Milestone 2: Closed Loop Control

Colby Clark and Ryan Baker
Rowan University

December 3, 2018

1 Design Overview

In this milestone project, an MSP430F5529 launchpad was used to create a closed loop control system. The system's function was to keep a regulator at a certain temperature. During the course of this project, the method for obtaining temperature had to be decided upon. Then, the software had to be configured to accept the desired temperature as an input over UART. As for the outputs, the current temperature had to be transmitted over UART for monitoring and a PWM signal needed to be output to the fan to cool the regulator. By the end of the project, a fully functioning closed loop cooling system had been designed and verified by testing different temperatures and comparing the voltage going into the ADC to the expected voltage.

1.1 Design Features

These are the design features:

- Capable of receiving desired temperature over UART
- Capable of transmitting current temperature over UART
- Capable of controlling fan speed such that the current temperature reaches the desired temperature.

1.2 Featured Applications

These are the featured applications:

- Regulating the temperature of a system with a sensor and a fan

1.3 Design Resources

$$https://github.com/RU09342-F18/introtoembedded-f18-milestone2-magic-from-ti$$

1.4 Block Diagram

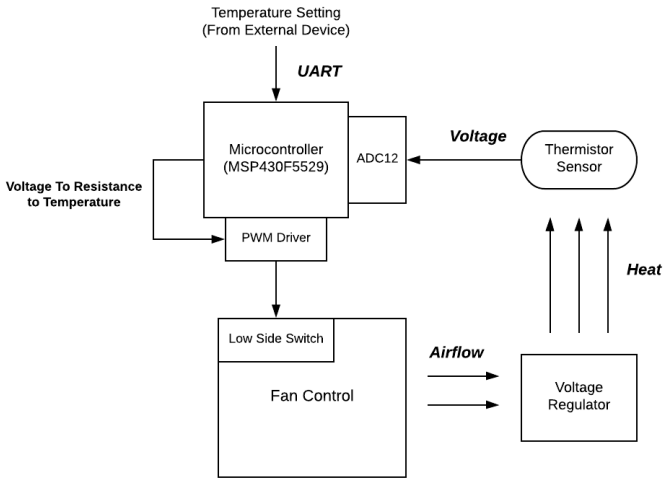

Figure 1: Block Diagram for Closed loop Control

1.5 Board Image

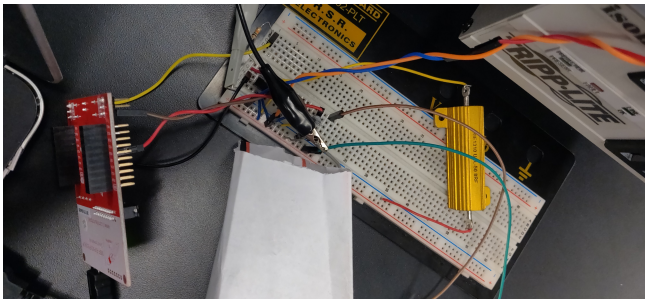

Figure 2: Board Image

2 Key System Specifications

PARAMETER	SPECIFICATIONS	DETAILS
Input Temp.	10 - 99	Specify the desired temperature of the system
Current Temp.	10 - 99	Output the current temperature of the system
PWM Signal	0% - 100%	Output the signal needed for the fan to keep the temperature at the desired temperature

3 System Description

This system is designed to be a closed loop control cooling system. The task is to use an MSP430F5529 launchpad to create a control system that can monitor the temperature of a regulator and adjust a fan as needed to keep the temperature to a desired value.

3.1 Detailed Block Diagram

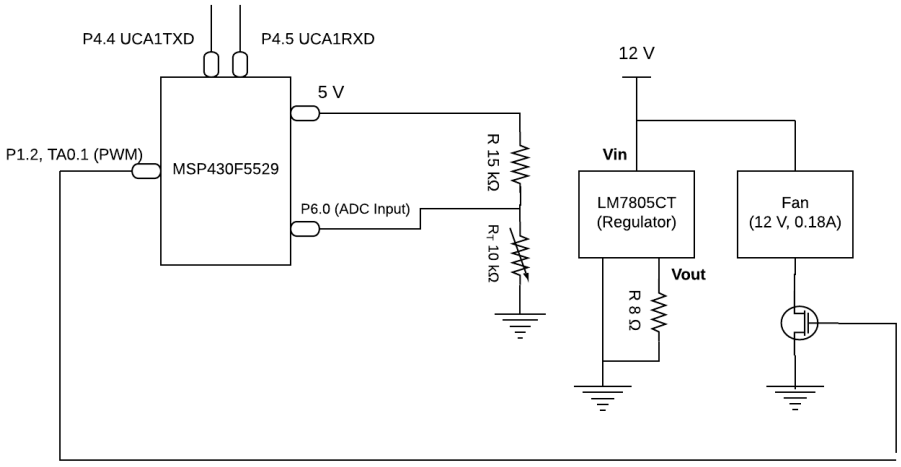

Figure 3: Block Diagram for Closed loop Control

3.2 Highlighted Devices

- MSP430F5529 - This device was used to generate a PWM and monitor temperature through software
- 2N7000 - A NMOS transistor is used in a low-side switch configuration to control the fan.

3.3 MSP430F5529

The MSP430F5529 is used in the system as an ADC (analog to digital converter), a PWM driver, and a UART transmitter/receiver. The ADC is used to convert the voltage across the thermistor to logical bits that can be stored and used to calculate the current temperature of the regulator/heat source. The temperature calculated is then sent out over UART to another device. A temperature setting is also received by the microcontroller over UART in order to generate a PWM signal. The PWM is based on the difference between the current temperature and the desired temperature.

3.4 Low-side Switch

The low-side switch (LSS) is used to control the amount of power supplied to the fan. A PWM is fed to the gate of an NMOS transistor used in the LSS. The transistor's source is connected to ground and the transistor's drain is connected to the output of the fan. The wider the width of the pulse, the more power supplied to the fan.

4 SYSTEM DESIGN THEORY

For this system, the desired temperature is taken as an input over UART. That temperature is then compared to the current temperature. If the desired temperature is higher than or equal to the current temperature, the duty cycle of the PWM signal is set to zero percent. If the desired temperature is lower than the current temperature, the duty cycle is set to fifty percent if its a one degree difference and one hundred percent if its more than a one degree difference. This PWM signal directly affects the speed of the fan which affects the temperature of the regulator.

4.1 Temperature Tolerance

This project required the system to consistently hold a temperature within 3 degrees Celsius of the desired temperature. The voltage (which correspnds to current temperature) can be seen in Fig. 4.

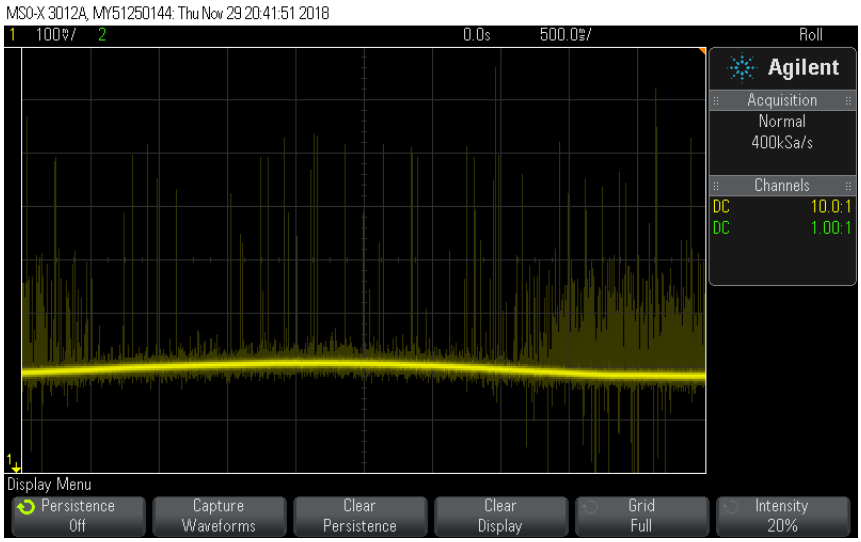

Figure 4: Voltage across Thermistor

4.2 Thermistor

In order to determine the temperature of the regulator, a sensor was made using a thermistor. The sensor consisted of a voltage divider with the thermistor placed at the output. As the thermistor heated up, its resistance dropped down, so the voltage across it changed. By characterizing the relationship between temperature and the resistance of the thermistor, the microcontroller could determine the resistance from the voltage and therefore temperature.

5 Getting Started/How to use the device

To use this device, connect the board to a computer. Let the drivers automatically install then open up device manager and take note of the COM number for the UART connection to the board. Then open realterm. In the display tab, under display as, check unsignedint. Then, in the port tab, switch the baud rate to 9600 and the port to the COM number from device manager, then hit the change button. In the send tab, type the desired temperature using numbers. Then hit send. The system will now try to achieve that temperature.

6 Getting Started Software/Firmware

This section will discuss how to communicate with the device as well as how to specifically program for this device.

6.1 Communicating with the Device

To begin communicating with this device, simply connect it to a computer using the included USB cable. The drivers will automatically install and this will allow for the code to be flashed onto the microcontroller.

6.2 Device Specific Information

To flash code onto the MSP430F5529 launchpad, in specific, follow these steps. Once the board is connected, open code composer studio. Create a new project that uses the MSP430F5529. Copy the provided code into the main.c file or write new code. Debug and let code composer studio flash the code to the launchpad. If there are any updates, let the board update. The device can be used or tested once the code has been flashed.

7 Test Setup

To test this device, connect the board to a computer. Make sure the drivers are installed then check the COM number for the UART connection to the board in device manager. Open realterm to begin testing. In the display tab, under display as, check unsignedint. Then, in the port tab, switch the baud rate to 9600 and the port to the COM number from device manager, then hit the change button. In the send tab, type any temperatures that need to be tested. Then hit send. The system will now try to achieve that temperature. The current temperature can be seen in realterm to verify that the system reaches the desired temperature.

7.1 Test Data

INPUT	OUTPUT
60	60-61
30	31-32
55	55-56

8 Design Files

8.1 Bill of Materials

- MSP430F5529
- 10 kΩ Thermistor
- Fan

- 2N7000 or equivalent