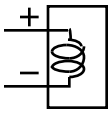
Jonathan W. Valvano First:_____ Last:_____
February 28, 2001

This is an open book, open notes exam. You must put your answers on these pages only, you can use the back. You have 50 minutes, so please allocate your time accordingly.
Please read the entire quiz before starting.

(20) Question 1. Design the hardware interface that allows the computer to control a DC motor. The computer will use pulse-width modulation on PT0. No software is required, just the hardware. The motor coil needs about 24 V at 0.5A to run at maximum speed. The system should NOT be isolated, so no 6N139 or IL-5 can be used. There will be both positive and negative back EMF voltages, so protect the electronics. Show the hardware circuit, labeling all chip numbers and component values.

PT0----



(20) Question 2. The SCI port is not used for any function other than debugging. The following debugging instrument has been added to a real-time embedded system.

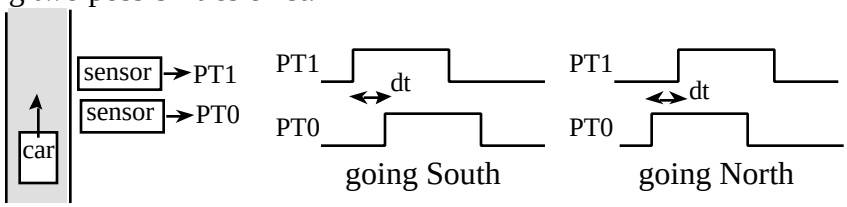
```
OutString("r="); OutUDec(r); OutChar(13);
```

The serial I/O system uses the interrupting version as implemented in SCI12A.C.

(10) Part a) The programmer claims that this debugging instrument in minimally intrusive. What exactly does she mean? (Be very specific)

(10) Part b) What condition, if it were to occur, would cause this debugging instrument to be intrusive?

(60) Question 3. The objective of this program is to design a speed-meter using two car sensors and input capture. Each car sensor signals high when it senses a car. The sensors are located 5 feet apart, so that as a car travels down the road, one sensor triggers, then the other one. The time difference (**dt**) between rising edges on the sensors is a function of the car **Speed**. Since cars are at least 5 feet long, the following two possibilities exist.



The **Speed** (measured in mph) is inversely proportional to time difference (measured in ms)

Speed = (5ft)/**dt** * (1mi/5280ft) * (3600sec/hr) * (1000ms/s) = 3409/**dt**

You will use three background interrupts (one OC and two IC) to measure **dt** and calculate **Speed**. You may assume only one car approaches at a time and the **Speed** ranges from 1 to 100 mph. Therefore, 34ms<**dt**<3409ms. The hardware is fixed and can not be changed. You may use any other Port T timer feature. The measurements occur in the background and once initialized, the foreground is free to execute other unrelated functions. Do not worry about direction, just measure the **Speed** in mph and write each new measurement into the global variable.

```
volatile unsigned short Speed; // car speed in mph
```

Show the ritual and three ISRs.

Bonus question: what is the Speed resolution?