**Reference Scheme for Multiple TFmini-I2C Used in Arduino**

Version：v1.0

Update date：2019.04.28

Author：Mincy Zhao

（zhaomingxin@benewake）

Directory

[1 Introduction **错误!未定义书签。**](#_Toc21141)

[2 Tools **错误!未定义书签。**](#_Toc2897)

[3 Connection Testing **错误!未定义书签。**](#_Toc14531)

[3.1 Setting TFmin-I2C address **错误!未定义书签。**](#_Toc43)

[3.2 Connection **错误!未定义书签。**](#_Toc30376)

[3.3 Code **错误!未定义书签。**](#_Toc14658)

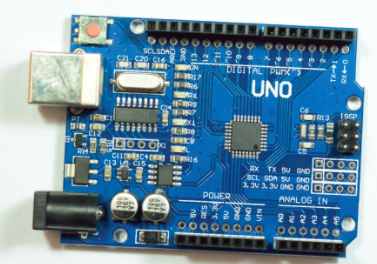
[4 Test Effect **错误!未定义书签。**](#_Toc6835)

# Introduction

In this scheme, four TFmini-I2C radars are used to receive data through a set of I2C buses on the Arduino UNO development board.

I2C communication adopts master-slave mode, Arduino development board as master, TFmini-I2C as slave. Each TFmini is configured as a separate address, and the host queries the data corresponding to the TFmini address and waits for the data to be returned.

# Tools

1. Four TFmini-I²C radars
2. Arduino UNO
3. 5VDC/2A source
4. DuPont Line

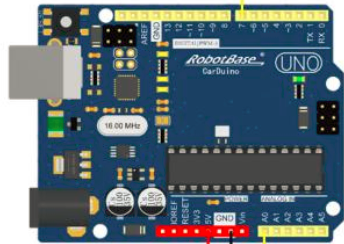
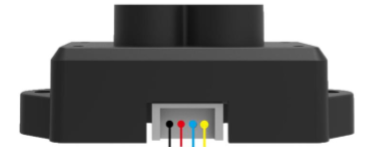
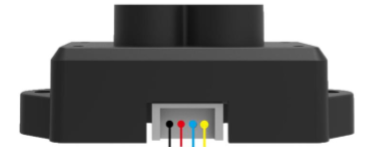
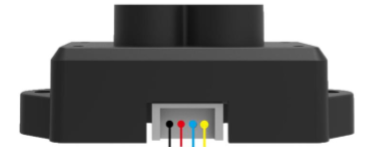
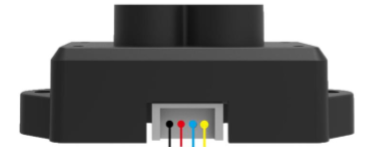
# Connection Testing

## Setting TFmin-I2C address

Setting method can refer to "Product Manual". In this scheme, the slave addresses of four TFminis are set to 0x10, 0x11, 0x12 and 0x13 respectively.

## Connection

Note: Because the output current of computer USB interface is limited, it needs external 5V power supply to connect multiple TFminis at the same time. At the same time, radar and Arduino development board need to be common-ground.



SDA

SCL

5VDC

## Code

/\*This is a reference code about Arduino UNO receive TFmini-I²C Data from I²C bus

\* Arduino is Master, TFmini-I²C is slave. Master send check

\* Author:zhaomingxin@Benewake

\* Update date:2019.4.28

\*/

#include <Wire.h>

void setup() {

Wire.begin(); // join i2c bus (address optional for master)

Serial.begin(115200); // start serial for output

pinMode(LED\_BUILTIN, OUTPUT);//LED

}

void loop() {

Get\_LidarDatafromIIC(0x10);

Get\_LidarDatafromIIC(0x11);

Get\_LidarDatafromIIC(0x12);

Get\_LidarDatafromIIC(0x13);

delay(250);

digitalWrite(LED\_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)

delay(250);

digitalWrite(LED\_BUILTIN, LOW); // turn the LED off by making the voltage LOW

}

void Get\_LidarDatafromIIC(byte address){

char i = 0;

byte rx\_buf[7] = {0};

Wire.beginTransmission(address); // Begin a transmission to the I2C Slave device with the given address.

Wire.write(1); // Reg's Address\_H

Wire.write(2); // Reg's Address\_L

Wire.write(7); // Data Length

Wire.endTransmission(0); // Send a START Sign

Wire.requestFrom(address, 7); // request 7 bytes from slave device address

//print the result via serial

Serial.print("Address=0x");

Serial.print(address,HEX);

Serial.print(": ");

while ( Wire.available())

{

rx\_buf[i] = Wire.read(); // received one byte

Serial.print("0x");

Serial.print(rx\_buf[i],HEX);

Serial.print(";");

i++;

if(i>=7)

{

i=0;

Serial.print("---------->");

Serial.print("Distance=");

Serial.print(rx\_buf[3]\*256+rx\_buf[2]);

Serial.print(";");

Serial.print("Strength=");

Serial.print(rx\_buf[5]\*256+rx\_buf[4]);

}

}

Serial.print("\r\n");

}

# Results

Arduino queries radar data through I2C bus and prints the data through serial port.

