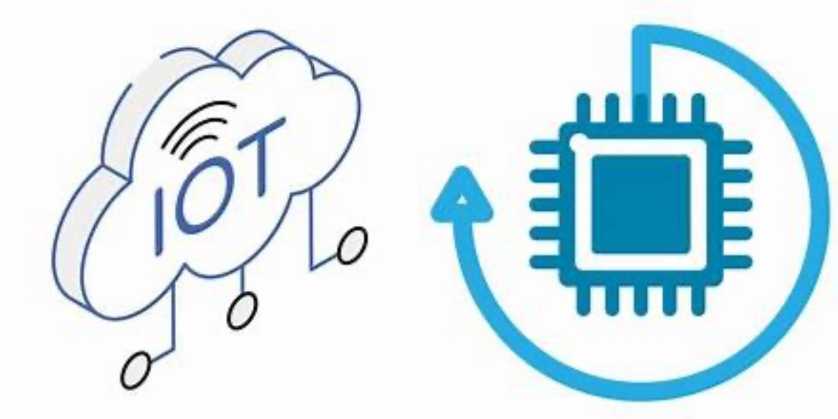
Internet of Things (IoT) Systems



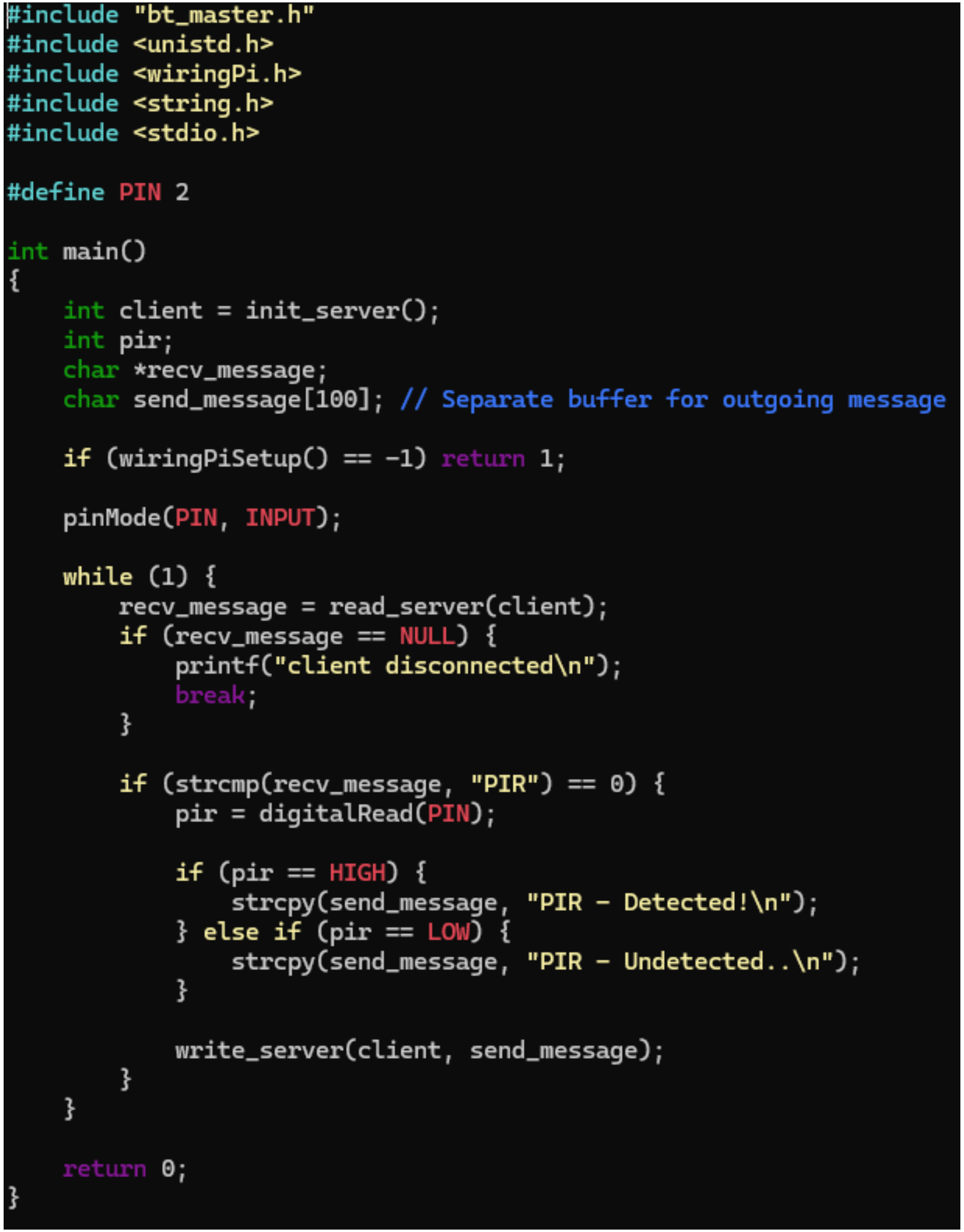
Week 10

Raspberry Pi Programming

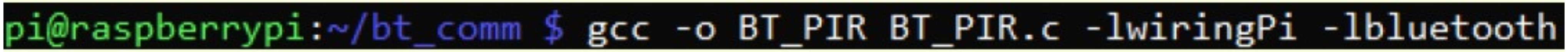
Ikram Syed, Ph.D.
Associate Professor
Department of Information and Communication Engineering
Hankuk University of Foreign Studies (HUFS)

Spring – 2025

PIR Sensor:
Human Detection Sensor

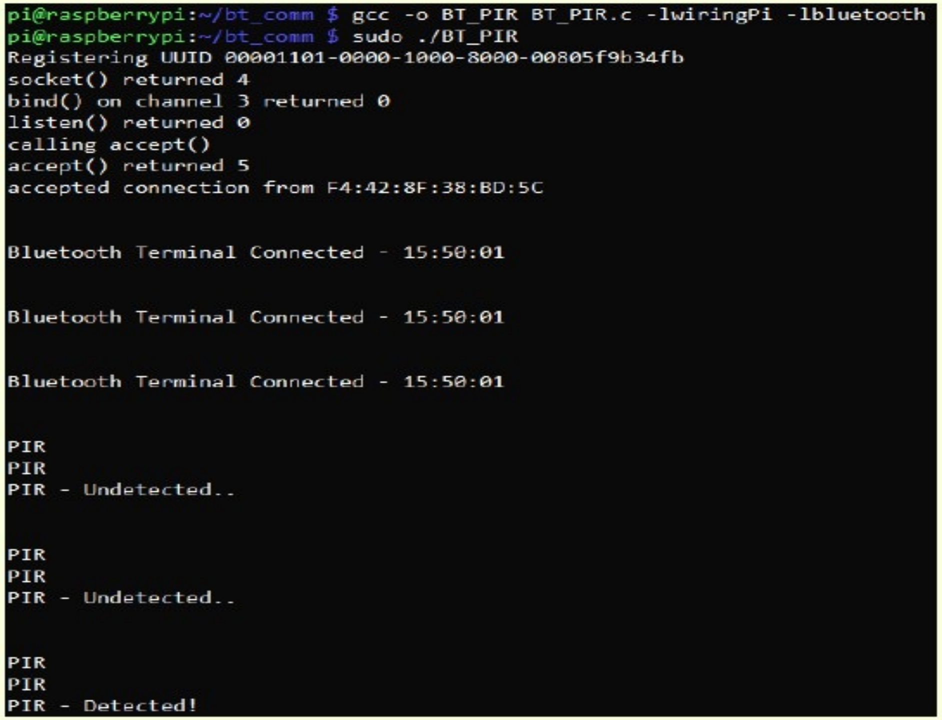


```
#include "bt_master.h"
#include <unistd.h>
#include <wiringPi.h>
#include <string.h>
#include <stdio.h>

#define PIN 2

int main()
{
    int client = init_server();
    int pir;
    char *recv_message;
    char send_message[100]; // Separate buffer for outgoing message

    if (wiringPiSetup() == -1) return 1;

    pinMode(PIN, INPUT);

    while (1) {
        recv_message = read_server(client);
        if (recv_message == NULL) {
            printf("client disconnected\n");
            break;
        }

        if (strcmp(recv_message, "PIR") == 0) {
            pir = digitalRead(PIN);

            if (pir == HIGH) {
                strcpy(send_message, "PIR - Detected!\n");
            } else if (pir == LOW) {
                strcpy(send_message, "PIR - Undetected..\n");
            }

            write_server(client, send_message);
        }
    }

    return 0;
}
```



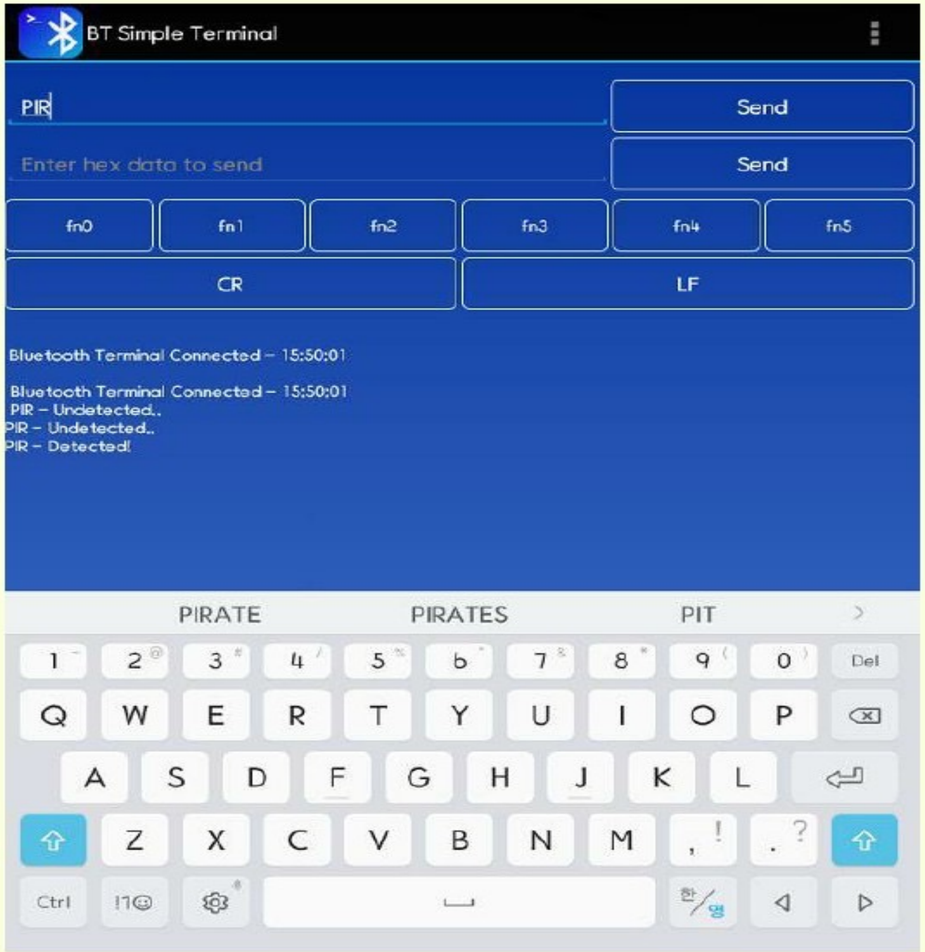
```
pi@raspberrypi:~/bt_comm $ gcc -o BT_PIR BT_PIR.c -lwiringPi -lbluetooth
```

[Figure 4-33] Compiling BT_PIR.c



```
pi@raspberrypi:~/bt_comm $ gcc -o BT_PIR BT_PIR.c -lwiringPi -lbluetooth
pi@raspberrypi:~/bt_comm $ sudo ./BT_PIR
Registering UUID 00001101-0000-1000-8000-00805f9b34fb
socket() returned 4
bind() on channel 3 returned 0
listen() returned 0
calling accept()
accept() returned 5
accepted connection from F4:42:8F:38:BD:5C


Bluetooth Terminal Connected - 15:50:01


Bluetooth Terminal Connected - 15:50:01


Bluetooth Terminal Connected - 15:50:01


PIR
PIR
PIR - Undetected..


PIR
PIR
PIR - Undetected..


PIR
PIR
PIR - Detected!
```

[Figure 4-34] The Screen of Raspberry Pi

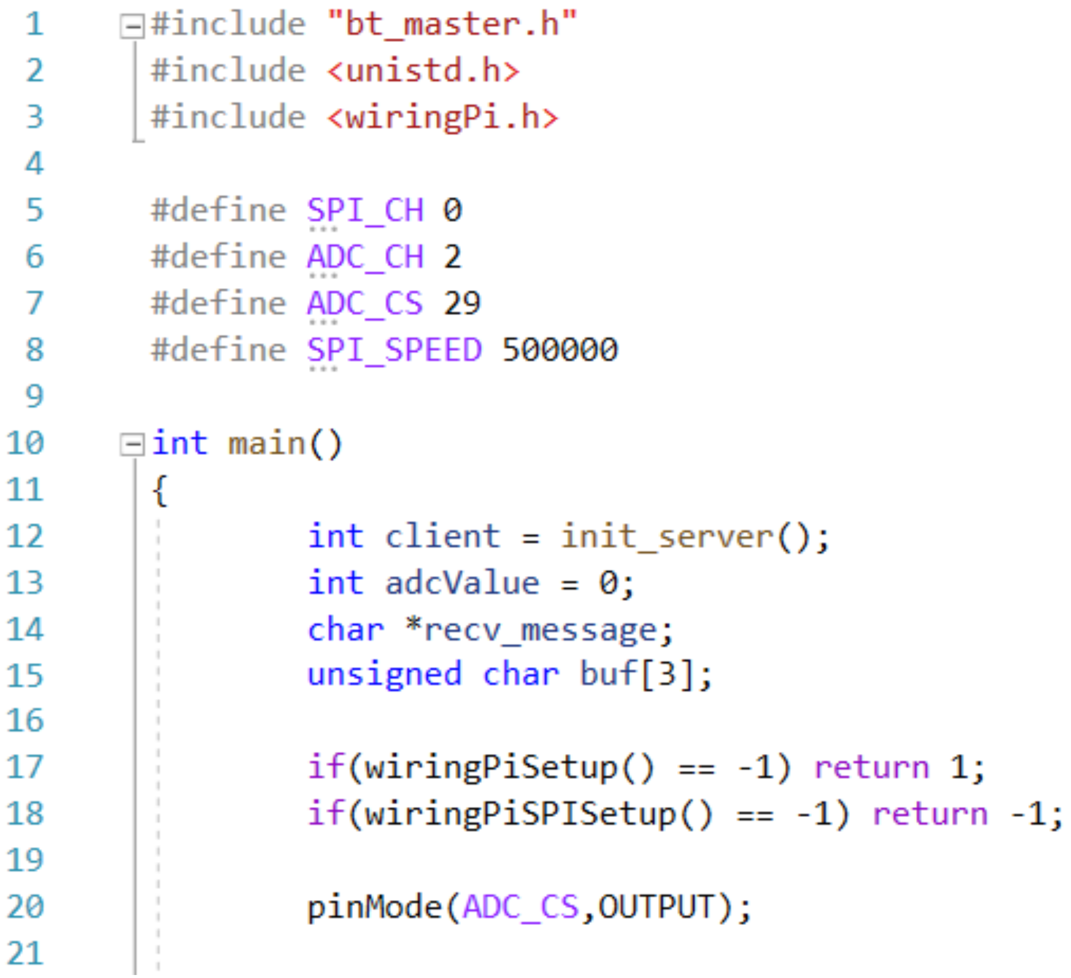


PIR

```
Bluetooth Terminal Connected - 15:50:01

Bluetooth Terminal Connected - 15:50:01
PIR - Undetected..
PIR - Undetected..
PIR - Detected!
```

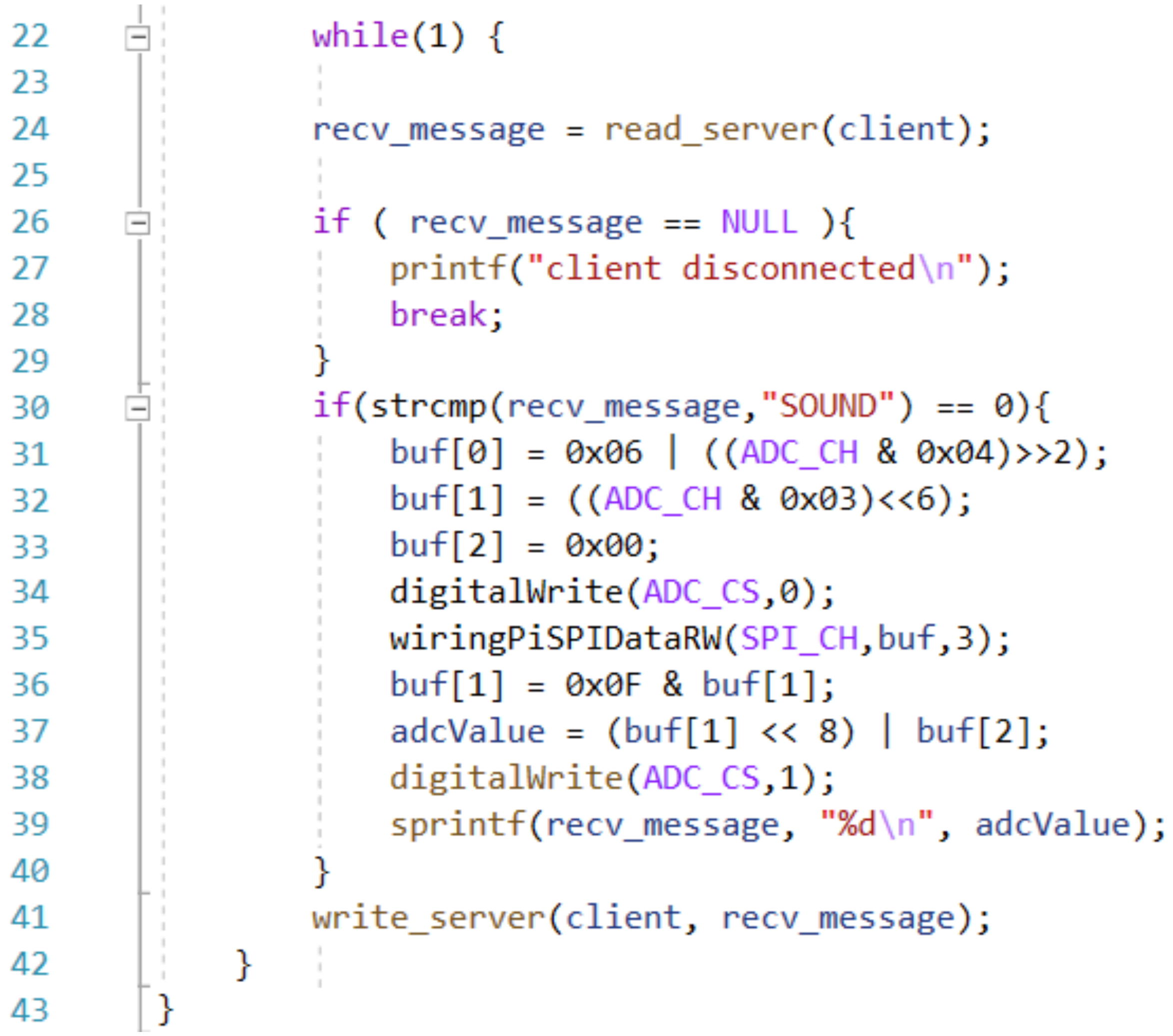
[Figure 4-35] The Screen of Android

Sound Sensor

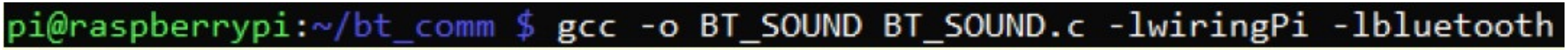


```
#include "bt_master.h"
#include <unistd.h>
#include <wiringPi.h>

#define SPI_CH 0
#define ADC_CH 2
#define ADC_CS 29
#define SPI_SPEED 500000

int main()
{
        int client = init_server();
        int adcValue = 0;
        char *recv_message;
        unsigned char buf[3];

        if(wiringPiSetup() == -1) return 1;
        if(wiringPiSPISetup() == -1) return -1;

        pinMode(ADC_CS,OUTPUT);

```

Sound Sensor

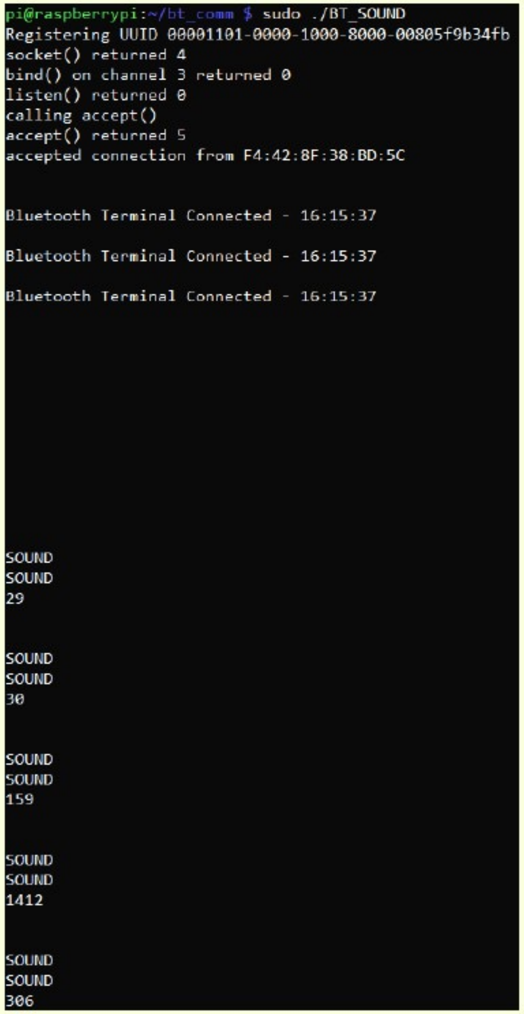


```
        while(1) {

        recv_message = read_server(client);

        if ( recv_message == NULL ){
            printf("client disconnected\n");
            break;
        }
        if(strcmp(recv_message,"SOUND") == 0){
            buf[0] = 0x06 | ((ADC_CH & 0x04)>>2);
            buf[1] = ((ADC_CH & 0x03)<<6);
            buf[2] = 0x00;
            digitalWrite(ADC_CS,0);
            wiringPiSPIDataRW(SPI_CH,buf,3);
            buf[1] = 0x0F & buf[1];
            adcValue = (buf[1] << 8) | buf[2];
            digitalWrite(ADC_CS,1);
            sprintf(recv_message, "%d\n", adcValue);
        }
        write_server(client, recv_message);
    }
}
```



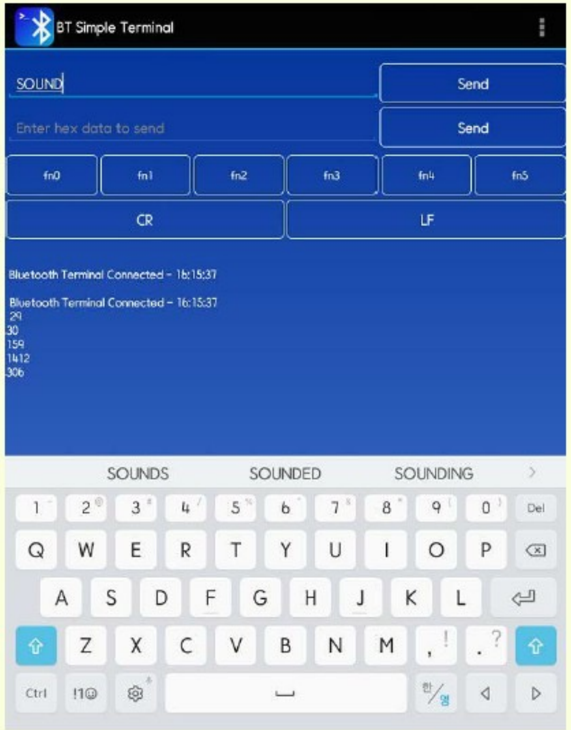
```
pi@raspberrypi:~/bt_comm $ gcc -o BT_SOUND BT_SOUND.c -lwiringPi -lbluetooth
```

[Figure 4-37] Compiling BT_SOUND.c



```
pi@raspberrypi:~/bt_comm $ sudo ./BT_SOUND
Registering UUID 00001101-0000-1000-8000-00805f9b34fb
socket() returned 4
bind() on channel 3 returned 0
listen() returned 0
calling accept()
accept() returned 5
accepted connection from F4:42:8F:38:BD:5C


Bluetooth Terminal Connected - 16:15:37

Bluetooth Terminal Connected - 16:15:37

Bluetooth Terminal Connected - 16:15:37











SOUND
SOUND
29


SOUND
SOUND
30


SOUND
SOUND
159


SOUND
SOUND
1412


SOUND
SOUND
306
```

[Figure 4-38] The Screen of Raspberry Pi

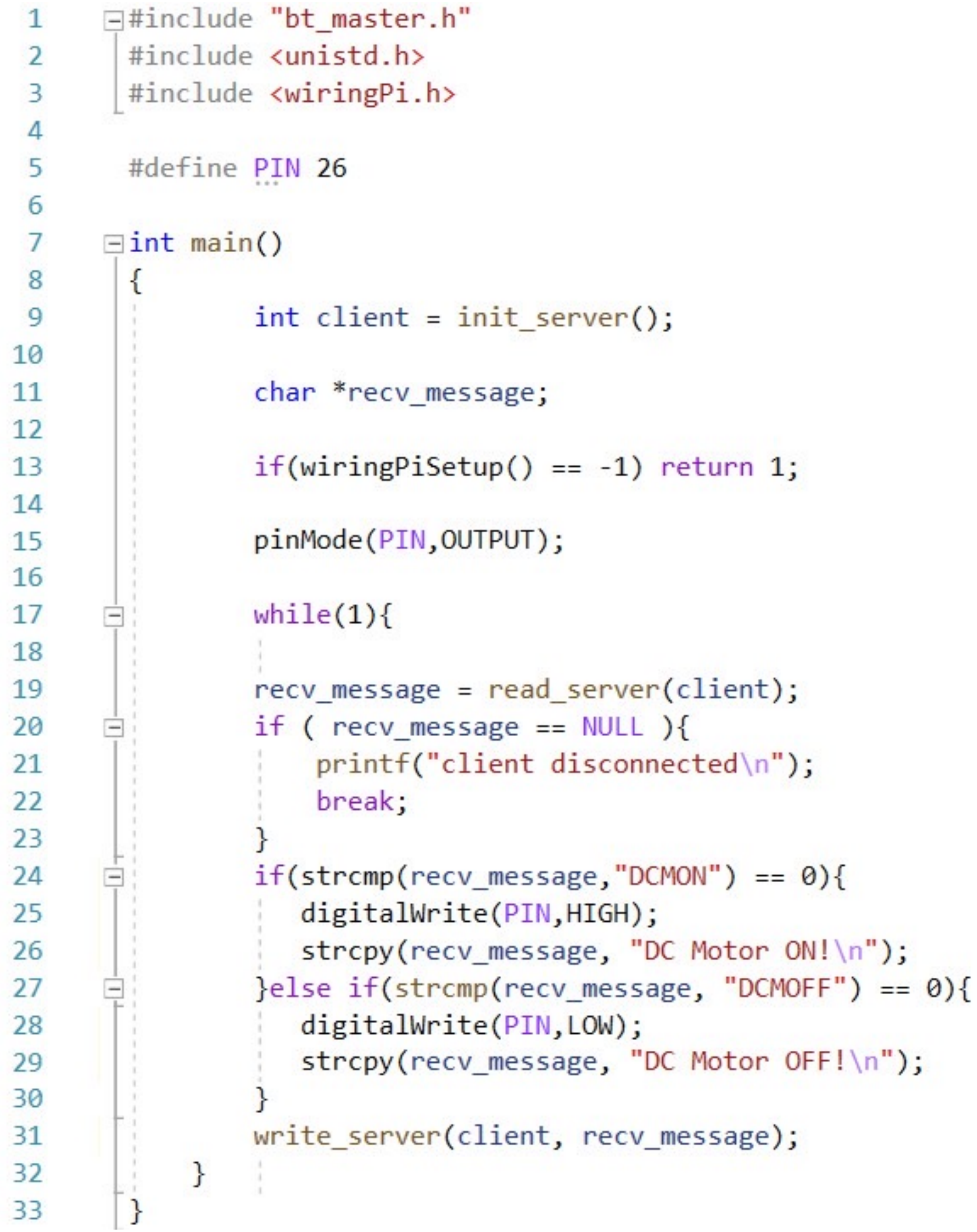


SOUND

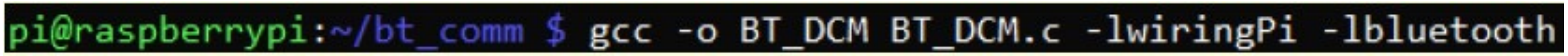
```
Bluetooth Terminal Connected - 16:15:37

Bluetooth Terminal Connected - 16:15:37
29
30
159
1412
306
```

[Figure 4-39] The Screen of Android

DC Motor

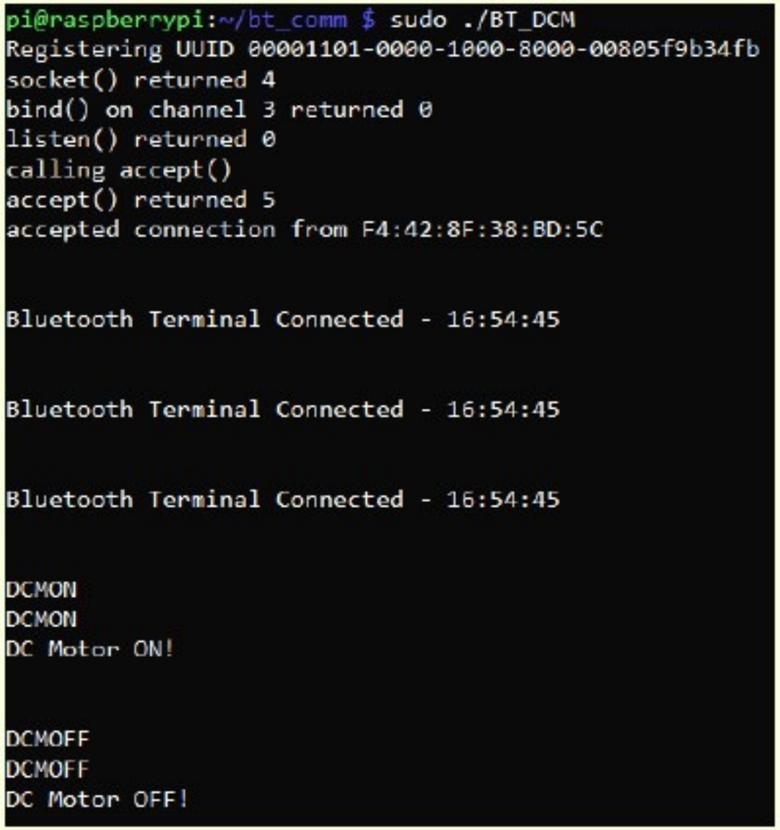


```c
#include "bt_master.h"
#include <unistd.h>
#include <wiringPi.h>

#define PIN 26

int main()
{
        int client = init_server();

        char *recv_message;

        if(wiringPiSetup() == -1) return 1;

        pinMode(PIN,OUTPUT);

        while(1){

        recv_message = read_server(client);
        if ( recv_message == NULL ){
            printf("client disconnected\n");
            break;
        }
        if(strcmp(recv_message,"DCMON") == 0){
           digitalWrite(PIN,HIGH);
           strcpy(recv_message, "DC Motor ON!\n");
        }else if(strcmp(recv_message, "DCMOFF") == 0){
           digitalWrite(PIN,LOW);
           strcpy(recv_message, "DC Motor OFF!\n");
        }
        write_server(client, recv_message);
    }
}
```



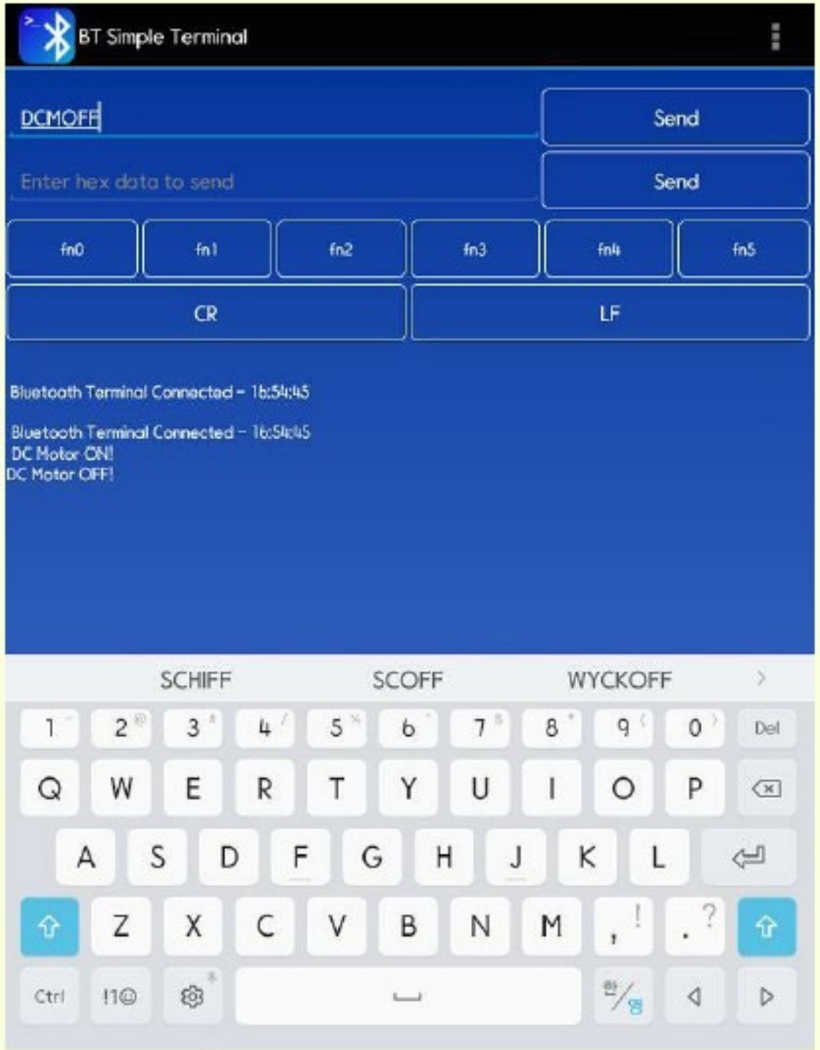
```
pi@raspberrypi:~/bt_comm $ gcc -o BT_DCM BT_DCM.c -lwiringPi -lbluetooth
```

[Figure 4-45] Compiling BT_DCM.c



```
pi@raspberrypi:~/bt_comm $ sudo ./BT_DCM
Registering UUID 00001101-0000-1000-8000-00805f9b34fb
socket() returned 4
bind() on channel 3 returned 0
listen() returned 0
calling accept()
accept() returned 5
accepted connection from F4:42:8F:38:BD:5C


Bluetooth Terminal Connected - 16:54:45


Bluetooth Terminal Connected - 16:54:45


Bluetooth Terminal Connected - 16:54:45


DCMON
DCMON
DC Motor ON!


DCMOFF
DCMOFF
DC Motor OFF!
```

[Figure 4-46] The Screen of Raspberry Pi

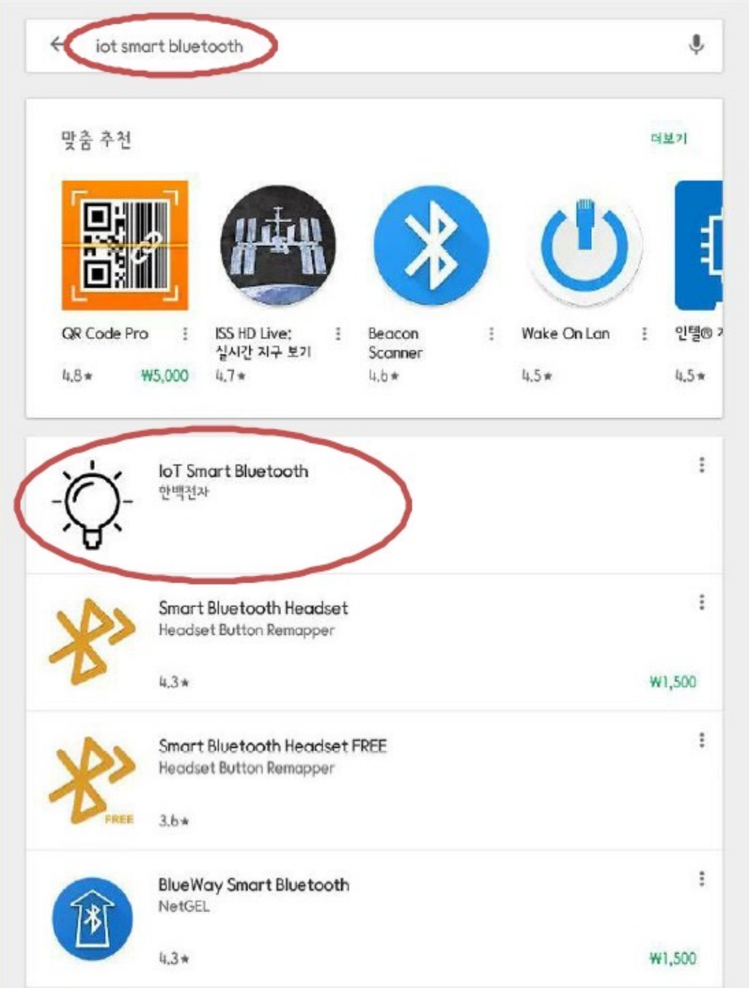


DCMOFF

```
Bluetooth Terminal Connected - 16:54:45

Bluetooth Terminal Connected - 16:54:45
DC Motor ON!
DC Motor OFF!
```

[Figure 4-47] The Screen of Android

Bluetooth IoT System



iot smart bluetooth

맞춤 추천

더보기

QR Code Pro
4.8★ ₩5,000

ISS HD Live: 실시간 지구 보기
4.7★

Beacon Scanner
4.6★

Wake On Lan
4.5★

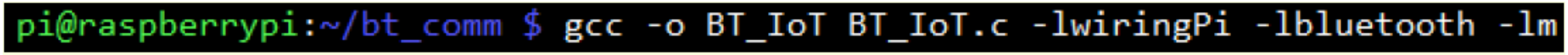
인텔®
4.5★

IoT Smart Bluetooth
한백전자

Smart Bluetooth Headset
Headset Button Remapper
4.3★ ₩1,500

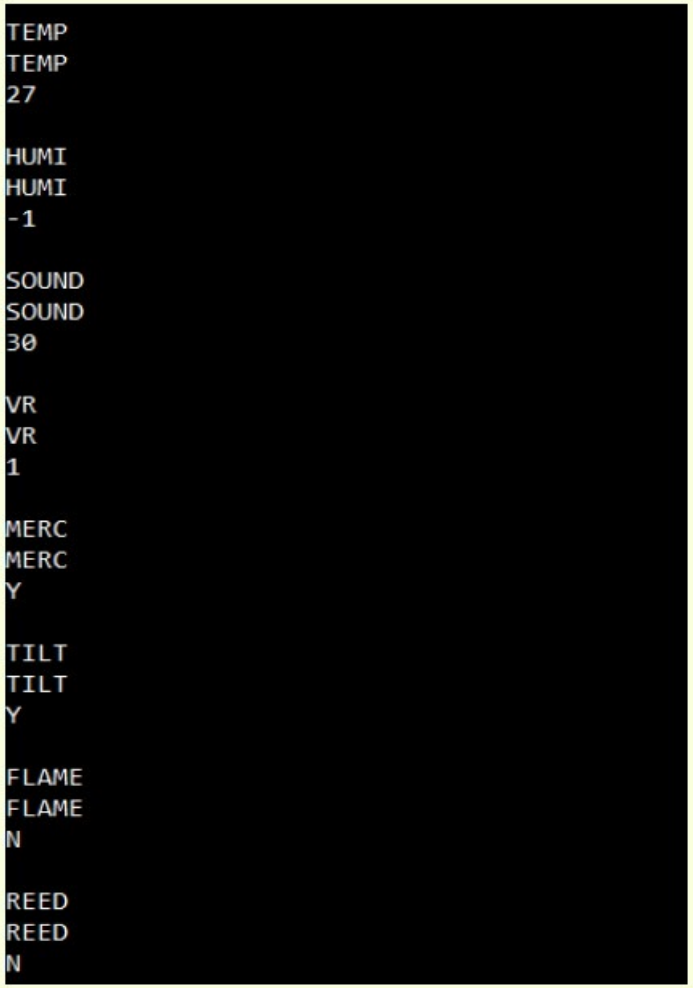
Smart Bluetooth Headset FREE
Headset Button Remapper
3.6★

BlueWay Smart Bluetooth
NetGEL
4.3★ ₩1,500



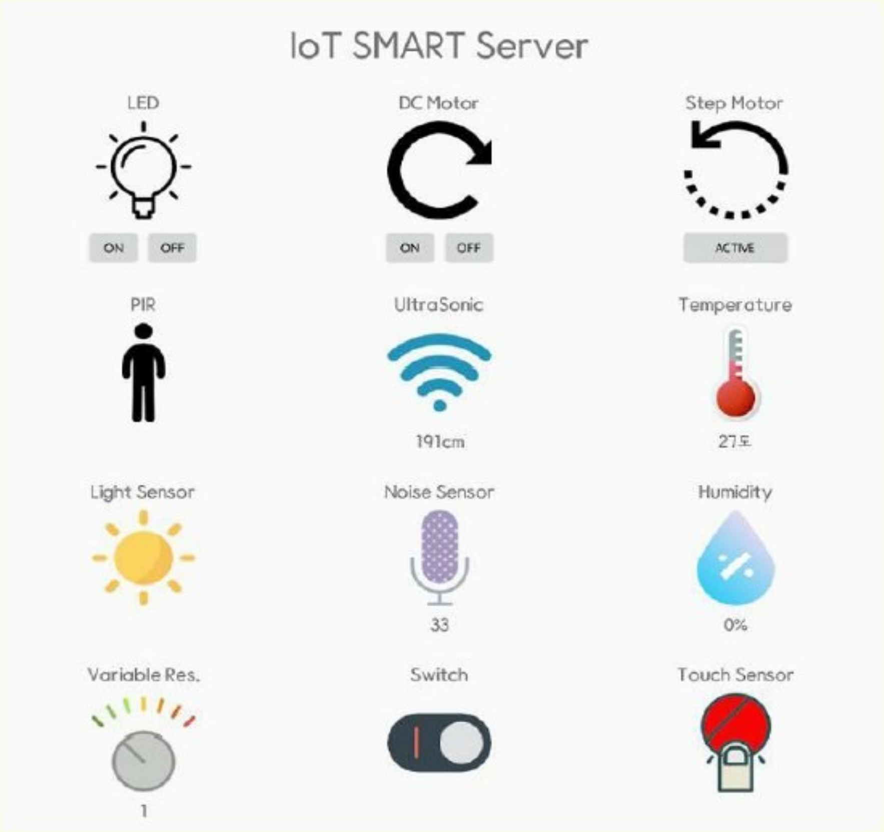
```
pi@raspberrypi:~/bt_comm $ gcc -o BT_IoT BT_IoT.c -lwiringPi -lbluetooth -lm
```

[Figure 4-152] Compiling BT_IoT.c



```
TEMP
TEMP
27

HUMI
HUMI
-1

SOUND
SOUND
30

VR
VR
1

MERC
MERC
Y

TILT
TILT
Y

FLAME
FLAME
N

REED
REED
N
```

[Figure 4-153] The Screen of Raspberry Pi



[Figure 4-154] The Screen of Android

Any Questions!