STM32 drives RD-03D radar module to light up different LED lights at different areas

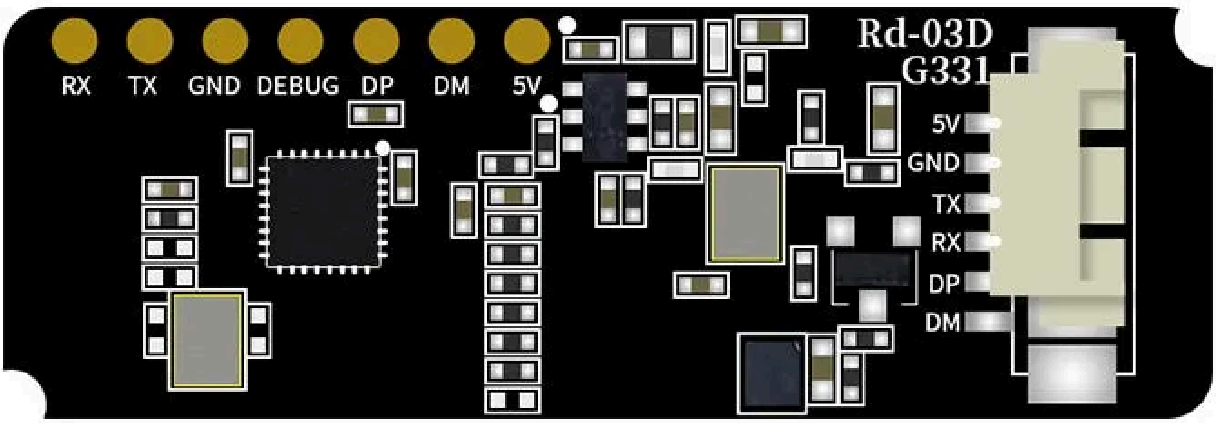
Tara Qiu(Ai-Thinker) · Follow

5 min read · Sep 25, 2024

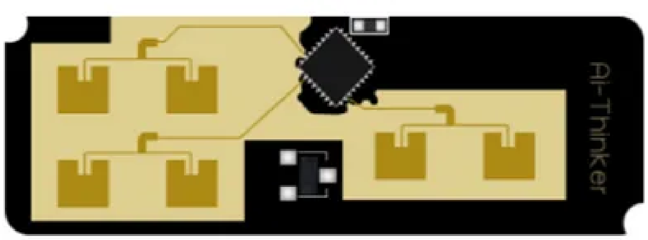
Ai-Thinker radar module Rd-03D is a module use 1T2R antenna to achieve target tracking and achieve distance, angle and speed measurement of targets in the area. This application example uses STM32 to parse the serial port data of Rd-03D and send the results to the VC-02 development board through another serial port to realize real-time broadcast of the current number of people.

Pin definition

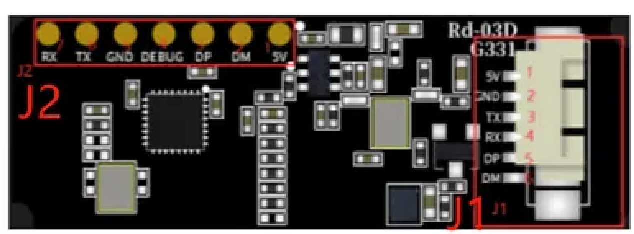
Rd-03D module is connected to a total of 4 pins, such as pin schematic pin function definition table interface definition



No.	Name	Function
1	5V	Input power
2	GND	Ground
3	TX	UART_TX
4	RX	UART_RX



(a) 正面



(b) 反面

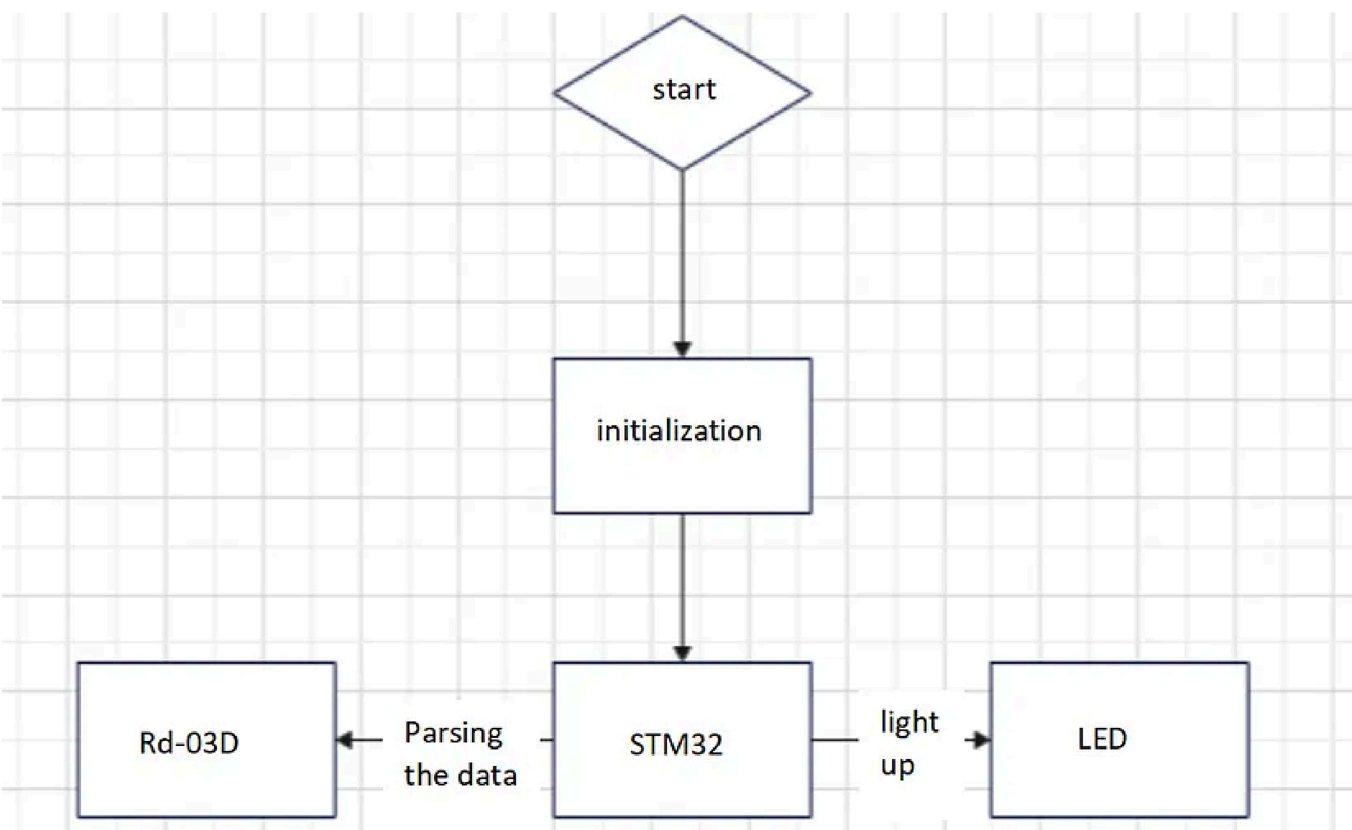
J1 pins description：

J#PIN#	Name	Function	Description
J1PIN1	5V	Power input	Connect to serial port convert board **5V**
J1PIN2	GND	Connect to ground	Connect to serial port convert board **GND**
J1PIN3	TX	UART_TX	Connect to serial port convert board **RXD**
J1PIN4	RX	UART_RX	Connect to serial port convert board **TXD**
J1PIN5	DP	Flash data positive signal	If a 4-pin connector is used, this pin is not used.
J1PIN6	DM	Flash data negative signal	If a 4-pin connector is used, this pin is not used.

J2 pin description

J#PIN#	Name	Function	Description
J2PIN1	5V	Power input	5V
J2PIN2	DM	Flash data negative signal	-
J2PIN3	DP	Flash data positive signal	-
J2PIN4	DEBUG	Debug serial port TXD	For debugging firmware

Software Design Framework

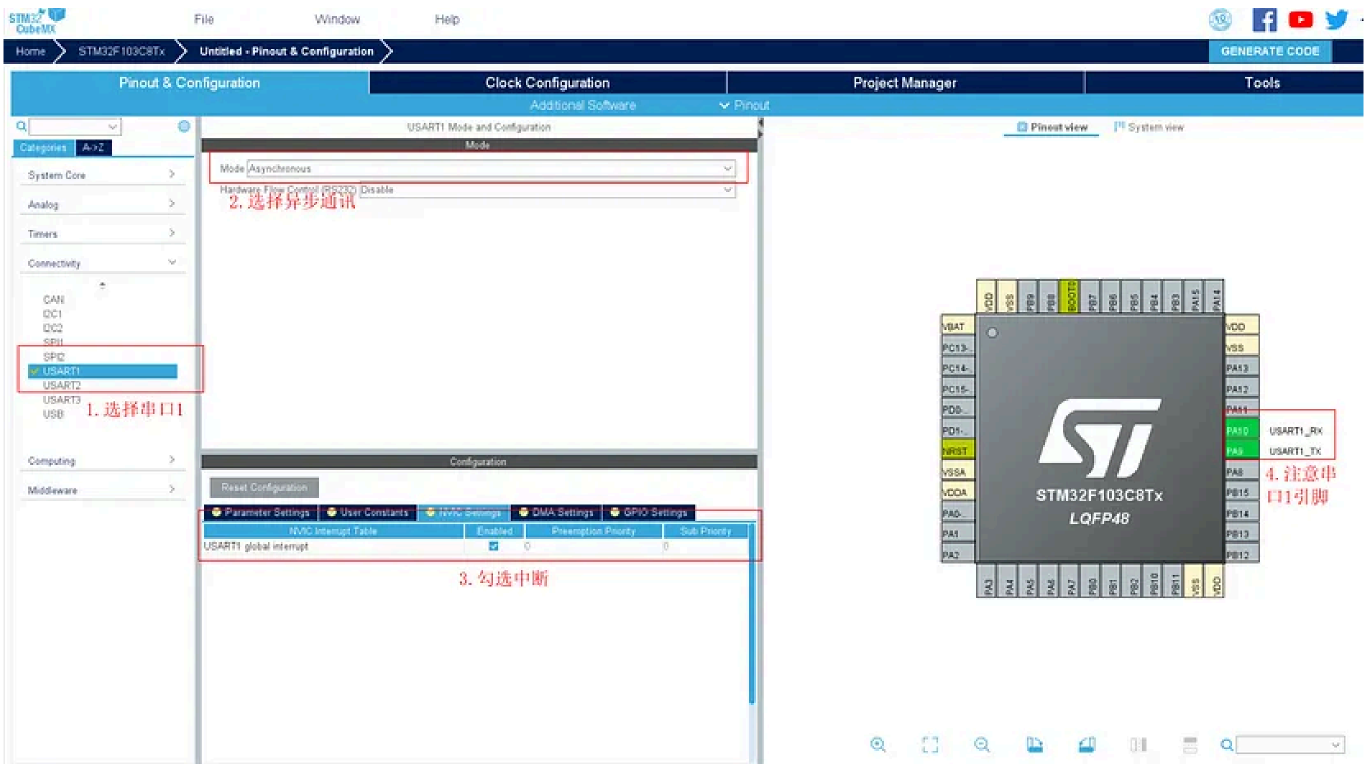


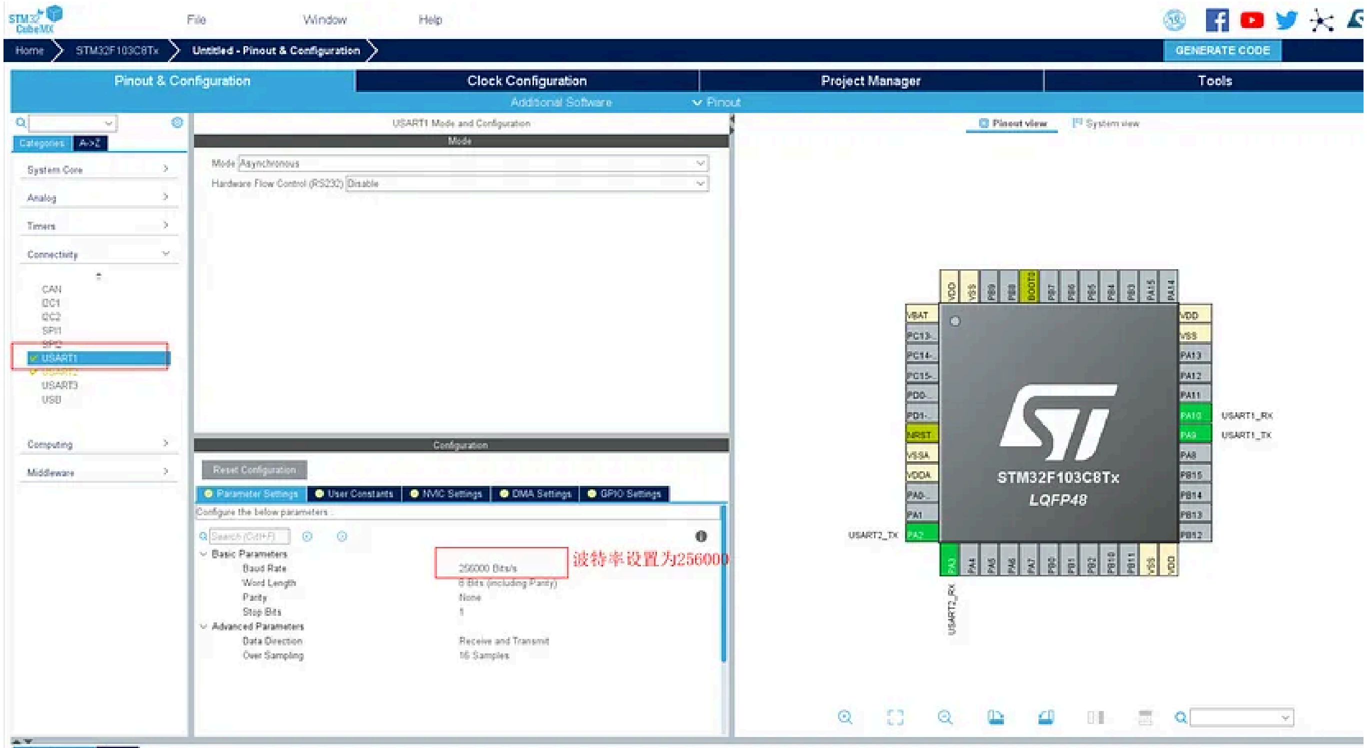
STM32F103C8T6 uses CubeMX with HAL library configuration

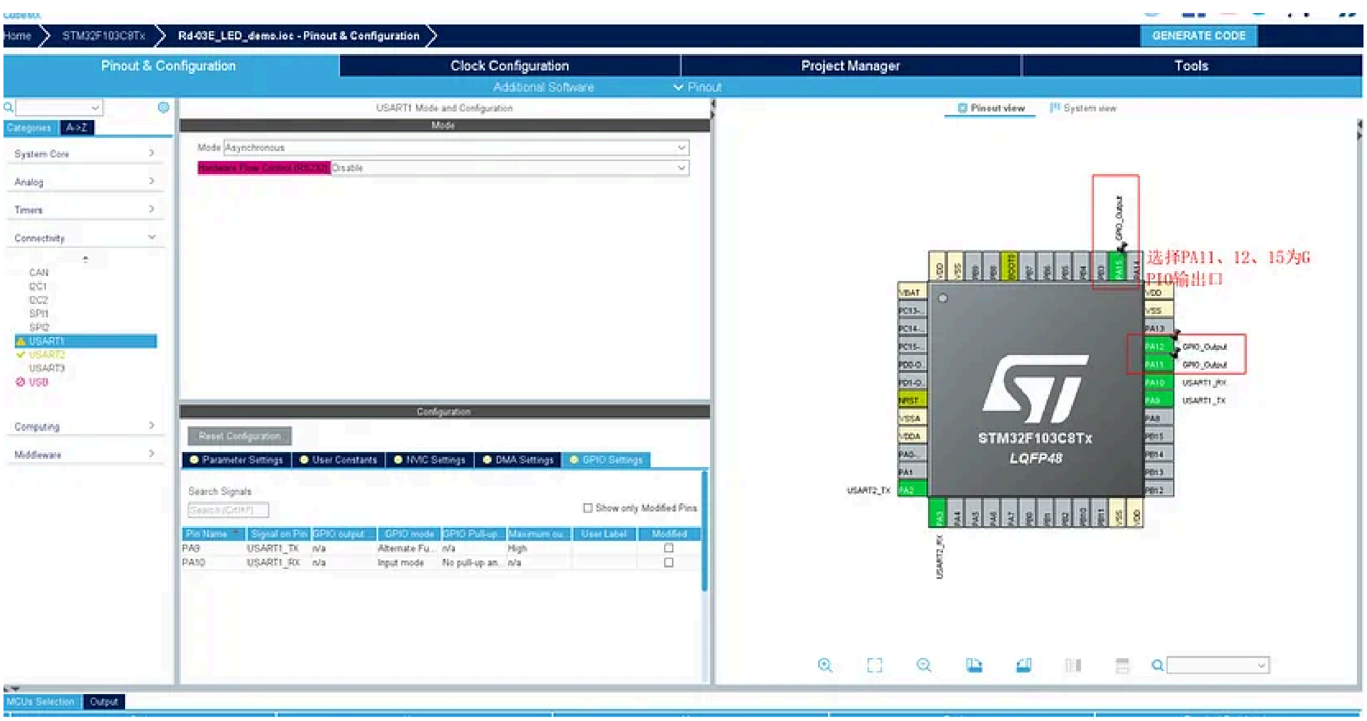
Open CubeMX and select STM32F103C8T6.

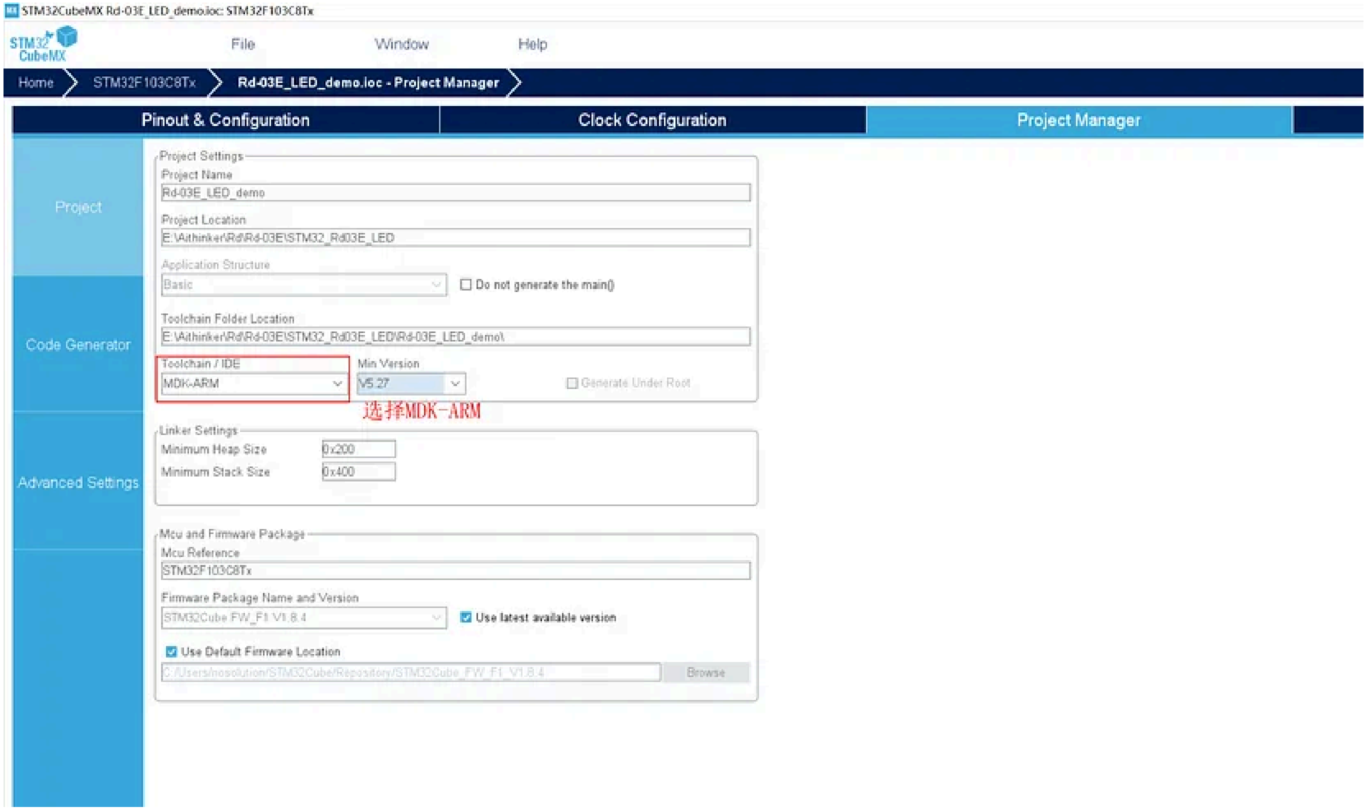
Select two serial ports, serial port 1 and serial port 2, PA9 is USART1_TX, PA10 is USART1_RX, PA2 is USART2_TX, and PA3 is USART_RX.

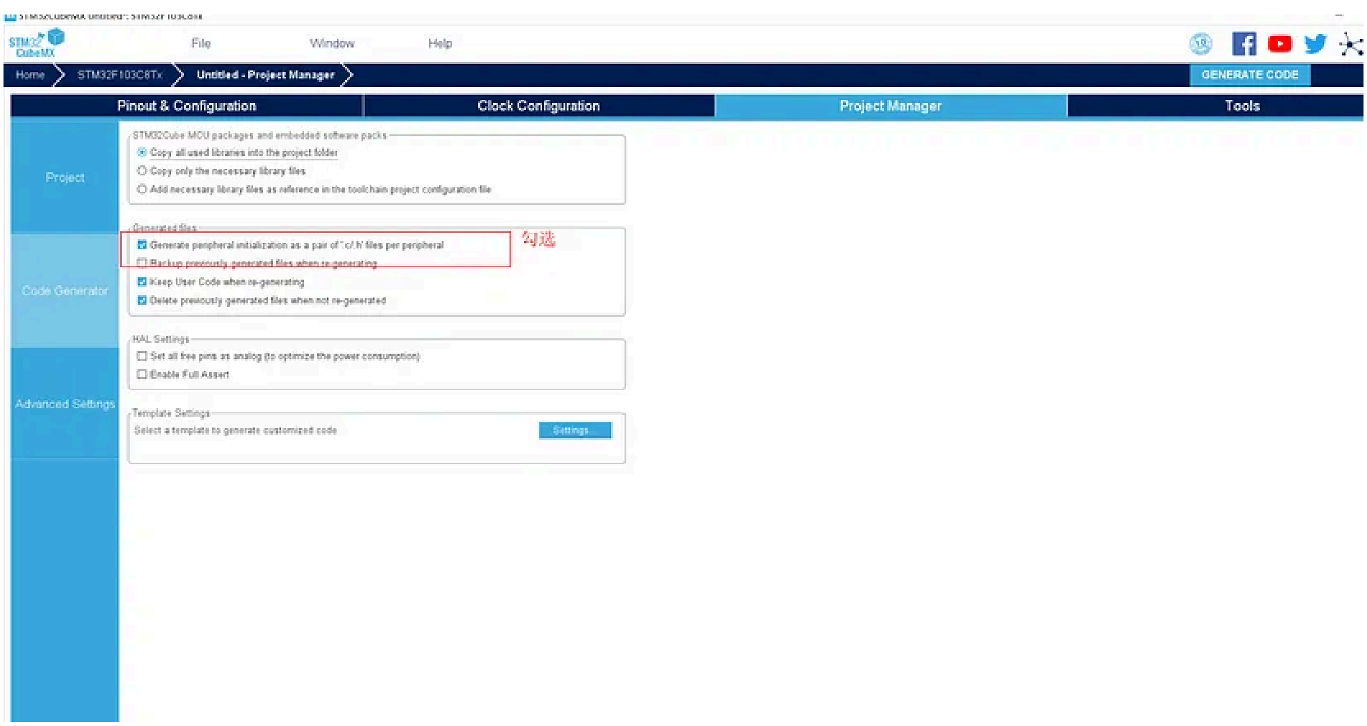
Select asynchronous communication, check interrupt, note: the baud rate of serial port 1 is 256000, and the baud rate of serial port 2 is 115200.




File
Window
Help
Home
STM32F103C8Tx
Untitled - Pinout & Configuration
GENERATE CODE
Pinout & Configuration
Clock Configuration
Project Manager
Tools
Additional Software
Pinout
USART1 Mode and Configuration
Mode
Mode Asynchronous
Hardware Flow Control (RS232) Disable
Configuration
Reset Configuration
Parameter Settings
User Constants
NVIC Settings
DMA Settings
GPIO Settings
Basic Parameters
Baud Rate 256000 Bits/s
Word Length 8 Bits (including Parity)
Parity None
Stop Bits 1
Advanced Parameters
Data Direction Receive and Transmit
Over Sampling 16 Samples
波特率设置为256000
System Core
Analog
Timers
Connectivity
CAN
I2C1
I2C2
SPI1
SPI2
USART1
USART3
USB
Computing
Middleware
Pinout view
System view
STM32F103C8Tx
LQFP48
USART1_RX
USART1_TX
USART2_TX
USART2_RX


Home
STM32F103C8Tx
Rd-03E_LED_demo.ioc - Pinout & Configuration
GENERATE CODE
Pinout & Configuration
Clock Configuration
Project Manager
Tools
Additional Software
Pinout
USART1 Mode and Configuration
Mode
Mode Asynchronous
Hardware Flow Control (RS232) Disable
Configuration
Reset Configuration
Parameter Settings
User Constants
NVIC Settings
DMA Settings
GPIO Settings
Search Signals
Show only Modified Pins
PA9 USART1_TX n/a Alternate Fu... n/a High
PA10 USART1_RX n/a Input mode No pull-up an... n/a
选择PA11、12、15为GPIO输出口
GPIO_Output
GPIO_Output
GPIO_Output
USART1_RX
USART1_TX
USART2_TX
USART2_RX
STM32F103C8Tx
LQFP48
MCUs Selection
Output





Wiring of STM32, Rd-03D and LED light

Rd-03D		STM32		LED
5V	——	5V	——	
GND	——	GND	——	LED1\2\3 negative electrode
TX	——	PA10		
RX	——	PA9		
		PA11	——	LED1positive electrode
		PA12	——	LED2 positive electrode
		PA15	——	LED2positive electrode

Serial port data processing

Rd-03D serial port data

The Rd-03D module communicates with the outside through the serial port (TTL level). The default baud rate of the radar serial port is 256000, 1 stop bit, and no parity bit. The radar outputs the detected target information, including the x-coordinate, y-coordinate, and speed value of the target in the area (little end mode).

The report is in data frame format:

Frame Header	Intra-frame data	Frame end
AA FF 03 00	Target 1 information,Target 2 information,Target 3 information	55 CC

Data example:

AA FF 03 00 0E 03 B1 86 10 00 68 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 CC

The red part indicates the information of target 1, the blue part indicates the information of target 2, and the green part indicates the information of target 3.

This example shows the analysis of angle information in the single target mode, so the data in a single target needs to be analyzed (i.e. target 1), where the specific contents of a single target are as follows:

Target x-coordinate	Target y coordinate	Target speed	Pixel distance value
signed int16 type; the highest bit 1 corresponds to positive coordinates, 0 corresponds to negative coordinates; the remaining 15 bits represent the absolute value of the x coordinate, unit mm	signed int16 type; the highest bit 1 corresponds to positive coordinates, 0 corresponds to negative coordinates; the remaining 15 bits represent the absolute value of the y coordinate, unit mm	signed int16 type; the highest bit 1 corresponds to positive speed, 0 corresponds to negative speed; the remaining 15 bits represent the absolute value of speed, unit cm/s	Uint16 type; single pixel distance value, unit: mm.

According to the target X, Y coordinate data frame description, if the above data example is used, the module converts the angle data of target 1 into the relevant coordinate information as shown below:

Target 1x coordinate: *OxOE+Ox03*256= 782*

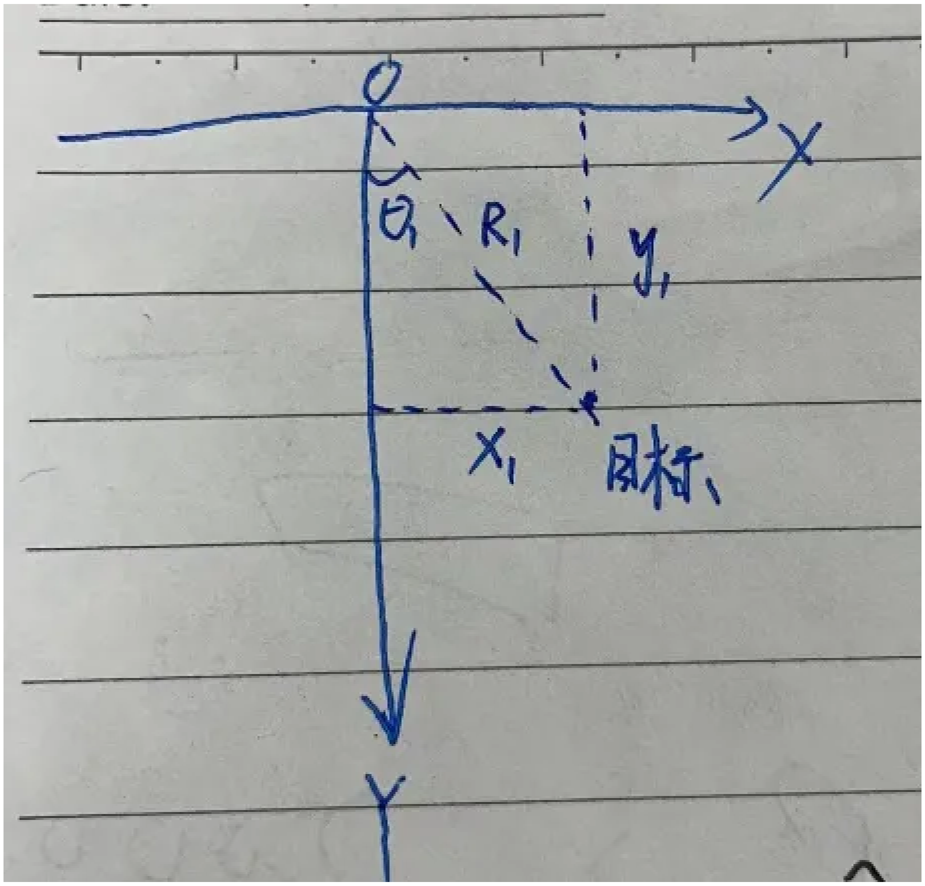
0–782= -782 mm;

Target 1y coordinate: *OxB1+Ox86*256 = 34481*

34481–2^{15}= 1713 mm;

STM32 data processing

The angle calculation process uses trigonometric function knowledge to find the tan(θ) angle of the opposite side relative to the adjacent side (i.e. x/y), as shown in the figure:



Specific implementation code:

```
#define LED_ON_2 HAL_GPIO_WritePin(GPIOA, GPIO_PIN_11, GPIO_PIN_SET);
#define LED_OFF_2 HAL_GPIO_WritePin(GPIOA, GPIO_PIN_11, GPIO_PIN_RESET);
#define LED_ON_4 HAL_GPIO_WritePin(GPIOA, GPIO_PIN_12, GPIO_PIN_SET);
#define LED_OFF_4 HAL_GPIO_WritePin(GPIOA, GPIO_PIN_12, GPIO_PIN_RESET);
#define LED_ON_6 HAL_GPIO_WritePin(GPIOA, GPIO_PIN_15, GPIO_PIN_SET);
#define LED_OFF_6 HAL_GPIO_WritePin(GPIOA, GPIO_PIN_15, GPIO_PIN_RESET);

#define PI 3.14159265
uint8_t RX_BUF[64]={0};  //Cache array
uint8_t RX_count=0;      //Counting position
uint8_t RX_temp;         //Cache Characters
uint16_t range;          //Sensing distance
int16_t x_pos;           //x-axis coordinate
int16_t y_pos;          //y-axis coordinate
double angle;          //angle

void HAL_UART_RxCpltCallback(UART_HandleTypeDef *huart)//Callback function, whic
 if(huart == &huart1){   //Serial port 1 triggers interrupt
   RX_BUF[RX_count++] = RX_temp;//Store the buffered characters in the buffer ar
   if((RX_BUF[RX_count-1] == 0xCC)&&(RX_BUF[RX_count-2] == 0x55)){//Determine th
```

```
//sprintf(str,"4: %02x,5: %02x,6: %02x,7: %02x\r\n",RX_BUF[4],RX_BUF[5],RX_B
             //HAL_UART_Transmit(&huart2,(uint8_t *)str,strlen(str),0xFFFF);
if((0x00==RX_BUF[5])&&(0x00==RX_BUF[14])&&(0x00==RX_BUF[22]))//Data comparis
{
  LED_OFF_2;
          LED_OFF_4;
           LED_OFF_6;

}

else {
 x_pos=(RX_BUF[5]<<8)+RX_BUF[4];//Get the X coordinate
 y_pos=(RX_BUF[7]<<8)+RX_BUF[6];//Get the Y coordinate
 int16_t tempx,tempy;
 tempx = 0x8000&x_pos;//Keep the highest bit
 x_pos = x_pos&0x7fff;//Clear the highest bit
 if(!tempx){//Determine whether the highest bit is 0
  x_pos = -x_pos;//If it is 0, it is negated (that is, the coordinate become
 }
 //Same as X coordinate
 tempy = 0x8000&y_pos;
 y_pos = y_pos&0x7fff;
 if(!tempy){
  y_pos = -y_pos;
 }



            char x_y_pos[32]={0};
sprintf(x_y_pos,"x_pos: %d,y_pos: %d\r\n",x_pos,y_pos);
            HAL_UART_Transmit(&huart2,(uint8_t *)x_y_pos,strlen(x_y_pos),0xF

angle = atan2 (x_pos,y_pos) * 180.0 / PI;//Use the trigonometric formula inv
char angle_str[32]={0};
sprintf(angle_str,"angle: %lf,\r\n",angle);
             HAL_UART_Transmit(&huart2,(uint8_t *)angle_str,strlen(angle_str

   if((angle>=(-60)&&(angle<(-20))))   //-60 degrees to -20 degrees
 {
   LED_ON_2;
   LED_OFF_4;
   LED_OFF_6;
 }
else  if((angle>=(-20)&&(angle<=20)))       //-20 degrees to 20 degrees
 {
   LED_OFF_2;
   LED_ON_4;
   LED_OFF_6;
 }
 else if((angle>20&&(angle<=60)))       //20 to 60 degrees
```

```
        {
          LED_OFF_2;
          LED_OFF_4;
          LED_ON_6;
        }

      }
       while(HAL_UART_GetState(&huart2)==HAL_UART_STATE_BUSY_TX);//Determine whethe
       memset(RX_BUF,0x00,sizeof(RX_temp));//Clear the cache array
       RX_count = 0;//Count position zero
```


Demo Demonstration

Refer to original link video: link

Source code:

CODING | 一站式软件研发管理平台

Coding.net 是一个面向开发者的云端开发平台，提供 Git/SVN 代码托管、任务管理、在线 WebIDE ·Cloud Studio、开发协作、文件管理、Wiki 管理、提供个人服务及企业版本的服务

e.coding.net

Contact us

Official website: https://www.ai-thinker.com

Development materials: https://docs.ai-thinker.com/

Official forum: http://bbs.ai-thinker.com

Business cooperation: tara@aithinker.com

LinkedIn: Ai-Thinker official

Company address: Building C 410, Huafeng Smart Innovation Port, Gushu, Xixiang, Bao’an District, Shenzhen

Contact number: 0755–29162996

— — — — — — — — — — — — — — — — — — —

Original link: https://blog.csdn.net/Boantong_/article/details/119462888

Radar Module Detect Module IoT

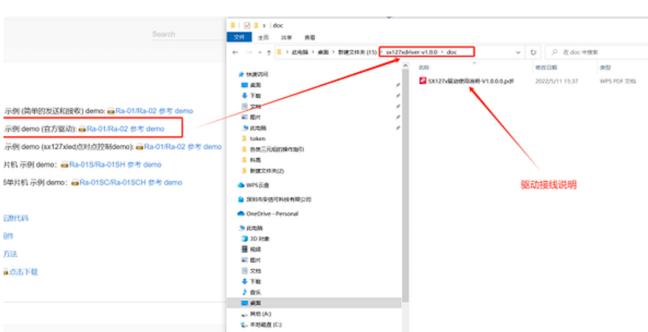
Written by Tara Qiu(Ai-Thinker)

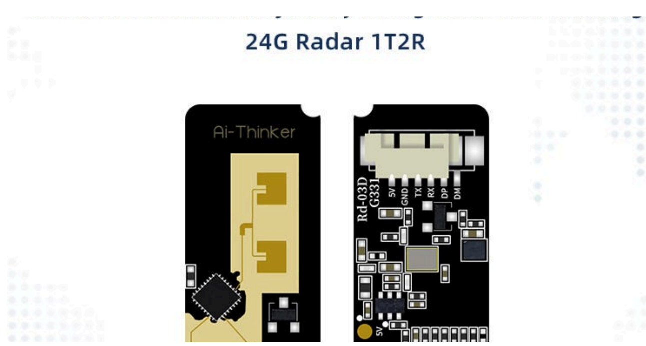
Follow

27 Followers

IoT module&Antenna information sharing, tara@aithinker.com

More from Tara Qiu(Ai-Thinker)





Tara Qiu(Ai-Thinker)

Ai-Thinker LoRa module Ra-01 ,Ra-02 change to Ra-01S /Ra-01SC…

This article mainly introduces the software details tutorial of LoRa SX1278 change to…

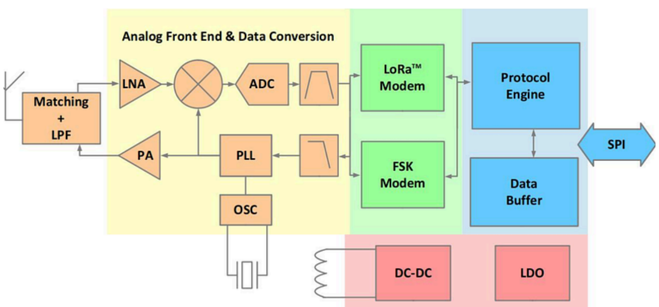
Jun 7

Tara Qiu(Ai-Thinker)

Ai-Thinker radar module Rd-03D

mmWave Radar Sensor Modules

Jan 9 · 52

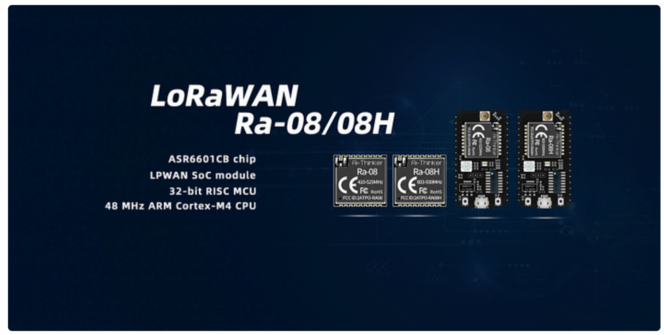


Tara Qiu(Ai-Thinker)

Ra-01SCH-P LoRa module PA version

#LoRa module #PA version #Details #Toturial tara@aithinker.com

Aug 4



Tara Qiu(Ai-Thinker)

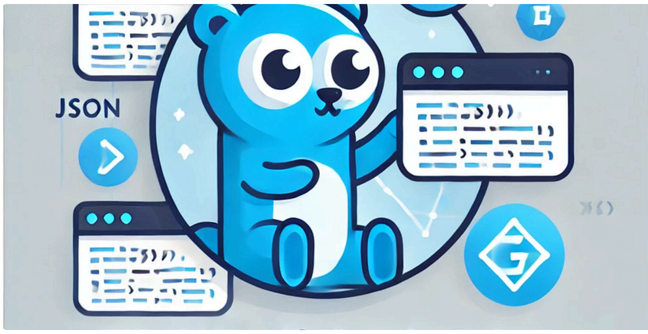
Ai-Thinker LoRaWAN module Ra-08/08H

Overview

Jan 3 · 52

See all from Tara Qiu(Ai-Thinker)

Recommended from Medium



Nando Septian Husni in Stackademic

Working with JSON in Golang

In this article we discuss json data, how we can convert string data types into json forma…

3d ago 263 5

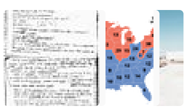
MCristina Content by Maria Cristina DeBriyn

Homemade Berry Pie with Crumb Topping

By: Maria Cristina DeBriyn

Oct 21 245 2

Lists



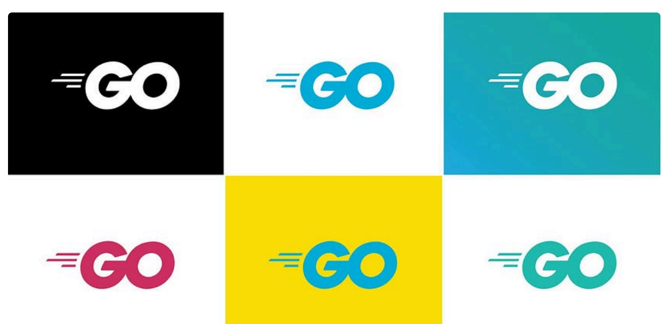
Staff Picks

765 stories · 1436 saves

Harendra

How I Am Using a Lifetime 100% Free Server

Get a server with 24 GB RAM + 4 CPU + 200 GB Storage + Always Free



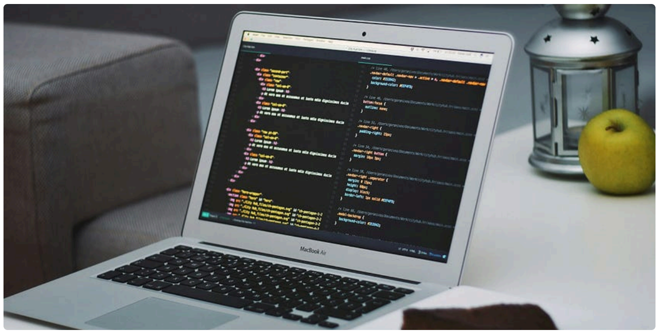
Raghav Joshi in ITNEXT

The Beauty of GO

Embracing the Elegance and Power of the GO Programming Language

Oct 26 4.5K 55

6d ago 92



Tom Deneire

Five ways to display LSP diagnostics in Neovim

Language Server Protocol

Oct 29 50

Ibrahim Nasribrahim

AI Test Automation Engineer Agent: Automating PR Review…

In modern software development, keeping code clean and adhering to guidelines is…

Sep 6 146

See more recommendations