The Vending Machine

Martin Schoeberl

Technical University of Denmark
Embedded Systems Engineering

April 13, 2023

Overview

- Your final grade
- Online exam
- GoL in hardware
- The Vending Machine project
-
- How did it go with the UART?

Your Final Grade

1. Your lab work, the vending machine
   - What is working (TA checks)
   - Your report
   - Basic functions is a 7, extra functions needed for a 10 or 12
2. Written exam
   - Two hour written exam

Exam Topics and Questions

- The pensum (reading lis) is on the web site
- Compute maximum frequency and delays of a given circuit
- Given a Chisel description of a circuit, draw it
- Given a circuit drawing, sketch the Chisel description
- Basically what we have done in the lab
- No surprises (at least not too many ;-)

Game of Live

- Conway's Game of Life
- Any live cell with two or three live neighbors survives.
- Any dead cell with three live neighbors becomes a live cell.
- All other live cells die in the next generation. Similarly, all other dead cells stay dead.

Game of Live

- You did an implementation in Java
- The problem is highly parallel
- I will show you a Chisel (and Java) implementation
- FPGA version is extremely fast compared to the Java implementation
- https://github.com/schoeberl/game-of-live

Performance Comparison

World	Cells	Execution time (us)			FPGA Speedup	
		Mac	Rasperry	FPGA	Mac	Rasperry
10x10	100	0.10	1.783	0.0040	25	445
20x20	400	0.33	5.137	0.0040	82	1284
30x30	900	0.70	9.965	0.0041	170	2430
40x40	1600	1.21	17.212	0.0040	302	4302
50x50	2500	1.81	25.204	0.0044	411	5728
60x60	3600	2.76	37.822	0.0045	613	8404
70x70	4900	3.54	57.665	0.0040	884	14416
80x80	6400	4.81	64.396	0.0047	1023	13701
90x90	8100	6.50	81.309	0.0045	1444	18068
100x100	10000	7.51	109.964	0.0048	1564	22909

A Vending Machine from 1952

Source: Minnesota Historical Society, CC BY-SA 2.0

The Vending Machine

- Final project is a vending machine
- Description is on GitHub: README.md
- Will repeat the overview now
- Group work
- Final version shall be run in an FPGA
- A lot can be done with testing and simulation

The Vending Machine

- Inputs: coins, buy
- Display: price and current amount
- Output: release can or error
- Small challenge to multiplex the display
- State machine with data path is the *brain* of the VM
- Guided step by step over several weeks

Vending Machine Specification I

- Sell 1 item and not returning any money
- Set price with 5 switches (1–31 kr.)
- Display price on two 7-segment displays (hex.)
- Accept 2 and 5 kr. (two push buttons)
- Display sum on two 7-segment displays (hex.)
  - Amount entered so far
- Does not return money, left for the next purchase

Vending Machine Specification II

- Push button *Buy*
  - If not enough money, activate *alarm* as long as buy is pressed
  - If enough money, activate *release item* for as long as *buy* is pressed and reduce *sum* by the price of the item

Optional Extras

- Needed for a 10 or 12
- Display decimal numbers
- Supplement alarm by some visuals (e.g., blinking display)
- Count coins and display an alarm when compartment is full ($>$ 20 coins)
- Have some text scrolling on the display
- Connect a UART to your VM and sending messages to your laptop
- ...
- Your ideas :-)

Design and Implementation

- Implementation shall be a state machine plus datapath
- Design your datapath on a sheet of paper
- Datapath
  - Does add and subtract
  - Contains a register to hold the sum
  - Needs some multiplexer to operate
- Display needs multiplexing
  - Implemented with some counters and a multiplexer
- Show each part of your design to a TA
  - 7-segment decoder, 7-segment with a counter, display multiplexer, complete vending machine

Draw Figures

- Drawings/schematics is another language to describe (digital) circuits
- Draw, draw, draw boxes and arrows!
- Use drawing during development
- If you cannot draw your circuit you do not understand it
- Use drawings to communicate with the TA
- Have drawings in your report
- You will *for sure* need to draw circuits at the exam ;-)

Vending Machine Design and Implementation Steps

- We started in week 6 (now we are in week 10)
- lab 6: Hexadecimal to 7-segment decoder and counter
- lab 8: Multiplexed Seven-Segment Display
- lab 10–13: Complete Vending Machine
- *Show your working design to a TA*

Final Report

- One report per group
- A single PDF
  - Your group number is part of the file name (e.g., group7.pdf)
  - Code as listing in an appendix (no .zip files)
  - Hand in in DTU Inside
- Content
  - Abstract
  - Preface (Who did what)

1. Introduction and Problem Formulation
2. Analysis and Design
3. Implementation
4. Testing
5. Results
6. Discussion
7. Conclusion

- List of References
- Appendix: Chisel code

Material on the Lab GitHub

- A top-level component
- XDC file for Basys pins and frequency
- A start of a tester generating waveforms
- A simulation of the board
- Show it (in IntelliJ)

An Optional Lab

- Testing the a Vending Machine
- Black box testing (you don't see the implementation)
- I give you two implementations
- One is OK, one is broken
- Which one is broken, and what it the error?
- Issue is that you need Verilator and a C compiler to run the tests
- Therefore, only if you really want to do it
- Lab 10

Questions on Final Project?

Summary

- Now you have four weeks for the Vending Machine
- Should be plenty of time
- Standard solution is good for a standard grade
- Add features as you like
- Have a good time with your Vending Machine construction