This is the closed book section. You must put your answers in the boxes. When you are done, you turn in the closed-book part and can start the open book part.

(4) Question 1. Showing a plot of a PMF, give an example of how the Central Limit Theorem applies to embedded systems.

(4) Question 2. What is the advantage of a +6V/-6V NRZ communication protocol over simple 3.3V/0V digital encoding?

A) no advantage
B) less EMI emissions
C) it is differential
D) both high and low use energy so it has a larger diameter
E) it can drive less current
F) faster because the capacitance is less

(5) Question 3. Consider the differences between tantalum and ceramic capacitors. Pick the answer that best differentiates the two capacitor types. Place a T for tantalum, a C for ceramic, a B for both, or an N for neither.

A) Which capacitor is nonpolarized? -----------------------------------------------
B) Which capacitor has a larger ESR? -----------------------------------------------
C) Which capacitor should we use for precision high-frequency analog filters? ------
D) Which capacitor should we use for precision high-frequency digital filters? ------
E) Which capacitor should we use between 3.3V power and ground? ---------------

(4) Question 4. There are ten points of the IEEE Code of Ethics. Which of the following points is not one of the ten points?

1. to give responsibility consistent with the safety, health and welfare of the public;
2. to avoid real or perceived conflicts of interest whenever possible, and to disclose them;
3. to be honest and realistic in stating claims or estimates based on available data;
4. to reject bribery in all its forms;
5. to improve the understanding of technology, its application, and consequences;
6. to maintain and improve our technical competence;
7. to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors;
8. to treat fairly all persons;
9. to avoid injuring others, their property, reputation, or employment by false or malicious action;
10. to assist colleagues and to support them in following this code of ethics.
(4) Question 5. Consider these ADC performance parameters:

A) linearity          B) accuracy          C) resolution
D) bandwidth          E) monotonicity      F) repeatability
G) precision

Listed here are experimental procedures one might use to measure ADC performance. State the ADC parameter determined by each procedure. There is one best answer. Place one letter A to G into each box.

Part a) The input is slowly changed from minimum to maximum. The input voltage, \( V_i \), that causes a change in digital output is recorded. The average of the differences \( V_{i+1} - V_i \) is calculated.

Part b) The input is slowly changed from minimum to maximum. The input voltage, \( V_i \), that causes a change in digital output is recorded. The number of \( V_i \) recordings is calculated.

Part c) The input is held constant, and the digital output is recorded multiple times. The standard deviation of these recordings is calculated.

Part d) The input is slowly changed from minimum to maximum. The input voltage, \( V_i \), that causes a change in digital output is recorded. A linear regression is performed on the input/output data set. What ADC parameter does the correlation coefficient of this regression represent?

(10) Question 6. Consider an interrupt-driven data flow problem. The arrival of data triggers an input interrupt. The input ISR reads the data, puts them into a FIFO, and arms the output ISR. Reading data acknowledges the input interrupt. The output ISR is triggered when the output device is idle and it is armed. The output ISR gets data from the FIFO and if there are data the output ISR writes the data to the output. Writing data acknowledges the output interrupt. Both ISRs are running at the same priority and the main program is doing unrelated tasks. Initially, the input ISR is armed and the output ISR is disarmed. Arming means the software set bits in the IM register; disarming means clearing bits in the IM register.

(5) Part a) What should you do if the input ISR gets a full error when calling FIFO put?

A) disarm the input ISR  D) increase the size of the FIFO
B) disarm the output ISR  E) decrease the size of the FIFO
C) discard data          F) none of the above

(5) Part b) What should you do if the output device gets an empty error when calling FIFO get?

A) disarm the input ISR  D) increase the size of the FIFO
B) disarm the output ISR  E) decrease the size of the FIFO
C) discard data          F) none of the above

(14) Problem 7. Consider the following Systick interrupting system with its corresponding assembly code generated by the Keil uVision compiler. You may assume Systick interrupts occur every 1 ms. The listing includes absolute addresses. ROM starts at 0x00000000, and RAM starts at 0x20000000. **Count** is a 32-bit variable at address 0x20000000.

```c
volatile uint32_t Counts = 0;

EnableInterrupts:
0x00000324 B662 CPSIE I
0x00000326 4770 BX lr

WaitForInterrupt:
0x00000336 BF30 WFI
0x00000338 4770 BX lr
```
```c
void static Add(uint32_t n) {
    Counts = Counts + n;
}

void SysTick_Handler(void) {
    Add(1);
}

int main(void) {
    Init(); // includes SysTick_Init
    EnableInterrupts();
    while(1) {
        WaitForInterrupt();
        Add(-1);
    }
}
```

### Add

<table>
<thead>
<tr>
<th>Address</th>
<th>Instruction</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x000003C4</td>
<td>4902</td>
<td>LDR r1,[pc,#8] ;@0x000003D0</td>
</tr>
<tr>
<td>0x000003C6</td>
<td>6809</td>
<td>LDR r1,[r1,#0x00]</td>
</tr>
<tr>
<td>0x000003C8</td>
<td>4401</td>
<td>ADD r1,r1,r0</td>
</tr>
<tr>
<td>0x000003CA</td>
<td>4A01</td>
<td>LDR r2,[pc,#4] ;@0x000003D0</td>
</tr>
<tr>
<td>0x000003CC</td>
<td>6011</td>
<td>STR r1,[r2,#0x00]</td>
</tr>
<tr>
<td>0x000003CE</td>
<td>4770</td>
<td>BX lr</td>
</tr>
<tr>
<td>0x000003D0</td>
<td>20000000</td>
<td>DCD 0x20000000</td>
</tr>
</tbody>
</table>

```
SysTick_Handler:
0x000004C4 | B500 | PUSH {lr} |
0x000004C6 | 2001 | MOVS r0,#0x01 |
0x000004C8 | F7FFFF7C | BL Add |
0x000004CC | BD00 | POP {pc} |
```

```
main:
0x00000510 | F7FFFF60 | BL Init |
0x00000514 | F7FFFF06 | BL EnableInterrupts |
0x00000518 | E005 | B 0x00000526 |
0x0000051A | F7FFFF0C | BL WaitForInterrupt |
0x0000051E | F04F30FF | MOV r0,#0xFFFFFFFF |
0x00000522 | F7FFFF4F | BL Add |
0x00000526 | E7F8 | B 0x0000051A |
```

### (4) Part a)
Is there a critical section in the software system shown above?
- **A)** no critical section
- **B)** yes, with LR
- **C)** yes, access to R0
- **D)** yes, access to Counts in main
- **E)** yes, access to Counts in SysTick_Handler
- **F)** yes, access to Counts in Add

### (2) Part b)
What is the numerical value of R2 at the end of executing Add?

### (2) Part c)
What is the low-power feature used in this system?

### (2) Part d)
What does the volatile qualifier for Counts mean?
- **A)** private in scope
- **B)** stored in ROM
- **C)** stored in global RAM
- **D)** the value is fixed and cannot be changed by the function
- **E)** tells the compiler to fetch a new value, and do not optimize
- **F)** promoted to the next high precision

### (2) Part e)
What does the static qualifier for the function Add() mean?
- **A)** function is public in scope
- **B)** function is stored in ROM
- **C)** run with interrupts disabled
- **D)** the parameters are fixed and cannot be changed
- **E)** function is stored in RAM
- **F)** none of the above

### (2) Part f)
How does the return from interrupt instruction POP {pc} change context?
- **A)** gets the PC value from vector table
- **B)** gets the PC value from RAM table
- **C)** moves PC to LR, then pops 8 values
- **D)** pops 0xFFFFFFFF9 off stack, then pops 8 more
- **E)** tries to move LR to PC, then pops 8 values
- **F)** pops the return address off stack into PC
(5) **Question 8.** This problem addresses the issue of capacitive loading on a high-speed serial transmission line like SSI. The SSI port of a TM4C123 is connected via a long cable to a DAC. We will model this cable as a single 1-kΩ resistor in series with a 1-nF capacitor, as shown on the left figure below. Consider a 3.3-V 500-kHz clock from the microcontroller to the DAC. The figure on the right plots the output of the microcontroller, labeled PA2.

![Cable Model Diagram](image)

Assume the SCLK has been low for a long time while the SSI has been idle and the clock begins to oscillate at time 0, as data is being transferred at 500 kHz. Develop an equation for the SCLK input at the DAC input as a function of time for the time-region 0 to 1 μs. Use the equation to make a rough guess (without a calculator) about the voltage of the DAC input at time equals 1 μs.

**Equation:**

**SCLK at 1 μs**

(5) **Question 9.** Consider three different ADC techniques: flash, sigma delta and successive approximation. Pick the ADC technique that best answers each question. Place an **F** for flash, an **SD** for sigma delta, or an **SA** for successive approximation.

A) Which technique is best for high-precision audio sampling? ____________________________

B) Which technique is best for low-precision high-frequency sampling? ________________

C) Which technique is used in the TM4C123? ______________________________________

D) Which technique has a conversion speed linearly related to the number of bits? __________

E) Which technique has a cost exponentially related to the number of bits? _______________
Open book, open notes, calculator (no laptops, phones, devices with screens larger than a TI-89 calculator, devices with wireless communication). You must put your answers on these pages. Please don’t turn in any extra sheets.

(10) Question 10. This software measures the 24-bit period on PB6 from rising edge to rising edge using Timer 0A interrupts. Change the software to use PB4 on Timer 1A. Change it to measure period on the falling edges. Cross out parts of the code you wish to delete and insert necessary additions.

```c
uint32_t Period, First, Done;

void PeriodMeasure_Init(void) {
    SYSCTL_RCGCTIMER_R |= 0x01;
    SYSCTL_RCGCGPIO_R |= 0x02;
    First = 0;   Done = 0;
    GPIO_PORTB_DIR_R &= ~0x40;
    GPIO_PORTB_AFSEL_R |= 0x40;
    GPIO_PORTB_DEN_R |= 0x40;
    GPIO_PORTB_PCTL_R = (GPIO_PORTB_PCTL_R&0xF0FFFFFF)+0x07000000;
    TIMER0_CTL_R &= ~0x00000001;
    TIMER0_CFG_R = 0x00000004;
    TIMER0_TAMR_R = 0x00000007;
    TIMER0_CTL_R &= ~0x0000000C;
    TIMER0_TAILR_R = 0x0000FFFF;
    TIMER0_TAPR_R = 0xFF;
    TIMER0_IMR_R |= 0x00000004;
    TIMER0_ICR_R = 0x00000004;
    TIMER0_CTL_R |= 0x00000001;
    NVIC_EN0_R = 1<<19;
}

void Timer0A_Handler(void) {
    TIMER0_ICR_R = 0x00000004;
    Period = (First - TIMER0_TAR_R)&0x00FFFFFF;
    First = TIMER0_TAR_R;
    Done = 1;
}
```
(15) **Question 11.** Interface this transducer to the ADC. The information is encoded as $V_1$, and it is relative to ground. The transducer output ranges from -0.15 to +0.15V, in other words, $-0.15 \leq V_1 \leq +0.15$. Design the analog circuit to create an ADC input range of 0 to +3V. One of the tricks in creating a linear and high-accuracy system is avoiding the extremes of the analog circuits including the ADC. In this system the interesting transducer range is actually only -0.10 to +0.10 V; therefore the interesting signals at the ADC will range from 0.5 to 2.5 V. Include an antialiasing analog filter ($f_c = 1$kHz). Show all resistors, capacitors, and chip numbers. The available power supply voltage is 3.3V. Assume $R_1$ and $R_2$ are already chosen to achieve a reference of 1.5V.

![Circuit Diagram]

- $V_1$ Transducer
- $0$ to $3$V
- $3.3$V
- $10$ kΩ
- $1.5$V
- LM4041 Adjustable
- $V_2$ ADC
(10) Question 12. Write an integer function in C that calculates \( \text{output} = \frac{1,000,000}{\text{input}} \), where \( \text{input} \) and \( \text{output} \) are signed 32-bit integers. No floating point allowed. You may assume the \( \text{input} \) is not zero, so overflow cannot occur. However, please implement rounding to the closest integer. In particular test your solution with the following four test cases.

- If \( \text{input} \) is +589 then the \( \text{output} \) should be +1698 (close to 1697.79287).
- If \( \text{input} \) is +5 then the \( \text{output} \) should be +200,000 (it should be perfect for all exact cases).
- If \( \text{input} \) is -7 then the \( \text{output} \) should be -142,857 (close to -142,857.142857).
- If \( \text{input} \) is -589 then the \( \text{output} \) should be -1698 (close to -1697.79287).

(5) Question 13. Consider a brushed DC motor. The coil resistance is 10 \( \Omega \) and the coil inductance is 1 \( \mu \text{H} \). Using circuits, equations, and formulas explain the experimental results that a steady state 2 A flowed through the motor when a steady state 10 V was applied across the motor.

(5) Question 14. Consider a simplex synchronous serial interface from master to slave. The master clock is 50% duty cycle 1 MHz \( \text{Clock} \). The master shifts data out on the rising edge of the \( \text{Clock} \). The maximum propagation delay from \( \text{Clock} \) to data output is 200 ns. The slave shifts data in on the falling edge of the \( \text{Clock} \). The slave hold time is 300 ns and the setup time is 100 ns. Complete the timing diagram to scale showing data available and data required timing. Show the transfer of one bit (not the entire frame).