Application Note: Milestone 2

Kyle Rettig and Eric Kryzanekas
Rowan University

December 3, 2018

1 Design Overview

The goal of this project was to use a MSP430F5529 microcontroller to regulate the temperature of a small space. This was achieved by making a voltage divider circuit containing a thermistor. The analog signal produced by this circuit was then converted into a usable digital signal for the microcontroller to then dictate the PWM of a fan that cooled the thermistor. A voltage regulator was also used to heat the thermistor to simulate the increase in temperature of a small space. Using RealTerm, a desired temperature could be set as the input and then the microcontroller would use this temperature and compare it to the existing temperature to dictate the necessary speed of the fan needed.

1.1 Design Features

These are the design features:

- Analog to Digital Conversion (ADC)
- PWM Controlled Fan
- Modify PWM Controlled Fan to Change Speed to Make an Overheating Regulator Reach Certain Temperatures
- Communication through UART and Realterm

1.2 Featured Applications

These are the featured applications:

- Termperature Measurement
- Temperature Control
- Temperature Sensing

1.3 Design Resources

Located at https://github.com/RU09342-F18/introtoembedded-f18-milestone2-team-one

1.4 Block Diagram

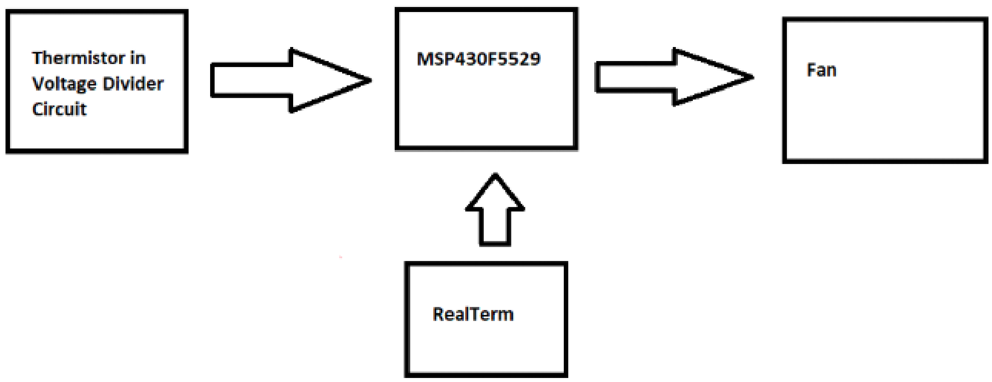

Figure 1: Block Diagram

1.5 Board Image

To be included once Demo is completed

2 Key System Specifications

PARAMETER	SPECIFICATIONS	DETAILS
Board Voltage	5V	
Thermistor Voltage	3.3V	
Fan Voltage	12V	
Regulator Voltage	12V	
Thermistor	NTC LE100E3	Negative Temperature Coefficient

3 System Description

The system is made to simulate a small space that is monitored and cooled when necessary. The Voltage regulator is used to heat up the thermistor. The thermistor is then connected to the microcontroller which powers the fan to cool it down. The system utilizes UART for each block to communicate with each other. The block diagram below shows how each block of the system reacts with each other.

3.1 Detailed Block Diagram

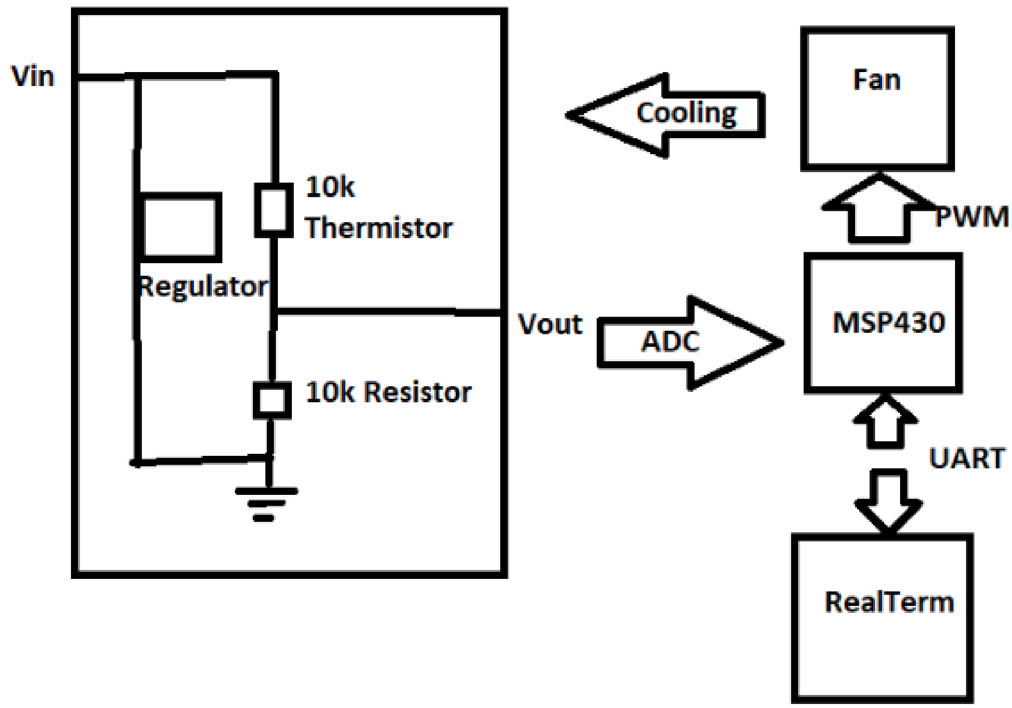

Figure 2: Detailed Block Diagram

3.2 Highlighted Devices

- MSP430F5529: Controls PWM; Sends and receives UART commands; Receives readings from thermistor
- Circuit: Created on breadboard, circuit is connected to the MSP430F5529 and connects all of the devices used together
- NTCLE100E3 Thermistor: R25 value of 10 kOhm
- 4 pin CPU fan
- UA78M33C 5V Regulator

3.3 MSP430F5529

The microcontroller was imperative for several reasons for this project. Its ADC conversion was used to convert the analog signal from the voltage divider and convert it into a usable signal to control the PWM of the fan. The way it controlled the fan was by manipulating pins based on the values of the digital signals and how they compared to the Capture Compare Registers from the initialized timers. The microcontroller was also useful for UART in this project. RealTerm was used to send and receive data through the microcontroller with the TX and RX pins. When data from RealTerm was sent through the TX pin, an interrupt was enabled.

3.4 NTC Thermistor

The thermistor used was a 10k NTC thermistor. NTC stands for Negative Temperature Coefficient which means that the resistance of the thermistor decreases when the temperature increases. This thermistor was used in a voltage divider circuit with a 10k resistor to feed into the ADC input of the microcontroller.

3.5 CPU Fan

This project utilized 3 pins from the fan. The first two being power and ground. Power was connected to a 12V power supply and the ground was connected to a common ground between all the other components. The last pin was the PWM pin that controlled the fans frequency. This pin was connected to the microcontroller and was changed based on the change in temperature.

3.6 Regulator

The voltage regulator was used to heat the thermistor. This was done to simulate a small space increasing in temperature so the resistance in the thermistor would decease, causing the microcontroller to react accordingly by turning on the fan or increasing the PWM of the fan to cool the thermistor.

4 SYSTEM DESIGN THEORY

There are several requirements for this design to ensure that it works properly. First, the voltage divider is needed as it is a great way to utilize the thermistor. This is because a change in temperature will lead to a change in resistance on the thermistor, and that change in resistance will lead to a change in the voltage being put into the ADC port of the microcontroller. This set up causes a favorable solution to transferring a change in temperature into measurable data that can be used by the device. Next, an input voltage of 12V is needed from the power supply to power the fan. Additionally, the regulator needs a voltage of 5V for it to start increasing in temperature. This regulator needs to be connected with several resistors in parallel to draw current so the regulator heats up. Next, BNC cables are required so one can use an oscilloscope to monitor the voltage drop from the thermistor, to ensure that the thermistor is either cooling down or heating up.

4.1 Design Requirement 1: UART Communication

UART is the communication routine used to send and receive data to the MSP430F5529 system. To use this, you have to enable UART in your code, and for this specific project UCA1 was enables with a baud rate of 9600 with the 1 MHz small clock. RX interrupt was also enabled to ensure that the equations in the code would update whenever a new temperature was sent.

How UART in this project works is, when a temperature is sent through RealTerm over UART, the MSP430 detects an interupt, and the interrupt vector is checked and sorted through a case switch statement using the new information. The machine would then use the new data to either change what it is doing, or keep the current setting.

4.2 Design Requirement 2: Analog Digital Conversion (ADC)

This is a series of equations used in the code to determine what the temperature is from the thermistor. As the MSP430F5529 is only able to detects things like resistance, voltage, and current, There is a need to convert that to a useable temperature to make it easier to specify the correct temperature the regulator needs to meet. In this specific scenario, the equation ended up being Resistance = 10000*((4095/Rt)-1), where Rt is thermistor resistance (found in another equation) and Resistance is the circuit total resistance. This tells us the resistance for the circuit, which tells us the temperature when a certain resistance is met.

4.3 Design Requirement 3: Pulse Width Modulation(PWM)/Fan Speed Control

The PWM is a system on the MSP430F5529 to allow change in clock speed for a device. In this case, the CPU fan was directly connected to the PWM, allowing for the CPU fan to slow down or speed up to regulate the temperature of the regulator. This is setup using the CCR0 and CCR1 capture/control Registers along with a clock (smclock) and a timer (Timer A1). What happens to the fan is that the CCR0 contains a certain amount of clock cycles while the CCR1 register contain another number. These two numbers determine how quickly the fan is Pulsed, or turning on and off. If the difference between CCR0 and CCR1 is larger, the fan will pulse for quickly, causing the fan to spin more slowly. This allows for great control of the fan, allowing for accurate temperature control.

5 Getting Started/How to use the device

To use the device, your must use UART on the MSP430F5529. This is accomplished relatively easy with a program called RealTerm. Plug the microcontroller in, start RealTerm, and open the device on the correct USB port at a baud rate of 9600. From there you are able to send numbers to the microcontroller and the device will adjust itself accordingly.

6 Test Setup

One can test this system by using the code and compiling it in code composer while having the microcontroller connected to the computer as well as having the rest of the system set up. Then one has to open RealTerm and input a value which would be the desired temperature. The current temperature values in real time will be displayed on the RealTerm feed. The voltage drop through the thermistor can also be monitored through an oscilloscope to further ensure that the system is working properly.

6.1 Test Data

Will be added after successful demo