TRANSPARENT KRYBOARD

Venkata Pavan Rama Sai Bharadwaj. Gorthy
Department of Electrical and Computer Engineering
Colorado State University

Yasith. Mir
Department of Electrical and Computer Engineering
Colorado State University

Abstract- **An attempt to manufacture a prototype keyboard with transparency as a goal and to make it look aesthetic enough, at a cheaper price than the existing Bastron glass keyboard [1] and close to that of the laser projection keyboard.**

Keywords- ***InfraRed Light Emitting Diode (IR LED), InfraRed Receiver Sensor (IR Sensor), Teensy 3.6*, Universal Serial Bus (USB).**

I. FOUNDATION AND BACKGROUND

When it comes to I/O devices, particularly the keyboard in a workstation set-up, there has not been much of a design change since the genesis of it, when considered in a commercial, mass consumer domain. Although, there have been different mechanical changes in the design say in terms of arrangement of the keys or the distance they travel when a key is pressed, never was any product released in the market for commercial use that was aesthetic enough and changed the look of the keyboard and the way it is perceived up until when Laser projection keyboard was introduced at a cheaper price by a company called Canesta that manufactures CMOS-based (Complementary Metal oxide Semiconductor) single chip 3-D sensors in 2002 but was not in use up until 2010 as technology scaling and the manufacturing costs of Integrated Circuits (ICs) came down. But even then, to invest $60-100 on just an input device for commercial use was a thing to be given a thought but as aesthetics of appliances became a trend and further reduction in manufacturing costs of ICs and Printed Circuit Boards (PCBs) was witnessed, it no longer became a thing to think of, to buy one. But, Projection keyboards are a bit costly when compared to a normal mechanical USB keyboard which only costs around $11. Another variety of a keyboard that offers transparency is the glass keyboard offered by two companies primarily are Brookstone [2] and Bastron both offering the keyboards at around $100. But none of these have the flexibility to use them everywhere. The projection keyboard needs a plain surface to project and detect the key press and as for the glass keyboards, they are quite costly for a commercial application and cannot be mounted onto surfaces with that ease. So, we propose a design of a keyboard which can be mounted onto screens of regular laptop screens or monitors of desktop Personal Computers (PCs) and can be manufactured at a cheaper rate than the glass keyboard and close to that of the projection keyboard.

II. INTRODUCTION

The proposed transparent keyboard works on the basic principle of IR beam breaking circuit. Whenever IR beam is interrupted, a signal has to be recorded/sent to a microcontroller and based on the sensor inputs, a particular key is transmitted from the output USB port of the controller. Arrangement of the IR emitter LEDs and the IR receiver sensor are arranged on the outer frame which is in the shape of a rectangle of which two of the adjacent sides occupy the IR transmitters and the opposite adjacent sides house the IR receiving sensors. Figure 1 shows the proposed arrangement of the keyboard and Figure 2 shows the arrangement of IR LEDs and IR receiving sensors.

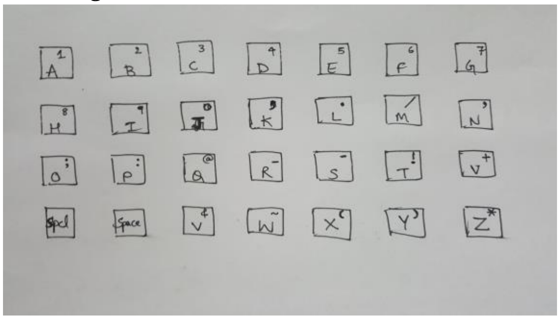

Figure_1: Arrangement of the proposed keyboard

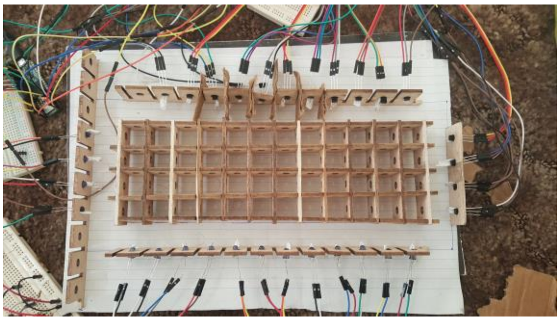

Figure_2: Arrangement of IR LEDs and Sensors

Note that the IR LEDs are arranged on the left and bottom side of the rectangle and the sensors on the right side and top side of the rectangle (They can be arranged in any of the arrangement and not limited but the only and obvious restriction is that the sensor and emitter set should always be opposite to each other for obvious reasons). The entire emitting, sensing, sending a signal, registering a key and sending it through a USB works on two simple principles. One, the emitting light from the IR LED is of 38KHz pulse signal which is detected by the IR receiving sensor whose data pin is connected to one of the digital pins of the controller which takes in a logic ‘0’ when light is detected by the sensor and a logic ‘1’ when the sensor does not detect light. This condition is checked and accordingly alphabets are sent through the USB and it works like a normal computer keyboard.

III. PROTOTYPE COMPONENTS

We call it the prototype because of the test state that the keyboard is in and the components we used in our prototype are as follows:

(i) Teensy 3.6

Teensy as the name suggests is tiny and a very breadboard-friendly development board with lots of features in it. As any other controller, it has a bootloader and Teensy can be programmed using the on-board USB connection. Programming can be done on any editor using C or there is an add-on software to install on PCs called Teensyduino. It is called so because Teensy is a tweaked version of Arduino specifically for USB applications. Thus, Teensy can be used with some other periphery to make it work like a mouse as well. But, we will stick to the USB Keyboard functions of the board. Since, being application specific, it is very small and suits best for us because of easy enclosure that can be made into an outer frame for aesthetics of the system on an entirety. Not only it can be used for USB type applications but also can be extended to USB Musical instrument digital Interface (MIDI) and other Human Interface Device (HID) projects as well.

Technical Features and specifications of Teensy3.6:
*180MHz ARM Cortex-M4 with floating point unit
*1M Flash, 256K RAM, 4K EEPROM
*USB High speed (480 Mbit/sec) Port
*2 CAN Bus Ports
*32 General Purpose DMA Channels
*22 PWM Outputs
*4 I2C Ports
*11 Touch sensing inputs

Here is a figure of the controller Teensy 3.6 that is used in the project.

Figure_3: Teensy 3.6 [3]

(ii) IR LED Emitter

IR Led emitters are most common in low end and remote communication devices and serves best when coupled with the right pair of sensing devices. We say remote here, because of the obvious fact that many Television sets are operated “remotely” with the help of devices called ironically enough, The Remote and is a cheap and best solution to control functions on devices at a distance. More so that these IR LEDs can be switched ON and OFF at user’s discretion and can even be modeled to send pulse waveforms which are the signals that are essentially used in a Television remote controller. These pulses are decoded and action is taking accordingly based on the decoded input from the sensor in the Television set. Because IR is invisible to the naked eye but can be seen through any camera (even the one on a mobile device), IR LEDs are used in detected heat signatures most of the times. Also, IR LEDs are used in Night vision sensing and is a demanding area now-a-days because of automatic driving and driver assist technologies. Thus, its applications in the field of communication surpass most of the other fields. The one that we used in our project is a super-bright 5mm IR LEDs. The wavelength that the IR LEDs emit in our project is 940nm requiring 100mA when operating continuously and 1000mA when operating in pulse mode and work on 1.6V forward voltage. The wavelength of InfraRed ranges from 700nm to 1mm. The LEDs that we picked operate in the near IR region. Wavelength is important to know because in the project we are trying to block the radiation from other IR LEDs inline so as to isolate the sensors from not detecting the light coming from other LEDs other than the one in its Line of Sight (LOS). Wavelength comes into picture when we implement a glass/acrylic frame in the center. A sample picture of IR LED that we used is shown below.

Figure_4: 940nm IR LED [4]

(iii) *IR Receiving Sensor*

The Counter-part of emission is reception and is done in the project by TSOP38238 IR Receiver Sensor. Not only these sensors just detect the light but also can decode what kind of a light, in this case pulse is being received. This kind of reception is very useful when it comes to detecting only particular coded pulse emitting from one of the IR emitters which is perfect for our project. Although there are difficulties in encoding the IR LEDs to respond properly to the sensors which we tried to encode but will see why it did not work out for us in the project later but, it can be in a way done. But, TSOP32238 as the first two numbers in the product description say that the sensors are tuned to 38KHz, perfect for receiving encoded commands or just to detect plain 38KHz sampled signals (pulses) and give an output. This 38KHz is specific because of the decoding circuit that is tuned to decode the samples at that particular frequency. It works for supply voltage ranges 2.5V to 5.5V and can operate even under very low supply current. There is also a preamplifier that is assembled in the tiny frame enclosure which is why it can work on lower currents as well while it receives signals from emitters and is thus very sensitive, which sometimes is a problem because it detects even the slightest leakage from the surrounding IR emitters. Also, the packaging feature acts as a special function of IR filter. A figure below gives a visual appeal of the sensor we used in the project.

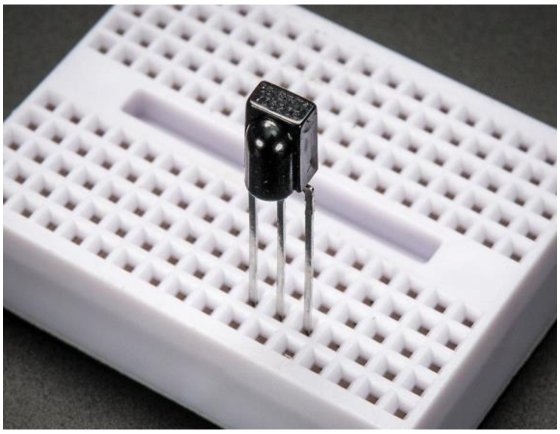

Figure_5: TSOP38238 IR receiving Sensor [5]

IV. PROTOTYPE-WORKING

As for the working of the prototype, it is quite simple. It got Simple because of the library that is already pre-defined in the Teensy loader as a header file. We would not want to go into the code but there are a few things to mention about how we interpreted the library and how we tweaked the code to suit our needs. Before we get into those aspects, let us get familiarized with how the set-up of the keyboard works.

As already described in the Introduction section of the paper, IR LED Emitters are placed to the left and bottom sides of the frame and the IR Receiving sensors on the opposite sides of the emitters. Now, the IR receiving sensors detect a signal and send a logic '1' from its data pin to the controller when sensor detects no light i.e., we interpret it as whenever a finger blocks the IR light, it gives a logic '1' from the data pin of the Sensor and vice versa. This so happens because of the fact that, the sensor data pin is active low. Now, In order to detect just the key press and to avoid long durational key presses, we have implemented a small tweak in the program in a way that we store what state the key press was previously and compare it with the current state of the key press which translates to saying that when the finger drops in and breaks the line, there is a change detected from the previous state and the logic is written in such a manner that only positive edge change in state throws out an output.

V. DIFFICULTIES IN IMPLEMENTATION

Before we went on with the implementation of the keyboard with IR LEDs and Sensors, we checked it with switches and everything worked fine but, when we have set all the IR LEDs and Sensor up, there was a problem that arose which was say when 1^{st} column's LED was glowing, the slits did not stop the light from passing through when a finger was placed and another key was displayed. When we analyzed the faults, we found that the connections made to the pins to the Teensy board were loose and had to keep on adjusting until we got the correct outputs. Another problem was with the distance between the Emitters and Sensors in a row was a little farther that the sensors were going crazy so from the initial thought of a 12x4 matrix of layout, we had to reduce it to 7x4 which solved two problems. One was the distance and detection problem and the other was that the emitters and sensors at the corners were doing their own work and the emitters at the corners were getting detected because of the design at the corner was such that the circular slits that were made were not holding the light but there were strange reflections happening and when we took the sensors and emitters off, of the corners and it did work at the end.

VI. CONCLUSION AND FUTURE WORK

This project had a great deal of interfacing with sensors to a microcontroller to deal with which posed as a Real-time Embedded system with real-time operational difficulties that we had to face with. That was something we did not expect and what the course Hardware/Software design of Embedded Systems always aimed at solving Real-time problems. Also, we initially wanted to design the inner frame with glass to give the design complete transparency but even with Balsa wood, there were improper detections with the sensors. Not that we could not do it but with the time-frame and budget constraints, it was not possible to have dealt with. The solution to make it completely transparent with Acrylic, a thin coating of Tin Oxide would absorb IR light emitting from the LEDs but pass through the circular slits made just like on the wooden frame. Thus, complete transparency can be achieved when this is implemented. The cost of the entire prototype was around $61.25 (controller: $15, LEDs: $4.8832, Sensors: $23.36, Frame material: $3, Frame Manufacturing: $15) which can be further pulled down if the product is manufactured in bulk. The transparent keyboards that are offered by Bastron and Brookstone are around $100 and $90 respectively from which we can say that the design that we have proposed can be put into manufacturing and released in the market. Also, one additional promised feature that we could not implement was that the frame could be mounted onto monitors of any Laptop PCs or Desktop PCs. This was not possible for us because the interconnects between the emitters, sensors and the controller were not that intact and would disturb the entire setup. It can also be implemented by just soldering the interconnects or by taking a simple PCB and laying paths on it to the controller that can be easily concealed in the outer frame.

VII. REFERENCES

[1] Bastron Glass Keyboard
http://www.bastron.com.cn/products_detail.php?id=16&cid=1
Cost of it:
https://www.amazon.com/gp/product/B018Q7MFBK/ref=s9_dcacsd_dcoop_bw_c_x_3_w

[2] Brookstone Glass Keyboard
http://www.brookstone.com/pd/transparent-wireless-bluetooth-glass-keyboard/974831p.html

[3] Teensy 3.6
https://www.pjrc.com/store/teensy36.html

[4] IR LED Emitter
https://www.adafruit.com/products/387

[5] IR Receiver Sensor
https://www.adafruit.com/products/388