

ATK-MB021 模块使用说明

激光测距模块

使用说明

正点原子

广州市星翼电子科技有限公司

修订历史

版本	日期	原因
V1.0	2024/11/01	第一次发布

目 录

1, 硬件连接.....	1
2, 实验功能.....	2
2.1 ATK-MB021 激光测距模块测试实验.....	2
2.1.1 功能说明.....	2
2.1.2 源码解读.....	2
2.1.3 实验现象.....	8
3, 其他.....	9

1，硬件连接

这里以正点原子 M48Z-M3 最小系统板 STM32F103 版为例，给大家介绍一下模块和板卡的连接方法。其它板卡与模块的硬件连接方法，请大家在“**ATK-MB021 激光测距模块**3，**程序源码\相应板卡例程文件夹\readme.txt**”路径下查看。

ATK-MB021 激光测距模块可通过杜邦线与正点原子 M48Z-M3 最小系统板 STM32F103 版进行连接，具体的连接关系，如下表所示：

模块对应开发板	连接关系					
ATK-MB021 激光测距模块	VCC	GND	SDA	SCL	INT	XS
M48Z-M3 最小系统板 STM32F103 版	3.3V/5V	GND	PA3	PA2	PB5	PB6

表 1.1 激光测距模块与 M48Z-M3 最小系统板 STM32F103 版连接关系

2，实验功能

2.1 ATK-MB021 激光测距模块测试实验

2.1.1 功能说明

在本实验中，串口会打印激光测距模块所测量的距离值。需要查看这部分实验信息的用户，可用杜邦线将最小系统板 STM32F103 的 PA9 引脚和 GND 连接至外部的 USB 转串口设备，这样就可以通过 XCOM 上位机查看串口打印的信息了。

2.1.2 源码解读

打开本实验的工程文件夹，能够在./Drivers/BSP 目录下看到 ATK_TOF 文件夹和 IIC 文件夹，其中 ATK_TOF 文件夹中包含了 ATK-MB021 激光测距模块的驱动文件，IIC 文件夹中包含了 IIC 的驱动文件，如下图所示：

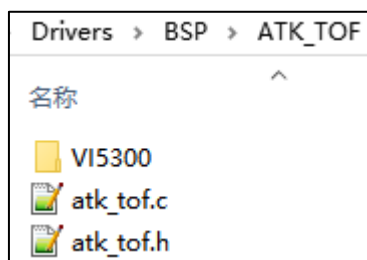


图 2.1.2.1 ATK-MB021 激光测距模块驱动代码

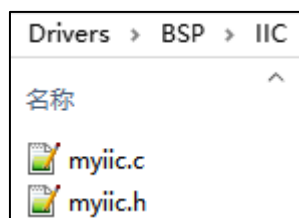


图 2.1.2.2 IIC 驱动代码

在图 2.1.2.1 中，VI5300 文件夹内存放的是芯片官方提供的驱动文件，用户不需要自行编写寄存器相关的底层驱动，只需要关注 API 函数即可，这大大提升了开发的效率。

2.1.2.1 ATK-MB021 激光测距模块驱动

首先看 atk_tof.c 中几个重要的函数。

1. 函数 atk_tof_hw_init()

该函数用于初始化 ATK-MB021 激光测距模块，具体的代码如下所示：

```
/**
 * @brief      ATK-TOF 模块硬件初始化
 * @param      无
 * @retval     无
 */
void atk_tof_hw_init(void)
{
    VI530x_Status ret = VI530x_OK;
```

```
/* 初始化 IIC 接口 */
iic_init();

/* 软件中断，GPIO 引脚直接上拉 */
VI530x_Cali_Data.VI530x_Interrupt_Mode_Status = 0x00;

/* 初始化 VI5300 */
ret |= VI530x_Chip_Init();

/* 写入 VI5300 固件 */
ret |= VI530x_Download_Firmware((uint8_t *)VI5300_M31_firmware_buff,
                                FirmwareSize());

/* 配置标定值 */
ret |= VI530x_Set_Californiation_Data(
    VI530x_Cali_Data.VI530x_Calibration_Offset);

/* 开启温度校准，0x00:关，0x01:开 */
ret |= VI530x_Set_Sys_Temperature_Enable(0x01);

/* 配置帧率，积分次数 */
ret |= VI530x_Set_Integralcounts_Frame(30, 131072);

/* 配置数据读取模式 */
#if CONTINUOUS_READ == 1
    ret |= VI530x_Start_Continue_Ranging_Cmd();    /* 连续模式 */
#else
    ret = VI530x_Start_Single_Ranging_Cmd();      /* 单次模式 */
#endif

if(ret)
{
    printf("VI5300 初始化失败!\r\n");
}
else
{
    printf("VI5300 初始化成功!\r\n");
}
}
```

atk_tof_hw_init()函数的配置流程较多，首先调用 IIC 初始化函数，使 IIC 接口能正常工作，接着配置中断方式为软件中断，即 INT 引脚不启用，然后初始化 VI5300、写入固件、设置标定值、开启温度校准、配置帧率、积分次数和数据读取模式（默认为连续模式），最后判断是否初始化成功，并在串口打印初始化的结果。值得注意的是，写入的固件、标定值、

帧率和积分次数都不建议修改，芯片厂家已经测试好这些参数了，直接使用即可。

2. 测距相关的数据收发函数

该系列函数用于收发激光测距模块的数据，具体代码如下所示：

```
/**
 * @brief 写 N 个字节到 ATK-TOF 模块的寄存器
 * @param reg_addr: 寄存器地址
 * @param data: 写入数据
 * @param len: 数据长度
 * @retval 写入结果
 * @arg 0 : 成功
 * @arg 其他: 失败
 */
VI530x_Status atk_tof_write_nbytes(uint8_t reg_addr, uint8_t *data, uint16_t
len)
{
    iic_start(); /* 起始信号 */

    iic_send_byte(ATK_TOF_IIC_ADDR); /* 发送 IIC 通讯地址 */
    if (iic_wait_ack() == 1) /* 等待应答 */
    {
        iic_stop();
        return VI530x_IIC_ERROR;
    }

    iic_send_byte(reg_addr); /* 发送寄存器地址 */
    if (iic_wait_ack() == 1) /* 等待应答 */
    {
        iic_stop();
        return VI530x_IIC_ERROR;
    }

    for (uint16_t i = 0; i < len; i++) /* 循环发送数据 */
    {
        iic_send_byte(data[i]);
        if (iic_wait_ack() == 1) /* 等待应答 */
        {
            iic_stop();
            return VI530x_IIC_ERROR;
        }
    }

    iic_stop(); /* 停止信号 */

    return VI530x_OK;
}
```

```
}

/**
 * @brief  写一个字节到 ATK-TOF 模块的寄存器
 * @param  reg_addr:   寄存器地址
 * @param  data:       数据
 * @retval  写入结果
 * @arg    0    : 成功
 * @arg    其他: 失败
 */
VI530x_Status atk_tof_write_byte(uint8_t reg_addr, uint8_t data)
{
    uint8_t ret = 0;

    ret = atk_tof_write_nbytes(reg_addr, &data, 1);    /* 写入一个字节数据 */

    if(ret != 0)
    {
        return VI530x_IIC_ERROR;
    }
    else
    {
        return VI530x_OK;
    }
}

/**
 * @brief  从 ATK-TOF 模块读取 N 个字节数据
 * @param  reg_addr:   寄存器地址
 * @param  data:       数据存储 buff
 * @param  len:        数据长度
 * @retval  读出结果
 * @arg    0    : 成功
 * @arg    其他: 失败
 */
VI530x_Status atk_tof_read_nbytes(uint8_t reg_addr, uint8_t *data, uint16_t len)
{
    iic_start();                                /* 起始信号 */

    iic_send_byte(ATK_TOF_IIC_ADDR);           /* 发送 IIC 通讯地址 */
    if (iic_wait_ack() == 1)                    /* 等待应答 */
    {
        iic_stop();
        return VI530x_IIC_ERROR;
    }
}
```

```
}

iic_send_byte(reg_addr);          /* 发送寄存器地址 */
if (iic_wait_ack() == 1)          /* 等待应答 */
{
    iic_stop();
    return VI530x_IIC_ERROR;
}

iic_start();                      /* 起始信号 */

iic_send_byte(ATK_TOF_IIC_ADDR | 0x01);
if (iic_wait_ack() == 1)          /* 等待应答 */
{
    iic_stop();
    return VI530x_IIC_ERROR;
}

while (len)                       /* 循环接收数据 */
{
    *data = iic_read_byte((len > 1) ? 1 : 0);
    len--;
    data++;
}

iic_stop();                      /* 停止信号 */

return VI530x_OK;
}

/**
 * @brief 从 ATK-TOF 模块读取 1 个字节数据
 * @param reg_addr: 寄存器地址
 * @param data: 数据存储 buff
 * @arg 0 : 成功
 * @arg 其他: 失败
 */
VI530x_Status atk_tof_read_byte(uint8_t reg_addr, uint8_t *data)
{
    uint8_t ret = 0;

    ret = atk_tof_read_nbytes(reg_addr, data, 1); /* 读取一个字节数据 */

    if(ret != 0)
```



```

{
    return VI530x_IIC_ERROR;
}
else
{
    return VI530x_OK;
}
}
    
```

以上 4 个函数通过 IIC 的协议与激光测距模块进行通信，实现数据的收发。

除了上述的初始化配置和数据收发函数之外，我们还需要关注距离值的获取，接下来简单介绍一下 VI530x_API.c (路径为 BSP/ATK_TOF/VI5300) 中的 VI530x_Get_Measure_Data() 函数，其函数原型如下所示：

```
VI530x_Status VI530x_Get_Measure_Data(VI530x_MEASURE_TypeDef *measure_data);
```

该函数的形参，如下表所示：

参数	描述
*measure_data	指向返回数据的结构体指针

表 2.1.2.1 VI530x_Get_Measure_Data 函数形参描述

获取距离值成功后，将会返回 VI530x_OK 的状态。

注意：VI5300 激光测距传感器的通信协议时序与命令格式详见“ATK-MB021 激光测距模块V4，参考资料\芯视界_VI5300_数据手册_V1.5”第 5.1 章节。

2.1.2.2 IIC 驱动

在图 2.1.2.2 中，myiic.c 和 myiic.h 为 IIC 驱动文件，里面实现了 IIC 引脚的初始化、IIC 协议相关的系列函数。关于 IIC 的驱动介绍，请查看正点原子各个开发板对应的开发指南中 IIC 章节。

2.1.2.3 实验测试代码

实验的测试代码在 demo.c 文件中，该文件在工程根目录下的 User 文件夹。测试代码的入口函数为 demo_run()，具体的代码，如下所示：

```

/**
 * @brief      例程演示入口函数
 * @param      无
 * @retval     无
 */
void demo_run(void)
{
    VI530x_Status ret = VI530x_OK;
    VI530x_MEASURE_TypeDef result;

    atk_tof_hw_init();    /* 初始化 ATK_TOF 模块 */

    while (1)
    {
        ret = VI530x_Get_Measure_Data(&result);    /* 读取距离数据 */
    }
}
    
```

```

        if(ret == VI530x_OK)                                /* 读取成功 */
        {
            printf("距离 = %4d mm \r\n", result.correction_tof);
        }
    }
}
    
```

从上面代码可以看出，整个测试代码的逻辑相对简单。初始化激光测距模块后，在 while 循环中调用相关函数开始测距值，如果读取正常，则打印测出的距离值。

2.1.3 实验现象

将 ATK-MB021 激光测距模块按照第一节“硬件连接”中介绍的连接方式与开发板连接，并将实验代码编译烧录至开发板中。本实验使用串口输出调试信息，因此需将开发板的 PA9 连接至 DAP 虚拟串口（或 USB 转 TTL 模块）的 RX 引脚。完成连接后，可通过串口调试助手 XCOM 查看实验信息输出，如下图所示：

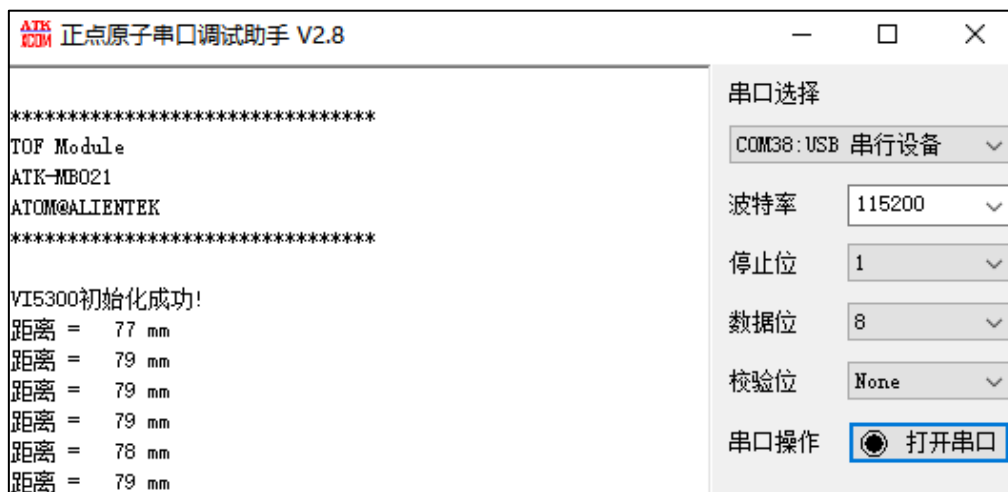


图 2.1.3.1 串口调试助手显示内容

注：其它现象请看 2.1.1 功能说明。

3，其他

1、购买地址：

天猫：<https://zhengdianyuanzi.tmall.com>

淘宝：<https://openedv.taobao.com>

2、资料下载

模块资料下载地址：<http://www.openedv.com/docs/index.html>

3、技术支持

公司网址：www.alientek.com

技术论坛：<http://www.openedv.com/forum.php>

在线教学：www.yuanzige.com

B 站视频：<https://space.bilibili.com/394620890>

传真：020-36773971

电话：020-38271790

