Cognichip Hackathon
Network Arbiter

Gary Guan, Az Li

Problem Statement

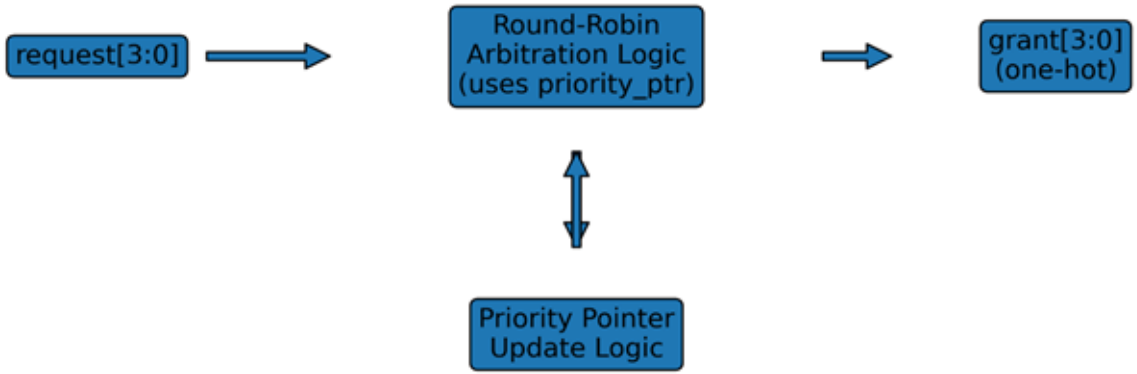
- Network arbiter chip designs is an extremely specification sensitive challenge in design
- Design, algorithm, or policy errors in arbitration logic can result in reduced performance
    - Many services, such as data centers, rely on high performing arbitration logic for performance
- Using Cognichip, an AI-driven hardware design tool, allows hardware engineers to speed up the design process, verify implementation, and implement more custom arbiter designs
    - Cognichip will allow engineers to reduce cost and turnaround time

Design Methodology

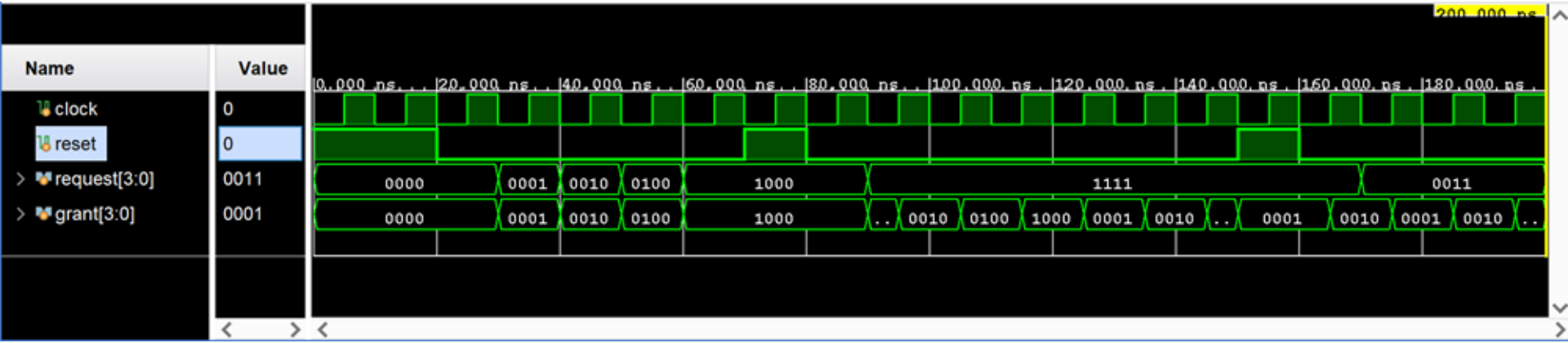
- We tested a lightweight but capable 4 input round robin arbiter
  - 4 ports that can be used for transmit and receive (input and output)
- After succeeding, we then built a 8x8 network switch arbiter
- We used the Congnichip platform to design, test, and validate the chip
- Cognichip’s platform allowed us to quickly generate Verilog designs
- Built in simulation and waveform viewers allowed all the steps to be completed in Cognichip

4-Input Round-Robin Arbiter

- 4 request inputs (request[3:0])
- 1 shared resource
- One-hot grant output (grant[3:0])
- Rotating priority pointer (priority_ptr)



Simulation Results (4)



Architecture (8x8)

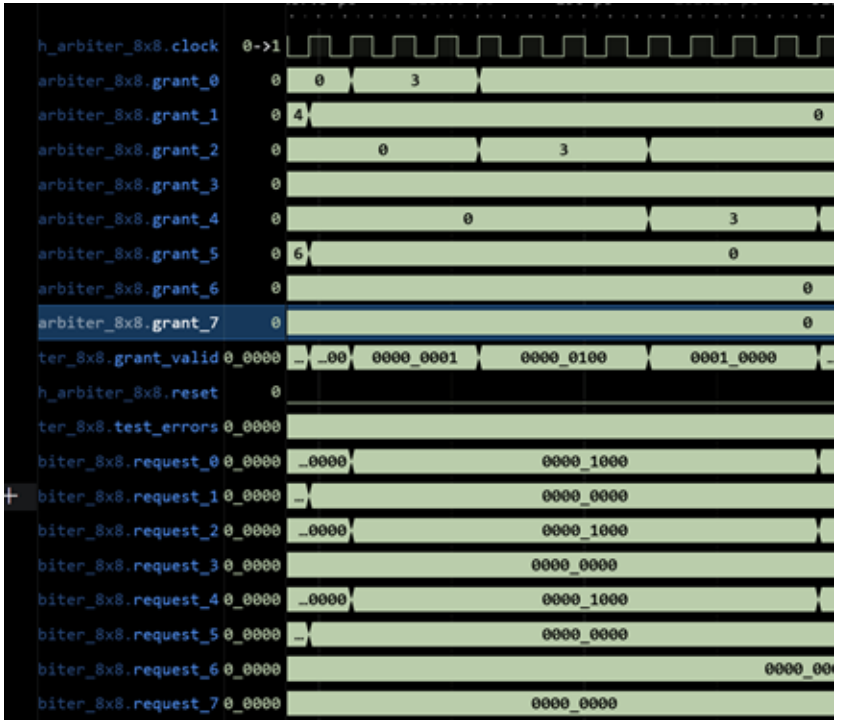
- 8x8-bit request vectors
- 8-bit acknowledgement bus
- Eight 3-bit grant signals
- 8-bit grant valid vector
- Synchronous reset
- 8x8 crossbar arbiter with one independent round-robin arbiter for each output port
    - Allows for a maximum of 8 grants per cycle when there are no conflicts
    - Conflicts are resolved using a round-robin approach

Simulation Results (8x8)

- Using the Verilog design and testbench generated from Cognichip, we were able to pass all test cases
- We did realize that the current

  design lacks an idle state, but

  defaults to 0



Github Repository

- The full design, testbench, and simulation files for both the 4x4 and the 8x8 design can be found in our Github Repository

https://github.com/azdli/Cognichip_NetworkArbiter

Challenges

- We sometimes had some bugs that Cognichip was unable to fix on its own, so we had to inform the AI of how to implement a fix
- Message limits
- Errors getting the simulation tool to work reliably
- Sometimes it gets stuck trying to do the same task and failing over and over again

Future Work

- We can explore designs involving a larger radix (such as 16x16)
- We could also explore alternative arbiter algorithms
- Our current design could be implemented through a FPGA board or as a Tiny Tapeout chip