

Seeed MR60BHA1 Presence, Breathing, and Heartbeat Sensing

Seeed MR60BHA1: Presence, Breathing, and Heartbeat Sensing

 Difficulty **Medium**

 Duration **3 hour(s)**

 Categories **Electronics**

 Cost **60 USD (\$)**

Contents

- Step 1 - About MR60BHA1 60GHz mmWave Module
- Step 2 - Get PCBs for Your Projects Manufactured
- Step 3 - Seeed Studio MR60BHA1 60GHz mmWave Module
- Step 4 - Enhanced Health and Wellness Monitoring:
- Step 5 - Automated Device and Appliance Control:
- Step 6 - Privacy-Protected and Secure:
- Step 7 - Radar module operating range
- Step 8 - Installation method
- Step 9 - How the Project Works
- Step 10 - Project Code
- Comments

Materials

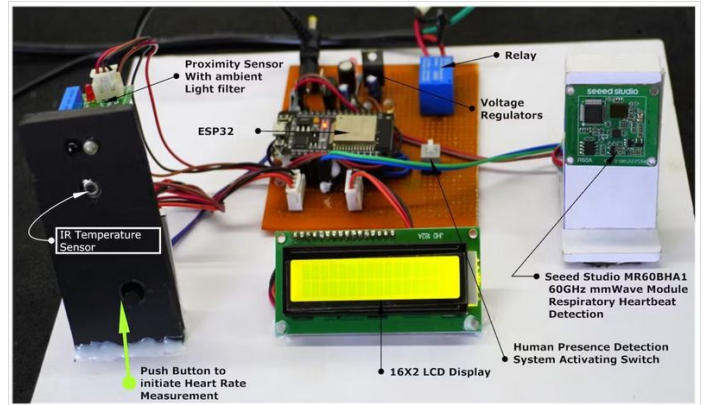
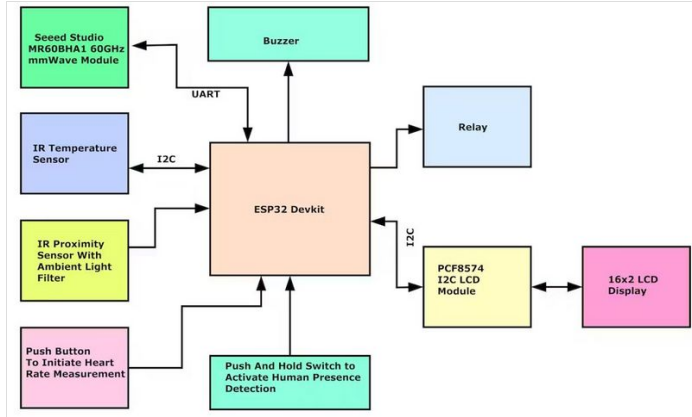
Tools

Step 1 - About MR60BHA1 60GHz mmWave Module

The MR60BHA1 60GHz mmWave Module is a versatile sensor that utilizes Frequency Modulation Continuous Wave (FMCW) detection to accurately measure breathing rate and heart rate, ensuring a completely private and secure environment, free from external interference. Equipped with a built-in standard algorithm and onboard antenna, the unit delivers simultaneous signal output with exceptional precision. It serves as an ideal solution for developing high-accuracy, self-regulating, privacy-protected, and secure biotic radar systems in consumer electronics, healthcare, and industrial applications.

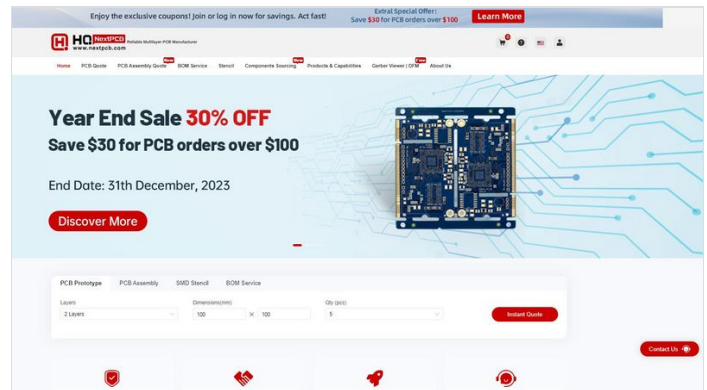
Block Diagram

Beyond heart rate and respiration detection, the radar sensor can detect human presence in the operating area, enabling the creation of an automated device and appliance control system that conserves energy during unoccupied periods. Additionally, the system incorporates an IR temperature sensor to measure body temperature, allowing for illness and stress detection. Upon detecting significant vital variations, the buzzer activates as an alert. Simultaneously, all details are displayed on an LCD display and a mobile phone app via the ESP32 Wi-Fi module. The project involved various components, which can be identified in the Below provided image.



Step 2 - Get PCBs for Your Projects Manufactured

This project was successfully completed because of the help and support from NextPCB. Guys if you have a PCB project, please visit their website and get exciting discounts and coupons. NextPCB offers high-quality, reliable PCB starting at \$1.9, and multilayer starting at \$6.9. Also, everyone can enjoy free PCB assembly for 5 boards! Also, NextPCB is having a year end sale in which anyone can register through their website and get a \$30 Coupon which can be used for ordering PCBs. You can also try HQDFM free online PCB Gerber viewer to check your PCB design and avoid costly errors.



Step 3 - Seeed Studio MR60BHA1 60GHz mmWave Module

The Seeed Studio MR60BHA1 60GHz mmWave Module is a versatile sensor module that utilizes Frequency Modulation Continuous Wave (FMCW) detection technology to accurately measure vital signs, including breathing rate and heart rate. It operates in a completely private and secure environment, free from external interference, making it an ideal solution for privacy-sensitive applications. Equipped with a built-in standard algorithm and onboard antenna, the unit delivers simultaneous signal output with exceptional precision. It serves as an ideal solution for developing high-accuracy, self-regulating, privacy-protected, and secure biotic radar systems in consumer electronics, healthcare, and industrial applications

Key Features:

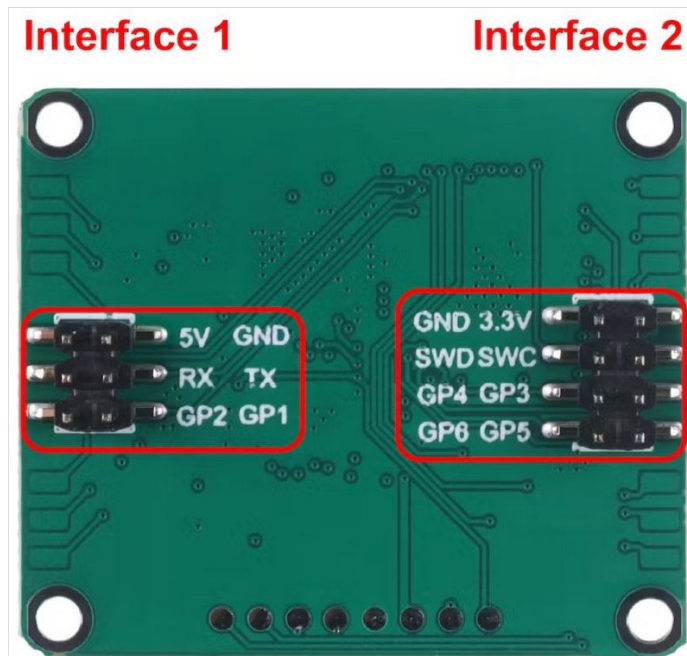
- **Non-contact sensing for comfort and privacy**
- **High-accuracy, self-regulating, privacy-protected, and secure**
- **Low power consumption for extended battery life**
- **Compact size and lightweight for easy integration**

Applications:

- **Consumer Electronics:** Smart home automation, wearable devices, sleep monitoring
- **Healthcare:** Patient monitoring, vital signs detection, illness and stress detection
- **Industrial Applications:** Presence detection, safety monitoring, automation control

Additional Features:

- **IR temperature sensor to measure human temperature**
- **Buzzer to alert when vital variations are found**
- **LCD display and mobile phone app for data visualization**



Step 4 - Enhanced Health and Wellness Monitoring:

The MR60BHA1 60GHz mmWave Module can be used to develop a comprehensive health and wellness monitoring system. By combining heart rate, respiration, and temperature measurements, the system can provide valuable insights into an individual's overall health and well-being.

Step 5 - Automated Device and Appliance Control:

The MR60BHA1 60GHz mmWave Module can also be used to create an automated device and appliance control system. By detecting the presence of a human in the operating area, the system can automatically turn on lights, adjust thermostats, and control other devices.

Step 6 - Privacy-Protected and Secure:

The MR60BHA1 60GHz mmWave Module operates in a completely private and secure environment, ensuring that personal health data is protected. The sensor does not require any physical contact with the user, and all data is processed locally on the device.

Interfacing the MR60BHA1 60GHz mmWave Module with the ESP32 requires establishing a communication link between the two devices. This can be achieved using a Serial interface(**Interface 1**).

For an implementation using the serial interface, refer to the following :

- Connect the MR60BHA1's **TX** pin to the ESP32's **RXD (16)** pin.
- Connect the MR60BHA1's **RX** pin to the ESP32's **TXD(17)** pin.
- Connect the MR60BHA1's **GND** pin to the ESP32's **GND** pin.
- Connect the MR60BHA1's **3.3V** pin to the ESP32's **3.3V** pin.

Interface 2 is typically used for firmware upgrades on the MR60BHA1 60GHz mmWave Module. Interface 1 is typically used for data communication between the module and a host device.

The following table summarizes the recommended use of each interface:

However, there may be some cases where it is not recommended to use Interface 2 for firmware upgrades. For example, if the module is already running the latest firmware version, then there is no need to upgrade it. Additionally, if the module is experiencing instability or other problems, then it may be best to avoid upgrading the firmware until the problems have been resolved.

In general, it is always a good idea to consult the user manual or other documentation for the MR60BHA1 60GHz mmWave Module before attempting to upgrade the firmware. This will help to ensure that the upgrade is performed safely and correctly. Please note that upgrading the firmware may void the warranty on the module. It is important to read and understand the warranty information before proceeding with any firmware upgrades.

For More Technical details on MR60BHA1 60GHz mmWave Module follow the link https://wiki.seeedstudio.com/Radar_MR60BHA1/

Interface	Recommended Use
Interface 1	Data communication between the module and a host device
Interface 2	Firmware upgrades

Step 7 - Radar module operating range

The radar module beam coverage is shown. The radar coverage is a three-dimensional sector of 80° horizontally and 80° tilted.

Due to the radar beam characteristics, the radar has a relatively long range in the direction normal to the antenna face, but a shorter range if it deviates from the antenna normal.

When the radar is mounted on top or at an angle, the radar beam range and the effective radiation space will reduce the radar range, which needs to be taken into account when using the radar.

Step 8 - Installation method

As the radar works mainly based on the respiratory heart rhythm causing undulating movements on the surface of the large muscles, the undulation of the human chest and back will be more pronounced, so this radar needs to be installed in the correct position to the human chest or back.

Based on the radar mode of action, the following mounting options are considered for radar installation.

(A) Overhead installation

The radar beam is positioned vertically downwards towards the body, with the center of the radar beam corresponding to the position of the body's chest cavity.

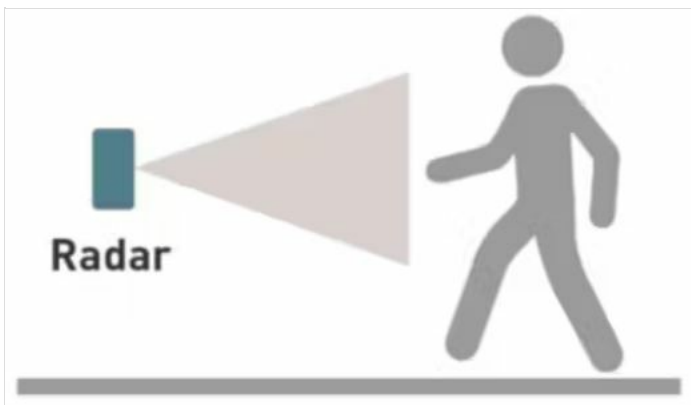
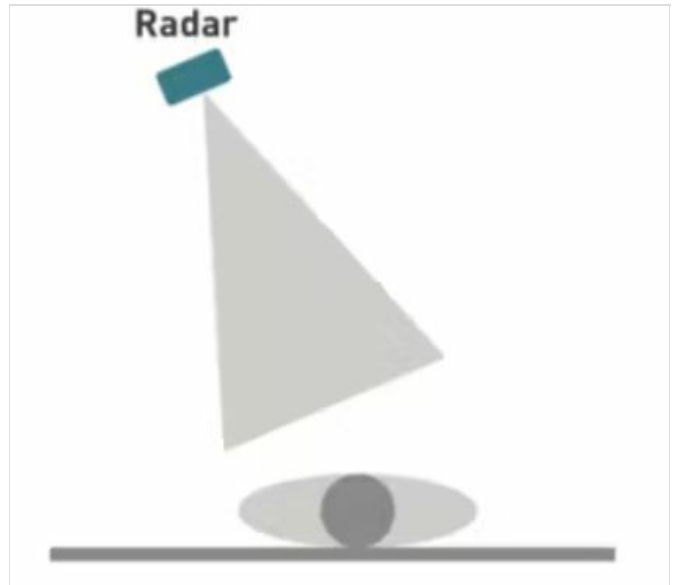
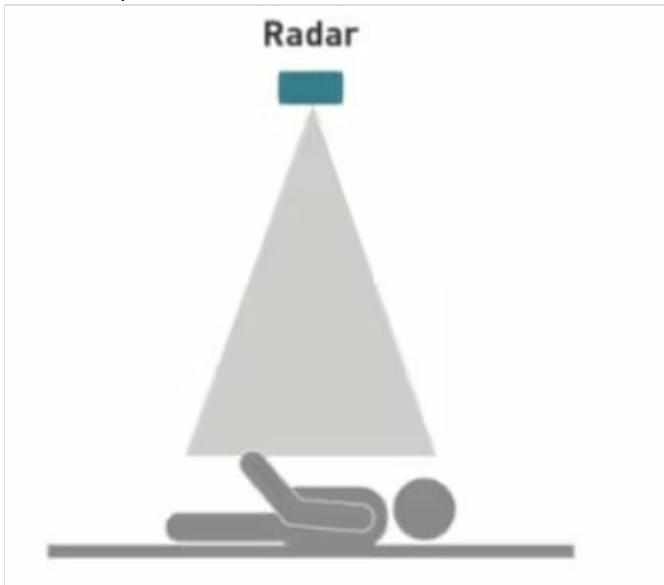
In this installation mode, a distance of ≤ 2 m is required between the radar and the body to be measured.

(B) Tilt mounting

With tilt mounting, the radar is fixed to a wall or bedside, the radar is mounted at an angle (as shown below) and the radar beam is directed at the body at an angle, with the center of the radar beam corresponding to the position of the human chest cavity. In this installation mode, the radial distance between the radar and the body to be measured is ≤ 2 m.

(C) Horizontal installation

The radar is placed horizontally (as shown in Fig Below), the radar is fixed to a wall, or placed on a table, the radar beam is directed toward the human body and the center of the radar beam corresponds to the position of the human thorax. The distance between the radar and the human body to be measured in this installation mode is ≤ 2 m.



Step 9 - How the Project Works

The project comprises an MR60BHA1 60GHz mmWave module for respiration, heart rate, and human presence detection, along with an MLX90614 sensor for contactless body temperature measurement. This project caters to two sectors: wellness monitoring and home or office automation.

The ESP32 microcontroller serves as the project's central processing unit (CPU) and WiFi module. The MR60BHA1 sensor is connected using the ESP32's UART channels and the MLX90614 sensor is connected using ESP32's I2C Channels(SDA, SCL).

To measure **body temperature**, the user needs to bring their hand near the sensor. A proximity sensor is deployed near the MLX90614 sensor to detect the presence of a hand and initialize the body temperature measurement. The measured temperature details will be displayed on the LCD panels and Blynk mobile application in Celsius units.

To initiate **heart rate** and **respiration rate** measurement, the user must first deactivate the human presence detection switch. Once deactivated, the user should press the measurement button, and the ESP32 will initiate the measurement process by emitting a buzzer alert. The algorithms require a few seconds to collect and process the data, so please expect some minor delays before the accurate readings are displayed on the LCD screen and the Blynk mobile application.

To activate the automation system based on human presence, the human presence detection switch must be activated. Once the switch is activated, the MR60BHA1 transitions into human presence detection mode. When human movements are detected within the radar's operating range, the ESP32 sends a pop-up notification to the Blynk app alerting the owner. Subsequently, an LED bulb will be turned on using a relay, demonstrating the MR60BHA1's capability in automating equipment. This human presence detection system also serves as a security product.

The entire project is powered by a 12-volt DC power adapter. Considering the power requirements of the sensors, microcontroller, and other components, the voltage is regulated to 5 volts using a 7805 voltage regulator and to 3.3 volts using an AMS1117 voltage regulator.

Step 10 - Project Code

```
#include "Arduino.h"
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <Adafruit_MLX90614.h>
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#define BLYNK_PRINT Serial
#define BLYNK_TEMPLATE_ID "TMPL3gy4Fq2LC"
#define BLYNK_TEMPLATE_NAME "Project"
#define BLYNK_AUTH_TOKEN "
char auth[] = "cxwnZjdeqqhWcGuObkEZ5OVt1WC_227g";
char ssid[] = "wifi";
char pass[] = "123456788";
Adafruit_MLX90614 mlx = Adafruit_MLX90614();
#define RXD2 16
#define TXD2 17
#include <60ghzbreathheart.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
BreathHeart_60GHz radar = BreathHeart_60GHz(&Serial2);
unsigned long delay_Intial = 0;
const long delay_interval = 1000;
unsigned long measure_Intial = 0;
const long measure_interval = 20000;
int Human_pesence_pin = 27;
int Human_presence;
int Heart_rate_measure_button_pin = 26;
int Heart_Button = 0;
int IR_Sensor_pin = 33;
int IR_Status;
int relay = 32;
int Buzzer = 25;
void _delay()
{
  delay_Intial = millis();
  while((millis() - delay_Intial) <= delay_interval)
  {
  }
}
```

```
void Measure_Breath_Heart_rate()
```

```

{
radar.Breath_Heart();      //Breath and heartbeat information output
if(radar.sensor_report != 0x00)
{
  switch(radar.sensor_report)
  {
    case HEARTRATEVAL:
      Serial.print("Sensor monitored the current heart rate value is: ");
      Serial.print(radar.heart_rate, DEC);
      Serial.println();
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("Heart Rate: ");
      lcd.print(radar.heart_rate);
      Blynk.virtualWrite(V0,radar.heart_rate);
      break;
    case BREATHVAL:
      Serial.print("Sensor monitored the current breath rate value is: ");
      Serial.print(radar.breath_rate, DEC);
      Serial.println();
      lcd.setCursor(0,1);
      lcd.print("Breath_rate: ");
      lcd.print(radar.breath_rate, DEC);
      Blynk.virtualWrite(V1,radar.breath_rate);
      break;
  }
}
}

void human_presence()
{
radar.HumanExis_Func();      //Human existence information output
if(radar.sensor_report != 0x00){
  switch(radar.sensor_report){
    case BODYVAL:
      Serial.print("The parameters of human body signs are: ");
      Serial.println(radar.bodysign_val, DEC);
      Serial.println("-----");
      if(radar.bodysign_val >=15)
      {
        Blynk.virtualWrite(V3,20);
        measure_Intial = millis();
        delay(1000);
        Blynk.virtualWrite(V3,0);
        while((millis() - measure_Intial) <= measure_interval)
        {
          digitalWrite(relay,HIGH);
          lcd.clear();
          lcd.setCursor(0,0);
          lcd.print("Presence Found");
          lcd.setCursor(0,1);
          lcd.print("Lights ON");
          _delay();
        }

        digitalWrite(relay,LOW);
        lcd.setCursor(0,1);
        lcd.print("Lights OFF");
        _delay();
      }
      break;
  }
}
}

void HR_BR()
{
  measure_Intial = millis();
  while((millis() - measure_Intial) <= measure_interval)
  {
    Measure_Breath_Heart_rate();
  }
}
}

```

```

void Measure_IR_Temperature()
{
  Serial.print("**CltObject = ");
  Serial.print(mlx.readObjectTempC());
  Serial.println("**C");
  lcd.clear();
  lcd.setCursor(0,1);
  lcd.print("Body Temp: ");
  lcd.print(mlx.readObjectTempC());
  Blynk.virtualWrite(V2,mlx.readObjectTempC());
  _delay();
}

void setup()
{
  Serial.begin(115200);//RX0, TX0 OF ESP32
  Serial2.begin(115200, SERIAL_8N1, RXD2, TXD2);//RX2, TX2 OF ESP32
  pinMode(Heart_rate_measure_button_pin,INPUT);
  pinMode(IR_Sensor_pin,INPUT);
  pinMode(Human_pesence_pin,INPUT);
  pinMode(relay,OUTPUT);
  pinMode(Buzzer,OUTPUT);
  Wire.begin();
  lcd.begin();
  lcd.backlight();
  mlx.begin();
  radar.ModeSelect_fuc(1);
  Blynk.begin(auth, ssid, pass);
}

void loop()
{
  Heart_Button = digitalRead(Heart_rate_measure_button_pin);
  IR_Status = digitalRead(IR_Sensor_pin);
  Human_presence = digitalRead(Human_pesence_pin);
  lcd.clear();
  digitalWrite(relay,LOW);
  digitalWrite(Buzzer,LOW);
  Blynk.virtualWrite(V0,0);
  Blynk.virtualWrite(V1,0);
  Blynk.virtualWrite(V2,0);
  Blynk.virtualWrite(V3,0);
  switch(IR_Status)
  {
    case 1:
      Measure_IR_Temperature();
      break;
    case 0:
      break;
      lcd.clear();
  }

  switch(Heart_Button)
  {
    case 1:
      digitalWrite(Buzzer,HIGH);
      _delay();
      digitalWrite(Buzzer,LOW);
      _delay();
      Serial.println("Heart_rate_monitoring_system_acitve");
      HR_BR();
      break;
  }

  switch(Human_presence)
  {
    case 1:
      Serial.println("Human_presence_detection_active");
      human_presence();
      break;
    case 0:
      digitalWrite(relay,LOW);
      break;
  }
}

```



```
Blynk.run();  
}
```
