BRAC UNIVERSITY

Inspiring Excellence

CSE360: Computer Interfacing
Section: 06

Group: 4

Project Title:

Smart Blind Stick With Arduino UNO

Detects obstacles, notify the person using sound, and sends a message to the emergency contact number.

	Name	ID
Submitted by:	Md. Mohiuzzaman	20301361
	Ayon Roy	20201018
	Mohammad Rafid Hamid	20101491
	Sadman Sharif	20101107
	Iftekhar Ahmed	19201097
Submitted To:	Shakir Rouf	Faculty

I. Introduction

The Smart Blind Stick is an innovative project designed to assist visually impaired individuals in navigating their surroundings with ease and safety. This device is equipped with multiple ultrasonic sensors, which detect objects in front, left, and right directions. Whenever an obstacle is detected, the device emits a distinct sound in Morse code indicating the direction of the obstacle.

In addition, the Blind Stick includes a push button for emergency messages, which can be used to alert someone in case of an emergency situation. This feature helps visually impaired individuals to seek assistance quickly and easily.

Also, the GSM module is an important component of this project as it enables the device to send messages to a designated phone number in case of an emergency. The device also includes an LED light, which provides visual feedback to the user in low-light situations.

Hence, the Blind Stick is an excellent project which greatly benefits visually impaired individuals, providing them with a sense of independence, safety, and security while navigating their surroundings.

II. Application Area

The Blind Stick can have various areas of application, including:

Personal Use: The device can be used by visually impaired individuals to navigate their surroundings with ease and safety, both indoors and outdoors. It can help them detect obstacles, avoid collisions, and move around more confidently.

Education: The Blind Stick can also be used as an educational tool for teaching students about the challenges faced by visually impaired individuals and the technology used to assist them.

Raising awareness: It also has the potential to raise awareness among the public about the challenges faced by visually impaired individuals and the importance of developing assistive technologies to improve their quality of life.

Rehabilitation: The device can be used in rehabilitation centers to help visually impaired individuals regain their mobility and independence.

Public Spaces: The Blind Stick can be installed in public spaces such as hospitals, airports, and shopping centers to assist visually impaired individuals in navigating these areas with ease.

Research: The device can also be used for research purposes, to gather data on the mobility patterns and challenges faced by visually impaired individuals and to develop new assistive technologies to address these challenges.

The Smart Blind Stick has several areas of application, and it finds usefulness in personal use by visually impaired individuals, enhancing accessibility and independence for them in various settings.

III. Technology and tools

Software Technologies used for this project:

The Arduino Integrated Development Environment (IDE) is a software application that is used to write and upload the code to the Arduino Uno board. The Blind Stick project uses the Arduino IDE for programming the microcontroller board and its logical interactions with the other components.

Hardware technologies and tools used by this project:

Part	Quantity
Arduino UNO R3	1
HC-SR04 Ultrasonic sensor	3
Buzzer (3-9v) HQ	1
4 Pin Push Switch (Large)	1
SIM900A GSM/GPRS Module	1
Red Led (5mm)	2
9V Battery (PAKKO)	2
Breadboard (Big)	1
LM2596 (Voltage Regulator)	1
9V Battery Connector	2

IV. Programming language

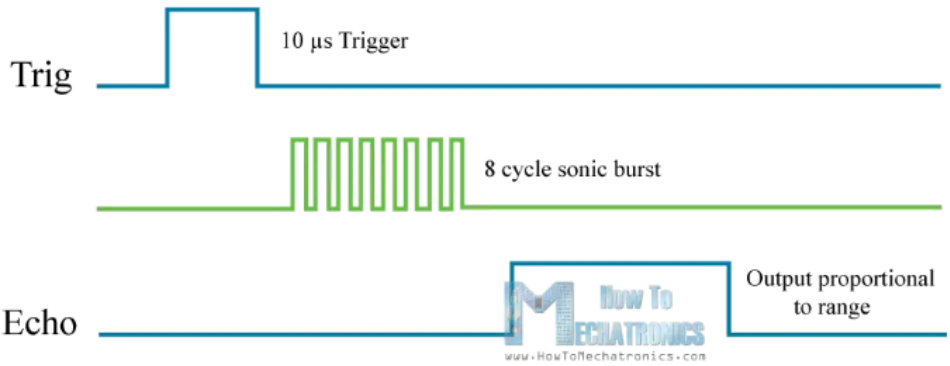
The code for Arduino is written in C++ language, which is a language that can be read and understood by humans. The Arduino code is created in the form of a "sketch" file, which contains special methods and functions. When this sketch file is processed and compiled, it is translated into machine language, which the Arduino can understand and execute as instructions.

V. Working mechanism of Sensors

Ultrasonic/ Sonar Sensor:

Ultrasonic sensors operate by sending out high-frequency sound waves continuously to the environment. Whenever an obstacle comes close to the sensor the wave reflects off that object and returns to the sensor. The sensor detects the time it takes for sound waves to return and uses the speed of sound to figure out the distance to the object. In our case, we use three ultrasonic sensors in the blind stick to detect the obstacle nearby to the stick. We have an ultrasonic sensor at the front side of the stick which continuously emits high frequency waves and whenever the wave detects any object it sends a signal to the buzzer. The buzzer sounds like a morse code of the alphabet 'F' which gives the indication of an obstacle in front of the blind man. In the same manner there are also two more ultrasonic sensors to the left and right side of the stick which gradually sound the buzzer like morse code of 'L' and 'R' whenever any obstacle is found nearby. There also has a LED to indicate the sensors are on work or power is supplied for working of the sensors. GSM Module We use a GSM module named GSM 900A mini to our Blind stick. It connects to a mobile network and transmits data using GPRS technology. It communicates with a microcontroller or mobile phone using AT (Attention) commands. It is a simple and efficient way to enable wireless communication in devices. The device is placed in the stick and there is a connector switch attached to the stick. The GSM module will be connected to the Arduino uno for power supply and the power will come through a converter as GSM module 900A is not capable of taking power directly from Arduino uno. There is a sim card in the GSM which establishes the mobile network to transmit data. Whenever the blind man will be in danger, he can press a button to connect the GSM and whenever it is connected it sends a message to a beforehand added trusted person's number that the blind man is in danger.

To produce ultrasound, we must activate the Trig pin by setting it to a High State for 10 µs. This will emit an 8-cycle burst of ultrasound that will travel at the speed of sound. The Echo pin will immediately become High and begin listening for the ultrasound to bounce back after hitting an object. If no object is detected or no pulse is reflected back, the Echo pin will return to a Low state after 38ms.



When an object reflects an ultrasonic burst generated by setting the Trig pin on a High State for 10 µs, the Echo pin goes high and listens for the reflected pulse. If no reflected pulse is received, the Echo pin times out after 38ms. However, if a reflected pulse is received, the Echo pin goes down sooner than 38ms. By measuring the amount of time the Echo pin was high, we can calculate the distance between the sensor and the object using the formula ***Distance = Speed x Time***. The speed of sound is a known value of 340m/s, and we need to divide the end result by 2 because we are measuring the time it takes for the sound wave to travel to the object and back.

VI. Connection with ICs

Here, the IC is ARDUINO UNO R3 which has the following devices connected with it:

1. **Ultra sonar1:**
    a. +5 Vcc
    b. GND

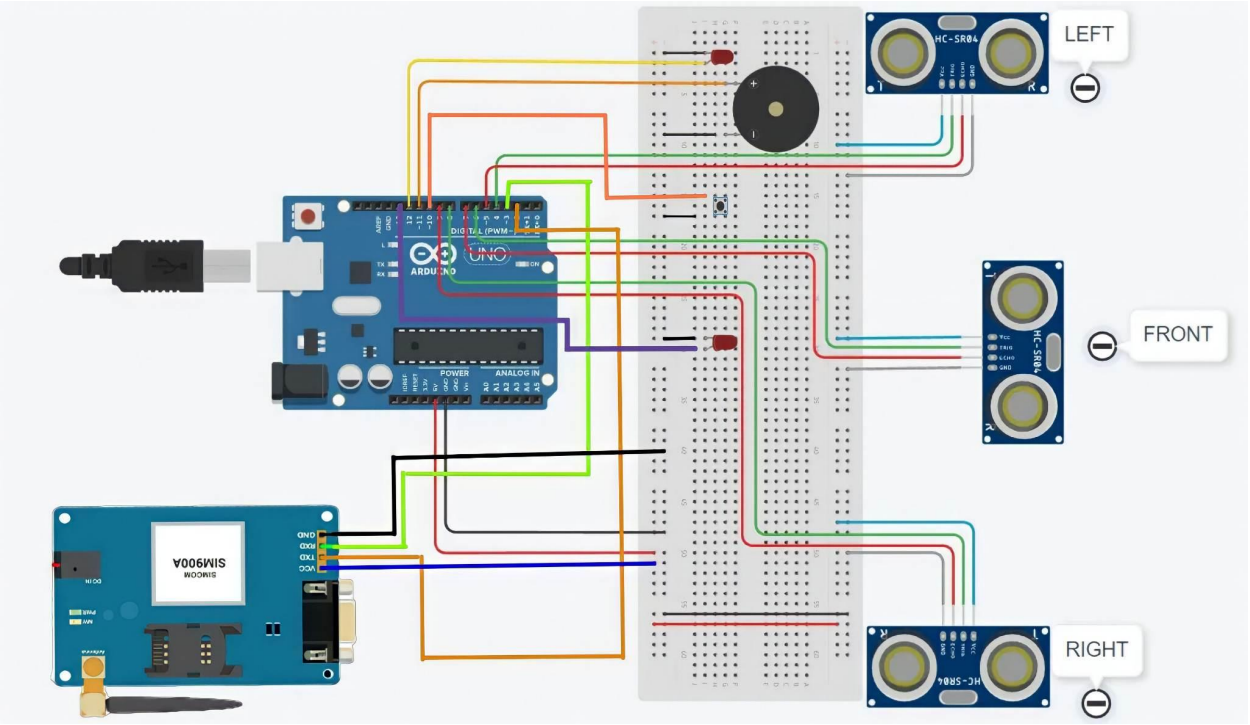
c. Trigpin1 connected with digital pin 4 of Arduino UNO R3.
d. Echopin1 connected with digital pin 5 of Arduino UNO R3.

2. **Ultra sonar2:**

a. +5 Vcc
b. GND
c. Trigpin1 connected with digital pin 6 of Arduino UNO R3.
d. Echopin1 connected with digital pin 7 of Arduino UNO R3.

3. **Ultra sonar3:**

a. +5 Vcc
b. GND
c. Trigpin1 connected with digital pin 8 of Arduino UNO R3.
d. Echopin1 connected with digital pin 9 of Arduino UNO R3.



4. **LED-1:**

a. +5 Vcc connected to digital pin 12 of Arduino UNO R3.

b. GND

5. **LED-2:**

    a. +5 Vcc connected with digital pin 13 of Arduino UNO R3.
    b. GND

6. **Push Button (2-Pin):**

    a. +5 Vcc connected with digital pin 10 of Arduino UNO R3.
    b. GND

7. **Buzzer:**

    a. +5 Vcc connected with digital pin 11 of Arduino UNO R3.
    b. GND

8. **GSM Module:**

    a. +5 Vcc
    b. TXD 5v pin connected to digital pin 2 of Arduino UNO R3.
    c. RXD 5v pin connected to digital pin 3 of Arduino UNO R3.
    d. GND

VII. Data flow from sensors through ICs to I/O devices

We utilized various components and mechanisms to detect objects and obstacles and provide feedback to the user. The data flow in this project can be explained as follows:

Ultrasonic Sensors: The project uses three ultrasonic sensors to detect objects in front, left, and right directions. These sensors emit high-frequency sound waves and measure the time taken for the sound waves to bounce back from obstacles. The data from these sensors is in the form of analog signals, which need to be converted to digital signals for processing.

Analog to Digital Conversion: The analog signals from the ultrasonic sensors are fed into an Analog to Digital Converter (ADC) IC. The ADC IC converts the analog signals to digital signals that can be processed by the microcontroller.

Microcontroller: The Arduino Uno microcontroller receives the digital signals from the ADC IC and processes them to determine the distance and direction of the obstacle. Based on the data received, the microcontroller sends signals to the buzzer and LED light to provide feedback to the user.

Buzzer and LED Light: The buzzer produces a sound in Morse code indicating the direction of the obstacle detected. For example, a short beep can indicate an obstacle in front, while a long beep can indicate an obstacle to the right or left. The LED light also provides visual feedback to the user in low-light situations.

Push Button and GSM Module: The project also includes a push button for emergency messages and a GSM module for sending messages to a designated phone number. The push button is connected to the microcontroller, which sends a message through the GSM module to a predefined phone number in case of an emergency situation.

As a result, we can see that the data flow in the Blind Stick project begins with the analog signals from the ultrasonic sensors, which are converted to digital signals through an ADC IC. The digital signals are then processed by the microcontroller and are sent to the buzzer and LED light for providing feedback to the user. The push button and GSM module also provide additional functionality for emergency situations.

VIII. Code

```
#include <SoftwareSerial.h>
SoftwareSerial gsmSerial(3, 2);

#define trigPin1 4
#define echoPin1 5

#define trigPin2 6
#define echoPin2 7

#define trigPin3 8
#define echoPin3 9

#define LED  13
#define PUSH_BUTTON  10
```

```
void setup() {
Serial.begin (9600);
pinMode(trigPin1, OUTPUT);
pinMode(echoPin1, INPUT);

pinMode(trigPin2, OUTPUT);
pinMode(echoPin2, INPUT);

pinMode(trigPin3, OUTPUT);
pinMode(echoPin3, INPUT);

pinMode(11, OUTPUT);

pinMode(12, OUTPUT);

pinMode(PUSH_BUTTON, INPUT_PULLUP);
pinMode(LED, OUTPUT);

gsmSerial.begin(9600);
delay(1000);
gsmSerial.println("AT+CMGF=1"); // set SMS text mode
delay(1000);
gsmSerial.println("AT+CNMI=2,2,0,0,0"); // set SMS notification mode
delay(1000);
}

void loop() {

  if (digitalRead(PUSH_BUTTON) == LOW) {
    digitalWrite(LED, HIGH); // turn on LED
    // send SMS message
    gsmSerial.println("AT+CMGS=\"01878759001\""); // replace with your
phone number
    delay(1000);
    gsmSerial.print("Emergency:\ncall me at 01720874666");
    gsmSerial.write(0x1A);
    delay(1000);
  } else {
```

```
    digitalWrite(LED, LOW); // turn off LED
  }

char* l = ".-..";
char* f = "..-.";
char* r = ".-.";


long duration1, distance1;
digitalWrite(trigPin1, LOW);
delayMicroseconds(2);
digitalWrite(trigPin1, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin1, LOW);
duration1 = pulseIn(echoPin1, HIGH);
distance1 = (duration1/2) / 29.1;

long duration2, distance2;
digitalWrite(trigPin2, LOW);
delayMicroseconds(2);
digitalWrite(trigPin2, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin2, LOW);
duration2 = pulseIn(echoPin2, HIGH);
distance2 = (duration2/2) / 29.1;


long duration3, distance3;
digitalWrite(trigPin3, LOW);
delayMicroseconds(2);
digitalWrite(trigPin3, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin3, LOW);
duration3 = pulseIn(echoPin3, HIGH);
distance3 = (duration3/2) / 29.1;


if (distance1 <= 10){

for (int i=0; i<4; i++){
```

```
  if (l[i]=='.'){
    digitalWrite(11,HIGH);
    digitalWrite(12,HIGH);
    delay(200);
    digitalWrite(11,LOW);
    digitalWrite(12,LOW);
    delay(1000);
  }
  else if (l[i]=='-'){
    digitalWrite(11,HIGH);
    digitalWrite(12,HIGH);
    delay(2000);
    digitalWrite(11,LOW);
    digitalWrite(12,LOW);
    delay(1000);
  }
}

  // tone(11,400,10);
  // delay(200);
}
else {
  digitalWrite(11,LOW);
  digitalWrite(12,LOW);

}
delay(50);


if (distance2 <= 10){
for (int i=0; i<4; i++){
  if (f[i]=='.'){
    digitalWrite(11,HIGH);
    digitalWrite(12,HIGH);
    delay(200);
    digitalWrite(11,LOW);
    digitalWrite(12,LOW);
    delay(1000);
  }
  else if (f[i]=='-'){
```

```
    digitalWrite(11,HIGH);
    digitalWrite(12,HIGH);
    delay(2000);
    digitalWrite(11,LOW);
    digitalWrite(12,LOW);
    delay(1000);
  }
}

}
else {
  digitalWrite(11,LOW);
  digitalWrite(12,LOW);

}
delay(50);


if (distance3<= 10){
for (int i=0; i<3; i++){
  if (r[i]=='.'){
    digitalWrite(11,HIGH);
    digitalWrite(12,HIGH);
    delay(200);
    digitalWrite(11,LOW);
    digitalWrite(12,LOW);
    delay(1000);
  }
  else if (r[i]=='-'){
    digitalWrite(11,HIGH);
    digitalWrite(12,HIGH);
    delay(2000);
    digitalWrite(11,LOW);
    digitalWrite(12,LOW);
    delay(1000);
  }
}

}
else {
```

```
  digitalWrite(11,LOW);
  digitalWrite(12,LOW);
}
}
```

IX. Estimated cost analysis

Components	Price
1. Arduino UNO R3	1075.63
2. 9V Battery	79*2=158
3. 9V Battery Connector	34.9*2=69.8
4. SIM900A GSM Module	1185.2
5. ULtrasonic Sensor	94.9*3=284.7
6. Jumper Wire (Male to Male)	42.9
7. Jumper Wire (Male to Female)	42.9
8. Jumper Wire (Female to Female)	42.9
9. Buzzer	21.8
10. LED	1.98*2=3.96
11. Push Switch (4 pin)	7.8
12. Breadboard	155
13. LM 2596	100
14. Metal Walking Stick	500
Total	**3690.59**

X. Responsibilities of each member

Together in google meet we have decided what our project would be and how to utilize the Project.

Md. Mohiuzzaman- Has arranged all the meetings where we have decided what to do, how to do and when to do all our work. Connection with IC`s(Partial), Contributed teamwork. Editor of Data flow from sensors through ICs to I/O devices, Code.

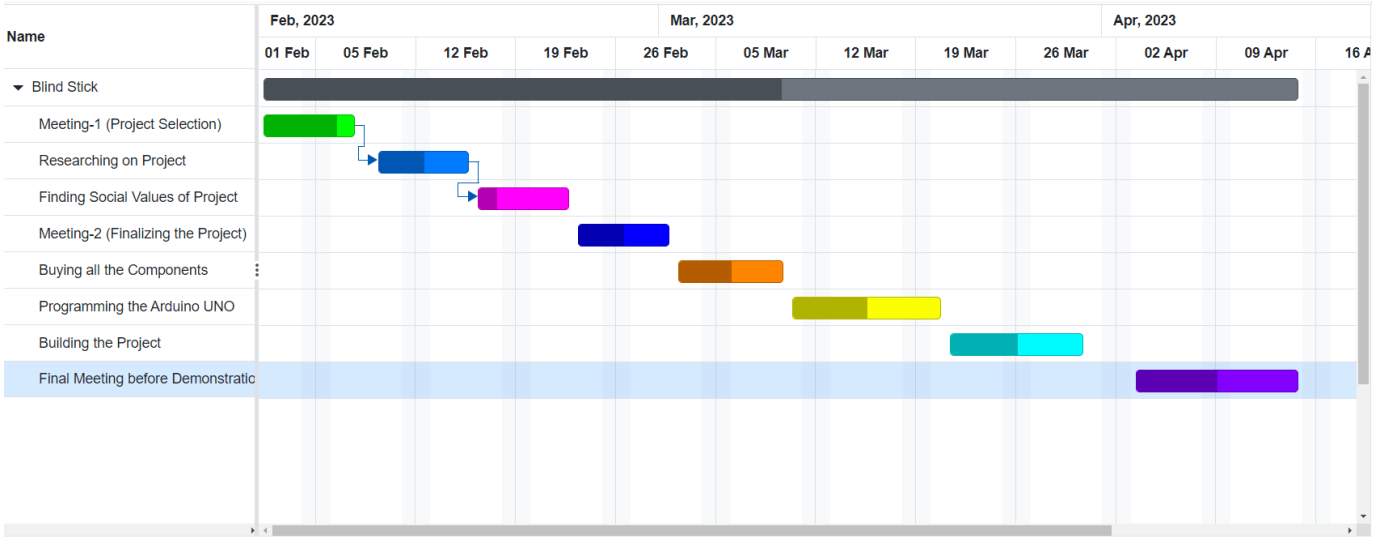
Ayon Roy – Connection with IC`s(Partial), Editor Programming language, Working mechanism of Sensors, Data flow from sensors through ICs to I/O devices, Code.

Sadman Sharif – Workplan (Gantt Chart), Also contributed in the research work with responsibilities & estimated cost analysis, Editor Programming language.

Rafid Hamid – Responsible for cover page Introduction, Application Area, technology and tools. Also contributed to the research work.

Iftekhar Ahmed- Editor of Responsibility of each member. Took part in Introduction, Conclusion and Reference. Contributed teamwork.

XI. Workplan (Gantt Chart)



XII. Conclusion

In conclusion, the Smart Blind Stick is a remarkable device designed to assist visually impaired individuals in navigating their environment with ease and confidence. Equipped with ultrasonic sensors, a push button for emergency messages, a GSM module, and an LED light, this innovative project can greatly enhance the independence, safety, and security of those with visual impairments. By detecting obstacles and providing direction, the Blind Stick can enable visually impaired individuals to move around freely and confidently, making it an invaluable tool for anyone with visual impairments.

XIII. References

- https://store.roboticsbd.com/sensors/22-ultrasonic-sonar-sensor-hc-sr04-robotics-bangladesh.html
- https://www.arrow.com/en/research-and-events/articles/ultrasonic-sensors-how-they-work-and-how-to-use-them-with-arduino#:~:text=Ultrasonic%20sensors%20work%20by%20emitting,return%20after%20hitting%20an%20object.
- https://www.instructables.com/Arduino-Distance-Detector-with-a-Buzzer-and-LEDs/
- https://robodocbd.com/?gclid=CjwKCAjwue6hBhBVEiwA9YTx8JxeFNq4KfY05K76F-q9IYBaZCvh6-jdqMZZ9cERaksGawDT7AarHhoCIGEQAvD_BwE
- https://www.onlinegantt.com/#/gantt
- https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/