Jonathan W. Valvano Solution

Calculator is allowed (no laptops, phones, devices with wireless communication). You must put your answers in the boxes. Do not write on the back of the page. When you are done, you turn in the closed-book part and can start the open book part.

(4) Question 1. Which digital controller type has the best accuracy? Choose A, B, or C.

A) Proportional
B) Integral
C) Derivative

B) Integral

(6) Question 2. In the context of internet communication, list three items contained in a socket.

A socket is an application endpoint for communication that encapsulates IP address, the transport protocol, and the port
Ok if you included data message

(4) Question 3. Consider the following five architectures: **successive approximation**, **resistor string, flash, R-2R, sigma-delta**.

a) Which DAC type is most likely to be monotonic? Resistor string

b) Which DAC type uses fewer components to build? R-2R

(6) Question 4. Choose which capacitor type to use for each application. Give one answer for each: tantalum, ceramic or neither.

Building an analog high pass filter: Ceramic

Removing high amplitude low frequency noise on the power line: Tantalum

Increasing the bandwidth on a high-speed digital communication: Neither

(5) Question 5. Which inequality is not required to connect one digital output to one digital input? Choose one answer

A) $V_{IL} < V_{OL}$
B) $V_{IH} < V_{OH}$
C) $I_{IL} < I_{OL}$
D) $I_{IH} < I_{OH}$

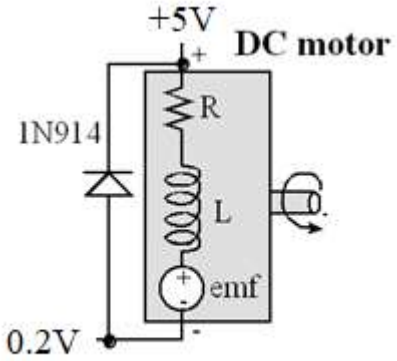
A) $V_{IL} < V_{OL}$ it should have been $V_{IL} > V_{OL}$

(5) Question 6. Consider measurement of period performed in Lab 10. What factor determines the period measurement resolution?

The resolution of period measurement using input capture is the period of the bus clock used to latch the timer

(5) Question 7. Consider motor interface powered by 5V and a MOSFET like Lab 10. An ohmmeter measures the motor coil resistance of 100 ohms, when not powered and disconnected from the circuit. However, a current meter inline with the powered circuit measures a current of 200 mA, where 200mA is much larger than 5V/100ohms = 50mA. Explain how this happened?

There is an **emf** that couples mechanical and electrical energy. Friction causes the emf term to create a negative electrical voltage, making the current larger than 5V/100ohms

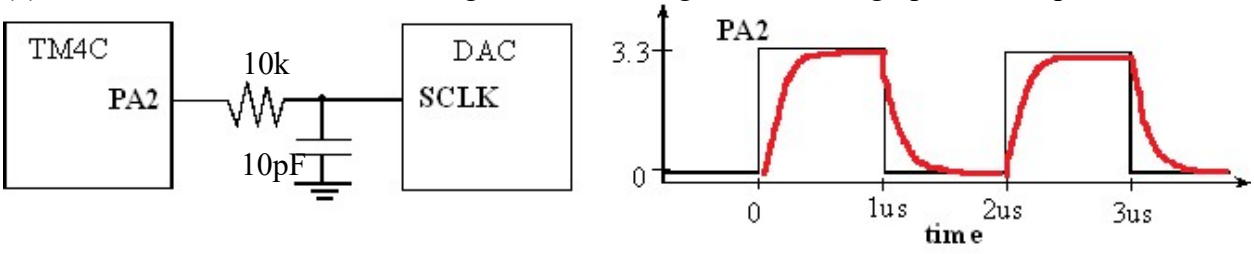


Note: this is NOT back emf. Back emf occurs across the inductor on large dI/dt, caused by V = L dI/dt and removed by the 1N914 snubber

(4) Question 8. Consider these four architectures for power regulation: A) linear, B) boost, C) buck, D) buck-boost. For each system choose the best regulator type. Each system requires a 3.3V power supply. If there is more than one good answer, just give one of the answers.

a) The system is powered with a 1.5V battery.	B or D
b) The system requires a power supply with low noise.	A
c) The system must dissipate the least amount of heat.	B C or D
d) The system must run on any battery voltage from 3 to 12V.	D

(5) Question 9. Sketch the SCLK signal on the voltage versus time graph. 10k*10pF =0.1us



Jonathan W. Valvano First:_____ Last:_____

Open book, open notes. You may use a calculator or tablet with screen lays flat on the table and no wireless communication. Basically, the screen cannot be visible to other students You must put your answers in these boxes. Please don't turn in any extra sheets or write on the back of the pages.

(10) Problem 10. Assume the following edge-triggered ISR is invoked every 1000 to 2,000,000us (this period is variable and unknown). The time to execute **`UserTask()`** ranges from 10 to 100us (this time is variable and unknown). Timer2 has been initialized so 32-bit **`TIMER2_TAR_R`**. The prescale is set so the timer decrements every 1usec. The 32-bit **`TIMER2_TAIL_R`** is set to 0xFFFFFFFF. On each interrupt (it is ok if the first interrupt calculates incorrectly), measure the period between this interrupt and the previous interrupt. Second, measure the execution time of this running of the ISR. Write C code to measure **`Period`** and **`Time`**.

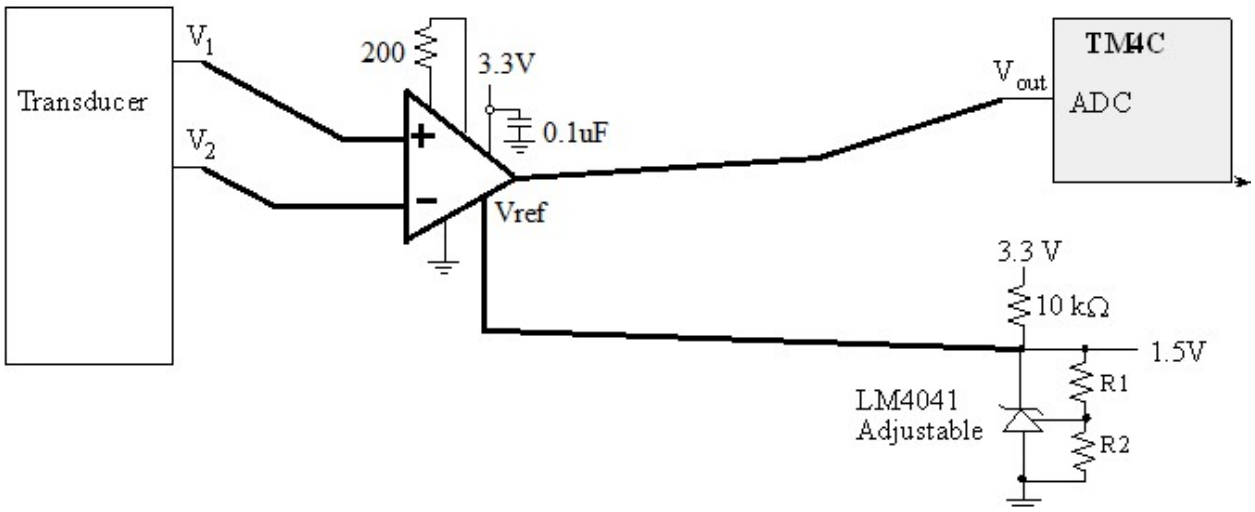
```
uint32_t Period; // usec time between interrupt invocations
uint32_t Time;   // usec time to complete current ISR invocation
uint32_t Percentage; // decimal fixed point, 0.01% resolution
void GPIOPortF_Handler(void){
  static uint32_t Last=0;
  uint32_t start = TIMER2_TAR_R;
    Period = Last - start;
  GPIO_PORTF_ICR_R = 0x10; // acknowledge
  UserTask();
  Time = start - TIMER2_TAR_R;
  Last = start;
  Percentage = (10000*Time)/Period; // units 0.01%
}
```

Bonus Question:

When Valvano shouts "*World Domination!*", he is referring **time** as an input parameter (input capture) or output (PWM). Time is low cost and highly effective (precision) I/O parameter

(10) Question 11. Design an analog circuit that has two inputs and one output, such that the output V_{out} is $1005*(V_1-V_2)+1.5$. The input voltages are constrained so the output will range between from 0 to 3.3V. No analog filter is required. The only available power supply voltage is 3.3V. Assume R1 and R2 are already chosen to achieve a reference of 1.5V. Full credit for a circuit with low noise and excellent CMRR. You may use any analog chip from lab, book, or lecture PowerPoints. Show design steps. Specify values for all resistors, capacitors, diodes, inductors, and chip numbers. Show power supply connections for all chips.

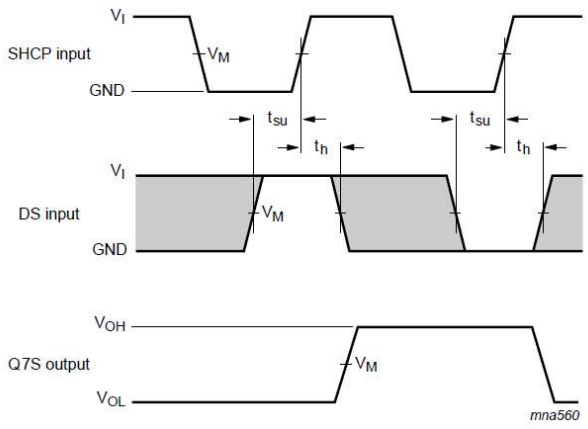
Good CMRR, low noise, differential means instrumentation amp INA122
Rg = 200k/(G-5) = 200k/1000 = 200; Notice the gain resistor is NOT connected to the output.



Trying to build this with regular op amps will not have good CMRR or be low noise

(5) Question 12. Consider this shift register. SHCP is the clock input. DS is the data input. Q7S is a digital output. Show the timing equation for the **data required interval** using symbols from the figure.

DR = (↑SHCP – t_{su}, ↑SHCP + t_h)



(6) Question 13. This FIFO queue implementation has shared globals **`Size`**, **`GetI`**, **`PutI`**. Consider the read modify write access to **`Size`**. It can store up to 16 elements. **`Fifo_Put`** is called by one ISR, and **`Fifo_Get`** is called by a second ISR.

```
int Fifo_Put(int32_t data){
  if(Size==16) return 0; // full
  FIFO[PutI] = data;     // save
  PutI = (PutI+1)&0x0F;  // next
  Size++;
  return 1;
}
```

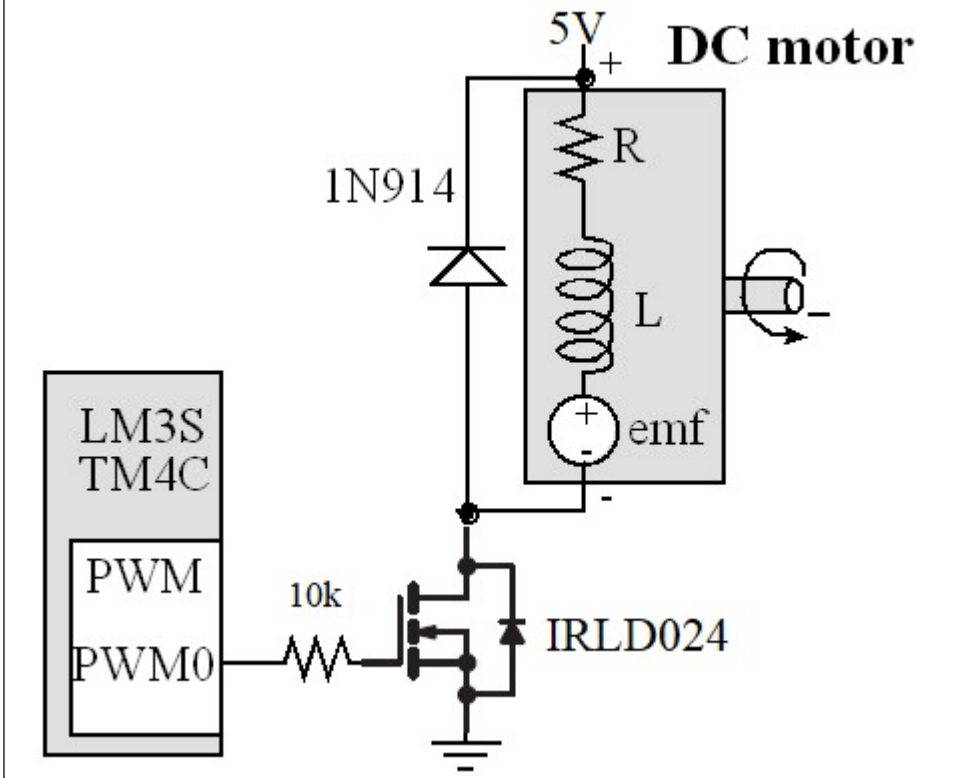
```
int Fifo_Get(int32_t *datapt){
  if(Size == 0) return 0; // empty
  *datapt = FIFO[GetI];   // retrieve
  GetI = (GetI+1)&0x0F;   // next
  Size--;
  return 1;
}
```

How do I configure the two interrupts, so these two functions are not critical to each other? No changes to the Fifo functions are allowed.

Make the priority of the time interrupts equal so the two ISRs become atomic with respect to each other

(10) Question 14. Interface a DC motor to the TM4C123. The desired voltage is 4 to 5 V, and the expected current is 0.1 to 2A. You may use any chip from lab, book, or lecture PowerPoints. Specify values for all resistors, diodes, inductors, capacitors, and chip numbers. To which TM4C123 pin would you connect the interface?

Connect to any PWM pin (from Lecture slides) This is the lab 10 circuit, you could also have used an N-channel TIP120 Darlington. Could have used H bridge like DRV8838 or DRV8848

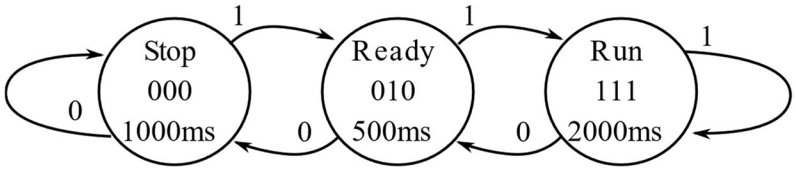


(10) Question 15. Run this PID controller

```
void SysTick_Handler(void){ // runs every 10ms, dt = 0.01s
static int32_t LastE=0; int32_t dE;
  E =  SetPoint- ReadDirection();
  P = (1256*E)/10000;        // 0.1256 = 1256/10000
  if(P < -1000) P = -1000;  // limit the range of P to -1000 to +1000
  if(P > +1000) P = 1000;   // limit the range of P to -1000 to +1000
  I = I + (23*E)/1000;       // 2.3*dt = 0.023 = 23/1000
  dE = E-LastE; // slope
  LastE = E;
  D = (15*dE)/100;           // 0.0015/dt = 0.15 = 15/100
  if(D < -100) D = -100;    // limit the range of D to -100 to +100
  if(D > +100) D = 100;     // limit the range of D to -100 to +100
  U = P+I+D;
  if(U < -4000) U = -4000;  // limit the range of U to -4000 to +4000
  if(U > +4000) U = 4000;   // limit the range of U to -4000 to +4000
  UR = 5000+U;
  UL = 5000-U;
  Right_Duty(UR); Left_Duty(UL); // Output to two motors
}
```

Unused Questions

Run this FSM in the background using SysTick interrupts. There is one input on PC4, and there are three outputs on PC7-5.



FSM data structure and main program are fixed and should not be changed. You do not need to write the initialization functions called from main.

```
struct State {   // 1-bit input
  uint32_t Out;  // 3-bit output
  uint32_t Time; // 1ms units
  const struct State *Next[2];};
typedef const struct State State_t;
#define Stop  &FSM[0]
#define Ready &FSM[1]
#define Run   &FSM[2]
State_t FSM[3]={
{0, 1000,{Stop,Ready}},
{2,  500,{Stop,Run}},
{7, 2000,{Ready,Run}}};
```

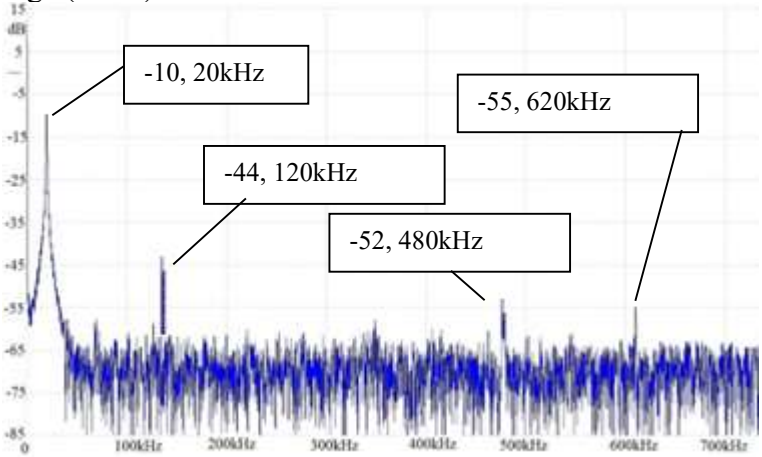
```
State_t *Pt;  // state pointer
int main(void){
  PLL_Init(); // 80 MHz
  NVIC_ST_RELOAD_R = 79999;
  NVIC_ST_CURRENT_R = 0;
  NVIC_ST_CTRL_R = 7;
  PortC_Init(); // Init PC7-4
  Pt = Stop;
  EnableInterrupts();
  while(1){
  }
}
```

Write the SysTick ISR in C. You may not call any functions within the ISR. You will get 0 points if you try to use **for**, **do-while**, or **while** loops. You must use the periodic interrupt to affect the time delay in each state. You may add local and/or static variables to the ISR. Make the code friendly.

```
void SysTick_Handler(void){ // runs every 1ms
static time=0;
  time++;
  if(time == Pt->Time){
    time = 0;
    Pt = Pt->Next[(GPIO_PORTC_DATA_R&0x10)>>4];
    GPIO_PORTC_DATA_R = (GPIO_PORTC_DATA_R&0x1F)+(Pt->Out)<<5;
  }
}
static time=1;
  time--;
  if(time == 0){
    Pt = Pt->Next[(GPIO_PORTC_DATA_R&0x10)>>4];
    GPIO_PORTC_DATA_R = (GPIO_PORTC_DATA_R&0x1F)+(Pt->Out)<<5;
    time = Pt->Time;
  }
}
```

(5) Question 2. Let **r** be a distance in cm, **I** is the integer part of a binary fixed-point number with resolution of 0.25cm. Let **A** = 3.1415*$\mathbf{r}^2$ in cm^2, **J** is the integer part of a binary fixed-point number with resolution of 0.125cm^2. Show the software that calculates **J** given **I**, using integer math.

Consider a real-time data acquisition system with an 8-bit ADC sampled at f_s. The following data were collected at the input of the ADC. The desired signal exists in the 0 to 50 kHz range, and the rest of what you see in this spectrum is noise. Yes, it needs an analog filter. However, if you were to sample this signal exactly like this, what is the slowest sampling rate f_s allowed that will prevent aliasing? $dB_{FS} = 20 \log_{10}(V/3.3) = -48$dB for this 8-bit ADC



120 kHz signal would alias, so sample faster than 240 kHz

(5) Question 13. Design a voltage regulator that converts a 3.7V LiIon battery into +5V supply. You may use any chip from lab, book, or lecture PowerPoints. Specify values for all resistors, diodes, inductors, capacitors, and chip numbers.

Figure 9.4 in book