



A76xx_R3_Series_OpenSD K_用户开发与DEMO使用指南

LTE Module

SIMCom Wireless Solutions Limited

Building B, SIM Technology Building, No.633, Jinzhong Road
Changning District, Shanghai P.R. China

Tel: 86-21-31575100

support@simcom.com

www.simcom.com

文档名称:	A76xx_R3_Series_OpenSDK_用户开发与 DEMO 使用指南
版本:	2.00.00
日期:	2024.4.1
状态:	已发布

版权声明

本手册包含芯讯通无线科技（上海）有限公司（简称：芯讯通）的技术信息。除非经芯讯通书面许可，任何单位和个人不得擅自摘抄、复制本手册内容的部分或全部，并不得以任何形式传播，违反者将被追究法律责任。对技术信息涉及的专利、实用新型或者外观设计等知识产权，芯讯通保留一切权利。芯讯通有权在不通知的情况下随时更新本手册的具体内容。

本手册版权属于芯讯通，任何人未经我公司书面同意进行复制、引用或者修改本手册都将承担法律责任。

芯讯通无线科技(上海)有限公司

上海市长宁区金钟路 633 号晨讯科技大楼 B 座 6 楼

电话：86-21-31575100

邮箱：simcom@simcom.com

官网：www.simcom.com

了解更多资料，请点击以下链接：

<http://cn.simcom.com/download/list-230-cn.html>

技术支持，请点击以下链接：

<http://cn.simcom.com/ask/index-cn.html> 或发送邮件至 support@simcom.com

版权所有 © 芯讯通无线科技(上海)有限公司 2024，保留一切权利。

前言

感谢使用 SIMCom 提供的 A76xx 系列模块。该文档适用于 SIMCom A76xx Series,用于说明 Open SDK simcomDemoLinker 工具的使用方法，DEMO UI 界面介绍与使用及示例。

使用前请仔细阅读用户手册，您将领略其完善的功能和简洁的操作方法。

此模块主要用于语音或者数据通讯，本公司不承担由于用户不正常操作造成的财产损失或者人身伤害责任。请用户按照手册中的技术规格和参考设计开发相应的产品。同时注意使用移动产品应该关注的一般安全事项。在未声明之前，本公司有权根据技术发展的需要对本手册内容进行修改。

版本历史

版本	日期	章节	变更人	变更描述
V1.00.00	2023.12.15		谢必灏	新版本
V2.00.00	2024.1.15		谢必灏	新版本

目录

目录.....	4
1 引言.....	5
2 simcomDemoLinker 使用.....	5
2.1 DemoLinker.....	5
2.2 FileExplorer.....	9
2.3 LogViewer 窗口.....	12
3 DEMO 介绍与使用.....	16
3.1 DEMO UI 界面介绍.....	16
3.2 DEMO 操作方法.....	17
4 交互示例.....	18
4.1 获取菜单.....	18
4.2 列出所有文件.....	20
4.3 发送文件.....	20
4.4 读取文件.....	22

1 引言

本文档详细介绍了用户如何使用 simcomDemoLinker 工具，DEMO UI 界面的介绍与使用及示例。可以让用户快速上手，增加开发效率。

2 simcomDemoLinker 使用

2.1 DemoLinker

连接好模块，打开安装好的 simcomDemoLinker 工具。会自动进行连接，连接成功会进入交互界面，在 DemoLinker 的 Menu 菜单中显示 Demo 列表，选择想要测试的 Demo，点击右侧的 Do 执行 Demo。右下角会显示原始数据传输进度，左下角显示连接状态：绿色表示已连接，红色表示未连接。（若未连接，关闭工具再重新打开）

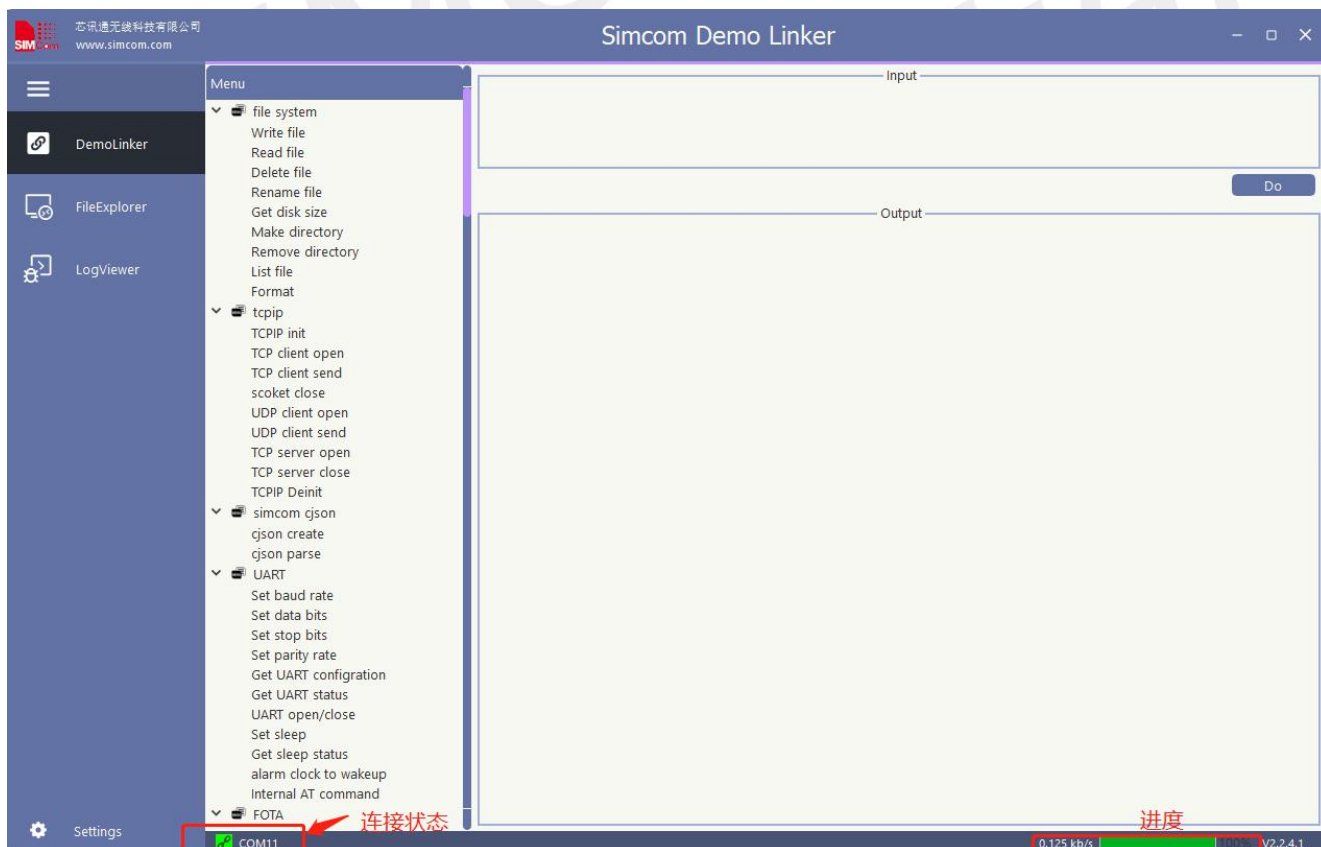


图 1 Linker 界面

在 DemoLinker 工具的左下角点击打开【Settings】，【Port】打开可以设置连接的端口（Auto 会自动遍历所有端口进行连接），下方的【Connect/Refresh】进行连接/重连，【Disconnect】断开当前连接。【Send speed(kb/s)】设置发送速率。

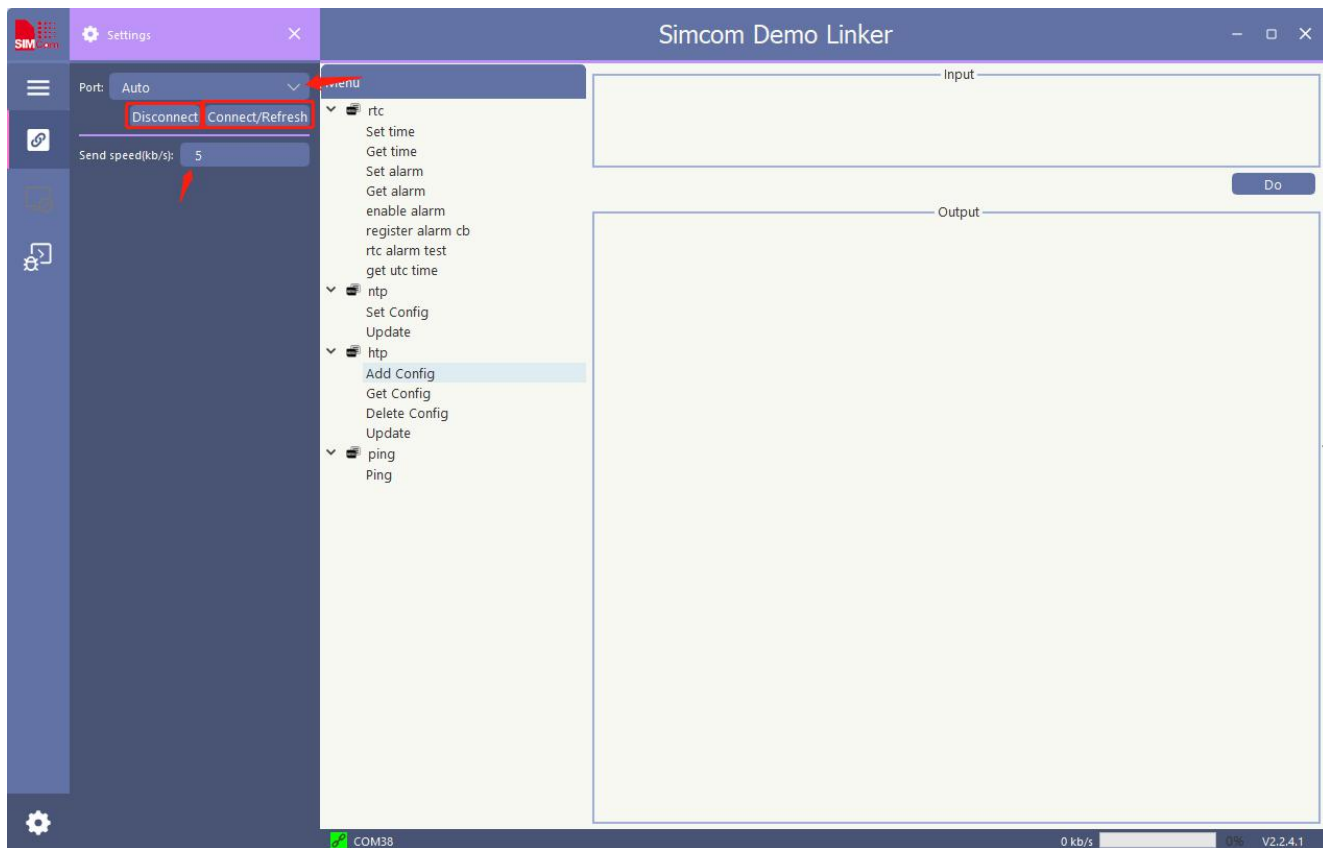


图 2 连接设置

例：

双击选择【Write file】写文件，在右边窗口中输入文件路径名、文件大小与文件数据，文件路径名可勾选右侧的【Browse】选择本地路径下的文件，文件的数据可以勾选右侧的【Browse】可以选择本地的原始数据作为写入文件的数据（不能多选），最后点击【DO】发送，下方【Output】中会显示发送状态。（注：DATA窗口中，可以直接粘贴数据。在【DATA】中输入数据时，【fileSize】会自动识别数据大小根据输入的数据调整大小。当文件存在时写入，会删除原来数据写入新数据）

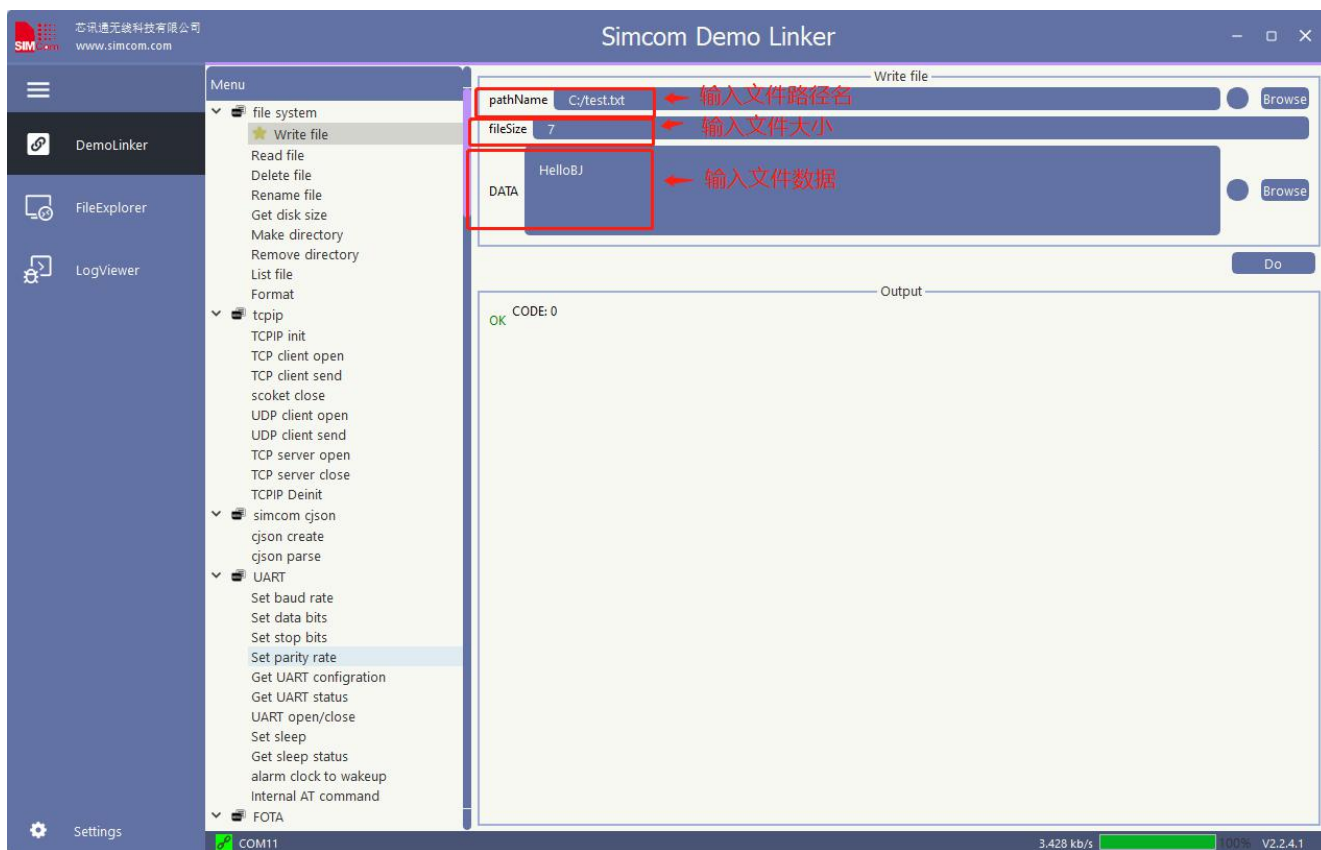


图 3 Write file 界面

双击选择【Read file】读取文件，在右侧窗口中输入文件路径名（可在【List file】中查看存在的文件），点击【Do】读取文件，在【Output】的 DATA 窗口中显示文件数据。点击右侧的【Save DATA】可选择本地路径保存该文件。（注：在 DATA 窗口中的数据，可以使用鼠标选中复制）

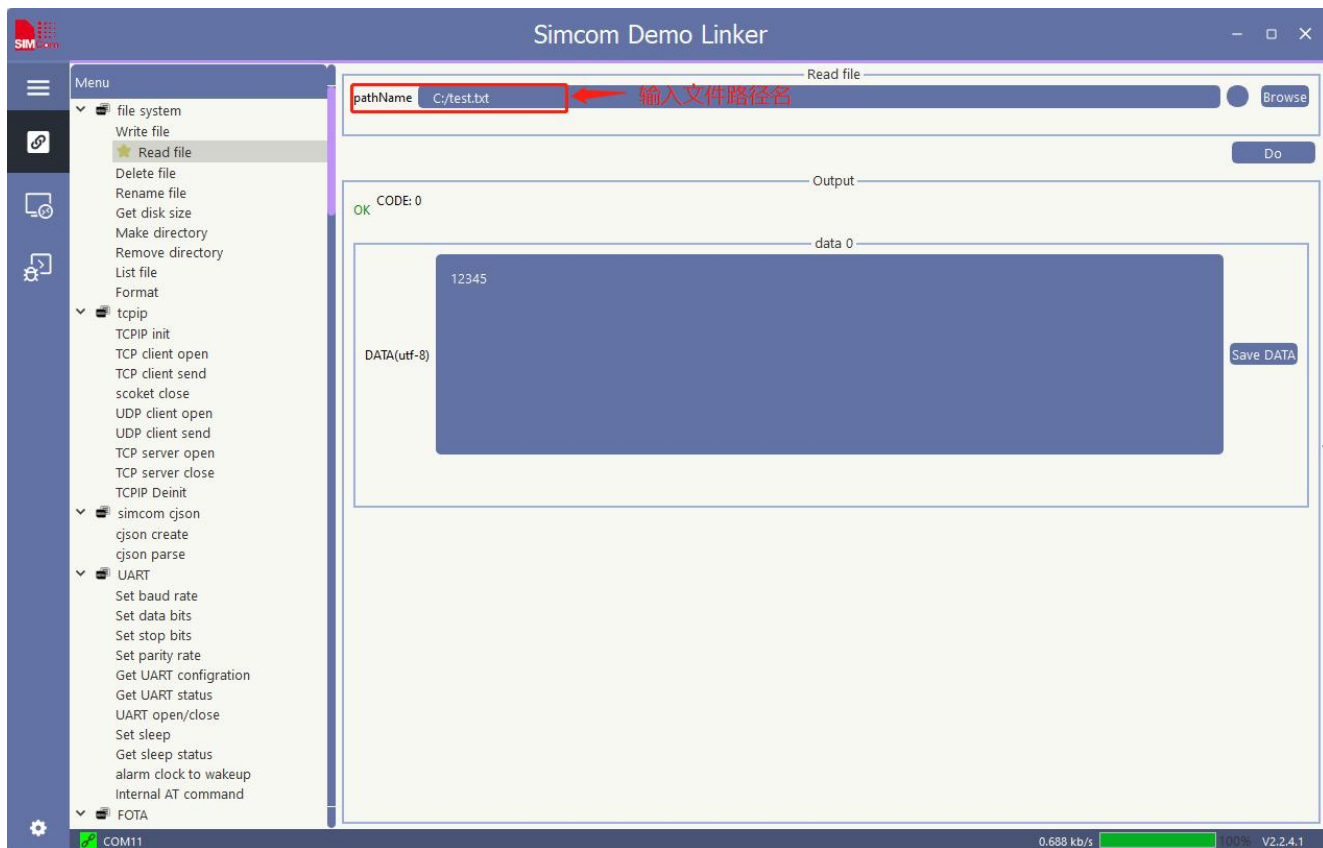


图 4 Read file 界面

双击选择【List file】查看目录数据，在右侧输入目录路径（注：不能输入文件名），点击【Do】查看该目录下的内容：名称、类型以及大小。可复制列表下的文件名在【Read file】中读取文件内容。

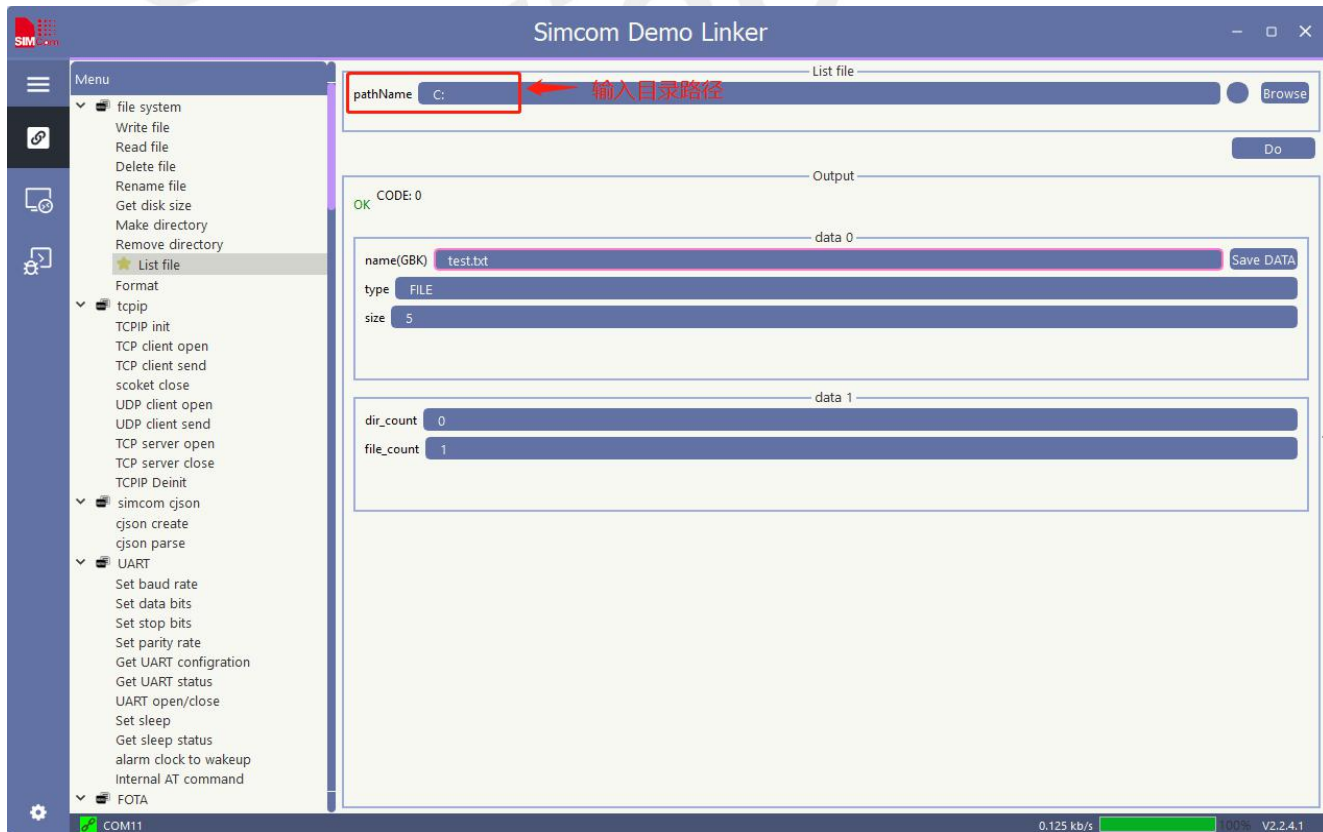


图 5 List file 界面

2.2 FileExplorer

连接成功后，会出现 FileExplorer 窗口，显示磁盘状况。打开最右侧的<处，可以设置支持的磁盘。

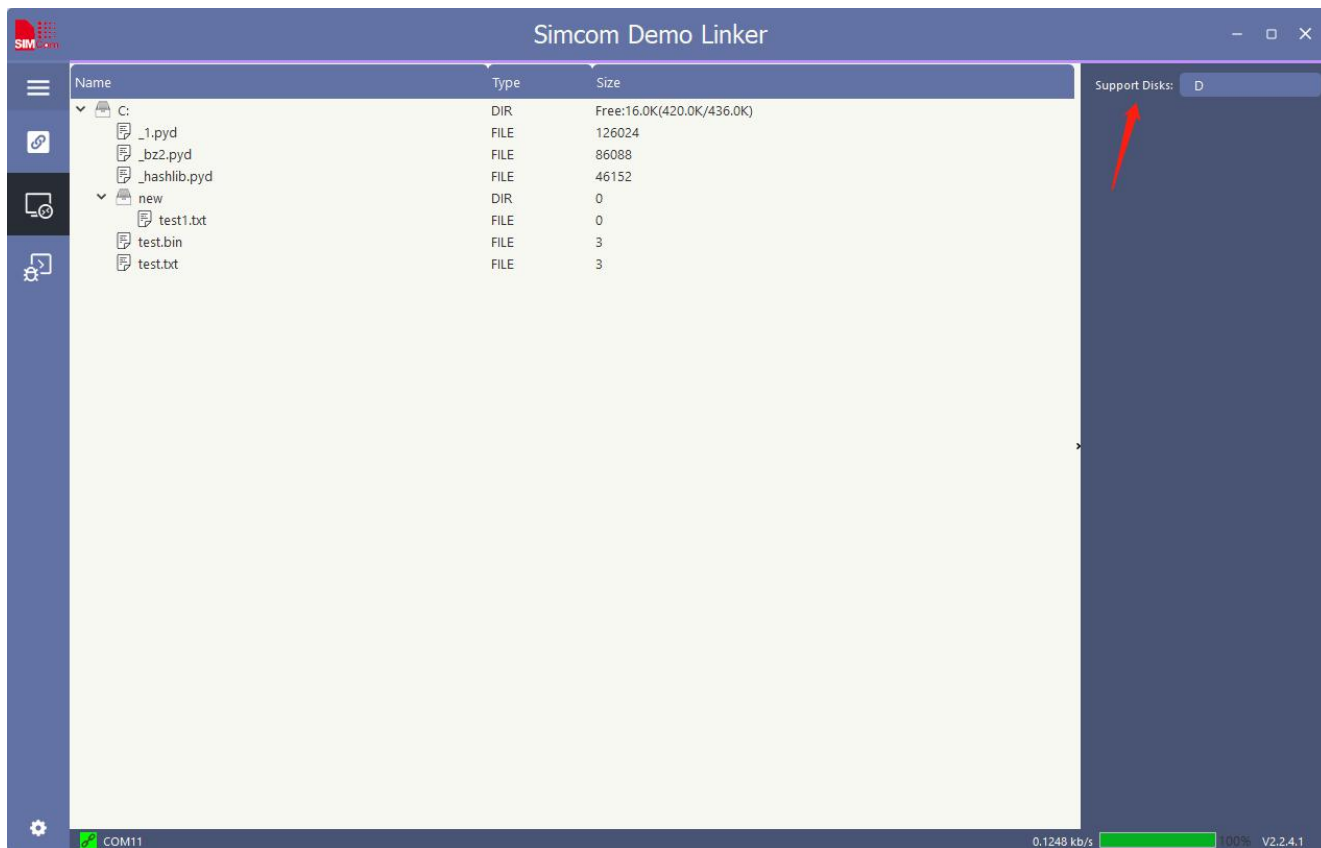


图 6 FileExplorer 界面

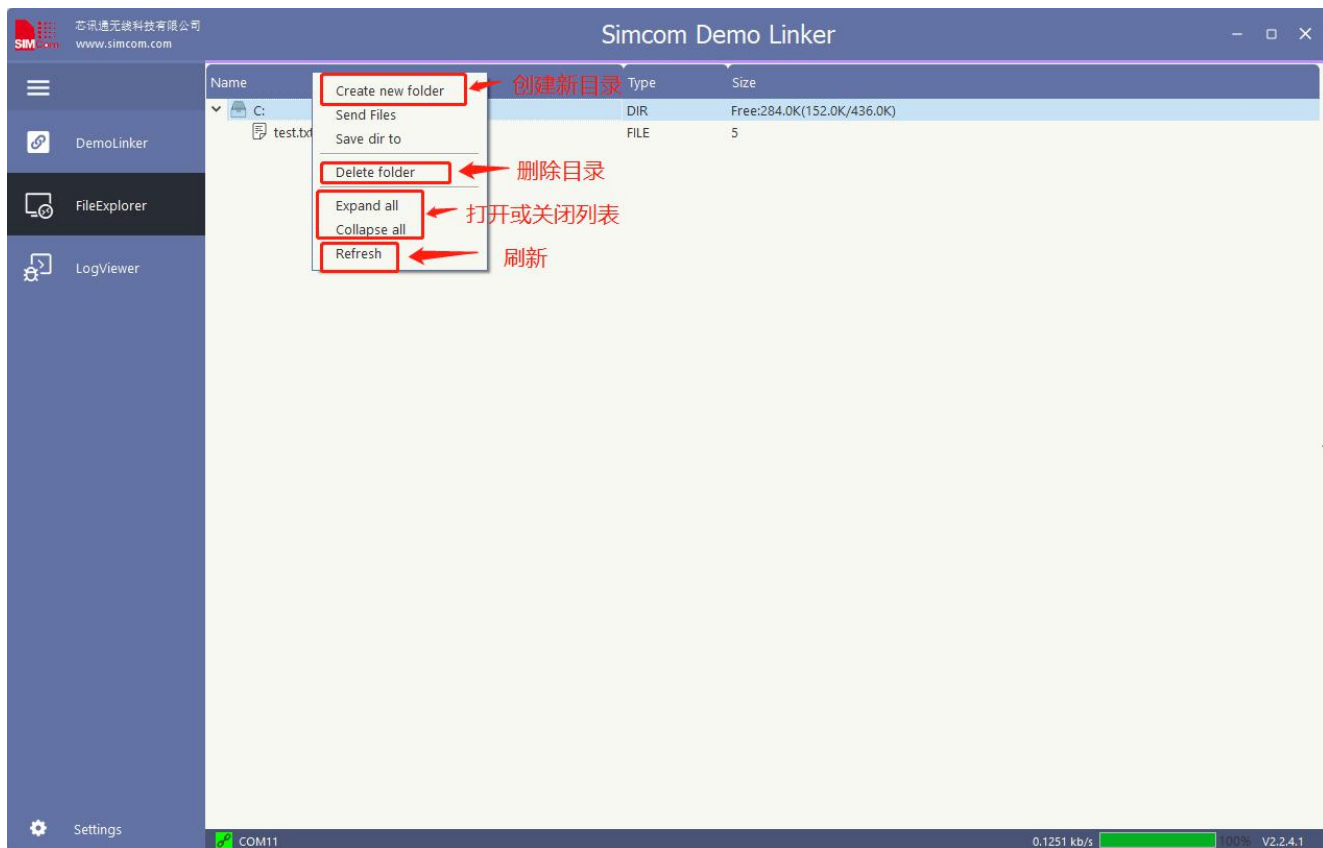


图 7 目录操作

右键点击磁盘，可进行创建新目录、删除目录、打开所有列表、关闭所有列表、刷新以及发送文件和保存目录。（也可直接空白处点击右键，进行刷新、打开关闭列表操作）

点击【send Files】，进入如下界面选择想要发送的文件，可以选择多个文件，最后点击【打开】。

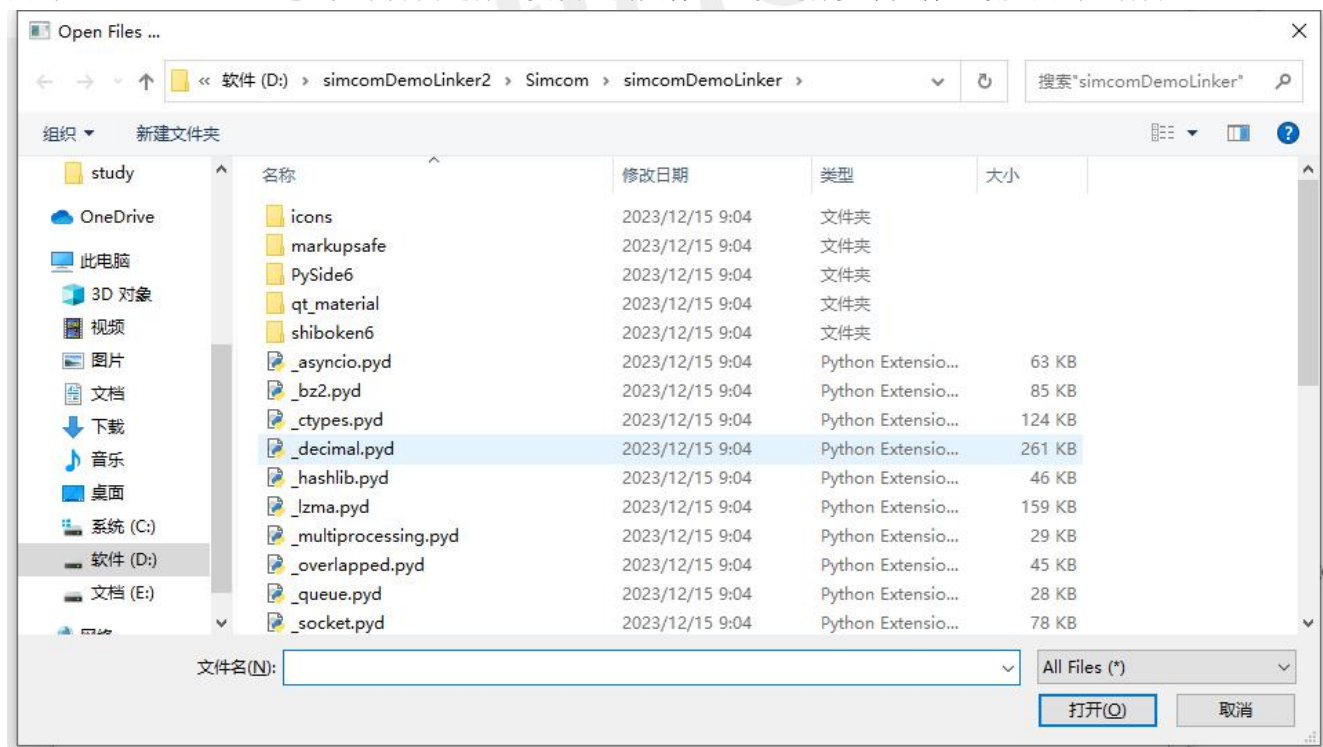


图 8 发送文件

当发送的文件大小大于空闲内存的大小时，会报错，Warning 窗口将会列出所有不能发送的文件。

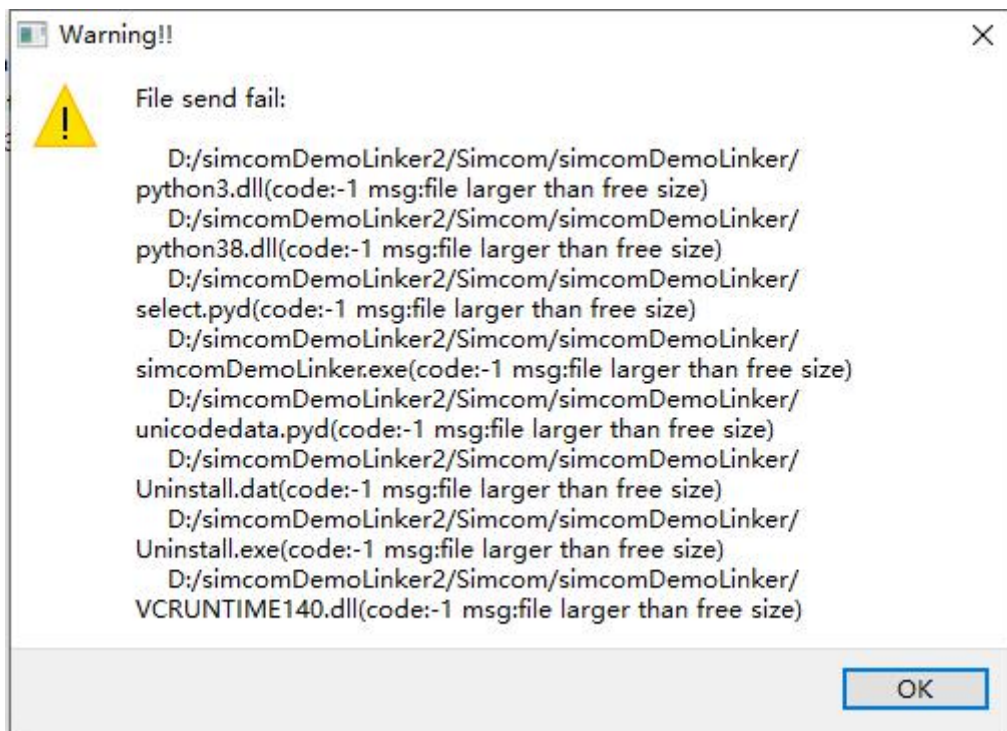


图 9 发送警告

用户可选择目录下的一个或多个文件，点击右键选择【Save files to】可以选择本地路径进行保存。也可以【Delete Files】删除文件。双击文件可以对文件进行重命名。

（注：当多选选中了目录时，进行保存时不会将该目录保存，只会保存该目录下的文件。）

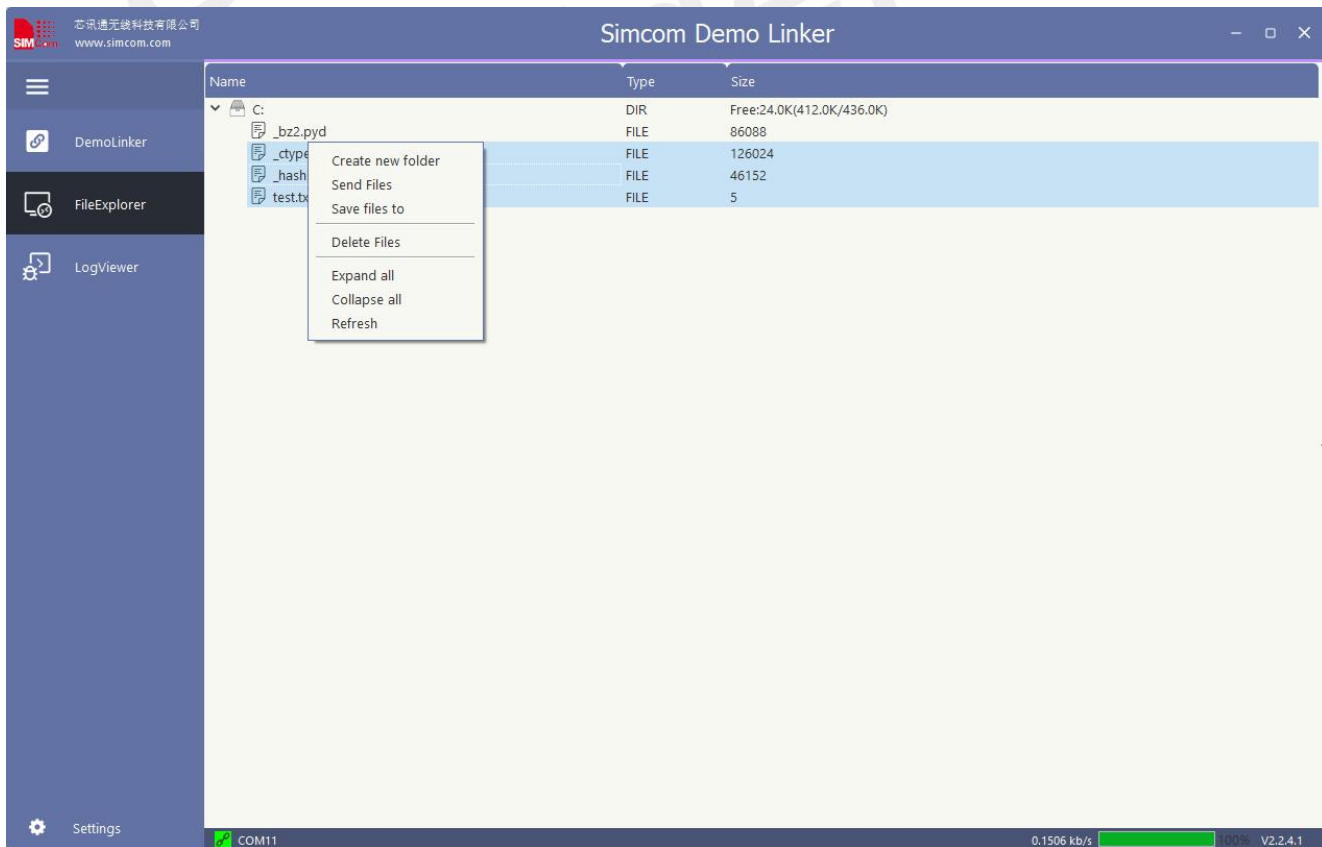


图 10 目录下文件操作

当想要保存目录时，右键点击目录，点击【Save dir to】选择本地路径保存目录。（保存该目录下所有文件和目录）

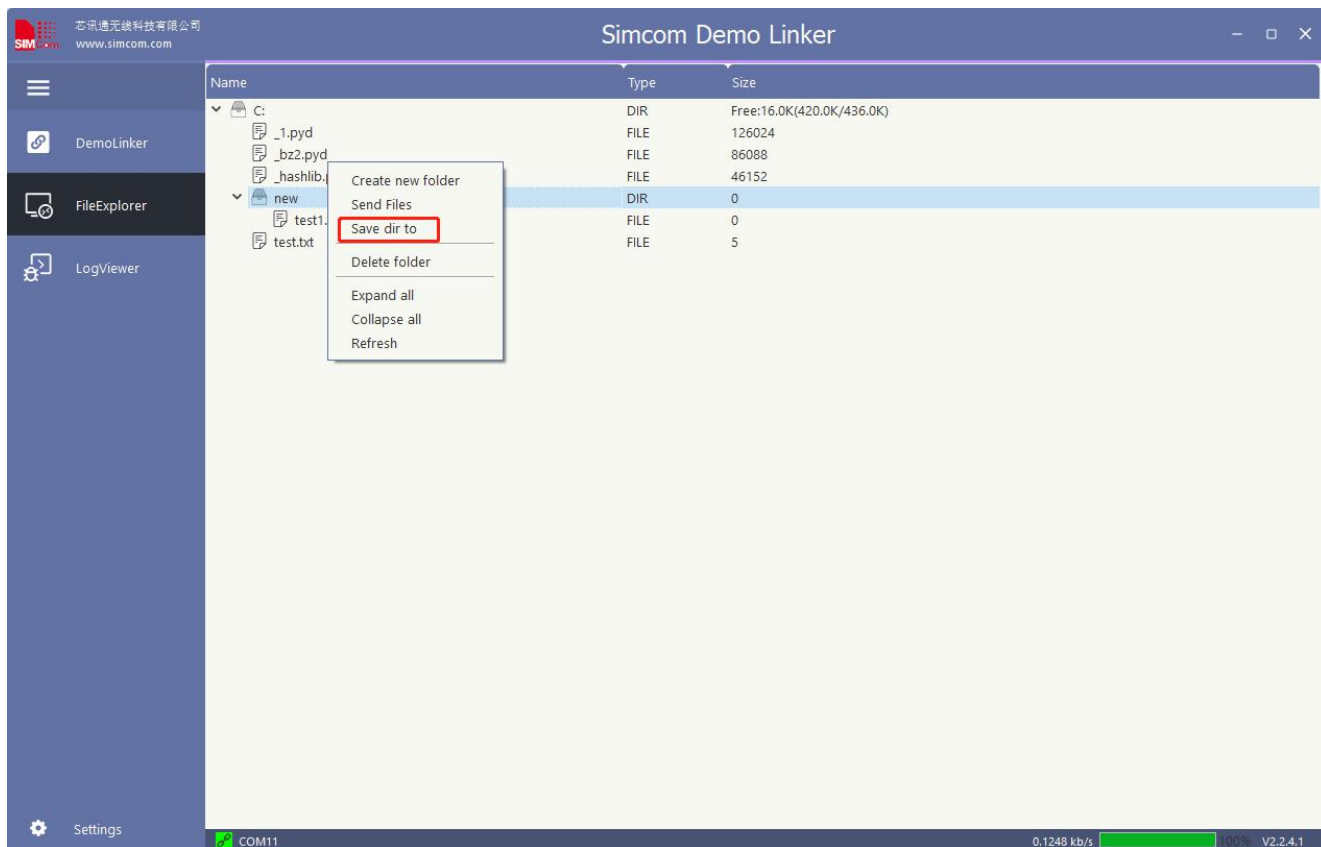


图 11 保存目录

底部框中会显示操作状态。

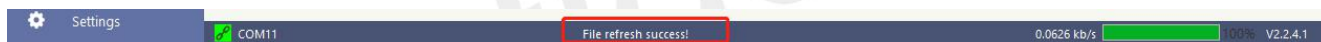


图 12 操作状态

2.3 LogViewer 窗口

在窗口中显示 log（以上的 demo 操作都会打印对应的 log），用户可在下方的输入栏中输入指令，点击 Send 发送。可勾选 Add <LF>添加换行符。

打开最右侧的<处，可以选择【Auto Wrap】自动换行，【Auto Save】自动保存 log 文件，【Max lines】设置最大 log 打印行数（默认为 5000），【log path】log 保存的路径（默认路径为：此电脑\文档\simcomDemoLinker\log），【Browse】选择自己想要保存的路径，【Open】打开“此电脑\文档”路径。

下方的【Save Log】可以保存 log 到想要的路径下，【Clean】清空当前窗口打印的 log。



图 13 log 窗口

打开工具会自动生成一个 **err.log** 文件，里面会自动保存一些重要的错误数据，这些数据不会在 **log** 窗口中显示。当工具出现问题时，需将 **err.log** 文件送回检查，**err.log** 文件会自动保存到默认路径下。

共享 查看

此电脑 > 文档 > simcomDemoLinker > log

名称	修改日期	类型	大小
autoLog - Tue Jan 16 09-15-42 2024.L...	2024/1/16 9:15	文本文档	2 KB
autoLog - Tue Jan 16 09-16-16 2024.L...	2024/1/16 9:17	文本文档	7 KB
autoLog - Tue Jan 16 09-17-05 2024.L...	2024/1/16 9:17	文本文档	3 KB
autoLog - Tue Jan 16 09-17-50 2024.L...	2024/1/16 9:18	文本文档	2 KB
autoLog - Tue Jan 16 09-18-05 2024.L...	2024/1/16 9:18	文本文档	7 KB
autoLog - Tue Jan 16 09-18-52 2024.L...	2024/1/16 9:20	文本文档	4 KB
autoLog - Tue Jan 16 09-20-23 2024.L...	2024/1/16 9:24	文本文档	10 KB
autoLog - Tue Jan 16 09-24-12 2024.L...	2024/1/16 9:24	文本文档	3 KB
autoLog - Tue Jan 16 09-24-54 2024.L...	2024/1/16 10:01	文本文档	278 KB
err.log	2024/1/16 9:25	文本文档	1 KB

图 14 log 文件默认保存路径

用户可在 **log** 窗口中点击右键，可以对 **log** 进行清除和查询。

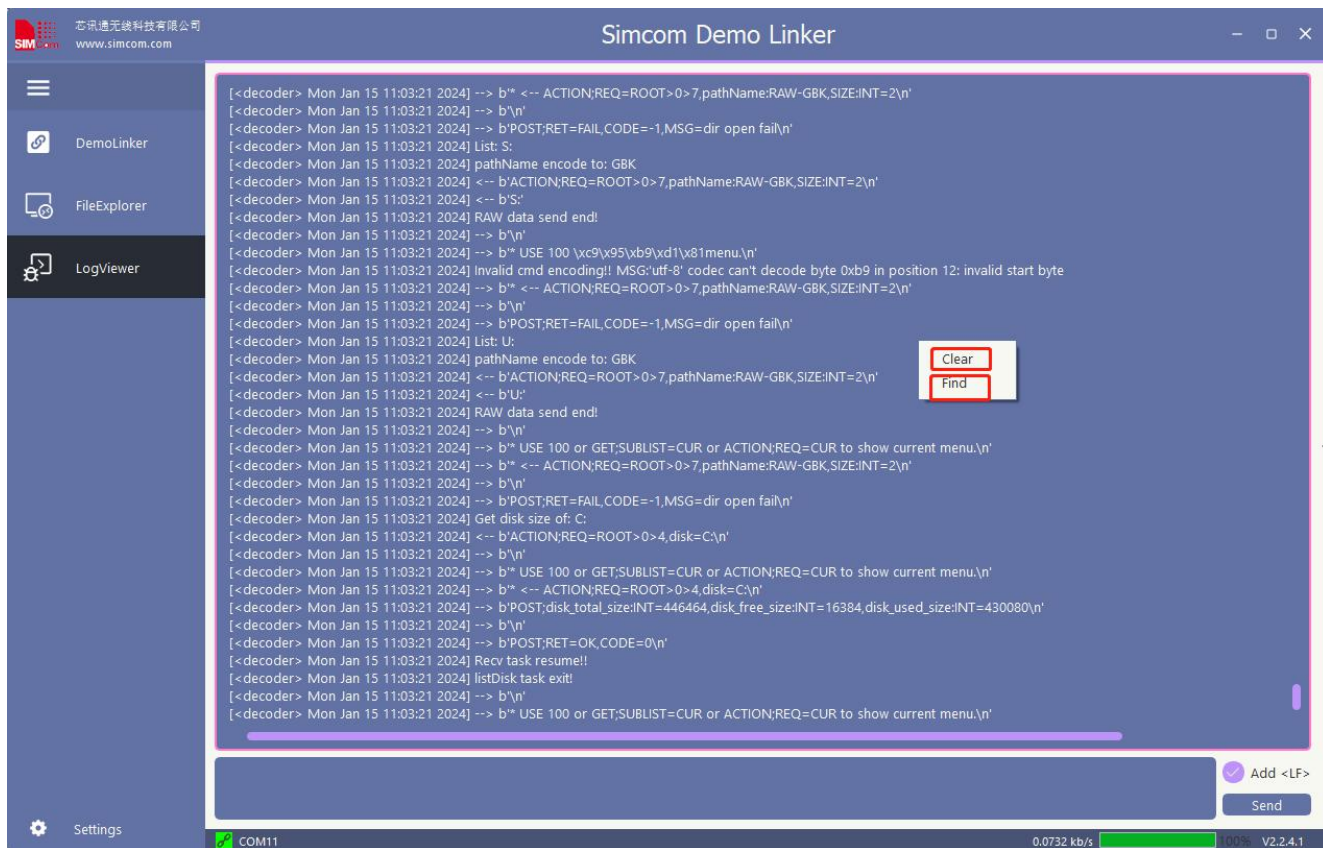


图 15 log 操作

用户可以选择【find】,输入要查询的内容, 点击【Find next】挨个查询。可勾选以下操作:

- | | |
|--------------------|-------|
| Regular expression | 正则表达式 |
| Backward | 向上查询 |
| Case sensitively | 区分大小写 |
| Whole words | 整个单词 |
| Loopback | 循环 |

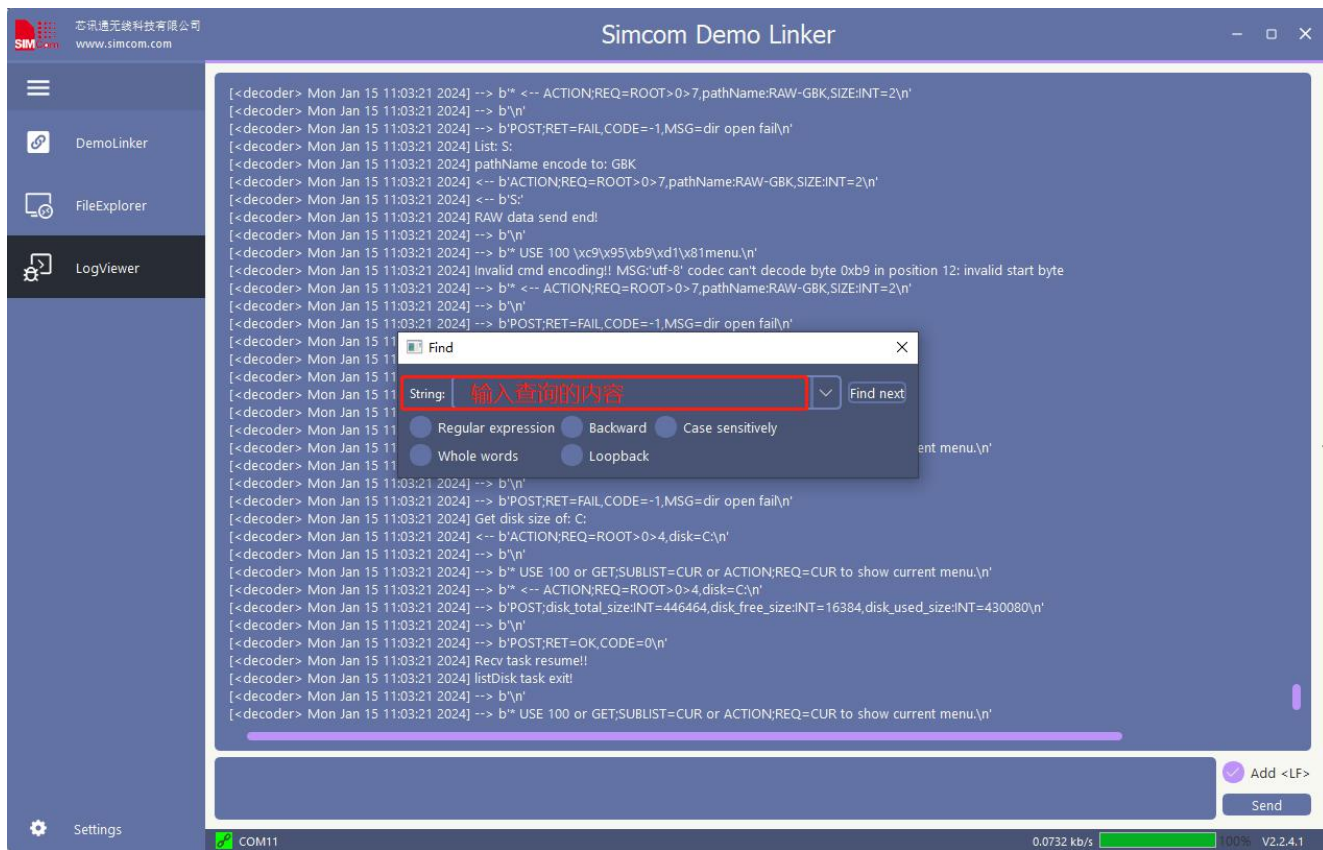


图 16 log 查询

3 DEMO 介绍与使用

A76xx 系列的二次开发 demo 采用了菜单(UI 界面)的形式,用户可以通过向串口输入参数来运行对应的 demo 程序。这种方式能够将 demo 的功能更直观的展现给用户,使用户快速了解 API 的使用方法。

3.1 DEMO UI 界面介绍

步骤 1: 打开串口的 Enhanced 口。

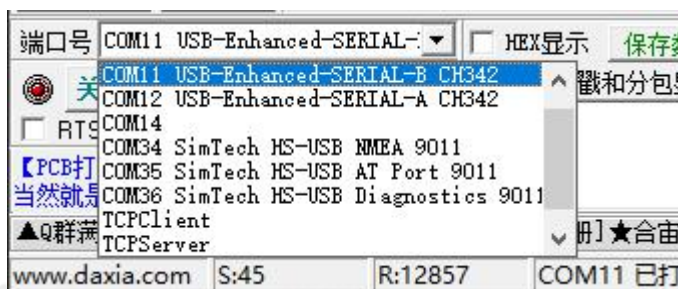


图 17 Enhanced 口

步骤 2: 模块上电开机,即会出现 UI 界面。



图 18 demo 界面

用户可以根据不同的选项来运行不同的 demo 程序。下面以 file system demo 为例,讲解如何运行具体的 demo 应用程序。

NOTE

当开启 Demo UI 展示后, Enhanced 口输入的所有信息都会被认为是对 demo UI 的操作。

3.2 DEMO 操作方法

从界面可以看到, file system 的 demo 是第 0 选项, 我们在 Enhanced 中输入 0 后(也可以通过指令进行交互, 具体实现查看示例), 就切换到 file system demo 的界面。

```
* <- 0
POST;SUBLIST:LIST-RAW, SIZE:INT=358, TYPE=MENU, PATH=ROOT>file system
*****
0. Write file      || 1. Read file      || 2. Delete file
3. Rename file    || 4. Get disk size   || 5. Make directory
6. Remove directory || 7. List file      || 8. Format
*****
POST;RET=OK, CODE=0
* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu.
```

图 19 file system demo 界面

目前, file system 主要提供以上 9 个功能的 demo 展示。用户可以任意执行自己想要的功能。这里以选项 7 (查询列表文件) 为例, 在该界面输入 7 后, 出现如下图的提示:
提示输入目录路径。

```
* <- 7
* pathName: Support disks: C:
GET;pathName:RAW-GBK
```

图 20 输入目录路径

输入路径 C:, 会列出该路径下的所有文件信息。

```
* <- C:
POST;name:RAW-GBK, SIZE:INT=6, type:STR=FILE, size:INT=126024
_1.pydPOST;name:RAW-GBK, SIZE:INT=8, type:STR=FILE, size:INT=86088
_bz2.pydPOST;name:RAW-GBK, SIZE:INT=12, type:STR=FILE, size:INT=46152
_hashlib.pydPOST;name:RAW-GBK, SIZE:INT=3, type:STR=DIR
newPOST;name:RAW-GBK, SIZE:INT=8, type:STR=FILE, size:INT=0
test.txtPOST;dir_count:INT=1, file_count:INT=4
POST;RET=OK, CODE=0
* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu.
```

图 21 查询结果列表

4 交互示例

以文件读写为例展示交互过程。（标红的为输入）

4.1 获取菜单

①严格指令交互模式

```

```C
* <-- GET;SUBLIST=ROOT //获取根节点菜单
POST;SUBLIST:LIST-RAW,SIZE:INT=733,TYPE=MENU,PATH=ROOT //对 GET 请求的应答

0.file system || 1.tcpip || 2.simcom cJSON
3.UART || 4.FOTA || 5.LOCAL_APP_UPDATER
6.WTD || 7.rtc || 8.ntp
9.SSL || 10.htp || 11.ping
12.WIFI scan || 13.HTTPS || 14.ftps
15.network || 16.SPI || 17.Extern NorFlash
18.Extern nandFlash || 19.I2C || 20.APP DOWNLOAD
21.GPIO || 22.FLASH || 23.PWM
24.PMU

POST;RET=OK,CODE=0 //输出的结果与错误码

* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu. //展示当前菜单命令提示

* <-- GET;SUBLIST=ROOT>0 //选择根节点下编号为 0 子节点的菜单
POST;SUBLIST:LIST-RAW,SIZE:INT=358,TYPE=MENU,PATH=ROOT>file system //对 GET 请求的应答

0.Write file || 1.Read file || 2.Delete file
3.Rename file || 4.Get disk size || 5.Make directory
6.Remove directory || 7.List file || 8.Format

POST;RET=OK,CODE=0 //输出的结果与错误码

* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu. //展示当前菜单命令提示

* <-- GET;SUBLIST=ROOT>0>0 //选择根节点下编号为 0 菜单下的编号为 0 的功能
POST;SUBLIST:LIST-RAW,SIZE:INT=277,TYPE=FUNC,PATH=ROOT>file system>Write file

```

\*\*\*\*\*

\* pathName: Support disks: C:. e.g: C:/test.txt  
 \* RAWSizeLink: fileSize  
 pathName:RAW-GBK || fileSize:INT || DATA:RAW

\*\*\*\*\*

POST;RET=OK,CODE=0 //输出的结果与错误码

\* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu. //展示当前菜单命令提示

\* <-- GET;SUBLIST=ROOT>0>8 //选择根节点下编号为0菜单下的编号为8的功能  
 POST;SUBLIST:LIST-RAW,SIZE:INT=178,TYPE=FUNC,PATH=ROOT>file system>Format

\*\*\*\*\*

disk:STR=C:|D:

\*\*\*\*\*

POST;RET=OK,CODE=0 //输出的结果与错误码

\* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu. //展示当前菜单命令提示

~~~

## ②兼容交互模式（进入文件系统模式）

~~C

\* <-- 99 //输入 99 查询根节点菜单列表

POST;SUBLIST:LIST-RAW,SIZE:INT=733,TYPE=MENU,PATH=ROOT //POST 应答

\*\*\*\*\*

|                     |          |                     |
|---------------------|----------|---------------------|
| 0.file system       | 1.tcpip  | 2.simcom cJSON      |
| 3.UART              | 4.FOTA   | 5.LOCAL_APP_UPDATER |
| 6.WTD               | 7.rtc    | 8.ntp               |
| 9.SSL               | 10.http  | 11.ping             |
| 12.WIFI scan        | 13.HTTPS | 14.ftp              |
| 15.network          | 16.SPI   | 17.Extern NorFlash  |
| 18.Extern nandFlash | 19.I2C   | 20.APP DOWNLOAD     |
| 21.GPIO             | 22.FLASH | 23.PWM              |
| 24.PMU              |          |                     |

\*\*\*\*\*

POST;RET=OK,CODE=0 //输出的结果与错误码

\* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu. //展示当前菜单命令提示

\* <-- 0 //直接输入编号 0 进入文件系统

POST;SUBLIST:LIST-RAW,SIZE:INT=358,TYPE=MENU,PATH=ROOT>file system //POST 应答

\*\*\*\*\*

```
0.Write file || 1.Read file || 2.Delete file
3.Rename file || 4.Get disk size || 5.Make directory
6.Remove directory || 7.List file || 8.Format
```

\*\*\*\*\*

POST;RET=OK,CODE=0

//输出的结果与错误码

\* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu. //展示当前菜单命令提示

## 4.2 列出所有文件

```C

* <-- ACTION;REQ=ROOT>0>7 //选择根节点下编号为 0 菜单下的编号为 7 的功能

* pathName: Support disks: C: //提示支持的路径

GET;pathName:RAW-GBK

* <-- POST;pathName=C: //输入要查询的路径

POST;name:RAW-GBK,SIZE:INT=6,type:STR=FILE,size:INT=126024 //查询结果

_1.pydPOST;name:RAW-GBK,SIZE:INT=8,type:STR=FILE,size:INT=86088

_bz2.pydPOST;name:RAW-GBK,SIZE:INT=12,type:STR=FILE,size:INT=46152

_hashlib.pydPOST;name:RAW-GBK,SIZE:INT=3,type:STR=DIR

newPOST;name:RAW-GBK,SIZE:INT=8,type:STR=FILE,size:INT=0

test.txtPOST;dir_count:INT=1,file_count:INT=4

POST;RET=OK,CODE=0

//输出的结果与错误码

* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu. //展示当前菜单命令提示

4.3 发送文件

严格指令交互模式，多参传入

```C

\* <-- ACTION;REQ=ROOT>0>0 //选择根节点下编号为 0 菜单下的编号为 0 的功能

\* pathName: Support disks: C:.. e.g: C:/test.txt

GET;pathName:RAW-GBK,fileSize:INT

\* <-- POST;pathName=C:/test.bin,fileSize=3 //输入文件的路径名与文件大小

GET;fileSize:INT

\* <-- 120 //输入延时时长

GET;DATA:RAW,SIZE:INT=120

POST;RET=OK,CODE=0

//输出的结果与错误码

\* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu. //展示当前菜单命令提示

严格指令交互模式，单参传入

```C

* <-- ACTION;REQ=ROOT>0>0

//选择根节点下编号为 0 菜单下的编号为 0 的功能

* pathName: Support disks: C:. e.g: C:/test.txt

GET;pathName:RAW-GBK,fileSize:INT

* <-- POST;pathName=C:/test.bin

//输入文件路径名

GET;fileSize:INT

* <-- POST;fileSize=3

//输入文件大小

GET;fileSize:INT

* <-- 6

//输入延时时长

GET;DATA:RAW,SIZE:INT=6

POST;RET=OK,CODE=0

//输出的结果与错误码

* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu. //展示当前菜单命令提示

<--ACTION;REQ=ROOT>0>0

可做如下替代;

<--ACTION;REQ=ROOT

-><输出菜单列表>

<--ACTION;REQ=0

-><输出菜单列表>

<--ACTION;REQ=0

~~~

\* <-- ACTION;REQ=ROOT>0>0

//选择根节点下编号为 0 菜单下的编号为 0 的功能

\* pathName: Support disks: C:. e.g: C:/test.txt

GET;pathName:RAW-GBK,fileSize:INT

\* <-- POST;pathName=C:/test.bin,fileSize=3

//输入文件的路径名与文件大小

已知参数列表的情况下，可做如下替换

\* <--ACTION;REQ=ROOT>0>0,pathname=C:/test.bin,fileSize=3

~~~

兼容交互模式，多参传入（结合获取菜单操作，已进入文件系统模式）

```C

\* <-- 0

//选择写入文件

\* pathName: Support disks: C:. e.g: C:/test.txt

GET;pathName:RAW-GBK,fileSize:INT

\* <-- C:/test.bin,2

//输入文件路径名与大小

GET;DATA:RAW,SIZE:INT=2

[输入数据]

POST;RET=OK,CODE=0

\* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu.

## 4.4 读取文件

严格指令交互模式，多参传入

```C

* <-- ACTION;REQ=ROOT>0>1 //选择读取文件

* pathName: Support disks: C:

GET;pathName:RAW-GBK

* <-- POST;pathName=C:/test.bin //输入文件路径名

POST;DATA:RAW,SIZE:INT=3 //文件类型与大小

10 //文件数据

POST;RET=OK,CODE=0

* USE 100 or GET;SUBLIST=CUR or ACTION;REQ=CUR to show current menu.