

## 解决ina260设备的I2C函数调用问题 (2025-10-14 15:46Z)

**User (2025-10-16 14:30Z)**

```
{
  "resource": "/e:/stm32projects/stm32workspace/curvoltpowMonitor/Core/Lib/ina260.c",
  "owner": "generated_diagnostic_collection_name#3",
  "code": "-Wunused-function",
  "severity": 4,
  "message": "Unused function 'i2c_read'",
  "source": "clang",
  "startLineNumber": 497,
  "startColumn": 12,
  "endLineNumber": 497,
  "endColumn": 20,
  "tags": [
    1
  ],
  "origin": "extHost1"
}
```

但是有ina260 device.i2c read，会调用static int i2c\_read和static int i2c\_write吗？

Assistant (copilot/claude-sonnet-4.5)

看了您的代码,问题在于**不会调用**。让我解释原因:

在 `ina260_init()` 函数中:

```

int ina260_init(i2c_func i2c_read, i2c_func i2c_write, uint8_t addr) {
    if (i2c_read == NULL || i2c_write == NULL) {
        ina260_device.i2c_read = NULL;
        ina260_device.i2c_write = NULL;
        return INA_STATUS_ERROR;
    }
    // ...
    ina260_device.i2c_read = i2c_read;    // 赋值传入的函数指针
    ina260_device.i2c_write = i2c_write;
    // ...
}

```

而在 main.c 中调用时:

```

int ret = ina260_init(NULL, NULL, INA260_SLAVE_ADDRESS);

```

传入的是 NULL ,所以 ina260\_device.i2c\_read 和 ina260\_device.i2c\_write 都被设置为 NULL ,不会调用底层的 static int i2c\_read() 和 static int i2c\_write() 。

## 解决方案

需要在 ina260.c 中声明这两个静态函数,然后在初始化时传入:

```

// 将这两个函数的声明移到文件前面
static int i2c_read(uint8_t addr, uint16_t *data, uint8_t data_len);
static int i2c_write(uint8_t addr, uint16_t *data, uint8_t data_len);

```

然后在 main.c 中修改初始化调用:

// 在 ina260.c 中添加一个便捷的初始化函数

// 或者直接在 main.c 中这样调用：

// 方法1：在 ina260.c 添加默认初始化函数

```
int ina260_init_default(uint8_t addr);
```

// 方法2：或者导出这两个函数(改为非 static)

```
extern int i2c_read(uint8_t addr, uint16_t *data, uint8_t data_len);
```

```
extern int i2c_write(uint8_t addr, uint16_t *data, uint8_t data_len);
```

// 然后在 main.c 中：

```
int ret = ina260_init(i2c_read, i2c_write, INA260_SLAVE_ADDRESS);
```

**推荐方法:**在 ina260.c 中添加便捷函数:

// 在文件末尾添加

```
int ina260_init_default(uint8_t addr) {  
    return ina260_init(i2c_read, i2c_write, addr);  
}
```

然后在 main.c 中:

```
int ret = ina260_init_default(INA260_SLAVE_ADDRESS);
```

并在 ina260.h 中添加声明:

```
int ina260_init_default(uint8_t addr);
```

这样就能正确调用底层的 I2C 读写函数了。