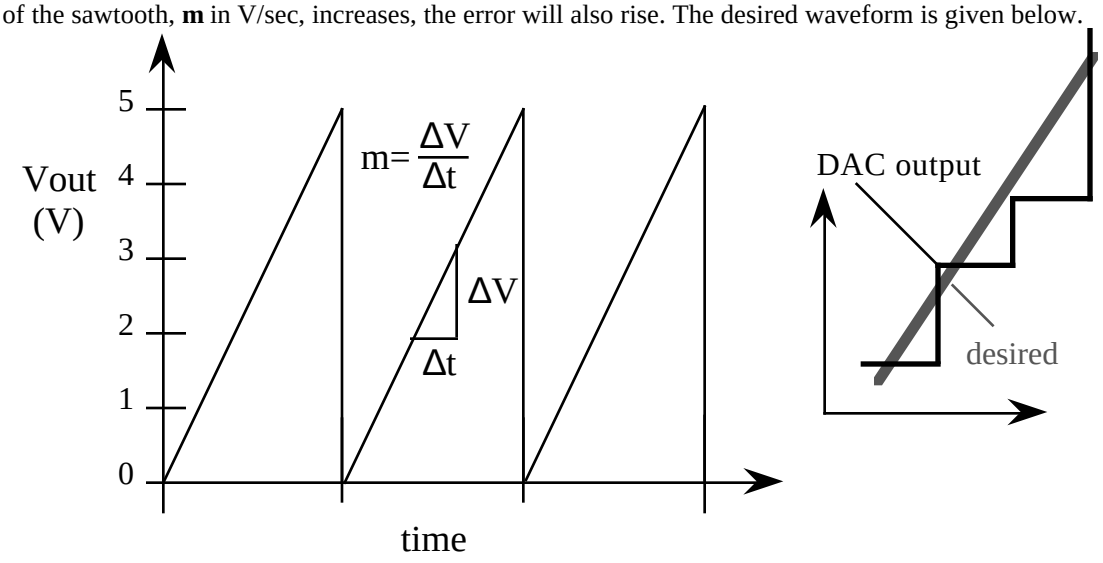
Jonathan W. Valvano First Name: _____ Last Name:_____
May 16, 2000, 9 am to 12 noon

This is an open book, open notes exam. You may put answers on the backs of the pages, but please don't turn in any extra sheets.

(10) Question 1. The accuracy of a DAC-based signal generator is a function of the DAC precision, **n** in bits, and the software output rate, $\mathbf{f_s}$ in Hz. In this problem, the DAC range is fixed at 0 to +5V, and the signal to be generated is a simple sawtooth. As it turns out, the accuracy of the signal generator also depends on the signal itself. In particular, as the slope of the sawtooth, **m** in V/sec, increases, the error will also rise. The desired waveform is given below.



Let **e(t)** be the difference between the desired and actual waveform. There are many definitions of accuracy. We could define the error in this signal generator as the maximum difference, $\mathbf{e_{max}}$. A better estimate is the average difference between the desired and actual signals.

$$\text{average error} = \frac{1}{T}\int_0^T |e(t)|dt$$

The best estimate of accuracy is the mean squared error.

$$\text{MSE} = \sqrt{\frac{1}{T}\int_0^T e(t)^2 dt}$$

Derive an equation that relates the maximum error, $\mathbf{e_{max}}$, to the DAC precision **n**, the DAC output rate, $\mathbf{f_s}$, and the slope of the sawtooth, **m**. (Do not calculate average error or MSE.) Hint: there are two error components, one due to the finite DAC resolution and the other due to the DAC output rate, $\mathbf{f_s}$. Assume a periodic ISR (interrupting at $\mathbf{f_s}$) outputs to the DAC. Assume at the time of the periodic interrupt, the software chooses the digital value that minimizes e(t) at that time. (One could write ISR software that minimizes the **e(t)** over the entire interval, but the derivation of the $\mathbf{e_{max}}$ equation would become more complex.)

(30) Question 2. Design a pulse width measurement system with a range of 10ms to 50s and a resolution of 10ms. Measure the time from the rising edge to the falling edge. You may use any of the PORTT input capture, output compare features, but you MUST USE interrupt synchronization. Assume the digital signal is just connected to PT0. You need not worry about pulse width too small, but you need to check for overflow, i.e., if the pulse-width is longer than 50s, then an error condition must be set.

(10) Part a) Design the user-level device driver prototypes. You will need to define initialization, and measurement functions. In particular, give the *.H header file, including documentation. This file contains the public functions and public data structures. This section will be graded on style. Feel free to be creative as to exactly how it will be used.

(20) Part b) Show the C software implementation of the device driver. E.g., give the *.C program file, including documentation. There is a main program that you will not write that will call your functions defined in part a) as needed. **You MUST use interrupt synchronization with NO backward jumps in your ISRs.**

(40) Question 3. In this problem, you will design a 6812-based ohmmeter. The resistance range is 0 to 1000 Ω. Because you are using the 8-bit ADC and a linear analog circuit, the measurement precision will also be 8 bits. The frequencies of interest are 0≤f<1 Hz. You will design an analog circuit that passes a 1 mA constant current through the unknown resistance, then use a differential amplifier to produce a 0 to +5V signal for the internal ADC. Let

R is the unknown resistance to be measured (0≤R≤1000Ω)

I_1 is the current across the unknown resistor, (a constant value of 1 mA).

V_1 is the differential voltage drop across the unknown resistor, ($0 \le V_1 = I_1/R \le 1V$).

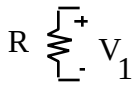
V_2 is the ADC input voltage, ($0 \le V_2 \le 5V$).

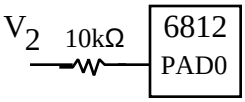
(5) Part a) What will be the measurement resolution in Ω.

(5) Part b) What sampling rate would you choose? Why?

(5) Part c) What is the best way in this system to eliminate 60 Hz noise pickup? Justify your answer. Be sure to implement this noise rejection scheme in the following sections.

(15) Part c) Design the analog circuit that connects R to the 6812 ADC. Specific chip numbers and component values.





(5) Part e) Show the ritual that initializes the system (data structures, port configuration, and interrupt arm). The main program will not directly participate in the measurement, but be free to execute other functions. You will not write the main program. Add and explain any variables you need.

(5) Part f) Show the interrupt service routine that samples the ADC and calculates resistance in Ω. Store the resistance measurement into a global variable.

(20) Question 4. Consider the 8K RAM MC68HC812A4/60L64 interface presented in Section 9.7.2 (pp. 539-541.) There are three options when a particular memory chip is not fast enough for the 8 MHz 6812. The first option, as illustrated in Chapter 9, is to add cycle stretches to the memory accesses. Section 9.7.2.2 (pp. 545-548) develops the timing equations that justify the need for one cycle stretch for the MC68HC812A4/60L64 interface.

(10) Part a) The second option is to purchase a faster memory chip. The following table lists the timing parameters of the particular 60L64 RAM chip used in Chapter 9. Fill in the **desired** specifications for each parameter that would allow this 6812/60L64 interface to operate without cycle stretching. If a parameter value is OK as is, then do not change it. I.e., change only those parameters that must be changed. Show you work.

parameter	symbol	actual value (ns)	desired value (ns)
address access	t_{AVQV}	150	
E1 access	t_{E1LQV}	150	
G access	t_{GLQV}	70	
address deactivation	t_{AXQX}	20	
E1 deactivation	t_{E1HQZ}	0	
G deactivation	t_{GHQZ}	0	
data set up	t_{DVWH}	60	
data hold	t_{WHDX}	0	

(10) Part b) The third option is to slow down the processor. Choose the fastest E clock frequency that would allow this 6812/60L64 interface to operate without cycle stretching. Show you work.