SMART LAPTOP CHARGER / BATTERY MANAGER

AN IDEA LAB PROJECT

SUBMITTED BY:

AMIT PANIGRAHI (13BEE009)

ARKA SAIN (13BEE011)

PALAK GOYAL (13BEE063)

MENTORED BY:

PROF. P.N. KAPIL

ELECTRICAL ENGINEERING DEPARTMENT

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

APRIL 2017

DECLARATION

We do hereby declare that the technical project report submitted is original, and is the outcome of the independent investigations/research carried out by us and contains no plagiarism. The research is leading to the discovery of new facts/techniques/correlation of scientific facts already known. This work has not been submitted to or supported by any other University or funding agency.

We do hereby further declare that the text, diagrams or any other material taken from other sources have been acknowledged, referred and cited to the best of our knowledge and understanding.

07 April 2017

Place: Ahmedabad, Gujarat

Amit Panigrahi	Arka Sain	Palak Goyal	Prof. P. N. Kapil
(13BEE009)	(13BEE011)	(13BEE063)	Asst. Prof. - EE

NIRMA UNIVERSITY INSTITUTE OF TECHNOLOGY
IDEA LAB

ELECTRICAL ENGINEERING DEPARTMENT

Final Report of the work done on the Idea Lab Project:

1. Project Title: Smart Laptop Charger / Battery Manager.
2. Period of Project: 1 Year
3. (a) Name of Students – Amit Panigrahi (13BEE009)

   Department – Electrical Engineering

   (b) Name of Students – Arka Sain (13BEE011)

   Department – Electrical Engineering

   (c) Name of Students – Palak Goyal (13BEE063)

   Department – Electrical Engineering

   (d) Name of Mentor – Prof. P. N. Kapil

   Department – Electrical Engineering
4. Project Start Date: March 2016.
5. (a) Total Amount Approved – 3000/- INR.

   (b) Total Expenditure – 743/- INR.

Brief objective of the project: This project aims at designing a smart laptop battery charger, which automates the charging process without any kind of human intervention, thus connecting and disconnecting the charger when the battery falls below a fixed value and when the battery charge reaches the full value or the maximum set value respectively.

Signature
Amit Panigrahi
13BEE009

Signature
Arka Sain
13BEE011

Signature
Palak Goyal
13BEE063

Signature
Prof. P. N. Kapil
Asst. Professor
Electrical Engg. Department
Institute of Technology,
Nirma University,
Ahmedabad

Signature
Prof. P. N. Kapil
Idea Lab Co-ordinator,
Electrical Engg. Department
Institute of Technology,
Nirma University,
Ahmedabad

Signature
Dr. Ankit Thakkar
Idea Lab Co-ordinator,
Institute of Technology,
Nirma University,
Ahmedabad

Signature
Dr. P. N. Tekwani
Head of Department,
Electrical Engg. Department,
Institute of Technology,
Nirma University,
Ahmedabad

Signature
Dr. Alka Mahajan
Director,
Institute of Technology,
Nirma University,
Ahmedabad

CONTENTS:

1.1 Introduction

This project aims to create a simple laptop battery charging management system which is aimed towards aiding those people who generally do not shutdown their systems and keep using them for a longer duration of time. Such long usage of laptop leads to frequent charging and discharging of the laptop battery. This leads to a situation where the user has to permanently keep their chargers connected to laptop to avoid the discharging of the laptop.

Due to the above mentioned case, the laptop battery starts holding less and less charge in it as the proper charging and discharging cycle is not followed. This leads to a condition where the charge holding capacity of the battery is reduced considerably and the battery becomes prone to malfunctions.

Hence, our hardware and software aims to create an automatic charge monitoring system which connects and disconnects upon the state of battery discharge and charge respectively.

We have managed to furnish the hardware and have successfully written the code to make this possible by using an Integrated Development Environment using java as the base programming language and were thus, able to create a working model of the above mentioned system.

1.2 Literature Survey

The majority of laptop batteries used nowadays are Lithium-Ion batteries. While there are many advantages of Lithium Ion Cells such as having high energy density, low maintenance and no requirement for priming, Lithium Ion batteries have their fair share of disadvantages too. Some of the very few disadvantages are discussed below:

1. **Li-Ion batteries need protection:** Lithium-Ion batteries are generally, not as robust compared to other type of cells such as Nickel-Cadmium and Lithium Cobalt Oxide cells. They require constant protection from being over charged and discharged. In addition, they need to have the current maintained within safe limits. Thus, they require a well-fabricated protection circuitry incorporated to ensure that they are kept within their safe operating limits.

2. **Ageing:** Lithium Ion batteries suffer from ageing and age related problems. The ageing process of the battery is not only time or calendar dependent, but it is also heavily influenced by the number of charge discharge cycles that the battery has undergone. A typical Lithium-Ion battery or cell needs to be stored in a partial charged condition ranging from around 40% to 50% and kept in a cool storage area. This increases the life and performance of the battery drastically and in applications where the scale of use is high as in huge server rooms or corporate offices, this small thing can help reduce costs to a greater effect.
3. **Cost:** A typical Lithium-Ion battery is almost 40% more costly than a conventional Nickel-Cadmium battery. This has to be a major factor considering their use in mass produced consumer items such as Laptops and servers and where additional costs can become a major issue. [1]

This project is based on the premise that we will be able to control the charging and discharging process of the cell or battery at our own will and thus solve the afore-mentioned problems discussed above. Proper charge management will not only provide protection and safety against overcharging but will also increase the longevity of the cell and will in turn help reduce costs drastically.

1.3 Objectives Proposed

- To create a charger add-on module which is capable of connecting and disconnecting the charger automatically without any human intervention.
- To create a setup which increases the longevity of the Lithium Ion through proper charge management of the Battery.
- To increase the usage efficiency of the Lithium Ion battery and help reduce costs on a larger scale.
- To make the above said product be able to run at system start-up automatically.
- To make the above said product to be able to execute the program by itself and by not using help of the external IDE.

1.4 Components Used

Arduino Uno Board: Arduino Uno is a Development board manufactured by Arduino Inc. It consists of an Atmel Atmega 328p - 8-bit micro controller on board which is used to program the different test cases of our project. It also has an on board 16 MHz Crystal along with other discrete electrical components and a USB Interface. The Uno board can be programmed using the Arduino Integrated Development Environment.

Relays: Relays are used as a switch to make or break the charging circuit based on the signals from the Arduino. The afore-mentioned relays are rated at 6 Volts DC.

Discrete Electrical Components: Many discrete electrical components have been used during the course of designing the Hardware such as Green Terminals, Budstrips and laptop charger pin connectors.

1.5 Softwares Used

Codeblocks IDE: This IDE was used to write the program for battery state detection in C++. The program fetches the real time battery charge level and stores it into a text file.

Eclipse IDE: This IDE was used to write the programs for battery state and charge control. The programs were written in JAVA Standard Edition 8, making use of the installed Java Development Kit 8 on our systems.

Arduino IDE: This was used to write the code for the interfacing the Arduino with the system files programmed in JAVA and C++.

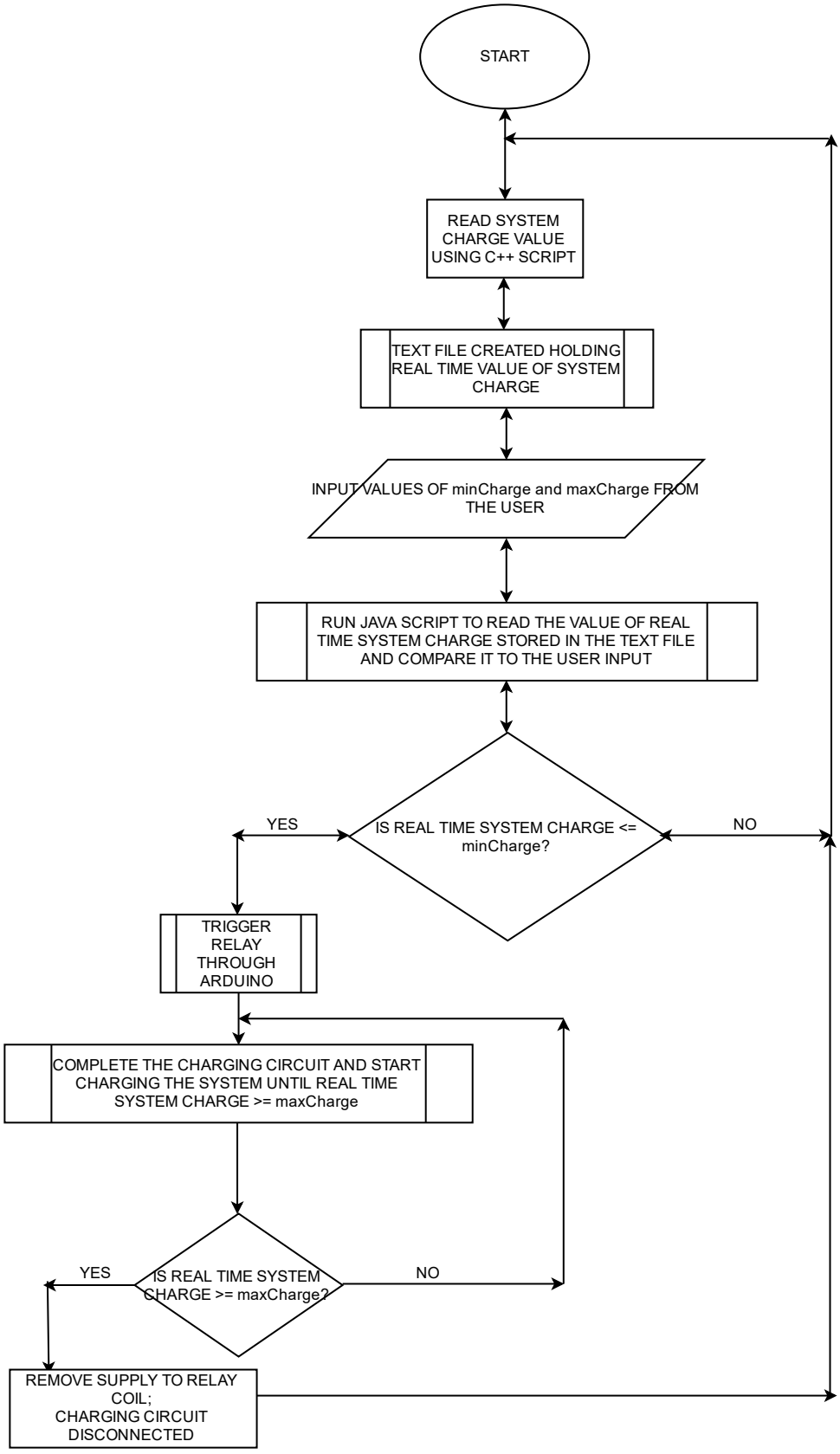
**** The Programs are attached in the Appendix section in the proper order.**

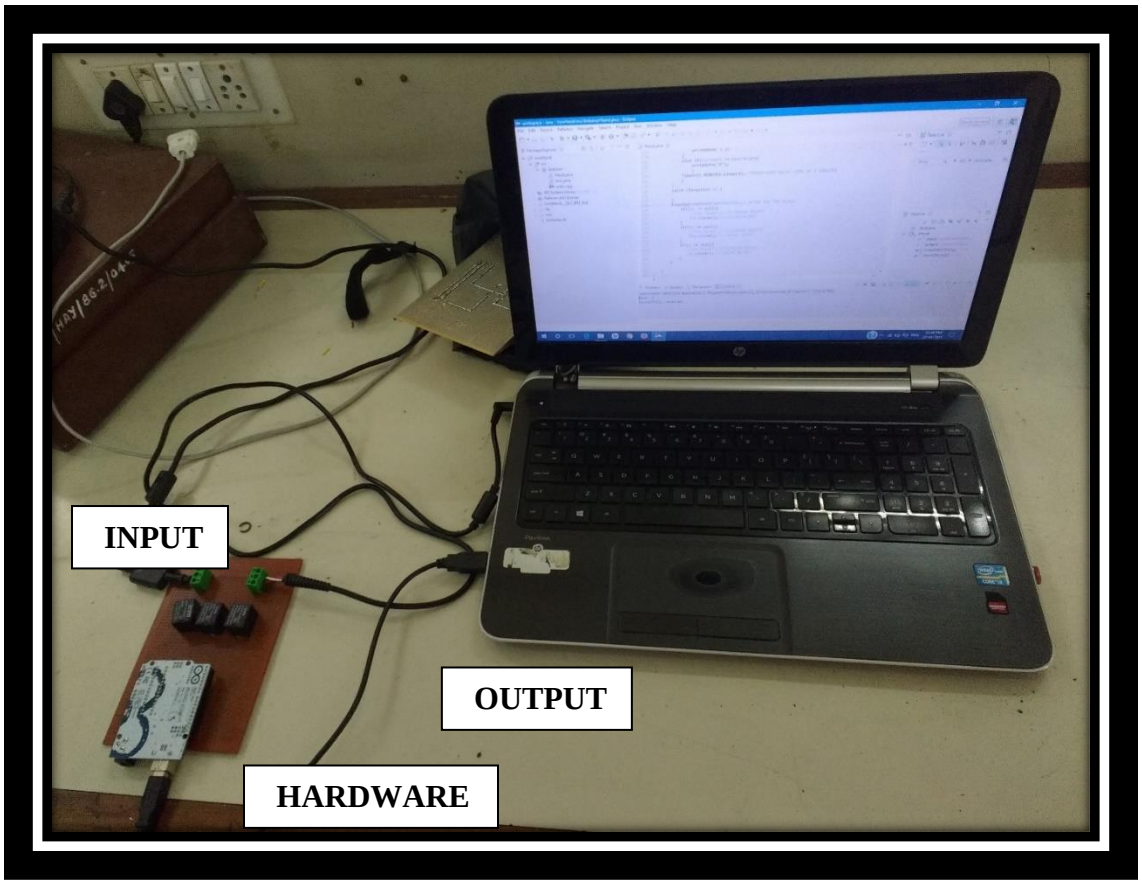
1.6 Evaluation Methodology

The whole project is based on the premise that we will be able to control the charging and discharging of the Lithium Ion cell of the laptop. To achieve this objective, we developed a new hardware circuit which uses a microcontroller to turn on or off a relay so as to make or break the charging circuit. The Laptop adapter / charger is connected on to the input side while another power cord is connected at the output side through the green terminals which is coupled with the laptop. The relay acts as a bridge between the input and output side and makes or breaks the charging circuit depending on the signals from the Arduino or the controller.

The software part comprises of programs written in C++ and Java. The first part comprises of the program written in C++ which uses an in built library provided by Microsoft to obtain the real time charge status of the laptop battery and we use that to store that particular information in a text file named "Stats.txt". The next part comprises of two java programs, which are used to obtain the values of minimum charge level and maximum charge level that the user wants to set to control the charging and discharging automatically. We store the corresponding values in a text file named “Settings.txt”. The user has to then upload the “IronHand_-_SLC_BM_Ard.ino” file into his/her Arduino development board following which the user has to open and execute the two java programs with the help of Eclipse IDE. The “Hand.java” program then checks the battery status every three minutes and depending on the values of minimum charge and maximum charge set by the user executes the program to control the charging of his/her Lithium Ion battery. Thus, the user with the help of the hardware setup and the software resources provided to him/her, the user is able to control the charging of his laptop battery in a safe and prescribed way or in a way he desires, thus enabling proper charging and discharging process, thereby increasing the efficiency, performance and longevity of his/her battery. The flowchart for the complete procedure is attached in the next page.

P.S. A series of Video Tutorials are provided with the hardware setup that neatly explains the installation and execution of the concerned files.


START
READ SYSTEM CHARGE VALUE USING C++ SCRIPT
TEXT FILE CREATED HOLDING REAL TIME VALUE OF SYSTEM CHARGE
INPUT VALUES OF minCharge and maxCharge FROM THE USER
RUN JAVA SCRIPT TO READ THE VALUE OF REAL TIME SYSTEM CHARGE STORED IN THE TEXT FILE AND COMPARE IT TO THE USER INPUT
IS REAL TIME SYSTEM CHARGE <= minCharge?
YES
NO
TRIGGER RELAY THROUGH ARDUINO
COMPLETE THE CHARGING CIRCUIT AND START CHARGING THE SYSTEM UNTIL REAL TIME SYSTEM CHARGE >= maxCharge
IS REAL TIME SYSTEM CHARGE >= maxCharge?
YES
NO
REMOVE SUPPLY TO RELAY COIL;
CHARGING CIRCUIT DISCONNECTED



Final Experimental Setup

The above model works as follows:

- The java program running in eclipse IDE will time to time check the current battery charge percentage value and store it into an external text file.
- The percentage value saved in the text file is retrieved by the java program and stored into a variable.
- Then the current value is compared with the minimum and maximum values of the activation set by the user.

- If the current battery value is less than the set minimum value or if the current battery value is more than the set maximum value, then a command is sent to Arduino by the java program.
- Depending upon the command sent to the Arduino, the switching action takes place with the help of relays and the supply to battery is either turned on or turned off.

1.7 Objectives Achieved

First objective was achieved in which we managed to create the hardware which consists of a relay setup and this setup is used to connect the input and output of the laptop charger's Male terminal and laptop's female terminal. The Arduino is used to provide the control signal to the relay. Arduino is interfaced with the laptop with the Eclipse IDE, which is also used for JAVA programming.

The IDE here is used to read the system's current Battery percentage value, read the minimum and maximum value for which the charger add-on will activate and deactivate as per the customer's needs. This battery value is saved into a text file using a C++ program which reads the battery statistics and from there it pulls out the current battery value and then saves it into a text file. Then the battery percentage value from the text file is collected by IDE and then this reading is compared with the minimum and the maximum value of the battery which the user has given as input. Then according to the situation, the data is sent to Arduino for activating or deactivating the relay.

1.8 Technical Difficulties Faced

- The program is working on the Eclipse IDE successfully but when we try to execute the program via command line, an error appears which we aren't able to resolve hence the program is not able to load itself at the start-up and has to be activated manually all the time through the IDE.
- The program is not able to run itself without the command line hence the whole aim to make a standalone software is not achieved due the above said error in java program compilation.

1.9 Budget Analysis

- Total Budget Approved → 3000/- INR.
- Total Expenditure → 743/- INR.

1.10 Conclusion and Future Scope

We would like to conclude this project by stating that the objectives achieved by us led to the advent of a new feature rather than a new product. This is a feature, which if applied in suitable necessary areas can perform its function in a most efficient and cost effective manner. We were able to, at the end of the project; control the charging and discharging process of the Lithium Ion Battery of our laptops using just few scripts of code and an Arduino. This feature thus enables us to efficiently manage and mitigate the problems associated with the longevity and performance of the Lithium Ion battery.

The future scope of this project is quite large as the "feature" that we have developed can be incorporated within the laptops manufactured in the future. The hardware, if developed and manufactured using Surface Mounted Devices can actually prove to be very cost effective as well as being very size effective. In fact, the hardware can actually be fabricated on a printed circuit board of dimensions – 2*2 Inches. Further, the Software component can be incorporated as some sort of driver software that could be made available to the user to install as part of the whole package.

REFERENCES

[1] - http://www.radio-electronics.com/info/power-management/battery-technology/lithium-ion-battery-advantages-disadvantages.php

[2] - http://playground.arduino.cc/Interfacing/Java