

如何使用 6 台 TFmini-S 实现无人机避障 功能

www.benewake.com
Benewake (Beijing) Co., Ltd.



TFmini-S

<http://en.benewake.com/product/detail/5c345e26e5b3a844c472329c.html>

TFmini-S 与其他产品相比，具有成本低、体积小、功耗低、接口多等优势，能满足不同使用者的需求。其最大检测距离为 12 米，FOV 为 2 度。TFmini-S 具有较强的抗干扰能力，可于室外强光下工作。TFmini-S 激光雷达采用 UART (TTL) /I2C 通信接口，供电电压为 5V，平均功耗为 0.6w，重量仅有 4.7g。

产品配置方法

为实现点阵的避障方式，我们需要单独设置每台雷达的 I2C 地址并切换到 I2C 通讯接口。配置指令如下表所示：



输出模式	5A 05 05 01 65	5A 05 05 01 65	标准 9 字节(cm)	√
	5A 05 05 02 66	5A 05 05 02 66	字符串格式(m)	/
	5A 05 05 06 6A	5A 05 05 06 6A	标准 9 字节(mm)	/
波特率	5A 08 06 H1 H2 H3 H4 SU	5A 08 06 H1 H2 H3 H4 SU	设置波特率 ^② 例： 256000(DEC)=3E8 00(HEX), H1=00,H2=E8,H3= 03,H4=00	115200
输出开关	5A 05 07 00 66	5A 05 07 00 66	关闭数据输出	/
	5A 05 07 01 67	5A 05 07 01 67	使能数据输出	√
通信接口设置	5A 05 0A MODE SU	/	0 (UART) 1 (I2C)	UART
修改 I2C 从机地址	5A 05 0B ADDR SU	原指令	修改 I2c_slave_addr	0x10

使用串口转 USB 转接板将雷达与电脑连接，通过串口助手或者北醒上位机，发送配置指令更改 TFmini-S 的通讯接口，示例如下：

①更改 I2C 地址

配置指令： 5A 05 0B ADDR SU

SU 代表校验和，ADDR 代表 I2C 地址。

将 TFmini-s 设置为地址 11（雷达默认 I2C 地址是 10），发送指令：5A 05 0B 11 7B。更改 I2C 地址为 12、13、14、15 的指令如下：

5A 05 0B 12 7C

5A 05 0B 13 7D

5A 05 0B 14 7E

5A 05 0B 15 7F

②切换通讯接口为 I2C

配置指令： 5A 05 0A 01 6A

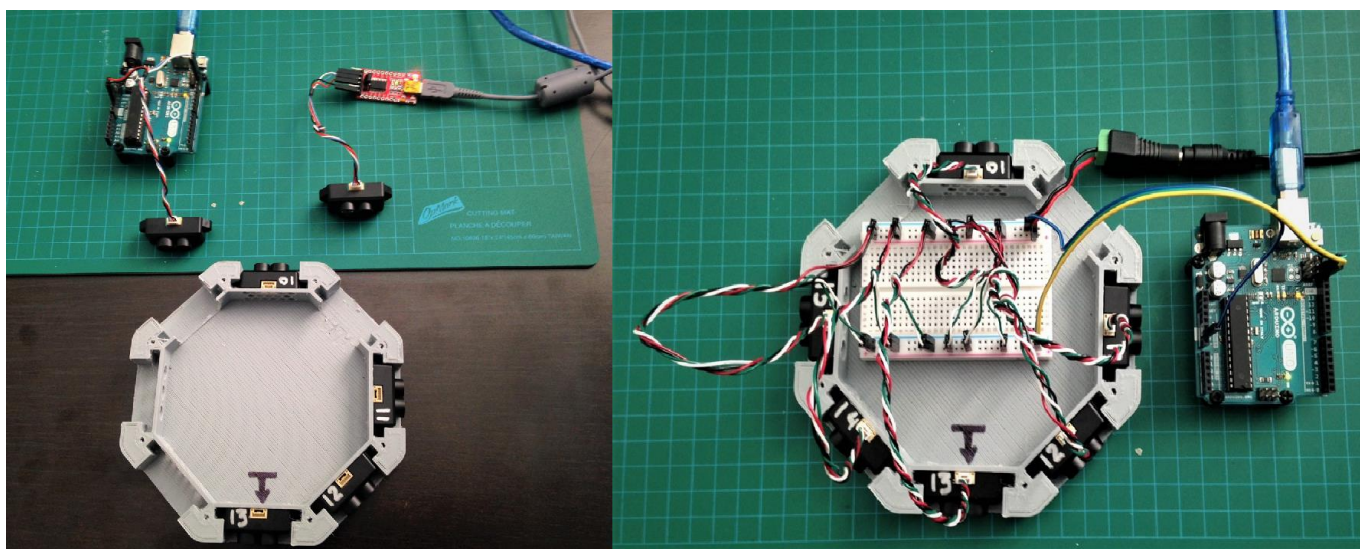
③保存设置

指令：5A 04 11 6F

通过 Arduino 检查配置结果

将 i2c_scanner.ino 程序加载到 Arduino 中，然后按照下方图片进行连接。

左侧图片：将每台 TFmini-S 进行 I2C 地址配置后用银色记号笔标记。右侧图片：将 TFmini-S 与开发板连接在一起，通过 5V 电源进行供电，同时将雷达的数据线连接到 Arduino 上。



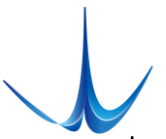
如果配置成功，可以看到如下结果：

```
Scanning...
I2C device found at address 0x10 !
I2C device found at address 0x11 !
I2C device found at address 0x12 !
I2C device found at address 0x13 !
I2C device found at address 0x14 !
I2C device found at address 0x15 !
done
```

如何将 TFmini-S 的接口配置恢复为串口？

通过使用 Bud Ryerson 的软件库 (<https://github.com/budryerson/TFMini-Plus-I2C>)，我们通过 Arduino 设置 -TFmini-S 的参数。

下面是将雷达的接口切换为串口的代码：



```
#include <Wire.h>      // Arduino standard I2C/Two-Wire Library

#include "printf.h"     // Modified by to support Intel based Arduino
#include <TFMPI2C.h>    // TFMMini-Plus I2C Library v0.2.2
TFMPI2C tfmP;          // Create a TFMMini-Plus I2C object

void setup()
{
    Wire.begin();       // Initialize two-wire interface
    Serial.begin( 115200); // Initialize terminal serial port
    printf_begin();     // Initialize printf library.
    delay(20);

    printf( "BACK to SERIAL: ");
    if( tfmP.sendCommand( SET_UART_MODE, 0))
    {
        printf( "passed.\r\n");
    }
    else tfmP.printErrorStatus();
}

// = = = = = = = = = =  MAIN LOOP  = = = = = = = = = =
void loop()
{
}
}
```

推荐使用 Arduino 进行 I2C 接口的测试和调试。在 ArduPilot 中使用的大部分数据库都基于 Arduino。



配置 ArduPilot

配置基于以下链接：

<https://ardupilot.org/dev/docs/code-overview-object-avoidance.html>

您可以配置多个雷达以便覆盖多个"扇区"（避障区域）。当前方案使用 5 个 TFmini-S 覆盖前向 180° 的避障区域，保留 1 个 TFmini-S 做后向的避障：

AVOID_ANGLE_MAX, 1000AVOID_BEHAVE,0

AVOID_DIST_MAX,1

AVOID_ENABLE,2

AVOID_MARGIN,1

PRX_TYPE,4

RNGFND2_ADDR,16

RNGFND2_FUNCTION,0

RNGFND2_GNDCLEAR,10

RNGFND2_MAX_CM,700

RNGFND2_MIN_CM,10

RNGFND2_OFFSET,0

RNGFND2_ORIENT,4

RNGFND2_PIN,-1

RNGFND2_POS_X,0

RNGFND2_POS_Y,0

RNGFND2_POS_Z,0

RNGFND2_PWRRNG,0

RNGFND2_RMETRIC,1
RNGFND2_SCALING,3
RNGFND2_STOP_PIN,-1
RNGFND2_TYPE,25

RNGFND3_ADDR,17
RNGFND3_ORIENT,6
RNGFND3_PIN,-1

RNGFND4_ADDR,18
RNGFND4_ORIENT,7

RNGFND5_ADDR,19
RNGFND5_ORIENT,0

RNGFND6_ADDR,20
RNGFND6_ORIENT,1

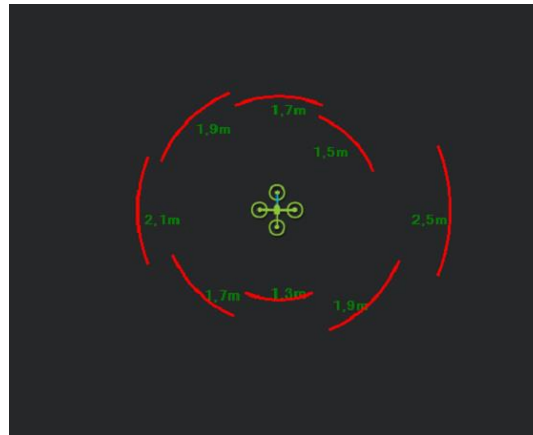
RNGFND7_ADDR,21
RNGFND7_ORIENT,2

定高功能使用 TFmini Plus ，保持默认的串口接口与 SERIAL 3 相连接，配置如下：

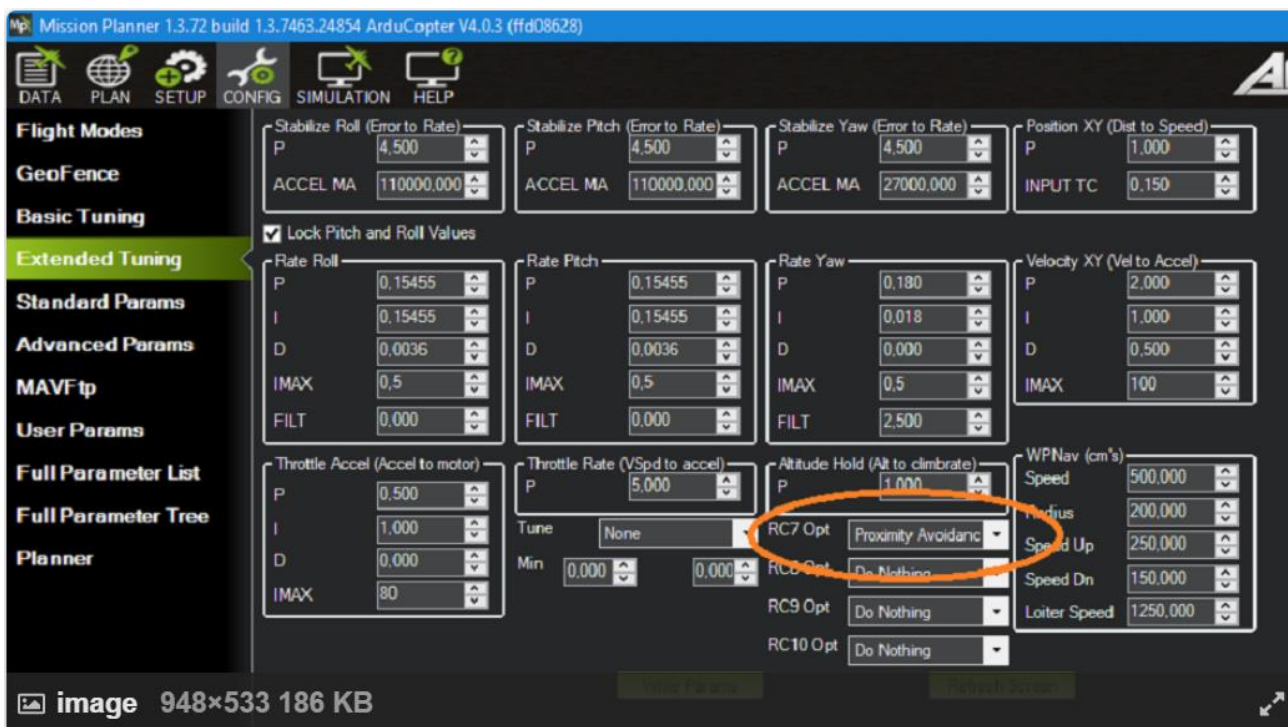
RNGFND1_ADDR,0
RNGFND1_GNDCLEAR,10
RNGFND1_MAX_CM,700
RNGFND1_MIN_CM,20
RNGFND1_ORIENT,25
RNGFND1_TYPE,20

测试

静态测试:在 Flight Data 界面下,按 Ctrl-F 和按下 Proximity 按钮可以打开下图界面。将"PRX_TYPE"参数设置为使能状态，界面可以显示雷达的距离数据。



将 RC7 设置为避障模式。



测试项目

1. 起飞准备
2. 定高飞行
3. 启动避障动能
4. 飞向墙体，观察无人机的反应

测试视频如下：

<https://youtu.be/yRIFXSe8uFc>