Jonathan W. Valvano First:________________ Last:____________________

March 3, 2017, 9:00am-9:50am. This is a closed book exam, with one 8.5 by 11-inch crib sheet. You have 50 minutes, so please allocate your time accordingly. ***Please read the entire quiz before starting***.

(15) Question 1. You are asked to consult on a project because they have weird and intermittent bugs. The system runs on a Freescale 9S12, which is nothing like the Cortex M, but you decide to look at it anyway. You see many read-modify-write accesses to output ports. To set bit 2, the software executes **`PTT |= 0x04;`** To clear bit 0, the software executes **`PTT &= ~0x01`**; To set bit 7, the software executes **`PTT |= 0x80;`** To investigate, you find this assembly code generated by the compiler.

`LDAA PTT`
`ORAA #4`
`STAA PTT`
	`LDAA PTT`
	`ANDA #$FE`
	`STAA PTT`
		`LDAA PTT`
		`ORAA #$80`
		`STAA PTT`

Do these read-modify-write accesses to Port T create critical sections? Circle your choice and justify

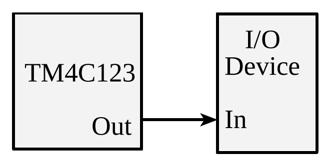
Yes, they are *critical*

No, they are *not critical*

Justify your answer

On most computers individual instructions execute atomically. Notice the access to the shared global requires three instructions. It is critical because the read-modify-write accesses are nonatomic.

(10) Question 2. You have connected a TM4C123 output pin to an unknown device, with 8-mA mode selected. Your software outputs a 1 to the pin, but your voltmeter measures only 2.3 V.



Is it broken? Specify *OK* or *Broken*: Broken

If *OK*, explain why. If *Broken*, show at least one parameter/equation not satisfied.

It is broken, because the output voltage is less than V_{OH} = 2.4V

(5) Question 3. Consider an ideal capacitor. Which is correct? There is one answer, put letter in box.

A) Voltage is directly proportional to current.
B) Current is proportional to a change in voltage.
C) Voltage is proportional to a change in current.
D) At DC, the capacitor can be considered a short circuit.
E) None of the above.

B) I=C*dV/dt

(10) Question 4. Let *N1 N2 N3 N4* be the values of four 16-bit signed decimal fixed-point numbers each with a resolution of 0.1 Assume **`I1`**, **`I2`**, **`I3`**, and **`I4`** are the corresponding integer parts. Write the body of the function that implements fixed-point math, *N4 = N3*N2+N1*. Minimize dropout, but don't worry about overflow.

```
int16_t Math(int16_t I1, int16_t I2, int16_t I3){ int16_t I4;
```

```
// step 1, write desired action: N4 = N3*N2+N1
// step 2, substitute definitions: I4/10 = I3/10*I2/10+I1/10
// step 3, solve for I4, factor, simplify:
I4 = (I3*I2)/10 + I1;
```

```
  return  I4;
}
```

(10) Question 5. Show the C code to create a signed 16-bit global variable that is shared between the main program and an ISR. For example, define **`Count`** in the correct manner for this use case.

```
int16_t volatile Count;
// adding static is ok, const is very wrong
```

```
void main(void){
  Init(); // Systick interrupts every 1ms
  while(1){
    Count = 1000;
    while(Count>0){}
    GPIO_PORTF_DATA_R ^= 0x02;
  }
}
void SysTick_Handler(void){
  Count--;
}
```

(10) Question 6. How much stack space does this ISR need? Give your answer in 32-bit words.

```
uint32_t Count=0;
void Timer5A_Handler(void){
  Count++;
  TIMER5_ICR_R = 0x00000001;  // acknowledge timeout
}
```

8 registers are pushed, each is 1 word, 8 words are needed

`Count` is not local, it is static; meaning it does not get placed on the stack.
Interrupts push R0,R1,R2,R3,R12,LR,PC,PSW on stack

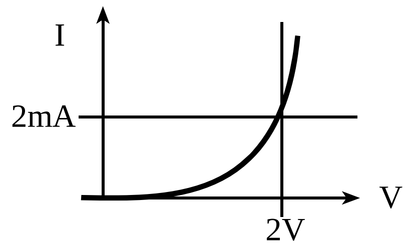
(5) Question 7. What is the response from the internet when a TCP packet is lost?

TCP guarantees delivery, so it will try again. Contrast with UDP, which will not retransmit.

(15) Question 8) You are asked to configure Timer 5A to interrupt every 100us. The bus clock is 80 MHz. Put your answers in the boxes. Make Timer 5A an interrupt with the lowest priority. Timer 5 priority is in bits 7,6,5 of PRI23 register. Timer 5A is interrupt 92, which is bit 92-64=28 of EN2.

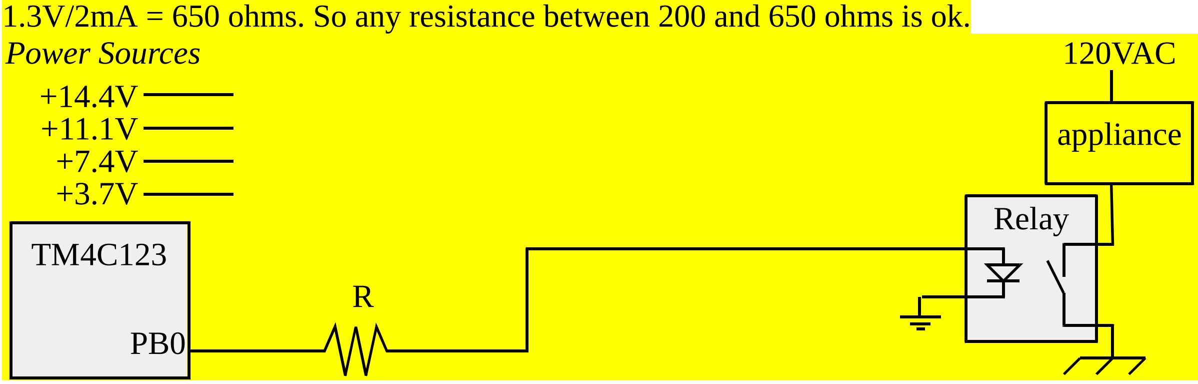
```
SYSCTL_RCGCTIMER_R |= [0x20] ;

while((SYSCTL_PRTIMER_R&[0x20] )==0){};

TIMER5_CTL_R = 0x00000000;    // disable during setup
TIMER5_CFG_R = 0x00000000;    // 32-bit mode
TIMER5_TAMR_R  = 0x02;        // count down period

TIMER5_TAILR_R = [7999] ;

TIMER5_TAPR_R = 0;            // 12.5ns resolution
TIMER5_ICR_R = 0x00000001;    // clear timeout flag

TIMER5_IMR_R = [0x01] ;

NVIC_PRI23_R=(NVIC_PRI23_R&[0xFFFFFF00])|[0xE0] ;

NVIC_EN2_R = [0x10000000] ;

TIMER5_CTL_R = 0x00000001;    // enable timer5A
EnableInterrupts();
```

(15) Question 9. Interface a solid state relay to the microcontroller. A digital output on PB0 controls the relay. PB0 is an output with 8-mA selected. If PB0 is high, a 2V signal should be applied across the relay control, and the 120VAC switch will activate causing the appliance to turn on. If PB0 is low, no current should flow through the relay control, and the 120VAC switch will deactivate causing the appliance to turn off. The desired set-point to activate the relay is 2V, 2mA. The following graph plots the voltage current relation on the relay control.



You may use +3.7V, +7.4V, +11.1V, or 14.4V power. Decide whether to use no transistor ($0.00), a 2N2222 ($0.44), or a TIP120 ($0.72). Select the least expensive circuit that will operate the relay. Show your work including resistance values. No software needed, just the hardware circuit.

The interface is a simple positive LED interface with no driver. It is ok to pick any V_{OH} between 2.4 and 3.3 V. At V_{OH} =2.4V, R = (2.4-2)/2mA = 0.4V/2mA = 200 ohms. At V_{OH} =3.3V, R = (3.3-2)/2mA = 1.3V/2mA = 650 ohms. So any resistance between 200 and 650 ohms is ok.



Parameters for the TM4C123 microcontroller, with 8-mA mode selected

I_{OL} = 8mA, I_{OH} = 8mA, I_{IL} = 2μA, I_{IH} = 2μA,

V_{OL} = 0.4V, V_{OH} = 2.4V, V_{IL} = 1.3V, V_{IH} = 2.0 V

	31–3	2–0	Name
$4003.0000		GPTMCFG	TIMER0_CFG_R

	31–4	3	2	1-0	
$4003.0004		TAAMS	TACMR	TAMR	TIMER0_TAMR_R

	14	13	11-10	8	6	5	3-2	0	
$4003.000C	TBPWML	TBOTE	TBEVENT	TBEN	TAPWML	TAOTE	TAEVENT	TAEN	TIMER0_CTL_R

	31-11	10	9	8	7-4	2	1	0	
$4003.0018		CBEIM	CBMIM	TBTOIM		CAEIM	CAMIM	TATOIM	TIMER0_IMR_R

	31-11	10	9	8	7-4	2	1	0	
$4003.001C		CBERIS	CBMRIS	TBTORIS		CAERIS	CAMRIS	TATORIS	TIMER0_RIS_R

	31-11	10	9	8	7-4	2	1	0	
$4003.0020		CBECINT	CBMCINT	TBTOCINT		CAECINT	CAMCINT	TATOCINT	TIMER0_ICR_R

	31–16	15–0	
$4003.0028	TAILRH	TAILRL	TIMER0_TAILR_R

	31–8	7-0	
$4003.0038		TAPSR	TIMER0_TAPR_R

	31–8	7-0	
$4003.0040		TAPSMR	TIMER0_TAPMR_R

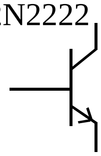
	31–16	15-0	
$4003.0048	TARH	TARL	TIMER0_TAR_R

Address	31 – 29	23 – 21	15 – 13	7 – 5	Name
0xE000E400	GPIO Port D	GPIO Port C	GPIO Port B	GPIO Port A	NVIC_PRI0_R
0xE000E404	SSI0, Rx Tx	UART1, Rx Tx	UART0, Rx Tx	GPIO Port E	NVIC_PRI1_R
0xE000E408	PWM Gen 1	PWM Gen 0	PWM Fault	I2C0	NVIC_PRI2_R
0xE000E40C	ADC Seq 1	ADC Seq 0	Quad Encoder	PWM Gen 2	NVIC_PRI3_R
0xE000E410	Timer 0A	Watchdog	ADC Seq 3	ADC Seq 2	NVIC_PRI4_R
0xE000E414	Timer 2A	Timer 1B	Timer 1A	Timer 0B	NVIC_PRI5_R
0xE000E418	Comp 2	Comp 1	Comp 0	Timer 2B	NVIC_PRI6_R
0xE000E45C	Wide Timer 0B	Wide Timer 0A	Timer 5B	Timer 5A	NVIC_PRI23_R
0xE000ED20	SysTick	PendSV	--	Debug	NVIC_SYS_PRI3_R

Address		30	28	19		6	5	4	3	2	1	0	Name
0xE000E100		F		Timer0A		UART1	UART0	E	D	C	B	A	NVIC_EN0_R
0xE000E104											UART2		NVIC_EN1_R
0xE000E108			Timer5A										NVIC_EN2_R

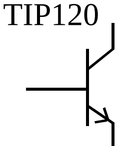
Address	31-24	23-17	16	15-3	2	1	0	Name
$E000E010	0	0	COUNT	0	CLK_SRC	INTEN	ENABLE	NVIC_ST_CTRL_R
$E000E014	0	24-bit RELOAD value						NVIC_ST_RELOAD_R
$E000E018	0	24-bit CURRENT value of SysTick counter						NVIC_ST_CURRENT_R

2N2222



V_{ce} = 0.3V
V_{be} = 0.6V
h_{fe} = 100
I_{ce} =500mA max

TIP120



V_{ce} = 0.8V
V_{be} = 1.5V
h_{fe} = 2000
I_{ce} =5A max