# ME4405 - Fundamentals of Mechatronics (Fall 2019)

### Lab Assignment Nine

## Constructing and Characterizing a Thermal System Using the MSP432 and I2C Serial Communication

**Due Thursday, Nov 14th, 2019**

**Objective:** The objective of this lab is to build a thermal system, implement an open loop controller using an MCU, and characterize the step response and control sensitivity using experimental data.

## Deliverables and Grading:

To get credit for this lab assignment you must:

1. Demonstrate proper operation of your code (the I2C sensor readout and the PWM output) to the TA or instructor during lab or office hours. **(20 points)**
2. Submit all plots and answer all questions listed at the end of the lab manual **(40 points)**.
3. Submit the commented final version of your code on Canvas. **(Pass/Fail)**

## Setup:

This lab requires Code Composer Studio, the MSP432, and a thermal system comprised of a film canister, heating element, and temperature sensor. The lab uses onboard features of the device including GPIO, Timer A, PWM, and I2C.

## Problem Statement:

Create a thermal system consisting of a heating element (resistor) and an I2C temperature sensor enclosed in a film canister. The microcontroller will be used to apply a variable current signal to the heating element using PWM. This lab will characterize two fundamental properties of the dynamic system – the system's step response, and the system's control sensitivity. In this case, control sensitivity refers to the steady-state temperature changes resulting from applying various control inputs.

Two experiments will be performed. In the first experiment, an ambient temperature calibration will first be performed without heating the resistor by taking temperature readings at a rate of 1 Hz for 120 seconds with a duty cycle of 0. Then, a step input will be applied by changing the duty cycle to 50%. Again, temperature readings will be taken at a rate of 1 Hz until the temperature stops changing. The step response is shown by plotting this time series.

In the second experiment, an array of duty cycles (10-70% in 20% increments) will be applied to the system. At each duty cycle, temperature readings should be taken at 1 Hz. Once the system reaches steady state at that duty cycle, record the average of 30 readings at steady state. This process will be performed at all tested duty cycles. A plot of duty cycle vs steady state temperature can then be constructed. The control sensitivity is the slope of the line in this plot.

## Background:

### I2C:

The Inter-integrated Circuit (I2C) Protocol is a protocol intended to allow multiple "slave" digital integrated circuits to communicate with one or more "master" chips (microprocessors and microcontrollers). I2C requires a mere two wires, like asynchronous serial, but those two wires can support up to 1008 slave devices.

Each I2C bus consists of two signals: SCL and SDA. SCL is the clock signal, and SDA is the data signal. The clock signal is always generated by the current bus master. *Each signal line should have a pull-up resistor on it, to restore the signal to high when no device is asserting it low.*

The MSP432 has two different types of serial modules; eUSCI_A modules and eUSCI_B modules. The eUSCI_A modules support both the UART and SPI protocols while the eUSCI_B modules support the SPI and I2C protocols. Using the MSP432 datasheet you can find which MSP432 pins correspond to the modules.

In I2C mode, the eUSCI_B module provides an interface between the device and I2C-compatible devices connected by the two-wire I2C serial bus. External components attached to the I2C bus serially transmit and/or receive serial data to/from the eUSCI_B module through the 2-wire I2C interface. The driverlib functions are provided to initialize the I2C modules, to send and receive data, obtain status, and to manage interrupts for the I2C modules.

I2C module can generate interrupts. The I2C module configured as a master will generate interrupts when a transmit or receive operation is completed (or aborted due to an error). The I2C module configured as a slave will generate interrupts when data has been sent or requested by a master. However, you do not need to use I2C interrupts for this lab.

Some functions relevant to the I2C are:

- `I2C_enableModule`: Enables I2C
- `I2C_initMaster`: Initializes the I2C Master and its clock source and clock rate
- `I2C_setMode`: Sets the mode of the I2C device to Transmit or Receive
- `I2C_setSlaveAddress`: Sets the address that the I2C Master will place on the bus
- `I2C_masterSendSingleByteWithTimeout`: Sends START, transmits the byte to the Slave and sends STOP. **Note: as stated in the lecture slides, you will need to tell the I2C which register you want to read from (see datasheet) before you can start receiving data.**
- `I2C_masterReceiveSingleByte`: Receives a byte that has been sent to the I2C Master Module

Please see the documentation for the specific argument list for each function. Furthermore, please see the example code provided with this lab via Canvas: Resources/Example Codes/I2C. Note that the example code is written for another type of I2C sensor, so consult the datasheet for your sensor to determine what I2C functions you need to interface with your sensor. There may be additional I2C functions (listed above) needed for your lab that are not in the example.

### Thermal System:

The thermal system consists of a heating element (resistor) and the TC74 temperature sensor. These components are enclosed in a film canister so that the heat generated by the element heats the air inside the canister. The heating element is controlled by a PWM signal applied to the transistor gate that controls the current flow into the element. The TC74 temperature sensor from is used to detect the temperature inside the canister.

First, construct the thermal system using a film canister, the 10Ω resistor, and TC74 sensor. Cut three holes in the film canister – two for each of the resistor leads, and one for the TC74. **Do NOT use the soldering iron to melt holes**; this will damage the iron. Place the resistor and TC74 inside the canister. A diagram is shown in Figure 1. Unlike Figure 1, it may be easiest to have each resistor lead coming out opposite sides of the film canister so they do not interfere with one another and short. **Make sure that the resistor itself is not touching the walls of the film canister – this can melt the plastic, which will prevent you from being able to reuse your system for Lab 10.** Try to mount the components such that the sensor is above the heating element (10Ω resistor) so that it senses the hot air as it rises inside the canister. Do not make very big holes to mount components as these will be difficult to seal to prevent heat loss.

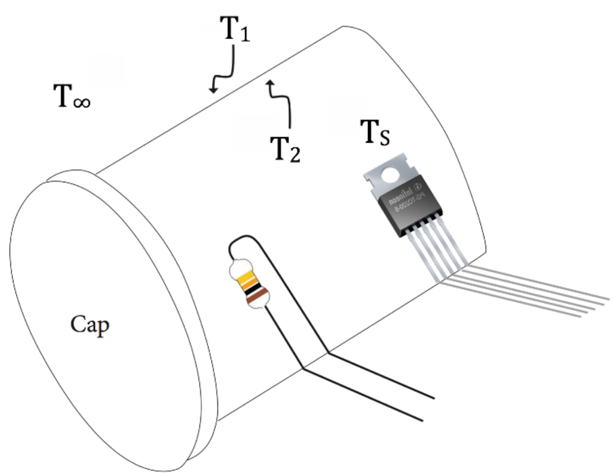


**Figure 1**. Thermal System for Labs 9-10.

Once the thermal system is built, construct the circuit for powering it as shown below in Figure 2. Note that this is a schematic and not an exact circuit diagram. To power the heating element (left DC source), use a 5V power supply. **Do not try to power the heating element using the MSP +5V pin.** To power the TC74 sensor, you can use the 3.3V pin on the LaunchPad.

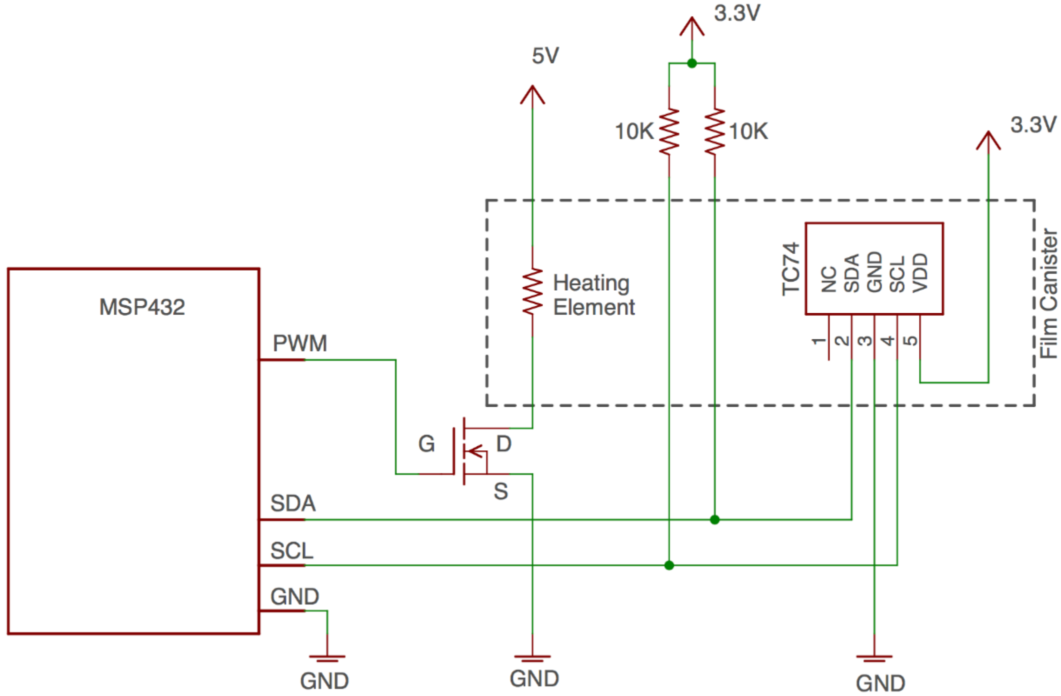


**Figure 2**. Schematic of circuit hardware setup.

**Hardware:**

The TC74 temperature sensor should be connected to an I2C pins on the MSP (e.g. pin 6.4 for SDA and pin 6.5 for SCL). You need to simply solder two wires from the sensor I2C pins (SDA and SCLK) and connect them to the corresponding I2C pins on MSP432. Remember to use a common ground. In this case, the output of the temperature sensor will be an 8-bit value with the resolution of 1°C and a range of -40°C to 125°C. The sensor can be driven by a 3.3V supply. Make sure you convert your I2C reading to provide a proper temperature value.

The MOSFET gate should be connected to the MSP on an output pin that provides PWM. By controlling the transistor gate using PWM, the amount of current flow through the resistor can be varied by changing the duty cycle (which is the control input). The heating element circuit should be driven from a 5V power supply. **Do not drive this circuit from 12V as it may overheat the resistor**.

**Software:**

There are two experiments for this lab. In the first experiment, set the duty cycle to zero and record temperature data at 1 Hz for 120 readings. This is the calibration step – the average of these 120 readings is the ambient temperature. Then, set the PWM duty cycle to 50% and start recording temperature readings at 1 Hz in a separate array. Record data until the temperature is stabilized.

This represents the step response of the system. Once complete, present this part of your assignment to receive a check-off**.**

The second experiment will apply different control inputs and the resulting steady state temperatures will be recorded. Start at 10% duty cycle. Record measurements until they stop changing, and calculate the average. Repeat these steps for duty cycles 30-70%, in 20% increments. Compute the average steady state temperature for each duty cycle applied.

When performing these tests, if you run the first experiment (to capture the step response), make sure to power off the system and let everything cool down before moving on to the second experiment (measuring steady state values). Remove the cap of the canister and wait for 5-10 min for the system to cool down.

**When taking temperature measurements, hold the canister so that the sensor is above the heating element (toward the ceiling)**. As the heating elements heats the surrounding air, the air rises. If the sensor is placed below the heating element, it may not register much of a temperature difference.

**Thermal System:**

This closed canister containing a heating element can be modeled using Fourier's Law and the heat equation. The resulting dynamic equation that governs the system is given by:

$$(R_T C)\dot{T_C} + T_C = R_T Q_{in} + T_\infty$$

where RT is the thermal resistance of the canister walls, C is the thermal capacitance of the canister, $T_c$ is the internal temperature of the canister, and $T_\infty$ is the ambient temperature outside the canister. Defining $\tilde{T} = T_C - T_\infty$, this equation can be written as,

$$(R_T C)\dot{\tilde{T}} + \tilde{T} = R_T Q_{in}$$

The applied heat Qin is given by,

$$Q_{in} = V_{res}^{\,2} / R = \left(V \times DC\right)^2 / R$$

where R is the resistor value (10 Ω), Vres is the voltage drop across the resistor, V is the 5V supply voltage, and DC is duty cycle. Thus, given the above dynamic equation the temperature Tc should respond as a first-order system.

## <u>Caution</u>:

- **Please be very careful while working with the heating element. The heating element may get very hot. Do not touch it as it may burn you.**
- **The heating element carries a large current and can shock a person if touched. Even a 10mA current can cause serious muscle contractions and you may not be able to**

**release the electrified object. Avoid touching the resistor or resistor leads when voltage is applied to it.**

- **The heating element gets very hot. Keep an eye on it to make sure it does not start smoking. This indicates the current is too high and the resistor is failing. Disconnect the circuit immediately if you see smoke or smell something burning.**

## Requirements:

1. Fill in the missing steps for deriving the open-loop transfer function of the thermal system above, assuming zero initial conditions. This transfer function should describe the input-output relationship between the output $\tilde{T}(s)$ (which is the difference between the canister and outside temperatures) and the input $Q_{in}$. **(15 points)**

2. Collect the calibration and step response data from your system. Plot the calibration data vs time (at 0 duty cycle), and the step response data vs time. Recall that the time constant for a first-order system is the time it takes for the output to reach 63.2% of its final steady-state value, relative to its starting point. Calculate and state the time constant for this system. Show the time constant on your step response plot. **(10 points)**

3. Make a plot of the steady-state temperature values as a function of duty cycle, from 10%-70% (in increments of 20%: i.e. 10%, 30%, 50%, and 70%). Extrapolate this curve to zero duty cycle. Does the value at zero duty cycle correspond to what you expect? If not, discuss why. State the control sensitivity for this system, in units of °F / % duty cycle. **(15 points)**