ESET 369 M01
Embedded Systems Software
Spring 2024
Multidisciplinary Engineering

Final Project

Project Report
AC System using the AD22100 Temperature Sensor With User Interface Through PUTTY

Author's Name:
Mark Perez

Lab Due Date: May 3, 2024

Lab Submitted Date: May 3, 2024

IS THIS LAB LATE? No

All of the information contained in this report is my own work that I completed as part of this lab assignment. I have not used results or content from any other sources or students.

Description:

In this final project, my goal was to use my newfound knowledge on the concepts that were taught this semester, and apply them to design a prototype for a simple, small-scale air conditioning (AC) system using a temperature sensor and a DC fan. The design would allow for the user to enter a temperature value to cool the system to, in which our “system” is the surrounding area around the temperature sensor itself. Once the user enters a value, the fan would then power on until the temperature sensor reaches the value that the user entered based on a tolerance level set by the program itself, and at that point, the fan will shut off and display the final temperature that it has cooled the system down to based on the user entered value. In creating this project, it will allow me to gain insight as to how air-conditioning systems work, and how to design one on a smaller scale.

Motivation:

The motivation behind the creation of this project was to apply all the knowledge I have attained throughout this course in order to develop a working prototype based on what needed to be accomplished. In terms of the project that I developed, I was able to use various concepts such as UART, ADC, and PWM initialization and utilization all within my C Language program. The programming skills were also combined with my prior knowledge of hardware circuitry in order to connect all hardware components in order to create a working prototype of the design. In completing this project successfully, I was able to fully assess my knowledge on the concepts and apply the knowledge on the concepts I have learned in a hands-on manner that mimics a product we use in our everyday lives, the modern air-conditioning (AC) system.

Materials:

In terms of the hardware materials that were used in the construction of the project, I used the following materials: a MSP432 microcontroller, an AD22100 linear temperature sensor, the L298N motor driver, and a DC Fan. In terms of the software that was used to program the project, the Code Composer Studio (CCS) application was used to create the C Language program that would allow for our project to be functional as well as the PUTTY application that was used to communicate with the user and display the results of the temperature sensor. The MSP432 microcontroller was used as the “brain” of our project since our program would be uploaded to the device in order for the microcontroller to receive and transmit data throughout the hardware components, The AD22100 linear temperature sensor was used to read the temperature of the surroundings, which transmitted a voltage signal to the microcontroller based on the temperature reading. Lastly, the motor driver and DC fan work hand in hand since the motor driver receives a signal and amplifies it to a voltage level that would have the ability to power the DC fan. All of these components work together in order to accomplish the final goal of having a mock AC system that can cool to a temperature given by the user.

Methods:

In terms of the methods discussed in the course that were used in order to develop this project, the project consisted of using UART communication, Analog-to-Digital Conversions (ADCs), and Pulse Width Modulation (PWM). The combination of each of these software components within our program will allow for us to communicate with the user, obtain data from the sensor, and transmit data to control the fan.

Within the project, Universal Asynchronous Receiver-Transmitter (UART) Communication was used alongside the PUTTY application in order to obtain data from the user of the project as well as transmit data to the user via the communication portal. In our program, the UART communication protocol is used to display information to the user, in which it asks the user to enter a temperature value in Fahrenheit to cool the system to, and it uses the same protocol in order to read in the temperature value the user entered through PUTTY. Once this has been done, the last portion that uses the UART communication protocol is through the sending an updated current temperature to the PUTTY terminal every three seconds until the system has been cooled to the desired temperature.

Along with UART communication, the Analog-to-Digital Converter found within our MSP432 microcontroller was also used in order to obtain the analog voltage data from the sensor and convert it to digital data that can be processed by the computer. In terms of using the ADC on the microcontroller, the registers of the ADC had to be configured according to what we wanted to accomplish, and the values were chosen by using the technical data sheet for the microcontroller. This feature was used to obtain an initial value from the sensor, recurring measurements as it was called within the loop, and a final measurement when the data value is within a tolerance of the user inputted value.

The last major method that was used in the development of this program was Pulse Width Modulation (PWM), which was used to send data to the motor driver and control the DC Fan. Pulse width modulation refers to the act of altering the duty cycle, which is the ratio of on versus off time within a certain period of time. In using pulse width modulation, we are able to successfully send signals to the motor driver that would allow the DC fan to rotate accordingly based on the signals we are meaning to send to the motor driver. Without the use of pulse width modulation, we would not be able to send an on and off signal to the corresponding input on the motor driver in order to allow for the motors to rotate.

Results:

In terms of results, we can discuss the overall workings of the program that was developed along with how the system works as a whole once both the software and hardware portions had been implemented. In terms of the program itself, we can split the program into two main areas, which are the main code as well as the functions that were created. The main functions that were created in this program mainly deal with the initialization of the three main methods that had been discussed above, which included the use of UART, ADCs, and PWMs. Each of these functions are used to initialize how these concepts will be used in regards to what we want to accomplish. In terms of the main source code of our program, we first start by calling each of these functions in order to initialize them, and then display to the user via PUTTY a statement asking for them to enter a value to cool the system to in Fahrenheit. Once the user enters a value through PUTTY, the program will convert this value from a string to a decimal value that can be used for calculations within our program. Then, the ADC is used to obtain voltage values, which are converted to temperature values via the formula given in the data sheet since it is a linear sensor. The program then goes into a loop to constantly check the value from the temperature sensor against the value entered by the user in order to stop the fan once the sensor temperature has reached the value entered by the user based on a one percent tolerance. Also found within this loop is a statement that prints the current temperature value of the system every three seconds in order to update the user with its current temperature. Lastly, once the system has reached the desired temperature, a final statement is printed to the user regarding the final value, and the entire program is essentially reset, in which it now asks the user to enter a new value to cool the system to.

Conclusion:

All in all, the completion of this final project has allowed me to not only gain insight into how the concepts that I have learned can be used within programming, but also how they can be used within the creation of products that I encounter in my everyday life. This particular project tied together concepts from the course such as programming the microcontroller using UART, ADC, and PWM, and also tied in my knowledge of hardware circuitry in order to bring the idea to life. Seeing that this final project has summed up the course concepts while also bringing in concepts from real life, we can see how these projects are monumental to gaining experience that we can use in the real world once working in the engineering industry.

Along with having the positive attributes of allowing me to attain real world applications of the learned concepts, there were also many other positive outcomes from this project and areas where this project idea can be expanded. Some major positive attributes to this project include the fact that it taught me how to overcome obstacles I am faced with along with allowing me to develop my own ideas into a project while working individually. Since I am currently the only student enrolled in this course, the final project was based on my own ideas, and was solely completed by myself with help from my professor. Although this was very daunting at first, it was also very rewarding when I was able to see the final product of all the hard work I had put in. Although I was able to successfully implement my ideas into a working design, I was faced with various challenges that needed to be overcome in order for the successful completion of this project, which opened my eyes to the importance of asking questions.

Appendix:

Below you will find the links to several documents that were quintessential to completing this project, which include the data sheets to hardware components such as the MSP432 microcontroller, the AD22100 temperature sensor, and the L298N motor driver. Along with the data sheets that were used to gain insight into the hardware components, you will also find the link to the program I developed in order to create this small-scale AC system. Lastly, you will find the link to a video showing the final testing of the system once it has been completed.

MSP432 Microcontroller Data Sheet
AD22100 Temperature Sensor Data Sheet
L298N Motor Driver Data Sheet
Small Scale AC System C Language Program
Small Scale AC System Final Implementation Video

Resources:

In the following links found below, you will find some of the resources that were used in order to gain some background knowledge on how to design a small-scale AC system such as the one that was developed for this final project.

Small Scale AC System Example via Instructables