– ECE 09.321: Final Project –

Fingerprint Duo Authentication

Brian Harlow and David Gaffney
Rowan University

December 22, 2018

1 Design Overview

In this project, an MSP430F5529 microcontroller was used to interact between multiple devices to create a duo authentication service based off of fingerprint verification to open a lock. The system begins by scanning a finger print. If the finger print is verified, a notification is sent to a user's phone via an MQTT server. This phone user can then decide whether or not to allow the person who scanned their finger access to the device that a servo motor unlocks. The servo motor will only unlock the device when the owner (the phone user) allows it.

1.1 Design Features

This design features the following:

- MQTT communication between Microcontroller through ESP8266 Wifi Module and User
- Arduino Uno controlled Fingerprint Scanner GT-511C1R at a Baud rate of 9600
- Serial UART communication at a Baud rate of 115200 between ESP8266 WiFi Module and microcontroller
- Hardware PWM integration to control door lock simulated by servo motor SG-5010
- Implementation on MSP430F5529 microcontroller

1.2 Featured Applications

A list of different technologies this system could be applied to:

- Secure facilities where security must confirm identification of workers.

- Homeowner or rental property double security for children without keys/guests who needs access, etc.
- Miscellaneous double security measures.

1.3 Design Resources

For all relevant resources and files, visit the linked repository:

The Study of Scrumbo Repository

1.4 Block Diagram

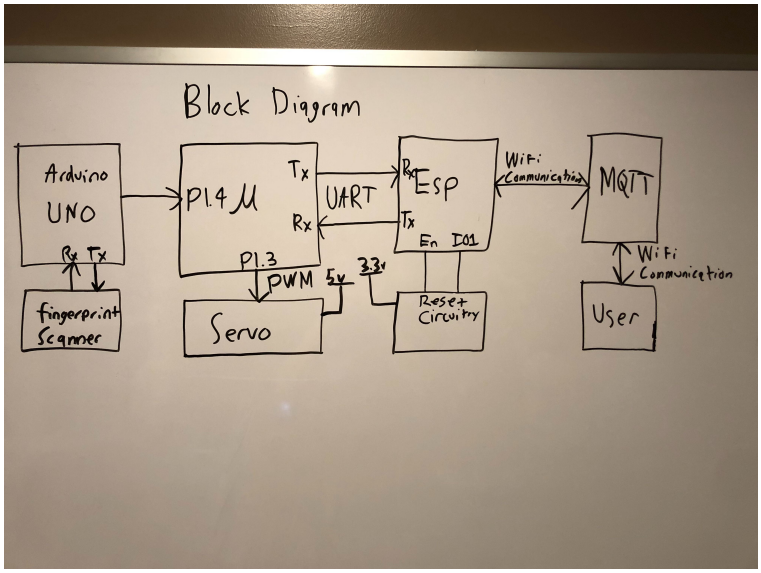

Figure 1: Simple diagram of The Duo Authentication Service

1.5 Board Image

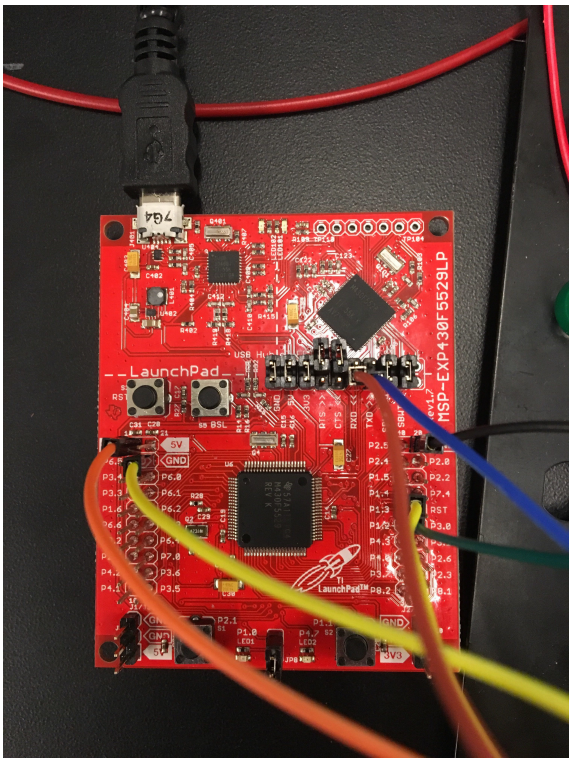

Figure 2: Layout of MSP Device

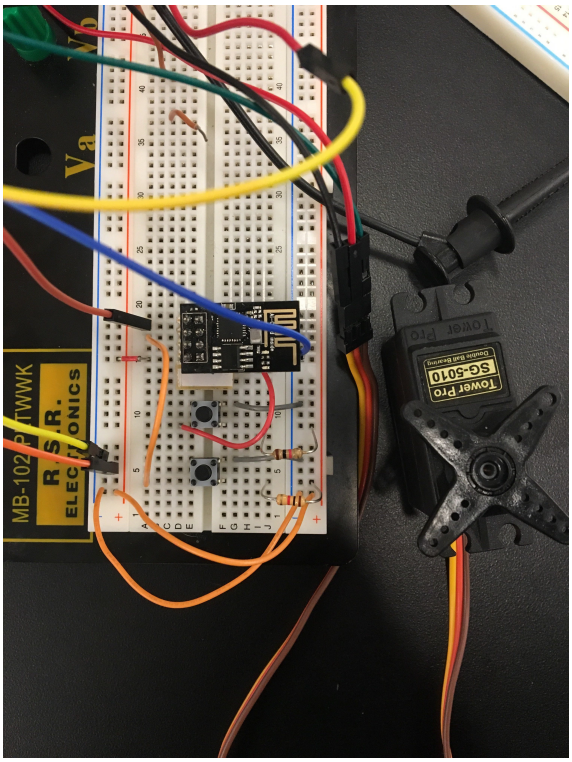

Figure 3: Layout of ESP Device and Servo

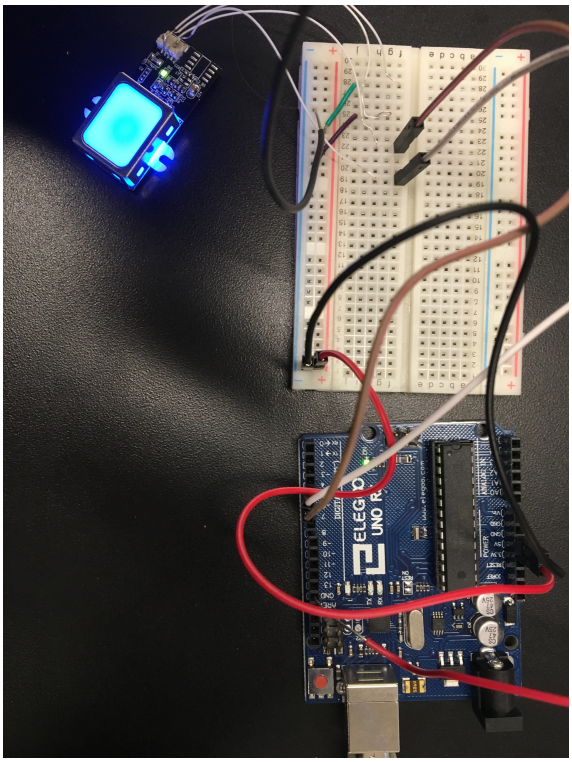

Figure 4: Arduino UNO Layout with Fingerprint Scanner

2 Key System Specifications

See the table on Page 6.

3 System Description

This system is designed to create a duo-security system where a fingerprint is scanned and verified, and a user inputs whether the lock should be opened or closed over an app. It utilizes an Arduino UNO to communicate to the GT-511C1R Fingerprint Scanner, and send a verification bit of data to the MSP430F5529 which is then sent over a UART Serial Communication connection to the ESP8266 WiFi Module.

3.1 Detailed Block Diagram

The ”User” block is an app on a mobile device that the user utilizes to send and receive data to and from the system. The data to the user is whether the fingerprint is verified, and the data from the user is a 1 or 0, which designates whether the lock opens or

Parameter	Specification	Description
Voltage	5V Power Supply	5V power input to the servo motor
Baud Rate	Data Transmission Frequency	The baud rate is the rate at which data is transferred between two UART devices. If the baud rates are not the same frequency, the data will not transfer properly and there will be timing issues.
RX/TX Buffer	Data In/Out 8 Bit Registers	The RX Buffer is a register which stores a byte of data to be taken in by the microcontroller. The TX Buffer stores the data to be transferred to the user
ESP8266	Secondary Microcontroller	The ESP8266 is a WiFi module which wirelessly communicates with an MQTT server hosted remotely. It allows for data transfer to and from the server.
Fingerprint Sensor	Fingerprint Verification	Sensor used to store and scan fingerprints. If the finger being scanned matches a stored fingerprint, the finger will be verified.
Arduino UNO	Secondary Microcontroller	An additional microcontroller used to operate and interact with the fingerprint sensor.
Servo Motor	PWM Controlled Servo Motor	Servo Motor whose rotational position is determined by a PWM pulse width.

closes. This data is sent using an MQTT server which is connected to by the ESP8266 WiFi module and the user's phone using WiFi communication. The ESP then sends that data to the microcontroller to be used to open the lock using a PWM signal to the servo. The fingerprint scanner block sends data to the Arduino UNO which sends whether the fingerprint is valid to the microcontroller which is then sent to the ESP to be shown to the user using the MQTT server. The circuitry connected to the ESP is used to set the ESP in bootloader mode which is used to change the settings on the device.

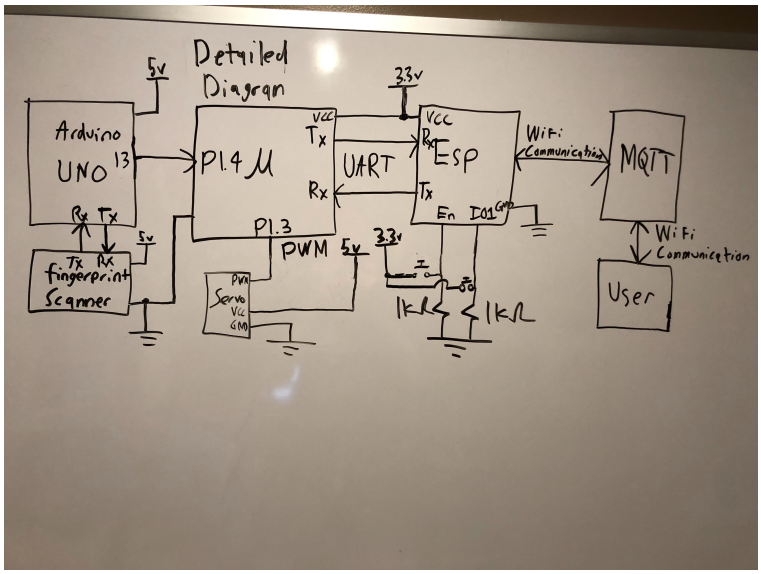

Figure 5: A detailed diagram of the Duo Authentication Service

3.2 Highlighted Devices

Essential components list:

- MSP430F5529
- ESP8266 WiFi Module
- GT-511C1R Fingerprint Sensor
- SG-5010 Servo Motor

3.3 MSP430F5529

The device being used is the MSP430F5529. Inside of the processor, a couple of different components were used: the timer module and its corresponding CCRs, UART

communication with RX and TX buffers, and the GPIO peripheral. The timer A module was used to control the duty cycle of the PWM which allowed for changing the rotational position of the servo motor. The UART protocol was used to allow communication between the microcontroller and the ESP8266, as well as between the Arduino UNO and the fingerprint sensor. Finally, the GPIO peripheral was used to receive a verification signal from the Arduino UNO on the MSP430F5529 to notify the main microcontroller when a fingerprint was verified.

3.4 ESP8266 WiFi Module

The ESP8266 device is used to communicate with a MQTT server being hosted on a Raspberry Pi, which is shown on a WiFi network. This device utilizes Arduino code that connects to the MQTT server over the WiFi network. It regularly pings the server to make sure it is connected, and is sent responses from the server. The programmer then sends an array of characters that spell out the syntax to certain commands. These commands are used to subscribe and unsubscribe to certain topics, publish to certain topics, and read messages from subscribed topics. These commands are commented in the Arduino code uploaded on the Github repository. This device is a very powerful and yet simple way to implement WiFi communication for multiple devices.

3.5 GT-511C1R Fingerprint Sensor

The fingerprint sensor is ran using Arduino example code to ”enroll” a fingerprint, or save it as a valid fingerprint. Example code is also used to check if the finger being scanned is a valid fingerprint. The fingerprint sensor communicates with the Arduino UNO via a UART interface. Code was added to the example code to configure a GPIO pin on the Arduino UNO to send a pulse to the MSP430F5529 when a fingerprint has been verified.

3.6 SG-5010 Servo Motor

The servo uses 5V. It utilizes PWM created by the microprocessor to determine the angle of how it rotates. This PWM is created using a Timer on the MSP.

4 SYSTEM DESIGN THEORY

The main components of this project that allow the system to work are the WiFi communication, UART communication and the Servo PWM. These components are used in specific ways that are described in sections 4.1, 4.2, and 4.3.

4.1 Servo Pulse Width Modulation Control

The Pulse Width Modulation Control was used to create the duty cycle that controls the rotational position of the servo motor. The servo motor required a PWM frequency of 50 Hz. The motor would turn for any duty cycle between 3 percent and 12 percent. Pin 1.3 on the microcontroller was set to output the reset/set signal of the PWM and was used as the data input to the servo motor. This is the signal that actually allowed the servo motor to interpret the PWM and rotate accordingly.

4.2 UART Communication

The serial UART communication protocol interface implemented in the MSP430F5529 was essential for relaying information from the user to the system and vice versa. The microcontroller was configured to receive and send data to and from the ESP8266. The data sent was whether the fingerprint scan was verified, and the data received was a 1 or a 0. Those values defined whether the lock would open or shut. UART was also used to communicate from the fingerprint scanner to the Arduino UNO. This data was used for internally verifying the profiles.

4.3 WiFi Communication

WiFi communication was used in order to send data from the device to the user. This was necessary in order for the entire project to function. This wireless component was the main portion of the project, which was to seperate the user and the fingerprint scanner spacially. The ESP8266 was used to send data to the MQTT server which then is shown to the user. The user then is able to connect to the MQTT server, see that information, and send information back to the device. The ESP8266 is a very versitile device because it can be used with devices other than Arduino devices. This allows for better development and more uses for the device.

5 Getting Started/How to use the device

In order to set up this project, first the microcontroller is plugged into a USB power source, and connected to the different devices. Pin 1.4 is connected to pin 13 of the Arduino UNO, which is connected to power using the USB cable, and is connected to the Fingerprint scanner using TX and RX. The scanner is also plugged into 5V and GND. Pin 1.3 on the MSP device is then connected to the servo motor's signal wire, and then connected to 5V and GND as well. The MSP is then connected via TX and RX to the ESP8266. The ESP device is then connected to 3.3V and GND, and connected to the button reset circuitry displayed in Figure 5. Once this is completed, make sure to load your ESP into bootload mode by toggling the reset buttons, and flash the device using TX and RX of the MSP device connected to TX and RX respectively of the ESP device. Make sure the two far right jumpers are taken off. Then flash the MSP by reconnecting the jumpers, and reconnecting TX and RX of the MSP

device to RX and TX of the ESP device. Once this is done, the MQTT should show an ”ON!” message for the topic TestTopic2, and you can begin testing.

6 Test Setup

Connect the device as described in section 5, and pictured in Figure 2, 3, and 4.