COMP/EMEC 462 Embedded Systems
Final Project Report
(Fall, 2021)

It's Too Early!

Department of Computer Science
California State University, Channel Islands

Prepared by
Roger Lorelli

I. Problem Statement

It's Too Early began as an attempt to meet the needs of a friend with issues sleeping, providing them with an automated curtain control system. The goal was to have a system that could be configured through an internet interface, be manipulated through a remote control, and, through use of sensors, sense the light level of the room to determine an action.

II. Proposed Solution

To accomplish this goal, the system broken into three parts; the main unit, a transmitter and a webserver. The main unit is responsible for the sensors, receivers (IR and wireless), motor direction control, and the logic needed to determine when to engage the motor. The webserver is a paired with a wireless transmitter to send threshold updates to the main unit when the internet interface is used.

When in operation, the main unit starts in an open state and in automated mode. The mode can be changed using an IR remote, switching between automated and remote. When in either mode the unit is to first check if there are any updated values from the webserver. If the mode is set to remote, the system will idle until a signal is sent by the remote control to either open the curtains (>>|), close the curtains (|<<), or to switch the system to automated (PWR). If the system is set to automated then the system will use the photoresistors to sample the light levels inside and outside the room, comparing the values to the upper and lower threshold values. If both the outside and inside values are greater than the upper threshold the curtains will be closed. The condition to open the curtains is for the outside light level to be less than the lower threshold.

The webserver-transmitter pair hosts the webpage that keeps a history of the of the values provided to the system and allows for the user to input an upper and lower threshold. Once the values are submitted, a PHP script parses the request and calls a Python script to communicate the values to the transmitter by serial communication. The transmitter parses the serial message and translates the values for wireless transmission to the main unit.

III. Development

The development of the system started with the main unit, based around an Arduino Mega2560 board with the following components: 4 photoresistors in 2 pairs (inside vs outside sensors each providing an average value), an IR receiver paired with an IR remote for remote control, an nRF24L01 chip for wireless communication with the webserver, and a 3-6V DC motor controlled by a LD293D IC Motor Control Driver for manipulating the curtains. An external power supply, 9V battery, was connected to the LD293D to supply power to the motor. The coding libraries used for the different components are: IR remote, nRF24L01. These libraries were chosen for the simple fact of they were the first found that could accomplish the assigned tasks.

The webserver-transmitter unit was the second portion of the system to be developed. This portion is made-up of a webserver hosted on a Raspberry Pi 3B+, utilizing Nginx to act as the webserver framework with a combination of PHP and Python3 to communicate the information retrieved from the web-interface to the transmitter using a USB cable. The library used for the serial protocols was pyserial. The transmitter is constructed using an Arduino Uno with the second nRF24L01 wireless chip attached to allow communication with the main unit. The only non-standard libraries used by the Arduino was the nRF24L01 library.

The main tools used during development were programming IDEs (Integrated Development Environments), text editors, and internet guides on learning about the new components, HTML and PHP. The IDEs used JetBrains PyCharm and Arduino IDE 2.0. The text editors used were Sublime Text, GNU Nano command line text editor for Linux, and Notepad++. Guides will be listed in the references section.

Changes were made from the initial project proposal, primarily with the exclusion of an internal timer-based trigger because of the increased complexity of the trigger interactions. A second major change was the inclusion of an Arduino Uno to act as an intermediary between the Raspberry Pi webserver and the Arduino Mega2560. During development the libraries for

the nRF24L01 wireless transceiver chip I could find for Python did not communicate in a reliable fashion (i.e., the Mega would receive garbage values). To get around this issue I used the same library as the Mega for the Uno and linked the Uno to the webserver via USB serial communication.

There were a few minor changes to the initial proposal that are of note. During testing the number of photo resistors was increased to account for the possibility of outside variables such as partial shading or cloud coverage. An issue arose with the motor used where the torque generated when the curtain was drawn open would drop, making the motor stall. To combat this, a second external power supply, 9V battery, was added in parallel to the first in order to double the current going to the motor.

IV. Conclusion

Originally built to be able to manipulate full size bedroom blackout curtains, a proof-of-concept prototype had to be built due to an underpowered motor being used. While the demo model of the system uses an underpowered motor, the rest of the system (main unit and webserver) are fully functional as intended. The system makes use of a web-interface for configuration, uses more than a single external sensor (photoresistor, IR receiver and Wireless transceiver), is a combination of Raspberry Pi and Arduino interactions, and uses an actuator (DC motor). Some of these components were the same as used in previous labs but as many were components that were new, requiring research into their usage and requirements.

This project also required a fair bit of improvisation on my part in the realm of mechanical engineering. While not necessarily polished, and probably woefully inefficient, the system as it is works. Future plans for the system, once I buy my own kits to replace what is being returned, will include an LCD display for the main unit to show the current sensor values, bi-directional wireless communication to update the webpage with current sensor values, multi-stage motor control so that curtains could be closed in stages instead of all or nothing, and a more powerful IR remote/receiver since the current versions lose a cohesive signal after 9 inches.

REFERENCES

https://www.php.net/manual/en/tutorial.php

https://circuitdigest.com/microcontroller-projects/wireless-rf-communication-between-arduino-and-raspberry-pi-using-nrf24l01

https://www.makerguides.com/ir-receiver-remote-arduino-tutorial/

APPENDICES

https://github.com/rogerlorelli663/It-s-Too-Early