Hardware simulation and test vectors

- ‘Recent’ Work
- Plans
- Details of CP-chip test vectors
- Experience with debugging FPGA

Recent work (since July!)

- Development of GTM model
  - Uncovered nasty bug in ostrstream
- Addition of several sets of CP-FPGA test-vectors
- Code put into RAL cvs repository
- Various bits of tidying up
  - Obeys more ATLAS coding conventions
- Debugging FPGA code with James

Still to do... (lots)

- More CP test-vectors
- Full integration of Bill's work
- Extensive comments and User Documentation
- Serious thoughts about integration with rest of world

CP FPGA test-vectors

- For straight FPGA simulation and GTM set-up we have test-vectors for:
  - ‘Physics’ data (Alan’s original vectors)
  - Random data (carefully tuned)
  - Threshold behaviour
  - Sum testing
- To do:
  - Full BC-demux treatment

Experience with VHDL simulation and real hardware

- Looking from the outside:
  - 108 inputs at 160 MHz
  - 40 outputs at 40 MHz
  - ie huge loss of information
- Still possible (but difficult) to diagnose obvious large errors, eg
  - Incorrect overflow behaviour in CP FPGA
  - Reversal of eta in GTM set-up
  - Data for one cell in wrong place in GTM set-up
    - Required VHDL to look internally at signals
- Less obvious problems require more work
  - OK in VHDL, possible in final system?

Conclusion

- Debugging FPGAs is going to be tough!
  - Some of you probably knew this already
- Have we got enough tools to do it?
  - Extended VHDL simulations?
  - Chip-scope?
- Need well tested stable code before we have lots of boards