Jonathan W. Valvano First:_____ Last:_____

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

(2) Question 1. Give the parameter	**(2) Question 7c.** Select A-F
(4) Question 2a. Select A-K	**(2) Question 8a.** Baud rate
(2) Question 2b. Select A-F	**(2) Question 8b.** Hexadecimal number
(4) Question 3. Number of bytes	**(4) Question 9a.** Select A-F
(4) Question 4. Select A-F	**(4) Question 9b.** Select A-F
(4) Question 5. Select A-F	**(1) Question 10a.** DAC parameter
(4) Question 6. Select A-F	**(1) Question 10b.** DAC parameter
(2) Question 7a. Select A-F	**(1) Question 10c.** DAC parameter
(2) Question 7b. Select A-F	**(1) Question 10d.** DAC parameter

(4) Question 11. Assume you are a practicing engineer building products for a company that sells the products for profit. You see software posted on the web without instructions on how to use it. What must you do before incorporating that software into one of your products?

(4) Question 12. Give the equation that relates sampling jitter to measurement error.

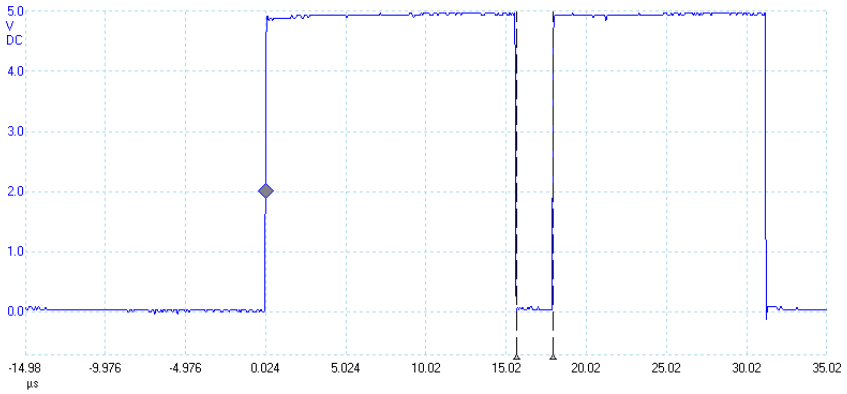
(4) Question 13. Give the equation that defines power budget for a battery-powered embedded system.

(2) Question 14. To what do the terms C0G X7R and Z5U refer?

(2) Question 1. Assume you are asked to make minor adjustments on a communication channel to improve the data transfer rate. In general, more of what electrical parameter must you put into the system to improve bandwidth.

(6) Question 2. The goal of this system is to output a sound at 1000 Hz. The **`Fluke`** buffer has 32 entries per cycle. The interrupt rate should be 32 kHz (31.25μS), where each execution of the ISR outputs one point of the buffer to the DAC. The debugging profile on PP7 seems to indicate interrupt periods are either 2.3 or 13.2 μs, as shown in the following plot. This is actual data measured on an actual 9S12, running this program. The PLL is active (24MHz) and TSCR2 is 5 (divide by 32).

```
void interrupt 15 OC7ISR(void){
  TFLG1 = 0x80;
  PTP ^= 0x80;  // profile
  DAC_Out(Flute[Index]);
  Index = (Index+1)&0x1F;
  TC7 = TC7 + 375;
// (24*1000)/32;
}
void main(void) {
  DAC_Init();
  OC7_Init();
  DDRP |= 0xC0;
asm cli
  for(;;) {
    PTP ^= 0x40; // profile
  }
}
```



(4) Part a) Which of the following changes would you make to fix the bug in the above program?
A) Increase the 375 so the output compare interrupts occur less frequently.
B) Decrease the 375 so the output compare interrupts occur more frequently.
C) Increase the TSCR2 or change the PLL so the TCNT counts slower.
D) Decrease the TSCR2 or change the PLL so the TCNT counts faster.
E) Change **`TFLG1=0x80;`** to **`TFLG1 |= 0x80;`**
F) Change **`TFLG1=0x80;`** to **`TFLG1 &= ~0x80;`**
G) Add **`sei`** to the top of the ISR and **`cli`** to the bottom of the ISR.
H) Remove the line **`PTP ^= 0x40;`**
I) Reduce the number of points in the buffer from 32 to 16, so the interrupts can keep up.
J) Change **`TC7=TC7+375;`** to **`TC7=TCNT+375;`**
K) None of the above changes will remove the bug.

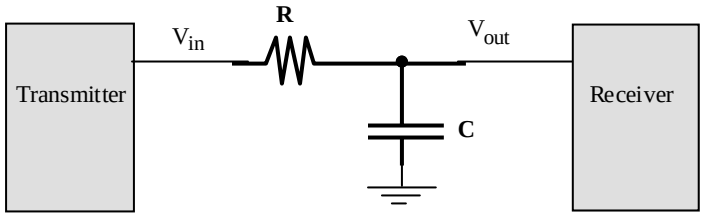
(2) Part b) How would you characterize the debugging profile in the above figure?
A) Nonintrusive **B)** Highly intrusive **C)** Invasive
D) Happy **E)** Minimally intrusive **F)** CPU bound

(4) Question 3. How many bytes are pulled off the stack by the **`rti`** instruction?

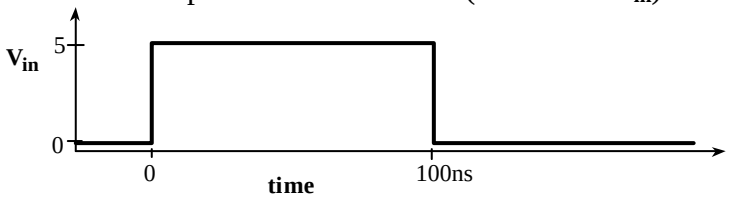
(4) Question 4. Consider the situation in which a FIFO queue is used to buffer data between a main program and an output interrupt service routine. The main program calls **`SCI_OutChar`**, which in turn puts one byte into a FIFO. The SCI ISR gets data from the FIFO and outputs it to the transmitter. The baud rate is 10,000 bits/sec. Experimental measurements show that the average rate at which the FIFO put is called is 2000 times/sec. What does it mean? Choose A-F.

A) The system could work, but the system is CPU bound
B) The system does not work, but could be corrected by increasing FIFO size
C) The system could work, but the system is I/O bound
D) The system does not work, but could be corrected by increasing baud rate
E) The system could work, but the FIFO is not needed and could be replaced by a global variable
F) The system could work, but interrupts are not needed in this system

(4) Question 5. This problem addresses the issue of capacitive loading on a high-speed serial transmission line like SPI. The SPI ports of two 9S12s are connected with a VERY long cable. We will model this cable as a single resistor in series with a capacitor, as shown in the figure below.



Consider a 5-V 100-ns pulse on the output of the transmitter (labeled as $\mathbf{V_{in}}$)



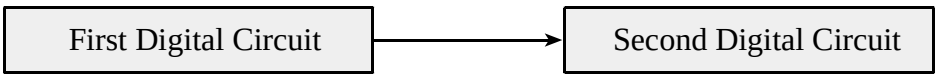
At time just after 100ns, what effect does the capacitor have on the circuit?

A) The dI/dt causes a large voltage spike
B) The capacitor behaves like a short causing $\mathbf{V_{out}}$ to be zero.
C) The stored charge on the capacitor is shorted into the transmitter causing a spark in the circuit
D) The capacitor prevents data from being transferred and the SPI does not work
E) The capacitance is polarized so it only affects the circuit at time = 0
F) The dV/dt at 100ns will be about 5V/(**RC**)

(4) Question 6. Consider a system that needs to represent the numbers from -100.00 to 100.00 with a resolution of 0.01. No human output will be required. I.e., only calculations internal to the computer will be calculated and there are a lot of multiplies, divides, adds and subtracts with these numbers. Which number system would be best on a microcontroller like the 9S12?

A) 8-bit integer
B) Signed 16-bit decimal fixed-point with 0.1 resolution
C) Signed 16-bit decimal fixed-point with 0.01 resolution
D) Signed 32-bit decimal fixed-point with 0.001 resolution
E) Signed 16-bit binary fixed-point with 2^{-8} resolution
F) Floating-point

(6) Question 7. Consider the situation in which the output of one digital circuit is connected to the input of another digital circuit. The grounds are connected. You attach a current meter to the wire between the circuits.



(2) Part a) When the output is high, which direction does current actually flow?

A) Left to right **B)** Right to left
C) There is no current flowing **D)** Sometimes one way, sometimes the other
E) You cannot tell **F)** Current flows in a circle on this wire

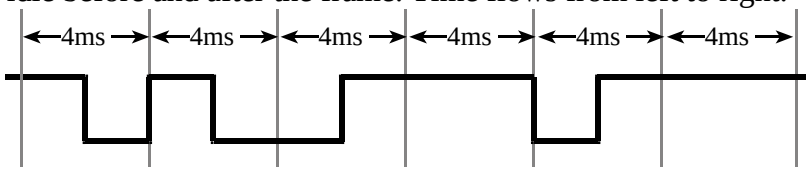
(2) Part b) When the output is low, which direction does current actually flow?

A) Left to right **B)** Right to left
C) There is no current flowing **D)** Sometimes one way, sometimes the other
E) You cannot tell **F)** Current flows in a circle on this wire

(2) Part c) What happens to this digital interface when the effective capacitance to ground is increased?

A) No change **B)** The bandwidth decreases
C) Increase in DC current **D)** The signal to noise ratio improves
E) Decrease in DC current **F)** Causes a potential back emf pulse solved by a snubber diode

(4) Question 8. The following waveform was measured on the PS1 output (voltage levels are +5 and 0) when one or more SCI transmissions occurs. The protocol is 1 start, 8 data and 1 stop bit. You may assume the channel is idle before and after the frame. Time flows from left to right.



(2) Part a) What is the baud rate in bits/sec?

(2) Part b) What data in hexadecimal were transmitted? Give one or two bytes as appropriate.

(8) Question 9. There are three decimal fixed point numbers: **X** = **I***10^{-1}, **Y** = **J***10^{-2}, **Z** = **K***10^{-2}.

(4) Part a) How do you multiply **Z**=**X***__Y__?

A) K = I*J
B) K = (I*J)*10
C) K = (I*J)/100
D) K = (I*J)/10
E) K = (I/10)*J
F) None of these choices is correct

(4) Part b) How do you add **Z**=**X**+**Y**?

A) K = I+J
B) K = (10*I)+J
C) K = I/100+J/10
D) K = (I+J)/10
E) K = I/10+J/100
F) None of these choices is correct

(4) Question 10. First, think of as many DAC performance parameters as you can. Listed here are experimental procedures one might use to measure DAC performance. State the DAC parameter determined by each.

Part a) The input is stepped from minimum to maximum. For each input change, the change in DAC output is measured. The results are processed by averaging all the changes in output.

Part b) The input is stepped from minimum to maximum. For each input change, the change in DAC output is measured. The results are processed by counting the number of changes in output.

Part c) The input is stepped from minimum to maximum. For each input change, the DAC output value is measured. The results are processed by averaging the absolute values of the differences between the measured output and the expected output.

Part d) The input is stepped from minimum to maximum. For each input change, the change in DAC output is measured. The results are processed to see if all the changes in output are positive.

end of closed book section

Jonathan W. Valvano First:_____ Last:_____

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.

(5) Question 15. You are building a thermometer using a linear transducer and linear amplifier. You measure the signal to noise ratio on the analog circuit to be 40 dB. I.e. if the signal is 5V,

$40 = 20 \log_{10}(5V/\textbf{noise})$.

To make the system better, we want more ADC bits, but to make it cheaper we want fewer bits. Considering SNR what is the smallest number of ADC bits that can be used?

(5) Question 16. You are asked to interface a DC motor with the following specifications: 12 V voltage, 1A max current, 10 μH inductance, 10ms time constant, 300 g-cm torque, and 600 RPM maximum speed. The microcontroller should be able to spin the motor in both directions. Which interface chip would you use to connect the microcontroller to the motor? You do not have to build the circuit, just give the part number.

(5) Question 17. The SPI channel is used to pass data in both directions between two microcontrollers. One microcontroller is master and the other is a slave. Assume the SPI clock frequency is 8 MHz. To communicate, the following sequence of steps occur in this order. The entire sequence occurs every 1 ms.

1) The slave puts 8-bit data in its SPI data register
2) The master puts 8-bit data in its SPI data register
3) The two SPI hardware systems active transmitting the data
4) The slave reads its SPI data register, getting the value sent by the master
5) The master reads its SPI data register, getting the value sent by the slave

What is the actual bandwidth being communicated in this system, in bytes/sec?

(10) Question 18. There is a negative logic push-button interfaced to PP0. There is a global variable, called **`Counter`**. You are asked to write the ISR so that this variable is incremented each time the button is pushed. The switch will not bounce. There are no other Port P pins that need interrupts in this system, just PP0. However, please make all accesses friendly. No pull-up or pull-down is required.

(4) Part a) Show the ritual used to initialize the system, including interrupt enable. The main program will call this ritual once at the beginning then perform other unrelated tasks.

(4) Part b) Show the Port P key wakeup ISR.

(2) Part c) Assume the E clock is 24 MHz. Which choice from the following list is the best estimate of the time to takes to run once instance of the ISR? {1ns, 10ns, 100ns, 1μs, 10μs, 100μs, 1ms, 10ms, or 100ms} Include the time to affect the context switching.

(10) Question 19. Design an analog filter needed for a data acquisition system. The signals of interest are 0 to 50 kHz, and the sampling rate is 100 kHz. Use rail to rail op amp(s), show your work and label all resistors and capacitors.

(5) Question 20. I downloaded from the internet the following implementation of a FIFO queue.

```
// Two-index implementation of the FIFO
// can hold 0 to FIFOSIZE elements
#define FIFOSIZE 16 // must be a power of 2
#define FIFOSUCCESS 1
#define FIFOFAIL    0
typedef char dataType;
unsigned short volatile PutI;  // put next
unsigned short volatile GetI;  // get next
dataType static Fifo[FIFOSIZE];
void Fifo_Init(void){
  PutI = GetI = 0;      // Empty
}
// return FIFOSUCCESS if successful
int Fifo_Put(dataType data){
  if((PutI-GetI) & ~(FIFOSIZE-1)){
    return(FIFOFAIL); // Failed, fifo full
  }
  Fifo[PutI&(FIFOSIZE-1)] = data; // put
  PutI++;  // Success, update
  return(FIFOSUCCESS);
}
// return FIFOSUCCESS if successful
int Fifo_Get(dataType *datapt){
  if(PutI == GetI ){
    return(FIFOFAIL); // Empty if PutI=GetI
  }
  *datapt = Fifo[GetI&(FIFOSIZE-1)];
  GetI++;  // Success, update
  return(FIFOSUCCESS);
}
```

(2) Part a) Assuming the FIFO is a power of 2, does this FIFO work at least as good as Program 4.14 in the book (or the FIFOs in SCI starter files)? Do you think it can hold from 0 to FIFOSIZE elements without being confused about being full versus empty? Answer "Yes it works" or "No, it has bugs"

(2) Part b) Assuming exactly one thread calls puts and exactly one thread calls gets, are there any critical sections? Answer "There are NO critical sections" or "There ARE critical sections"

(1) Part c) If I were to need a 16-bit FIFO can I simply change `typedef char dataType;` to `typedef short dataType;`? Answer "Yes it works" or "No, it has bugs"