Look Dear, Christmas carolers.
♫ Hark the ♫ teacher dropped the final, fa-la-la-la-la ♪
Aaron, I will not drop the final.
So should we leave or can we still sing Jingle Bells, We Would Like an A?

EE319K Final Review Spring 2012

Valvano Homework

There were 10 homeworks, and no pop quizzes. I will average the 10.

The final will be similar in style to Quiz1 and last year's Final

The final will cover material from book as listing in the reading assignment.
See course descriptor for a list of reading assignments
Good things to study are
the Homework questions and solutions
all the lecture notes aLec1 to aLec37 posted
C programming, Patt Book (C code and no LC3 assembly)
Chapter 12 variables
Chapter 13 control structures
Chapter 14 functions
Chapter 16 pointers and arrays
Chapter 14 functions
Section 19.2 structures
Or C primer http://users.ece.utexas.edu/~ryerraballi/CPrimer/
No C libraries required
None of these: stdio.h math.h string.h, malloc & free

It will be closed book, you can only have pen, pencil and eraser
It will be half short answer questions
Conversions, definitions, hand execution, bus cycles, hand assembly
And half longer questions
Local variables, FSM, SCI input/output, OC interrupt, FIFO, and ADC input
I will give you similar technical documents as I did with last year's Final

The numerical questions will be simple enough not to require a calculator. Except for the number conversion and hand assembly questions, you can get full credit if you show me the correct fundamental equation (even if you calculate the wrong answer).

Final review

Number conversions, 8-bit (fill in the blank)
convert one format to another without a calculator

signed decimal	e.g., **-56**
unsigned decimal	e.g., **200**
binary	e.g., **%11001000**
hexadecimal	e.g., **$C8**

volatile, nonvolatile
static efficiency, dynamic efficiency
structured program, flowchart, call graph, data flow graph
basis, precision,
fixed-point,
given resolution convert between value and integer
value = integer*Δ
given precision and range choose the fixed-point format
promotion, demotion
setting N, Z, V, C after 8-bit add or subtract
overflow, ceiling and floor, drop out,
bus, address bus, data bus,
bus cycle, read cycle, write cycle,
simplified cycle by cycle execution
assembly listing to execution cycles
machine code to execution cycles
IR, EAR, BIU, CU, ALU, registers,
device driver,
friendly,

```
    ldaa DDRT
    oraa #$80
    staa DDRT

    bset DDRT,#$80
```

mask, toggle,

```
    ldaa PTT
    eora #$80
    staa PTT
```

8-bit versus 16-bit data

reset vector, output compare vector, SCI interrupt vectors

direction register

baud rate, bandwidth, frame, start bit, stop bit, parity

busy-waiting, gadfly, or polling

Producer/consumer problem (buffered I/O)

FIFO queue implementation

FIFO queue usage

flow chart, call graph, data flow graph

device driver

public versus private

ADC 10-bit 0<Vin<5 to 0<Dout<1023 If Vin=2.5V, then Dout =512

8-bit 0<Vin<5 to 0<Dout<255 If Vin=2.5V, then Dout =128

Sample multiple channels: e.g., 4 5 and 6

Straight binary format, right justify

resolution, range, precision

```
Timer_Wait
      addd TCNT      ;TCNT at end of delay
Wloop cpd  TCNT      ;is EndT<TCNT
      bpl  Wloop
      rts
```

debugging

ScanPoint, intrusiveness, monitor, profiling, dump

measuring execution time, desk check, stabilization

functional debugging, performance debugging, profiling

Real time systems, bounded latency

Finite state machine, Mealy, Moore, linked list, table

Electronics with voltages and resistors

Ohms Law in resistor V=I*R

What is voltage, current, resistance

KCL, KVL, voltage divider, current divider

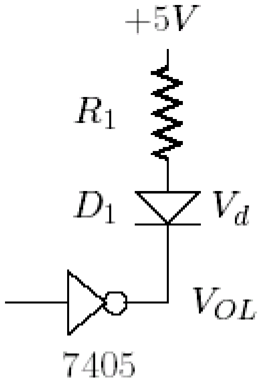
Power = V*I

Switch

Positive logic with 10k pull-down

Negative logic with 10k pull-up

LED interfacing (LED voltage versus current relation)



What is I_d?

What is V_d?

What is V_{OL}?

$5=I*R+V_d+V_{OL}$

decimal digits	exact range	exact alternatives	approximate bits
3	0 to 999	1,000	≈ 10
3½	0 to 1999	2,000	≈ 11
3¾	0 to 3999	4,000	≈ 12
4	0 to 9999	10,000	≈ 13
4½	0 to 19,999	20,000	≈ 14
4¾	0 to 39,999	40,000	≈ 15
5	0 to 99,999	100,000	≈ 17
5½	0 to 199,999	200,000	≈ 18
5¾	0 to 399,999	400,000	≈ 19
6	0 to 999,999	1,000,000	≈ 20
6½	0 to 199,999	2,000,000	≈ 21
6¾	0 to 3,999,999	4,000,000	≈ 22
N	0 to 10^N-1	10^N	≈ N*log$_2$(10)
N½	0 to 2*10^N-1	2*10^N	≈ N*log$_2$(10)+1
N¾	0 to 4*10^N-1	4*10^N	≈ N*log$_2$(10)+2

Standard definition of decimal digits.

Arrays and strings (address math, scanning, variable size)

scope local versus global

allocation dynamic vs permanent

local variables and parameters on the stack

S binding, X stack frame binding, Y stack frame binding

binding , allocation, access, deallocation

call by reference, call by value

`emul ediv rti` instructions

Practice Question 1 (hand execute and draw the stack picture)

```
P1 set  ???
P2 set  ???
L1 set  ???
Subroutine
    leas –2,sp   ;allocate L1

    ldd P2,sp
    addd P1,sp
    std L1,sp

    leas 2,sp
    rts
Main lds  #$4000
```

```
     movw #1000,2,-sp  ;p1
     movw #2000,2,-sp  ;p2
     jsr  Subroutine
     leas 4,sp
```

Practice Question 2

```
P1 set  ???
P2 set  ???
L1 set  ???
Subroutine pshx
    tsx
    leas –2,sp   ;allocated L1
    ldd  P2,x
    addd P1,x
    std L1,x
    txs
    pulx
    rts
Main lds  #$4000
     movw #1000,2,-sp  ; p1
     movw #2000,2,-sp  ; p2
     jsr  Subroutine
     leas 4,sp
```

Spring 2011	average 71
Fall 2010	average 78
Spring 2010	average 77
Fall 2009	average 79

Spring 2011 Final (should be able to write in both C and assembly)

Question 1. Definitions
Question 2. ADC parameters
Question 3. Ohms Law
Question 4. Fixed-point software
Question 5. C and assembly functions accessing an array
Question 6. LED interface
Question 7. Tricky stack question
Question 8. C programming using `struct` (*study this*)
Question 9. General question about overflow
Question 10. Output compare interrupt service routine
Question 11. Understanding PC, SP and stack during a subroutine call
Question 12. SCI output ISR (*hard*)

Fall 2010 Final (should be able to write in both C and assembly)

Question 1. DAC parameters
Question 2. SCI baud rate
Question 3. Ohms Law
Question 4. Fixed-point software
Question 5. SCI busy-wait software
Question 6. LED interface
Question 7. Stack instructions, addressing modes
Question 8. Overflow bug
Question 9. Carry overflow bit
Question 10. Sequence of events during an interrupt
Question 11. Cycle by cycle execution
Question 12. Draw SCI output wave
Question 13. FSM software in C, ADC sample software
Question 14. Local variables in assembly with stack frame addressing
Question 15. Fifo implementation in assembly

Fall 2009 Final (should be able to write in both C and assembly)

Question 1. Specifications of a DAC
Question 2. DAC output, Ohms Law
Question 3. ADC software, sequence length of 2
Question 4. SCI software, baud rate, input, output, busy-wait, strings
Question 5. Fixed-point software
Question 6. Carry overflow bit
Question 7. Cycle by cycle execution
Question 8. Draw SCI output wave
Question 9. Choose the best data structure
Question 10. Implement a Moore FSM, output compare interrupts
Question 11. Call by reference, binding
Question 12. Word bank

Fall 2008 Final (be able to write in both C and assembly)

Question 1. Nyquist Theorem
Question 2. Stepper motor rotational speed
Question 3. ADC software
Question 4. Fixed-point, dropout
Question 5. Tricky stack question
Question 6. Sequence of events during an interrupt
Question 7. OC interrupt software
Question 8. Draw SCI output wave
Question 9. SCI synchronization, communication delay
Question 10. DAC hardware, V=IR
Question 11. Instruction execution
Question 12. LED interface
Question 13. Design a Moore FSM, debounce
Question 14. Call by reference, binding
Question 15. Fifo implementation with index

Fall 2007 Final (be able to write in both C and assembly)
Question 1. ADC precision
Question 2. ADC output
Question 3. Carry bit
Question 4. Overflow bit
Question 5. Bit-level I/O software
Question 6. RAM is volatile
Question 7. Sequence of events during an interrupt
Question 8. Output compare interrupt period
Question 9. Machine code
Question 10. Stack rules
Question 11. Overflow and dropout
Question 12. Cycle by cycle
Question 13. LED interface
Question 14. Switch interface
Question 15. Software program with local variable
Question 16. ADC sampling software
Question 17. SCI input/output
Question 18. FSM implementation

Fall 2006 Final (be able to write in both C and assembly)
Skip question 9
Question 1. ADC precision
Question 2. ADC output
Question 3. Carry bit
Question 4. Overflow bit
Question 5. Fixed-point
Question 6. What action will set the RDRF bit
Question 7. What action will set the TDRE bit
Question 8. What action starts ADC
Question 10. Overflow and dropout
Question 11. Machine code
Question 12. Cycle by cycle
Question 13. LED interface
Question 14. Switch interface
Question 15. Software program with local variable
Question 16. FSM implementation
Question 17. Interrupt program (you do it with output compare interrupts)

Fall 2005 Final (be able to write in both C and assembly)
Skip question 1,2,14
Question 3. What is drop out?
Question 4. Which direction does data flow on the data bus during a write cycle?
Question 5. What action will set the RDRF bit
Question 6. What action that will clear the RDRF bit

Question 7. What event triggers the start of an ADC conversion on the 6812?
Question 8. How do you initialize a global variable on the 9S12C32?
Question 9. What is the precision in decimal digits?
Question 10. What will be the value of the carry (C) bit after subb
Question 11. What will be the value of the overflow (V) bit after subb?
Question 12. How does emul work
Question 13. A tricky stack question
Question 15. Show the machine code generated by the instruction `orab -5,y`
Questions 16 and 17 local variable and parameter binding, SP index
Question 18. What is the SCI bandwidth in bits/sec?
Question 19. After the ADC is triggered, out of which register does the software read the ADC result?
Question 20. A signed 8-bit binary fixed-point number has a resolution of 1/16 = 2-4. If the integer value stored in memory is $F0, what value does it represent?
Question 21. Which term best describes an interfacing method that the software checks the status of an I/O device, and proceeds once the device is ready?
Question 22. Which data structure has the following features? It can hold a variable number of fixed-size elements. It has two main operations, one to store data into itself, and a second operation to remove data. The data is removed in a “first come first served” order.
Question 23. Does this operation potentially cause overflow?
Question 24&25. Stack frame binding using Reg Y
Question 26. Sketch the output waveform occurring on the PS1=TxD output as one character is transmitted. Question 27. profiling.
Question 28. Conditional branch
Skip question 29
Question 30. Write an assembly main program that implements this Mealy finite state machine.

Fall 2004 final (be able to write in both C and assembly)

Question 1. The measurement system range is 0 to 19.9 and a resolution of 0.1. What is the precision in decimal digits?
Question 2. Multiple choice, definitions
Part a) Which direction does data flow on the data bus during a read cycle?
Part b) Which term best describes a computer system with a response time to external events that is short and bounded?
Part c) Which data structure has the following features? It can hold a variable number of fixed-size elements. It has two main operations, one to store data into itself, and a second operation to remove data. The data is removed in a “first come first served” order.
Part d) What is the difference between busy-waiting and gadfly synchronization?
Part e) What is **drop out**?
Question 3. Result, V-bit C-bit after an add.
Question 4. Binding with SP indexing
Question 5. Binding with Register X stack frame.
Question 6. A signed 16-bit decimal fixed point number system has a Δ resolution of 1/100. What is the corresponding value of the number if the integer part stored in memory is 1234?
Question 7. tricky stack question
Question 8. conditional branch

Question 9. Give the simplified memory cycles produced
Skip question 10
Question 11. Write a subroutine that inputs one ASCII character from the serial port.
Question 12. Hand assemble the program shown
Question 13. implement the following two-input two-output Mealy finite state machine.