

测试文件上传系统使用说明

目录

测试文件上传系统使用说明

目录

1. 总览

2. 主要功能

2.1 上传待分析的数据文件 (.txt)

2.1.1 通过选择的方式上传

2.1.1.1 通过选择单个或多个文件的方式上传数据

2.1.1.2 通过选择整个文件夹的方式上传数据

2.1.2 通过输入的方式上传

2.2 左侧边栏功能介绍

2.2.1 前往主页

2.2.2 管理分析脚本

上传脚本操作

删除脚本操作

2.2.3 显示说明文档

2.2.4 关于页面

2.2.5 系统更新

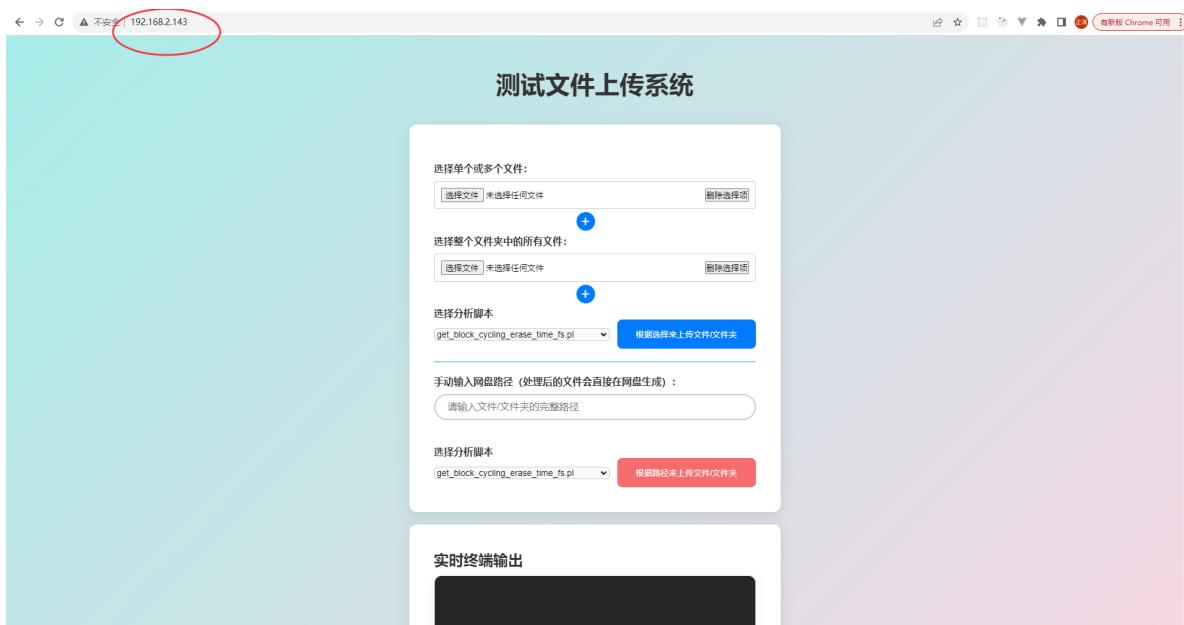
2.3 右侧边栏功能介绍

2.3.1 上传文件历史

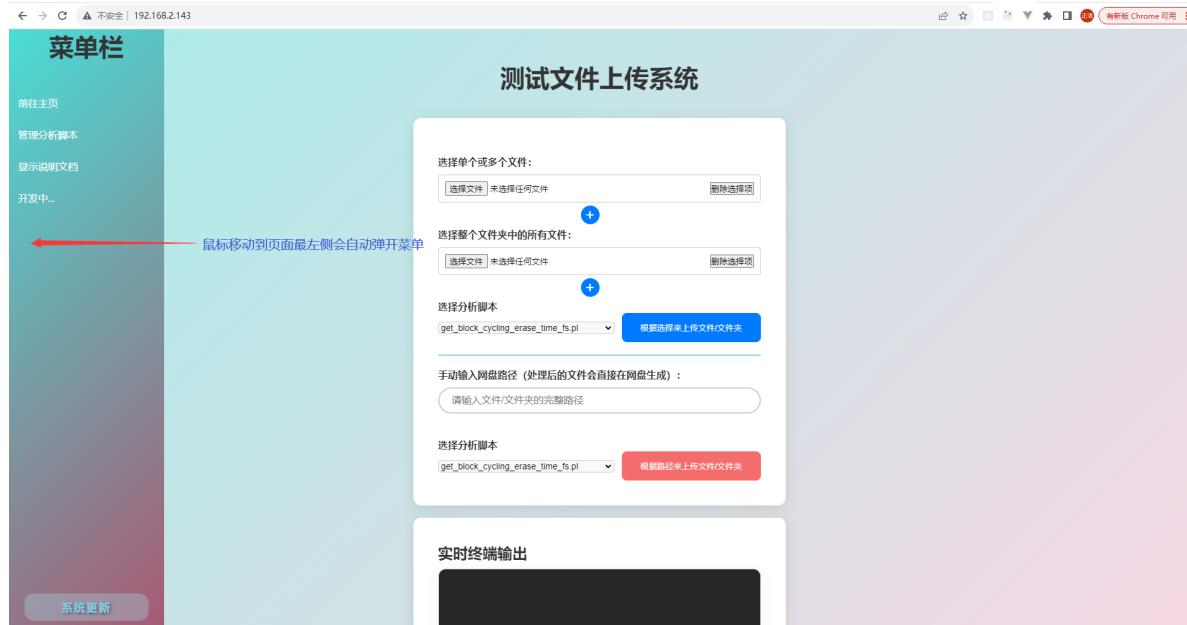
1. 总览

本 webApp 旨在为用户提供一个方便的平台，用于上传测试生成的数据文件，通过我们预制好的分析脚本（包括 Python 和 Perl）进行分析，并将分析结果作为 Excel 文件放在页面上供用户下载

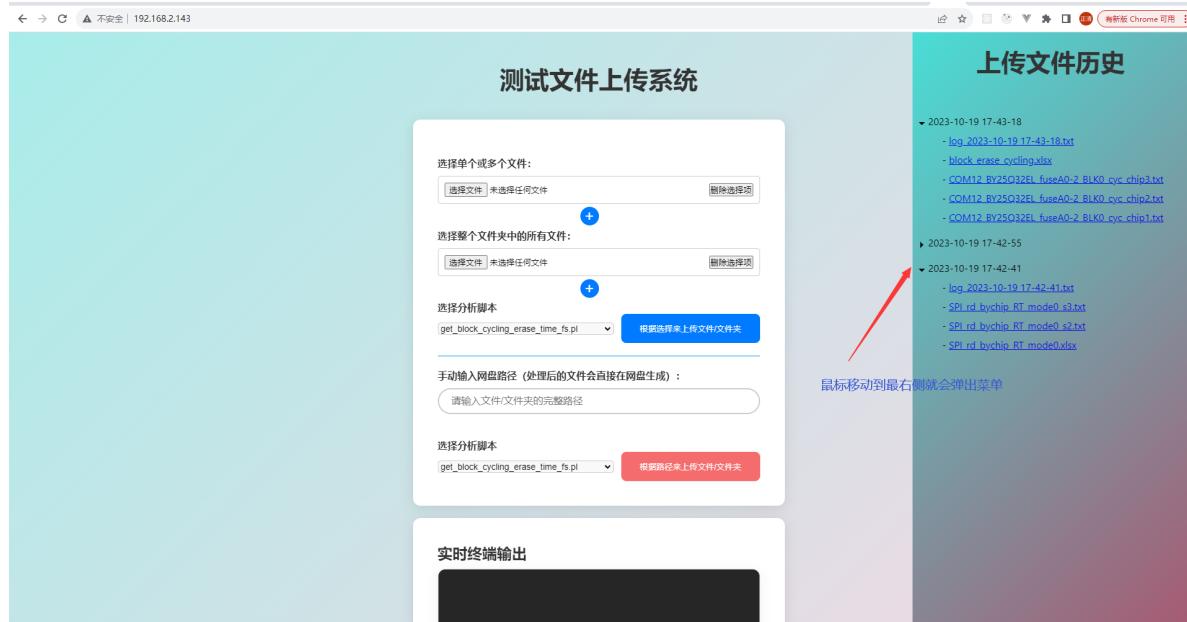
在浏览器地址栏（WindowsXP系统建议使用360极速浏览器）输入局域网网址<http://192.168.2.143>即可进入测试文件上传系统主页面：



鼠标放在页面最左侧就会出现左侧边栏，这个边栏提供了前往主页面、管理我们分析数据脚本、显示此说明文档，系统更新（专供开发人员后期新增功能的使用，用于拉取最新的本地仓库代码）的选项：



鼠标放在页面最右侧就会出现右侧边栏，这个边栏可以看到我们上传过的待分析的数据文件、处理数据的日志、以及经过分析脚本所生成的excel文件，它们会按照日期进行分类，我们随时可以点击超链接进行下载：



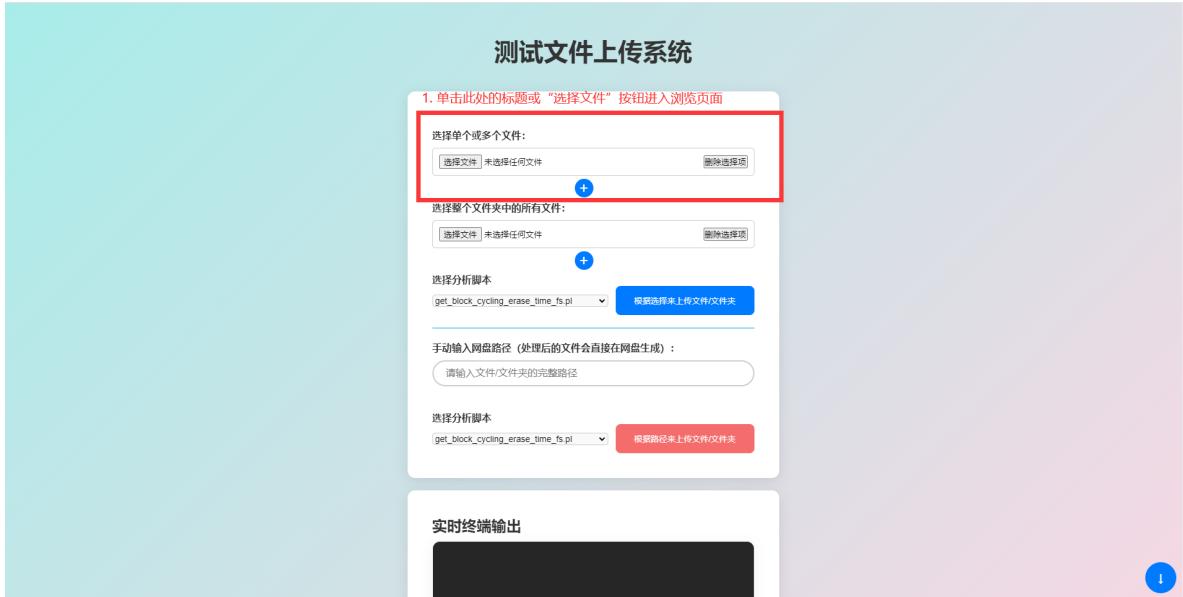
2. 主要功能

2.1 上传待分析的数据文件 (.txt)

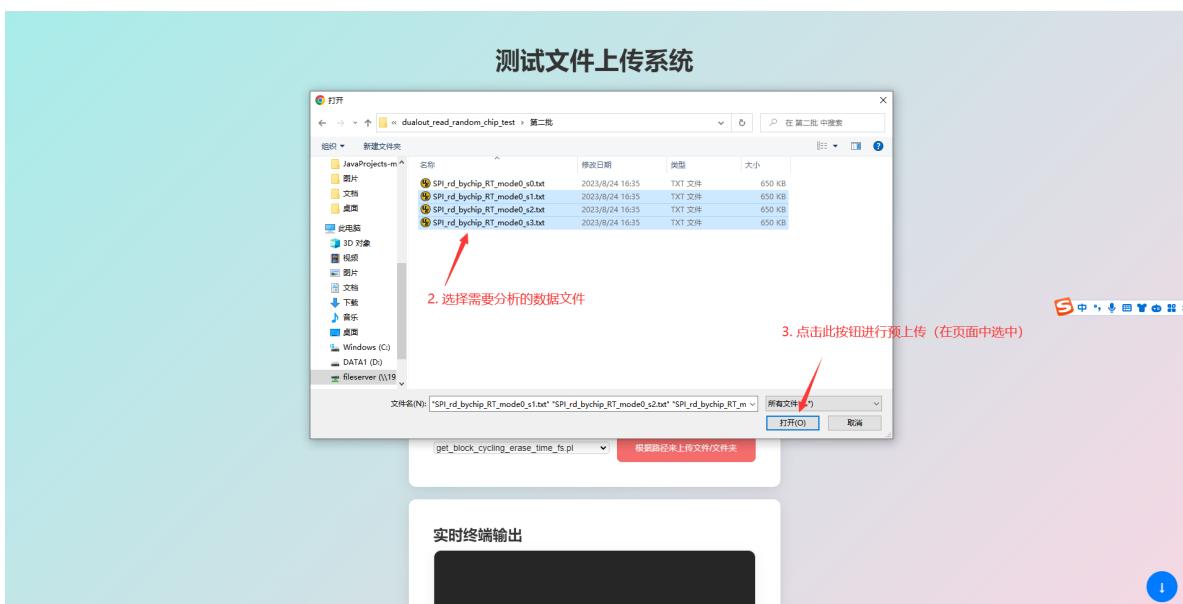
2.1.1 通过选择的方式上传

2.1.1.1 通过选择单个或多个文件的方式上传数据

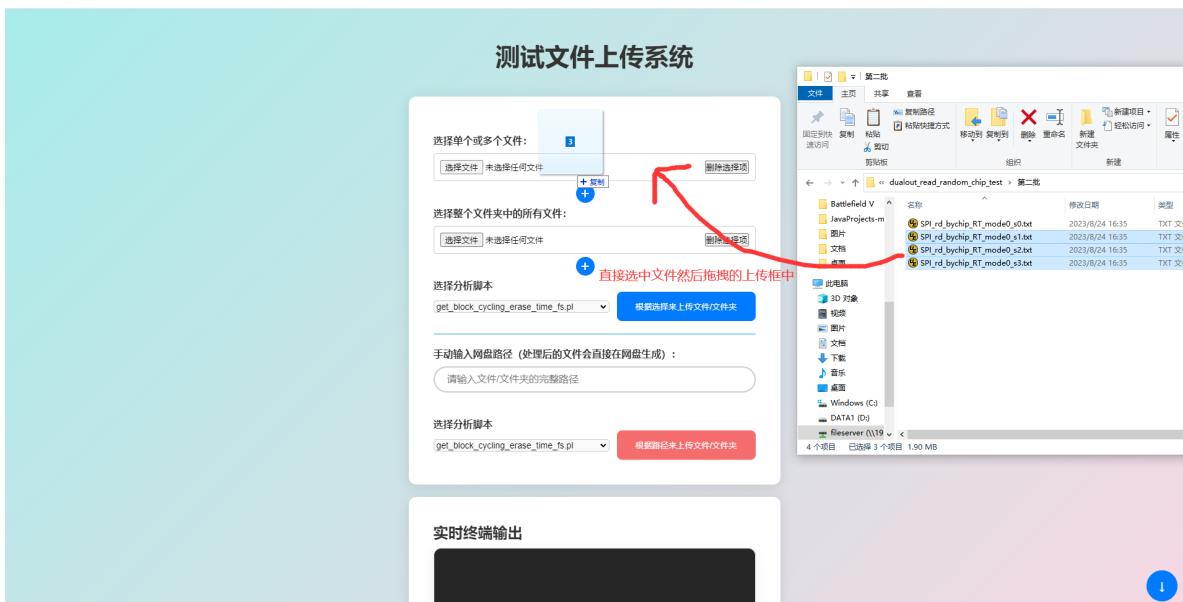
我们可以单击第一个“选择单个或多个文件”来上传我们的数据：



然后选择我们需要分析的数据文件：



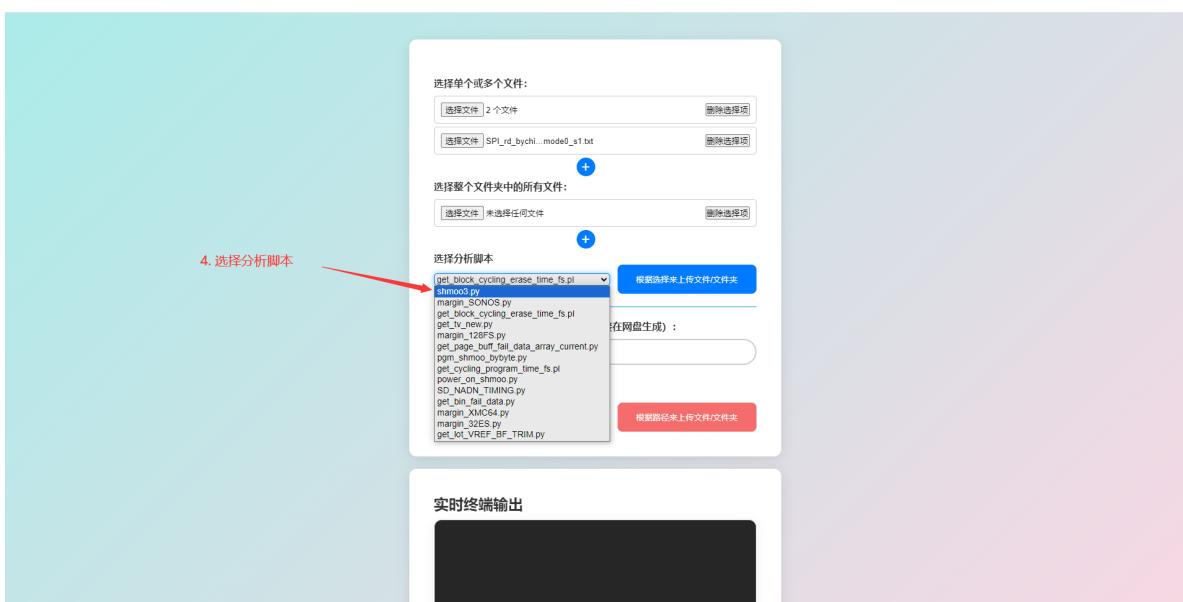
此外，我们也可以事先打开数据文件所在的文件夹，然后选中文件后通过拖拽的方式直接进行预上传：



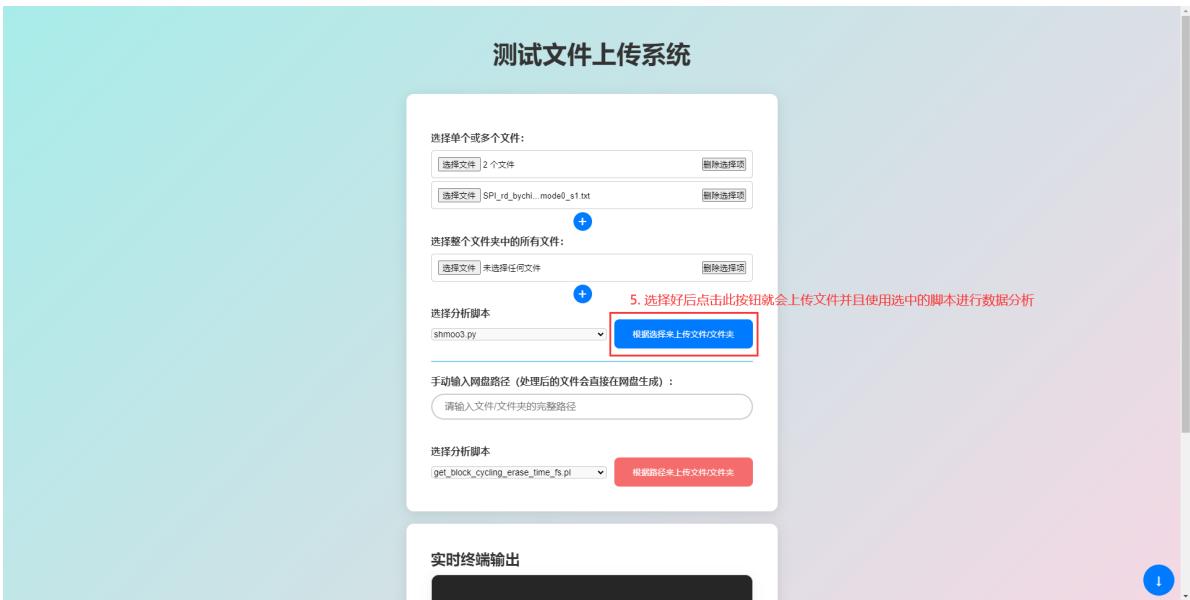
tips: 考虑到特殊情况，如果用户要同时上传不同文件夹里的不同文件，比如我们要同时上传A文件夹里的“a.txt”和B文件夹里的“b1.txt”、“b2.txt”，我们就可以使用到这里面的“+”按钮来解决这种情况：



然后选择我们的分析脚本：



最后，点击旁边的上传按钮：



脚本运行成功之后，页面上会罗列出一系列信息：

这里是经过分析之后生成的excel文件、终端日志，可以点击链接进行下载

这里是被成功分析的数据文件名

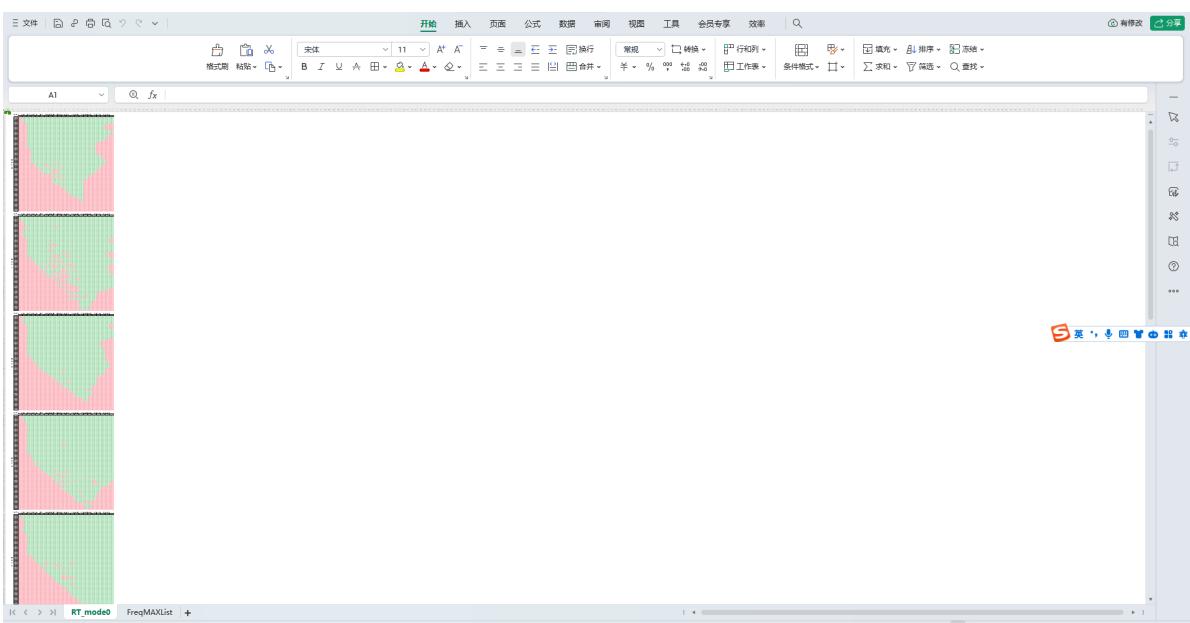
这里是flask应用的后台输出，可以在这里监测操作的处理状态

这里是上传的原文件、经过分析生成的excel表格以及记录终端输出的日志文件

上传文件历史

- 2023-10-19 18-38-15
 - log_2023-10-19 18-38-15.txt
 - SPI_rd_bychip_RT_mode0_s1.txt
 - SPI_rd_bychip_RT_mode0_s2.txt
 - SPI_rd_bychip_RT_mode0_s3.txt
 - SPI_rd_bychip_RT_mode0.xlsx
- 2023-10-19 17-43-18
 - 2023-10-19 17-42-55
 - 2023-10-19 17-42-41

预览生成的excel文件：

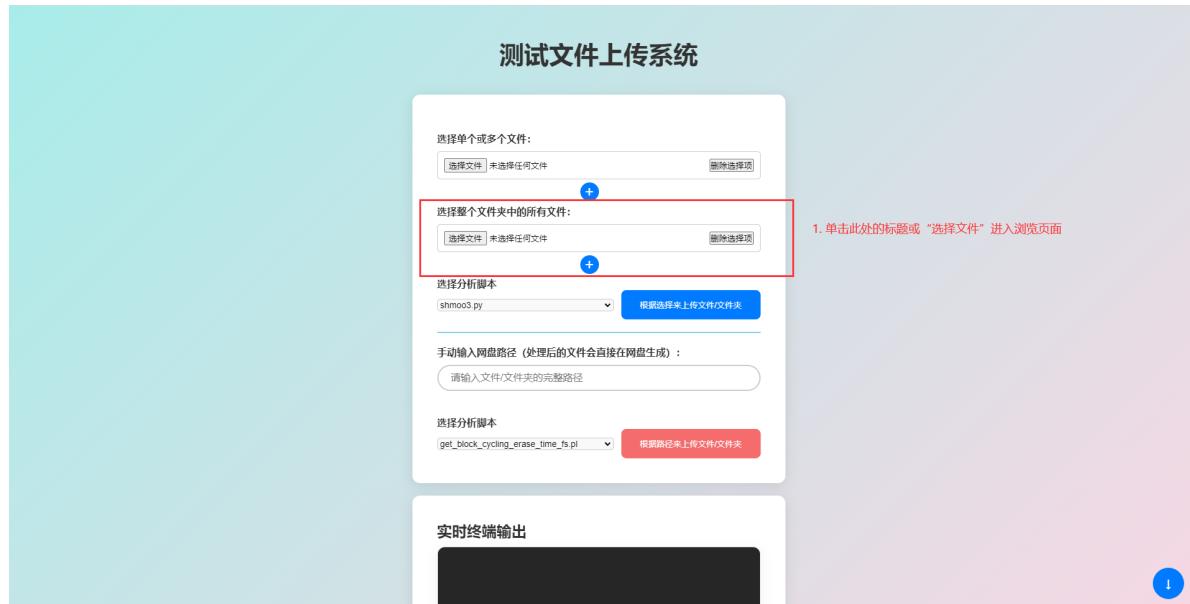


至此，我们根据选择上传单个或多个文件的分析数据流程已完毕

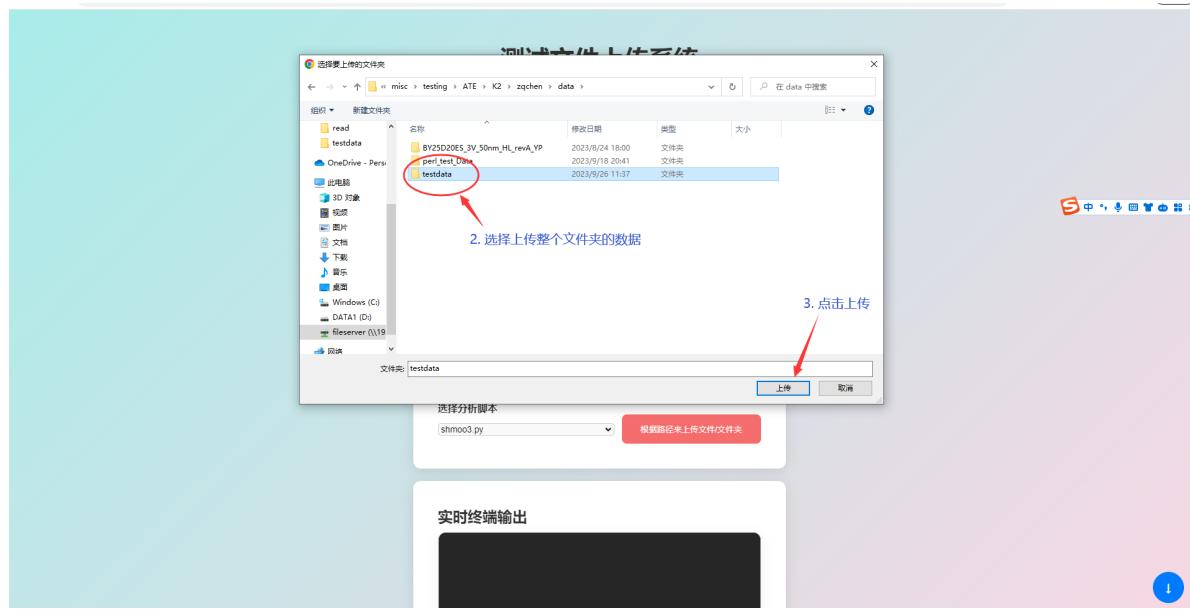
特别提醒：这个上传文件的方式同时支持上传**电脑本地**的文件和**网盘中的**文件，但是由于浏览器隐私政策的限制，这种方式在使用分析脚本后生成的Excel文件**不会生成到源路径之中**，我们需要自行点击超链接进行下载

2.1.1.2 通过选择整个文件夹的方式上传数据

除了第一种方法外，我们还可以单击第二个“选择整个文件夹中的所有文件”来上传我们的数据（假设文件夹中有4个txt文件，这里我们使用即将被perl脚本分析的数据）：



然后选择对应的文件夹，这里会上传文件夹里的所有文件



此外，我们同样也可以事先打开数据文件所在的文件夹，然后选中文件夹后通过拖拽的方式直接进行预上传：

The screenshot shows the 'Test File Upload System' interface. On the left, there are three main sections: '选择单个或多个文件:' (Select single or multiple files), '选择整个文件夹中的所有文件:' (Select all files in the entire folder), and '选择分析脚本' (Select analysis script). Each section has a '选择文件' (Select file) button, a preview area, and a '删除选项' (Delete option) button. A red arrow points from the 'Delete option' button in the 'Select all files in the entire folder' section to a 'Delete option' button in a Windows File Explorer window on the right. The File Explorer shows a folder structure with several files and folders, including 'read1(50%)', 'read1(100%)', and 'testdata'. Below the interface is a note: '直接拖拽到文件框中也可以实现相同操作' (You can also achieve the same operation by dragging and dropping directly into the file box).

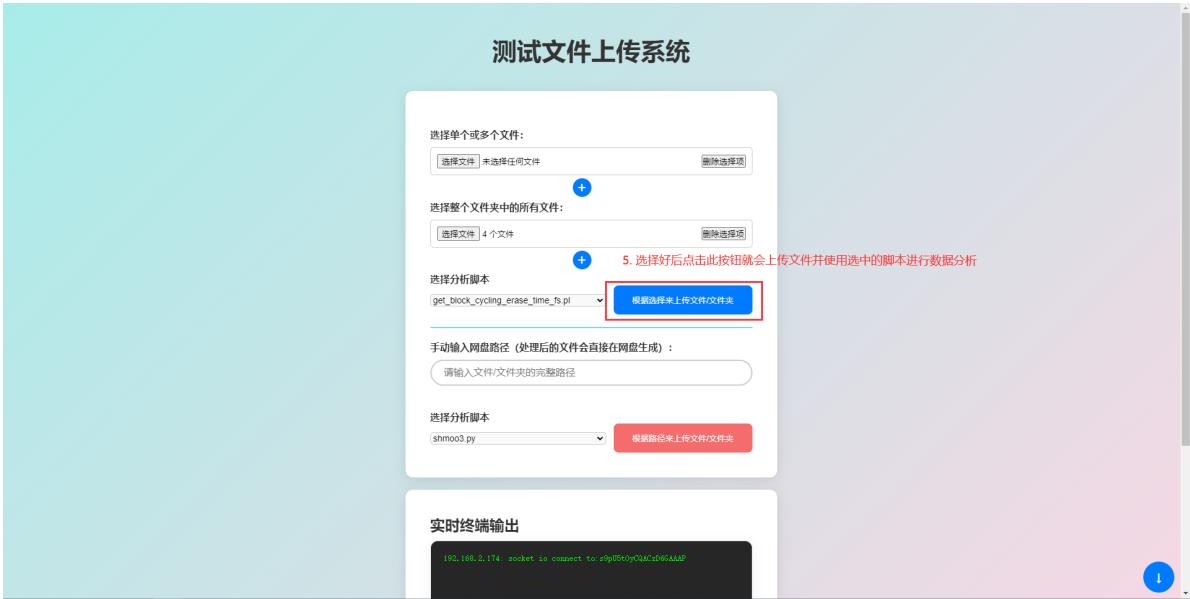
tips：同上面一样，考虑到特殊情况，如果用户要同时上传不同文件夹，比如我们要同时上传A文件夹和B文件夹（都代指里面的所有文件），我们就可以使用到这里面的“+”按钮来解决这种情况：

The screenshot shows the same 'Test File Upload System' interface as above, but with additional red annotations. A red box highlights the '+' button in the 'Select all files in the entire folder' section, with the text '1. 每点击一次 "+" 号，就会多出现一个文件框' (Every time you click the "+" sign, another file box will appear). Another red box highlights the 'Delete option' button in the same section, with the text '2. 这个文件框可以预上传文件夹A里的所有文件' (This file box can pre-upload all files in folder A). A third red box highlights the 'Delete option' button in the 'Select single or multiple files:' section, with the text '3. 这个文件框可以预上传文件夹B里的所有文件' (This file box can pre-upload all files in folder B). A fourth red box highlights the 'Delete option' button in the 'Select analysis script' section, with the text '4. “删除选项”按钮可以清空文件框的预上传状态，让文件框回到初始状态' (The "Delete option" button can clear the pre-upload state of the file box, returning it to its initial state).

然后选择我们的分析脚本：

The screenshot shows the 'Test File Upload System' interface with the 'Select analysis script' section highlighted. A red arrow points to the dropdown menu where 'shmoo3.py' is selected. The dropdown list includes 'shmoo3.py', 'margin_SNOMOS.py', 'get_nv_nvm_cycling_erase_time_ts.py', 'margin_128S.py', 'margin_nv.py', 'margin_page_buf_data_array_current.py', 'pgm_mmap_bytify.py', 'get_cycling_erase_time_fs.py', 'power_on_shmoo.py', 'SD_NAND_TIMING.py', 'get_bin_fail_data.py', 'margin_2G5S.py', and 'get_lot_VREF_EF_TRIM.py'. Below the interface is a note: '根据路径来上传文件/文件夹' (Upload files/folders according to the path).

最后，点击旁边的上传按钮：



脚本运行成功之后，页面上会罗列出一系列信息：

这里是经过分析之后生成的excel文件、终端日志，可以点击链接进行下载

这里是被成功分析的数据文件名

这里是flask应用的后台输出，可以在这里监测操作的处理状态

这里是上传的源文件、经过分析生成的excel表格以及记录终端输出的日志文件，它们是以文件夹结构来分类的

预览生成的excel文件：

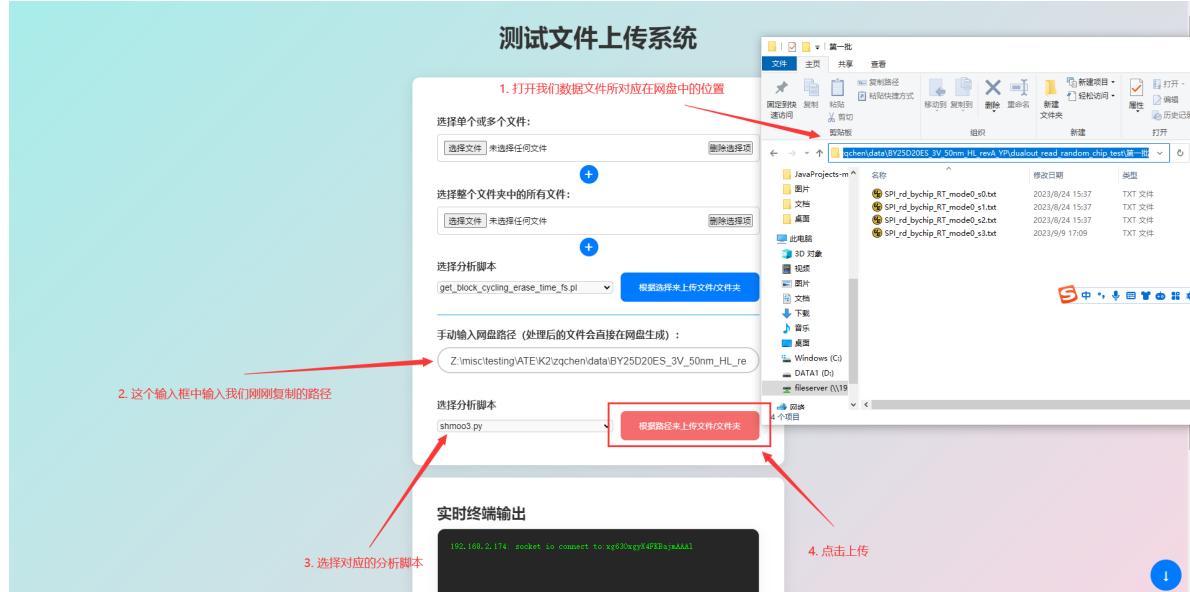
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
1		#1	#2	#3	#4																				
2	cyc1	100	150	134	128																				
3	cyc2	114	132	110	110																				
4	cyc3	112	140	112	108																				
5	cyc4	114	130	110	114																				
6	cyc5	118	136	110	112																				
7	cyc6	110	130	108	108																				
8	cyc7	114	138	108	112																				
9	cyc8	114	134	108	112																				
10	cyc9	114	130	108	112																				
11	cyc10	114	126	106	112																				
12	cyc11	112	130	106	106																				
13	cyc12	112	134	106	108																				
14	cyc13	112	128	106	108																				
15	cyc14	112	134	104	110																				
16	cyc15	110	130	104	104																				
17	cyc16	110	134	104	110																				
18	cyc17	110	134	104	108																				
19	cyc18	110	134	102	108																				
20	cyc19	110	134	104	108																				
21	cyc20	110	126	102	102																				
22	cyc21	108	130	104	108																				
23	cyc22	108	130	102	106																				
24	cyc23	108	130	102	106																				
25	cyc24	108	130	100	106																				
26	cyc25	108	130	100	106																				
27	cyc26	108	122	100	106																				
28	cyc27	108	130	100	106																				
29	cyc28	108	120	100	106																				
30	cyc29	108	130	100	106																				
31	cyc30	106	128	98	104																				
32	cyc31	106	130	98	104																				
33	cyc32	106	128	98	104																				
34	cyc33	106	128	98	104																				
35	cyc34	106	128	98	104																				
36	cyc35	104	128	98	104																				
37	cyc36	104	128	98	104																				
38	cyc37	104	126	98	104																				
39	cyc38	104	128	96	104																				
40	cyc39	104	128	96	104																				
41	cyc40	104	126	96	102																				
42	cyc41	104	126	96	102																				
43	cyc42	104	118	96	102																				

至此，我们根据选择上传单个或多个文件夹的分析数据流程已完毕

特别提醒：这个上传文件的方式同时支持上传**电脑本地**的文件和**网盘中的**文件，但是由于浏览器隐私政策的限制，这种方式在使用分析脚本后生成的Excel文件**不会生成到源路径之中**，我们需要自行点击超链接进行下载

2.1.2 通过输入的方式上传

除了前面两种提到的方法外，我们可以直接输入我们**数据文件**在网盘（fileserver）中的路径来上传文件：



脚本运行成功之后，页面上会罗列出一系列信息，都是和上面的差不多：

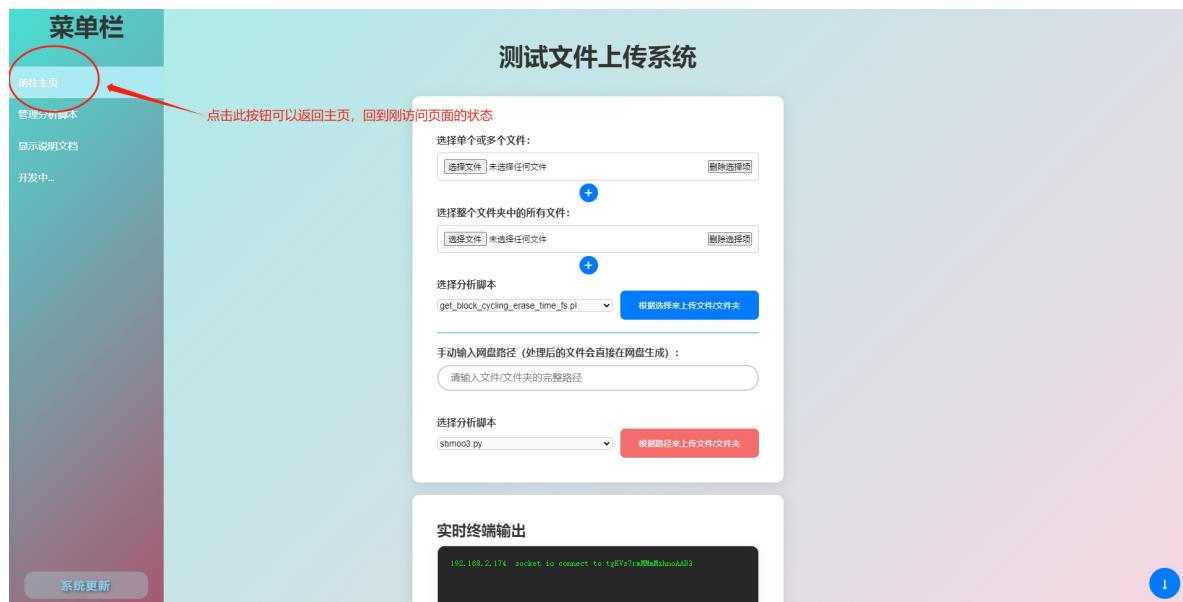
The screenshot shows the analysis results page. On the left, there's a 'Downloaded files and directory' section with 'SPI_rd_bychip_RT_mode0.xlsx' and 'log_2023-10-19 21-21-58.txt'. Below it is an 'Upload successful files or file夹' section with '第一批'. At the bottom is a 'Real-time terminal output' section showing log entries. On the right, a large 'Upload file history' sidebar lists upload logs from October 19, 2023, such as '2023-10-19 21-21-58' and '2023-10-19 21-18-14'.

特别注意：以路径方式上传网盘数据文件，生成的excel文件会直接存放在数据所对应的网盘路径中，不需要进行额外操作，所以更推荐这种方式来进行数据文件分析，当然，这些文件依然可以在网页的主页面以及上传文件历史中随时找到

2.2 左侧边栏功能介绍

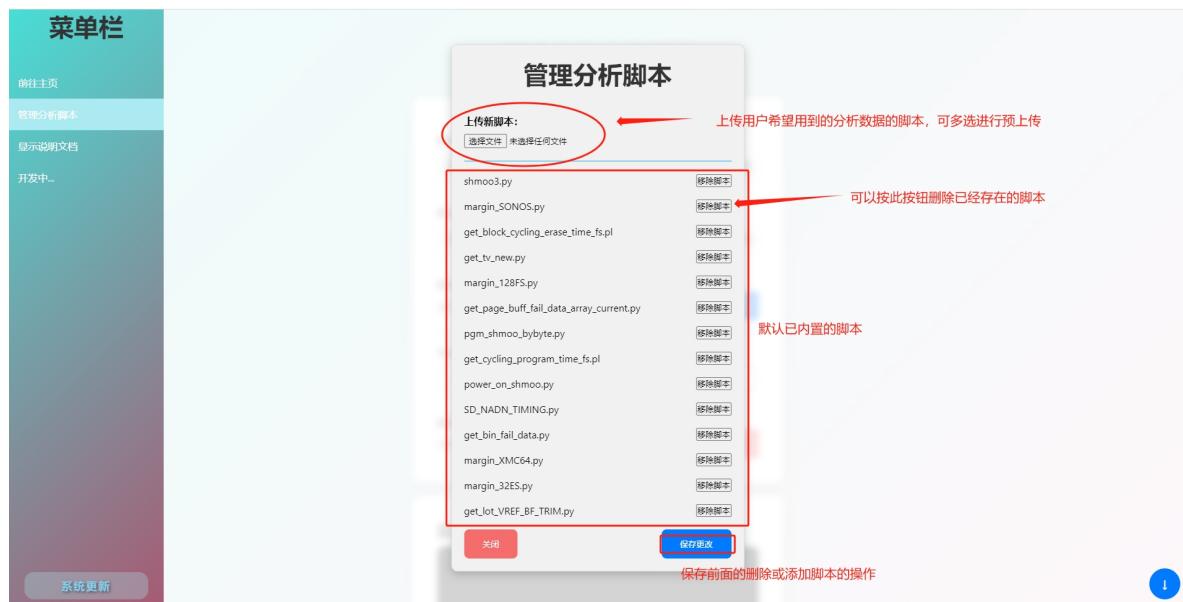
2.2.1 前往主页

我们可以把鼠标放在页面左边来显示左边栏，在弹出的边栏里点击“前往主页”选项，类似于浏览器的刷新按钮



2.2.2 管理分析脚本

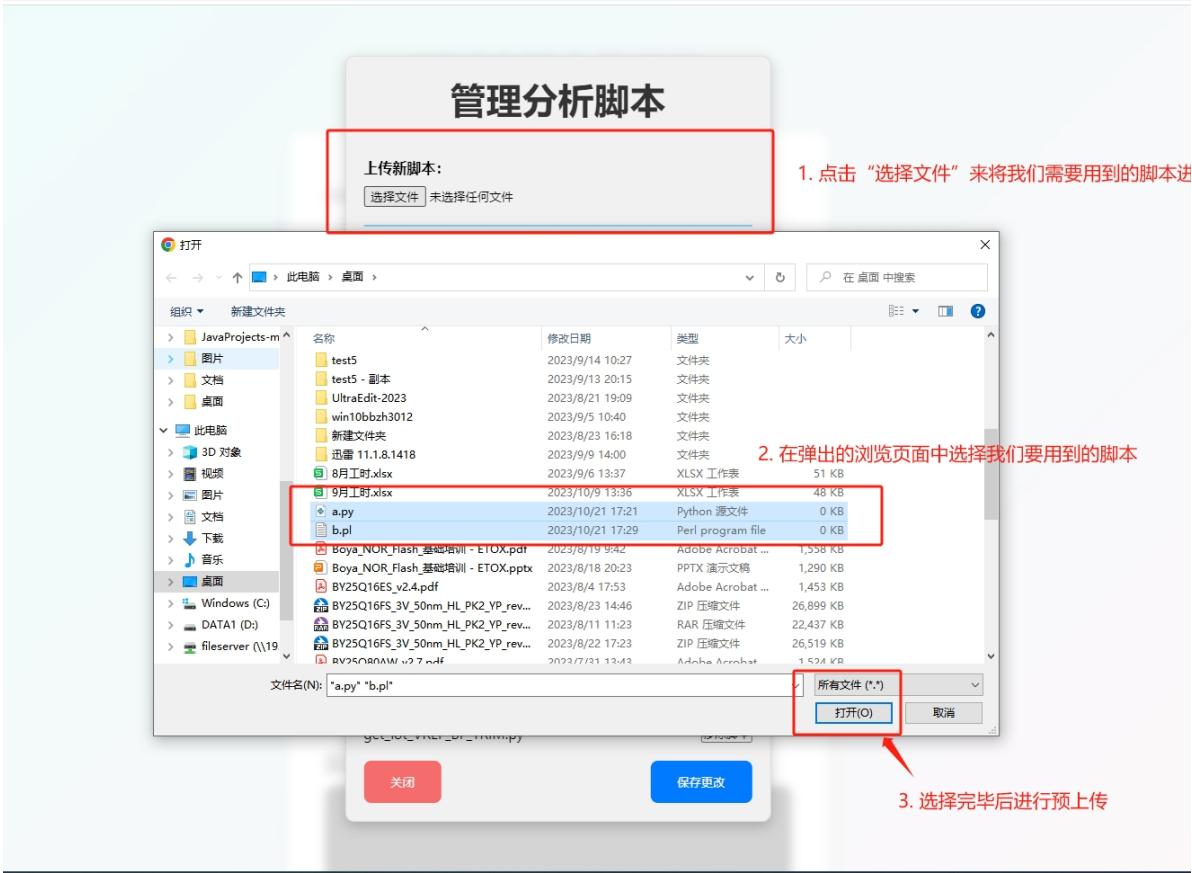
我们可以把鼠标放在页面左边来显示左边栏，在弹出的边栏点击“管理分析脚本”选项。



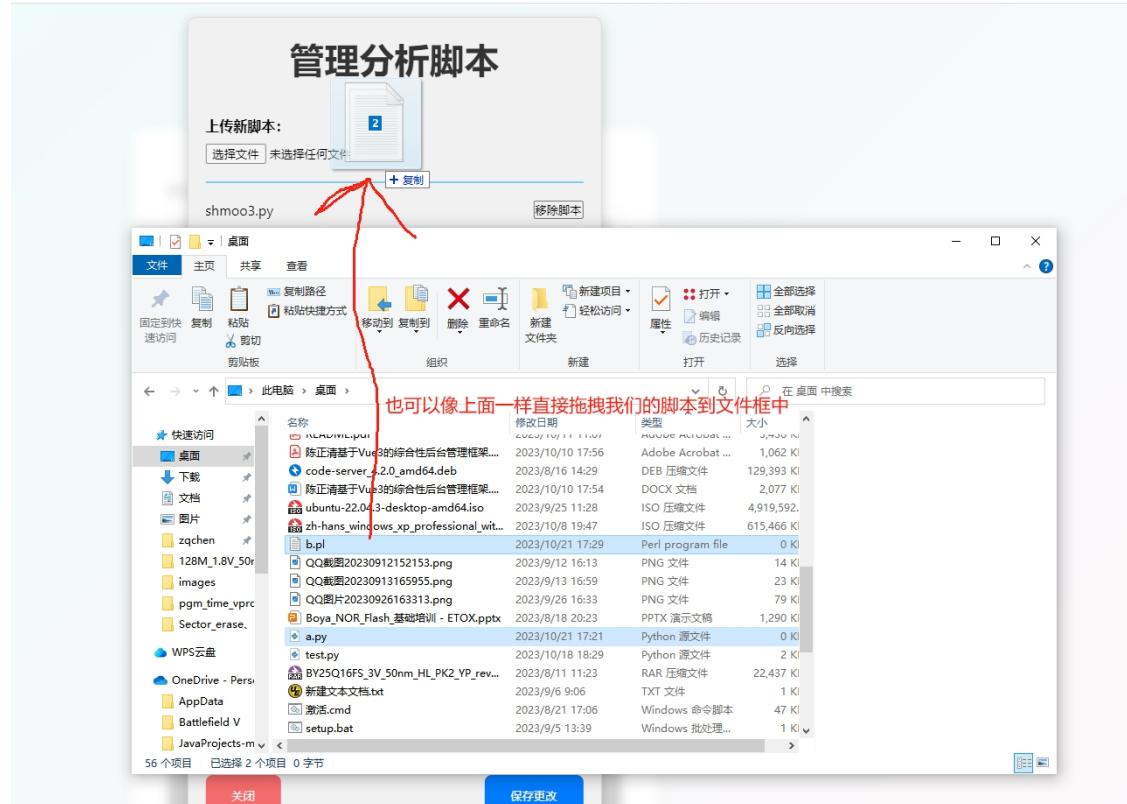
上传脚本操作

如果我现在需要上传a.py和b.pl这两个脚本来让页面支持调用，具体操作流程如下：

- 点击“选择文件”来预上传我们的新脚本：



这里也可以像上面一样将我们的脚本拖进来

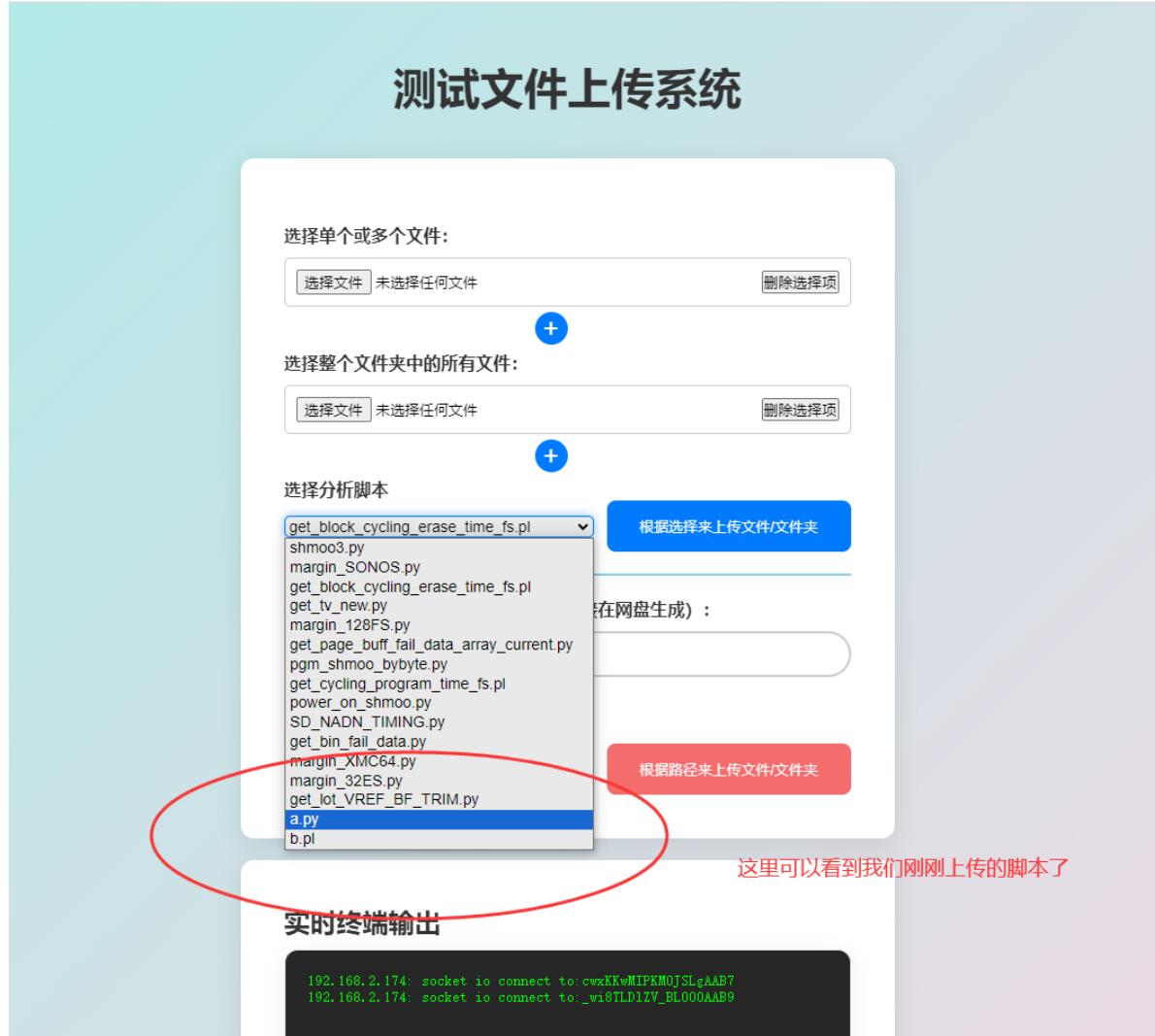


- 然后点击“保存更改”按钮



- 随后就发现可以在页面中选择我们刚刚上传的脚本了

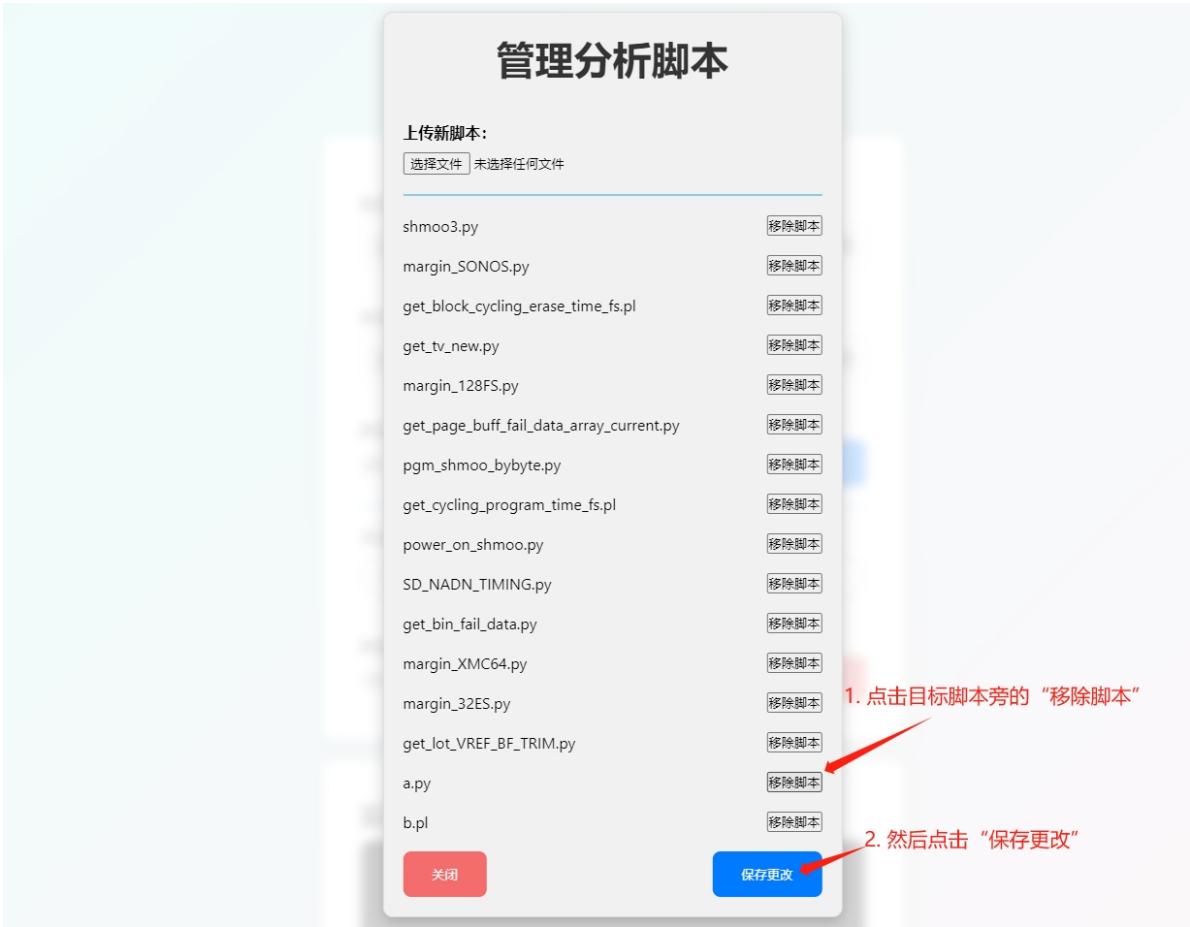
测试文件上传系统



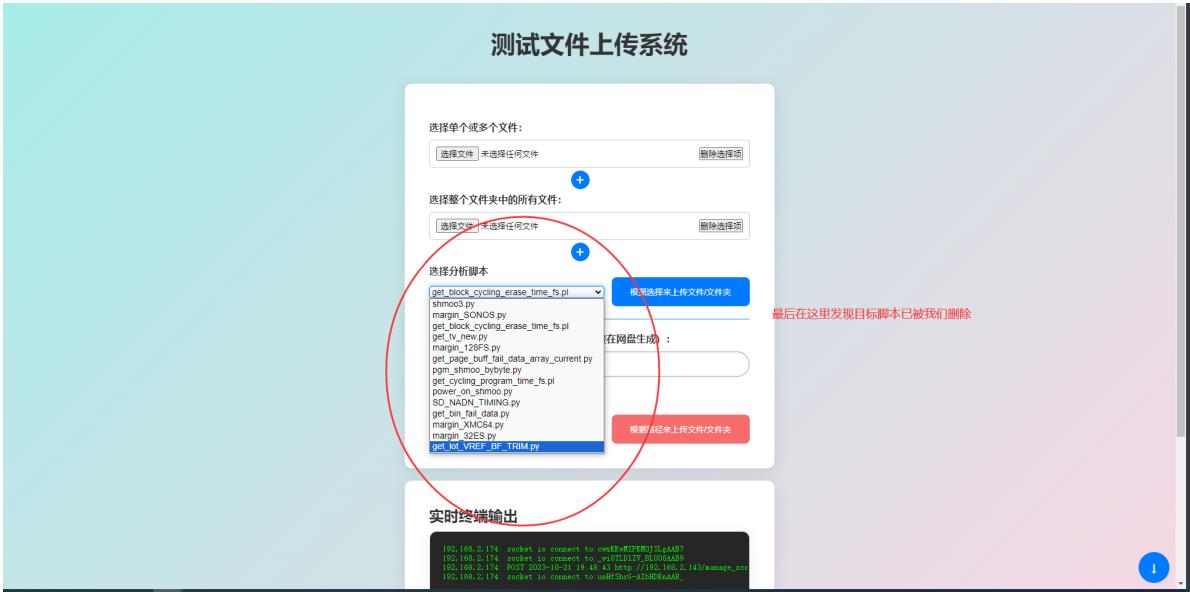
删除脚本操作

如果我现在需要删除在页面显示的a.py和b.pl这两个脚本，具体操作流程如下：

- 点击目标脚本旁边的“移除脚本”来删除



- 在弹出的确认框中选择“确定”
- 随后在页面中看到我们删除脚本操作已经生效



提示：由于脚本直接放在了APP的特定文件夹里，所有访问APP的设备都会共享这个文件夹里的脚本，所以任何设备对脚本的管理都会影响其他设备的访问效果，为了避免误删操作，请谨慎使用此功能，后续可能会加入类似鉴权或管理权限机制，敬请期待！

2.2.3 显示说明文档

我们可以点击侧边栏中的“显示说明文档”选项来弹出我们测试文件上传系统的文档介绍，并且这里也供给了下载文档和放大缩小文档等功能



2.2.4 关于页面

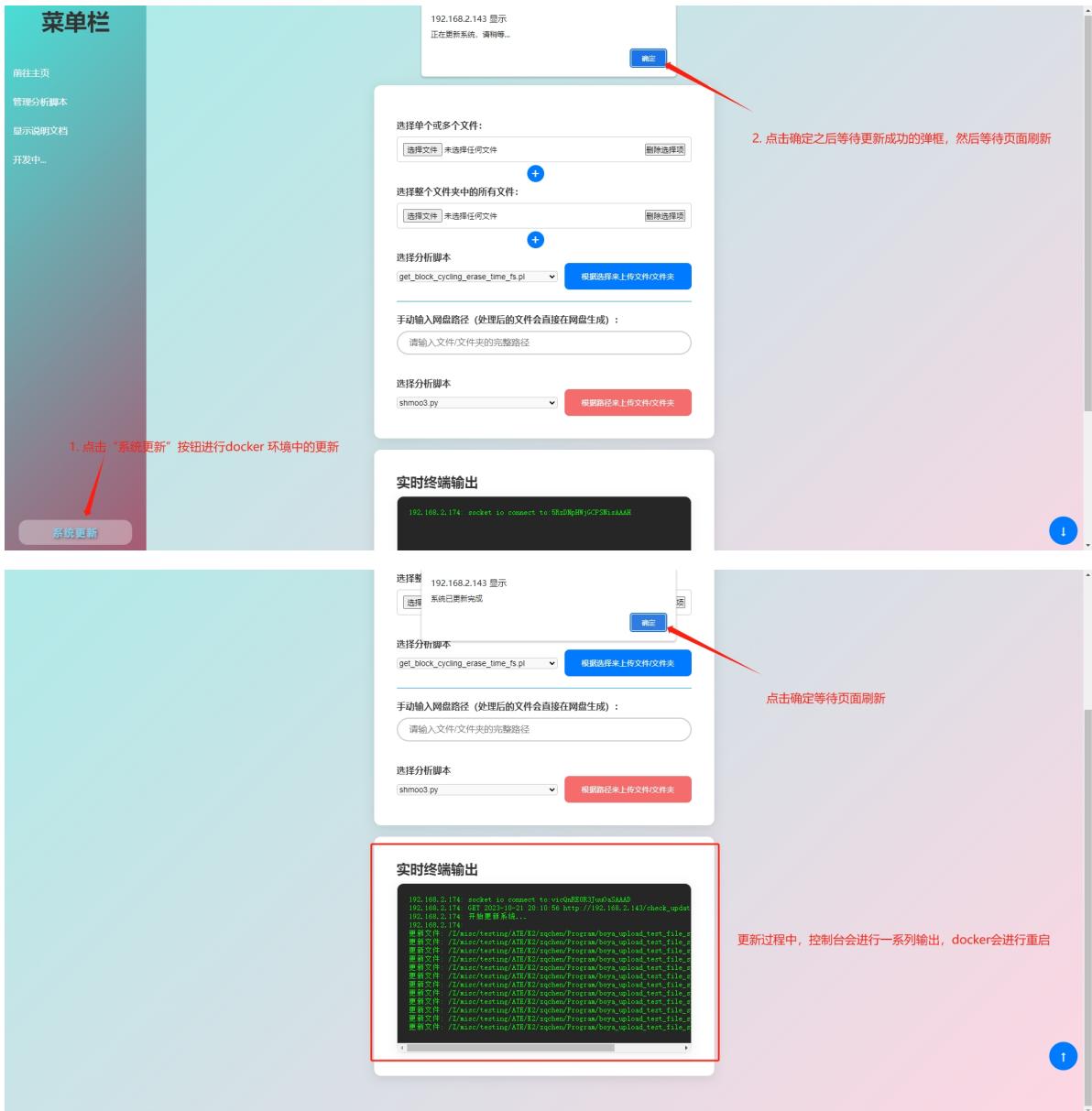
点击左侧边栏中的“关于”条目可以展示系统的关于信息以及版本更新信息：



2.2.5 系统更新

这个功能主要是开放给开发人员使用的，因为我们的APP运行在ubuntu server的docker环境中，docker环境和我们现在开发的windows环境会有些许不同，为了能让我们在windows的更改能够直接作用在docker中，省去打包镜像所浪费的时间，我们制作了这个功能，项目的源代码仓库路径为：

Z:\misc\testing\ATE\K2\zqchen\Program\boyau_upload_test_file_sys



2.3 右侧边栏功能介绍

2.3.1 上传文件历史

把鼠标放在页面右侧，右侧边栏就会出现，这个边栏主要是按时间降序分类了上传的数据文件、生成的Excel以及记录控制台的log文件，不管我们是通过选择上传的文件，还是通过路径上传的网盘文件，上面提及的文件我们都可以通过这个超链接进行下载。

这个文件历史也是有清理机制的：

- ① 每10分钟只保留最新的15个文件上传记录
 - ② 每天凌晨4点删除所有的文件上传记录

下载生成的文件以及日志:

block_erase_cycling.xlsx
log_2023-10-21 21-13-06.txt

删除输出文件并返回到主页

上传成功的文件或文件夹 (更多上传以及生成文件的历史可在右侧查看):

testdata

实时终端输出

```
192.168.2.174 socket to connect to SelectY-BY25Q2EL-7044A9P  
192.168.2.174 POST 2023-10-21 21 13 06 http://192.168.2.145/upload_by  
192.168.2.174 ----- 上将文件的完整路径为: I:\Users\testing  
192.168.2.174 原始操作: 读取并上传文件, 生成并返回上载数据 - 生成的excel文件  
192.168.2.174 请在文件中, 请确保参数...  
192.168.2.174 转到linux的路径为: /U/mmc/testing/ATK/K7/syphon/status  
192.168.2.174 读取并上传文件: get_block_cycling_erase_time.xlspl, 请等  
192.168.2.174 等片刻  
COM112_BY25Q2EL.fuseA0-2.BLKO_cyc_chip1.txt  
COM112_BY25Q2EL.fuseA0-2.BLKO_cyc_chip2.txt  
COM112_BY25Q2EL.fuseA0-2.BLKO_cyc_chip3.txt  
COM112_BY25Q2EL.fuseA0-2.BLKO_cyc_chip4.txt  
192.168.2.174 socket to connect to b37ydb15WQJextLCAAN  
192.168.2.174 socket to connect to b37ydb15WQJextLCAAN
```

上传文件历史

- 2023-10-21 21-13-06
 - testdata
 - block_erase_cycling.xlsx
 - COM112_BY25Q2EL_fuseA0-2_BLKO_cyc_chip1.txt
 - COM112_BY25Q2EL_fuseA0-2_BLKO_cyc_chip2.txt
 - COM112_BY25Q2EL_fuseA0-2_BLKO_cyc_chip3.txt
 - COM112_BY25Q2EL_fuseA0-2_BLKO_cyc_chip4.txt
 - log_2023-10-21 21-13-06.txt
- 2023-10-21 21-12-26
 - log_2023-10-21 21-12-03.txt
 - SPI_rd_bychip_RT_mode0_s3.txt
 - SPI_rd_bychip_RT_mode0_s2.txt
 - SPI_rd_bychip_RT_mode0_s1.txt
 - SPI_rd_bychip_RT_mode0_s0.txt
 - SPI_rd_bychip_RT_mode0.xlsx
- 2023-10-21 21-12-03

这里是按时间降序来排序的所有文件, 包括上传的原始数据文件、生成的excel文件以及控制台的输出log文件