October 7, 2003

- Three Ways of Doing I/O
- Tradeoffs
- Keyboard to Hex Display
- Keyboard to LCD
- Simulated and Real I/O

Three Ways of Doing I/O

- Directly control FPGA Pins
  - Handle-C Interfaces (bus_in, etc.)
- Use RC200E-specific macros
  - Non-portable
  - Cannot use PAL Virtual Platform window
- Use PAL macros
  - Most portable
  - Can use PAL Virtual Platform

Tradeoffs

- The lower the level (pins), the less the overhead, the less portable, and the more coding effort required
- Higher levels are okay to use
  - Macros are written by experts, so are efficient
  - Compiler optimizes your code
  - Get it to work first, then optimize the hotspots if performance is not satisfactory

Keyboard to Hex Display

- See Keyboard Project in Celoxica Examples
- When simulating, you have to enter *scan codes*.
  - See Laboratory IV Handout
- Celoxica code uses PAL Console for output to LCD
  - Your assignment is to draw seven segment displays on the LCD

Keyboard to LCD

- Version I – Use RC200E Macros
  - No simulation
- Use PAL Macros
  - Version II – Draw a Pattern on LCD
  - Version III – Draw Seven Segment Displays
  - Version IV – Integrate with Keyboard Input

Simulator *vs.* Pin I/O

- Simulator uses chanin/chanout to connect to files.
  - chanin unsigned Input with {infile="./data"};
  - chanout unsigned Output;// debug window
  - Note that channels have data types
- Use interfaces to connect to pins.
  - bus_in, bus_out