EELE465 Spring 2020 RML April 2, 2020

**Getting Started on Lab6&7**

In Lab6&7 we will add two devices to our I2C bus. The first on is the DS1337 Real Time Clock and the second is the LM92 Temperature Sensor. Please look at each of the datasheets available on D2L.

The DS1337 is very similar to the DS3231 used in Lab 1. Make sure to handle the grain of rice crystal very carefully. You can solder this crystal directly to the IC package or you can solder it to a header and then plug it into your breadboard, DO NOT PLUG the crystal directly into your breadboard as the leads will break. Also the crystal is NOT directional. Remember that the SQW output is an open drain output at requires a 3.3k pullup or something close to that. This pin can be used to make sure your crystal is oscillating and putting out the correct value. You can measure this on your Analog Discovery device. We will not be using the INTA output so this can be left open. Power and ground along with SDA and SCL are all that needs to be hooked up to your system. Application notes app504 and AN505 are both related to using the realtime clock and are provided for you as a resource for the future.

The LM92 Temperature sensor is the second device that you will connect to your system. You will connect power and ground along with SDL and SCL lines. The T_Crit_A alarm line and INT can be left open. There are two address lines that need to be connected to something. From the datasheet this allows you to have up to 4 of these devices on a single bus system each having their own address, I would tie them both to ground. From the top of page 9 of the datasheet the address is then MSB 1001000 LSB. The output data register of the LM92 is described on page 9 under temperature data format and again on page 12. It is 13 bit two's complement word with one bit equal to 0.0625 degrees C. You should get rid of the 3 lowest status bits. You should then be able to calculate easily the temperature in C and then convert to K. When you have this device working you should be able to place your finger on the device and see the readings change (another trick is to use an ice cube in a baggy to watch it get cold).

A **thermoelectric** (TE) **module**, also called a **thermoelectric** cooler or Peltier cooler, is a semiconductor-based electronic component that functions as a small heat pump, moving heat from one side of the device to the other based on the direction of the current flowing thru the device.

Open the TEC document and look at the schematic of the TEC test fixtures which we will be simulating. The 1N5817 diodes are used as fly back diodes when driving the relays with the RFD14N05L mosfets. The way this works is that the relays are being driven by a provided +12 volt supply. Notice that the red and black leads going to the TEC Module are connected to the common connection on all the relays. The Normally Closed relay connection is tied to ground and no current flows thru the TE Module because both red and black are grounded. If we ground say the TEC COOL lead relays LS3A and LS4A will both energize and switch the relay to the Normally open position and the current will begin to flow from the red to black lead in the TEC module making the LM92 side of the heatsink colder. To ground this with a microcontroller we can use a mosfet that is rated for the appropriate current as well as having a logic level threshold for switching that works with our 3.3 volts system. Notice that our mosfets the RFD14N05L will handle 14 Amps with an rdson of 0.1 ohms and a Vgsthreshold of 2 volts.

These are ideal as our TE module draws about 4 amps andwe can switch the mosfet with a 3,3 volts signal from our microcontroller tied to the gate of the device.  The source is grounded and the relay load is applied to the drain.  So when writing a 3.3 volt level signal to the gate we turn on the mosfet which takes the relay to ground and energizes the relay. If we ground say the TEC HEAT lead relays LS1A and LS2A will both energize and switch the relay to the Normally open position and the current will begin to flow from the black to red lead in the TEC module making the LM92 side of the heatsink hotter. Again we can use another mosfet to do this.  So we need to control lines from our microcontroller to turn the TEC module to heat or cool.  The off state is when the relays ae not grounded thru the msofets.  You cannot hurt the test fixture by grounding both sides as no current flows because the red and black leads would be tied to +12 volts.

For this lab we will be simulating the TEC test fixture by using leds on the control lines from your microcontroller.  The led bar graph can be used to do this and you can use two of the microcontroller lines that you used to drive the LED patterns as the ucHeat and ucCool lines.

There is a picture of a hand drawn sketch labelled Lab4&5 that I usually do in class on the D2L website. It has the details on how to wire up the TEC test fixture. We will not be doing this (bummer) but I thought you would like to see the end goal of the class and the details that are needed to make it play. It shows the flyback diodes to protect the mosfets as the relays deenergize.

The key to this is that the LM92 would be mechanically and thermally attached to the AL Block that we are trying to control to a certain temperature.  The LM92 would then give us the actual temperature of our AL block.  Imagine that you have a program that you wrote (Lab7 that we are not doing) that lets you enter a setpoint temperature.  Your circuit and program would then decide if you needed to heat or cool based on the actual temperature you are reading.  Your simple on/off controller would then heat or cool until you got to the set point temperature and then stop.  It would oscillate around the setpoint either heating or cooling as appropriate to maintain the set[point temperature.  Pretty cool!  Of course there are all kinds of control algorithms that you could implement like P, PI, PID or feed forward to help speed up or control the response of the control loop.  They all may require additional hardware and firmware to implement but you get the idea.

There are some draw back to the test fixture in that at some point it may self heat the temperature sensor because the heatsink cannot dissipate all of the heat it needs to,  This is a real thermal problem that combines heat transfer trades offs with stability and capability.