A Network of Time Division Multiplexing for FPGAs

Rosemary Francis
Motivation

- FPGAs are now home to complex Systems on-Chip
- Designs require the use of Network-on-Chip
- FPGA global wiring is simple in comparison with ASIC Networks-on-Chip
- Networks for FPGAs use lots of wires or lots of logic
- Hard blocks are limited by the soft IP blocks
Goals

• Improve wiring density through TDM
• Use TDM components for effective soft NoC implementation
• Funnel data to high-speed hard blocks
  – Hard NoC
  – Multipliers
  – Block RAM
Hierarchy of interconnect

Coarse-grain packet-switched network

Time-division multiplexed wires in a fine-grain network

Clusters of logic elements with local interconnect
Architecture: Stratix vs TDM

Switch box

Stratix Global routing

Local routing

Cluster of logic elements

TDM Global routing

Switch box

SRAM

Local routing

Cluster of logic elements with latched inputs

LUT
Wire Sharing

• Many wires can be shared without a problem
Wire Sharing

• Many wires can be shared without a problem
• Other configurations require a more intelligent approach

Conflict!!
Wire Sharing

• Many wires can be shared without a problem
• Other configurations require a more intelligent approach
• Signals can be delayed to allow more efficient wire use without rerouting
Parameter selection

- Assume infinite time slots to reduce wiring
  - Determine optimum number of TDM wires
Infinite resources

Number of TDM wires

Total number of wires needed
Parameter selection

- Assume infinite time slots to reduce wiring
  - Determine optimum number of TDM wires
- Vary number of time slots
  - Determine optimum number of time slots
  - Investigate the effect this has on latency
Determine number of time slots

Number of time slots (= number of configurations bits per mux)

Wires per switch box
Number of time slots vs latency

Number of time slots (number of configuration bits per mux) vs Normalised latency of critical path.
Parameter selection

• Assume infinite time slots to reduce wiring
  – Determine optimum number of TDM wires
• Vary number of time slots
  – Determine optimum number of time slots
  – Investigate the effect this has on latency
• Using optimum number of time slots
  – Re-evaluate optimum number of TDM wires
Limited resources
Architectural drawbacks

- Extra configuration SRAM
- High-speed interconnect clock
- Benchmarks run over three times slower
- New CAD tools needed
  - Re-routing in space as well as time
  - Optimise for TDM wiring at every stage
Conclusions

• Using TDM wiring we can reduce the number of wires whilst increasing the data rate within channels
  – 75% less wiring * 24 time slots * 3 times slower means 2 times channel data rate

• This will allow
  – the design of effective global interconnect
  – more efficient sharing of on-chip resources
  – simplification of multi-chip designs
Future Work

• Current scheduling algorithm gives
  • Large wire reduction
  • Large latency penalty

• Is there a better compromise?
  • Halve the wiring, small latency penalties

• How can we reduce latency in other ways?
  • Better scheduling algorithms
  • Circuit redesign
Thanks for listening...

Rosemary.Francis@cl.cam.ac.uk