ECE 445 Senior Design Project
Spring 2013

Mail Notification System

Ethan Ahn-Kang, Ryan Park, Dickson Salim

Abstract

For our senior design project, we designed and developed a mail notification system that detects mail with infrared sensors and sends the corresponding message from the mailbox unit to an LCD at home and user's email. The system is comprised of mailbox unit, which has infrared sensor array and buttons for locations other than the mailbox (e.g. On the porch, at the post office), and receiving unit, which displays the message received from the mailbox unit and sends the same message to the user's email account. Each unit consists of a XBEE as the wireless communication system and an Arduino Uno as the microcontroller. The mailbox unit is powered by a 9V battery, which can last up to approximately 4 months depending on the frequency of usage with the receiving unit powered by a wall outlet. With our Mail Notification System, a user can be notified as soon as expected mails or packages arrive. Furthermore, reducing the time to check an empty mailbox conserves energy.

Contents

1. Introduction 4
 1.1 Project Overview 4
 1.2 High-level Design 4
 1.3 Design Revision 5
  1.3.1 Reflect vs. Block 5
  1.3.2 Number of Infrared Sensors 6
  1.3.3 Utilization of 555 Timer 6

2. Design 7
 2.1 Mailbox Unit 7
  2.1.1 Infrared Sensor 7
  2.1.2 Power 7
  2.1.3 555 Timer 8
  2.1.4 Buttons 10
  2.1.5 Microcontroller 11
  2.1.6 Wireless Communication 12

 2.2 Receiving Unit 12
  2.2.1 LCD 12
  2.2.2 Wireless Communication 13
  2.2.3 Email 13

3. Requirements and Verifications 14

4. Costs 18
 4.1 Parts 18
 4.2 Labor 18

5. Conclusion 19
 5.1 Future Work 19
 5.2 Ethics 20

6. Reference 21

7. Appendix 22

1. Introduction

1.1 Project Overview

Our motivation for this project was to design and develop a product that can make our lives more expedient in various ways. The 'Mail Notification System' is a simple, yet elegant and convenient invention. It detects mails in the mailbox using infrared sensors and recognizes location buttons to notify the user as soon as mails and/or packages arrive.

1.2 High-level Design

The design consists of two main components, the mailbox unit and the receiving unit. Mailbox unit is responsible for detecting mail using infrared sensors and transmitting wireless signals. Also, it has buttons on the front to indicate the item location other than the mailbox. On the other hand, the receiving unit operates based on the signal received from mailbox and displays proper message along with an email notification.

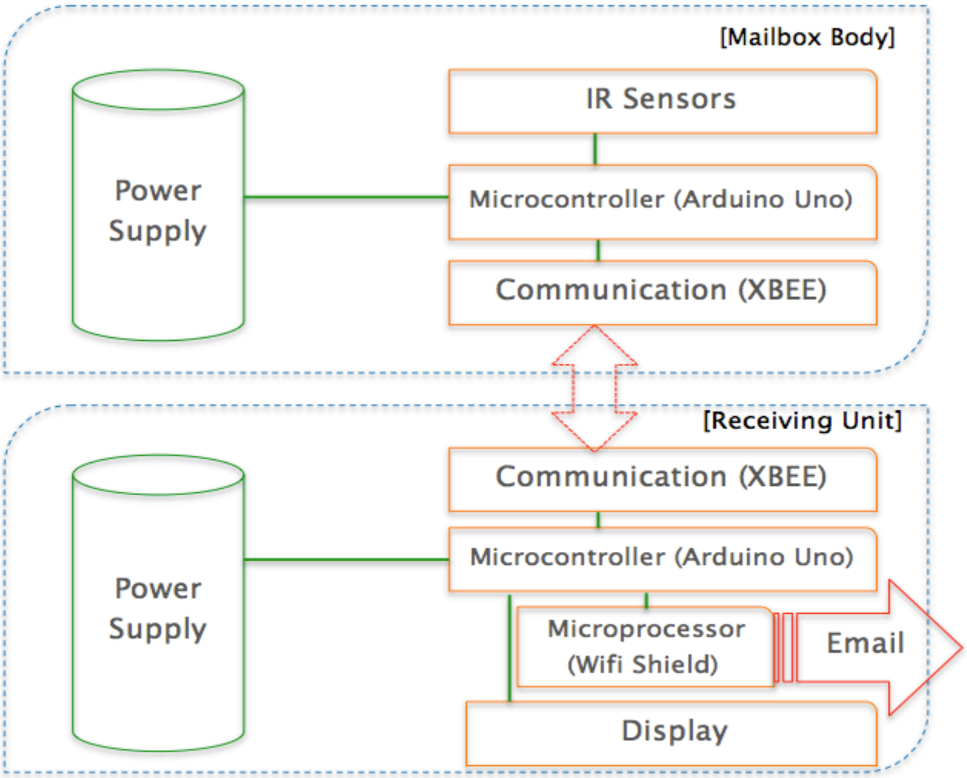


Figure 1: Block Diagram

Features	Benefits
• Easy installation & Plug and Play • Lightweight and portable • Durable • Arrival time and location display • Email notification	• Save unnecessary time and energy to check an empty mailbox • Ability to retrieve important and/or expected mail as quickly as possible

Table 1: Features and benefits

1.3 Design Revision

We faced several design challenges throughout the semester and revised the design several times in response to these challenges. First, we switched our sensing method of the mailbox from reflection to blockage for an increase in sensitivity. Secondly, we decreased the number of sensors to efficiently utilize the LED-emitting angle. Lastly, we added monostable mode of 555 timer to make the mailbox operate for one minute after the mailbox door opens, which increased the time for wireless communication and conserved battery life.

1.3.1 Reflect vs. Block

Sensitivity of sensors is one of the most critical parts of the design. Among many detection mechanisms, two prominent mechanisms were emerged on our discussion table, which were proximity sensor mechanism and blockage method. In the beginning, proximity sensor method was strongly considered and experimented due to the freedom of sensor maneuvering and the experience each member had obtained from ECE 110. However, the unpredicted technical hindrances such as sensitivity of sensors, mailbox cavity size, and operation difficulties put us on marathon discussions and numerous experiments with different types of proximity sensors. After long nights of frustrations, proximity method was no longer in use in the project. Blocking method was now up and running for our topic and the team quickly transitioned and adapted to new changes. Case was evolved to satisfy the new sensor mechanism and the design became much more reliable and accurate.

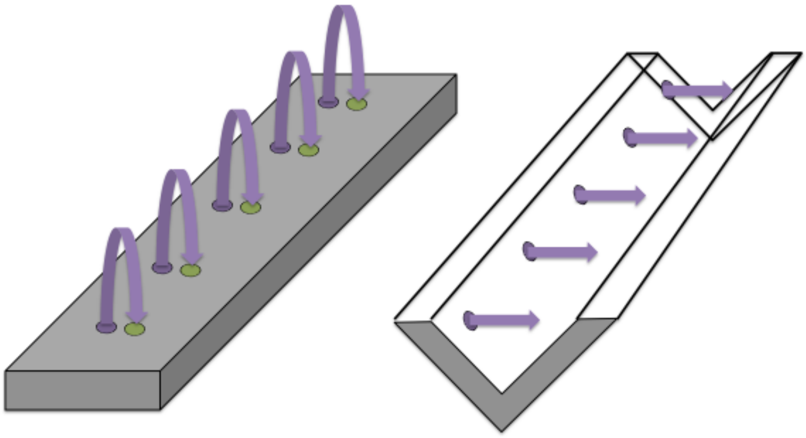


Figure 2: Flat (Reflect) vs. V-Shape (Block)

1.3.2 Number of Infrared Sensors

At first, effective sensor placement on the panel seemed to be simple; however, this was a game changer. Usually people tend to think the more the better. “Six sensors would suffice to cover the whole mailbox”, “Infrared sensors and detectors are cheap, why not use more sensors?” Questions and comments as such these were brought up and easily consented. Technical specification was simply ignored by the simplicity look of the design. It did not take long for us to realize how ignorant we were. If the mail is too small to cover the whole six sensors, the detector is still able to receive signals from the other uncovered sensors. The Infrared Light Emitting Diode (IR LED) was not emitting straight to the photo detectors that was aligned. Instead, rit was scattering as wide as 50 degrees angle. Datasheet of IR sensors were reviewed again and the number of sensors was reduced to three to enhance sensitivity.

1.3.3 Utilization of 555 Timer

Originally, battery time saving was considered using a simple mechanical switch, which normally breaks the circuit when the mail door was closed and operates only when the door was opened. However, if the mailbox was opened and closed in a fraction of a second for any odds, then the system would not have sufficient time to send signals. Thus, considering the processing time of the wireless communication unit, we decided to add monostable mode of 555 timer to extend the operational time to one minute. This implementation of 555 timer definitely increased the consistency and reliability of the wireless communication from the mailbox to the receiving unit.

2. Design

2.1 Mailbox Unit

Mailbox unit is the transmitting unit that detects mail and sends the message using wireless communication. Mailbox unit is composed of infrared sensors, 555 timers, buttons, microcontroller, and wireless communication unit.

2.1.1 Infrared Sensor

Three IR sensors are placed on the V-shaped plate of the mailbox unit to detect mails. IR 950nm LEDs were used to emit infrared lights and VS1838B IR receivers were used to receive the infrared lights. Sensors will output either high, when the mailbox is empty, or low, when the mailbox detects the mail. Outputs of greater than or equal to 0.1mV were considered as high, and below 0.1mV were considered low. The outputs of each IR receivers are connected to analog pins of the microcontroller to translate the voltages to either high or low. These digitized outputs will be coded into the controller unit for further action.

2.1.2 Power

A 9V battery in the mailbox unit supplies power. A mechanical switch and a 555 timer was connected in series with battery and acted as a switch to increase the wireless communication operation time and to prevent unnecessary power drainage.

<u>Battery Lifetime Calculations</u>

Assumptions:

- Mailman comes from Monday to Saturday per week (6 days per week)
- Mailman spends less than 1 minute to put mails in the mailbox (Assuming the mailman spends more than a minute to put mails in apartment mailbox, and less than a minute to put mails in single household mailbox)
- For current draw of ≈100mA, 9 V battery provides approximately 12000 J

$$Battery\ Lifetime = \frac{Battery\ energy\ capacity\ (J)}{(\frac{52weeks}{year})(\frac{0.017hours}{week})(\frac{3600seconds}{hour})(\frac{4.5volts \times ILoad}{week})}$$

Arduino Uno Current per I/O pin≈40mA

XBEE Current draws≈50mA

$$Battery\ Lifetime = \frac{12000J}{(\frac{52weeks}{year})(\frac{0.017hours}{week})(\frac{3600seconds}{hour})(\frac{4.5volts \times 1.48A}{week})}$$

Thus, with 9V battery, battery lifetime for the mailbox unit will be approximately 0.566 year. (Approximately 6 months)

Mailbox Unit Operation Time (using 555IC)

For longevity of battery life, we utilized 555 IC to control the circuit for 60 seconds.

The decay time was calculated using the formula below.

$$\tau = 1.1RC = 60K\Omega * 100\mu F = 60\ seconds$$

Applying 555IC to our mailbox unit will increase the operation time from 60 seconds to 120 seconds. (0.033 hours per week)

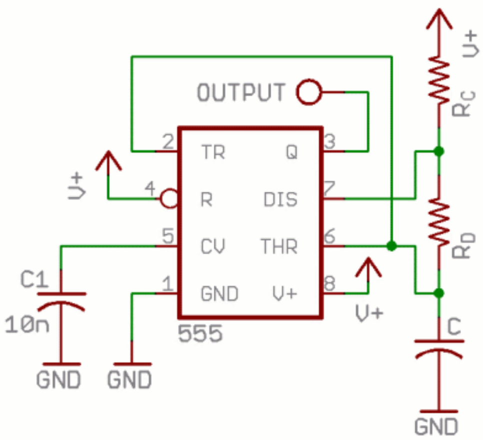
Therefore,

$$Battery\ Lifetime = \frac{12000J}{(\frac{52weeks}{year})(\frac{0.033hours}{week})(\frac{3600seconds}{hour})(\frac{4.5volts \times 1.48A}{week})}$$

≈0.292 year (which is approximately 3.5 months)

2.1.3 555 Timer

There are two 555 timers in the mailbox unit, astable mode and monostable mode. Astable mode was used in modulating IR LEDs since infrared photo detectors are fabricated to recognize infrared that is modulated at a certain frequency to improve signal to noise ratio (SNR). For instance, a random car remote control or TV remote controller will not operate the photo detector if the emitter was not modulated at a certain frequency. VS1838b photo detectors were designed to detect IR emission modulated at 38 KHz. Thus, 555 timer was used to make proper modulation for IR emitter to send corresponding emission to the receiver. To modulate the frequency, following formula was derived and used to calculate.



$$f = \frac{1.44}{(R_C + R_D + R_D) * C}$$

$$38KHz = \frac{1.44}{(140\Omega + 120\Omega + 120\Omega) * 0.1\mu F} = 37.894Hz \approx 38KHz$$

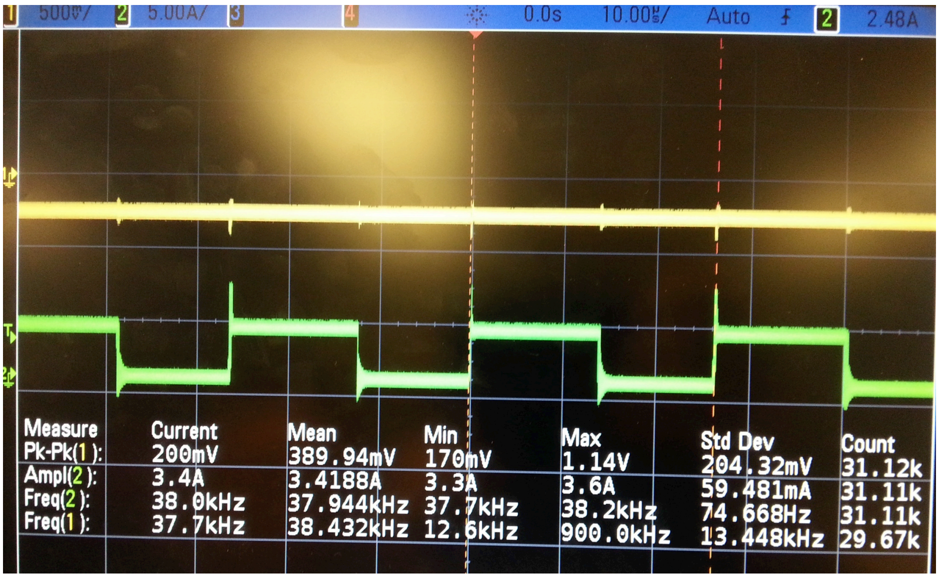


Figure 3: Astable Mode of 555 Timer Modulation at 38KHz

Monostable mode: As the mailbox door opens, mechanical switch triggers the 555 timer and connects the system circuit to the 9V battery. Thus, once the mechanical switch triggers the system, monostable mode of 555 timer holds the power high and operates the mailbox unit for a one minute.

Time Out Delay was calculated using the formula below.

$$\tau = 1.1 * R * C = 1.1 * 550K\Omega + \ 100\mu F = 60.5\ s$$

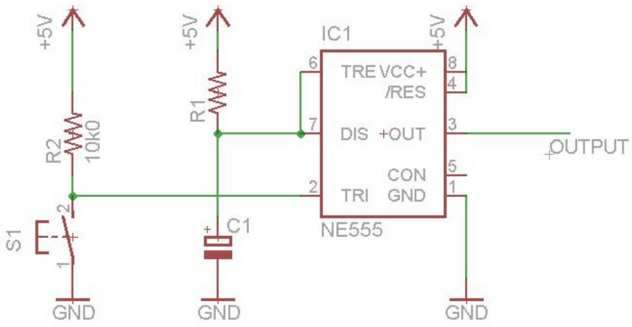


Figure 4: Monostable Mode of 555 Timer

2.1.4 Buttons

Two mini pushpin button switches are used to locate the packages outside the mailbox. Outputs of the buttons are directly connected to the analog pins of the microcontroller. Once the button is pressed, the microcontroller receives digital high, 4.95V, and take the value into the logic.

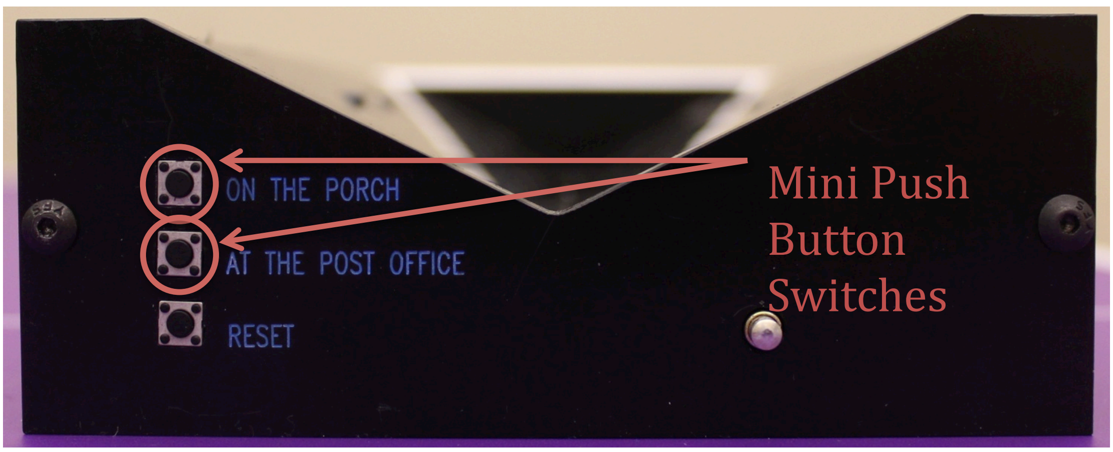


Figure 5: Mini Push Button Switches on the Mailbox Unit

2.1.5 Microcontroller

The controller unit will be programmed and operated to relay the signal to the transmitting unit. Output from the sensors and different options that were selected by the postman will be consolidated and the microcontroller will react accordingly. The choice of signal will be 1: Left on porch, 2: At the post office, 3: You got Mail. The logic operation expected from the microcontroller will go according to the following truth table shown below.

Expected Results:
00: no mail
01: at porch
10: pickup at post office
11: in mailbox

Sensors	Porch button (x)	Pickup button (y)	output
0	0	0	11
0	0	1	10
0	1	0	01
1	0	0	00
1	0	1	10
1	1	0	01

Output 1 = sensor'.x' + sensor.x'y
Output 0 = sensor'.y' + sensor.x.y'

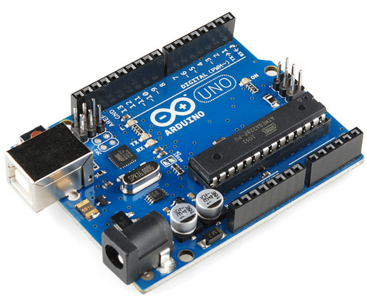


Figure 6: Arduino Uno

Arduino Uno will be used as the microcontroller. It will operate on 5 V, and has 14 digital I/O pins. The main reason to choose this over other microcontroller is because

Arduino Uno is widely used for various signal-processing projects, so it will be easier to integrate with other parts of the project. It is also based on C programming language, which we are more familiar with. Thus, Arduino Uno is probably the best choice for our kind of project.

2.1.6 Wireless Communication

XBEE was used to provide wireless capability to our design. XBEE in the mailbox was simply synchronized with the microcontroller, Arduino Uno, by placing the XBEE shield on top of the Arduino. Same pin assignments were shared to handshake and share data between the two. X-CTU program was utilized to ensure both XBEE on each end are synchronized to orchestrate at the same rate. XBEE acquired data from Arduino from designated pins and sent the data to the other side. It transmitted 17 bytes of data including pertinent information to communicate and the actual data. Our main concern was to overcome Fraday's cage created by the mailbox itself; however, the XBEE was able to transmit data through the mailbox. Furthermore, the design of antenna extension to enhance the range was taken into account when developing the mailbox unit.

2.2 Receiving Unit

2.2.1 LCD

The LCD will print out message and the time of arrival of the mail/package. It begins by checking the signal received by the receiving XBEE. Depending on the signal, different messages will be printed to the LCD to alert the user. At the same time, the microcontroller has DateTime header file that keeps track of the time as long as the system is powered. As soon as the signal is received, the microcontroller will immediately save the date and time into the assigned string. It is necessary to save the date and time into a string because this string will be used for displaying the message onto the LCD and used for the email notification system. This time string is printed on the second line of the LCD. It is necessary to only clear the LCD when new signal is received because the microcontroller always repeat itself in a loop, and will clear the screen the after the message is printed.

2.2.2 Wireless Communication

XBEE is utilized on the receiving end of the wireless communication. It will receive 17 hexadecimal containing the start hexadecimal, transmitter XBEE id, the actual data sent, and etc. However, most of the hexadecimals received are not useful. Thus, the first 15 hexadecimals are discarded and the 16th hexadecimal is assigned to a variable because the 16th hexadecimal is the hexadecimal representation of the data that is received. In the case of this project, the expected hexadecimals are 30, 31, 32, 33; where 30 is hexadecimal representation for ASCII 0, 31 represents ACII 1, 32 represents ASCII 2, and lastly 33 represents ASCII 3.

2.2.3 Email

The first step of sending notification email is connecting to an available WiFi network. In order to assist the user in assigning which WiFi network to connect, there is a SD card prepared with a text file in it that contains the network ssid, password, and recipient email address. This text file is formatted to have the network ssid on the first line, the network password on the second line, and the recipient email address on the last line. The first operation that the microcontroller will perform is open the text file in the SD card, assign the three lines to three different variables, which are ssid, password, and recEmail. Then, after connecting to the WiFi network, the microcontroller will print out the connection status in the serial monitor and try to connect to port 5000 of the web server's IP address. Port 5000 is chosen in order to use a separate port from the commonly used port, port 80, for http. On the server part, the router should allow port forwarding from public IP address to local IP address. Otherwise, the server can only be accessed from devices that are connected to the same network. Furthermore, the server has a php script that will echo to the recEmail (recipient email address), which has the same message as the one printed on the LCD.

3. Requirements and Verifications

Requirements	Verifications
Mailbox Body	
<u>Power</u>	
1. Battery supplies enough voltage	1. Measure at the terminals of the battery using multimeter to verify that it is fully charged at operating voltage 4.8V<Voltage of the battery<9V
2. Battery operates at operating voltage 4.8V<Voltage of the battery< 9V or at a threshold of 400mAh< Capacity of the battery<560mAh	2. Measure at the terminals of the battery using multimeter to verify that it is operating at operating voltage 4.8V<Voltage of the battery<9V (400mAh<Capacity of the battery<560mAh)
3. Battery is connected to the mailbox unit (connected to 9V battery snap)	3. Use voltmeter to check the voltage across each part to check sufficient voltage supplied
4. Battery supplies constant power to the circuit only for limited time period. (60 seconds)	4. Use multimeter to check the output of 5V with the error tolerance of ±0.25V
a) Resistors and Capacitors are connected in series between V_{cc} and GND	a) Ensure the resistor and capacitor has the right value for the desired decaying time for the limited time of circuit operation. Resistance must be 120 KΩ and Capacitance must be 100µF for with error range ~ 3%.
b) Power button triggers the 555 IC	b) When button is released, the V_{cc} should be connected to the trigger pin (2) of 555 IC. Ensure to measure 9V ±0.5 using multimeter
c) 555 IC Outputs High for 60 seconds	c) Measure the output voltage from output pin (3) for 60 seconds and ensure the value is 5 V ±0.25V.

<u>Sensor</u> 1. Mailbox unit detects mail with IR sensors 2. IR LEDs and IR receivers are connected 3. IR LEDs emit lights 4. IR receivers receive lights	1. Check the output of voltage of IR receiver (Mail detected: low less than 130mV, Mail not detected: high greater than or equal to 130mV) 2. Check the continuity of the circuit (Vcc, GND, resistor, Arduino connection of IR LEDs and IR receivers) 3. a. Use camera to check if IR LEDs are emitting light b. Use multimeter to check the volatage across the IR LEDs (2.1V<Voltage across the IR LED<4.2V) 4. Use multimeter to check the voltage across the IR receiver a. Output low less than 130mV,when object is detected **b.** Output high greater than or equal to 130mV, when object is not detected
<u>Button</u> When one of the buttons is pressed, corresponding button should output high.	Use multimeter to measure voltage is outputting High (4.95V)
Microcontroller	
<u>Microcontroller</u> 1. The Microcontroller should take inputs from sensors, and buttons (a) The Microcontroller has to AND all inputs from sensors into one signal. 2. The Microcontroller logic circuit must output two bit signal to D12, D11 (D12 being the most significant bit) in accordance with the truth table 3. The Microcontroller has to convert the two bits output into integer	1. (a) Output from buttons and sensor unit should be connected to the microcontroller. Open the Arduino program and add print command that prints the value of the variable that holds the result of the AND operation (it should outputs 1 only when all sensors send 1). 2. Connect input for x, y with either high (5 V) / low (0 V), then add print command on Arduino program to see if the printed values from serial monitor match the table on the requirement 3. Assign test values to output1 and output0, and add print command to print out the

	converted value to see if they match.
Communication Unit	
1. Both Xbees should have the same PAN ID 2. Transmitting Xbee should send signal once when signal is not zero. 3. Receiving Xbee should receive 17 hexadecimals and extract the data received. (a) Receiving Xbee should check if there's any available signal to read (b) Receiving Xbee should check if the available signal consists of 17 hexadecimal, and starts with x7E (c) Receiving Xbee should discard the next 14 hexadecimals after hexadecimal x7E (d) Receiving Xbee should save the next hexadecimal read. (e) Receiving Xbee should convert the hexadecimal to integer.	1. Connect both Xbees to PC, and use X-CTU to check both Xbees' PAN ID. 2. Set the signal variable of transmitting Xbee to a valid value, and put the condition for checking whether to send it or not in the loop function. Connect the receiving Xbee to PC and open terminal tab on X-CTU to see if the receiver Xbee on receives the signal once. 3. (a) Connect the receiving Xbee to PC, and open the code. Look for xbee.available() in the checking condition. (b) Visually check if xbee.available() == 17 is part of the condition. (c) visually check the code for a condition to check for xbee.read() == x7E, and followed by for loop that repeats itself for 14 times, and it assigns Serial.read() to a dummy variable. (d) visually check if there is signal = Serial.read() right after the for loop. (e) assign signal variable to a valid hexadecimal representation of an integer, and print the converted value to see if the convertion is functioning properly.
Display	
LCD should be wired correctly to Arduino, 5 V power source. LCD display code should check the value of receiver's output variables. • Print the corresponding message along with date and time.	Verify the wiring schematic to the figure provided above. Open arduino program, and check the switch statements for the LCD display **1.** Assign value to the two variables in the switch statements, and check for the printed message.
Wireless Internet (Email)	
1. Code successfully compiles without errors. 2. Arduino Wifi Shield is successfully	1. Verified by software compiler. 2. PIN 7 from Wifi Shield is used to handshake with Arduino Uno, and must

linked up with Mainboard (Arduino Uno). 3. Receives correct data from the microcontroller (Arduino Uno) 4. D10 (SS for Wifi) is High and should start scanning for available Wifi network. 5. Successfully scan for available network 6. Read SSID and Password from a .txt file in SD Card to connect to the internet 7. Web server should be up. 8. Wi-Fi shield should successfully send the GET request to server. 9. Web server should send the email to the recipient email address.	output high when synchronization was successful. 3. Two testing LEDs are placed on data output pins and must match with what was sent from XBee. Also, when connected to the PC, it will display transmitted data. 4. Use multimeter to measure voltage on D10 to ensure it is high. 5. Displays MAC address and available SSID 6. Type in valid SSID and Password on .txt file and print out "successful connection" upon connection along with a green LED that is connected to one of the output data pins. 7. Type in the web server IP address in web browser, and see if default web page comes out. **8.** Open the log file from the server side, and see if there is anything sent to the server. **9.** Login to the recipient email address and see if there's anything in the inbox.

4. Costs

The total cost for this project, including parts and labor, was $15423.53.

Labor	Parts	Total
$ 15120.00	$ 303.53	$ 15423.53

4.1 Parts

A list of parts purchased for the project is summarized in Table. The total cost of parts was $303.53

Part name	Qty	Price	Total
Basic 16 x 2 Character LCD : LCD 00255	1	$ 13.95	$ 13.95
Wall Adapter Power supply – 9 VDC 650 mA	1	$ 5.95	$ 5.95
Xbee Series 1 XBP24-AWI-001-ND	2	$ 32.00	$ 64.00
Xbee Explorer USB	1	$ 24.95	$ 24.95
Xbee Shield	2	$ 14.95	$ 29.90
Arduino Uno-R3	2	$ 29.95	$ 29.95
Arduino WiFi Shield	1	$ 84.95	$ 84.95
USB Cable A to B – 6 foot	1	$ 3.95	$ 3.95
Sparkfun USB Mini-B Cable – 6 foot	1	$ 3.95	$ 3.95
Solar Group E1600B00 Large Premium Steel Mailbox	1	$ 15.21	$ 15.21
50pcs Infrared LED - 60 Degree Clear 940nm	1	$ 13.95	$ 13.95
10pcs vs1838b Infrared receiver	1	$ 9.99	$ 9.99
Gino 2 Pcs 2.1 X 5.5mm Male Dc Plug to 9v Battery Clip	1	$ 2.83	$ 2.83
10kohm resister	1	$ 0.00	$ 0.00
220ohm resister	1	$ 0.00	$ 0.00
Total			$303.53

4.2 Labor

The total costs for labor are summarized in Table. Labor costs were calculated using the deal but realistic salary for each team member

Name	# of week	# of hours per week	Cost / Hour	Total Cost
Dickson Salim	12	12	$ 35.00	$ 5040.00
Ethan Ahn-Kang	12	12	$ 35.00	$ 5040.00
Ryan Park	12	12	$ 35.00	$ 5040.00
			Total	$ 15120.00

5. Conclusion

Overall, our project was very successful. Both mailbox unit and receiving unit functioned as we aimed in the beginning. Sensors successfully distinguished the difference between the existence and nonexistence of mails in the mailbox. Then, the microcontroller translated the analog output from the sensors to corresponding digital output. Subsequently, wireless communication module sent a numeric value (0: No mail, 1: On the Porch, 2: At the Post Office, 3: You got Mail!) to the receiving unit.
The receiving unit successfully received the wireless signal from the mailbox unit. Afterwards, the LCD displayed the matching message with received time. As signals were received from the transmitting unit, the microcontroller was connected to Wi-Fi network and sent the request to the web server. Upon receiving the request, webserver echoed the received message to the user's email address. Thus, sent the same message as the message displayed on the LCD with the arrival time.

5.1 Future Work

Although our project was successful, there are improvements and modifications that can be made to improve our product. On the mailbox unit, readjusting the placement of sensors according to the angle of IR emission would increase the sensitivity and output definite voltage changes. In addition, implementing MOSFET would also support amplifying the insignificant voltage drop for the mail detection. Moreover, synchronizing reset buttons on the mailbox unit and receiving unit to reset the past messages will be convenient for the user. On the receiving unit, employing directional buttons would allow the user to revisit the past messages. Furthermore, for security purpose, encryptions on email messages should be required to protect the users from possible cyber crimes such as identity theft. Lastly, for both mailbox and receiving unit, instead of using expensive finished product, customizing the circuit with only essential parts would reduce the overall production cost.

5.2 Ethics

We adhere to the statements of the IEEE Code of Ethics that pertain to our project as follows:

- "3. To be honest and realistic in stating claims or estimates based on available data"

  We will ensure that all calculations are accurate. All the conclusions drawn from experimental procedures will be supported by data calculations and simulation graphs.

- "6. To maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations;"

  We will apply our analytical and technical skills learned thus far to the best of our abilities in creating a product that will promote household convenience.

- "7. To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others."

Because this is a group project, each person will communicate their ideas on improving the design and provide feedback on each other's work in terms of quality and things that need improvement. In addition, all information used from outside sources will be credited in the Reference Materials section.

6. Reference

[1] "Arduino - ArduinoWiFiShield." *Arduino - ArduinoWiFiShield*. Arduino, n.d. Web. 18 Feb. 2013. <http://arduino.cc/en/Guide/ArduinoWiFiShield>.

[2] "Arduino - LiquidCrystal." *Arduino - LiquidCrystal*. N.p., n.d. Web. 23 Feb. 2013. <http://arduino.cc/en/Tutorial/LiquidCrystal>.

[3]"Date and Time on Arduino LCD Shield." *Hacks To Try*. N.p., 17 Oct. 2010. Web. 24 Feb. 2013. <http://hacks2try.blogspot.com/2010/10/date-and-time-on-arduino-lcd-shield.html>.

[4] "Home | Product Categories | 5mm | COM-09349." *LED-Infared 950nm*. Sparkfun Electronics, n.d. Web. 20 Feb. 2013. <https://www.sparkfun.com/products/9349>.

[5] "IEEE Code of Ethics." *IEEE*. N.p., n.d. Web. 10 Feb. 2013. <http://www.ieee.org/about/corporate/governance/p7-8.html>.

[6] "Index - Arduino Forum." *Index - Arduino Forum*. N.p., n.d. Web. 23 Feb. 2013. <http://arduino.cc/forum/index.php/topic,48326.0.html>.

[7] "IR Receiver." *Ir Receiver,View Ir Receiver,LFN Product Details from on Alibaba.com*. Shenzhen Lanfeng Technology Co., Ltd., n.d. Web. 21 Feb. 2013. <http://lfn.en.alibaba.com/product/374347181-210768193/ir_receiver.html>.

[8] Kamal, Ibrahim. "Infra-Red Proximity Sensor Part 1." *Infra-Red Proximity Sensor Part 1*. N.p., 22 Jan. 2008. Web. 15 Feb. 2013. <http://www.ikalogic.com/infra-red-proximity-sensor-part-1/>.

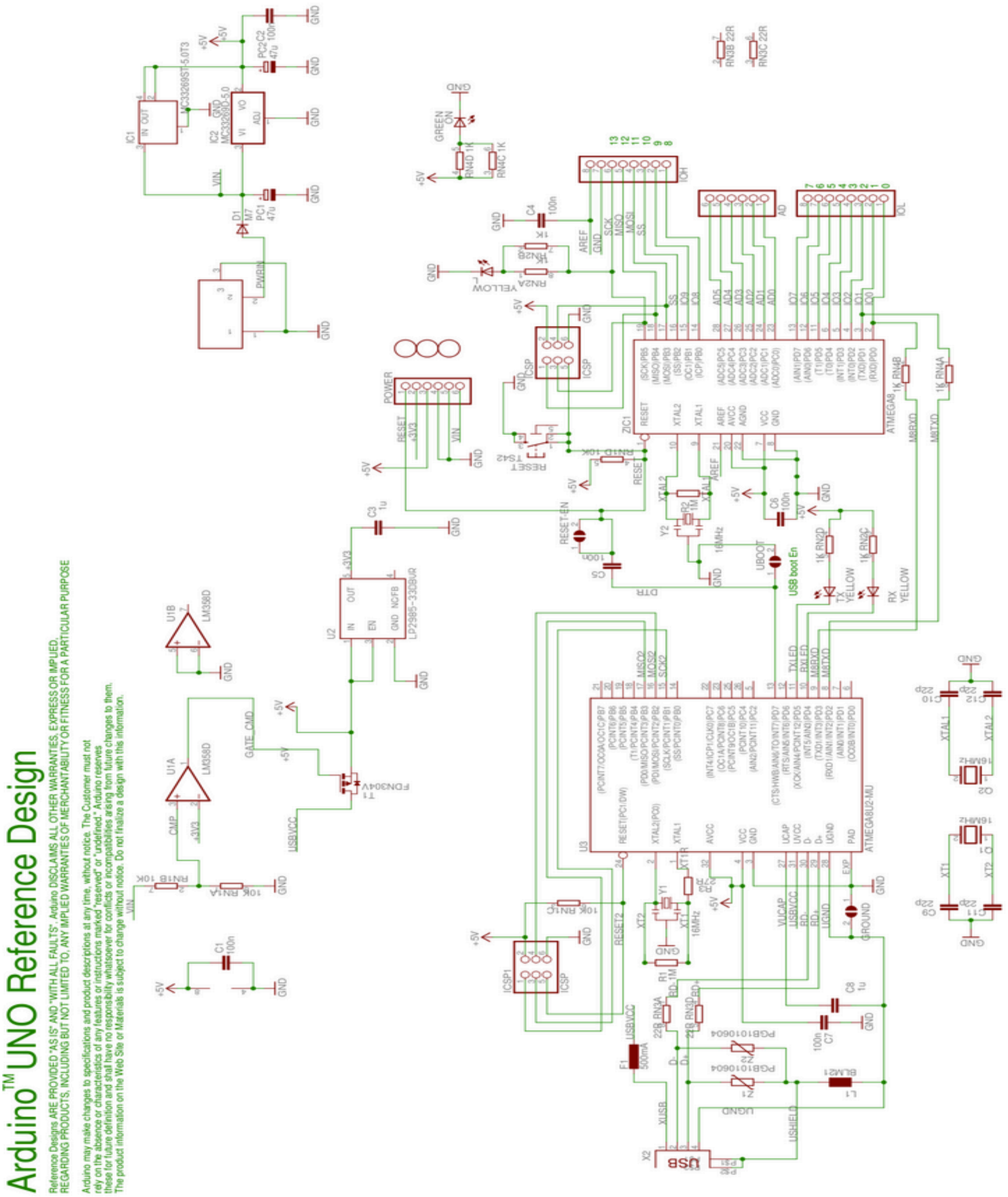
[9] "Kavasmlikon." *Kavasmlikon*. N.p., 21 Nov. 2011. Web. 27 Feb. 2013. <http://kavasmlikon.wordpress.com/2011/11/21/how-to-wirelessly-transmit-analog-signals-using-the-arduino-xbee-api/>.

[10] Pinal, Dave. "SQL Server Journey with SQL Authority." *SQL Server Journey with SQL Authority*. N.p., 23 Aug. 2008. Web. 19 Feb. 2013. <http://blog.sqlauthority.com/2008/08/23/sql-server-2008-configure-database-mail-send-email-from-sql-database/>.

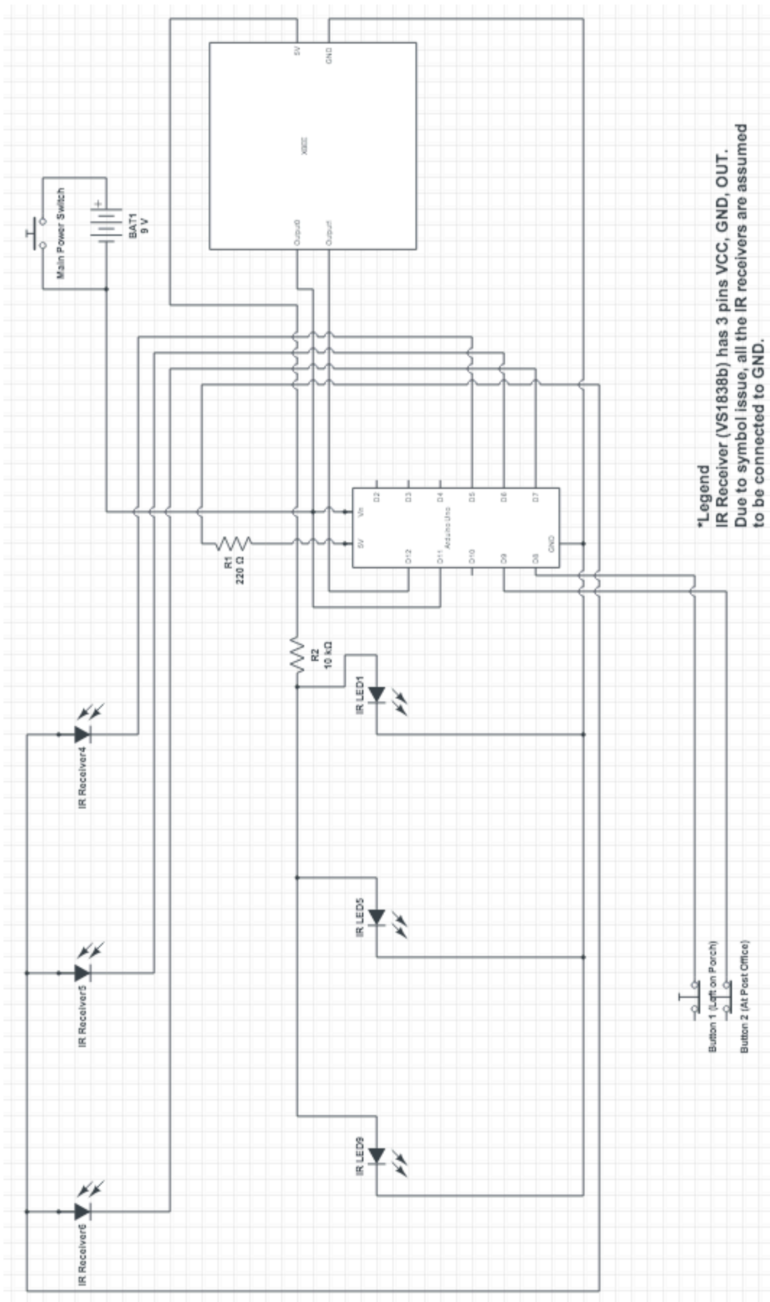
[11] "The 555 Timer." *The 555 Timer*. N.p., n.d. Web. 29 Apr. 2013.

7. Appendix

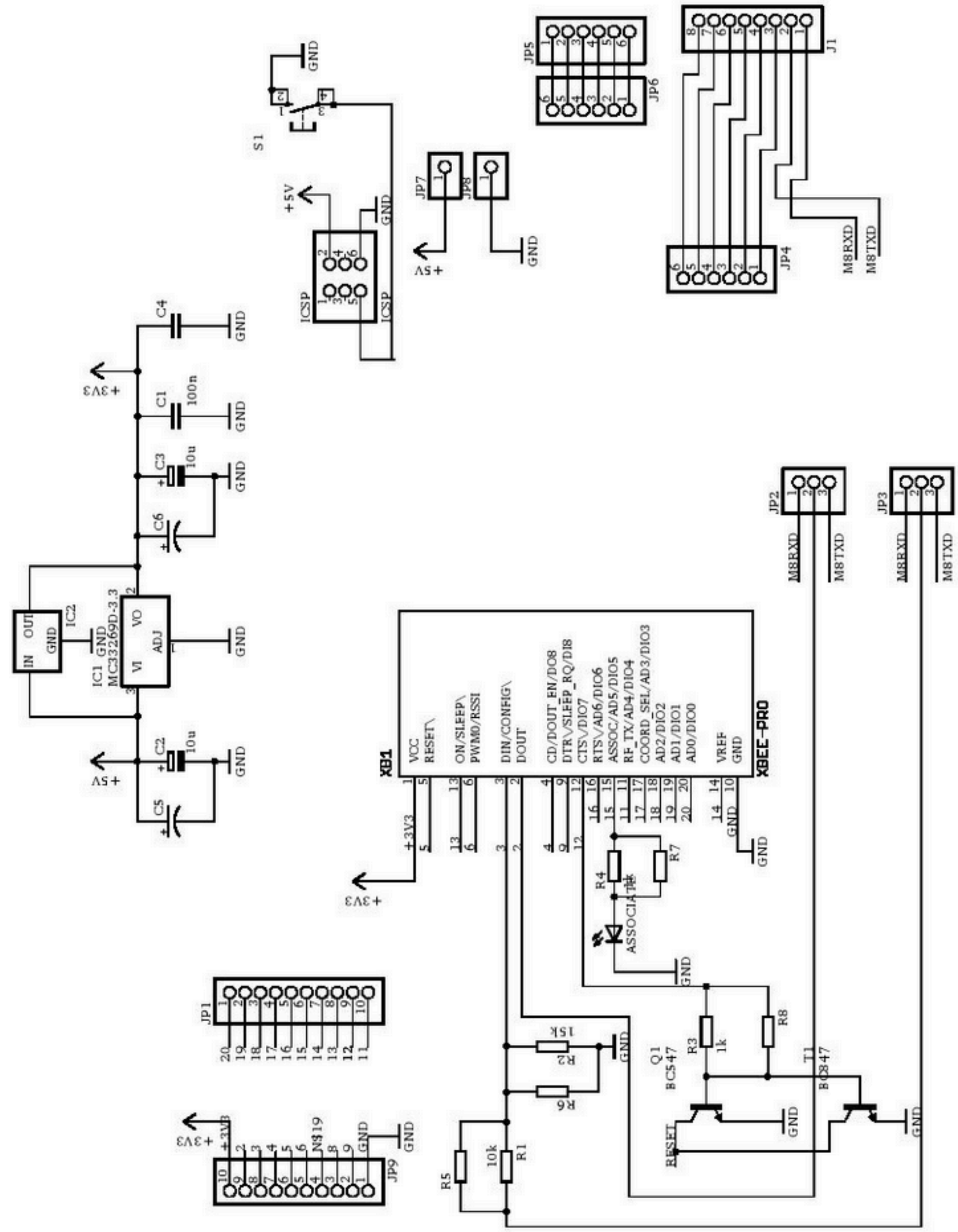
Schemetics of Arduino Uno



Schematics of Mailbox Unit

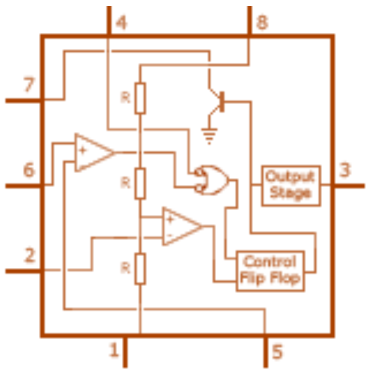


Arduino Wi-Fi Shield



Explore inside the 555 Timer

Since the 555 timer played a significant role in our design, we spent little more time on getting to know the device itself. What makes this 555 timer so powerful, steady, and user-friendly? It was first introduced in 1971 by the Signetics Corporation and was known as "The IC Time Machine".



Operating modes are monostable and astable. In monostable mode, whenever it is triggered by an input pulse, it switches to its temporary state and remains in that state for a period of time determined by RC circuit. The junction of the resistor and capacitor is connected to the threshold input to the upper comparator. The internal discharge transistor is also connected to RC circuit and the trigger input is fed into the lower comparator.

In astable mode, both the trigger and threshold inputs to the two comparators are connected with the external capacitor. The capacitor charges through the two resistors. Also the discharge pin is connected to the internal transistor to the junction of the two resistors. When power is applied to the circuit, the capacitor will be set to zero. The lower comparator turns the flip-flop to high which causes the transistor to be off. The capacitor will then immediately charge through the two resistors. As soon as the capacitor reaches about 2/3 of the supply voltage, the upper comparator will trigger flip-flop to reset, which will cause the output to become low. When the capacitor is discharging and reaches 1/3 of supply voltage, the lower capacitor will be triggered causing the flip-flop to set high. The cycle will continue repeatedly until there is no more power supplied.