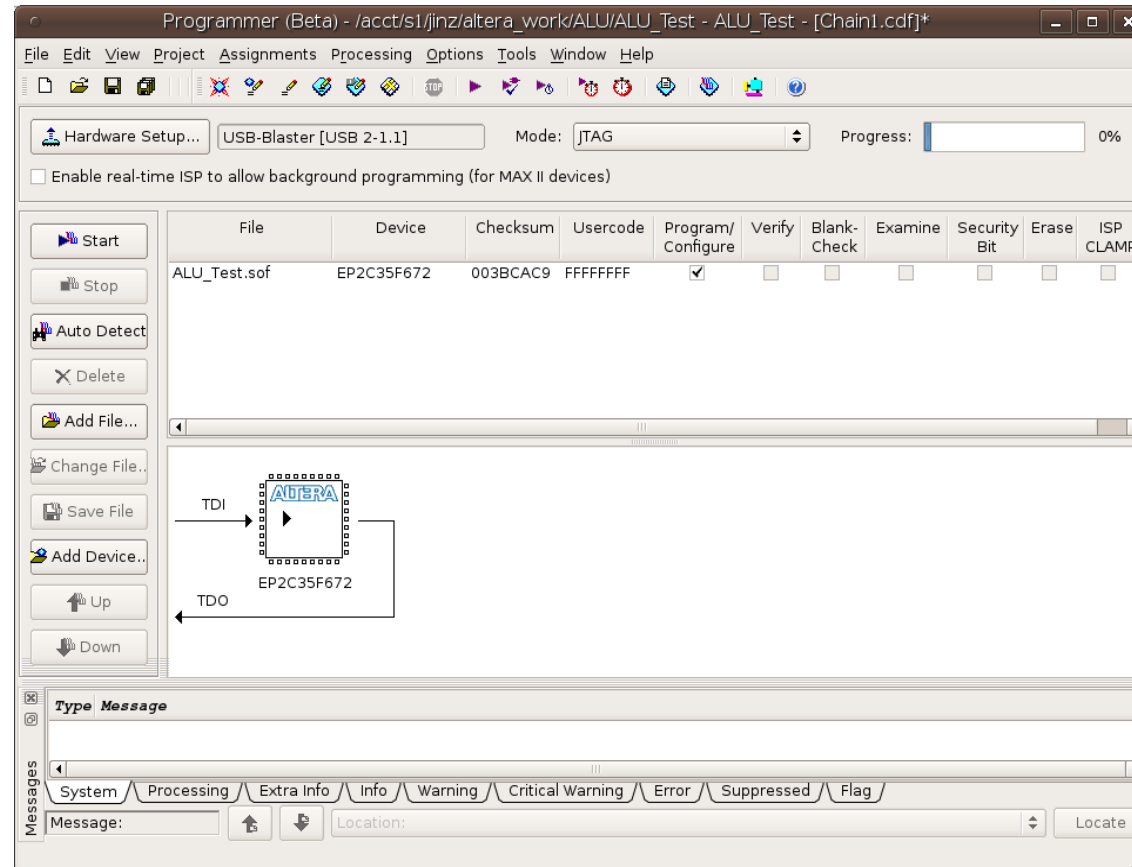


# FPGA Verification

- Design flow steps:
  - Design ALU
  - Behavioral verification against test bench
  - Timing verification against LUT and interconnect models
  - Verification on actual hardware using JTAG sources/probes



# FPGA Verification



---

# JTAG

---

- Original idea: boundary scan
  - Originally used to test chip package and PCB connections
  - Chip inputs and outputs
  - “Scan chain”
- Now used as a way to source and probe internal signals
- Also used to program FPGAs
- “Joint Test Action Group” (JTAG) standard
  - Test clock (TCLK)
  - Test mode select (TMS)
  - Test data input (TDI)
  - Test data output (TDO)

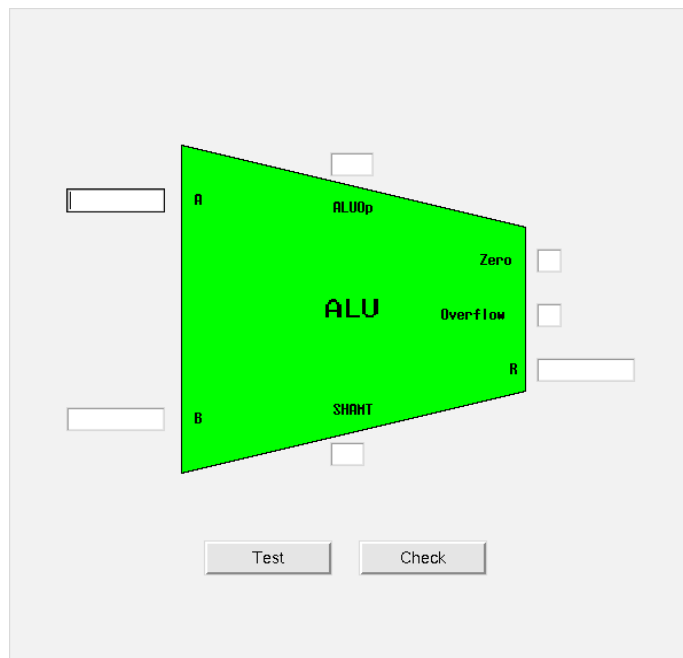


---

# Altera SignalTap

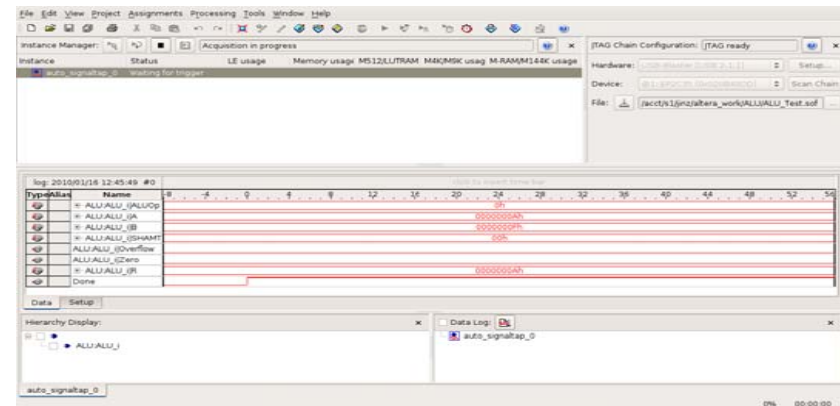
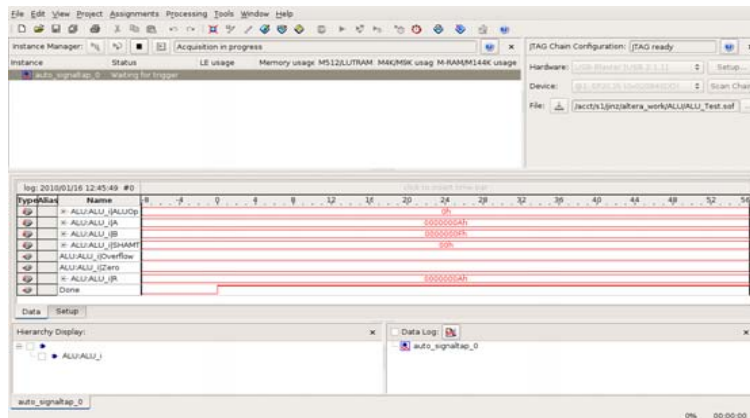
---

- We've written a Tcl/Tk script to test the ALU using JTAG/SignalTap
- After programming FPGA:
  - `quartus_stp -t /usr/local/3rdparty/csce611/ALU_Test.tcl`



# SignalTap

- You can also debug using the SignalTap tool
- Acts as a virtual logic analyzer
  - Allows you to enter probes for certain signals
  - Allows you set to set trigger conditions
  - Does not allow you to manipulate signals
    - Use the Tcl/Tk script in parallel to do this!



---

# SignalTap

---

- To use SignalTap to debug your ALU:
  1. Program FPGA from within HDL Designer
  2. Start up JTAG script from command line
  3. Start up Quartus from command line ("quartus")
  4. Open the Quartus project created in:
    1. /acct/s1/<username>/HDS/<library name>/qis
  5. Go to Tools | SignalTap
  6. Add probes and triggers (shown in tutorial 14)
  7. Re-synthesize, place-and-route, program FPGA
  8. Use ALU script to drive inputs to ALU

