HDCompiler

-Translate Esterel

-From PDG to Optimal Circuit

Jia Zeng

What's the Aim

- Optimal Circuitry
  - Approaches: PDG, State Machine
  - Procedures:
    - Encoding;
    - Circuit Translation;
    - Optimization;

What we did

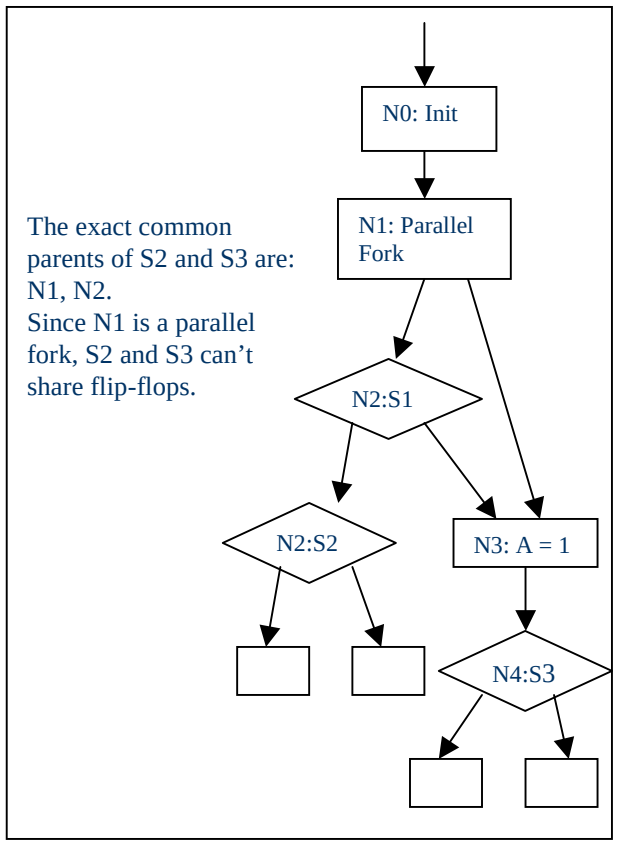
- State Encoding
  - One-hot, Compact, Combining
- Turning State Machine into Circuitry
  - Encoding State (i.e. decoding bits)
    - bit[i] = OR(Σ j Wires for state-value-emit statements for state_value[j] where bit[i] = 1 in the value code)
  - Decoding State
    - AND (State machine's entry, State value decoded)

What we did (cont)

- Flip-flops Sharing
  - Qualification:

    (Exact Common Parent

    ! = Parallel Fork)
  - For all shareable State Machines, Which should be chosen?



What we did (cont)

- Slack Computing
  - Why compute slacks
    - An upper bound of possible delay increase without violating the timing constrain
    - Represent of the potential capability of obtaining area/power reduction
  - How to compute
    - $arrival\_t\ (v)=MAX_{\ u \in FI(v)}\ (arrival\_t(u))+delay(v)$
    - $required\_t\ (v)=MIN_{\ w \in FO(v)}\ (required\_t(u)-delay(w))$
    - $slack\ (v) = required\_t\ (v) – arrival\_t\ (v)$

How well we did

- Circuits comparing before/after optimizing
  - see web page:http://www.cs.columbia.edu/~jia/testrecord/
- Circuits comparing between different encoding means:

	Flip-flops	Gates	Wires	Slacks
One-hot	10	56	80	16
Compact	6	59	85	22

Future Work

- Just a structure for real hardware translator, much more future work:
  - Choose state encoding means
  - Optimize based on slacks
  - Take real PDG input
  - Catch bugs