

SPK 使用指南

提示：用户可以前往【哔哩哔哩】搜索“SPK”相关内容即可观看 SPK 详细的功能演示和教学视频。

1 概述

Serial Port Kits（简称 SPK）是一款串口通讯软件。与市面上其他同类软件不同的是其设计目标：化繁为简，体验至上。SPK 并不追求繁多的功能，追求的是对每一个功能的用户体验和 UI 细节的打磨。功能的可用性及稳定性是 SPK 的基础，在此基础上，SPK 更加注重的是用户使用体验和视觉体验，例如：用户是否能快速使用软件；配置参数是否方便快捷；是否能智能的对接收的数据进行正确精准的换行；是否能根据输出 log 快速找到异常点以提高 debug 效率；软件运行速度是否够快；UI 界面是否干净简洁但又不失表达等等。

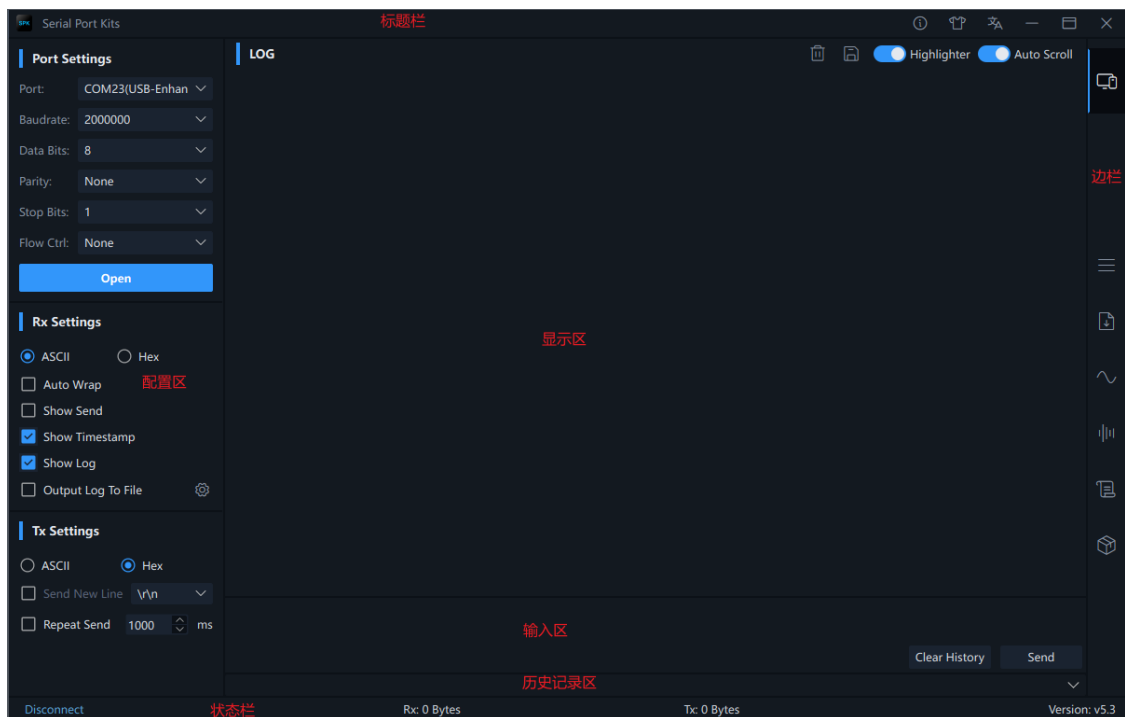
SPK 的 UI 是其亮点之一，采用了现代化的设计元素，但是为了保持极佳的性能，UI 元素的使用上保持了克制，也没有使用更耗性能的动画特性，在美观和性能上做了平衡。我想大多数人都梦寐以求一款颜值拉满又能高效 debug 的工具软件，SPK 也许是个不错的选择。

SPK 支持的功能如下：

1. 支持自动检测可用串口，无需要手动刷新
2. 支持串口参数配置，支持任意波特率配置
3. 支持 ASCII 和 Hex 收发模式
4. 支持自动添加换行
5. 支持回显发送的数据
6. 支持保存 log 到文件
7. 支持实时输出 log 到文件，对于 log 量大的情况非常有用
8. 支持时间戳显示
9. 支持数据高亮显示，在快速定位问题时非常有用
10. 支持 智能换行

11. 支持多帧发送和单帧发送，支持记录用户配置的多帧数据，重启软件自动加载
12. 支持记录历史发送数据，避免用户重新输入
13. 支持定时发送
14. 支持 7 种精美皮肤选择，无需要重启即可秒换肤。
15. 支持中英文
16. 支持长时间采集波形数据，实时绘制波形数据
17. 支持百万级波形数据绘制渲染依然流畅
18. 支持保存波形数据到 CSV 格式，支持 CSV 数据回放
19. 支持波形数据测量和统计学分析
20. 支持 6 种文件传输协议
21. 支持自动回连功能
22. 集成 Lua 脚本引擎，赋能任意协议解析和自动化测试，助力协议数据的多方位展示
23. 集成帧组装机，支持定义任意协议格式，快速生成协议帧
24. 自动检测软件新版本，安静的提示用户更新

2 软件功能详解



SPK 主界面

软件主界面分如下几个部分：

1. 标题栏
2. 配置区
3. 显示区
4. 输入区
5. 历史记录区
6. 状态栏
7. 边栏

2.1 标题栏



标题栏 提供了如下功能：

1. 点击左侧 logo 可以显示软件信息，如：软件版本，激活状态等



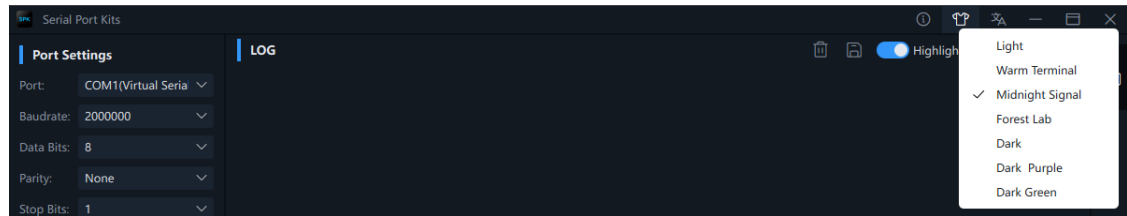
2. 点击 ⓘ 图标，将弹出菜单，提供了软件相关的使用手册



3. 点击 🎨 图标可以切换皮肤。SPK 提供了 7 种专业级皮肤供用户选择。

- 亮色主题
- 暖色终端主题
- 午夜信号主题

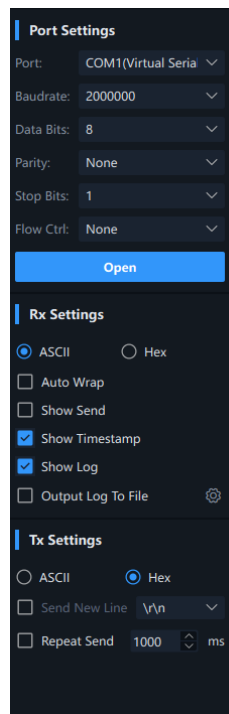
- 森林实验室主题
- 暗色主题
- 深紫主题
- 翡翠绿主题



4. 点击  图标，设置语言，SPK 支持中文和英文两种语言。



2.2 配置区



配置区分如下 3 个子区域：

- 串口设置区
- 接收设置区

- 发送设置区

2.2.1 串口设置区

串口设置区提供了串口设备相关参数的配置，例如：端口，波特率，数据位，奇偶校验位，停止位，流控等，以及串口设备的开关操作。

2.2.2 接收设置区

接收设置区用于配置如何处理接收的数据，SPK 提供了如下几种接收配置

1. 接收模式设置

可选 Hex 和 ASCII

2. 自动添加换行

SPK 在处理每一次接收数据后都会默认添加换行，适用于 Hex 接收模式

3. 显示发送

该功能使能后将在显示区显示每一条发送的数据

4. 显示时间戳

该功能使能以后将在显示区开头显示时间戳信息，时间精确到 ms。

```
[2026-08-22 18:25:10.662]R version: 1.2.0.fbb630a1
[2026-08-22 18:25:10.662]R Copyright(C) by GaoQiu
[2026-08-22 18:25:10.662]R *****
[2026-08-22 18:25:10.678]R [W][RCC]FT Info Load Successfully:
[2026-08-22 18:25:10.678]R   - Chip Version      : 0x80
[2026-08-22 18:25:10.678]R   - Chip FT Version   : 2
[2026-08-22 18:25:10.678]R   - Chip MAC Address  : 6C:13:00:0C:00:D0
[2026-08-22 18:25:10.678]R   - Chip Flash Size   : 1024 KB
[2026-08-22 18:25:10.678]R [I][track] System Start...
[2026-08-22 18:25:10.678]R [I][track] System Clock: 48000000 Hz
[2026-08-22 18:25:10.678]R [I][track] Vector Table Address: 0x00000000
[2026-08-22 18:25:10.678]R *****
[2026-08-22 18:25:10.678]R
[2026-08-22 18:25:10.678]R      Used Size      Total Size      Usage(%)
[2026-08-22 18:25:10.678]R FLASH:      150512 B          508 KB          28.9%
[2026-08-22 18:25:10.678]R RAM :        36376 B           64 KB          55.5%
[2026-08-22 18:25:10.678]R *****
[2026-08-22 18:25:10.678]R [I][nvs] nvs init start...
[2026-08-22 18:25:10.695]R [I][nvs] nvs init ok - ram entry:4, free_space_offset:432
[2026-08-22 18:25:10.761]R [I][DM_] sec:0, key:ble_hs/sec_0, age:2, peer addr:C2:3A:66:66:66:66
[2026-08-22 18:25:10.761]R [I][DM_] sec num:1, sec max size:1, newest aging counter:2
[2026-08-22 18:25:10.761]R [I][DM_] CCC num:0, ccc max size:0
[2026-08-22 18:25:10.761]R [W][LL_] LL PM: Enable
[2026-08-22 18:25:10.761]R [I][APP] Heap Free Size: 1100 B
[2026-08-22 18:25:10.761]R [I][APP] BB Free Size: 5244 B
[2026-08-22 18:25:10.761]R [I][APP] BD_ADDR: F5:01:6C:57:3B:BD
[2026-08-22 18:25:10.761]R [I][dfu_client] image address: 0x42000
[2026-08-22 18:25:10.761]R [W][LL_] LL_SetExtScanEnable:0
[2026-08-22 18:25:10.761]R [W][LL_] LL_SetExtScanEnable:1
[2026-08-22 18:25:10.761]R [I][APP] Scan Start...
```

5. 显示 log

该功能使能后，接收到的数据将在显示区显示出来，否则显示区不显示。

6. 输出 log 到文件

该功能使能后，接收到的数据将实时保存到文件，保存文件的路径和文件名称可通过点击该选项后的来设置。

NOTE:

1. 文件采用了追加模式，也即对已存在的数据不会删除，仅仅只是在后面追加新数据。
2. “输出 log 到文件”和“显示 log”功能是相互对立的，互不影响。也即显示 log 的同时也可以将其实时保存至文件。由于内存的原因，当显示区数据量太大时，内存不够，可能导致软件卡顿。在实际的 debug 过程中，如果需要长时间采集数据时，可以将“显示 log”功能关闭，仅使能“输出 log 到文件”功能，可以偶尔开启“显示 log”功能查看当前状态。这对长时间测试，但偶尔需要查看当前状态非常有用。

2.2.3 发送设置区

发送设置区用于配置如何处理发送的数据，SPK 提供了如下几种发送配置

1. 发送模式

可以选择 Hex 和 ASCII。

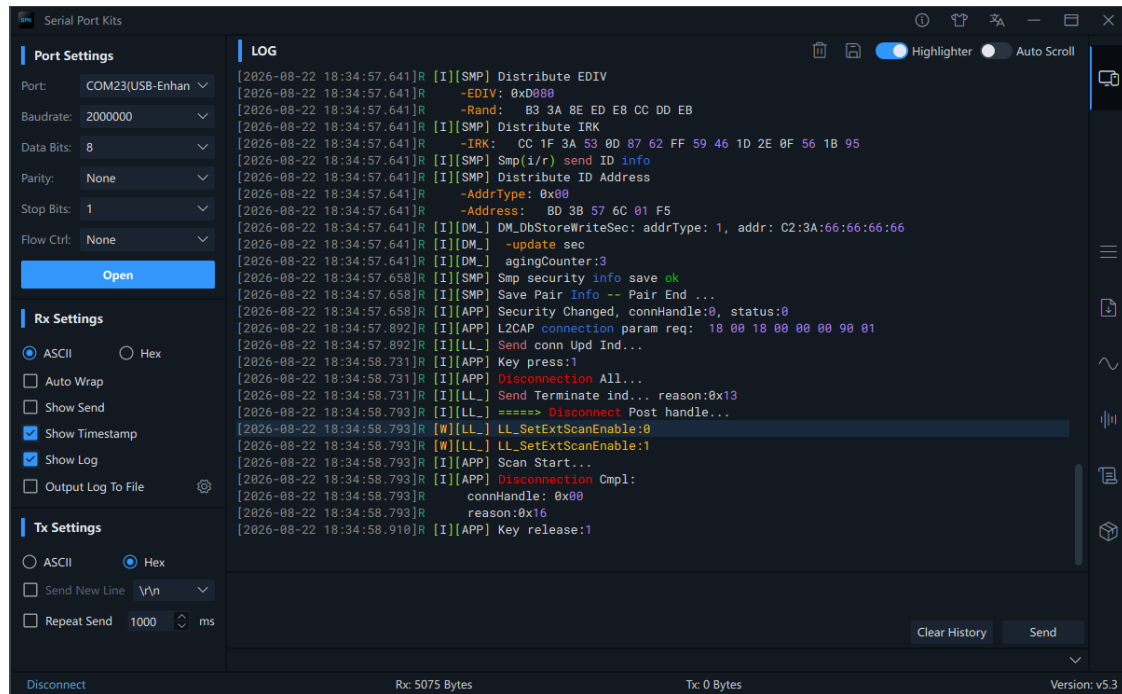
2. 发送新行

使能该选项后，在每一个发送数据帧的后面自动添加换行，换行符可以自行选择。**该功能只适用于 ASCII 模式。**

3. 定时发送

该功能用于定时发送数据，定时时间可配。

2.3 显示区



显示区提供了如下功能:

1. 显示接收的数据
2. 点击  删除显示区内容
3. 点击  保存显示区内容到文件
4. 点击  高亮显示区的内容

NOTE:

该功能使能后, 显示区中的内容将按照内建规则对一些特征数据进行着色, 这个在实际 *debug* 中非常有用, 能快速定位找到异常点。实战中亲自体验, 效果很好。

对于高速通信的场景不要开启高亮器, 可能会影响性能。因此高速通信时, 先将高亮器关闭, 等数据采集完以后, 再开启高亮器查看数据。

5. 点击  自动滚动到显示区最底部, 并保持
6. 通过鼠标右键菜单可以设置显示区的字体, 也可以清除当前显示内容

2.4 输入区

输入区提供如下功能：

1. 发送数据的输入
2. 发送按钮
3. 发送历史记录
4. 清除发送历史

2.5 状态栏



状态栏提供了如下功能：

1. 显示串口连接状态和当前使用的参数
2. 统计 Rx 和 Tx 字节数
3. 显示 SPK 当前的版本，点击版本标签将进行软件新版本检查，并高亮提示可用的新版本。

2.6 边栏

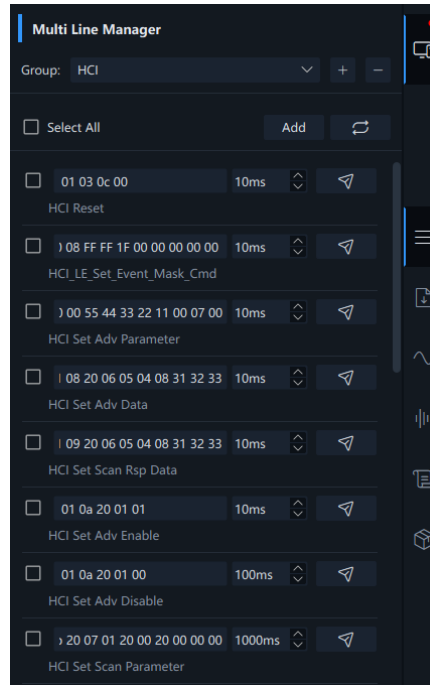
边栏提供了高级功能。

2.6.1 配置区显示控制

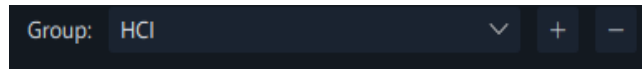
点击  可以控制"配置区"的显示和隐藏。"红点"是一个特殊设计的指示器，用于指示设备是否处于工作状态，如果设置处于工作状态，那么这个小"红点"将被点亮，便于用户观察设备状态。

2.6.2 多帧管理

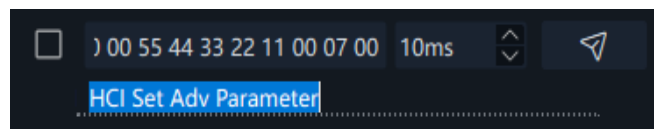
点击  即可使能多帧管理模块。



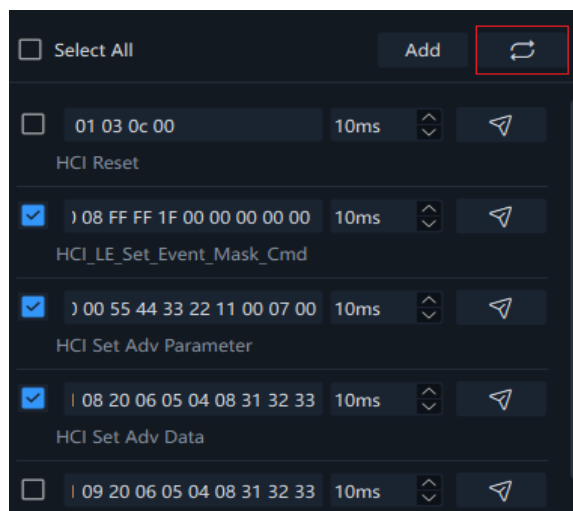
多帧管理器用于管理多帧发送以及相关的配置。多帧管理器是按照组来管理数据的，用户可以点击 $+$ 添加组或者点击 $-$ 删除当前组。用户可以按照协议类型对发送数据帧进行分类，例如，对于蓝牙 HCI 应用来说，可以分 HCI EVT, HCI CMD, HCI ACL 等，每组只包含特定类型的数据帧。当然用户也可以不分组，将所有数据帧放在一个组下。



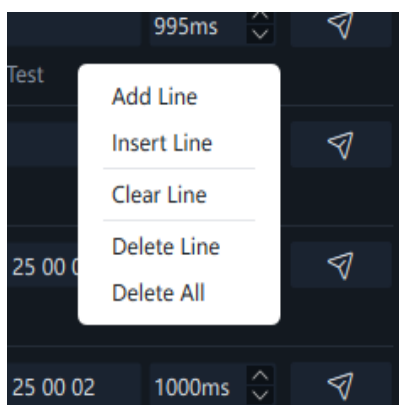
建组后，用户点击 Add 按钮添加一个新数据帧，并可以给数据帧添加描述(双击 label 即可编辑)



在多帧管理器中即可单条发送数据帧，也可以多条发送数据帧。单条发送时，直接点击其后的 $\rightarrow$ 按钮即可。如果是多条发送，需要选择发送的多条数据帧，如下图，并设置定时时间，然后点击 $\rightarrow$ 即可发送。点击 $\square$ Select All 可以选中或者取消选中当前组中的所有数据帧。

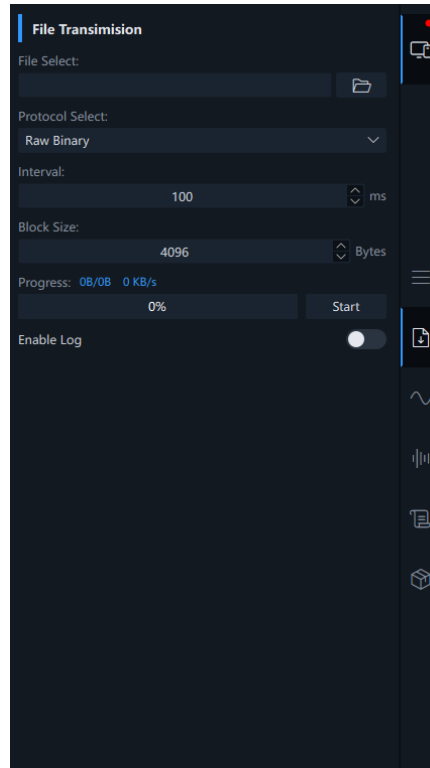


组中的数据帧可以鼠标拖动改变其位置。另外还提供了上下文菜单方便用户操作组中的数据。



2.6.3 文件传输

点击即可使能文件传输功能



SPK 提供了 6 中文件传输模式，分别如下：

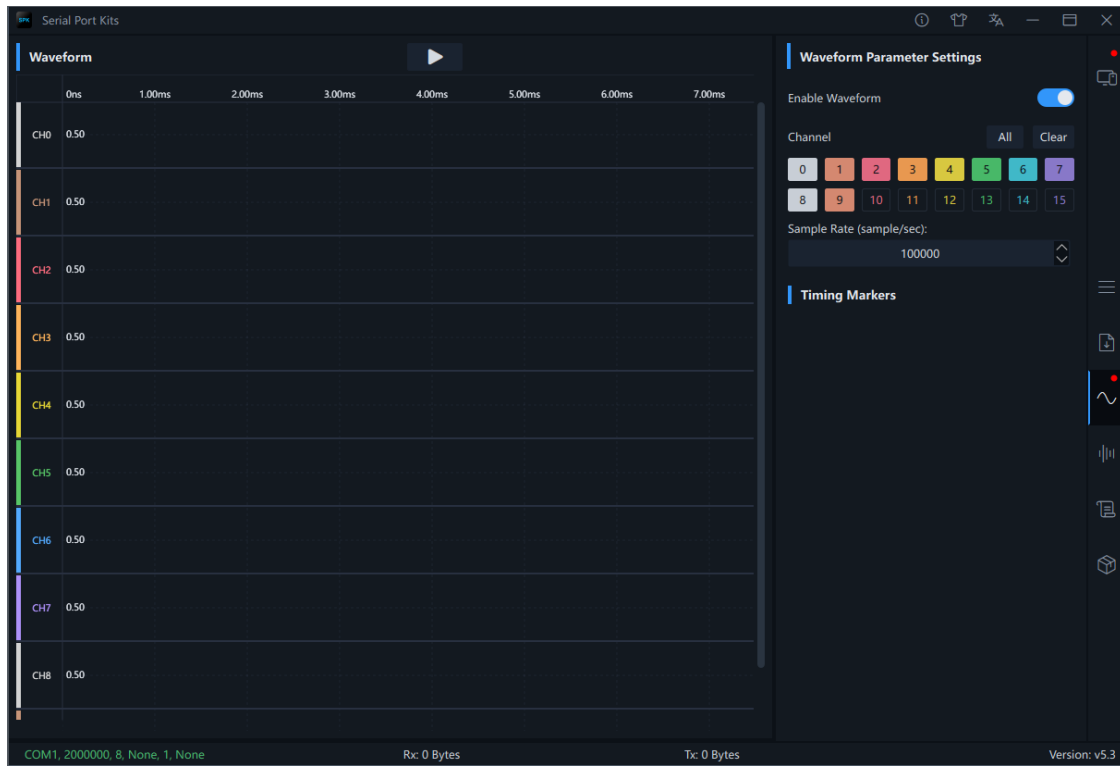
1. **Raw binary** - 二进制传输
2. **PlainText** - 文本传输
3. **XMODEM-CRC** - 支持 CRC 校验的 XMODEM 协议文件传输
4. **XMODEM-1K** - 支持块大小为 1024B 的 XMODEM 协议文件传输
5. **YMODEM** - 支持 YMODEM 协议文件传输
6. **ZMODEM** - 支持 ZMODEM 协议文件传输

用户可以选择文件传输类型，选择传输时间间隔，选择每次传输数据块大小。参数配置好后，点击【**Start**】按钮即可启动传输，传输进度和速度都将展示在 UI 上。

如果用户想查看文件传输过程中的 log，可以开启上述图中【**Enable Log**】开关

2.6.4 实时波形绘制功能

点击即可打开波形参数控制面板。在波形控制面板中打开【**Enable Waveform**】开关即可打开波形控件，如下图。



波形控件提供了 16 个波形绘制通道，16 个通道可以单独的控制其显示和隐藏。波形绘制的采样率可以根据实际数据采样率来设置，如果数据没有采样率，可以使用默认配置。

波形控件支持的数据帧格式如下：

{data0,data1,data2, ... data15}\n

数据以字符串的形式发送，每一个数据对应一个 channel，按照数据的顺序依次在相应的通道中绘制。不需要给 16 个 channel 的数据，可以根据实际情况给。例如：4 通道数据 {0.111, 0.2321, 3.989, 10.676}\n，数据类型可以是整型，也可以是浮点类型。

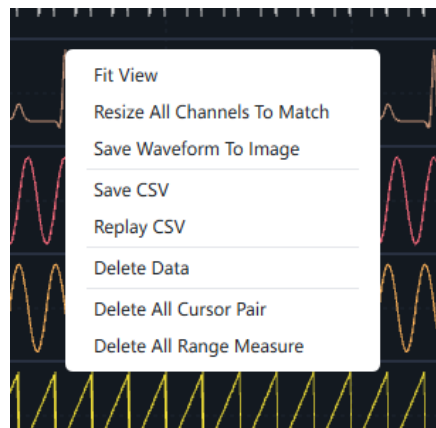
串口参数配置好并打开串口后，点击  按钮或者按快捷键 R 即可启动波形实时绘制功能。

如果波形显示过大或者过小，可以通过鼠标右键菜单中的 **【Fit View】** 功能自动调整波形到合适的大小。SPK 提供的波形控件不需要用户设置 Y 轴范围，软件自动计算，并以最合适的比例展示。通过鼠标滚轮可以缩放波形，Ctrl+鼠标滚轮可以缩放 Y 轴。

将鼠标放在任意 channel 头部区域的下边界处可以改变当前通道的垂直高度，如果想让其他 channel 和当前通道具有相同的垂直高度，只需在当前 channel 上单击鼠标右键，在下拉菜单中点击 **【Resize All Channel To Match】** 即可实现。

数据采集完以后，可以通过鼠标右键菜单中的功能对波形数据进行一系列操作,如下图所示。右键菜单提供了如下功能：

1. 自动填充波形
2. 让所有通道高度保持一致
3. 保存波形到图片
4. 保存波形数据到 CSV 文件
5. 回放已存在的 CSV 文件
6. 删除当前波形数据
7. 删除所有测量光标对
8. 删除所有数据段统计分析内容



为了对采集的数据进行分析，SPK 波形控件提供了 2 种数据分析工具：

1. 光标对
2. 统计学计算

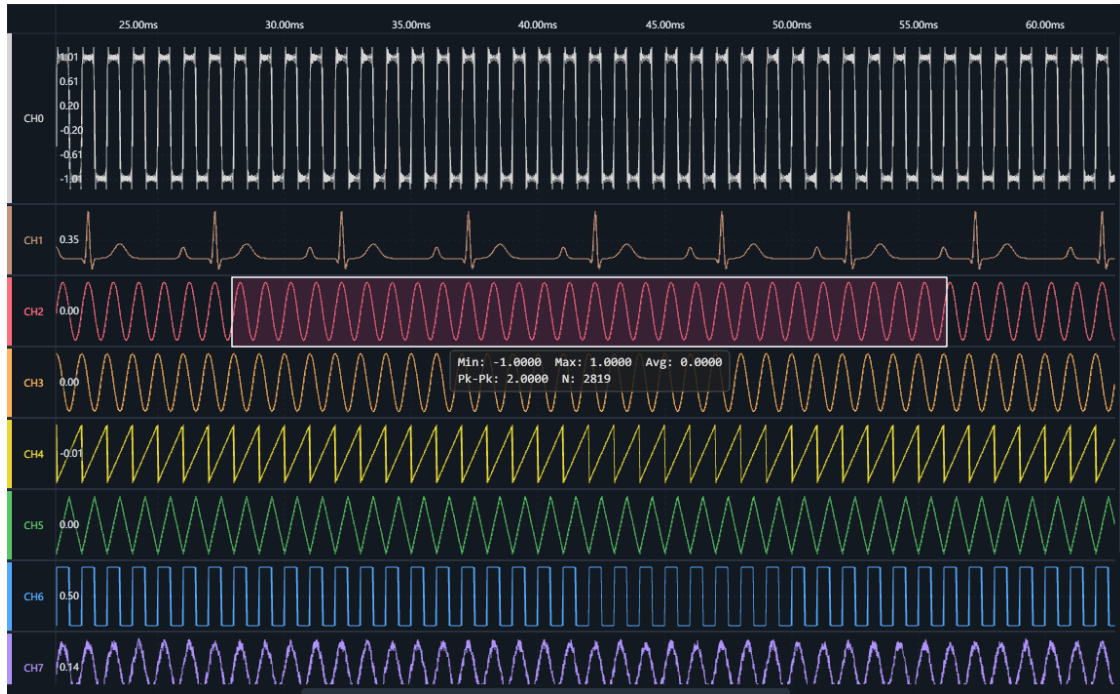
2.6.4.1 光标对

通过快捷键 **"Ctrl + T"** 即可启动光标对创建，用户通过鼠标放置相应的光标对，光标对对应的数据将显示在左侧波形控制面板中，如下图所示。可以拖动任意一个光标改变其位置，对应的数据也会跟着改变。可以创建多组光标对。



2.6.4.2 统计学计算

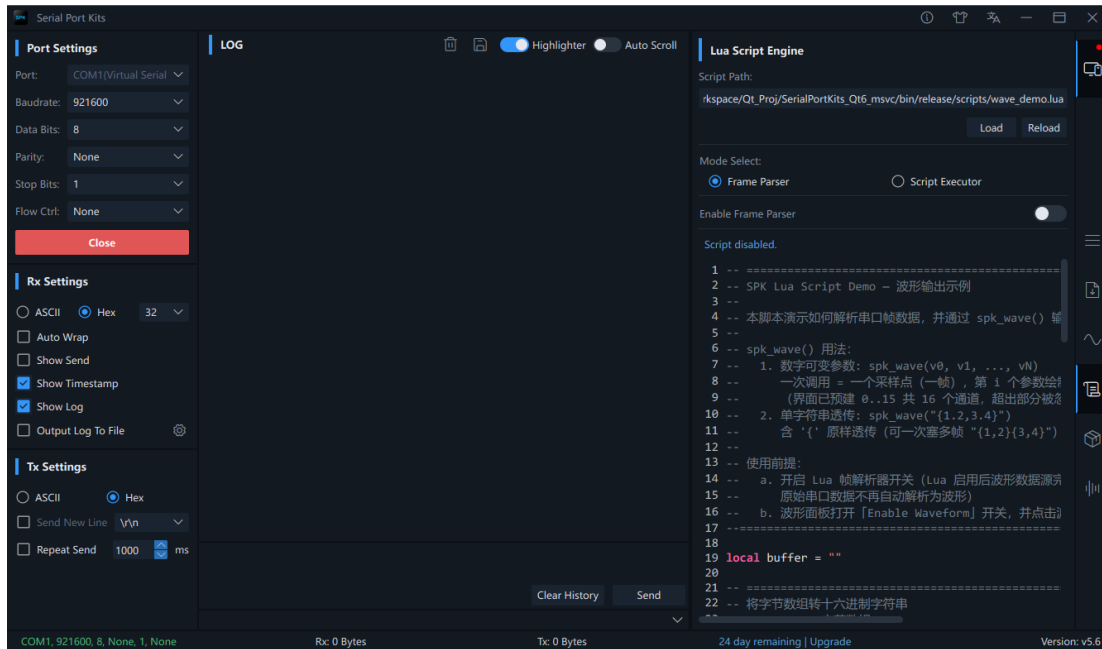
SPK 提供了对采集数据中的部分或者全部数据进行统计学计算。通过快捷键 "**Ctrl + G**"即可启动该功能，用户只需要选择需要计算的数据区域即可输出该段数据的统计学参数，例如最大值，最小值，峰峰值，平均值，数据个数等（后续根据需要添加其他的数据统计参数）



2.6.5 Lua 脚本引擎

点击  即可打开 Lua 脚本引擎控制面板，如下图所示。在 SPK 安装目录的/scripts/ 文件夹中默认提供了一些 lua 脚本的 demo，详细的展示了 lua 脚本的编写规则。用户可以参考 demo 编写适合自己协议的 lua 脚本。

为了使用 Lua 引擎，用户需要先点击 **【Load】** 按钮加载 lua 脚本，选择 Lua 引擎模式为 **【Frame Parser】** 或者 **【Script Executor】** 模式，然后打开 **【Enable Frame Parser】** 或者 **【Enable Script Executor】** 开关，lua 引擎即可启动。如果用户在 SPK 代码编辑器中修改了 lua 脚本，需要按快捷键 **Ctrl+S** 保存修改，然后点击 **【Reload】** 重载 lua 脚本；如果是通过其他编辑器修改了脚本，只需要点击 **【Reload】** 重载 lua 脚本即可。



2.6.5.1 LUA 引擎工作模式

SPK Lua 引擎支持两种工作模式：

- 1) Frame Parser Mode (帧解析器模式)
- 2) Script Executor Mode (脚本执行器模式)

2.6.5.1.1 Frame Parser Mode (帧解析器模式)

这种模式用于被动解析串口数据并将解析的数据帧显示在 UI 上，另外，还能将解析的数据帧绘制在波形控件中。

用户只需要在 lua 脚本中实现一个名为“**spk_frame_parser**”的 lua 函数，如下所示伪代码。这个函数只有一个参数 **data**, **data** 是上位机接收的串口数据，是一个字节数组。可以通过 lua 中的 **string** 模块提供的接口访问。

```
local buffer = ""
function spk_frame_parser(data)
    buffer = buffer .. data
    local frame
    -- 解析完整的数据帧
    return frame; -- 发送给上位机显示
end
```

lua 帧解析器模式有三种玩法：

- 解析一个完整数据帧（字节数组）给上位机
- lua 脚本自己解析数据帧中的各个字段(Field)，并通过 print 展示出来。
- lua 脚本解析数据后，通过 spk_wave() lua 函数发送数据到波形控件绘制波形。

1. 解析一个完整数据帧

这种模式下，lua 脚本每次解析一个完整的数据帧后，将其返回给上位机处理，返回的数据是字节数组，上位机将以 Hex 字符串的方式显示数据。lua 脚本在解析数据帧时，有两种可能的返回值，如下：

1. 返回 nil，表示数据不全，无法解析一帧数据，部分数据将被缓存在 lua 脚本的 buffer 中，待数据足够时再解析
2. 返回一个完整的帧

2. lua 脚本解析数据帧中的各个字段(Field)

这种模式下，用户可以将一个完整帧中的各个字段使用协议源语表示，方便阅读和 debug。lua 脚本此时有两种可能的返回值：

1. 返回 nil, 表示数据不全，无法解析一帧数据，数据将被缓存在 lua 脚本的 buffer 中，待数据足够时解析
2. 返回空串（""）表示还有数据可以解析，通知上位机不能停止 lua 脚本执行。

3. 通过 spk_wave() lua 函数绘制波形

这种模式下，用户可以将数据这种的数据通过 spk_wave() lua 函数发送给波形控件，用于将数据绘制成波形，便于分析。

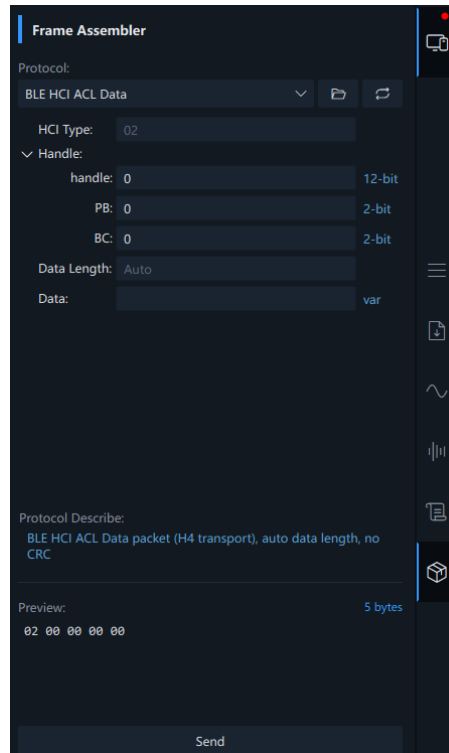
2.6.5.1.2 Script Executor Mode (脚本执行器模式)

这种模式主要用于自动化测试，比如：自动按照给定的执行流执行【发送->等待应答->判断】等。SPK 提供了若干用于自动化测试的 lua 函数，这些 Lua API 使用方法参看《SPK Lua 脚本引擎 API 文档.pdf》（可以在软件关于菜单中找到）。

另外，在 SPK 安装目录/scripts/文件夹下可以找到参考 demo，同时也可以前往哔哩哔哩搜索“SPK”看相关的视频教程。

2.6.6 数据帧组装器

点击  即可打开帧组装器控制面板，如下图所示。



SPK 设计一种帧描述文件，通过 json 文件描述协议中各个字段的信息来实现帧的自动生成。用户可以在《Custom Protocol Definition Guide.pdf》（可以从标题栏  菜单中查看）中了解帧描述文件支持的字段，并将这些字段按照自己的协议拼装成一个协议描述文件，SPK 软件将自动解析这个描述文件，生成相应的帧组装器，用户只需要填写必要字段的数据即可自动生成协议数据帧，并可直接发送，或者保存到多帧管理器中重复使用。

在 SPK 安装目录中有一个 protocols 文件夹，存放了一些协议描述文件，用户可以参考编写自己的协议描述文件。用户自己写的描述文件也必须放在该文件夹下，软件才能解析。

点击  即可打开 protocols 文件夹，查看支持的协议描述文件。

点击  即可加载 protocols 文件夹的所有帧描述文件。

点击 **【Send】** 按钮即可直接发送组装好的数据帧。