CS-308-2016 Final Report

Switch Easy

Team Members:

1. Ch Nitish Chandra (120050071)
2. Siddhant Rajagopalan (120100006)
3. D Saicharan (120050060)
4. TP Varun (120050077)

Table of Contents

1. Introduction

Now a days there is a surge in usage and production of smart appliances like fans, lights etc., Main advantage of these devices is that they can be controlled using a remote or another smart device like a phone.

But the problem with these devices is that they are expensive and hence out of the reach of common people. Also if one wants to use a smart fan that can be controlled by a remote, one has to discard the old – may be perfectly working – fan.

We propose a solution to address these drawbacks while maintaining the fact that devices can be controlled by a remote. We call it Switch Easy. One thing that is common in many homes is that there are switch boards and most of the devices can be controlled by these switch boards. So, instead of controlling the devices themselves using a remote, we control the switch boards and in a plug and play manner without changing their internal circuitry. This solves the problem of controlling the devices by a remote and is cheap and works out-of-the-box with the existing devices – thus addressing both the drawbacks of traditional smart device based approach.

Switch Easy can be used in all the homes with switch boards.

2. Problem Statement

We need a mechanism by which to control the switch board using a remote. The mechanism must be such that it must be fixed on a switch board without twiddling with the board's circuitry.

3. Requirements

3.1 Functional Requirements

1. We need a system that controls the switch board

2. It must communicate wirelessly with a remote.

3. The system must accept commands from remote and must toggle the appropriate switch.

4. To account for the cases where remote is damaged and cannot be used to control the system, manual toggling of switches must also be made possible even with the system being present on outside of the board.

3.2 Non-Functional Requirements

1. Because the system is present on the outside of the board, toggling a switch must be done physically.
2. To keep the cost low, communication with the remote must be done using minimal overhead
3. To provide for proper functioning of the system, appropriate mechanism for charging the microcontroller must be provided

3.3 Harwdare Requirements

1. TIVA board as a microcontroller
2. Bluetooth module
3. A soap case for holding all the hardware components
4. A phone (as a remote)
5. Micro servos for physical toggling of switch

3.4 Software Requirements

1. An app to communicate with the micro controller
2. Code for the microcontroller to communicate with bluetooth module and move the servos appropriately

4. System Design

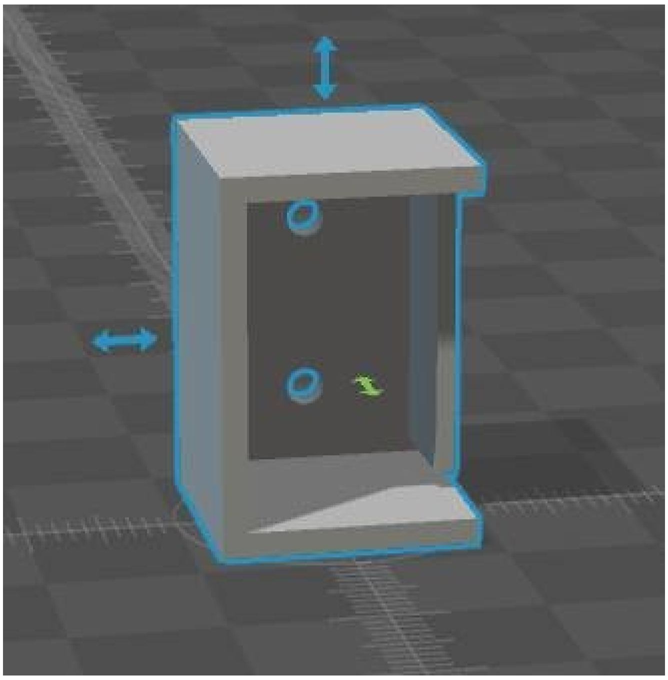
App:

Following is the design of the app.

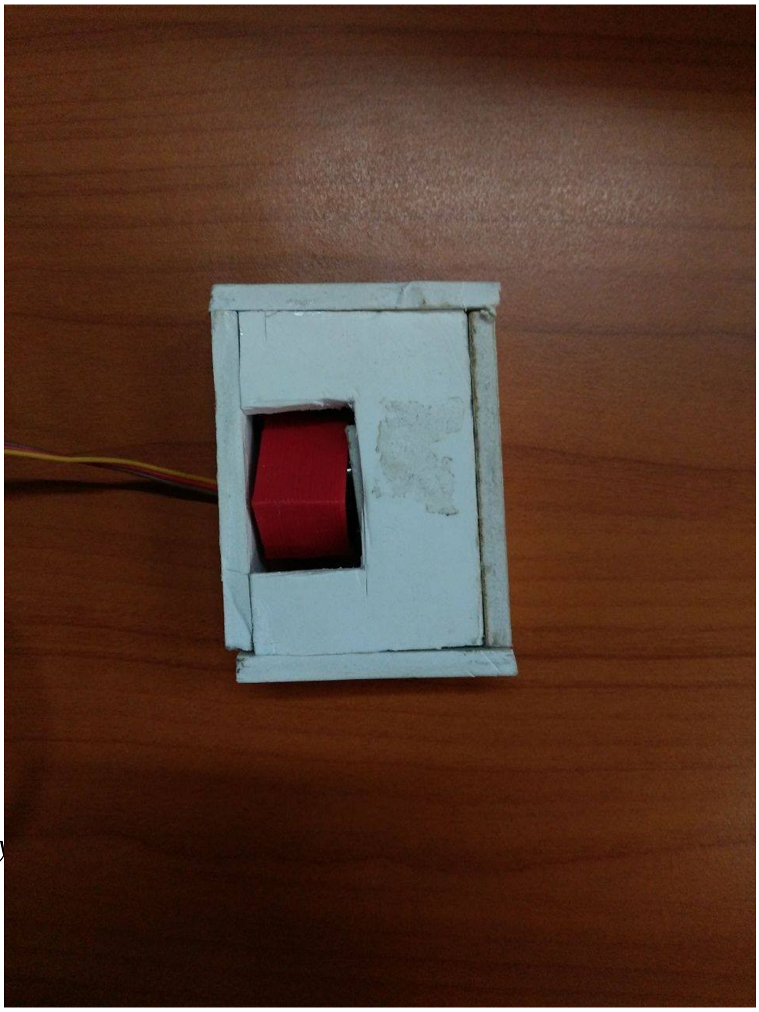
For the purpose of demo, we have implemented the system on a switch board with 2 switches. The app gives a userfriendly way to control these switches.

Case to attach onto a switch (switch case):

Following is the proposed design of the case. We planned to 3d print the case and mount it on the switch.



But the result of the printing was too thick and because of time crunch, we constructed the case using the materials available in workshop:



The red switch present in the above image serves the purpose of providing manual control when the controller – for some reason – cannot be controlled using remote. We have printed the red switch using 3d printing.

The case has the mechanism to physically toggle the switch that is present beneath the switch case.

Micro controller case:

Following is the design of the case which is used per switch board. This holds the micro controller and bluetooth module.

How everything fits:

1. We have used bluetooth for communication between phone and micro controller.

2. Each switch board in house has a single micro controller case that houses the micro controller and bluetooth module

3. On each switch of a switch board, a switch case is mounted. All the switch cases talk to the controller case of the switch board

4. App in the phone talks to the controllers present in home

5. Manual switching is provided for on the switch case

6. Temporarily we have used a power bank as a power source

5. Working of the System and Test results

The app sends a different signal for each of the 4 buttons. The bluetooth module present in the controller case reads the data sent by the phone using UART and based on the signal, performs a specific action.

For example, if the signal was to turn the switch 1 on, then the TIVA board sends instructions to the corresponding switch case to rotate appropriately.

Following is the side view of the switch case:

On receipt of the signal to turn on the switch, the rotor turns in 'on-direction' making the red color block and the shaft move in the same direction. When the shaft hits the switch, it turns the switch on.

Because we have provided a window on the switch case, it can be used for manual control of the switches.

We attached the above switch case to a switch using double sided tape.

Testing the system:

We mounted the switch cases and controller case on a switch board with two switches and successfully controlled the switches using phone. This verifies that the communication between phone and the microcontroller and also between microcontroller and switch case is working. It also verifies that the rotor is moving appropriately to toggle the switch.

6. Discussion of System

a) What all components of your project worked as per plan?

The TIVA board, motors and the android app worked as per planned initially. Setting the bluetooth module baud rate and configuring it took more time than expected.
The 3D printed models of the switch and its holder didnt work as per our plan. The thickness of the 3D models were more than what we expected. So, the motor did not fit into the holder.

b) What we added more than discussed in SRS?
We tried to use 3D printed models for the switch and it's case. We used solidworks to make the design.

c) Changes made in plan from SRS:
As the 3D models didnt work, we had to make the switch and holder using cardboard. We made the box(containing TIVA and bluetooth module) using acrylic. We couldn't print new 3D models as they

take lot of time to get printed.

7. Future Work

Possible Extensions:

1. Our current model works for a limited range of switch lengths. The switch holder can be designed in such a way so that multiple switch lengths and sizes could be taken care of.
2. The motor remains turned on all the time which consumes more power. A possible extension would be to turn on the motor only when the bluetooth module recieves a signal. This also ensures that all motors are not used all the time.
3. A single TIVA board could be used to control the whole house rather than onr board per each switch board.
4. Rather than using the power outlets of the switch boards, we can use more power efficient and user-friendly mechanism for power input.

Re-usable components:

1. The switch holders are of plug-and-play model. So they can be removed from one switch and put on another. Also the holders can be plugged into each other making them space efficient.
2. A single TIVA board and bluetooth module can be used to control multiple switches.

8. Conclusions

We have realised that designing and implementing an embedded application is much more than just coding and arranging parts. It involves many engineering issues. We have tried to solve a problem that many people face and had to make many concessions due to space restrictions.

9. References

1. Getting Started with the Tiva™ TM4C123G LaunchPad Workshop.
https://www.cse.iitb.ac.in/~erts/html_pages/Resources/Tiva/TM4C123G_LaunchPad_Workshop_Workbook.pdf

2. Z-EK- TM4C123GXL Rev A Schematic
https://www.cse.iitb.ac.in/~erts/html_pages/Resources/Tiva/Z-EK-TM4C123GXL%20Rev%20A%20Schematic.pdf

3. Android App development reference
http://developer.android.com/reference/packages.html

4.Serial Port Bluetooth Module : HC-05
http://wiki.iteadstudio.com/Serial_Port_Bluetooth_Module_%28Master/Slave%29_:_HC-05

5. Base code for Bluetooth communication
https://github.com/janosgyerik/bluetoothviewer