# AUTOMATIC CLOTH FOLDING MACHINE

By

Xudong Li
Anran Su
Suicheng Zhan

Final Report for ECE 445, Senior Design, Spring 2017

TA: Yuchen He

3 May 2017

Project No. 43

## Abstract

The purpose of this project is to make a cost-effective cloth folding machine that could automatically fold the clothes when a piece of laundry is put on the machine. The photo sensors on the board detect the presence of the clothes while the passive infrared sensor detects the movement of human hands and arms. The light-emitting diode on the folding board indicates the state of the folding process. After testing, the cloth folding machine is able to detect the presence of the cloth and automatically fold them in a nice way.

# Contents

# 1. Introduction

## 1.1 Motivation

Washers and dryers have become so commonplace that people do not think of them as new concepts. Since Hamilton Smith patented the rotary washing machine in 1858, our ways to deal with laundry have not changed for almost 160 years. For many people, the worst part of doing laundry is having to fold all the clothes once they come out of the dryer. This activity could be tedious and time-consuming. Therefore, some people just dump their laundry into the closet without organizing them. This behavior often leaves a mess and gives trouble when people are finding their clothes.

In order to address the issues stated above, we have built a cost effective folding machine that could automatically detect and fold the clothes. The operation of the machine requires little human involvement, which is significantly useful for people who are not willing to organize their clothes.

## 1.2 Functions

Our design of the folding machine consists of four lithium batteries, four plastic boards, three servo motors, two photosensors, one light-emitting diode (LED), one passive infrared (PIR) sensor, a wooden frame and several printed circuit boards (PCBs). The four lithium batteries powers electrical components of the project. The four plastic boards fold the cloth, and the servo motors provide the torque to fold. The two photosensors are responsible for cloth detection, and the PIR sensor is for hands and arms detection. The LED is user interface, which will indicate the current operating status of the machine. At last, the PCBs control the operation of the machine, and the wooden frame supports all the components.

The operation of the cloth folding machine is autonomous. It can detect the presence of the cloth and then check for the obstacles. If there is nothing but the cloth on the plastic boards, then the folding procedure initiates. The machine will rotate the plastic boards in a specific order until the cloth becomes a neat rectangle.

The folding machine can handle various types of clothes regardless color, size and material. It is also cost effective to build, operate and maintain without compromising robustness. We also implemented two protection mechanisms to ensure the safety of the users. The first mechanism is hands and arms detection. As long as user’s hands and arms are on the plastic board, the folding procedure will not initiate. The other one is the obstruction feedback of the servo motors. When any object hinders the rotation of the plastic boards, the control unit receives an obstruction signal and then disconnects the power of all servo motors.

## 1.3 Block Diagram

The overall design project consists of five blocks: power supply, control unit, sensing unit, user interface and mechanical components.

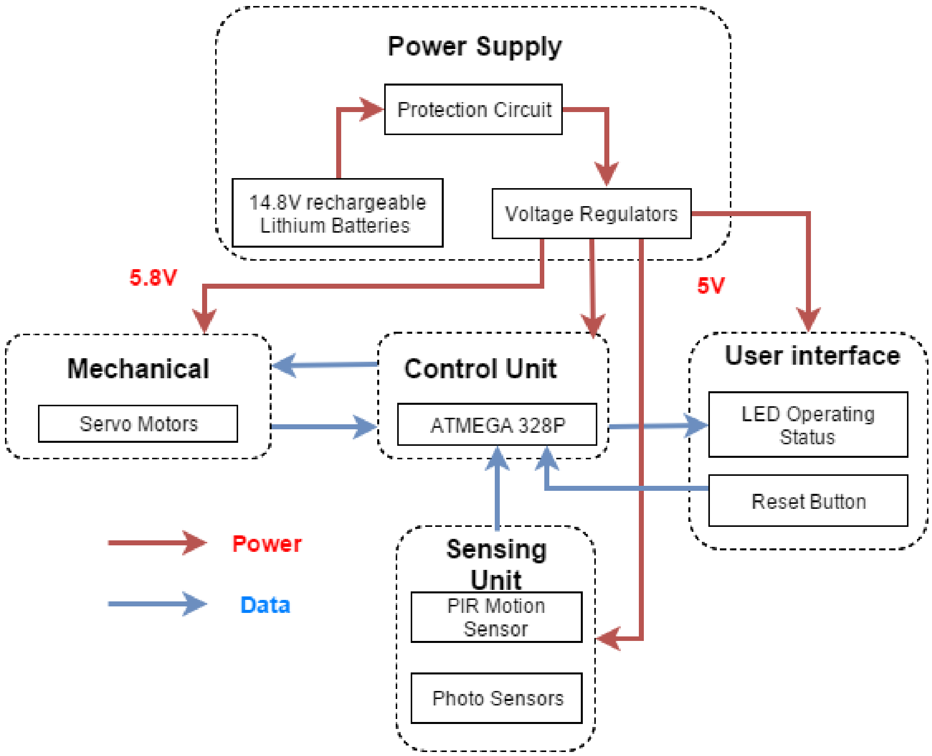


Figure 1: Block Diagram

### 1.3.1 Power Supply

The power supply block consists of four 3.7 V rechargeable lithium-ion batteries, two voltage regulators and a protection circuit. The lithium-ion batteries are the power source of the project. The protection circuit has both under-voltage and over-voltage protection to handle the over-charge and over-discharge of the battery. The first type of voltage regulators step down the battery voltage to 5.8 V to supply the servo motors; while the second type of voltage regulator steps down the battery voltage to 5 V to supply the control unit, sensing unit and user interface.

### 1.3.2 Sensing Unit

The sensing unit consists of two photosensors and one PIR sensor. The photosenors are used for cloth detection when a piece of cloth is placed on the machine, while the PIR sensor checks for the user's hands and arms in the board area. Both sensors were specifically customized to meet the requirements of this project.

### 1.3.3 User Interface
The user interface includes a RGB LED and a reset button. The LED can change colors to indicate the operating status of the folding machine for the user. The reset button enables the user to reset the current operation when the machine enters an error state.

### 1.3.4 Mechanical
The mechanical unit consists of three servo motors, which provide high torques to flip the plastic boards during the folding process. We opened each servo motor and soldered an extra wire on the motor driver to get the feedback signal.

### 1.3.5 Control Unit
The control unit is a single ATMEGA 328P chip, which includes a flash memory that stores the operation code. This chip can interpret signals from other blocks and send corresponding commands to each module. The software code is written in Arduino programming language.

# 2. Design

## 2.1 Physical Design

Our physical design is based on existing folding board on the market. The ECE machine shop split the board into boards A, B, C and D and made three custom hinges with extended rod between the boards. These rods are attached to the rotation axis of the servos. When the servos rotate, they also rotate the rods and hinges therefore flipping the boards. Two photosensors denoted as PS1 and PS2 are mounted on board A and D to detect presence of clothes. One PIR motion sensor is mounted on top of motor 1 to detect human hands and arms motion within the board area, and one RGB LED is mounted on top of motor 2 to indicate current state of operation. The folding board is lifted by 10 cm with a wooden board beneath the folding board. All electrical components such as power and reset button, batteries and PCBs are taped beneath the wooden board.

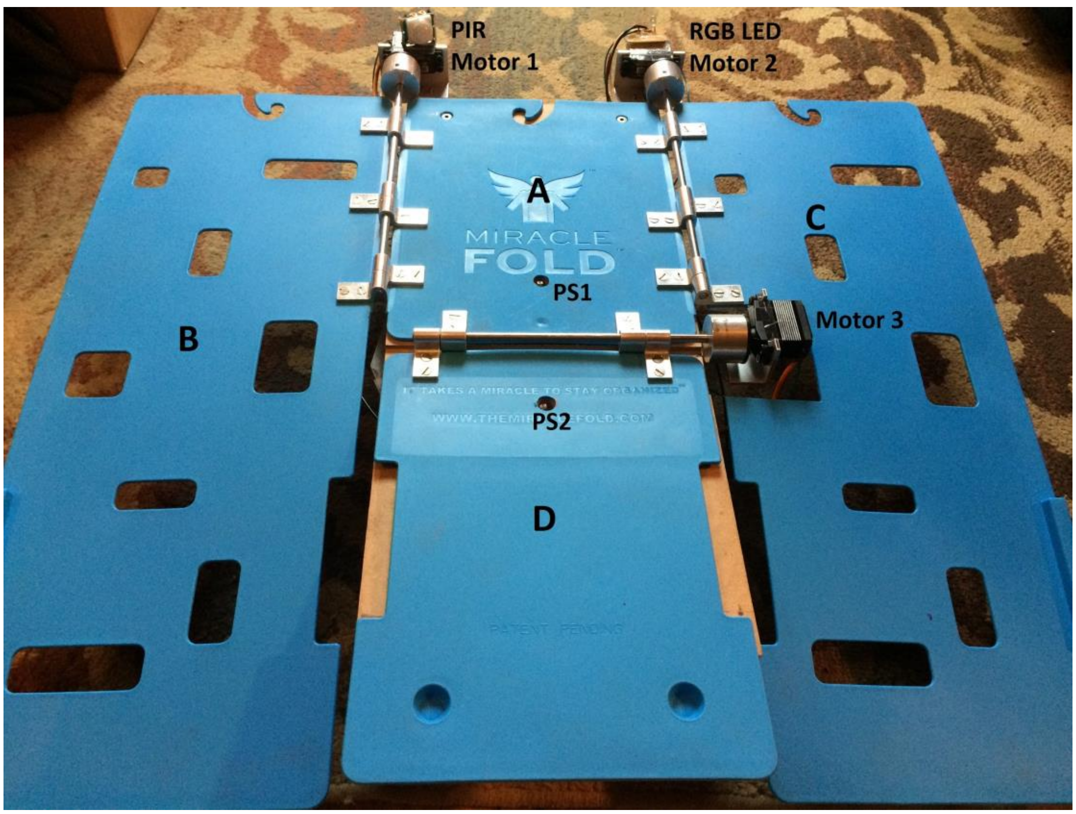


Figure 2: Physical Design

## 2.2 Power Supply Design

In our project, we use a series-combined rechargeable Lithium battery on the market which provides 14.8 V DC voltage. The output from the lithium battery is connected to our custom made protection and regulator circuits before passing into the control unit, sensing unit, mechanical unit, and user interface.

### 2.2.1 Protection Circuit

For the protection circuit, we use the chip LTC4365 [1] to be our battery circuit protection chip. The chip prevents the battery from over discharge by disconnecting the circuit. When the battery provides a voltage less than 4.7 V, it will disconnect the supply voltage source and therefore prevent the motor from burning.

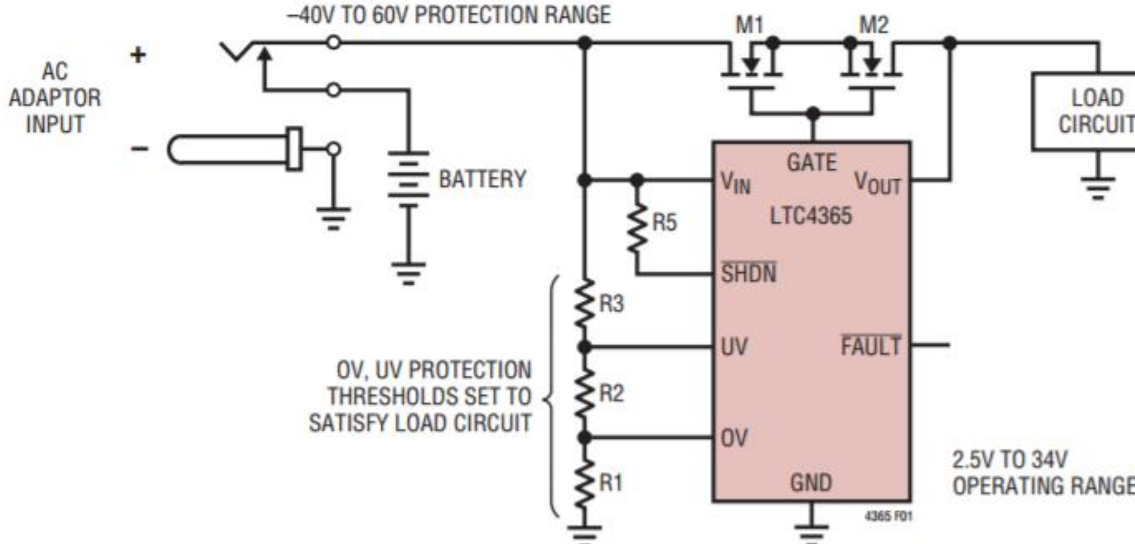


Figure 3: Protection Circuit Schematic [1]

$$R_3 = \frac{V_{OS(UV)}}{I_{UV}} * \frac{UV_{TH} - 0.5V}{0.5V} \quad (2.1)$$

$$R_1 = \frac{(V_{OS(UV)}/I_{UV}) + R_3}{OV_{TH}} * 0.5 \quad (2.2)$$

$$R_2 = \frac{V_{OS(UV)}}{I_{UV}} - R_1 \quad (2.3)$$

According to equations [1] 2.1, 2.2 and 2.3, the under-voltage $V_{OS(UV)}$ offset voltage, which we chose to be 0.3 mV, $UV_{TH}$ is the under-voltage threshold voltage, which we chose to be 4.5 V. $OV_{TH}$ is the overvoltage threshold voltage, which we chose to be 16 V. The value of $I_{UV}$ is typically 10 nA and the value of R5 is typically 510 Ω from the datasheet [1].

Therefore, we calculated that R3 = 240 kΩ, R2 = 2.156 kΩ, R1 = 8.437 kΩ.

### 2.2.2 Voltage Regulators

We use the IC chip LM317 [2] to build four pieces of low-dropout voltage regulators. One of the voltage regulator, which is used to drive the microcontroller, photo sensors, PIR sensor, regulates the output voltage to be 5 V. The other three voltage regulators, which are used to drive the three servo motors, regulates the output voltage to be 5.8 V. The input voltage of the regulators is 14.8 V, which is provided by the output of protection circuit.

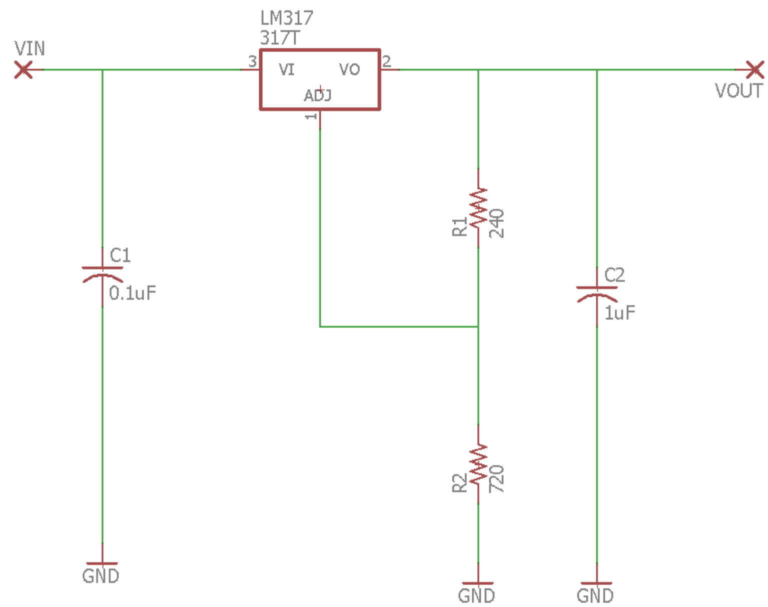


Figure 4: Schematic of 5 V Voltage Regulator

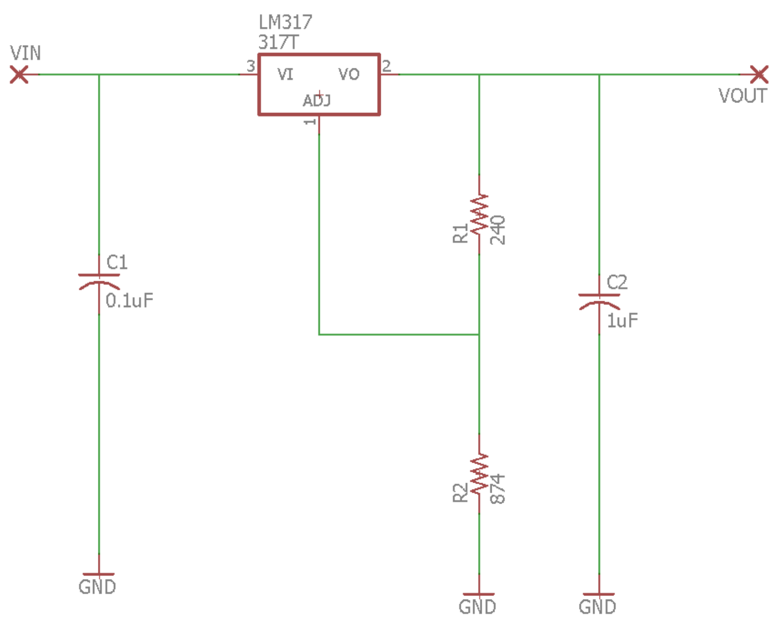


Figure 5: Schematic of 5.8 V Voltage Regulator

Calculation:
We chose the value of R1 to be 240 Ω.
For the regulator that provides 5 V output voltage:

$$V_{OUT} = V_{ref} * (1 + \frac{R_2}{R_1}) \quad (2.4)$$

$$5V = 1.25V * \left(1 + \frac{R_2}{240\Omega}\right) \quad (2.5)$$

$$R_2 = 720\Omega \quad (2.6)$$

For the regulator that provides 5.8 V output voltage:

$$V_{OUT} = V_{ref} * (1 + \frac{R_2}{R_1}) \quad (2.7)$$

$$5.8V = 1.25V * (1 + \frac{R_2}{240\Omega}) \quad (2.8)$$

$$R_2 = 874\Omega \quad (2.9)$$

In order to reduce the ripples in the output voltage and make it more stable, we added two coupling capacitors C1 and C2, which hold the value of 0.1 µF and 1 µF.

## 2.3 Sensing Unit Design

### 2.3.1 Cloth Sensing

We implemented the cloth sensing function using simple photo resistor and regular resistor and voltage divider rule. Figure 6 is the schematic of our cloth sensing circuit:

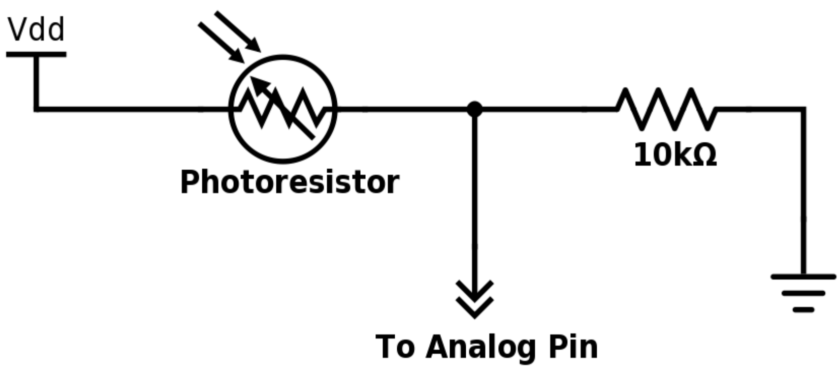


Figure 6: Cloth Sensing Schematic

Working principle of the cloth sensing circuit is simple. Without any cloth covering the photoresistors, they produce relatively higher voltage values back to the Arduino analog input pins. When they are covered, the cloth will block out most of the light previously received by the photosensors. Then photosensors can produce relatively lower voltage values back to the microcontroller analog input pins. In the microcontroller software program, we then made use of these high and low voltage values to achieve cloth sensing.

### 2.3.2 Obstruction Sensing

In the original design review, we planned to use ultrasonic sensor to detect hands and arms in the board area. The advantage of ultrasonic sensor is that it can be programmed to ignore objects detected outside the range of interest (70 cm in our case). But as we progressed, we discovered two crucial disadvantages of ultrasonic sensor. First is that it does not properly detect human hands and arms outside of approximately 50 cm. Because the surface area of human hands and arms “seen” by ultrasonic sensor is not large enough to reflect ultrasonic wave back to the sensor. Second it can be easily interfered by irrelevant objects on the folding board such as hinges and rods.

Therefore, we explored possibility of replacing ultrasonic sensor with passive infrared (PIR) motion sensor. The advantage of PIR motion sensor is that it will not be interfered by irrelevant objects other than human hands and arms as it only detects the infrared emitted by human hands and arms motion. However, the disadvantages of PIR motion sensor are first its detection range cannot be limited to 70 cm which is our range of interest, therefore it could be interfered by human motion outside the range of our interest. Second it cannot detect hands and arms that are not moving. Third it takes one minute to stabilize every time it is powered up.

We weighed the pros and cons of ultrasonic sensor and PIR motion sensor and decided to opt for PIR motion sensor for hands and arms sensing. The connection of PIR motion sensor requires no additional PCB. The use of PIR is shown in figure 7:

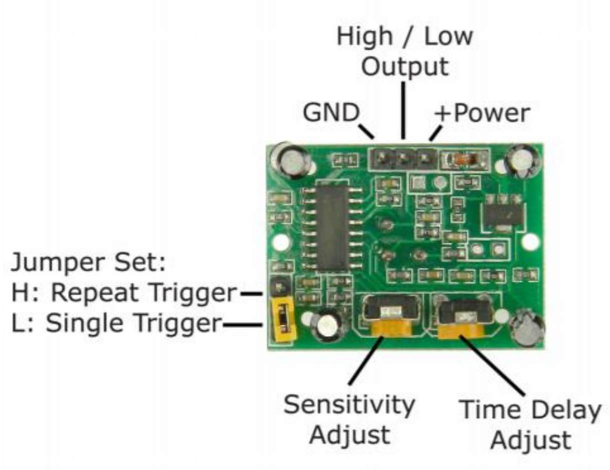


Figure 7: PIR Motion Sensor PCB[3]

When the PIR senses hands and arms motion, it outputs a digital high of 3.3 V and outputs a digital low of 0 V when no motion detected. The time delay adjust is used to set how long the output remain high before it outputs low again. For our application, it is set to roughly 4 seconds. The trigger jumper set works as follow: in single trigger mode, sensor outputs high when motion detected, after the set delay time (4 seconds) has passed, it automatically outputs low regardless of whether continuing motion is present. In repeat trigger mode, sensor outputs high when motion detected, remains high if continuing motion is present, outputs low after motion has ended and the set delay time has passed. For our application, the repeat trigger is chosen because we want the PIR sensor keep outputting high if user’s hands and arms are still moving in the board area. The sensitivity adjust is set to be minimum range of 3 meters for our application.

## 2.4 User Interface Design

### 2.4.1 RGB LED

The RGB LED is responsible indicating the current state of operation to the user according to table 1:

Table 1: LED Status

| State Number | Status | LED Output |
|---|---|---|
| 1 | IDLE | White |
| 2 | Clothes Ready | Blue |
| 3 | Folding | Red |
| 4 | Error | Blinking Red |
| 5 | Folding Completed | Green |

Figure 8 is the circuit schematic of our RGB LED:

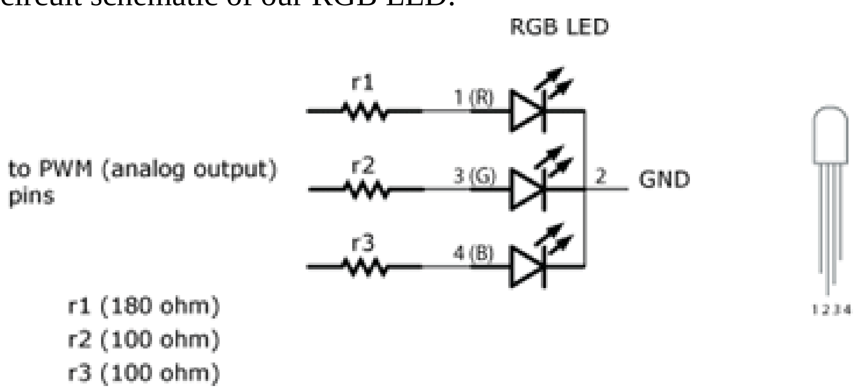


Figure 8: RGB LED Circuit Diagram [4]

We implemented the schematic in figure 8 using perf board. The logical control of RGB LED is explained in the control unit design section.

### 2.4.2 Reset Button

The reset button is responsible for resetting the folding board back to initial idle state after obstruction in folding state has caused the error state. It also provides a method for the user to interrupt the operation for any reason. It needs to be able to pull the reset pin of ATMEGA328P to ground to achieve program resetting purpose. Figure 9 is the schematic of reset button.

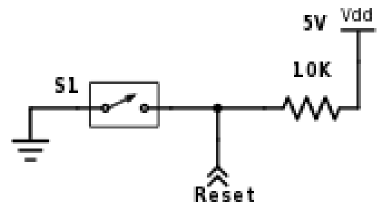


Figure 9: Reset Button Schematic

## 2.5 Mechanical Design

The mechanical block of our project consists of only high torque servo motors. One of the most critical task we faced in this project is to achieve position and obstruction feedbacks from the servo motors back to the microcontroller. Obstruction feedbacks are useful feature because we want the servo motors to be able to stop rotating when obstructed to prevent themselves from burning. And we want the entire project to enter an error state where all operations are stopped. Normal servo motor has built-in position potentiometer inside the housing. It is a close loop system itself but an open loop system with respect to microcontroller as illustrated in figure 10:

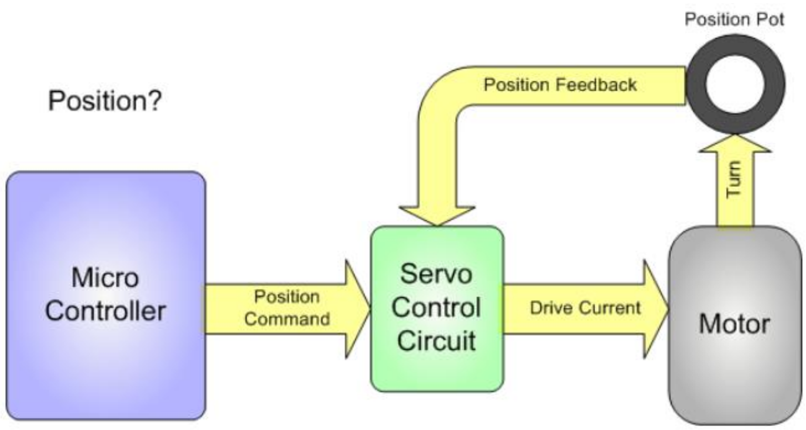


Figure 10: Non-Feedback Servo Motor [5]

In order to close this feedback loop, we opened a high torque servo motor housing, found the position potentiometer pin, soldered an additional analog feedback wire onto that pin and routed this wire back to microcontroller analog input pin. The green wire shown in figure 10 is the feedback wire:

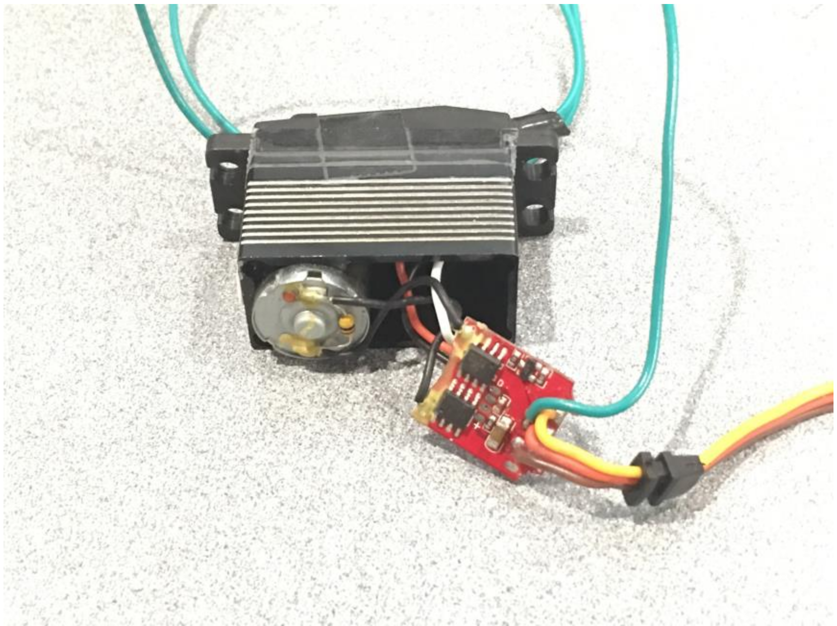

Figure 11: Servo Motor Feedback Modification

Now with the feedback wire, we closed this feedback loop as illustrated in figure 12:

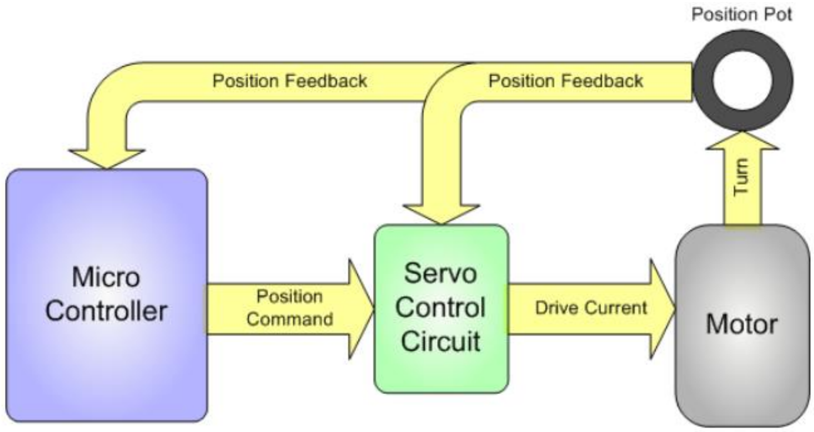


Figure 12: Feedback Servo Motor [5]

## 2.6 Control Unit Design

The core of the project is the control unit which consists of a programmable microcontroller ATMEGA328P. It processes data from all components and send commands accordingly for proper operation. Its design can be better explained with operation flowchart and code snippets.

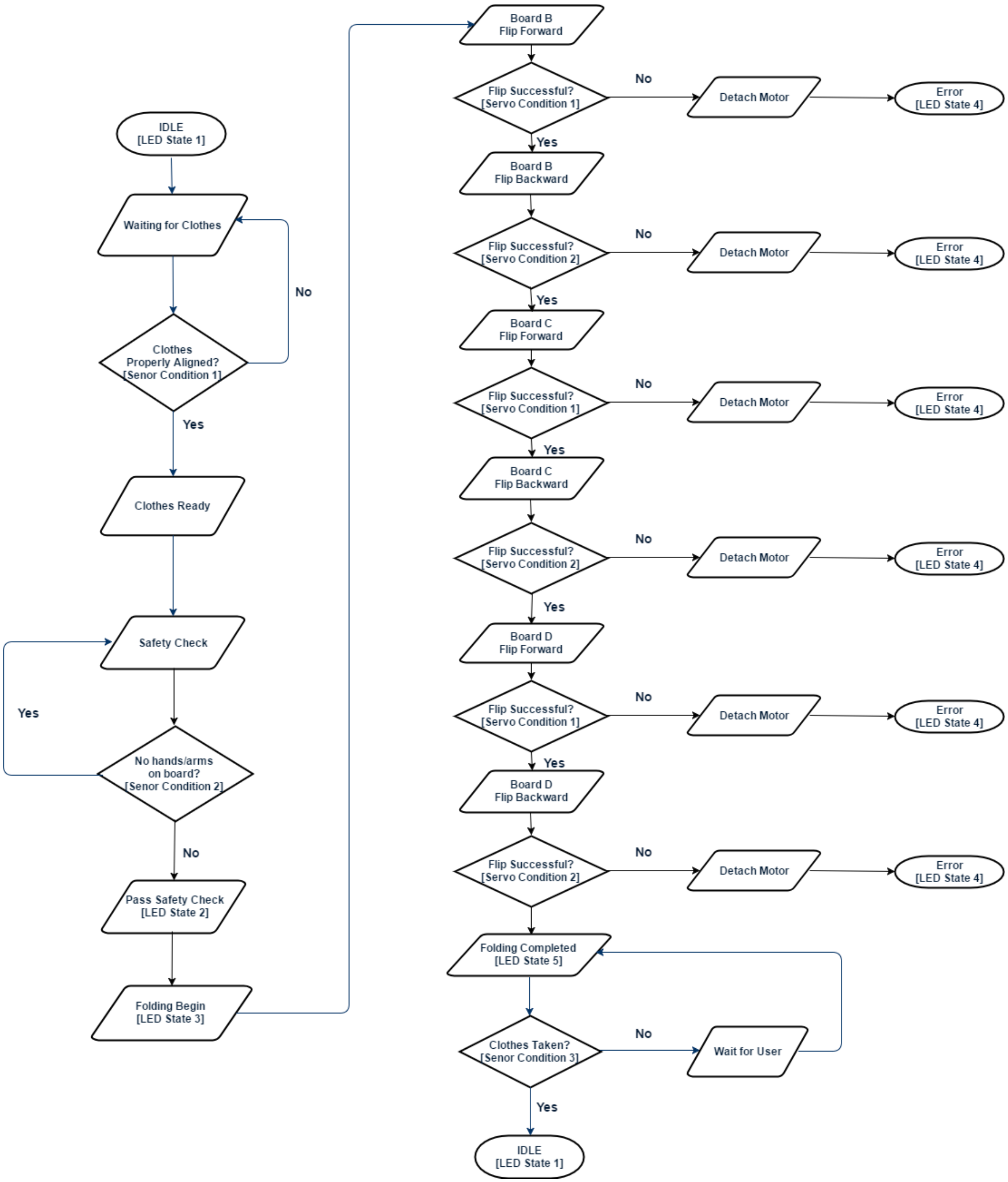


Figure 13: Operation Flowchart

Tables 2 and 3 list the conditions for the states in Figure 13.

Table 2: Sensor Conditions for Figure 13 Flowchart

| States | Name | Photosensor 1 | Photosensor 2 | PIR Sensor |
|---|---|---|---|---|
| **Sensor Condition 1** | Cloth Presence | 1 | 1 | X |
| **Sensor Condition 2** | No Hands | 1 | 1 | 0 |
| **Sensor Condition 3** | Cloth Taken | 0 | 0 | X |

Table 3: Servo Conditions for Figure 13 Flowchart

| States | Direction | Target | Success if Achieve | Fail if not achieve in |
|---|---|---|---|---|
| **Servo Condition 1** | Flip Forward | 180° | 170°-180° | 1 second |
| **Servo Condition 2** | Flip Backward | 0° | 0°-10° | 1 second |

Figure 14 shows the global variables used by all functions:

```
//all global variables here
//sensors definition
int PIR=7;
int PS1=A1;
int PS2=A2;
bool cloth,hands;

//LED definition
int RED=11;
int GREEN=5;
int BLUE=6;

//motors definition
Servo servo1,servo2,servo3;
int servo1Pin=9;
int servo1feed=A0;

int servo2Pin=10;
int servo2feed=A3;

int servo3Pin=3;
int servo3feed=A4;

int minFeed = 45;  //0 degree
int maxFeed = 567; //170 degree
```

Figure 14: Global Variable Initialization

### 2.6.1 Achieving Cloth Sensing

For detecting cloth, we implemented two helper functions called cloth_put() and cloth_taken() as shown in figure 15:

```
//return true if there is cloth on board. Return false otherwise
bool cloth_put(){
    double voltage1 = map(analogRead(PS1),0,1023,0,5);
    double voltage2 = map(analogRead(PS2),0,1023,0,5);
    if (voltage1<=3.0 && voltage2<=3.0) return true;
    else return false;
}

//return true if cloth taken off the board after folding. Return false otherwise
bool cloth_taken(){
  double voltage1 = map(analogRead(PS1),0,1023,0,5);
  double voltage2 = map(analogRead(PS2),0,1023,0,5);
   if (voltage1>=3.0 && voltage2>=3.0) return true;
   else return false;
}
```

Figure 15: Implementation of Cloth Sensing

We experimented and found out after the photoresistors are covered with a piece of cloth, they output a voltage below at least 3.0 V back to the microcontroller.

The ATMEGA328P analog to digital converter works as follow: it divides voltage range of 0 to 5.0 V into 1024 steps, with digital 0 mapped to 0 V and digital 1023 mapped to 5.0 V. The Arduino built-in analogRead() function reads the analog voltage value and converts it into a digital value between 0 and 1023. The Arduino built-in map(A,B,C,D,E) function maps parameter B to parameter D, maps parameter C to parameter E and maps parameter A to a value between parameters D and E. In this case, the map() function maps analogRead() to an analog voltage value between 0 V and 5.0 V and returns it.

### 2.6.2 Achieving Hands and Arms Sensing

For achieving hands and arms sensing we implemented one helper function called hands_detect() as shown in figure 16:

```
//return false if no hands detected on board. return true otherwise
bool hands_detect(){
  if(digitalRead(PIR)==HIGH) return true;
  else return false;
}
```

Figure 16: Implementation of Hands and Arms Sensing

### 2.6.3 Controlling RGB LED

The software code that controls the color of our RGB LED is straightforward as illustrated in figure 17:

```
//LED Status IDLE - White
analogWrite(RED,255);
analogWrite(GREEN,255);
analogWrite(BLUE,255);
```

Figure 17: RGB LED Color Control

We looked up the RGB values for the colors of interest and changed these values accordingly. Blinking red color error state is achieved by a conditional infinite while loop. The program will be trapped in the while loop until the user press the reset button:

```
int servo1pos = getPos(servo1feed);

while(servo1pos>20){ //enter obstructed error state
  servo1.detach();
  digitalWrite(RED,LOW);
  digitalWrite(GREEN,LOW);
  digitalWrite(BLUE,LOW);
  delay(500);
  analogWrite(RED,255);
  analogWrite(GREEN,0);
  analogWrite(BLUE,0);
  delay(500);
}
```

Figure 18: Blinking Red LED Error State

The getPos() function is a helper function which will be explained in details later.

### 2.6.4 Controlling the Servo Motors

We implemented two helper functions for realizing motor position feedback control:

```
void calibrate(Servo myservo, int analogPin, int minDegree, int maxDegree){
  myservo.write(minDegree);
  delay(1500);
  minFeed = analogRead(analogPin);

  myservo.write(maxDegree);
  delay(1500);
  maxFeed = analogRead(analogPin);
}

//returns the current position of motor to microcontroller
int getPos(int analogPin){
  return map(analogRead(analogPin),minFeed,maxFeed,0,170);
}
```

Figure 19: Motor Feedback Code Snippet [5]

We only ran the calibrate function in the early stage of our design to obtain the digital values of minFeed (which corresponds to motor position of 0°) and maxFeed (which corresponds to motor position of 170°). The raw output of servo motor feedback pin is an analog voltage and the feedback pins are connected to ATMEGA328P analog input pins. The analogRead() function converts analog voltage into digital value. With minFeed and maxFeed values known, we can obtain the current position of our servo motor by calling the getPos() function. This function takes in the digital value of feedback pin and returns a position between 0° and 170°. With the position in degree known, the program can determine if the folding board has reached designated position or has been obstructed.

# 3. Design Verifications

Appendix A shows the complete requirements and verifications. In this chapter, we mainly present quantitative result of verification.

## 3.1 Power Supply System

### 3.1.1 Circuit Protection

We connected the input of the protection circuit to a DC power supply, the output of it to the oscilloscope, and swept the input voltage from 18 V to 0 V with the step of 0.5 V. The output voltage dropped to nearly 0V when the input voltage was below 4.5 V and above 16 V.

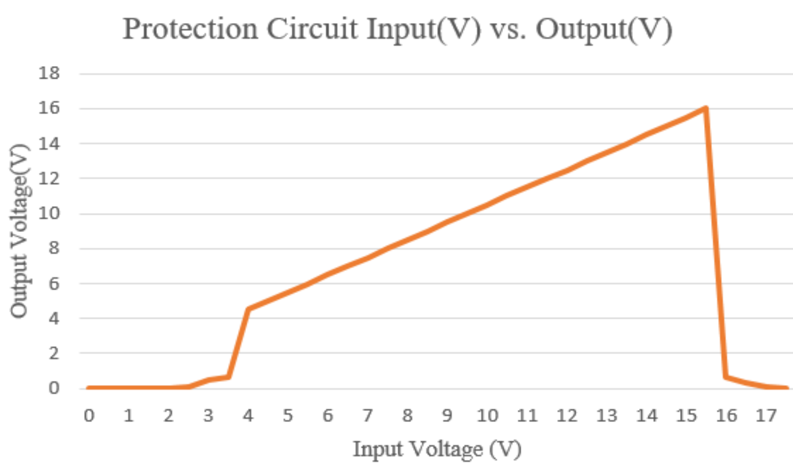


Figure 20: Protection Circuit Input Voltage vs. Output Voltage

### 3.1.2 Voltage regulator

We connect the input of the 5 V voltage regulator to a DC power supply and then swept the input voltage from 10 to 18 V, we verified the output voltage of the regulator connected to the oscilloscope was 5 V within ± 0.1 V. We repeated the same process with the 5.8 V voltage regulator, then the output voltage of the regulator connected to the oscilloscope was 5.8 V within ± 0.1 V.

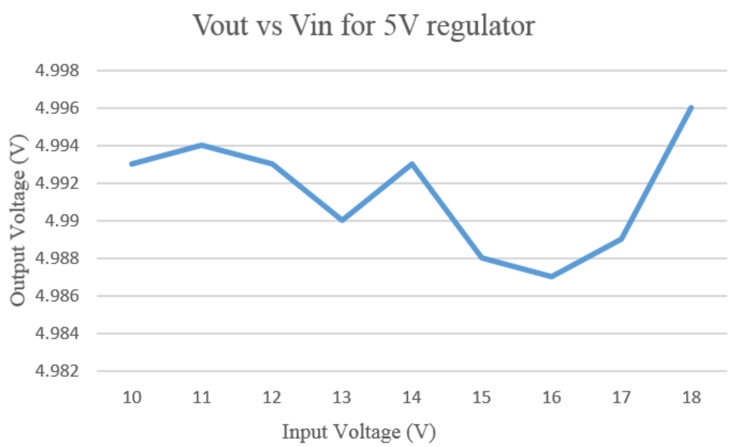


Figure 21: The graph of input voltage vs Output voltage for 5 V voltage regulator

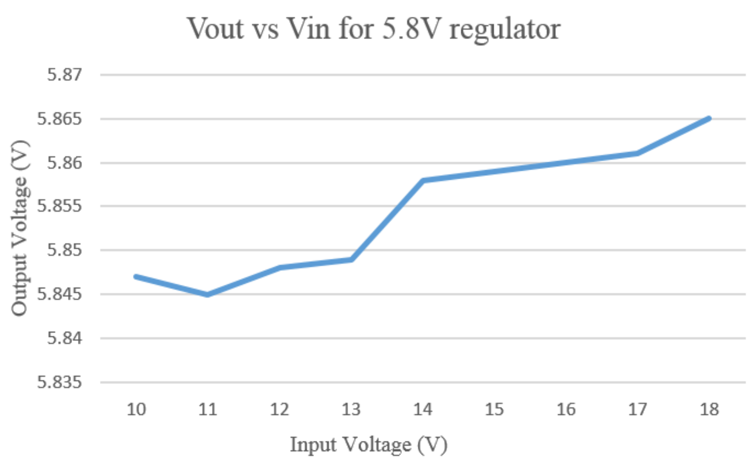


Figure 22: The graph of input voltage vs Output voltage for 5.8 V voltage regulator

### 3.1.3 Battery

We performed a current drain test by connecting a 10 kΩ resistor to the battery to check the stability of the output voltage.

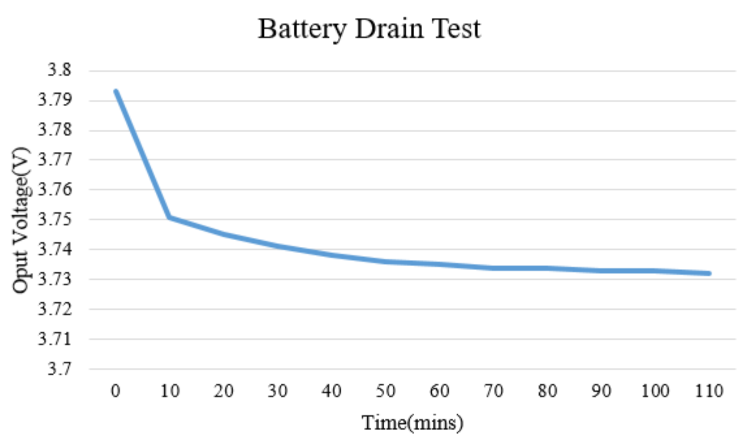


Figure 23: The graph of Time vs Output Voltage for Lithium-ion Battery

## 3.2 Sensing Unit

### 3.2.1 Photosensor

The photosensor output voltage must be able to drop below 3.0 V and consume less than 5 mW of power. We tested scenario where photosensor was uncovered, and covered by black, grey and white colors of cloth. $V_{IN}$ = 5 V always and quantitative results are organized in table 4:

Table 4: Quantitative Measurements of Photosensor

| | **Uncovered** | **Covered Black** | **Covered Grey** | **Covered White** |
|---|---|---|---|---|
| **Current(mA)** | 0.425 | 0.07 | 0.3 | 0.372 |
| **$V_{OUT}$(V)** | 4.26 | 0.6 | 2.6 | 2.86 |
| **Power(mW)** | 2.125 | 0.35 | 1.5 | 1.86 |

### 3.2.2 PIR Motion Sensor

The PIR sensor should be able to correctly detect presence of human hands and arms motion. This was verified by completion of the entire project. Quantitative requirement is that it consume less than 1mW of power. $V_{IN}$ = 5 V always and we measured the current it draws when outputting logic high and low:

Table 5: Quantitative Measurements of PIR Sensor

| | **Outputting high** | **Outputting low** |
|---|---|---|
| **Current(μA)** | 140 | 40 |
| **Power(mW)** | 0.7 | 0.2 |

## 3.3 User Interface

### 3.3.1 RGB LED

The RGB LED should be able to correctly indicate the current state of operation. This is verified by completion of the entire project. Quantitative requirement is that it draws less than 40 mA of current. We measured the current for red, green, blue and white colors:

| | **Red** | **Green** | **Blue** | **White** |
|---|---|---|---|---|
| **Current(mA)** | 11.94 | 9.34 | 8.52 | 28.33 |

Table 6: Quantitative Measurements of RGB LED

### 3.3.2 Reset Button

The reset button should be able to pull the reset pin of ATMEGA328P to 0 V when pressed in order to reset the entire software program. This was verified by completion of the entire project and by the oscilloscope picture which shows high to low transition when the button is pressed:

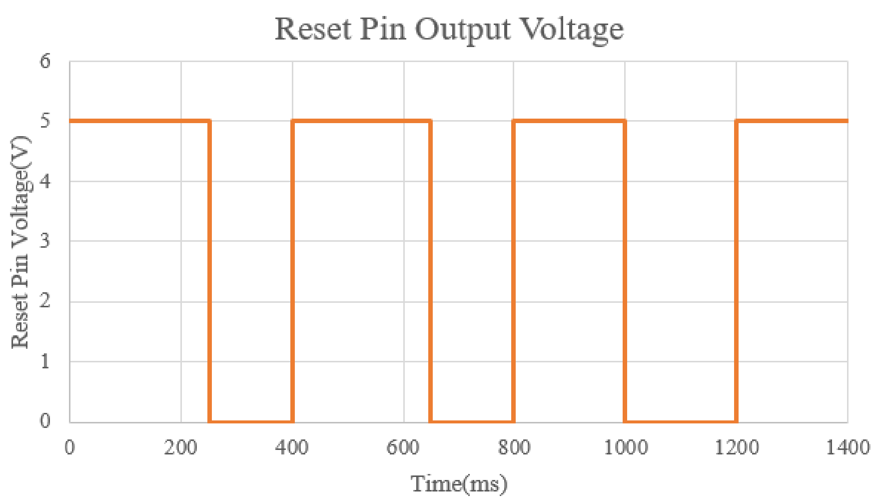


Figure 24: Reset Button Waveform

## 3.4 Mechanical Design

The requirements for the servo motors are: 1. Able to produce minimum of 5.38 kg-cm of torque in order to flip the board; 2. Able to rotate faster than 0.5 seconds/60°; 3. Able to reach within 5% of assigned target degree value; 4. Consume less than 15 W of peak power. The measured quantitative results are shown in table 7:

Table 7: Quantitative Measurements of Servo Motors

| | **Torque (kg·cm)** | **Speed (s/60°)** | **Peak Power Consumption (W)** |
|---|---|---|---|
| **Motor 1** | 18.38 | 0.23 | 14.5 |
| **Motor 2** | 18.98 | 0.25 | 14.4 |
| **Motor 3** | 18.56 | 0.26 | 13.8 |

Table 8: Quantitative Measurements of Servo Motor Angles

| Assigned Value | Actual Value | Assigned Value | Actual Value |
|---|---|---|---|
| 30° | 29° | 120° | 118° |
| 60° | 61° | 150° | 153° |
| 90° | 92° | 170° | 173° |

## 3.5 Control Design

Table 9: Quantitative Measurements of Feedback

| MinFeed(0°) | MaxFeed(170°) | MinFeed(0°) | MaxFeed(170°) |
|---|---|---|---|
| 44 | 564 | 35 | 568 |
| 16 | 558 | 44 | 566 |
| 22 | 577 | 15 | 567 |

MinFeed is the feedback value interpreted by ATMEGA328P when the servo is at 0° position, while MaxFeed is the feedback value interpreted by ATMEGA328P when the servo is at 170° position.

The requirement for the control unit is to process data and coordinate all units to achieve a functioning project. They were verified by completion of the entire project.

# 4. Costs

## 4.1 Parts Costs

Table 10 shows the total costs for all parts purchased for prototyping.

Table 10: Component Cost

| Vendor | Item | Unit price | Quantity | Total |
|---|---|---|---|---|
| Arduino | Uno development board with ATMEGA 328P | $16 | 1 | $16 |
| Amtel | ATMEGA 328P | $4.46 | 3 | $13.38 |
| ElecRight | Ultrasonic Sensor | $2 | 5 | $10 |
| D8 | Photo Sensor | $0.4 | 8 | $3.2 |
| TowerPro | MG958 Servo | $11.5 | 4 | $46 |
| Miracle | Folding board | $27 | 1 | $27 |
| TowerPro | MG995 Servo | $7 | 2 | $14 |
| TI | LM317 regulator | $0.44 | 5 | $2.2 |
| TI | LTC4365 Protection | $0.6 | 4 | $2.4 |
| Adafruit | Analog feedback servo | $14 | 1 | $14 |
| ECE Machine shop | Labor hours | N/A | 5 | N/A |
| EBL | Lithium batteries and charger | $14 | 1 | $14 |
| SparkFun | RGB LED | $3.9 | 4 | $15.6 |
| SparkFun | Power Switch | $1.0 | 3 | $3.0 |
| SparkFun | Reset Button | $1.0 | 2 | $2.0 |
| Parts Total | | | | $182.78 |

## 4.2 Labor Cost

Table 11 shows the breakdown of labor cost for this project. Assume the ideal salary for each group member is $35/hour, the total labor cost will be $15,750.

Table 11: Labor Cost

| Name | Hourly Rate | Total Hours | Total |
|---|---|---|---|
| Xudong Li | $35.00 | 150 | $5,250 |
| Anran Su | $35.00 | 150 | $5,250 |
| Suicheng Zhan | $35.00 | 150 | $5,250 |
| Total | $105.00 | 450 | $15,750 |

## 4.3 Total Cost

$$\text{Total Cost} = \text{Parts} + \text{Labor Cost} \times 2.5 = \$182.78 + \$15{,}750 \times 2.5 = \$39{,}557.80$$

# 5. Conclusion

In conclusion, our automatic cloth folding machine is able to detect the presence of the cloth which is placed on the board and automatically fold it in a neat way. The LEDs on the board indicates the state of user interface. For the consideration of safety, the folding board will reset to the idle state when there is obstruction during the folding process.

## 5.1 Accomplishments

For the power supply unit, the protection circuit is capable of cutting off the circuit when the input voltage is below 4.7V. The voltage regulators successfully regulate the output voltages to 5V and 5.8V. For the sensing unit, the photo resistors can detect the presence of the cloth and PIR sensor can detect the movement of human hands and arms. For the mechanical unit, the servo motors can provide a torque large enough to rotate the rod and flip the board and provide position and obstruction feedbacks back to the microcontroller. For the user interface unit, the RGB LED demonstrates the different user interface states by showing various color.
In general, all the components achieve their objectives and work together as a functional project.

## 5.2 Uncertainties

The obstruction occurred during the folding process may cause damage to the plastic disk socket that is in direct contact with high torque motor rotation metal gear. This happens because when the board is obstructed, the servo motor still attempt to spin at high torque while the plastic disk remains still. The motor metal gear therefore has a chance of grinding the disk socket resulting loss of friction between motor metal gear and disk socket. Without the friction, it will be difficult for the motor to flip the board.

## 5.3 Future Work

Possible future improvements are: 1. Better wiring on the back of the board; 2. Replacement of the plastic disk socket with metal disk socket to address the obstruction damage issue; 3. Improvement of the cloth sensing algorithm for better sensing under darker lighting environment.

## 5.4 Ethical Consideration

For the ethical consideration, we strictly obey the principle of IEEE Code of Ethics #1 and #9: "to accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment", "to avoid injuring others, their property, reputation, or employment by false or malicious action" [6]. In our project, we focus on two safety problems: the power supply safety problem and user hands obstruction problem. We designed the protection circuit and PIR sensors to avoid the potential endanger to the public as well as the injury to the user.

# References

[1] Linear Technology, “Overvoltage, Undervoltage and Reverse Supply Protection Controller” [Online]. Available:
http://cds.linear.com/docs/en/datasheet/4365fa.pdf

[2] Text Instrument, “LM317 3-Terminal Adjustable Regulator” [Online]. Available:
http://www.ti.com/lit/ds/symlink/lm317.pdf

[3] “HC-SR501 PIR Motion Detector datasheet” [Online]. Available:
https://www.mpja.com/download/31227sc.pdf

[4]”RGB LED by Barragan” [Online]. Available:
http://wiring.org.co/learning/basics/rgbled.html

[5] Adafruit, “About Servos and Feedback”[Online]. Available:
https://learn.adafruit.com/analog-feedback-servos/about-servos-and-feedback

[6] IEEE, “IEEE Code of Ethics” [Online]. Available:
https://www.ieee.org/about/corporate/governance/p7-8.html

# Appendix A: Requirements and Verifications

| Module | Requirements | Verifications |
| --- | --- | --- |
| Protection Circuit | 1. The protection circuit should turn off the circuit below 4.5V and above 16.0V in order to protect the circuit from burning out. | (a) Connect the input of the protection circuit to a DC power supply and then sweep the input voltage from 17.5V to 0V with step of 0.5V.<br><br>(b) Connect the output of the protection circuit to the multimeter or oscilloscope to confirm output voltage drops to near 0V when the input voltage is below 4.5 V and above 16.0V |

| Module | Requirements | Verifications |
| --- | --- | --- |
| Voltage Regulator | 1. The output voltage for LDO should be within ±0.1V of the assigned value. There are two types of voltage regulators in this project: 5V output and 5.8V output. | 1 (a) Connect the input of the voltage regulator to a DC power supply and then sweep the input voltage from 10 to 18V.<br><br>(b) Connect the outputs of the voltage regulators to a multimeter or oscilloscope to see whether the output voltage is 5V ± 0.1V or 5.8V ± 0.1V for two regulators. |

| Module | Requirements | Verifications |
| --- | --- | --- |
| Control Circuit | 1.Able to send PWM signal to the servo motor to spin both clockwise and counterclockwise for 180°. | 1.(a) Power up the entire project and let it go through the entire operation.<br>(b) Observe if the motor can flip the board both clockwise and counterclockwise for 180 degree at the appropriate stage. |
| | 2.Able to interpret analog position and obstruction feedbacks from servo motors. | 2. (a) Power up the entire project and let it go through the entire operation.<br>(b) Use hand to obstruct the flipping board when moving, then observe if the flipping stops and the LED flashes red to indicate error state. |
| | 3. Able to interpret digital | 3. (a) Power up the entire project and let it |

| | | |
|---|---|---|
| | data from the PIR motion sensor. | go through the entire operation.<br>(b) Observe if the LED stays at blue color after cloth is on and during the entire time when user is using his/her hands to adjust the cloth on board.<br>(c) After user has finished adjusting the cloth and removed hands from board area, observe if the LED turns red to indicate folding about to begin. |
| | 4. Able to interpret data signals from the photosensors. | 4. (a) Power up the entire project and let it go through the entire operation.<br>(b) Place a piece of cloth covering the PS1 and PS2 sensors and confirm if the LED turns blue to indicate cloth is on.<br>(c) After folding is complete and LED is in green color, remove the cloth and confirm if the LED turns white to indicate ready for next cloth. |

| Module | Requirements | Verifications |
|---|---|---|
| Photosensors | 1. Must be able to drop output voltage below at least 3.0V when the sensor is covered. | 1. (a) Connect the input of photosensors PCB to 5V and output to a multimeter.<br>(b) Use different color of cloth to cover the photosensor and read from multimeter to confirm output voltage is below at least 3.0V for white color(as white color block out the least amount of light)<br>(c) Remove cloth from the board, read from the multimeter to confirm a higher voltage reading than in (b). |
| | 2. Consumes less than 5 mW of power. | 2. (a) connect the sensor to a multimeter to measure the current.<br>(b) calculate the power consumption to see if it meets 5 mW requirement. |

| Module | Requirements | Verifications |
|---|---|---|

| | | |
|---|---|---|
| PIR Sensor | 1. Must be able to detect moving hands and arms at distance within the board area.<br><br>2. Consumes less than 1mW of power. | 1. (a) Power up the entire project and let it go through the entire operation.<br>(b) Observe if the LED stays at blue color after cloth is on and during the entire time when user is using his/her hands to adjust the cloth on board.<br>(c) After user is done adjusting the cloth and has removed hands from board area, observe if the LED turns red to indicate folding about to begin.<br><br>2. (a) connect the sensor to a multimeter to measure the current.<br>(b) calculate the power consumption to see if it meets 1mW requirement. |

| Module | Requirements | Verifications |
|---|---|---|
| RGB LED | 1. The current goes through our RGB LED should be lower than 40 mA to be power efficient. | 1. (a)Connect the RGB LED circuit to ATMEGA328P chip and upload a simple code to ATMEGA to change the LED color.<br>(b) use multimeter to measure the current through LED at colors of red, green, blue and white.<br>(c) check if all current values are less than 40mA. |

| Module | Requirements | Verifications |
|---|---|---|
| Reset Button | 1. Able to pull the reset pin of atmega328P to near 0V when pressed, and stays at 0V as long as the button is pressed. | 1. (a) Connect the output of the button to oscilloscope.<br>(b) Verify the waveform to see if there is high to low transition when button is pressed. Also confirm if the output stays low as long as the button is pressed. |

| Module | Requirements | Verifications |
|---|---|---|

| | | |
|---|---|---|
| Battery | 1. Able to provide an output voltage of within 10% of 3.7V each cell. | 1. (a) Connect the output of the battery to multimeter in series with a 10kΩ resistor<br>(b) Record the output voltage every 10 mins to see if there is a voltage drop. |

| Module | Requirements | Verifications |
|---|---|---|
| Servo Motors | 1. Able to produce a torque of 5.38 kg·cm minimum at supply voltage of 5.8V. | 1. (a) Connect the signal pin of servo motor to Arduino.<br>(b) Connect the power of servo motor to a 6V supply.<br>(c) Attach a lever to the rotation axis of the motor.<br>(d) Attach a spring scale to the lever.<br>(e) Send a command to rotate the servo motor forward for 180° in one step.<br>(f) Read the force on the spring scale.<br>(g) Calculate torque to see if it is greater than 5 kg·cm. |
| | 2. Able to rotate both clockwise and counterclockwise faster than 0.5s/60°. | 2. (a) Connect the signal pin of servo motor to Arduino.<br>(b) Connect the power of servo motor to a 6V supply.<br>(c) Send a command to rotate the servo motor forward for 180°.<br>(d) Start timer as soon as the servo starts to rotate.<br>(e) Stop timer as soon as the servo stops.<br>(f) Check if the speed meets requirement.<br>(g) Repeat for backward rotation. |
| | 3. Able to achieve a degree of rotation within 5% of assigned value. | 3. (a) Connect the signal pin of servo motor to Arduino.<br>(b) Connect the power of servo motor to a 6V supply.<br>(c) Send a command to rotate the servo motor forward for 30°.<br>(d) Use a protractor to measure the degree of rotation<br>(e) Repeat the measurement for 60°, 90°, 120°, 150°, and 180°. |

| | | |
|---|---|---|
| | 4. Consumes less than 15W of power. | 4. (a) connect the sensor to a multimeter to measure the current.<br>(b) calculate the power consumption to see if it meets 15W requirement. |