EECS150 - Digital Design
Lecture 28 Course Wrap Up

Dec. 5, 2013
Prof. Ronald Fearing
Electrical Engineering and Computer Sciences
University of California, Berkeley

(slides courtesy of Prof. John Wawrzynek)

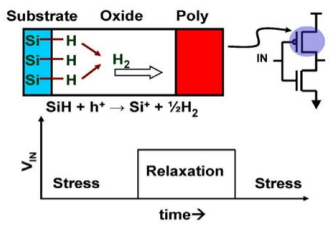
http://www-inst.eecs.berkeley.edu/~cs150

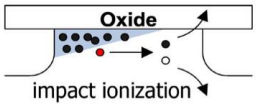
Recap 1

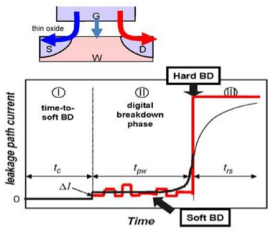
- *Single Event Upset*

$$Failure\ Rate = \frac{x^2 10^9}{2(No.\ of\ Devices)(No.\ of\ Hours)(Acc.\ Factor)}$$

- *Hard Faults:* negative bias temperature instability), HCI (hot carrier injection), TDDB (time-dependent dielectric breakdown) , Electro-migration







Recap 2

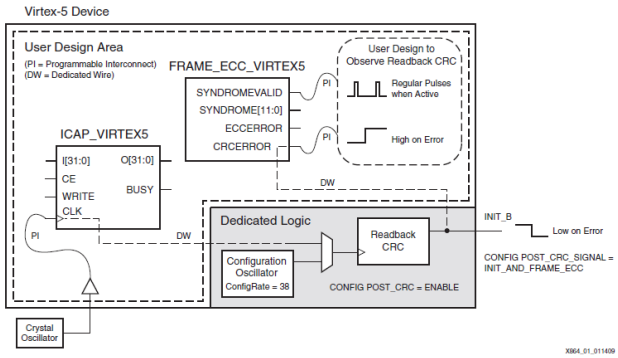


Figure 1: **Readback CRC Arrangement in a Virtex-5 FPGA**

Mitigation:
•CRC checks on configurat
•ECC
p1 = d1 xor d2 xor d4
p2 = d1 xor d3 xor d4
p3 = d2 xor d3 xor d4
Check:
c1 = p1 xor d1 xor d2 xor d4,
etc

Hamming Code

1	2	3	4	5	6	7
p_1	p_2	d_1	p_3	d_2	d_3	d_4

3

Outline

- *What's next for digital design?*
- *Course Summary*
- *Example Final Problems F99 (can be found at:*

http://www-inst.eecs.berkeley.edu/~cs150/fa03/misc.html)

4

Why Study and Learn Digital Design?

- We expect that many of our graduates will eventually be employed as designers.
  - ***Digital design is not a spectator sport.*** The only way to learn it, and to appreciate the issues, is to do it.
  - To a large extent, it comes with practice/experience (this course is just the beginning).
  - Another way to get better is to study other designs. Not time to do this during the semester, but a good practice for later.
- However, a significant percentage of our graduates will not be digital designers. What's in it for them?
  - Better manager of designers, marketers, field engineers, etc.
  - Better researcher/scientist/designer in related areas
    - Software engineers, fabrication process development, etc.
  - To become a better user of electronic systems.

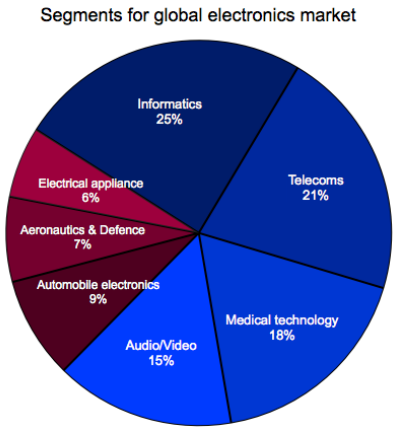
In What Context Will You be Designing?

Engineers learn so that they can build.
Scientists build so that they can learn.

- Electronic design is a critical tool for most areas of pure science:
  - Astrophysics – special electronics used for processing radio antenna signals.
  - Genomics – special processing architectures for DNA string matching.
  - In general - sensor processing, control, and number crunching. In some fields, computation has replaced experimentation – particle physics, world weather prediction (fluid dynamics).
- In computer engineering, prototypes often designed, implemented, and studied to "prove out" an idea. Common within Universities and industrial research labs. Lessons learned and proven ideas often transferred to industry through licensing, technical communications, or startup companies.
  - RISC processors where first proved out at Berkeley and IBM Research

Designs in Industry

- Of course, companies are the primary employer of designers. Provide desired products to society or government and make a profit for the shareholders.



Global Semiconductor market $300B
Global electronics ~$1600B

The Big Ideas from EECS150

1. **Modularity and Hierarchy** is an important way to describe and think about digital systems.
2. **Parallelism** is a key property of hardware systems and distinguishes them from serial software execution.
3. **Clocking** and the use of state elements (latches, flip-flops, and memories) control the flow of data.
4. **Cost/Performance/Power tradeoffs** are possible at all levels of the system design.
5. Hardware Description Languages **(HDLs) and Logic Synthesis** are a central tool for digital design.
6. **Datapath + Controller** is a effective design pattern.
7. **Finite State Machines** abstraction gives us a way to model any digital system – used for designing controllers.
8. **Arithmetic circuits** are often based on “long-hand” arithmetic techniques.
9. **FPGAs** give us a convenient and flexible implementation technology.

The Useful Skill from Class

We hope that after have taken this class that …

Given an English language description for the function of a digital system covering any of a wide variety of domains:

You can organize and describe a digital system, and

using Verilog and logic synthesis, generate a detailed circuit at the "logic gate level", and

map to an FPGA, and

debug it, and

optimize for cost or performance or both.

What We Didn't Cover

- Design Verification and Testing
  - Industrial designers spend more than half their time testing and verifying correctness of their designs.
    - Some of this covered in the lab. Didn't cover rigorous testing procedures and "formal verification".
  - Most industrial products are designed from the start for testability. Important for design verification and later for manufacturing test.
  - Built in self test (BIST), Automatic Test Vector Generation, Scan-chain techniques.
- Other High-level Optimization Techniques
  - Automatic Retiming (although FPGA tools do it)
- High-level Architectures: CPUs, video processing, network routers, ...
- Asynchronous Design

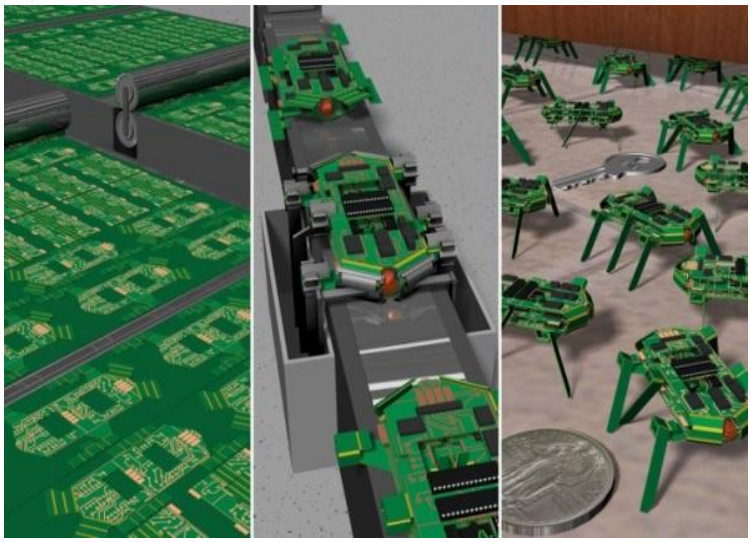
Most Closely Related Courses

- CS152 Computer Architecture and Engineering
  - Design and Analysis of Microprocessors
  - Applies basic design concepts from EECS150
- EE141 Digital Integrated Circuits
  - Transistor-level design of ICs
  - Understand how EECS150 circuits are mapped to silicon (ASICs)
- CS250 VLSI Systems Design
  - Learn how to design cell-based ASICs
  - Advanced-undergrad/grad course
  - "New" format, now design-based

Future Design Issues

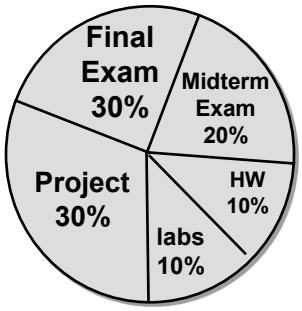
- Automatic High-level synthesis and optimization (with micro-architecture synthesis) and hardware/software co-design.
- Current trend is towards "system on a chip" (SOC) design methodology:
  - Pre-designed subsystems (processor cores, bus controllers, memory systems, network interfaces, etc.) connected with standard on-chip interconnect or bus.
- Increasing NREs will favor post-fabrication customization.
- Unreliable devices
- A number of alternatives to silicon VLSI have been proposed, including techniques based on:
  - molecular electronics, quantum mechanics, and biological processes. "Nano-devices"
  - How will these change the way we design systems?

Application for Printable Electronics



13

Course Grading & Final Exam



- **Exam** held in scheduled final exam slot: Tuesday Dec. 17, 0810-1100
- Room: **306 Soda Hall**
- "Comprehensive" Final Exam: covers material from the entire semester with emphasis on second half
- 30-40% of final will cover new material since Midterm exam - week 9, and on, *including guest lectures*.
- 70-60% of final will cover semester-long topics.

Important Topics from Second Half

- Timing
  - Relation of clock speed to performance.
  - Determination of maximum clock frequency from circuit.
  - Origin of logic delay.
  - Origin of flip-flop delay.
  - D type FF with transmission gates and cross-coupled inverter
  - Wire delay and mitigation.
  - Effects of clock skew.
  - Principle of flip-flop metastability.
  - Clock synchronization.

Important Topics from Second Half

- FSMs
  - Design and operation of finite state machines (FSM) as synchronous logic circuits.
  - "By hand" design procedure from state-transition-diagram (STD) to FSM circuit implementation.
  - Design procedure for STD to "one-hot" encoded FSM circuit.
  - Moore versus Mealy machine STDs and implementations and timing behavior.
  - Moore versus Mealy in Verilog specifications.

Important Topics from Second Half

- Adders
  - Carry-select adder design principle, cost/performance analysis, optimization.
  - Carry look-ahead adder design principle, cost/performance analysis, optimization.
  - Bit-serial adders design and operation.
  - Virtex-5 adder carry-chain.
- Multiplication
  - Binary multiplication principle.
  - Shift-and-add multiplier design and operation.
  - Extending multiplication techniques for signed multiplication.
  - Bit-serial multiplier structure and operation.
  - Combinational (array) multiplier structure and operation.
  - Cost performance analysis of alternative multiplication schemes.

Important Topics from Second Half

- Multiplication
  - Carry-save addition technique and application to multiplier design.
  - Multiplication by a constant, including canonic signed digit representation (CSD), and KCM factoring technique.
- Shifters and Cross-bar switch circuit
- Counters
  - Counters in controller design.
  - Binary up/down counter design.
  - "Asynchronous" counter circuit.

Important Topics from Second Half

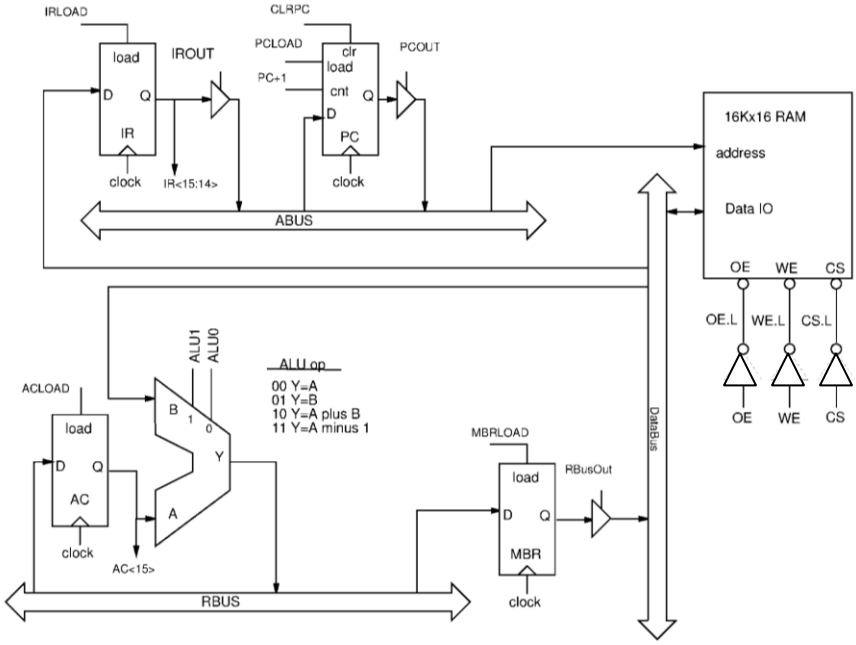
- Power and Energy Basics
  - Definitions of power and energy
  - Power consumption in CMOS circuits
  - Techniques for energy efficient design

- Faults and Error Correction
  - Types of faults in digital systems
  - Construction of Hamming codes for single error correction (SEC and double error detection (DED).

Important Topics from Second Half

- Graphics Processor Units
  - architectural features for high speed

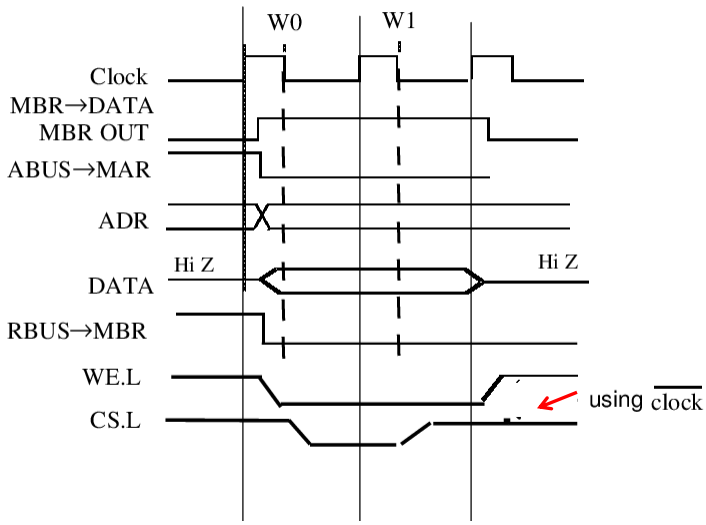
Datapath



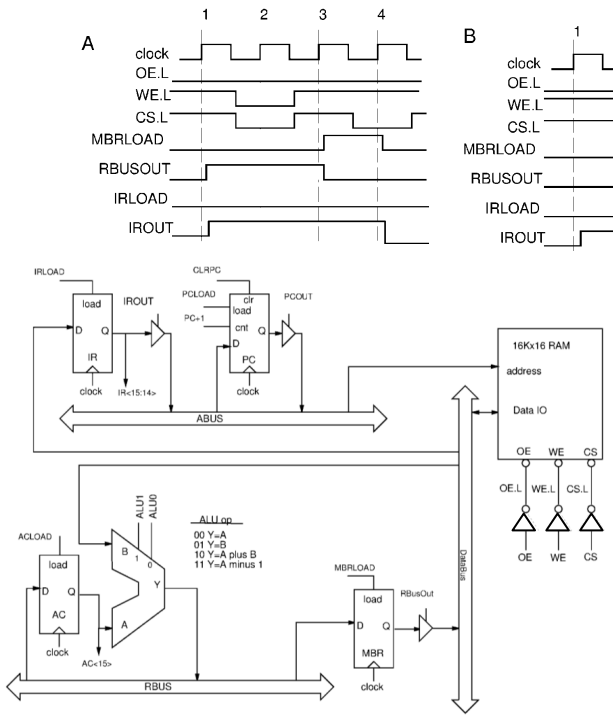
21

external RAM Memory interface

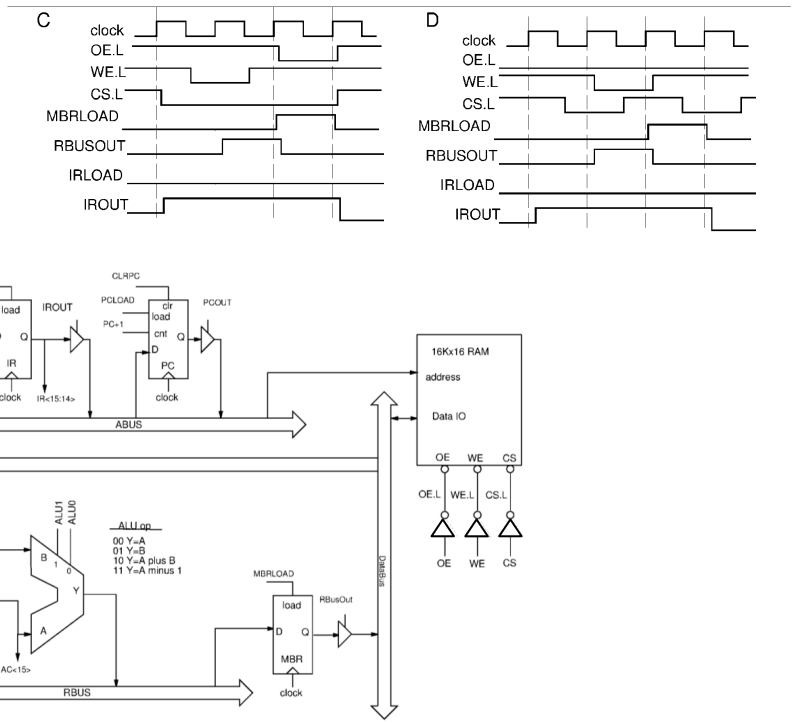
Write cycle (must be clean)



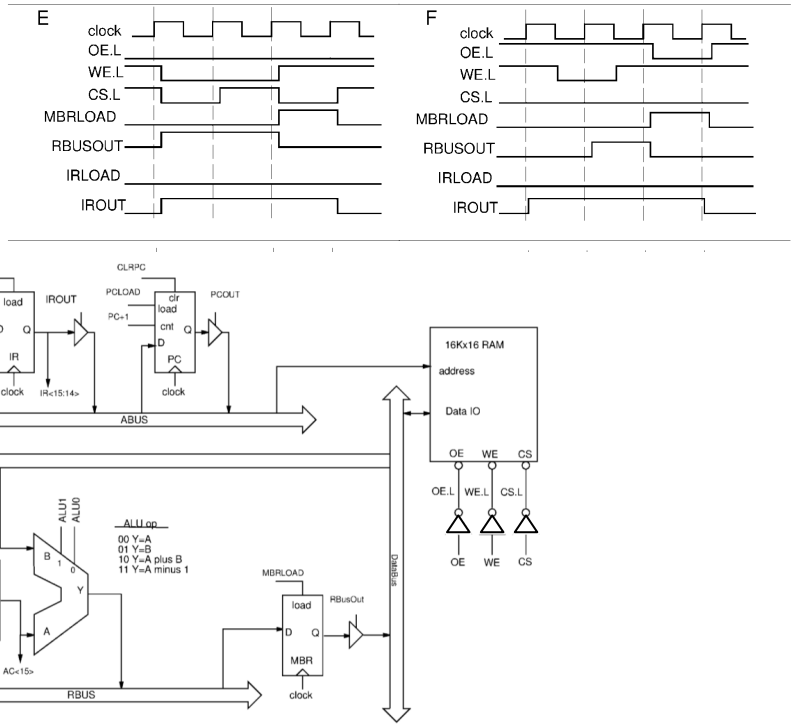
22


A
1
2
3
4
clock
OE.L
WE.L
CS.L
MBRLOAD
RBUSOUT
IRLOAD
IROUT
B
1
2
3
4
clock
OE.L
WE.L
CS.L
MBRLOAD
RBUSOUT
IRLOAD
IROUT
IRLOAD
load
IROUT
D
Q
IR
clock
IR<15:14>
CLRPC
PCLOAD
clr
load
PC+1
cnt
PCOUT
PC
clock
16Kx16 RAM
address
Data IO
OE
WE
CS
OE.L
WE.L
CS.L
OE
WE
CS
ABUS
ALU1
ALU0
ALU op
00 Y=A
01 Y=B
10 Y=A plus B
11 Y=A minus 1
ACLOAD
load
AC
clock
AC<15>
B
A
Y
MBRLOAD
load
RBusOut
MBR
clock
DRBUS
RBUS

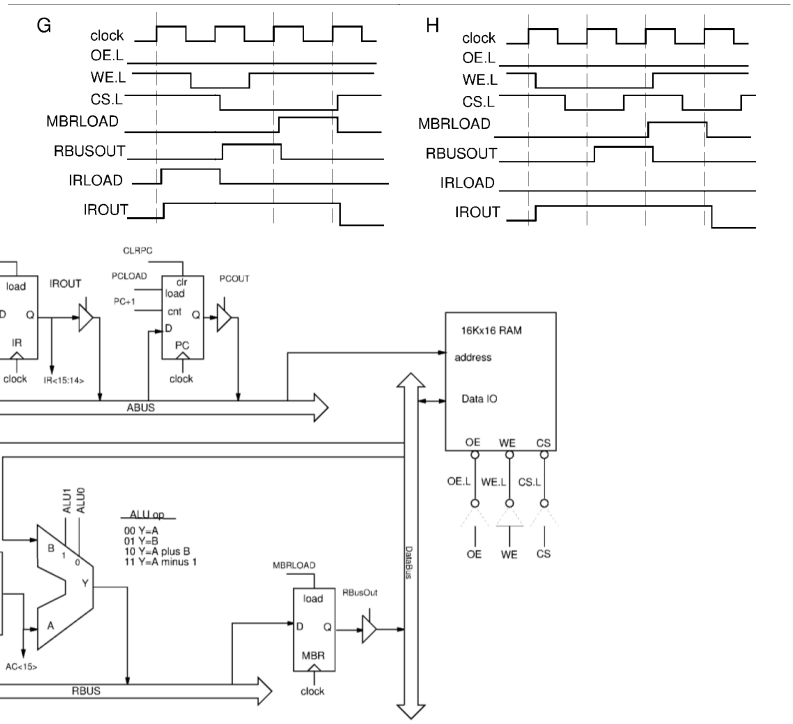
23


C
clock
OE.L
WE.L
CS.L
MBRLOAD
RBUSOUT
IRLOAD
IROUT
D
clock
OE.L
WE.L
CS.L
MBRLOAD
RBUSOUT
IRLOAD
IROUT
IRLOAD
load
IROUT
D
Q
IR
clock
IR<15:14>
CLRPC
PCLOAD
clr
load
PC+1
cnt
PCOUT
PC
clock
16Kx16 RAM
address
Data IO
OE
WE
CS
OE.L
WE.L
CS.L
OE
WE
CS
ABUS
ALU1
ALU0
ALU op
00 Y=A
01 Y=B
10 Y=A plus B
11 Y=A minus 1
ACLOAD
load
AC
clock
AC<15>
B
A
Y
MBRLOAD
load
RBusOut
MBR
clock
DRBUS
RBUS

24

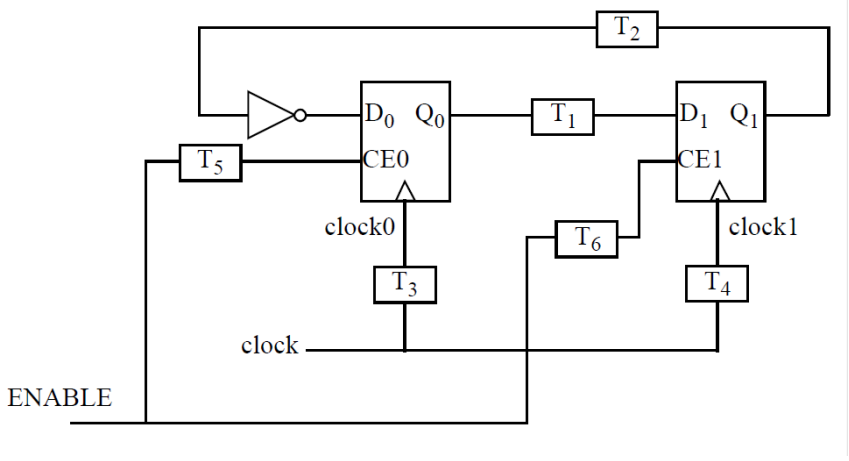

E
clock
OE.L
WE.L
CS.L
MBRLOAD
RBUSOUT
IRLOAD
IROUT
F
clock
OE.L
WE.L
CS.L
MBRLOAD
RBUSOUT
IRLOAD
IROUT
IRLOAD
load
IROUT
D
Q
IR
clock
IR<15:14>
CLRPC
PCLOAD
PC+1
clr
load
cnt
PC
PCOUT
ABUS
16Kx16 RAM
address
Data IO
OE
WE
CS
OE.L
WE.L
CS.L
ALU1
ALU0
ALU op
00 Y=A
01 Y=B
10 Y=A plus B
11 Y=A minus 1
ACLOAD
AC
B
A
Y
AC<15>
RBUS
MBRLOAD
RBusOut
MBR
DRBUS

25


G
clock
OE.L
WE.L
CS.L
MBRLOAD
RBUSOUT
IRLOAD
IROUT
H
clock
OE.L
WE.L
CS.L
MBRLOAD
RBUSOUT
IRLOAD
IROUT
IRLOAD
load
IROUT
D
Q
IR
clock
IR<15:14>
CLRPC
PCLOAD
PC+1
clr
load
cnt
PC
PCOUT
ABUS
16Kx16 RAM
address
Data IO
OE
WE
CS
OE.L
WE.L
CS.L
ALU1
ALU0
ALU op
00 Y=A
01 Y=B
10 Y=A plus B
11 Y=A minus 1
ACLOAD
AC
B
A
Y
AC<15>
RBUS
MBRLOAD
RBusOut
MBR
DRBUS

26

F99, prob 6, timing



The End.

- Special thanks to Austin and Stephen
- Good luck on the final.
- Thanks for a great semester!