

Design a Smart Wearable for Interface with Android Systems

Rubina Nehal¹, P. Sayanna², P. Santosh³

¹M.Tech Scholar, Sudheer Reddy College of Engineering & Technology Women, Nizamabad, Telangana

²HOD, ECE Department, Sudheer Reddy College of Engineering & Technology Women, Nizamabad, Telangana

³Associate Prof., ECE Department, Sudheer Reddy College of Engineering & Technology Women, Nizamabad, Telangana

¹ayesha.sabair74@gmail.com, ²sayanna6367@gmail.com, ³santu_523@yahoo.co.in

Abstract: A Bluetooth client device is connected to the controller which is used to communicate with the android smartphone. This Bluetooth device which is paired with the smartphone will send data when the switches are triggered. In the application, call will be enabled when a particular switch is pressed, message will be enabled when a particular switch is pressed, song will be played when a particular switch is pressed, song will be stopped when a particular switch is pressed. To each switch interfaced, a simple application will be executed at android smartphone.

Keywords: Android Application, Bluetooth Communication between Arduino UNO (Controller) and Android Application

I. INTRODUCTION

The calculator watch, introduced in the 1980s, was one original piece of widespread worn electronics. Ilya Fridman designed a Bluetooth headset into a pair of earrings with a hidden microphone. Twitter users can wear a "Pocket Tweet" using a Java application and cutting out and applying a Twitter text bubble to a person's shirt, one example of Do-it-yourself wearable tech that was part of an art exhibit for the Wearable Technology AIR project in spring 2009.

In the proposed solution, a simple, smart, low cost, secure hand Wearable Device is designed which communicates with the Android Devices simply using Bluetooth. The simple application is to alert the Police when Women is in Emergency.

A Bluetooth client device is connected to the controller which is used to communicate with the android smartphone. This Bluetooth device which is paired with the smartphone will send data when the switches are triggered. In the application, call will be enabled when a particular switch is pressed, message will be enabled when a particular switch is pressed, song will be played when a particular switch is pressed, song will be stopped when a particular switch is pressed. To each switch interfaced, a simple application will be executed at android smartphone.

II. SYSTEM ARCHITECTURE

The system architecture of this proposed system is divided into two different blocks.

Hardware End: In hardware implementation we have four major hardware blocks.

ARDUINO UNO: The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

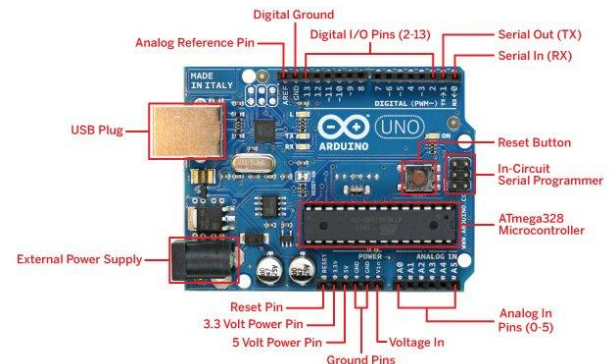


Figure 1: Arduino UNO Layout

Bluetooth Transceiver: Controlling communication is done with the help of Bluetooth transceiver. BT_BOARD V1.04 is the model which has used in Wearable device.

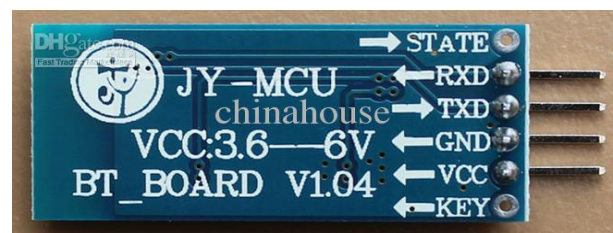


Figure 2: Bluetooth Transceiver

Switch Circuitry: Switches are used to control the ANDROID APP with the help of Bluetooth, switches plays key role in this, switches will be connected to the ARDUINO UNO. Below diagram shows circuit connection between switches and ARDUINO UNO.

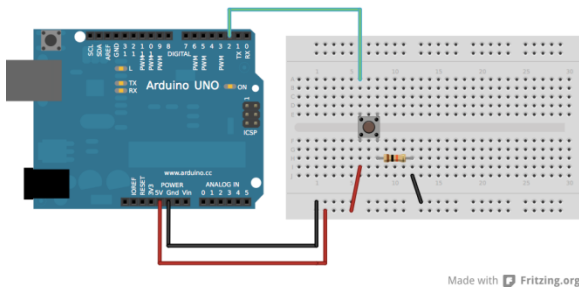


Figure 3: Interfacing Switch with Arduino

Power Circuitry: Power Supply block is designed and developed to generate power source for the ARDUINO (ARDUINO UNO) controller and its relevant components.

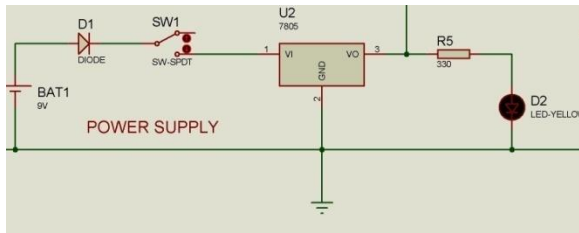


Figure 4: Power Circuit Schematic Diagram

Block Diagram

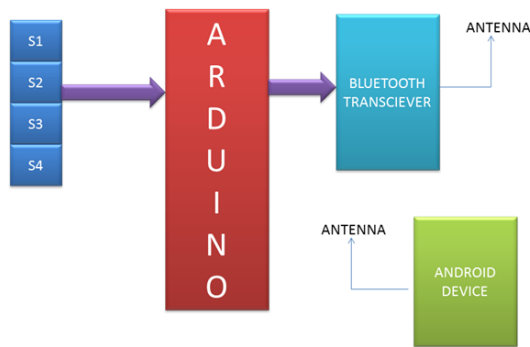


Figure 5: Block Diagram

ANDROID END: The BLUETOOTH module interfaced at ARDUINO UNO will send a no. of characters like 'a', 'b', 'c' so on as per our requirement. We have designed an Android Application so that when a character receives from ARDUINO UNO Android Smart will perform respected task as we have given in Android Application.

III. IMPLEMENTATION

Hardware: In hardware implementation, ARDUINO UNO Controller plays a key role in monitoring and controlling the security system. Low-power consumption ARDUINO UNO (ATMEGA 328P) operating at 3.3V, 50uA is designed and mounted on a PCB along with

Reset Circuit and a Clock Circuit. ATMEGA 328, an 8-bit microcontroller with AVR architecture and having 14 GPIO lines with a program memory of 32KB and a data memory of 512Bytes.

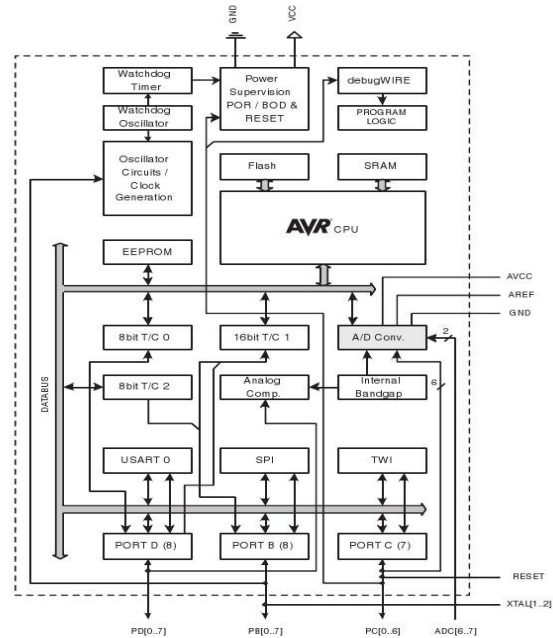


Figure 6: ATMEGA328 Architecture

In Hardware we have used mainly for communication BLUETOOTH transceiver and for input SWITCHES has interfaced to the controller and controller is ATMEGA 328P. Earlier we have seen each individual module circuit connection with the controller, let's see their specifications

Switches: The inputs of controller are switches. Each switch performs its individual task as we discussed, below will show configuration of switch.

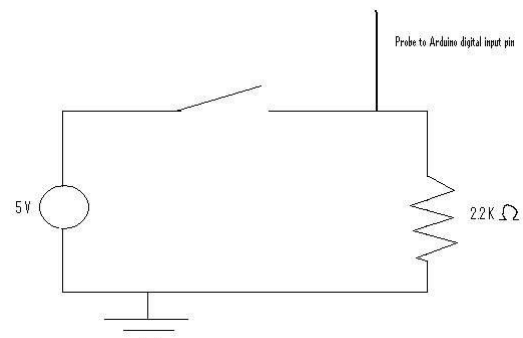


Figure 7: Switch Circuit

Here we are connecting switch in pull down mode, it releases in LOW (LOGIC 0) condition. The connection between Switch and ARDUINO UNO is showed above

Bluetooth: Bluetooth is a Transceiver that means it can be Transmit the data as well Receive the data. In this project Bluetooth transmit the input data (the data comes from the switches) to the Android Smart Phone, by our Android Application it perform its respected task. Connection of Bluetooth should be like, it operates on 5v DC and we have mainly 4 pins one is power supply, second RX (Receiver), third TX (Transmitter) and fourth is ground. Rx pin should be connected to UART1 of Tx pin of ARDUINO UNO and Tx pin should be connected to UART1 of Rx of ARDUINO UNO. Let's see connection between BLUETOOTH and ARDUINO UNO

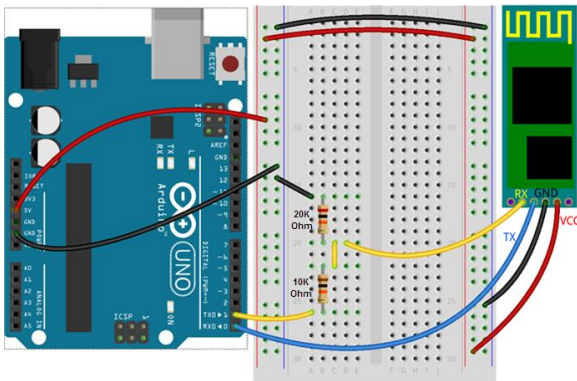


Figure 8: Bluetooth interfaced with Arduino UNO (ATMEGA 328)

Following is the final schematic diagram of the paper proposed:

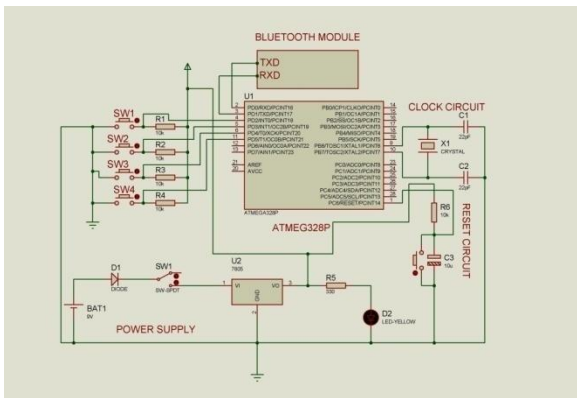


Figure 9: Schematic Diagram

Bluetooth was interfaced at UART of ARDUINO UNO ATMEGA 328 as per Figure 8 which transmits the input data. Inputs (switches) are connected to 4, 5, 6, 11 pins of ARDUINO UNO.

Software: Here, to program ARDUINO UNO controller, ARDUINO software was used as a cross-compiler and programmer. Android application was designed using Android Developer Tools with Eclipse.

IV. ALGORITHM & FLOWCHART

Algorithm:

Step 1: Initialize the Serial Communication.

Step 2: Set the Baud Rate 9600.

Step 3: Initialize Switches.

Step 4: Check whether any switch is pressed or not

Step 5: If the switch is pressed, then transmit its respective character to the android smartphone.

SWITCH	CHARACTER	ACTION
S1 is PRESSED	A	SONG PLAY
S2 is PRESSED	B	SONG STOP
S3 is PRESSED	C	CALL TO EMERGENCY NO
S4 is PRESSED	D	MSG TO EMERGENCY NO

Figure 10: Task Truth Table

Flowchart:

The flowchart of this paper is divided into three sub parts. The first flowchart is for ARM evaluation, and the second one is for Image Processing, and the last one is for web server interface.

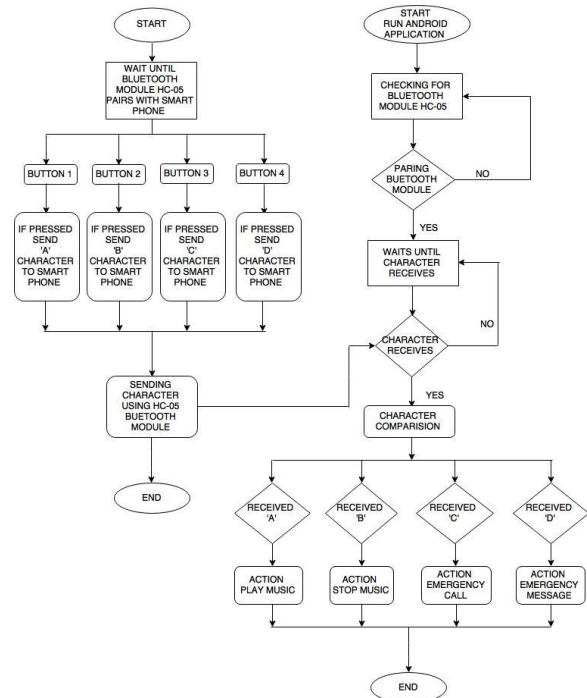


Figure 11: Flow Chart

V. RESULTS



Figure 12: Final Prototype 1

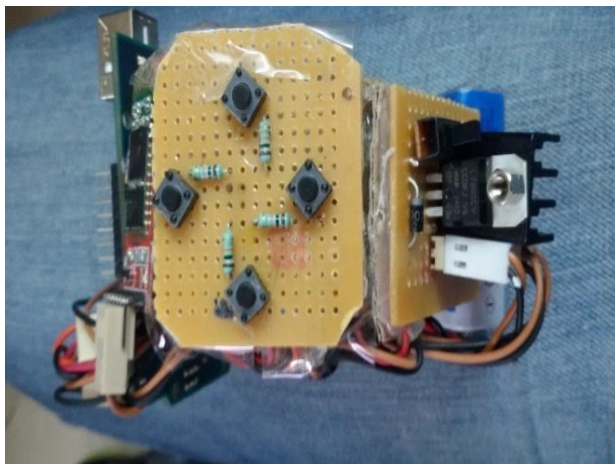


Figure 13: Final Prototype 2



Figure 14: Final Prototype 3

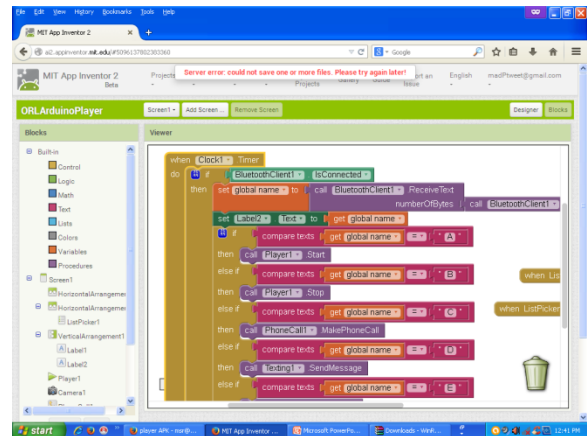


Figure 15: Final Result Analysis

VI. CONCLUSION

Smart, Low-Cost, Simple and Secured wearable device has designed and communicated with Android Smartphone with API Level 7 or greater.

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to Prof. Sayanna, HOD, ECE, ECE Department, Sudheer Reddy College of Engineering & Technology (Women), as well as our Dr. K. S Srikanth, Principal, ECE Department, Sudheer Reddy College of Engineering & Technology (Women), who gave me the golden opportunity to do this wonderful project on the topic (Wearable Technology), which also helped me in doing a lot of Research and i came to know about so many new things we are really thankful to them. And, secondly i would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame.

VII. REFERENCES

- [1] <https://www.arduino.cc/>
- [2] <http://playground.arduino.cc/Main>
- [3] <https://learn.sparkfun.com/tutorials/installing-arduino-ide>
- [4] <https://github.com/arduino/>
- [5] <https://developer.android.com/training/>
- [6] <https://www.udacity.com/course/android-development-for-beginners--ud837>
- [7] <http://www.coreservlets.com/android>
- [8] <http://www.instructables.com/id/Modify-The-HC-05-Bluetooth-Module-Defaults-Using-A/>

[09]http://www.computacaonaescola.ufsc.br/?page_id=536&lang

[10]<https://bytedebugger.wordpress.com/2014/06/27/tutorial-how-to-connect-an-android-device-with-arduino-and-bluetooth/>

[11]<http://www.electfreaks.com/677.html>

[12]<http://stackoverflow.com/questions/10327506/android-arduino-bluetooth-data-transfer>