Milestone 2: Temperature Sensor

Karlie Naphy and Anwar Noel
Rowan University

December 3, 2018

1 Design Overview

The following application note covers the usage and functionality of a closed loop system that responds to feedback. The design incorporates a 5 volt regulator, a PTAT, an MSP430, ADC/DAC, UART, and a fan to control the temperature of a PTAT. BAsed upon the desired temperature, the MSP430 listens over UART to receive the temperature reading from the PTAT and adjusts the pulse width modulation of the fan to either cool or let the 5 volt regulator heat the PTAT.

1.1 Design Features

The design of our projects incorporates the following features.

These are the design features:

- PWM
- UART
- Closed Loop functionality
- ADC/DAC

1.2 Featured Applications

- Temperature Regulation
- Pulse Width Modulation
- Analog to Digital Conversion

1.3 Design Resources

GitHub Link
CMD File
C File

1.4 Block Diagram

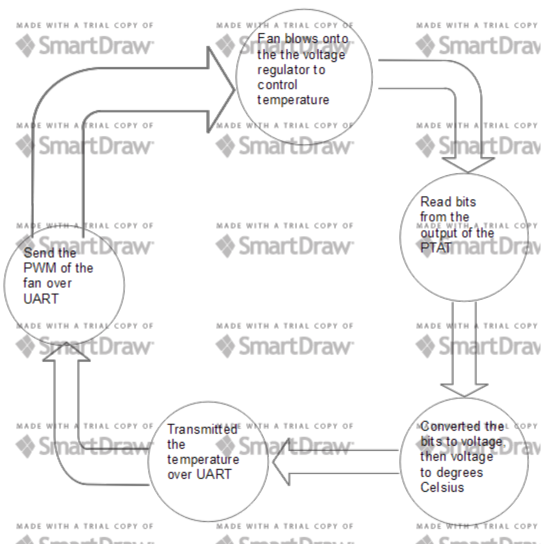

Figure 1: Block Diagram of Temperature Sensor

2 Key System Specifications

Key Specifications	Description
Hot To Cold	Can reduce temperature from 60 degrees Celsius to 20 degrees Celsius
Cold to Hot	Can increase temperature from 20 degrees Celsius to 60 degrees Celsius
Warm to Cold	Can reduce temperature from 40 degrees Celsius to 20 degrees Celsius
Warm to Hot	Can increase temperature from 40 degrees Celsius to 60 degrees Celsius
Temperature Display	Displays current temperature value in degrees Celsius over UART
PWM Receive	Can receive the PWM period for the fan over UART

Table 1: Key Specifications of Milestone 2 Project

3 System Description

The problem trying to be solved is to use the Analog to Digital Conversion portion of the microprocessor. There is really no problem here that is needed to be solved, but the ADC component of the microprocessor is introduced in this milestone. Also, other aspects are used again in this lab including UART and Pulse Width Modulation. All three aspects are incorporated into one lab to show how to use ADC.

3.1 Detailed Block Diagram

.PNG

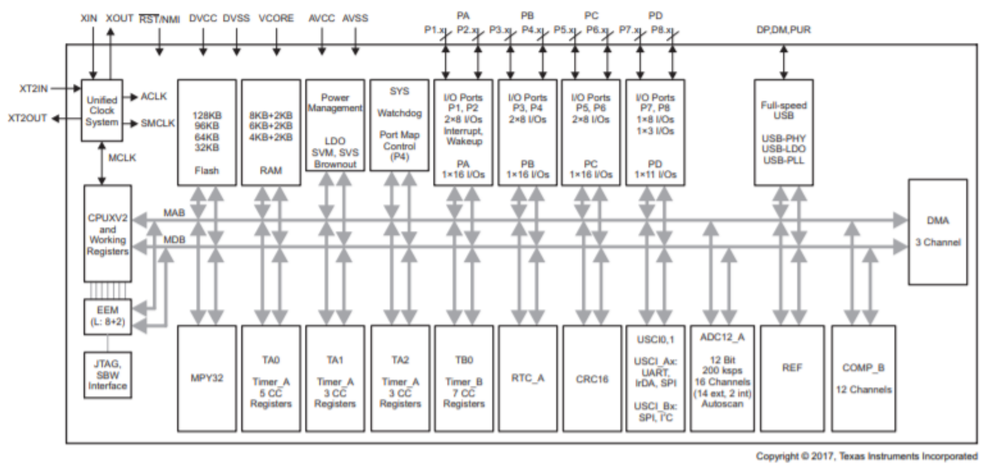

Figure 2: Block Diagram of MSP430F5529

3.2 Highlighted Devices

This just needs to be a bulleted list of what parts you used (not including passive components) with just a quick blurb of what it is doing in your system.

- MSP430F5529
- TI LM60 PTAT
- 5 Volt Regulator
- 12 Volt, 1.4 A Fan

3.3 MSP430F5529

In the temperature sensor circuit used in this milestone project, the MSP4305529 is used to convert the analog output of the PTAT into a digital voltage value, which is then converted into a voltage reading and finally converted into a temperature using the ADC section of the microprocessor. Then, using UART, the MSP430F5529 converts the temperature from an integer to a decimal number in ASCII by converting the temperature value to a ten's place and a one's place. The ten's place is done by dividing the temperature value by 10. The one's place is done by multiplying the value ten's place value by 10 and subtracting that value from the value of the original temperature. Finally, the ten's place and one's place are both transmitted over UART. The PWM of the fan is sent over UART by allowing the compare/capture register 1 of Timer A0 to the values that are received over UART. The fan is set to a hardware PWM, which means that the output of the compare/capture register is set directly to the PWM input of the fan. The PWM then controls the speed of the fan.

3.4 TI LM60 PTAT

The purpose of the PTAT in this circuit is to determine the temperature given off by the voltage regulator. The PTAT makes a physical connection between itself and the voltage regulator, but the two devices are not in the same trace. The PTAT was supplied with 5 volts and had an output at pin 2. The third pin was connected to ground.

3.5 5 Volt Regulator

The purpose of the 5 volt regulator is to supply the heat to the PTAT. The voltage regulator raises in temperature when the load resistor through the third pin is decreased. The issue with increasing the load resistor is that many resistors are not rated high enough to deal with a high wattage circuit. To deal with that issue, many resistors are placed in parallel, which increases their overall wattage rating. The voltage regulator is supplied with 12 volts in pin 1 and grounded at pins 2 and 3.

3.6 12 Volt, 1.4 Amp Fan

The purpose of the fan is to cool the voltage regulator to adjust the temperature read by the PTAT. The fan is connected to a 12 Volt supply and also has a pulse width modulation input which is connected to the hardware PWM output of the MSP430F5529 microprocessor. As the PWM period decreases, the rotation speed of the fan increases, which means that as TACCR1 increases in value, the fan rotates faster. The red wire of the input of the fan is the positive supply, the black wire is the ground supply, and the white wire is the PWM supply. This fan was selected because it was supplied with a PWM supply.

4 SYSTEM DESIGN THEORY

The four main components of our system are the MSP430, the PTAT, the 5 volt regulator, and the 12 volt fan. The system works in a closed loop fashion, receiving input from the PTAT in terms of temperature and adjusting the speed of the fan as an output, resulting in a change in temperature input. Then the process starts again. The PTAT is physically touching the 5 volt regulator in order to heat it up. Using a voltage divider circuit, the MSP430 receives a voltage value over UART that is associated to a certain temperature. The MSP430 converts the voltage to the temperature in Celsius. Based upon the temperature input, the MSP430 will adjust the PWM of the fan as needed.

4.1 Design Requirement 1

One design requirement that was needed some sort of temperature based component. By temperature based component, we mean a component that's output depends on its temperature. There were a few options for the component used, but we decided on the PTAT. The output of the PTAT varies linearly compared to how others vary exponentially. The output of each are both analog voltage levels, but the conversion from voltage to temperature is more complex while using other devices, so the decision to use the PTAT was made.

4.2 Design Requirement 2

The second design requirement was the use of pulse width modulation with the fan. Since the speed of the fan was to be changed depending on which temperature reading is wanted. The two ways this could be accomplished is by the use of a low side switch and powering the fan through the transistor used in the switch. Many of the fans did not consist of a pulse width modulation feature. The fan that was used in this circuit consisted of one. This allowed a direct connection from the fan to the microcontroller which results in more consistent control over the fan.

4.3 Design Requirement 3

The third design requirement was that the output must be a decimal value. To accomplish this, we converted an integer to decimal in ASCII by creating both digits of the number. This is done by dividing the temperature value received by ten and creating the tens place and subtracting the temperature value received by the value of the tens placed multiplied by 10 to receive the value in the ones place. Then each value can be transmitted simultaneously followed by a space to show a decimal value displayed in ASCII.

5 Getting Started/How to use the device

To power the device, connect the USB port to your computer and the other end to the microcontroller kit and run the set program. This programs the microcontroller with the code. The fan is connected to the 12 Volt supply from a power supply, the ground rail, and to P1.2 from the microprocessor's development board. The 5 volt supply from the microprocessor is then set to the first pin of the PTAT, P6.4 from the microprocessor is then connected to the second pin, and the third pin is then connected to the ground rail. The grounds from the microprocessor and the ground of the 12 volt supply are then connected together. Using the software RealTerm, the USB port of the computer is then selected and RealTerm is set to New Duplex mode. This allows you to send and receive values and view the transmitted and received results on the screen. To supply power to the fan, be sure that the power supply is turned on.

6 Getting Started Software/Firmware

To interface with this device using software, the code must be written in Texas Instrument's Code Composer Studio software. This software has the ability to flash the microcontroller with your written code and program the microcontroller accordingly. To test this software, it must be done using the software RealTerm. This is done by setting the display as to ASCII. This allows the user to send any numbers and also receive numbers in decimal. The numbers are set to be changed from decimal to ASCII by creating a tens digit and a ones digit and sending each individually, with each group separated by a space. Following that, the port chosen must be the UART port of the computer. Following that, you can now send values to the microprocessor to change the speed of the fan. The values that are then received back from the microcontroller are the temperature readings calculated by the microprocessor. This calculation is done by taking the analog voltage value received from the PTAT and converting it to bits by multiplying it by the voltage supplied divided by the amount of bits. Next, that value is then multiplied by the degrees Celsius per voltage parameter set by the PTAT. This will result in a temperature reading in degrees Celsius.

6.1 Hierarchy Chart

6.2 Communicating with the Device

In order to communicate with the device, you need RealTerm installed and a USB connection to the microcontroller. Set the software to half-duplex, ASCII, set the baud rate to 9600, and the port to the appropriate port the microcontroller is connected to. Once everything is configured, you can send decimal values to the microcontroller to set as the PWM rate of the fan.

6.3 Device Specific Information

The code was written to be compatible with an MSP430F5529. Pin 4.4 is the TXD output and 4.5 is the RXD input. Pin 6.0 is the ADC input pin that is connected to the output of the PTAT voltage divider circuit.Pin 1.2 is set as the output of TA0.1, the PWM output.

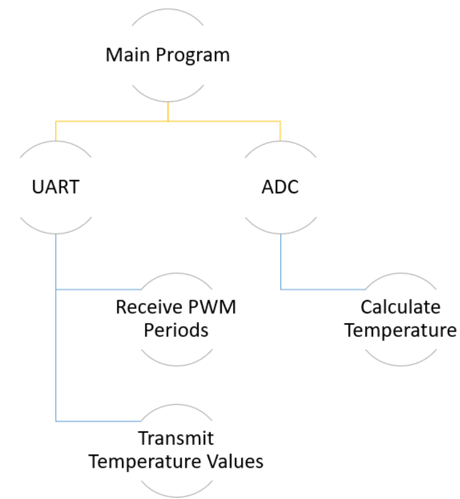

Figure 3: Hierarchy Chart of the Temperature Sensor Software

7 Test Setup

To set up this device for testing, the microprocessor is connected to the computer using the micro usb wire given in the development board kit. The ADC input is then connected to the output of the PTAT along with the input to an oscilloscope. The supply of the fan is connected to the 12 V rail and ground of the and the output of the PWM on the microcontroller is connected to the PWM input of the fan. The PTAT is connected to the 5 volts from the microprocessor and the ground of the microprocessor is connected to the ground of the 12 volt supply. The fan is aimed towards the voltage regulator to manipulate the temperature. RealTerm is opened and set to ASCII and New Duplex mode. The port of the USB is selected and the baud rate is set to 9600. Values can be sent over UART to change the speed of the fan.

7.1 Test Data

The test data received using this project was inaccurate, but a description of what should be received with a working temperature sensor will be explained.

When the temperature changes from hot to cold, the voltage output of the PTAT should decrease significantly due to the decrease in temperature, and the opposite should happen when the temperature increases from cold to hot. The same should happen with any increase or decrease in temperature regardless of variation. The temperature reading displayed over UART should be within 3 degrees Celsius of the desired temperature.

7.2 Schematics

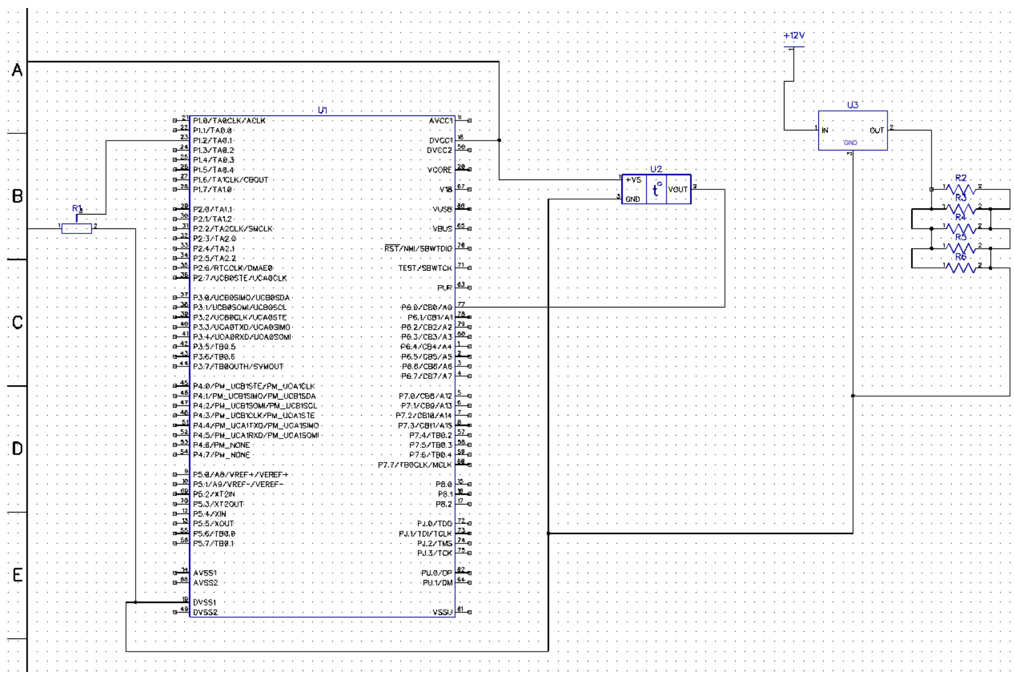

Figure 4: Schematic of Temperature Sensor Circuit

7.3 Bill of Materials

Item	Value	Quantity
MSP430F5529	-	1
LM60 PTAT	-	1
Resistor	100 Ohm	4
Fan	-	1
DC Power Supply	-	1

Table 2: Bill of Materials