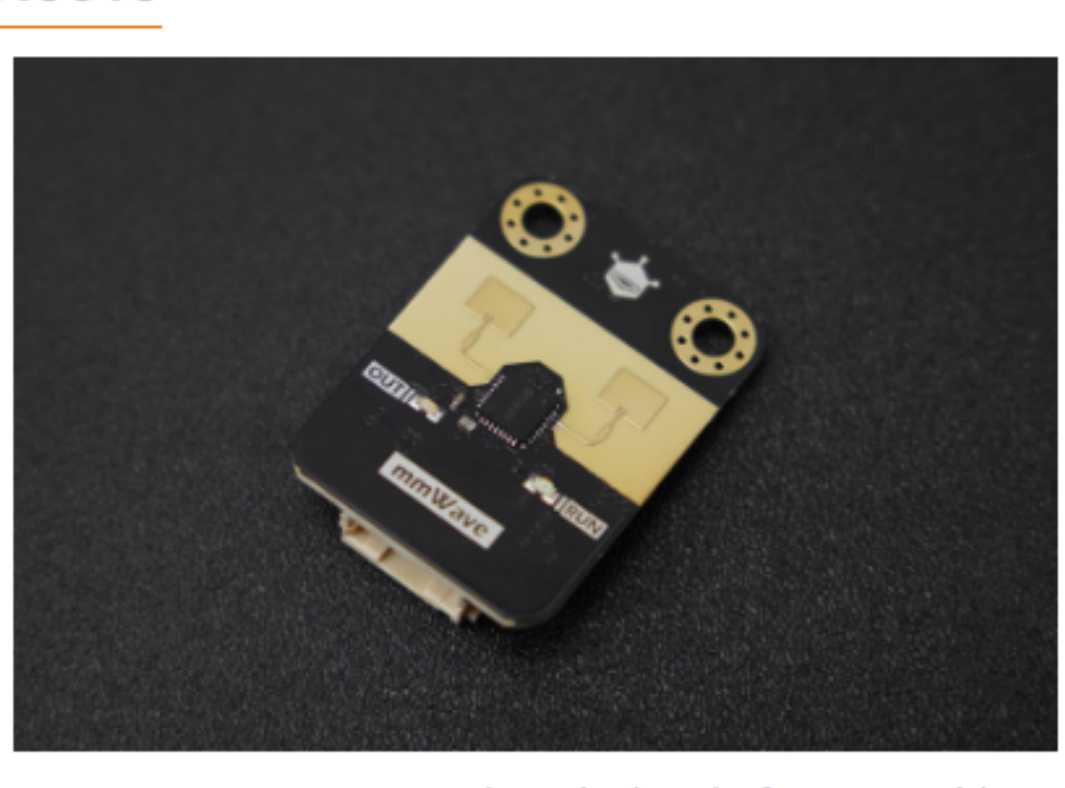
Introduction
Characteristics
Technical Specifications
Interface Definitions
Installation Method
Demo Program
FAQ
More Documents

SKU:SEN0610

Introduction



C4001 (12m) millimeter wave presence sensor uses a wavelength signal of 24GHz, with a detection range of 100° horizontally, presence detection range of 8 meters, and motion detection and ranging range of 12 meters.

Human detection

Compared to other types of human presence sensors, such as infrared sensors, the C4001 (12m) millimeter wave presence sensor has the advantage of detecting both static and moving objects. It also has a relatively strong anti-interference ability, making it less susceptible to factors such as temperature changes, variations in ambient light, and environmental noise. Whether a person is sitting, sleeping, or moving, the sensor can quickly and sensitively detect their presence.

	Millimeter-wave Presence Sensor	Infrared Sensor
Sensing Principle	TOF radar principle + Doppler radar sensing principle (active detection)	Pyroelectric infrared sensing principle (passive radiation)
Motion Sensitivity	Can detect presence, slight movement, and motion of human body	Can only detect motion and close-range slight movement of human body
Sensing Range	Can be adjusted to different sensing distances	Sensing range cannot be adjusted
Environmental Temperature Impact	Not affected by environmental temperature	Sensitivity decreases when temperature is close to human body temperature
Application Environment	Not affected by heat sources, light sources, air flow	Susceptible to heat sources and air flow
Penetration Ability	Can penetrate fabrics, plastics, glass, and other insulating materials	Can only penetrate some transparent plastics
Distance Measurement Support	Yes	No
Distance and velocity detection		

The C4001 (12m) millimeter wave presence sensor utilizes FMCW modulation for distance and speed measurement. It has a maximum detection range of 12 meters and a speed measurement range of 0.1 to 10 meters per second.

FMCW (Frequency Modulated Continuous Wave) is a type of radar system based on frequency modulation of continuous wave signals. Unlike traditional pulse radar, FMCW radar continuously emits a series of continuous wave signals with gradually changing frequencies while simultaneously receiving the reflected signals. By analyzing the received signals, measurements of parameters such as distance, speed, and angle can be achieved.

Compared to traditional pulse radar technology, FMCW radar can continuously measure the distance of an object. By utilizing the Doppler effect, it can also obtain information about the velocity of the target object. This makes FMCW radar suitable for applications that require tracking the movement of objects. Additionally, FMCW radar can achieve continuous frequency scanning, providing higher measurement resolution. Since it does not require waiting for the return of echo signals, FMCW radar is suitable for applications that require real-time monitoring and tracking of target objects.

What is a millimeter-wave radar sensor?

Millimeter-wave radar technology is a non-contact sensing technology used to detect objects and provide information about their distance, velocity, and angle (in the case of humans, for example). The signals emitted by millimeter-wave sensors fall within the high-frequency spectrum with wavelengths between 24GHz and 300GHz, also known as the millimeter (mm) range.

Characteristics

- Communication Methods: It supports two communication methods, I2C and UART.
- Interface: It uses the Gravity interface (PH2.0).
- Human Detection: It can detect the presence of humans up to 8 meters and detect human motion up to 12 meters.
- Distance Detection: It can measure distances from 1.2m to 12m.
- Speed Detection: It can detect speeds from 0.1m/s to 10m/s.
- Strong anti-interference capability: It is not affected by factors such as snow, haze, temperature, humidity, dust, light, and noise.
- Compact size and easy integration.

Technical Specifications

- Operating Voltage: 3.3V/5V
- Maximum Detection Range: 12m
- Beam Angle: 100*80°
- Modulation Mode: FMCW
- Operating Frequency: 24GHz
- Operating Temperature: -40~85℃
- Baud Rate: 9600
- I2C Address: 0x2A/0x2B
- Size: 22*30mm

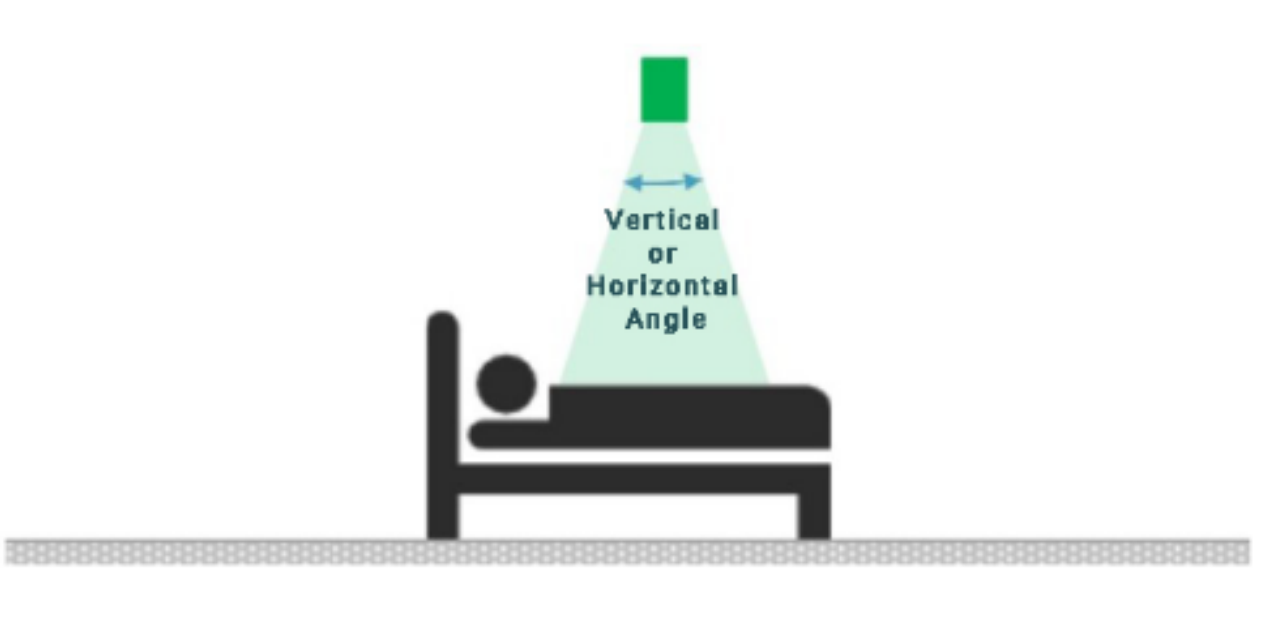
Interface Definitions

Definition	Explanation
+	Power
-	Ground
C/R	I2C Clock Line / RX
D/T	I2C Data Line / TX

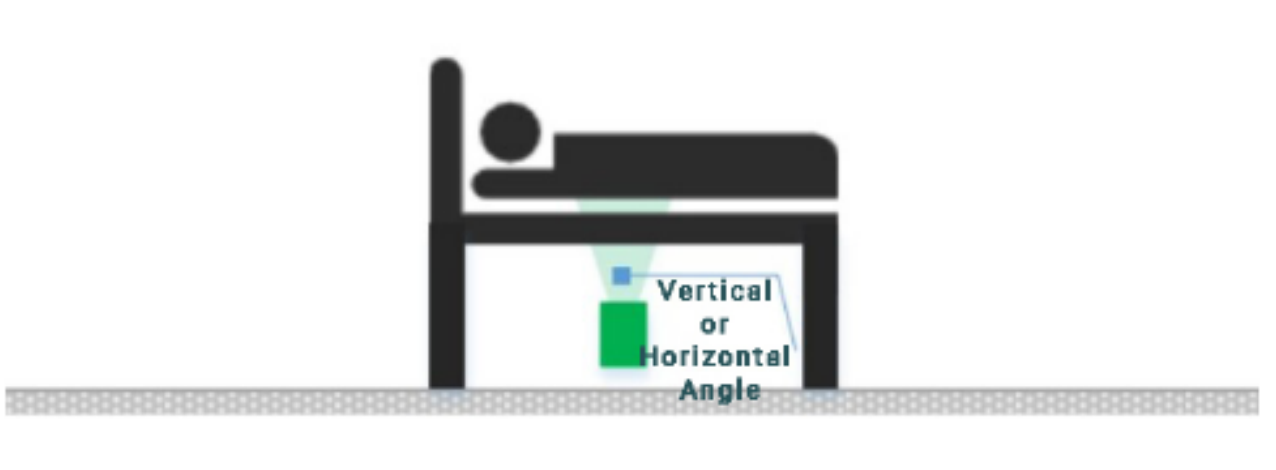
Installation Method

The millimeter-wave human body sensor is sensitive to the installation method, and improper installation can affect the performance and functionality of the sensor. The commonly used installation methods for this module include top installation, bottom installation, horizontal installation, and downward tilted installation.

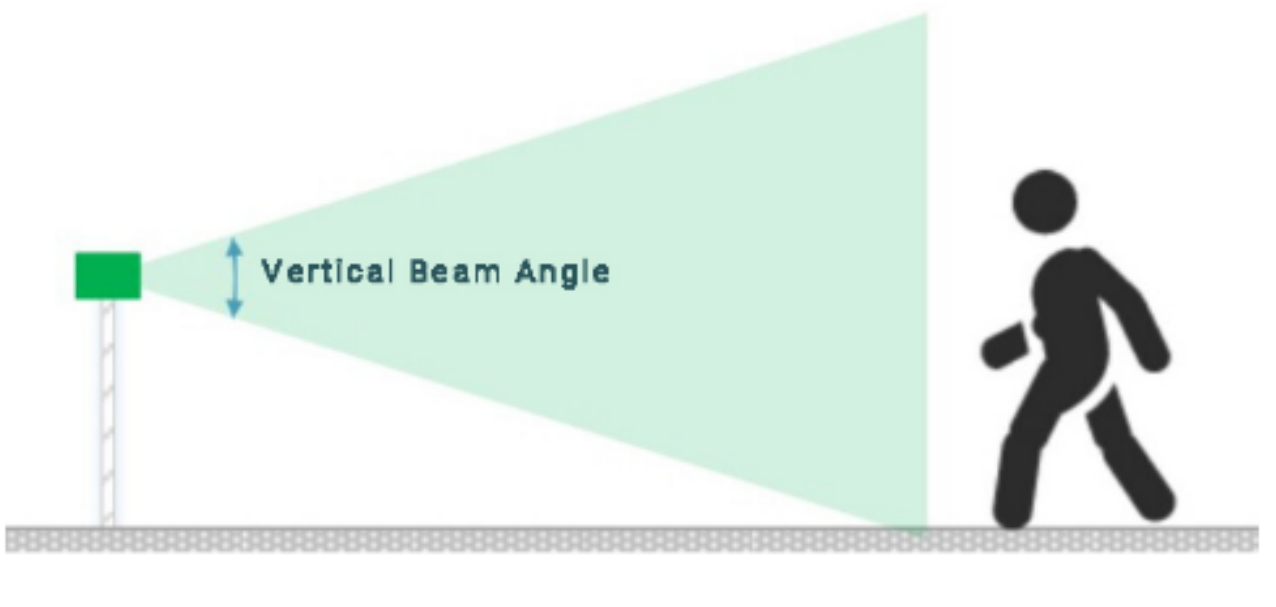
Top Installation



Bottom Installation



Horizontal Installation



Demo Program

Information Retrieval in UART Communication

Preparation

- Hardware
  - C4001 Millimeter Wave Presence Sensor 12m
  - Arduino Uno
- Software

  Arduino IDE，Click to download Arduino IDE

  DFRobot_C4001 library，Click to download the DFRobot_C4001 library.

Wiring diagram

mmWave	Arduino Uno
VIN	5V
GND	GND
C/R	D5
D/T	D4

Sample code

Please switch the DIP switch on the back of the sensor to the UART direction, and copy the following code to your Arduino IDE and upload it.

```
 /*!
  * @file  motionDetection.ino
  * @brief  Example of radar detecting whether an object is mo
  * @copyright Copyright (c) 2010 DFRobot Co.Ltd (http://www.d
  * @license The MIT License (MIT)
  * @author ZhixinLiu(zhixin.liu@dfrobot.com)
  * @version V1.0
  * @date 2024-02-02
  * @url https://github.com/dfrobot/DFRobot_C4001
  */

#include "DFRobot_C4001.h"

//#define I2C_COMMUNICATION  //use I2C for communication, but

#ifdef  I2C_COMMUNICATION
  /*
   * DEVICE_ADDR_0 = 0x2A     default iic_address
   * DEVICE_ADDR_1 = 0x2B
   */
  DFRobot_C4001_I2C radar(&Wire ,DEVICE_ADDR_0);
#else
/* ----------------------------------------------------------------
 *    board   |             MCU                | Leonardo/Mega
 *     VCC    |            3.3V/5V             |        VCC
 *     GND    |              GND               |        GND
 *     RX     |              TX                |     Serial1 T
 *     TX     |              RX                |     Serial1 R
 * ----------------------------------------------------------------
/* Baud rate cannot be changed */
  #if defined(ARDUINO_AVR_UNO) || defined(ESP8266)
    SoftwareSerial mySerial(4, 5);
    DFRobot_C4001_UART radar(&mySerial ,9600);
  #elif defined(ESP32)
    DFRobot_C4001_UART radar(&Serial1 ,9600 ,/*rx*/D2 ,/*tx*/D
  #else
    DFRobot_C4001_UART radar(&Serial1 ,9600);
  #endif
#endif

void setup()
{
  Serial.begin(115200);
  while(!Serial);
  while(!radar.begin()){
    Serial.println("NO Deivces !");
    delay(1000);
  }
  Serial.println("Device connected!");

  // exist Mode
  radar.setSensorMode(eExitMode);

  sSensorStatus_t data;
  data = radar.getStatus();
  //  0 stop  1 start
  Serial.print("work status  = ");
  Serial.println(data.workStatus);

  //  0 is exist   1 speed
  Serial.print("work mode  = ");
  Serial.println(data.workMode);

  //  0 no init    1 init success
  Serial.print("init status = ");
  Serial.println(data.initStatus);
  Serial.println();

  /*
   * min Detection range Minimum distance, unit cm, range 0.3~
   * max Detection range Maximum distance, unit cm, range 2.4~
   * trig Detection range Maximum distance, unit cm, default t
   */
  if(radar.setDetectionRange(/*min*/30, /*max*/1000, /*trig*/1
    Serial.println("set detection range successfully!");
  }
  // set trigger sensitivity 0 - 9
  if(radar.setTrigSensitivity(1)){
    Serial.println("set trig sensitivity successfully!");
  }

  // set keep sensitivity 0 - 9
  if(radar.setKeepSensitivity(2)){
    Serial.println("set keep sensitivity successfully!");
  }
  /*
   * trig Trigger delay, unit 0.01s, range 0~2s (0~200)
   * keep Maintain the detection timeout, unit 0.5s, range 2~1
   */
  if(radar.setDelay(/*trig*/100, /*keep*/4)){
    Serial.println("set delay successfully!");
  }


  // get confige params
  Serial.print("trig sensitivity = ");
  Serial.println(radar.getTrigSensitivity());
  Serial.print("keep sensitivity = ");
  Serial.println(radar.getKeepSensitivity());

  Serial.print("min range = ");
  Serial.println(radar.getMinRange());
  Serial.print("max range = ");
  Serial.println(radar.getMaxRange());
  Serial.print("trig range = ");
  Serial.println(radar.getTrigRange());

  Serial.print("keep time = ");
  Serial.println(radar.getKeepTimerout());

  Serial.print("trig delay = ");
  Serial.println(radar.getTrigDelay());

}

void loop()
{
  // Determine whether the object is moving
  if(radar.motionDetection()){
    Serial.println("exist motion");
    Serial.println();
  }
  delay(100);
}
```

Result

例程1结果

IIC communication distance speed acquisition

Preparation

- Hardware
  - C4001 Millimeter Wave Presence Sensor 12m
  - Arduino Uno
- Software

  Arduino IDE，Click to download Arduino IDE

  DFRobot_C4001 library，Click to download the DFRobot_C4001 library.

Wiring diagram

Sample code

Copy the following code and upload it to your Arduino IDE。

```
/*!
  * @file  mRangeVelocity.ino
  * @brief  radar measurement demo
  * @copyright Copyright (c) 2010 DFRobot Co.Ltd (http://www.d
  * @license The MIT License (MIT)
  * @author ZhixinLiu(zhixin.liu@dfrobot.com)
  * @version V1.0
  * @date 2024-02-02
  * @url https://github.com/dfrobot/DFRobot_C4001
  */

#include "DFRobot_C4001.h"

#define I2C_COMMUNICATION  //use I2C for communication, but us

#ifdef I2C_COMMUNICATION
/*
   * DEVICE_ADDR_0 = 0x2A     default iic_address
   * DEVICE_ADDR_1 = 0x2B
   */
DFRobot_C4001_I2C radar(&Wire, DEVICE_ADDR_0);
#else
/* ----------------------------------------------------------------
 *    board   |             MCU                | Leonardo/Mega
 *     VCC    |            3.3V/5V             |        VCC
 *     GND    |              GND               |        GND
 *     RX     |              TX                |     Serial1 T
 *     TX     |              RX                |     Serial1 R
 * ----------------------------------------------------------------
/* Baud rate cannot be changed */
#if defined(ARDUINO_AVR_UNO) || defined(ESP8266)
SoftwareSerial mySerial(4, 5);
DFRobot_C4001_UART radar(&mySerial, 9600);
#elif defined(ESP32)
DFRobot_C4001_UART radar(&Serial1, 9600, /*rx*/ D2, /*tx*/ D3)
#else
DFRobot_C4001_UART radar(&Serial1, 9600);
#endif
#endif

void setup() {
  Serial.begin(115200);
  while (!Serial)
    ;
  while (!radar.begin()) {
    Serial.println("NO Deivces !");
    delay(1000);
  }
  Serial.println("Device connected!");

  // speed Mode
  radar.setSensorMode(eSpeedMode);

  sSensorStatus_t data;
  data = radar.getStatus();
  //  0 stop  1 start
  Serial.print("work status  = ");
  Serial.println(data.workStatus);

  //  0 is exist   1 speed
  Serial.print("work mode  = ");
  Serial.println(data.workMode);

  //  0 no init    1 init success
  Serial.print("init status = ");
  Serial.println(data.initStatus);
  Serial.println();

  /*
   * min Detection range Minimum distance, unit cm, range 0.3~
   * max Detection range Maximum distance, unit cm, range 2.4~
   * thres Target detection threshold, dimensionless unit 0.1,
   */
  if (radar.setDetectThres(/*min*/ 11, /*max*/ 1200, /*thres*/
    Serial.println("set detect threshold successfully");
  }

  // set Fretting Detection
  radar.setFrettingDetection(eON);

  // get confige params
  Serial.print("min range = ");
  Serial.println(radar.getTMinRange());
  Serial.print("max range = ");
  Serial.println(radar.getTMaxRange());
  Serial.print("threshold range = ");
  Serial.println(radar.getThresRange());
  Serial.print("fretting detection = ");
  Serial.println(radar.getFrettingDetection());
}

void loop() {
  Serial.print("target number = ");
  Serial.println(radar.getTargetNumber());  // must exist
  Serial.print("target Speed  = ");
  Serial.print(radar.getTargetSpeed());
  Serial.println(" m/s");

  Serial.print("target range  = ");
  Serial.print(radar.getTargetRange());
  Serial.println(" m");

  Serial.print("target energy  = ");
  Serial.println(radar.getTargetEnergy());
  Serial.println();
  delay(100);
}
```

Result

例程1结果

result

例程1结果

FAQ

If you have any questions about using this product, please check theComprehensive Guide to Smart Home Presence Detection (FAQ)for that product for a corresponding solution.

And for any questions, advice or cool ideas to share, please visit the DFRobot Forum.

More Documents