

硬石电子—串口数据图像化工具

使用说明

v1.0

技术论坛：www.ing10bbs.com

电 话：020-29814159

QQ：2536843366

QQ 交流群：515110016（硬石电子交流群）

旺 旺：硬石电子

版本历史

版本	发布时间	修改内容	作者
V1.0	2020-2-18	新建文件	硬石

关于本文档几点说明

- 1) 本文档仅适用于解释“硬石电子-串口数据图像化”使用方法，以下简称上位机。
- 2) 实际使用时需要具有串口通信功能的设备配合使用。
- 3) 在使用前需要安装 USB-串口驱动，请根据所用的串口通信模块自行下载安装。
- 4) 如遇到软件 bug 或者有功能建议，可以到 <https://github.com/Ging-H/SerialCurve> 提交 issues。

开发板资料更新链接：

硬石电子：www.ing10bbs.com

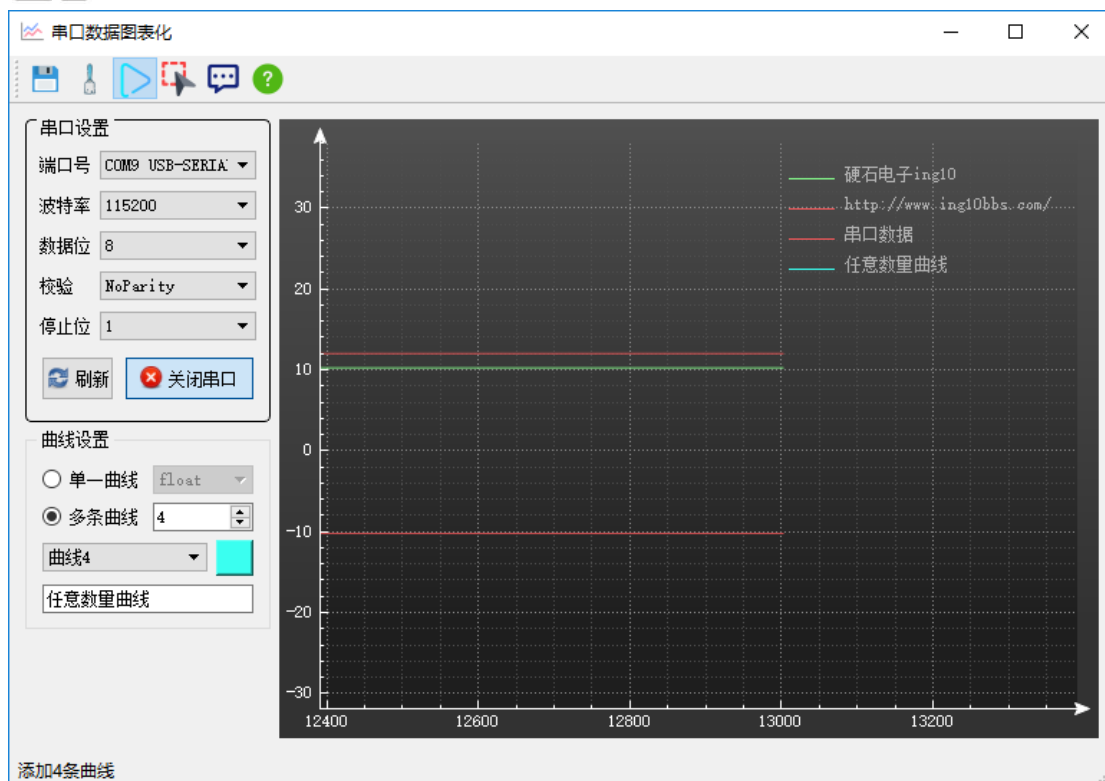
淘宝店铺：

硬石电子：<https://shop149744403.taobao.com/>

目录

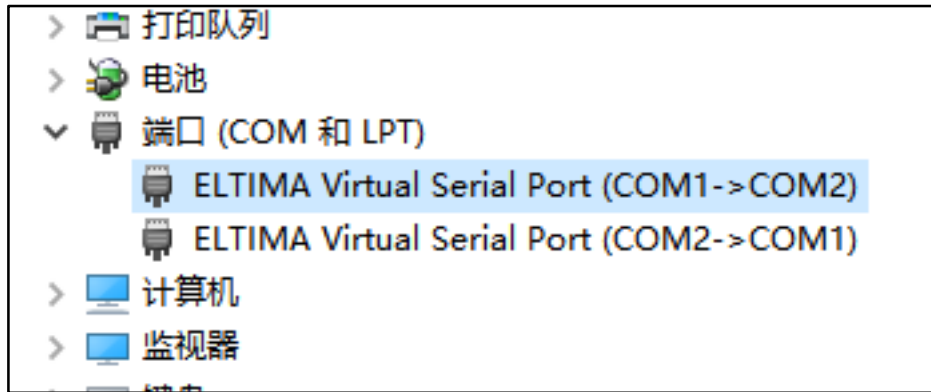
第 1 章	连接设备	5
第 2 章	曲线设置	5
第 3 章	工具栏	7
第 4 章	图像放大与拖动	7
第 5 章	通信协议	8

硬石stm32开发板



第1章 连接设备

- 在电脑的设备管理器当中找到通信设备的端口号。



- 选择正确的端口号，默认的波特率为 115200、数据位为 8bit、校验位为 NoParity，停止位为 1。通信参数需要与串口通信设备一致。



- 如果将串口通信设备连接到电脑之后，在端口号无法找到正确的端口号，可以点击刷新按钮，刷新端口号(刷新按钮只会在没有打开串口的时候有效)。在确认通信参数没有错误的情况下，点击打开串口，该软件将会占用该端口与串口通信设备通信。

第2章 曲线设置

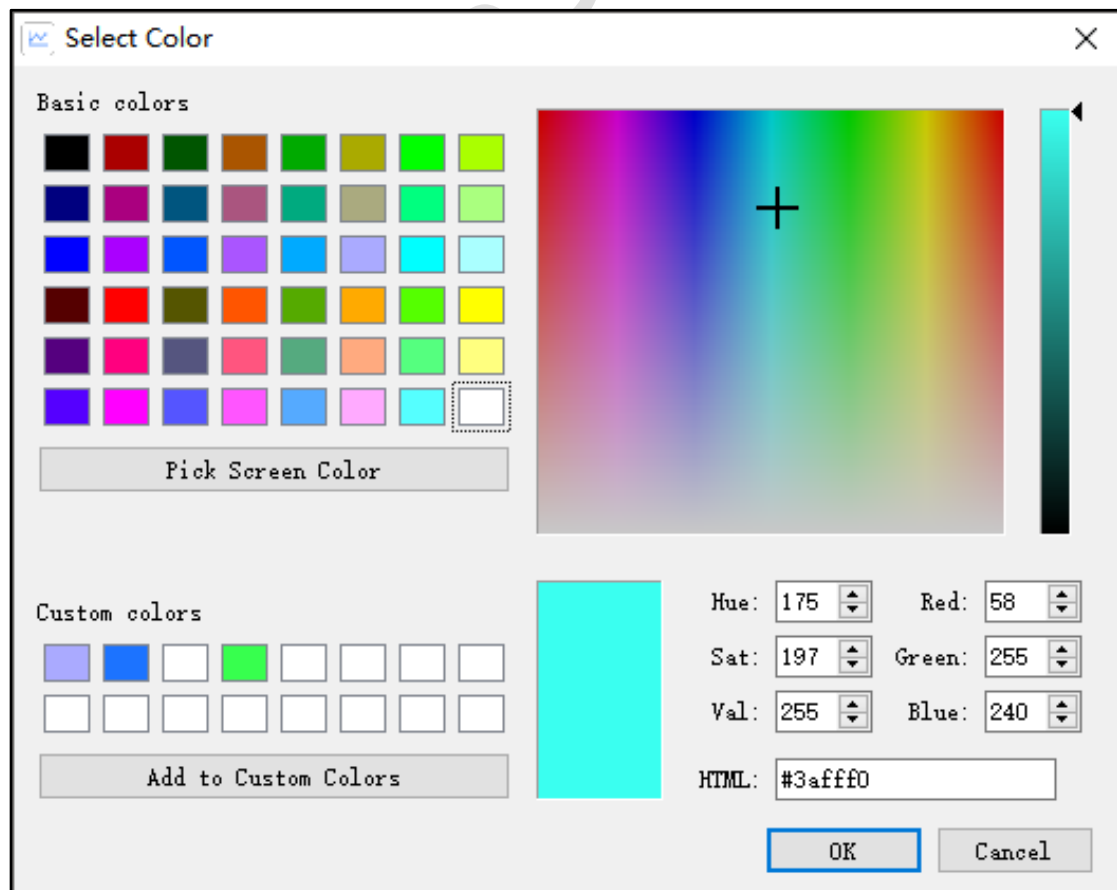
可以选择**单一曲线**或者**多条曲线**，单一曲线就是只显示一条曲线，多条曲线就是可以同时显示多条曲线，但曲线数量不多余所设定的数量。

- 如果选择了单一曲线，会根据串口传输的数据直接显示成曲线，数据类型由右边的下拉列表决定。
- 如果选择了多条曲线，则需要按照一定的格式传输数据，然后在右侧的图像区域显示多条曲线。








每一条曲线的名字和颜色都可以自由设置，首先选择需要设置的曲线，也就是下方的下拉列表，然后可以在文本输入框输入曲线名字。



点击右边的颜色方块之后，会弹出颜色选择对话框，可以根据个人喜好选择该曲线的颜色值，点击 **OK** 确认选择，点击 **Cancel** 则取消选择。



第3章 工具栏

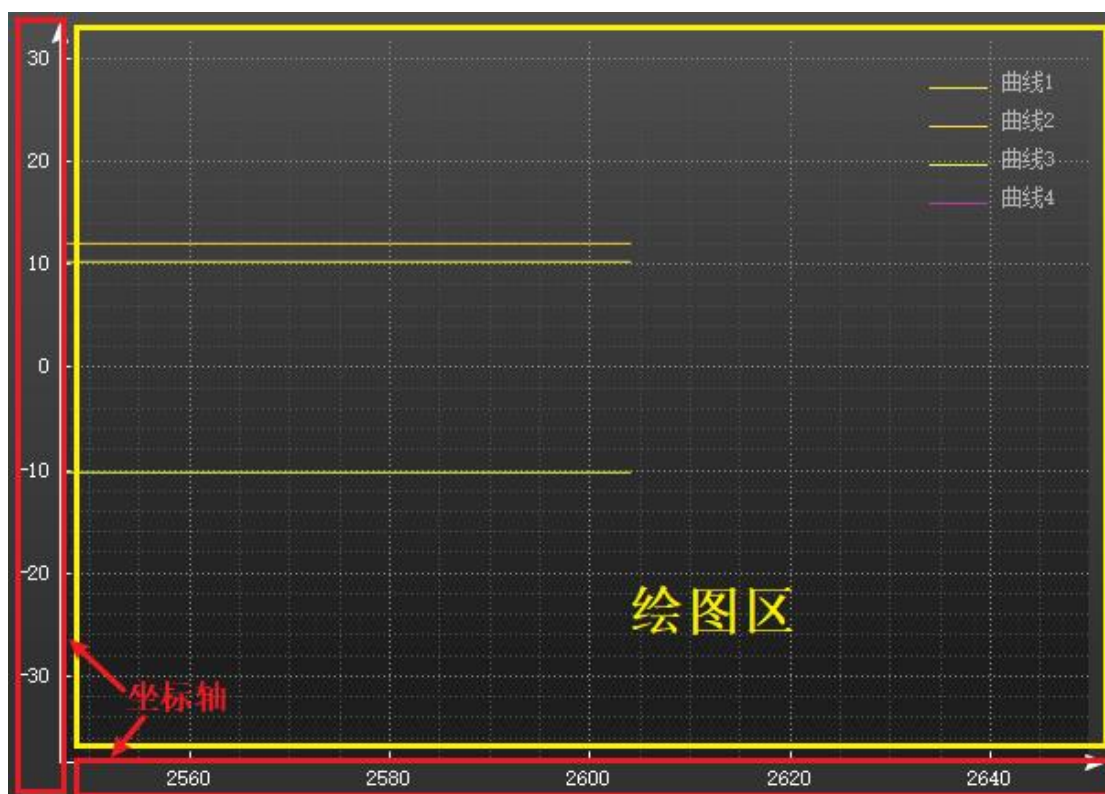
- **保存图像** ，可以将图像数据保存为 *.png *.jpg *.bmp *.pdf，可以自由设置宽度、高度和缩放比例。
- **清空图像** ，清空绘图区所有数据点。
- **暂停显示** ，暂停显示数据，但是不会暂停接收数据，会在再次点击之后继续显示新的数据点。
- **选择放大** ，对鼠标选择的矩形区域进行放大显示。再次点击恢复鼠标拖动模式。
- **坐标显示** ，在鼠标所在位置显示曲线的坐标值。
- **高亮背景** ，绘图区使用白色作为背景色，再次点击将会恢复。
- **使用说明** ，就是本文档。

第4章 图像放大与拖动

由于在接收数据点的时候需要自动调整坐标轴以保证总是显示最新的数据点，所以在数据接收期间使用鼠标滚轮是没有效果的，需要暂停显示之后才有效果。如果是多条曲线，则只会自动调整 x 坐标轴

- **整体缩放**：可以直接在绘图区使用鼠标滚轮对曲线放大缩小显示。
- **单独缩放**：可以将鼠标放置在任意一个坐标轴上，然后使用鼠标滚轮单独对该坐标轴进行缩放控制。
- **局部缩放**：使用工具栏上的选择方法功能，选择局部区域，该区域将会填充所有可显示空间。

- **拖动图像：**可以在暂停显示之后，按下鼠标左键不放，然后移动鼠标拖动图像显示位置。



第5章 通信协议

要想正确的显示传输的数据，则**串口通信设备**需要按照一定的格式发送数据。

单一曲线

如果选择了单一曲线，则需要同时选择数据类型，数据类型包括 `uint8_t`、`uint16_t`、`uint32_t`、`int8_t`、`uint16_t`、`uint32_t`、`float`。意味着**串口通信设备**传输过来的数据，只能是这些数据类型中的其中一种，例如：

- 如果选择了 `uint8_t/int8_t`，每一个字节就是一个数据点，依次发送 `0x12 0x34 0x56 0x78` 的时候，第一个数据将会被认为是 18(`0x12` 的十进制数)，第二个数据会被认为是 52(`0x34` 的十进制数)。
- 如果选择了 `uint16_t/int16_t`，每两个字节就是一个数据点，并且是低位字节在前，依次发送 `0x12 0x34 0x56 0x78` 的时候，第一个数据会被认为是 13330(`0x3412`)，第二个数据将会被认为是 30806(`0x7856`)。
- 如果选择了 `uint32_t/int32_t`，每 4 个字节就是一个数据点，并且是低位字节在前，依次发送 `0x12 0x34 0x56 0x78` 的时候，第一个数据会被认为是 2018915346(`0x78563412`)。
- 如果选择 `float`，同样是每 4 个字节是一个数据点，并且数据格式是遵循 IEEE754 标准的浮点格式，由符号位，指数位，尾数部分组成。通常单片机对浮点数据的存储方式都遵循这一标准，所以可以无需转换，像整型数据一样直接发送浮点型数据既可以，例如：`HAL_UART_Transmit(&huart1, (uint8_t*)&Data_f, 4, 0xffff);` `Data_f` 就是一个浮点型数据，直接根据首地址发送 4 个字节的数据即可(前提是单片机是小端字节序，STM32 系列的芯片就是小端字节序)。

多条曲线

如果选择了多条曲线，则需要按照相应的数据格式发送数据，需要对传输的数据进行格式化表达，规定每一条曲线的数据点格式是：曲线编号=数据值；

例如：发送 `"1=12.34;"` 就代表曲线 1 的数据点是 12.34。注意最后使用一个 `';` 作为结束符号(分割符号)。

可以在同一帧数据里面同时发送多条曲线的数据点。数据值可以是整型数据，也可以是浮点型数据，正数或负数都可以。曲线数量最多是 99 条。

例如:发送"**1=12.34; 2=12; 3=-34;**" 就代表设置曲线 1 的数据点是 12.34, 曲线 2 的数据点是 12; 曲线 3 的数据点是 -34。

单片机发送数据方法

以下方法仅针对 *stm32* 系列芯片的 HAL 库函数, 如果是其他的芯片或者其他的应用库, 请自行根据样例分析使用方法。

- 如果是单一曲线, 可以直接调用库函数 `HAL_UART_Transmit(&huart1, (uint8_t)&Data, 4, 0xffff);` Data 就是要发送的数据变量, 可以是 8 位, 16 位, 32 位的整型数据, 或者是浮点型数据。
- 如果是多条曲线, 可以调用 `printf()` 格式化输出函数 `printf("%d=%d;", num, Data);` 一次发送多条曲线的数据点: `printf("%d=%d; %d=%d;", num0, Data0, num2, Data2);`

