CPE Senior Project
Design Report

Automated Solar-Powered
Chicken Coop Door

Brenna Yagade
Advisor: Dr. John Oliver

June 2015

Table of Contents

List of Tables

List of Figures

Introduction

Project Overview
This project involved the planning, design, and building of an automated, low power system to open and close a chicken-coop door, which is powered through a solar panel.

Advisor/Client
Dr. John Oliver

Task Force
Brenna Yagade (Computer Engineering student)

Objective

System Requirements
- The system is automated and solar powered.
- The system opens chicken coop door automatically after sunrise.
- The system closes chicken coop door automatically after set time after sunset.
- The system illuminates chicken coop interior for a set time after sunset.
- Jumper connectors used to modify settings for duration of time system keeps chicken coop door open after sunset.
- Jumper connectors used to modify settings for duration of time LED strip illuminates chicken coop interior for after sunset.
- Potentiometer used to adjust light sensor sensitivity.

Design Concept

High-Level Layout
Figure 1 shows the high-level black box diagram for the entire system.

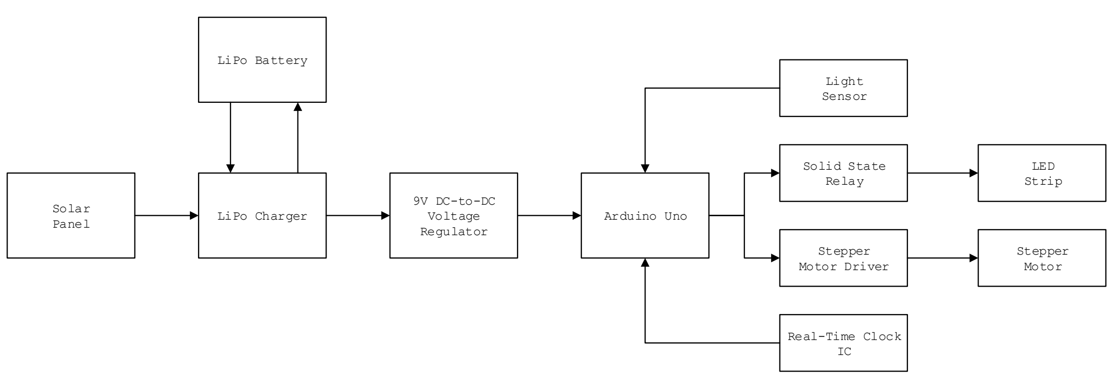


Figure 1: High-level system black box diagram

Components
The design matrix in Appendix B shows the calculated outcomes for each potential component within each system function category. The most important specifications

that were considered were power consumption, size, and accuracy. The comparison charts in Appendix C further detail the product analysis that went into each design decision.

Microcontroller

Multiple microcontrollers were considered for this project, including multiple Arduino boards, the Raspberry Pi, the BeagleBone Black, and a TI MSP Launchpad. Because of the number of peripherals, one of the key features was that the microcontroller have enough GPIO pins. All of the microcontrollers considered had more than enough GPIO pins

Although options like the TI MSP430G2553 Launchpad are more low power, the Arduino Uno has a friendlier developing environment and collection of libraries to access and use. The Arduino Uno also had double the flash memory, and having worked with the TI MSP430G2553 before, there were issues encountered when reaching that threshold.

The evaluated microcontrollers were fairly similar in specifications and because the Arduino Uno was already initially supplied, there wasn't an alternative that was significantly better to justify purchasing another microcontroller.

Table 1 shows the system specifications for the ATmega 328P microprocessor on the Arduino Uno development board. The most important specifications that were looked at were the active mode power consumption, which is approximately 9 mA when the Arduino Uno is powered at 5v, and the pin voltages and currents.

The pins can output a minimum high current of 20 mA and a maximum of 50 mA, which was important in setting the current limiting resistor to the solid state relay. The output and input voltages on the pins were also used to determine the range of resistors used for the jumper settings.

Specification	
Microprocessor	
Make	Atmel
Model	ATmega 328P
Power consumption	
Active mode:	9 mA
Idle mode:	2.7 mA
Power-save mode	2.6 µA
Operating voltage (V_{CC})	1.8 v – 5.5 v
CLK frequency	8MHz
Output high voltage, min (V_{OH})	2.3 v
Output low voltage, max (V_{OL})	0.6 v
Input high voltage (V_{IH})	
Min:	$0.6V_{cc}$
Max:	$V_{cc} + 0.5$ v
Input low voltage (V_{IL})	
Min:	-0.5 v
Max:	$0.3V_{cc}$
Output high current (I_{OH})	20 mA
Output low current (I_{OL})	-20 mA

Table 1: Atmel ATmega 328P microprocessor specifications

Solar Panel

There were many solar panel options to choose from, but the most important criteria was the cell efficiency and the current output, which determines how quickly the panel can charge the LiPo battery. Most of the solar cells researched had a cell efficiency of approximately 15% - 17% and were monocrystalline panels.

The Sparkfun 8V 5.2W solar cell and the Adafruit 6V 3.4W solar panel were found to be the optimal choices. The Sparkfun solar cell is a little bit better of a choice, as the current output is 650 mA, as opposed to the 530 mA output from the Adafruit solar panel, meaning that the battery can be charged a little faster.

However, the Adafruit 6V 3.4W solar panel was eventually used in the final design for multiple reasons. Although the Sparkfun solar cell has high current output, it is also a little less efficient, 15% versus the Adafruit panel's 17% efficiency. Also, as both panels were the same price and the Adafruit solar panel had already been provided, it wasn't deemed feasible to purchase another solar panel that wouldn't provide significantly more benefits.

Light Sensor

Both photocells and light sensors were considered in the design of this project. Photocells essentially act as a variable resistor and are very easy to implement. They are small, low power, and durable. However, they are fairly inaccurate. Light sensors are extremely precise and allow for exact lux calculations.

The TSL2561 Digital Light Sensor detects light ranges from 0.1 – 40,000 lux and the GA1A12S202 Log-scale Analog Light Sensor detects light ranges from 3 – 55,000 lux, making them both extremely accurate sensors.

A photoconductive cell was ultimately chosen as it is extremely low cost and is responsive to both very low light levels and very high light levels, i.e. moonlight and direct sunlight, respectively. The photoconductive cell requires less pins on the microcontroller and is, on average, 1/5 the price of either of the light sensor chips.

Although it is the less accurate option, its implementation in this project does not require the determination of precise lux levels. The digital and analog light sensors mentioned above have high precision, which makes them the ideal choice for applications requiring detection of very small light changes. Because the light sensor will only be used in this project to determine basic light transitions from light to dark and vice versa, such high precision is not required, and so the CdS (Cadmium-Sulfide) photoconductive cell is the preferable component choice.

LED Strip

There are a multitude of LED strips currently on the market, both white and RGB LEDs. Almost all of the LED strips are run off of 12V and have a current draw upwards of 1A per meter, wherein a meter contains 60 LEDs. As the LED strips are divided into segments, each of which contain three LEDs, the length of the strip can be cut into less segments, thereby decreasing the overall current draw.

The Lampux Flexible LED Strip and the HitLights Cool White LED Strip both came in a length of 5m and as the strips can be cut down into shorter segments, either would have been the most cost-efficient option. However, there was no data on the current draw of the strip and as such, it made it infeasible to use without knowing whether they could be integrated into the system.

Of the remaining four LED strip choices, the Sparkfun RGB strip was eliminated, as its current draw was too high, at 1800 mA. The LED Light Strips from SuperBrightLeds pulled the least current per meter (960 mA, or approximately 48 mA per segment). However, the Cool White Flexi-Strip from Adafruit was eventually chosen. Even though the maximum current consumption is 1200 mA, or 60 mA per segment, it is confirmed to have the capability to be run off of 9v, which would significantly lower the power consumption.

Figure 2 shows the calculations for the power consumption of the 1-meter LED strip at maximum current draw. Because the maximum current draw of the LED strip powered at 12v is so high, it is not feasible to have the LED strip powered at 12v if the one-hour time requirement is to be met.

$$W_{LiPo} = (I_{load}) * (V_{load}) * (\#\ of\ segments) * (time\ to\ light\ LEDs),$$
$$where\ (\#\ of\ segments) = \left(\frac{20\ segments}{meter}\right) * (length\ of\ LED\ strip)$$

$$9.25\ wH = (1.2A) * (12v) * (20\ segments) * (t)$$

$$\boldsymbol{t = 0.0321\ hours = 1.927\ minutes}$$

Figure 2: Power calculations for LED strip at 12V and maximum current draw

As the above power calculation was based on maximum current draw, tests were done to determine the actual current draw. When the LED strip was powered at 12v, the current draw was found to average at 920 mA. The LED strip can also be used with a 9v power supply, and at that voltage, the current draw was found to average at 320 mA.

The calculations, shown in Figures 3 and 4 were recomputed using this new data. At 12v, the system would be able to power a 1-meter LED strip for approximately 2.514 minutes and at 9v, the system would be able to power a 1-meter LED strip for approximately 9.635 minutes.

$$W_{LiPo} = (I_{load}) * (V_{load}) * (\#\ of\ segments) * (time\ to\ light\ LEDs)$$

$$9.25\ wH = (0.92A) * (12v) * (20\ segments) * (t)$$

$$\boldsymbol{t = 0.04189\ hours = 2.514\ minutes}$$

Figure 3: Power calculations for LED strip powered by 12v

$$W_{LiPo} = (I_{load}) * (V_{load}) * (\#\ of\ segments) * (time\ to\ light\ LEDs)$$

$$9.25\ wH = (0.32A) * (9v) * (20\ segments) * (t)$$

$$\boldsymbol{t = 0.1606\ hours = 9.635\ minutes}$$

Figure 4: Power calculations for LED strip powered by 9v

As the LED strip is the component that consumes the most power in the system, it is unable to meet the one-hour time requirement if it has a length of one meter. Therefore, either the length of the LED strip can be lessened to meet the one-hour time requirement or the time requirement can be reduced for the one-meter length.

The LED strip length would need to be reduced to 3.212 segments, or 9 LEDs, to meet the one-hour time requirement or alternatively, the time requirement would need to be reduced to 9.635 minutes to light the one meter LED strip. As either option is not particularly preferable, a middle ground was reached.

The LED strip with six segments, which is approximately one-third of the original one-meter length, can be lit for a maximum time of 22.57 minutes.

Motor
Initially, a standard 130 size DC motor was selected to lift and lower the chicken coop door. It has the advantage of being low power and is has an operating voltage range of 4.5v – 9v. The maximum current draw for the motor is 250 mA, but its rated load is only 10 g-cm, or 0.139 oz-in.

This ended up presenting problems, as the DC motor has high RPM, but low torque. As the mock chicken coop door that was constructed for testing was found to be heavier than initially anticipated, the DC motor was not sufficient to either lift the door or keep it in a stable open position.

A gearing mechanism was constructed to help offset some of the load, but still, the DC motor was unable to lift the door. As high RPM was not a crucial design factor, servo motors and stepper motors were evaluated for their feasibility in place of the DC motor.

Although servo motors can be controlled more precisely, the angle of rotation is limited to 180 degrees, making it unable to continuously rotate in order to wind up the fishing line string to pull up the door.

Eventually, a bipolar stepper motor, with a rated load of 68 oz-in, was chosen. Although not nearly as low power as the DC motor, the stepper motor driver limits the output current to the motor to a maximum of 750 mA. Upon actually testing the stepper motor, it was found that the current draw averaged 474.8 mA.

Figure 5 shows the initially chosen DC motor and Figure 6 shows the final stepper motor utilized in the design.

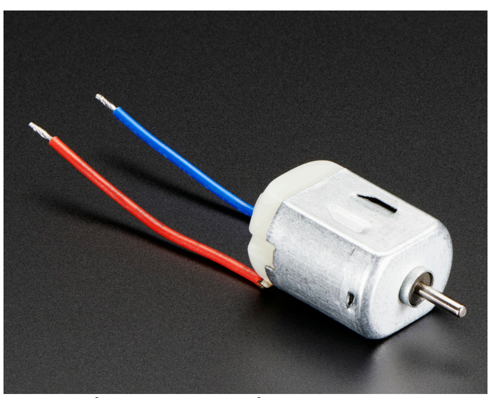


Figure 5: 130 size DC motor

Figure 6: Bipolar stepper motor

Door Mechanism
Because neither motor is capable of lifting and holding the chicken coop door in the open position, a gearing mechanism needed to be constructed to assist in the action. Because the focus of this project mostly centered on the electronics system, minimal

analysis went into the stress and physics calculations involving the gear system. Figure 7 shows the setup of the gears and the motor.

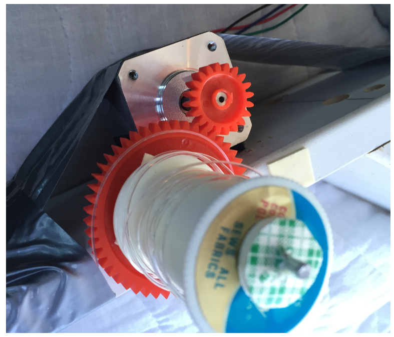


Figure 7: Gear system to lift chicken coop door

Timing Crystal Oscillator/Real-Time Clock IC

Originally, a 32.768 kHz crystal oscillator was going to be used to keep track of how much time has passed since sunset, so the system can close the chicken coop door. It was also going to be used to keep track of how long the LED strip had lit the interior of the chicken coop. The advantage of having an external crystal oscillator is that it is extremely low power, instead of using the Arduino as a timing source for such an extended period. Figure 8 shows the circuit integrating the 32.768 kHz crystal oscillator with the Arduino Uno.

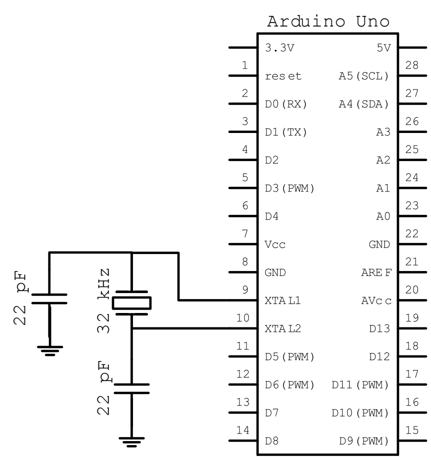


Figure 8: Circuit schematic for 32.768 kHz external crystal oscillator integrated with Arduino Uno

However, some issues arose because the Arduino Uno being used in this project has a surface-mount ATmega 328P. Because of this, there is no access to the XTAL1 and XTAL2 pins.

There were a few possible solutions to this problem.

1. Remove the 16 kHz crystal oscillator from the board and then replace it with the 32.768 kHz crystal oscillator.

2. Purchase a new Arduino Uno with through-hole mounting of the ATmega 328P and connect 32.768 kHz crystal oscillator to XTAL1 and XTAL2 pins.
3. Use an external chip that utilizes the 32kHz crystal oscillator.

To avoid making any changes to the board, a real-time clock IC was interfaced with the Arduino Uno. The DS1307 IC was chosen, as it is low power and has a built-in power-sense circuit to detect power failures and automatically switches to the backup power supply, without disrupting its time-keeping ability. The LiPo battery primarily powers the DS1307, but a coin-cell battery is used as a backup source.

When powered by the external source, the real-time clock draws a maximum active supply current of 1.5 mA and draws a maximum current of 500 nA when relying on the backup battery.

Voltage Regulator

As the LiPo battery is only rated at 3.7v and the LED strip and the motor driver require at least 9v to operate, a voltage regulator is needed.

Multiple DC-to-DC switching regulators were considered and evaluated, as they provided a very cost-efficient way of stepping up the voltage. Their efficiency ranged from approximately 80-90% and each needed a custom circuit designed to output 9v with an appropriate current. Out of all the DC-to-DC switching regulators that were evaluated, the most appropriate one was the TI TPS55300, which required the accompanying circuit, shown in Figure 9.

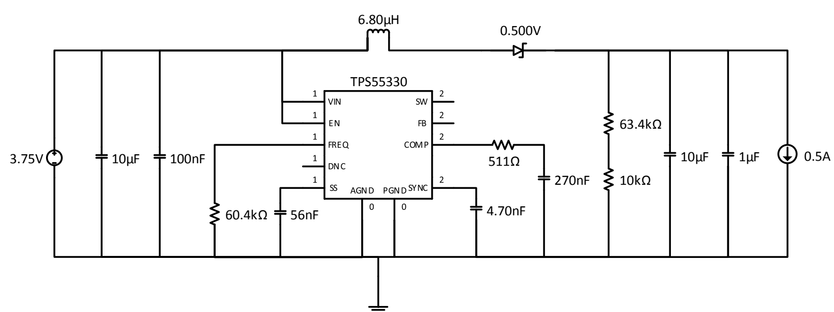


Figure 9: DC-to-DC switching regulator circuit

However, it was later determined that although a DC-to-DC switching regulator was more cost-efficient, the number of parts required and the increased difficulty to implement made using a 9v step-up voltage regulator that produces a fixed 9v output was a better design decision.

The 9v step-up voltage regulator's efficiency was also in the range of 80%-90% and is particularly useful in applications where the power supply voltage varies, as is the case with the LiPo battery, where the voltage begins above the rated value and then drops below that as the battery discharges.

Theoretical Power Calculations

Figure 10 shows the calculation for the minimum time needed to fully charge the LiPo battery to full capacity.

LiPo Battery (fully-charged):

$$W_{battery} = V_{battery} * q_{battery}$$

$$W_{battery} = (3.7v) * (2500mAh) = 9250mWh = 9.25\ Wh$$

Solar Power Energy:

$$W_{solar\ panel} = P_{solar\ panel} * t_{exposed\ to\ sunlight} * (natural\ system\ losses)$$

$$W_{solar\ panel} = (3.18W) * \left(t_{exposed\ to\ sunlight}\right) * (0.85)$$

$$W_{solar\ panel} = 2.703\ t_{exposed\ to\ sunlight}$$

Time Calculation:

$$W_{battery} = W_{solar\ panel}$$

$$\boldsymbol{t_{exposed\ to\ sunlight} = 3.422\ hrs}$$

Figure 10: Power calculation for time to fully charge LiPo battery

Under the following optimal conditions,

- Maximum solar panel cell efficiency,
- Constant “bright” sunlight,
- Average natural system losses,

the LiPo battery will be fully charged after approximately 3.422 hours of exposure to sunlight.

Given that the LiPo battery is fully charged, it holds a total charge of 9.25 Wh. Figures 11 – 14 show the power consumption calculations for all of the components in the system.

$$W_{LED\ strip} = (I_{load}) * (V_{load}) * (\#\ of\ segments) * (time\ to\ light\ LEDs)$$

$$W_{LED\ strip} = (320mA) * (9v) * (6\ segments) * (20\ minutes)$$

$$\boldsymbol{W_{LED\ strip} = 5.76\ Wh}$$

Figure 11: Power calculations for LED strip

$$W_{motor} = (I_{load}) * (V_{load}) * (time\ to\ run\ motor)$$

$$W_{motor} = (474.8mA) * (9v) * (10\ minutes)$$

$$\boldsymbol{W_{motor} = 0.7122\ Wh}$$

Figure 12: Power calculations for motor and motor driver

$$W_{Arduino} = (I_{load}) * (V_{load}) * (time\ to\ run\ Arduino)$$

$$W_{Arduino} = (32.8mA) * (9v) * (24\ hrs)$$

$$\boldsymbol{W_{Arduino} = 7.0848\ Wh}$$

Figure 13: Power calculations for Arduino Uno

$$W_{real-time\ clock} = (I_{load}) * (V_{load}) * (time\ to\ run\ real - time\ clock)$$

$$W_{real-time\ clock} = (1.5mA) * (5v) * (24\ hrs)$$

$$\boldsymbol{W_{real-time\ clock} = 0.18\ Wh}$$

Figure 14: Power calculations for real-time clock IC

The total power consumption of the system is 13.737 Wh. Although the total power available via a fully charged LiPo battery is 9.25 Wh, that is the result of a full charge time of 3.422 hours. The Arduino Uno consumes a significant amount of power, as it runs constantly, but throughout the day the LiPo battery will be able to recharge, theoretically making the system feasible as long as the battery is able to go through at least two fully charge cycles within a one day period.

Hardware

Table 2 shows the bill of materials listing all of the components that were used.

Item	Part Number	Supplier	Quantity	Price Each	Extended Price
Development Board	Uno	Arduino	1	$24.95	$24.95
Cool White Weatherproof Flexi-Strip 60 LED Strip	887	Adafruit	1	$19.95	$19.95
Photo Cell (CdS photoresistor)	161	Adafruit	1	$0.95	$0.95
Large 6V 3.4W Solar Panel	500	Adafruit	1	$44.95	$44.95
Lithium Ion Polymer Battery – 3.7V 2500mAh	328	Adafruit	1	$14.95	$14.95
Solar Lithium Ion/Polymer Charger – v2	390	Adafruit	1	$17.50	$17.50
Stepper Motor (68 oz./in.)	ROB-10846	SparkFun Electronics	1	$16.95	$16.95
EasyDriver Stepper Motor Driver	ROB-12779	SparkFun Electronics	1	$14.95	$14.95
Solid State Relay – 2A 60V	VO14642AT	Vishay Semiconductor	1	$2.66	$2.66
9V Step-Up/Step-Down Voltage Regulator	S18V20F9	Polulu Robotics & Electronics	1	$14.95	$14.95

Real-Time Clock Module (DS1307)	DFR0151	DF Robotics	1	$4.30	$4.30
Breadboard	FIT0009	DF Robotics	1	$7.50	$7.50
				Total	**$184.56**

Table 2: Bill of materials

Software

Figure 15 shows the software routine executed by the Arduino Uno, which is responsible for getting light sensor values from the photo cell, driving the stepper motor, and controlling the solid state relay, which switches the LED strip on and off.

On start-up, the user settings are checked and values are obtained for the door time, LED time, and the photocell sensitivity. From there, the Arduino constantly reads the value from the photocell.

If the photocell value signifies a “dark to light” transition, then the motor is powered to open the chicken coop door.

If the photocell value signifies a “light to dark” transition, then the LED strip is turned on and the LED event timer and the door event timers are set. The program then enters timer mode, which checks to see if either event time has been reached. If the LED event time has been reached, then the LED strip is turned off and if the door event time has been reached, then the motor is powered to close the chicken coop door.

The full code run on the Arduino Uno microcontroller can be found in Appendix D. The preexisting RTC library was used in conjunction with the DS1307 real-time clock IC.

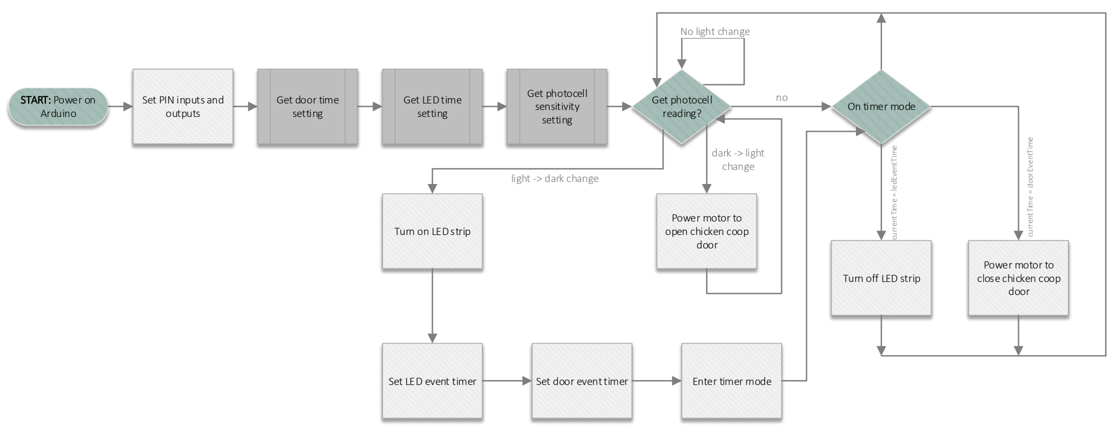


Figure 15: Software flowchart

System Integration

In order to test the system, a mock chicken coop door setup was assembled, as access to the actual chicken coop was limited. In addition to the chicken coop door itself, a wooden arm was constructed to simulate the ceiling, which is where the stepper motor would be hung from, in order to wind the 40 lb. fishing line to lift and lower the chicken coop door.

Figure 16 shows the setup that was used for testing purposes and demonstration. Figure 17 shows the circuit schematic for the entire system.

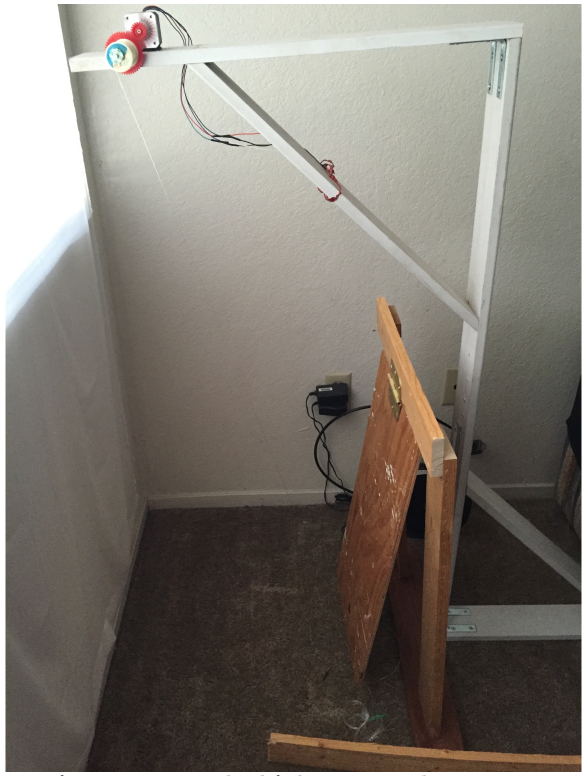


Figure 16: Mock chicken coop door setup

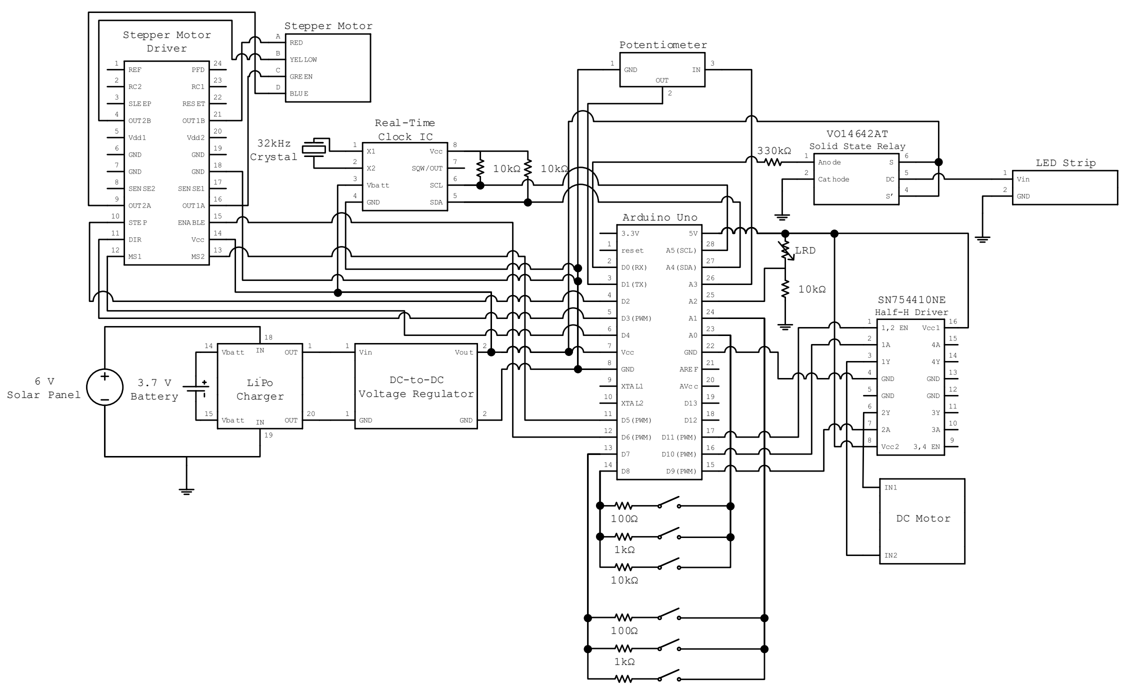


Figure 17: Circuit schematic for entire system

Modifiable User Settings

The user is able to adjust the system settings for the duration of time that the LED strip is lit for and the duration of time to wait after sunset before closing the chicken coop door. The user is also able to adjust the light sensor sensitivity of the photocell, to adjust to differing brightness levels in various environments.

Figure 18 shows the components for modifying the various settings.

- The potentiometer on the left adjusts the light sensor sensitivity. From low position to high position, the potentiometer sets the photocell transition value from 0 – 1000, respectively. That transition value represents the difference between "light" conditions and "dark" conditions.
- The three pairs of jumper pins in the middle adjust the time that the LED strip is lit for. From left to right, the settings are:
    - 10 minutes
    - 15 minutes
    - 20 minutes
- The three pairs of jumper pins on the right adjust the time to wait after sunset before closing the chicken coop door. From left to right, the settings are:
    - 15 minutes
    - 30 minutes
    - 1 hour

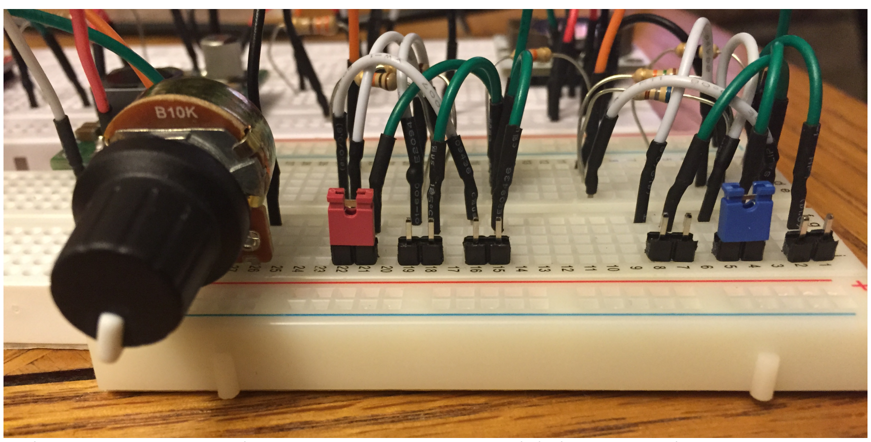


Figure 18: Adjustable user settings for photocell sensitivity, LED time, and door closure time

Testing

Each component was tested individually and then integrated into the system one component at a time. Tests were run at each stage of integration. A full list of all tests run can be found in Appendix E.

Light Sensor

Table 3 shows the tests that were run for the light sensor. For being indoors, in lab, the unblocked light sensor lux value was approximately 800 and the blocked light sensor lux value was approximately 600. Thus, for indoors, lux values less than 700 indicate "darkness."

In an outdoors setting, the unblocked light sensor lux value was found on average to be approximately 1000 and the blocked light sensor lux value was found on average to be approximately 900. Thus, for outdoors, lux values less than 950 were interpreted as "nighttime."

TEST PLAN		TEST REPORT	
Test Description	**Acceptance Criteria**	**Pass/Fail**	**Test Result**
Changes in lux readings from sensor when light is blocked/unblocked	Values printed in serial monitor fluctuate according to sensor being blocked/unblocked	✓	Lux values printed to serial monitor varied
Block light sensor for indoors – lux readings < set value (700), digital output = 5v	Output on DIGITAL PIN 7 = 5v	✓	DIGITAL PIN 7 output = 5.04v
Unblock light sensor for indoors – lux readings > set value (700), digital output = 0v	Output on DIGITAL PIN 7 = 0v	✓	DIGITAL PIN 7 output = 50mV
Block light sensor for outdoors – lux readings < set value (950), digital output = 5v	Output on DIGITAL PIN 7 = 5v	✓	DIGITAL PIN 7 output = 5.03v
Unblock light sensor for outdoors – lux readings > set value (950), digital output = 0v	Output on DIGITAL PIN 7 = 0v	✓	DIGITAL PIN output = 50mV

Table 3: Test plan for light sensor

LED Strip

Table 4 shows the tests that were run for the solid-state relay and Table 5 shows the tests that were run for the LED strip.

Because the LED strip is powered at 12v and will draw a maximum current of 1.2A, a solid-state relay is needed to ensure that the current drawn from the Arduino Uno digital output pin does not exceed its maximum value of 50mA. Thus, the solid-state relay output rail is tied to a 12v source and the output is triggered by the signal from the Arduino Uno digital pin.

Although the LED strip is intended to be powered by 12v, it is also capable of being powered by a 9v source. The LED strip at 9v is slightly dimmer than the LED strip at 12v, as shown in Figures 19 and 20, however for the purposes of its use in this project, that diminished brightness is negligible.

Figure 19: LED strip powered at 12v

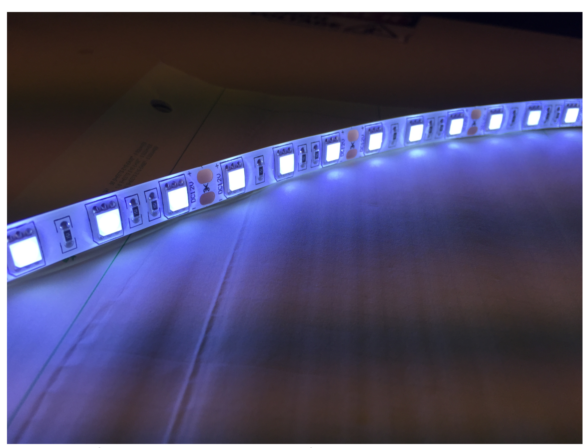


Figure 20: LED strip powered at 9v

The first set of tests was run with the system being powered by a 5v power supply. The current-limiting resistor between the 5v power source and the anode of the solid-state relay ensures that the current doesn't exceed 50 mA. A resistor value of 330Ω will theoretically result in a voltage drop of 3.3v across the resistor and 10mA current, which is sufficient to turn on the LED inside of the solid-state relay.

TEST PLAN		TEST REPORT	
Test Description	**Acceptance Criteria**	**Pass/Fail**	**Test Result**
Power w/ 5v – get voltage drop across current-limiting resistor and calculate current across resistor	Current across resistor ≈ 10 mA	✓	Voltage drop = 3.62v, current across resistor = 10.9 mA
Power w/ 5v and 12v rail on output – load output is 12v	Load output on PIN 5 is 12v	✓	PIN 5 output = 12.09v
Turn on/off 5v power source w/ 12v rail – load output should turn on/off	5v power source ON, load output is 12v; 5v power source OFF, load output is 0v	✓	5v power source ON, PIN 5 output = 12.09v; 5v power source OFF, PIN 5 output = 0v

Table 4: Test plan for solid-state relay

TEST PLAN		TEST REPORT	
Test Description	**Acceptance Criteria**	**Pass/Fail**	**Test Result**
Power w/ 12v from power supply	LED strip is ON	✓	LED strip is ON
Power w/ 9v from power supply	LED strip is ON	✓	LED strip is ON

Table 5: Test plan for LED strip

Motor

Table 6 shows the tests that were run for the motor and door mechanism.

TEST PLAN		TEST REPORT	
Test Description	**Acceptance Criteria**	**Pass/Fail**	**Test Result**
DC motor has bi-directional rotation	DC motor rotates clockwise, DC motor rotates counter-clockwise	✓	DC motor rotates clockwise, DC motor rotates counter-clockwise
DC motor triggered, door opens and is held in open position	DC motor rotates, door opens and holds in open position	✗	DC motor rotates, unable to lift door open
Stepper motor has bi-directional rotation	Stepper motor rotates clockwise, stepper motor rotates counter-clockwise	✓	Stepper motor rotates clockwise, stepper motor rotates counter-clockwise
Stepper motor triggered, door opens and is held in open position	Stepper motor rotates, door opens and holds in open position	✓	Stepper motor rotates, door opens and holds in open position

Table 6: Test plan for motor and door mechanism

Integrated System

Below are the tests that were run for each stage of system integration.

Table 7 shows the tests that were run for the Arduino Uno, solid-state relay, LED strip, and light sensor integrated system. The Arduino Uno was powered through USB at 5v and thus, the output of the digital pins was also 5v. The low output on Digital Pin 7, the digital output pin connected to the anode of the solid-state relay, was measured at 50mV and the high output was measured at 5.04v. The solid-state relay output the appropriate voltage and current and was sufficient to turn the LED strip on and off.

The system was tested with both a 12v and 9v rail on the solid-state relay. At 12v, the LED strip draws 0.92A and at 9v, it draws 0.32A. That difference in current draw significantly affects the power consumption, as shown in the power calculations in Figures 21 and 22. Because of this, 9v is being used to power the LED, as opposed to 12v.

If the LED strip is required to be on for a longer period of time, its length will have to be shortened. Alternatively, if a shorter period of time is required, the length of the LED strip can be increased.

TEST PLAN		TEST REPORT	
Test Description	**Acceptance Criteria**	**Pass/Fail**	**Test Result**
Power w/ 5v from Arduino and 12v rail on output, lux readings < set value (700) – LED strip is ON	LED strip is ON	✓	LED strip is ON, current draw = 920mA
Power w/ 5v from Arduino and 12v rail on output, lux readings > set value (700) – LED strip is OFF	LED strip is OFF	✓	LED strip is OFF
Power w/ 5v from Arduino and 9v rail on output, lux readings < set value (700) – LED strip is ON	LED strip is ON	✓	LED strip is ON and noticeably dimmer than w/ 12v rail, current draw = 320mA
Power w/ 5v from Arduino and 9v rail on output, lux readings > set value (700) – LED strip is OFF	LED strip is OFF	✓	LED strip is OFF

Table 7: Test plan for Arduino Uno, solid-state relay, LED strip, and light sensor system

$$P_{LiPo} = (I_{load}) * (V_{load}) * (\#\ of\ segments) * (time\ to\ light\ LEDs)$$

$$9.25\ wH = (920mA) * (12v) * (n) * (0.5hr)$$

$$\boldsymbol{n = 1.68\ segments = 5\ LEDs\ \approx 0.084m\ strip}$$

Figure 21: Calculation for number of LED strip segments powered by 12v

$$P_{LiPo} = (I_{load}) * (V_{load}) * (\#\ of\ segments) * (time\ to\ light\ LEDs)$$

$$9.25\ wH = (320mA) * (9v) * (n) * (0.5hr)$$

$$\boldsymbol{n = 6.42\ segments = 19\ LEDs\ \approx 0.321m\ strip}$$

Figure 22: Calculation for number of LED strip segments powered by 9v

Table 8 shows the tests that were run for the Arduino Uno, light sensor, stepper motor, stepper motor driver, and door mechanism integrated system. The Arduino Uno was powered through USB at 5v and the stepper motor driver was powered with a power supply at 9v. The stepper motor was connected to the door mechanism to be able to lift and lower the mock chicken coop door. At 9v, the stepper motor was measured to have an average current draw of 475 mA.

TEST PLAN		TEST REPORT	
Test Description	**Acceptance Criteria**	**Pass/Fail**	**Test Result**

Power w/ 5v from Arduino, 9v rail on stepper motor driver V_{cc}, lux readings < set value (700) -- triggers stepper motor, door closes	Door mechanism closes	✓	Stepper motor rotates, door closes, current draw = 0.475A
Power w/ 5v from Arduino, 9v rail on stepper motor driver V_{cc}, lux readings > set value (700) -- stepper motor does nothing, door remains open	System remains unchanged	✓	Stepper motor holds in static position, door remains open,

Table 8: Test plan for Arduino Uno, light sensor, stepper motor, stepper motor driver, and door mechanism system

Table 9 shows the tests that were run for the completely integrated system. The LiPo battery, charged by the solar panel, powers the system. The functionality each component, especially the LED strip and the motor/door mechanism was tested and determined to be acceptable.

TEST PLAN		TEST REPORT	
Test Description	**Acceptance Criteria**	**Pass/Fail**	**Test Result**
Power w/ LiPo battery -- lux readings < set value (700) - triggers stepper motor, door closes, LED strip is ON for 1 minute then turns OFF	Door mechanism closes, LED strip is ON for 1 minute then turns OFF, SSR rail is 9v	✓	Stepper motor rotates, door closes, LED strip turns ON, LED strip turns OFF after 1 minute, SSR rail = 8.97v
Power w/ LiPo battery -- lux readings > set value (700) - LED strip is OFF	LED strip is OFF, SSR rail is 9v	✓	LED strip is OFF, SSR rail = 8.97v
Power w/ LiPo battery -- lux readings > set value (700) - LED strip is OFF, stepper motor does nothing, door remains open	Door is still open, LED strip is OFF, SSR rail is 9v	✓	Stepper motor holds in static position, door remains open, LED strip is OFF, SSR rail = 8.97v

Table 9: Test plan for complete integrated system

Future Developments

The automated solar-powered chicken coop door system that was designed, tested, and demonstrated is functional, but there is still room for improvements and expansions. As is, the system was only demonstrated on the mock chicken coop door setup and not the actual chicken coop itself.

In order to reach the original one hour time requirement, particularly for lighting the LED strip, a larger solar panel and LiPo battery can be considered in a second design iteration.

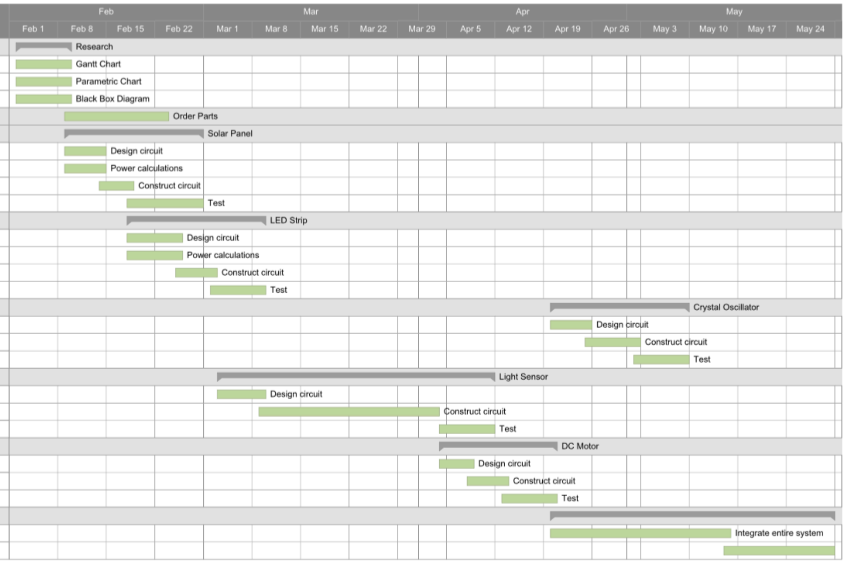
The door mechanism also has a lot of room for improvement. A sturdier, more permanent structure can be designed, as currently the stepper motor and gears are held in place with a combination of Velcro and duct tape. Because of this, the gears have a tendency to slip, resulting in the occasional, unintentional falling of the door from being held in the open position. Additionally, accurate stress analysis of the gearing system can be done to ensure maximum efficiency.

There is also the opportunity to expand the settings that can be modified by the user. The creation of an iOS or Android app would make it enable the user to easily manually override the ability of the door to open and close. The automated timing of the opening and closing of the door can also be adjusted according to the time of year. Therefore, in

winter days, the door could close earlier, as the sun sets earlier, and in summer days, the door would close at a later time.

Appendix A: Gantt Chart

	Task Name	Start Date	End Date	Duration
1	**Research**	**02/02/15**	**02/09/15**	**8**
2	Gantt Chart	02/02/15	02/09/15	8
3	Parametric Chart	02/02/15	02/09/15	8
4	Black Box Diagram	02/02/15	02/09/15	8
5	**Order Parts**	**02/09/15**	**02/23/15**	**15**
6	**Solar Panel**	**02/09/15**	**02/28/15**	**20**
7	Design circuit	02/09/15	02/14/15	6
8	Power calculations	02/09/15	02/14/15	6
9	Construct circuit	02/14/15	02/18/15	5
10	Test	02/18/15	02/28/15	11
11	**LED Strip**	**02/18/15**	**03/09/15**	**20**
12	Design circuit	02/18/15	02/25/15	8
13	Power calculations	02/18/15	02/25/15	8
14	Construct circuit	02/25/15	03/02/15	6
15	Test	03/02/15	03/09/15	8
16	**Crystal Oscillator**	**04/20/15**	**05/09/15**	**20**
17	Design circuit	04/20/15	04/25/15	6
18	Construct circuit	04/25/15	05/02/15	8
19	Test	05/02/15	05/09/15	8
20	**Light Sensor**	**03/03/15**	**04/11/15**	**40**
21	Design circuit	03/03/15	03/09/15	7
22	Construct circuit	03/09/15	04/03/15	26
23	Test	04/04/15	04/11/15	8
24	**DC Motor**	**04/04/15**	**04/20/15**	**17**
25	Design circuit	04/04/15	04/08/15	5
26	Construct circuit	04/08/15	04/13/15	6
27	Test	04/13/15	04/20/15	8
28	**Testing**	**04/20/15**	**05/30/15**	**41**
29	Integrate entire system	04/20/15	05/15/15	26
30	On-site testing	05/15/15	05/30/15	16



Appendix B: Decision Matrix

Project Name	Solar Powered Automated Chicken Coop Door	Engineering Requirements												
System Functions	Potential Component	Weight	Volume/Size	Power	Price	Accuracy/Efficiency	Documentation	Easily replaceable	Implementation feasibility	Weatherproof	**Weighted Sum +**	**Weighted Sum -**	**Weighted Sum S**	**Total Score**
	Specification Weight	10	20	25	10	18	5	5	5	2				
Solar Panel	Adafruit 6V 3.5W Large Solar Panel	S	S	S	-	+	S	S	S	S	18	10	72	30
	Adafruit 6.V 3.7W Large Solar Panel	S	S	S	+	+	S	S	S	S	28	0	72	50
	Adafruit 6V 2W Medium Solar Panel	+	+	-	S	+	S	S	S	S	48	25	27	31
	SparkFun Solar Cell Large - 8V 2.5W	S	S	-	S	S	S	S	S	S	0	25	75	-3
	SparkFun Solar Cell Huge - 8V 5.2W	S	S	+	-	S	S	S	S	S	25	10	65	35
	Blue Solar Charger SL8585mm										0	0	0	0
Light Sensor	Adafruit TSL2561	-	-	S	S	+	S	S	-	S	18	35	47	-3
	Sharp GA1A12S202 Log-scale Analog Light Sensor	+	S	S	S	+	S	S	+	S	33	0	67	53
	GL5528 CdS Photoconductive Cell	+	+	S	+	S	S	S	+	S	45	0	55	62
	Everlight ALS-PT19 Light Sensor	+	+	S	+	S	S	S	+	S	45	0	55	62
	API PDV-P8001 CdS Photoconductive Photocell	+	+	S	+	S	S	S	+	S	45	0	55	62
LED Strip	Adafruit Cool White 60 LED Weatherproof Flexi-Strip (1 m)	S	S	S	-	S	+	S	S	+	7	10	83	22
	LE Lampux Flexible LED Strip Lights (5 m)	S	S	S	+	S	S	S	S	-	10	2	88	34
	HitLights Cool White SMD3528 LED Strip	S	S	S	+	S	S	S	S	-	10	2	88	34
	SparkFun LED RGB Strip - Sealed (1 m)	S	S	+	+	S	S	S	S	+	37	0	63	56
Microcontroller	Arduino Uno	S	S	S	S	S	+	S	S	S	5	0	95	34
	Arduino Micro	+	+	S	S	S	+	S	S	S	35	0	65	55
	Raspberry Pi Model B	-	-	S	+	+	+	S	S	S	33	30	37	14
	Raspberry Pi Model A	-	-	S	S	+	+	S	S	S	23	30	47	7.1
	BeagleBone Black	-	-	S	-	+	S	S	S	S	18	40	42	-9
	TI MSP430 Launchpad	+	S	+	+	-	S	S	S	S	45	18	37	38
DC-to-DC Voltage Regulator IC	TI TPS55330	S	S	S	S	+	+	S	S	S	23	0	77	46
	TI TPS55340	S	S	S	S	S	+	S	S	S	5	0	95	34
	TI TPS61175	S	S	S	S	S	+	S	S	S	5	0	95	34
	AMS AS1345D	S	S	-	+	S	-	-	S	S	10	35	55	-9
	TI TPS6734IDR	S	S	-	-	-	S	S	S	S	0	53	47	-39
	TI LM2585T-12	S	S	+	-	S	S	S	S	S	25	10	65	35
Solid State Relay	Sharp COM-10636 SSR	S	S	+	S	-	S	S	S	S	25	18	57	24
	Panasonic AQV252GA	S	S	S	-	-	S	S	S	S	0	28	72	-6
	Kudom KSA240D2-5	S	S	S	-	S	S	S	S	S	0	10	90	17
	Vishay VO14642AT	S	S	S	+	+	S	S	S	S	28	0	72	50
	International Rectifier PVDZ172NSPBF	S	S	-	-	S	S	S	S	S	0	35	65	-16
	Omrom G3MC-202P-DC12	S	S	S	S	S	S	S	S	S	0	0	100	30

Appendix C: Product Comparison Charts

Microcontroller Comparisons

Product	Manufacturer	Price ($)	Weight	Size	Input Voltage Range	Current Draw	GPIO Pins	Flash Memory	SRAM	Clock Speed
Arduino Uno	Arduino	$24.95	25 g	68.6mm x 53.4mm	6-20V		14 (6 PWM output)	32 KB	2 KB	16 MHz
Arduino Micro (Atmega32u4)	Arduino	$24.95	13 g	48mm x 18mm	6-20V		20 (7 PWM output)	32 KB	2.4 KB	16 MHz
Raspberry Pi Model B	Raspberry Pi	$39.95	39.28 g	85mm x 56mm x 17mm		650 mA	40		512 MB	
Raspberry Pi Model A	Raspberry Pi	$29.95	30.8 g	85mm x 56mm x 15mm		300 mA	26		256 MB	
BeagleBone Black	Element 14	$55.00	40.55 g	89mm x 54mm x 15mm			69	4 GB	512 MB	
MSP430G2553 Launchpad	Texas Instruments	$9.99			1.8-3.6V	230 uA	24	16 KB	512 B	16 MHz

Solar Panel Comparisons

Product	Manufacturer	Price ($)	Weight	Size	Cell Efficiency	Cell Type	Current Output	Peak Power
Large 6V 3.4W Solar Panel	Adafruit	$44.95	143 g	4.4" x 8.4" x 0.18"	17%+	Monocrystalline	530 mA	3.4W
Blue Solar Charger SL8585mm								
Large 6V 3.7W Solar Panel	Adafruit	$29.95	150 g	6.7" x 6.7" x 0.1"	17%	Monocrystalline	550 mA	3.65W
Medium 6V 2W Solar Panel	Adafruit	$34.95	90 g	4.4" x 5.4" x 0.18"	17%	Monocrystalline	330 mA	2W
Solar Cell Large - 8V 2.5W	SparkFun	$34.95		4" x 7"	15-15.2%	Monocrystalline	310 mA	2.5W
Solar Cell Huge - 8V 5.2W	SparkFun	$44.95		7.09" x 8.66"	15-15.2%	Monocrystalline	650 mA	5.2W

Light Sensor Comparisons

Product	Manufacturer	Price ($)	Weight	Size	Digital/ Analog	Interfaces	Current Draw	Temperature Range	Peak Sensitivity	Supply Voltage
TSL2561 Digital Light Sensor	Adafruit	$5.95			Digital	I2C	0.5mA when actively sensing, < 15uA when in powerdown mode	-30 to 80 *C		2.7V to 3.6V
PDV-P8001 CdS Photoconductive Photocell	Adafruit	$0.95	0.25 g	4.46mm x 5mm x 2.09mm	Analog	Resistor -> Voltage (2.5V when lights, GND when dark)	< 1mA on average	-30 to 75 *C	520 nm	MAX 150V
GA1A12S202 Log-scale Analog Light Sensor	Adafruit	$3.95	0.2 g	10mm x 13mm x 1.5mm	Analog	Resistor -> Voltage (3V when lights, GND when dark)	1mA	-40 to 85 *C	555 nm	2.3V to 6V
GL5528 CdS Photoconductive Cell	SparkFun	$1.50		4.3mm x 5.1mm x 2.4mm	Analog	Resistor -> Voltage (3V when lights, GND when dark)		-30 to 70 *C	540 nm	MAX 150V
ALS-PT19 Light Sensor	Everlight	$0.95		1.7mm x 0.8mm x 0.6mm	Analog	Resistor -> Voltage (Vcc - 0.4V when lights, GND when dark)		-40 to 85 *C	630 nm	2.5V to 5.5V

LED Strip Comparisons

Product	Manufacturer	Price ($)	Length	# of LEDS	Voltage Required	Current Draw (per meter)	Waterproof
Cool White 60 LED Weatherproof Flexi-Strip (1m)	Adafruit	$19.95	1 m	60	9-12V	1200 mA	Yes
Lampux Flexible LED Strip Lights (5m)	Lighting EVER	$7.99	5 m	300	12V	--	No
Cool White SMD3528 LED Strip	HitLights	$10.99	5 m	300	12V	--	No
LED RGB Strip - Sealed (1m)	SparkFun	$14.95	1 m	60	12V	1800 mA	Yes
LED Light Strips (18 SMDs/ft)	SuperBrightLeds	$27.95	1 m	60	12V	960 mA	No
5050 Flexible LED Ribbon Light	The Led Light	$23.85	1 m	60	12V	1200 mA	No

Voltage Regulator Comparisons

Product	Manufacturer	Price ($)	Size	Input Voltage Range	Output Voltage Range	Iout Max	Efficiency	Shutdown Supply Current	Switching Freq Max
TPS55330	TI	$1.75	3mm x 3mm	2.9V - 16V	2.9V - 22V	500 mA	96%	2.7 uA	1200 kHz
TPS55340	TI	$1.85	3mm x 3mm	2.9V - 32V	2.9V - 38V	500 mA	90%	2.7 uA	1200 kHz
TPS61175	TI	$1.60		2.9V - 18V	2.9V - 38V	300 mA	93%	1.5 uA	2640 kHz
AS1345D	AMS	$0.49		2.9V - 5V	5.0V - 18V	40 mA	90%		
TPS6734IDR	TI	$3.86		2.7V - 11V	12V	120 mA	86%	3 uA	170 kHz
LM2585T-12	TI	$6.75		4V - 40V	12V	3A	93%		100 kHz

Solid State Relay Comparisons

Product	Manufacturer	Price ($)	Vac Max	Trigger Current Min	Output Current Max	ON-state voltage	Turn-on/off Time
COM-10636 SSR	Sharp	$4.95	500 V	8 mA	8 A	1.5 V	10 ms
AQV252GA	Panasonic	$6.44	60 V	60 mA	2.5 A	1.5 V	5 ms
KSA240D2-5	Kudom	$6.95	280 V	15 mA (MAX)	2 A	4 V	1 ms
VO14642AT	Vishay	$2.66	60 V	2 mA	2 A	1.3 V	800 us
PVDZ172NSPBF	International Rectifier	$7.85	60 V	10 mA	1.5 A		2 ms
G3MC-202P-DC12	Omrom	$4.81	264 V		2 A	1.6 V	1 ms

Appendix D: Program Code

```
#include <Wire.h>
#include "RTClib.h"

// Settings - door time duration
#define doorSettingOut 7  // D7
#define doorSettingIn  1  // A1

// Settings - LED time duration
#define ledSettingOut  8  // D8
#define ledSettingIn   0  // A0

// Settings - photocell sensitivity (potentiometer)
#define potSettingOut  1  // D1
#define potSettingIn   3  // A3

// Photocell
#define photocell      2  // A2

// Real time clock IC
#define clockSDA       4  // A4
#define clockSCL       5  // A5

// Solid state relay
#define SSR            9  // D9

// Stepper motor driver
#define motorDIR       2  // D2
#define motorSTEP      3  // D3
#define motorMS1       4  // D4
#define motorMS2       5  // D5
#define motorEN        6  // D6

#define DEFAULT_PHOTOCELL_SETTING  950
#define NUM_OPEN_MOTOR_REVS         30
#define NUM_CLOSE_MOTOR_REVS        10

#define LED_TIME_1              10  // 10 minutes
#define LED_TIME_2              15  // 15 minutes
#define LED_TIME_3              20  // 20 minutes
#define DOOR_TIME_1             15  // 15 minutes
#define DOOR_TIME_2             30  // 30 minutes
#define DOOR_TIME_3             60  // 60 minutes

RTC_DS1307 RTC;
int photocellSetting = DEFAULT_PHOTOCELL_SETTING;
int doorSetting = DOOR_TIME_1;
int ledSetting = LED_TIME_1;

int motorSpeed = 750;

/* if true, check if future event has been reached
   if false, get photocell reading */
boolean onTimer = false;
boolean ledDone = false;
boolean doorDone = false;

DateTime ledEvent;
DateTime doorEvent;

void setup() {
  // Set pin outputs
  pinMode(doorSettingOut, OUTPUT);
  pinMode(ledSettingOut, OUTPUT);
  pinMode(potSettingOut, OUTPUT);
  pinMode(SSR, OUTPUT);
  pinMode(motorSTEP, OUTPUT);
  pinMode(motorDIR, OUTPUT);
  pinMode(motorMS1, OUTPUT);
  pinMode(motorMS2, OUTPUT);
```

```
  pinMode(motorEN, OUTPUT);

  resetMotorPins();

  Serial.begin(9600);
  Wire.begin();
  RTC.begin();

  if (!RTC.isrunning()) {
    // Set RTC to date and time set on computer at time of compile
    RTC.adjust(DateTime(__DATE__, __TIME__));
  }

  getDoorSetting();
  getLedSetting();
  getPotSetting();
}

void loop() {
  if (onTimer == true) {
    DateTime now = RTC.now();

    if (now.unixtime() == ledEvent.unixtime()) {
      // Turn off LED strip
      digitalWrite(SSR, LOW);
      ledDone = true;
    }

    if (now.unixtime() == doorEvent.unixtime()) {
      // Close door
      rotateMotor(LOW);
      doorDone = true;
    }

    if (ledDone == true && doorDone == true) {
      ledDone = false;
      doorDone = false;
      onTimer = false;
    }
  }
  else {
    // Get photocell reading
    int photocellReading = analogRead(photocell);
    Serial.print("Analog reading = ");
    Serial.println(photocellReading);

    // Transition from light to dark
    if (photocellReading < photocellSetting) {
      DateTime now = RTC.now();

      ledEvent = now.unixtime() + (ledSetting * 60);
      doorEvent = now.unixtime() + (doorSetting * 60);
      onTimer = true;

      // Turn on LED strip
      digitalWrite(SSR, HIGH);
    }
    // Transition from dark to light
    else if (photocellReading > photocellSetting) {
      // Open door
      rotateMotor(HIGH);
    }
  }

  delay(1000);
}

/* Reset motor pins to default states */
void resetMotorPins() {
  digitalWrite(motorSTEP, LOW);
  digitalWrite(motorDIR, LOW);
  digitalWrite(motorMS1, LOW);
  digitalWrite(motorMS2, LOW);
```

```
  digitalWrite(motorEN, LOW);
}

/* Get door time jumper setting */
void getDoorSetting() {
  digitalWrite(doorSettingOut, HIGH);
  int sensorValue = analogRead(doorSettingIn);
  Serial.print("Door time: ");
  Serial.println(sensorValue);
  digitalWrite(doorSettingOut, LOW);

  if (sensorValue < 50) {
    doorSetting = DOOR_TIME_2;
  }
  else if (sensorValue > 50 && sensorValue < 500) {
    doorSetting = DOOR_TIME_1;
  }
  else {
    doorSetting = DOOR_TIME_3;
  }

  Serial.println(doorSetting);
}

/* Get LED time jumper setting */
void getLedSetting() {
  digitalWrite(ledSettingOut, HIGH);
  int sensorValue = analogRead(ledSettingIn);
  float voltage = sensorValue * (5.0 / 1023.0);
  Serial.print("LED: ");
  Serial.println(sensorValue);
  digitalWrite(ledSettingOut, LOW);

  if (sensorValue < 500) {
    ledSetting = LED_TIME_2;
  }
  else if (sensorValue > 500 && sensorValue < 900) {
    ledSetting = LED_TIME_1;
  }
  else {
    ledSetting = LED_TIME_3;
  }

  Serial.println(ledSetting);
}

/* Get potentiometer setting */
void getPotSetting() {
  digitalWrite(potSettingOut, HIGH);
  int sensorValue = analogRead(potSettingIn);
  photocellSetting = sensorValue * (1000 / 1023);
  Serial.print("Potentiometer: ");
  Serial.println(sensorValue);
  Serial.println(photocellSetting);
  digitalWrite(potSettingOut, LOW);
}

void rotateMotor(int dir) {
  Serial.println("Rotate motor");
  int steps = (dir == LOW) ? 360 * 0.556 * NUM_CLOSE_MOTOR_REVS : 360 * 0.556 *
NUM_OPEN_MOTOR_REVS;

  digitalWrite(motorDIR, dir);

  for (int x = 0; x < steps; x++) {
    digitalWrite(motorSTEP, HIGH);  // Trigger step
    delayMicroseconds(motorSpeed);
    digitalWrite(motorSTEP, LOW);   // Pull STEP pin low so it can be triggered again
    delayMicroseconds(motorSpeed);
  }
}
```

Appendix E: Test Plan

TEST PLAN					TEST REPORT	
Item No	**Specification**	**Test Description**	**Acceptance Criteria**	**Pass/Fail**	**Test Result**	**NOTES**
1	9V step-up/step-down voltage regulator	Power w/ 3.3v -- voltage output is 9v	Voltage output = 9v	P	Voltage output = 8.97v	
2		Power w/ 3.7v -- voltage output is 9v	Voltage output = 9v	P	Voltage output = 8.97v	
3	Light sensor	Changes in lux readings from sensor when light is blocked/unblocked	Values printed in serial monitor fluctuate according to sensor being blocked/unblocked	P	Values printed in serial monitor fluctuate according to sensor being blocked/unblocked	
4		Block light sensor for indoors -- lux readings < set value (700) -- digital output = 5v	Output on DIGITAL PIN 7 = 5v	P	DIGITAL PIN 7 output = 5.04v	In indoor lab, unblocked light and blocked light lux values were approximately 800 and 600, respectively
5		Unblock light sensor for indoors -- lux readings > set value (700) -- digital output = 0v	Output on DIGITAL PIN 7 = 0v	P	DIGITAL PIN 7 output = 50mV	In indoor lab, unblocked light and blocked light lux values were approximately 800 and 600, respectively
6		Block light sensor for outdoors -- lux readings < set value (850) -- digital output = 5v	Output on DIGITAL PIN 7 = 5v	P	DIGITAL PIN 7 output = 5.03v	In outdoor setting, unblocked light and blocked light lux values were approximately 900 and 800, respectively
7		Unblock light sensor for outdoors -- lux readings > set value (850) -- digital output = 0v	Output on DIGITAL PIN 7 = 0v	P	DIGITAL PIN 7 output = 50mV	In outdoor setting, unblocked light and blocked light lux values were approximately 900 and 800, respectively
8	Solid state relay	Power w/ 5v -- get voltage drop across current-limiting resistor and calculate current across resistor	Current across resistor ≈ 10 mA	P	Voltage drop = 3.62v, current across resistor = 10.9 mA	
9		Power w/ 5v and 12v rail on output -- load output is 12v	Load output on PIN 5 is 12v	P	PIN 5 output = 12.09v	
10		Turn on/off 5v power source w/ 12v rail -- load output should turn on/off	5v power source ON, load output is 12v; 5v power source OFF, load output is 0v	P	5v power source ON, PIN 5 output = 12.09v; 5v power source OFF, PIN 5 output = 0v	
11	LED Strip	Power w/ 12v from power supply	LED strip is ON	P	LED strip is ON	
12		Power w/ 9v from power supply	LED strip is ON	P	LED strip is ON and noticeably dimmer than w/ 12v power supply	
13	DC motor	DC motor has bi-directional rotation	DC motor rotates clockwise, DC motor rotates counterclockwise	P	DC motor rotates clockwise, DC motor rotates counterclockwise	
14	Stepper motor	Stepper motor has bi-directional rotation	Stepper motor rotates clockwise, stepper motor rotates counterclockwise	P	Stepper motor rotates clockwise, stepper motor rotates counterclockwise	
15	LiPo battery/9v voltage regulator	Power w/ fully-charged LiPo battery, voltage output is 9v	Voltage output = 9v	P	Voltage regulator output = 8.97v	
16	Solar panel/LiPo battery	Solar panel charges LiPo battery, voltage load output = 3.7v	LiPo charger charging light = orange, voltage load output = 3.7v	P	LiPo charger charging light = orange, LiPo charger load output = 3.9v	
17	Solar panel/LiPo battery/9v voltage regulator	Solar panel charges LiPo battery, LiPo charger voltage load output = 3.7v, output of voltage regulator = 9v	LiPo charger charging light = orange, LiPo charger voltage load output = 3.7v, voltage regulator output = 9v	P	LiPo charger charging light = orange, LiPo charger load output = 3.9v, voltage regulator output = 8.97v	
18	Solar panel/LiPo battery/9v voltage regulator/Arduino	Power w/ LiPo battery -- digital output = 5v	Output on DIGITAL PIN 7 = 5v	P	DIGITAL PIN 7 = 5.03v	
19	Solid state relay/LED strip	Power w/ 5v and 12v rail on output -- LED strip is on	LED strip is ON	P	LED strip is ON	
20		Turn on/off 5v power source w/ 12v rail -- load output should turn on/off	5v power source ON, LED strip is ON; 5v power source OFF, LED strip is OFF	P	5v power source ON, LED strip is ON; 5v power source OFF, LED strip is OFF	
21		Power w/ 5v and 9v rail on output -- LED strip is on, but less bright	LED strip is ON	P	LED strip is ON and noticeably dimmer than w/ 12v rail	
22		Turn on/off 5v power source w/ 9v rail on output -- load output should turn on/off	5v power source ON, LED strip is ON; 5v power source OFF, LED strip is OFF	P	5v power source ON, LED strip is ON and noticeably dimmer than w/ 12v rail; 5v power source OFF, LED strip is OFF	
23	Arduino/Solid state relay/LED strip/Light sensor/Real-time clock	Power w/ 5v from Arduino and 12v rail on output, lux readings < set value (700) -- LED strip is ON	LED strip is ON	P	LED strip is ON, current draw = 92 mA	
24		Power w/ 5v from Arduino and 12v rail on output, lux readings > set value (700) -- LED strip is OFF	LED strip is OFF	P	LED strip is OFF	
25		Power w/ 5v from Arduino and 12v rail on output -- LED strip is ON for 1 minute then turns OFF	lux reading < set value (700), LED strip is ON for 1 minute then turns OFF	P	LED strip is ON, current draw = 92 mA, LED strip turns OFF after 1 minute	
26		Power w/ 5v from Arduino and 9v rail on output, lux readings < set value (700) -- LED strip is ON	LED strip is ON	P	LED strip is ON and noticeably dimmer than w/ 12v rail, current draw = 32 mA	
27		Power w/ 5v from Arduino and 9v rail on output, lux readings > set value (700) -- LED strip is OFF	LED strip is OFF	P	LED strip is OFF	
28		Power w/ 5v from Arduino and 9v rail on output -- LED strip is ON for 1 minute then turns OFF	lux reading < set value (700), LED strip is ON for 1 minute then turns OFF	P	LED strip is ON, current draw = 32 mA, LED strip turns OFF after 1 minute	
29	Arduino/Light sensor/DC motor	Power w/ 5v from Arduino, lux readings < set value (700) -- triggers DC motor	DC motor is triggered	P	DC motor rotates	
30		Power w/ 5v from Arduino, lux readings > set value (700) -- DC motor does nothing	System remains unchanged	P	DC motor holds in static position	
31	Arduino/Light sensor/DC motor/Door mechanism	Power w/ 5v from Arduino, dark to light lux value transition -- triggers DC motor, door opens	Door mechanism opens	F	DC motor rotates, unable to lift door open	
32		Power w/ 5v from Arduino, lux readings > set value (700) -- DC motor does nothing, door remains open	System remains unchanged, door remains in open position	F	Untested because DC motor -> door opens test failed	
33		Power w/ 5v from Arduino, light to dark lux value transition -- triggers DC motor, door closes	Door mechanism closes	F	Untested because DC motor -> door opens test failed	
34	Arduino/Light sensor/Stepper motor	Power w/ 5v from Arduino, lux readings < set value (700) -- triggers stepper motor	Stepper motor is triggered	P	Stepper motor rotates	
35		Power w/ 5v from Arduino, lux readings > set value (700) -- stepper motor does nothing	System remains unchanged	P	Stepper motor holds in static position	
36	Arduino/Light sensor/Stepper motor/Door mechanism	Power w/ 5v from Arduino, 9v on stepper motor driver Vcc, dark to light lux value transition -- triggers stepper motor, door opens	Door mechanism opens	P	Stepper motor rotates, door opens	
37		Power w/ 5v from Arduino, 9v on stepper motor driver Vcc, lux readings > set value (700) -- stepper motor does nothing, door remains open	System remains unchanged, door remains in open position	P	Stepper motor holds in static position, door remains open	
38		Power w/ 5v from Arduino, 9v on stepper motor driver Vcc, light to dark lux value transition -- triggers stepper motor, door closes	Door mechanism closes	P	Stepper motor rotates, door closes	
39	Arduino/Solar panel/LiPo battery/9v voltage regulator/Solid state relay/LED strip/Light sensor	Power w/ LiPo battery -- lux readings < set value (700) - LED strip is ON for 1 minute then turns OFF	LED strip is ON for 1 minute then turns OFF, SSR rail is 9v	P	LED strip is ON for 1 minute then turns OFF, SSR rail is 8.97v	
40		Power w/ LiPo battery -- lux readings > set value (700) - LED strip is OFF	LED strip is OFF, SSR rail is 9v	P	LED strip is OFF, SSR rail = 8.97v	
41		Power w/ LiPo battery -- lux readings < set value (700) - LED strip is ON for 1 hour then turns OFF	LED strip is ON for 1 hour then turns OFF, SSR rail is 9v	P	LED strip is ON for 1 hour then turns OFF, SSR rail = 8.97v	
42	Arduino/Solar panel/LiPo battery/9v voltage regulator/Solid state relay/LED strip/Light sensor/Stepper motor/Door mechanism **[Entire System]**	Power w/ LiPo battery -- lux readings < set value (700) - triggers stepper motor, door closes, LED strip is ON for 1 minute then turns OFF	Door mechanism closes, LED strip is ON for 1 minute then turns OFF, SSR rail is 9v	P	Stepper motor rotates, door closes, LED strip turns ON, LED strip turns OFF after 1 minute, SSR rail = 8.97v	
43		Power w/ LiPo battery -- lux readings > set value (700) - LED strip is OFF	LED strip is OFF, SSR rail is 9v	P	LED strip is OFF, SSR rail = 8.97v	
44		Power w/ LiPo battery -- lux readings > set value (700) - LED strip is OFF, stepper motor does nothing, door remains open	Door is still open, LED strip is OFF, SSR rail is 9v	P	Stepper motor holds in static position, door remains open, LED strip is OFF, SSR rail = 8.97v	