Texas Instruments invites UT undergraduates to apply for summer internships. For more information contact see http://www.careers.ti.com/

1) The first part of the final is closed book. Definitions and simple concepts. No calculator, no crib sheet
2) The second part of the final is open notes, open book (you may use papers, text book, pencil, pen and eraser). You cannot use any electronic devices. You will not be allowed to share books with other students. You are allowed to write in your textbook and to place tabs on the pages to help you find things quickly (e.g., SSI timing, ADC registers, ADC parameters, fixed point definitions, UART, Systick, and Timer registers and examples, FSM examples, motor interfaces, timing equations/diagrams.
3) The format of the final exam will be as follows
 a) multiple choice questions over fundamental topics (closed book).
 b) questions about basic concepts of interrupts, programming, Nyquist theorem, fixed-point (closed book)
 c) 1 to 3 line simple C code. E.g.,
 make a pin an input, set an output high,
 arm for interrupt, acknowledge an interrupt, enable for interrupt,
 sample the ADC,
 set the transmission mode, receive data from the SSI, transmit data out the SSI,
 set the baud rate, read data from the UART, transmit data out the UART.
d) hardware interface of LED, solenoid, solid state relay, EM relay, DC motor, or stepper motor.
e) I/O interface design, implementation, or modification
f) timing analysis, or timing diagram
g) interrupt-driven device driver, interrupt-driven FSM
You will not be allowed to call any Tivaware/Stellarisware functions. You can use the standard I/O port definitions as found in the book and tm4c123gh6pm.h. I expect you to be able to initialize and use
 parallel ports (all of Table 4.2)
 Systick interrupts,
 edge-triggered interrupts,
 periodic timer interrupts,
 period measurement and pulse width measurement with input capture
 SSI busy-wait,
 ADC software start, busy-wait,
 ADC timer triggering.

Since the exam is closed book without calculator, you will need to know some basic constants:

2**0=1=0x01	2**6=64=0x40	2**12=4096=0x1000
2**1=2=0x02	2**7=128=0x80	2**13=8192=0x2000
2**2=4=0x04	2**8=256=0x0100	2**14=16384=0x4000
2**3=8=0x08	2**9=512=0x0200	2**15=32768=0x8000
2**4=16=0x10	2**10=1024=0x0400	
2**5=32=0x20	2**11=2048=0x0800	2**16=65536

Range of integers for 8-bit signed (-128 to +127), 8-bit unsigned (0 to 255)
Range of integers for 16-bit signed (-32768 to +32767), 16-bit unsigned (0 to 65535)

Fixed-point numbers (decimal/binary, signed/unsigned, 8/16 bit)
1) Convert value to integer. E.g., What integer is stored in the computer, when the value 2.1 is stored in 16-bit unsigned binary fixed-point, with a resolution of 2**-10? Answer: I = 2.1*1024, which is about 2150. I will make the math so easy a calculator will not be needed.
2) Convert integer to value. E.g., What is the value of an 8-bit signed decimal fixed-point number (resolution is 0.1) if the integer stored in memory is –123? Answer: –123*0.1 equals –12.3.
3) Basic concepts of range, resolution and rounding.
4) Given the range and resolution, choose the format

Programming (C on the Arm Cortex M) Techniques
1) How to check for overflow when performing integer calculations in C. Answer: promote to higher precision and check intermediate results.
2) Stack picture: function call, function parameters, locals, interrupt service stack frame
3) Memory allocation: global, local, constants, and reset/interrupt vectors
4) Modularity linking call graph to **`#include 'file.h'`**, private versus public
5) **`const volatile static`**
6) Data flow graph
7) Debugging: dumps, monitors, scans, breaks, filter, profiling
8) Putting prototypes to public functions in the **`'file.h'`**, implementations in the **`'file.c'`**

I/O concepts (on the TM4C123): focus on concepts and don't memorize details

1) Direction register. Friendly means setting only the bits that are needed, leaving the rest unchanged.
2) Concept of bit-specific access to GPIO ports. Not details, but the concepts of friendly and critical sections
2) Masking input bits to check for individual signals, making individual output bits high, low, or toggle.
3) Using SysTick to measure elapsed time, and to create a time delay
4) Periodic timer interrupts: rate, and vector
5) UART input/output (busy-wait and interrupt synchronization)
6) Concepts of edge-triggered interrupts
7) Input capture to measure period or pulse width

I/O programming (on the TM4C123)

1) FIFO implementation
2) FSM structures, Moore and Mealy FSM
3) DC motor control running the machine using periodic interrupts
4) Debugging dumps, profiles
5) Mailbox, FIFO, and semaphore

Interfacing

1) Switches (debouncing) and LEDs
2) DC motors (model for motor coil, back emf, jerk)
3) Speaker (where software creates a squarewave)
4) Solid state relay (just like a LED)
5) Stepper motor, solenoid or electromagnetic relay (just like the coil of a DC motor)
6) Simple circuit model for BJT in on/off applications

Definitions (match definition with the following terms)

Real-time, friendly, latency, ROM, RAM, computer, CPU, Harvard architecture, von Neumann architecture processor, ALU, BIU, CU, registers,-embedded computer, microprocessor, microcomputer, microcontroller, I_{OL}, I_{OH}, I_{IL}, I_{IH}, V_{OL}, V_{OH}, V_{IH}, V_{IL}, capacitive loading, nonvolatile, open collector, tristate, memory-mapped I/O, dynamic efficiency, static efficiency, functional debugging, performance debugging, non-intrusiveness, profile, desk check, instrument, stabilize, scan, break, thread, busy waiting, atomic, **critical section, reentrant**, interrupt vector, interrupt acknowledge, interrupt arm, interrupt enable, board support package, functional abstraction, complexity abstraction, cohesion, coupling, latency.

Variables

RAM versus ROM
Public scope versus private scope
Temporary versus permanent allocation
Precision

Noise

PMF, Central Limit Theorem

Communication

General: energy, Shannon Channel Capacity Theorem, duplex, guided, unguided, diameter
Wifi: Sockets, UDP/TCP, DNS, client/server, stack
BLE: How is BLE low energy, profile/service/characteristic, stack, advertise/scan

Old Quizzes and Exams (wherever you see RTI interrupts, replace with Systick interrupts) (wherever you see output compare, replace with periodic timer)

Definitions

Spring 2003 Final, Question 12, desk checking
Spring 2003 Final, Question 14, interface latency
Spring 2003 Final, Question 15, polled interrupt
Spring 2003 Final, Question 16, private
Spring 2003 Final, Question 17, intrusive
Spring 2003 Quiz 1, Question 12, scanpoint
Spring 2003 Quiz 1, Question 13, stabilization
Spring 2003 Quiz 1, Question 14, profile
Spring 2003 Quiz 1, Question 15, busy-waiting
Spring 2003 Quiz 1, Question 1-5, Where in memory are variables allocated?
Spring 2003 Quiz 1, Question 16, real time system
Spring 2003 Quiz 1, Question 17, interrupt acknowledge
Spring 2003 Quiz 1, Question 18, ALU
Spring 2003 Quiz 1, Question 19, volatile
Spring 2003 Quiz 1, Question 20, output low current
Spring 2003 Quiz 1, Question 21, nonintrusive
Spring 2003 Quiz 1, Question 22, friendly, set direction register, toggle output
Spring 2003 Quiz 1, Question 6, Data flow graph
Spring 2004 Quiz 1, Question 1, Which variables are stored on the stack?
Spring 2004 Quiz 1, Questions 6,9,12, volatile, open collector, busy-waiting
Spring 2004 Quiz 1, Questions 8,10,13, nonintrusive, stabilization, real time
Spring 2004 Quiz 2, Question 3, FIFO principles
Spring 2005 Final, Question 10, Fixed point
Spring 2005 Final, Question 16, Desk check
Spring 2005 Final, Question 17, Bandwidth
Spring 2005 Final, Question 18, Latency
Spring 2005 Final, Question 19, Polled interrupt
Spring 2005 Final, Question 2, Critical section
Spring 2005 Final, Question 20, Private

Spring 2005 Final, Question 21, Nonintrusive
Spring 2005 Final, Question 22, Buffered I/O
Spring 2005 Final, Question 24, Associative principle
Spring 2005 Final, Question 3, What does **`short`** mean?
Spring 2005 Final, Question 4, What does **`const`** mean?
Spring 2005 Final, Question 5, What does **`static`** mean?
Spring 2005 Final, Question 7, Debugging instruments
Spring 2005 Quiz 1, Question 2, Intrusive
Spring 2005 Quiz 1, Question 3, Open collector
Spring 2005 Quiz 1, Question 4, Stabilizing
Spring 2005 Quiz 1, Question 5, Tristate logic
Spring 2005 Quiz 1, Question 6, Latency
Spring 2006 Quiz 1, Question 1. I/O bound
Spring 2006 Quiz 1, Question 2. vectored interrupt
Spring 2006 Quiz 1, Question 3. intrusive
Spring 2006 Quiz 1, Question 4. stabilization
Spring 2006 Quiz 1, Question 5. tristate
Spring 2006 Quiz 1, Question 6. latency
Spring 2006 Quiz 1, Question 7. buffered I/O
Spring 2007 Quiz 1, Questions 1-5, Intrusive
Fall 2007 Quiz 1, Question 1, Intrusive
Fall 2007 Quiz 1, Question 2, Debugging instrument, which interrupt occurs first?
Spring 2007 Quiz 1, Questions 8-12, Where are variables allocated?
Fall 2007 Quiz 1, Question 4, Interface latency
Spring 2008 Quiz 1, Question 2, Intrusive
Spring 2008 Quiz 1, Question 6, Where are local variables allocated?
Spring 2008 Quiz 1, Question 8, V_{IH}, V_{IL}
Fall 2008 Quiz 1, Question 6, Where are variables allocated?.
Fall 2009 Quiz 1, Question 1, latency
Fall 2009 Quiz 1, Question 2, reentrant
Fall 2009, Question 9, Timing analysis
Fall 2010 Quiz 1, Intrusive debugging, looking at assembly code
Fall 2010, Question 4, FIFO analysis
Fall 2010, Question 20, FIFO implementation
Fall 2011 Final, Question 13 overflow
Fall 2011 Final, Question 10 GIT
Fall 2011, Question 9, FIFO analysis
Fall 2012, Question 9, What does **`volatile`** mean?
Fall 2015, Q1 Central Limit Theorem and PMF
Fall 2015, Question 7, volatile
Fall 2016, Questions 1-8, definitions

Interrupts RTI->SysTick, output compare->Timer0A, SCI->UART, SPI->SSI

Quiz 1 Spring 2001, Question 1, C programming
Spring 2002 Quiz 1, Question 3, Interrupt stack
Spring 2003 Final, Question 3, What happens if the ISR doesn't acknowledge?
Spring 2003 Final, Question 11,14,18, Definitions
Spring 2003 Final, Question 19d,19e, Acknowledge interrupt
Spring 2003 Final, Question 3, What happens if a ISR does not acknowledge?
Spring 2004 Quiz 2, Question 1, Interrupt programming
Spring 2004 Quiz 2, Question 3, I/O bound versus CPU bound
Spring 2004 Final, Question 2, When does an interrupt occur?
Spring 2004 Final, Question 3, What happens if the ISR doesn't acknowledge?
Spring 2004 Final, Question 17, Interrupt stack
Spring 2004 Final, Question 22, I/O bound versus CPU bound
Spring 2005 Final, Question 1, Items on the stack during the execution of an interrupt service routine

Spring 2005 Final, Question 9, What three conditions cause a RTI interrupt?
Spring 2007 Quiz 1, Question 7, Critical section
Spring 2007 Quiz 1, Question 13, What causes an output compare interrupt?
Fall 2007 Quiz 1, Question 3, Debugging instrument, critical section
Fall 2007 Quiz 1, Question 9, Output compare software
Spring 2008 Quiz 1, Question 5, How do we make a system real time?
Fall 2008 Quiz 1, Question 2, Critical section.
Fall 2010 Quiz 1, Question 2, Write code to see which interrupt occurs first
Fall 2011, Question 2, blind vs busy-wait, vs interrupts
Fall 2011, Question 5, critical section, parameter passing, variable allocation
Fall 2012, Question 2, interrupts and debugging
Fall 2012, Question 15, Timer0B interrupts and debugging
Fall 2015, Question 6, Data flow and FIFO queue analysis
Fall 2015, Question 7, critical section, interrupt stack
Fall 2016, Question 10, critical section
Fall 2016, Question 13, Data flow and FIFO queue analysis

Fixed-Point

Spring 2003 Final, Question 6, Fixed-point calculation
Spring 2003 Quiz 1, Question 7-10, Fixed point numbers
Spring 2004 Quiz 1, Questions 2,3,5, Fixed-point numbers
Final Spring 2004, Question 7 Fixed-point calculation
Spring 2005 Quiz 1, Questions 1,7, Fixed point numbers
Spring 2005 Final, Question 30, Fixed point multiply
Spring 2006 Quiz 1, Question 13, How to store the integer part of a fixed-point number
Spring 2007 Quiz 1, Question 6, How to store the integer part of a fixed-point number
Fall 2007 Quiz 1, Question 5, Choosing fixed-point format
Spring 2008 Quiz 1, Question 3, Choosing fixed-point format
Spring 2008 Quiz 1, Question 4, Writing fixed-point math software
Fall 2008 Quiz 1, Question 4, Fixed-point math, implemented in C.
Fall 2008 Quiz 2, Question 1, Table look up and interpolation
Fall 2008 Final, Question 3, Fixed-point calculation
Fall 2009 Quiz 1, Question 4, binary fixed point.
Fall 2009 Quiz 2, Question 1, Precision of various calculations
Fall 2010 Quiz 1, Question 5, Fixed-point math, implemented in C
Fall 2010, Question 6, fixed point
Fall 2010 Quiz 2, Question 5, Fixed-point design, implementation
Fall 2011, Question 8, Fixed-point math, implemented in C
Fall 2012, Question 1, Fixed-point math, implemented in C
Fall 2015, Question 12, Fixed-point math, implemented in C
Fall 2016, Question 12, Fixed-point math, implemented in C

FSM

Spring 2003 Final, Question 20, FSM controller
Spring 2003 Quiz 2, Question 2, FSM controller
Spring 2004 Final, Question 11, FSM controller
Spring 2004 Quiz 2, Question 6, FSM controller
Spring 2005 Final, Question 25, FSM analysis
Spring 2005 Quiz 2, Question 7, Output compare-driven FSM
Spring 2006 Quiz 2, Question 7, Output compare-driven FSM
Spring 2007 Quiz 1, Question 16, Output compare-driven FSM
Fall 2007 Quiz 1, Question 8, Drawing a FSM state graph
Fall 2008 Quiz 1, Question 5, Interpret a FSM (what is the output given the input)
Fall 2008 Quiz 1, Question 7, Drawing a FSM state graph, implement using interrupts
Spring 2008 Quiz 1, Question 9, Output compare-driven FSM
Fall 2009 Quiz 1, Question 5, Drawing a FSM state graph.

Fall 2010 Quiz 1, Question 3, Drawing a FSM state graph
Fall 2011, Question 15, FSM controller
Fall 2012, Question 10, FSM controller
Fall 2016, Question 16, FSM controller

Hardware interfacing

Spring 2003 Final, Question 5, Solid state relay
Spring 2003 Quiz 2, Question 3, LED interface
Spring 2003 Quiz 2, Question 4, Motor interface
Spring 2004 Final, Question 4, Solid state relay
Spring 2004 Quiz 2, Question 2 Solenoid interface
Spring 2004 Quiz 2, Question 4, LED interface
Spring 2005 Final, Question 29, relay interface
Spring 2005 Quiz 2, Question 2, Solenoid interface
Spring 2005 Quiz 2, Question 4, LED interface
Spring 2005 Quiz 2, Question 6, Stepper motor fundamentals
Spring 2006 Quiz 1, Question 11, LED interface
Spring 2006 Quiz 1, Question 14, Switch debouncing using interrupts
Spring 2007 Quiz 1, Question 14, LED interface
Spring 2007 Quiz 1, Question 15, Stepper interface
Fall 2007 Quiz 1, Question 6, LED interface
Fall 2007 Quiz 1, Question 7, Solenoid interface
Spring 2008 Quiz 1, Question 7, LED interface
Fall 2008 Quiz 1, Question 7, electromagnetic relay interface
Fall 2008 Quiz 1, Question 3, LED interface.
Fall 2008 Final, Question 5, RC time constant of signals on a cable
Fall 2009 Quiz 1, Question 3, V_{OL}, V_{IL}, V_{OH}, V_{IH} interfacing.
Fall 2009 Quiz 1, Question 6, stepper motor jerk.
Fall 2009 Quiz 1, Question 6, electromagnetic relay interface.
Fall 2010 Quiz 1, Question 6, DC motor interface
Fall 2010, Question 5, RC analysis
Fall 2010, Question 7, current
Fall 2009, Question 11, Effect of capacitance
Fall 2009, Question 16, LED interface
Fall 2010, Question 16, Motor interface
Fall 2011, Question 4, Effect of capacitance
Fall 2011, Question 11, RC circuit
Fall 2011, Question 14, DC motor interface circuit using TIP31
Fall 2011, Question 11, RC circuit and Laplace domain analysis
Fall 2012, Question 12, DC motor interface circuit using TIP142, reading a data sheet
Fall 2015, Question 8, Step response to RC circuit
Fall 2015, Question 13, Motor circuit analysis
Fall 2016, Question 9, Step response to RC circuit
Fall 2016, Question 14, LED interface, interrupts, dimming with PWM

UART

Fall 2009, Question 10, UART protocol
Fall 2010, Question 7, UART protocol

SSI (SPI) interfacing

Spring 2001 Final, Question 3, Two 6812 SPI interface
EE345M Spring 2000, Final, Q1 (b-e), SPI interface
EE345M Spring 2001, Final, Q3, SPI interface
Final Spring 2004, Question 6 SPI interface
Final Spring 2004, Question 16, Master/slave interface
Spring 2005 Quiz 2, Question 5, SPI interface

Spring 2005 Final, Question 12, DAC/SPI interface
Spring 2006 Quiz 2, Question 1, SPI interface to a shift register
Spring 2006 Quiz 2, Question 5, SPI interface
Spring 2007 Quiz 2, Question 2, SPI waveforms
Fall 2007 Quiz 2, Question 2, 74HC165 interface
Spring 2008 Quiz 2, Question 3, SPI timing and interface
Fall 2008 Quiz 2, Question 2, Virtual SPI using Port T
Fall 2008 Final, Question 4, SPI timing and interface
Fall 2009 Quiz 2, Question 5, SPI modes
Fall 2010, Question 17, SPI communication
Fall 2015 Final, Question 14, SPI timing diagram (setup and hold)

Analog interfacing

Spring 2002 Final, Question 5, ADC resolution
Spring 2002 Final, Question 6 Nyquist theorem
Spring 2003 Final, Question 4, ADC resolution
Spring 2004 Quiz 2, Question 5 Nyquist theorem
Final Spring 2004, Question 5, ADC resolution
Final Spring 2004, Question 10, ADC operation, where are the results?
Spring 2005 Quiz 2, Question 1, ADC resolution
Spring 2005 Final, Question 11, ADC resolution
Spring 2006 Quiz 2, Question 4, ADC resolution
Spring 2006 Final, Question 10, ADC precision
Spring 2006 Final, Question 11, 9S12 ADC initialization
Spring 2007 Quiz 2, Question 4, Data acquisition design
Fall 2007 Quiz 2, Question 4, ADC setup, and data acquisition ISR
Fall 2007 Quiz 2, Question 5, Analog amplifier
Spring 2008 Quiz 2, Question 4, Data acquisition design
Fall 2008 Quiz 2, Question 4, Data acquisition design
Fall 2009 Quiz 2, Question 6, Analog amplifier
Fall 2009, Question 15, Analog amplifier
Fall 2010 Quiz 2, Question 7, Analog amplifier
Fall 2010, Question 10, DAC parameters
Fall 2010, Question 15, Noise and SNR
Fall 2010, Question 19, analog filter
Fall **2011 Final, Question 12**, ADC parameters
Fall 2011, Question 16, Instrumentation amp and bridge
Fall 2012, Question 8, Nyquist theorem
Fall 2012, Question 16, Op amp circuit
Fall 2015, Question 5, ADC parameters
Fall 2015, Question 9, ADC conversion techniques
Fall 2015, Question 11, Op amp circuit
Fall 2016, Question 11, DAC types
Fall 2016, Question 15, threshold detector

Timer interfacing

Fall 2015, Question 10, Pulse width measurement

Systems and interfacing

Fall 2009 Quiz 2, Question 2, Capacitors
Fall 2009 Quiz 2, Question 3, C programming
Fall 2009 Quiz 2, Question 4, Parallel port expander
Fall 2010 Quiz 2, Question 1, Parallel I/O macro
Fall 2010 Quiz 2, Question 2, Pi filter
Fall 2010 Quiz 2, Question 4, Solve it with edge-triggered interrupts
Fall 2010 Quiz 2, Question 6, Capacitors, what does 123 printed on it mean?

Fall 2010, Question 13, Power Budget
Fall 2010, Question 14, Capacitors
Fall 2011, Question 3, what is VDO?
Fall 2012, Question 4, Capacitors
Fall 2015, Question 3, Capacitors
Fall 2016, Question 17, Debugging MACRO

Communications

Fall 2009, Question 1, Energy
Fall 2009, Question 13, Shannon Channel Capacity
Fall 2010, Question 1, Energy
Fall 2011, Question 5, Energy
Fall 2011, Question 11, Shannon Channel Capacity
Fall 2012, Question 5, Diameter
Fall 2012, Question 6, Shannon Channel Capacity
Fall 2012, Question 7, transmission line theory, fields and waves
Fall 2015, Question 2, NRZ and transmitting energy

Ethics

Final Spring 2006, Question 13, IEEE code
Final Spring 2007, Question 20, IEEE code
Spring 2008 Final, Question 9, Reasons why engineers cheat
Fall 2008 Final, Question 6, Discover a fault
Fall 2009 Final, Question 12, IEEE code
Fall 2010, Question 11
Fall 2011, Question 1, IEEE code
Fall 2011, Question 3, IEEE code
Fall 2015, Question 4, IEEE code