

Jonathan W. Valvano

First: \_\_\_\_\_ Last: \_\_\_\_\_

November 18, 2016, 10:00-10:50am. Open book, open notes, calculator (no laptops, phones, devices with screens larger than a TI-89 calculator, devices with wireless communication). You have 50 minutes, so please allocate your time accordingly. ***Please read the entire quiz before starting.***

**(10) Question 1.** Design a circuit that takes a 3.7V Li-Ion battery and produces a 5V power supply for an embedded system. In sleep mode, the system needs 0.05 mA of current. In active mode, the system requires 200 mA of current. Show all part numbers, but do not worry about exact values for the resistors, capacitors and inductors. It runs in sleep mode 99% of the time and runs in active mode 1% of the time.

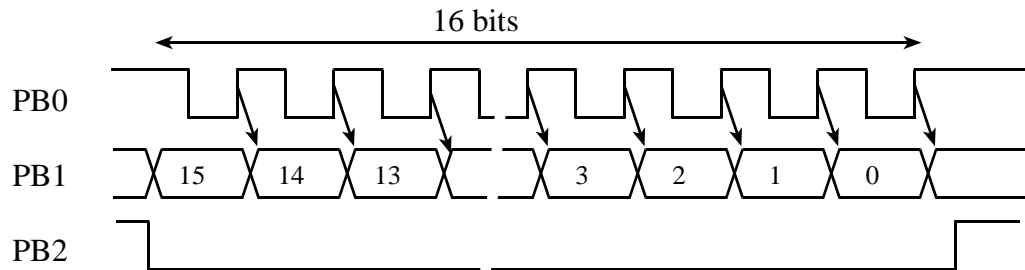
**(10) Question 2.** We have a temperature data acquisition system like Lab 9. Let  $n$  be the 12-bit ADC result, and assume the system is linear. The measured temperature calculated with a linear equation  $\mathbf{m} * n + \mathbf{b}$ , where  $\mathbf{m}$  and  $\mathbf{b}$  are calibration coefficients. Consider these factors that might affect performance.

- A) Varying the TM4C123 bus frequency from 40 MHz to 80 MHz using the PLL
- B) Switching from the 12-bit ADC on the TM4C123 to the 14-bit ADC on the MSP432
- C) Electromagnetic field noise pickup
- D) Calibration drift, meaning slope  $\mathbf{m}$  and offset  $\mathbf{b}$  drift over time.
- E) Varying the ADC sampling rate from 100 Hz to 10 kHz
- F) Using floating point versus fixed point in the linear equation  $\mathbf{m} * n + \mathbf{b}$
- G) Thermal noise in the analog electronics

**(5) Part a)** Which of the above factors affect **accuracy**? Choose all factors that affect temperature measurement accuracy. Place the appropriate letters in the box.

**(5) Part b)** Which of the above factors affect **resolution**? Choose all factors that affect temperature resolution. Place the appropriate letters in the box.

**(20) Question 3.** The goal is to receive synchronous serial data using regular input pins (not the built-in SSI module). The PB2-PB0 pins do not constitute a regular SSI port; you will solve this interface with three GPIO input ports. The 16-bit data follows this protocol. All three signals are outputs from the external device and inputs to your TM4C123.



You must use busy-wait synchronization and not interrupts. You may use any of the standard definitions in the `tm4c123gh6pm.h` file. In addition, you may use these bit-specific definitions

```
#define PB0  (*(volatile uint32_t *)0x40005004)
#define PB1  (*(volatile uint32_t *)0x40005008)
#define PB2  (*(volatile uint32_t *)0x40005010)
```

Assume the GPIO Port B has already been initialized so PB2, PB1, and PB0 are simple GPIO inputs. Show the function that inputs one 16-bit data using the above SSI protocol. You should first wait for the chip select on PB2, and then read 16 bits from PB1 using the clock on PB0 as synchronization. You may assume the software execution is fast compared to the clock period. You may assume the device always clears PB2 low, sends exactly 16 bits, and then sets PB2 high. Use comments to explain your approach.

```
uint16_t SSI_Input(void){
```

(15) **Question 4.** This software samples the analog input on PD3 at the rate specified by the period parameter. This is Program 8.8 in the book. Modify this software so it samples the input on **PE5**, which is analog channel 8. Cross out parts of the code you wish to delete and insert necessary additions.

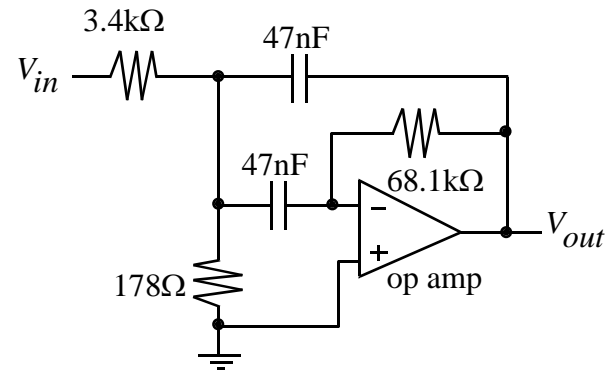
```

void ADC0_InitTimer0ATriggerSeq3PD3(uint32_t period){
    volatile uint32_t delay;
    SYSCTL_RCGCADC_R |= 0x01;
    SYSCTL_RCGCGPIO_R |= 0x08;
    delay = SYSCTL_RCGCGPIO_R;
    GPIO_PORTD_DIR_R &= ~0x08;
    GPIO_PORTD_AFSEL_R |= 0x08;
    GPIO_PORTD_DEN_R &= ~0x08;
    GPIO_PORTD_AMSEL_R |= 0x08;
    ADC0_PC_R = 0x01;
    ADC0_SSPRI_R = 0x3210;
    SYSCTL_RCGCTIMER_R |= 0x01;
    delay = SYSCTL_RCGCGPIO_R;
    TIMER0_CTL_R = 0x00000000;
    TIMER0_CTL_R |= 0x00000020;
    TIMER0_CFG_R = 0;
    TIMER0_TAMR_R = 0x00000002;
    TIMER0_TAPR_R = 0;
    TIMER0_TAILR_R = period-1;
    TIMER0_IMR_R = 0x00000000;
    TIMER0_CTL_R |= 0x00000001;
    ADC0_ACTSS_R &= ~0x08;
    ADC0_EMUX_R = (ADC0_EMUX_R&0xFFFF0FFF)+0x5000;
    ADC0_SSMUX3_R = 4;
    ADC0_SSCTL3_R = 0x06;
    ADC0_IM_R |= 0x08;
    ADC0_ACTSS_R |= 0x08;
    NVIC_PRI4_R = (NVIC_PRI4_R&0xFFFF00FF) | 0x00004000;
    NVIC_EN0_R = 1<<17;
    EnableInterrupts();
}
void ADC0Seq3_Handler(void){
    ADC0_ISC_R = 0x08;
    Fifo_Put(ADC0_SSFIFO3_R);
}

```

**(10) Question 5.** Consider the following analog filter with two 47-nF capacitors. This is a regular op amp powered with +12 and -12 V. Assume the input is  $V_{in} = 1V \cos(2\pi ft)$ . This is some sort of analog filter. It does pass signals at 1 kHz, with a gain of 10,  $1/(2\pi * 3.4k * 47nF) = 1$  kHz. Essentially, this question is asking you to classify this filter as a low pass, high pass, band pass or band reject filter.

**(5) Part a)** What will be the approximate output of this circuit for input frequencies much much smaller than 1 kHz? For example, what is the output if the input is DC? Show your work.



**(5) Part b)** What will be the approximate output of this circuit for input frequencies much much larger than 1 kHz? Show your work.

**(15) Question 6.** Assume you have a data acquisition system that measures voltage. There is a calibration procedure that populates the following static variables. Shown is an example calibration, but the system allows for dynamic calibration, so these values may change.

```
static int32_t V1=1000; // first voltage in 0.001 V units
static int32_t V2=3000; // second voltage in 0.001 V units
static int32_t N1=1274; // 12-bit ADC value corresponding to V1
static int32_t N2=3723; // 12-bit ADC value corresponding to V2
```

Write the C function that converts a 12-bit ADC measurement (0 to 4095) into the corresponding voltage in 0.001V units using a linear fit between ADC and voltage. Floating point is not allowed. The input can vary from 0 to 4095, so this is both interpolation and extrapolation. Make it as accurate as possible without causing overflow or dropout during an intermediate calculation.

*Test case: n=662 should yield a result of 500.*

```
int32_t Convert(int32_t n){
```

(15) Question 7. The embedded system has three modules:



Your job is to design the analog circuit that implements  $V_{out} = 10 \cdot V_{in} + 1.65V$ . The input voltage varies from -0.15 to +0.15V, so the output voltage will vary from 0.15 to 3.15V. The output,  $V_{out}$ , is connected to the microcontroller ADC. R1 and R2 are already chosen such that the analog reference is 1.65V. You may use any chips shown in the book or presented in class. Show your work and label all chip numbers and resistor values. You do not have to show pin numbers.

