

基本型串口 HMI 指令表

发布版本 V1.1 发布日期:2015 年 6 月 25 日

注:

1. 设备接受指令结束符为" 0XFF 0XFF 0XFF" 三个字节。
2. 所有指令名以及参数全部使用 ASCII 字符串格式, 非二进制数据, 便于阅读和调试。
3. 所有指令名使用小写字母(此处仅仅指的是指令名称为小写, 参数该大写的时候还是要大写)。

分类一:对象及系统操作指令

1. page	刷新页面
page pageid pageid:页面 ID 或页面名称 实例 1:page 0 (刷新页面 ID 为 0 的页面) 实例 2:page main (刷新页面名称为 main 的页面) 备注: 设备上电自动刷新第 0 页。	
2. ref	重绘控件
ref objname objname:控件名称 实例:ref t0 (重绘控件 t0) 备注: 上位界面编辑软件创建控件的时候默认加载方式为自动加载, 如果设置为手动加载, 就需要通过 ref 指令来加载, 或者当您的控件被你的 gui 指令绘图遮挡以后, 或者被手动加载的别的控件遮挡以后, 可以使用重绘指令来重绘控件。	
3. get	获取变量值
get att att:变量名称 实例 1:get t0.txt(获取控件 t0 的 txt 属性值) 实例 2:get j0.val(获取控件 j0 的 val 属性值) 备注: 1. 使用 get 指令获取的变量为字符串类型时, 返回的数据为 0X70+ASCII 码+结束符, 如果是数值类型(如进度条的 val 属性)设备返回 0X71+4 字节二进制数据+结束符。数值的存放模式为小端模式(即低位在前, 高位在后)。 2. 数据具体返回格式请查看本表格后面的"基本型串口 HMI 设备返回数据格式"表。	
4. print	让设备的串口发送自定义字符串
print str str:需要发送的字符串内容 实例:print abcd(让设备发送字符串 abcd 四个字节) 备注: 使用 print 指令发送数据的时候, 设备仅仅只发送指定的字符, 没有起始符, 也没有结束符。	

5. printh	让设备的串口发送自定义 16 进制 byte
printh hex hex: 需要发送的字符的 16 进制字符串表达式 实例: printh d0 a0 (让设备发送 0xd0 0xa0 两个字节) 备注: 1. 使用 printh 指令发送数据的时候, 设备仅仅只发送指定的字符, 不会发起始符, 不会发空格, 不会发结束符。 2. 参数中每组字符间必须有且只能有一个空格隔开, 16 进制的字符串表达式大小写均支持。	
6. sendme	发送当前页面 ID 号到串口
sendme 实例 1: sendme (此指令不需要参数) 备注: 设备收到此指令会立刻把当前页面的 ID 号发送到串口, 如果想要每次刷新页面自动发送页面 ID, 请在页面的初始化事件里写上 sendme 语句即可。发送格式请参看本表格后面的”基本型串口 HMI 设备返回数据格式”表格。	
7. touch_j	触摸校准
touch_j 实例 1: touch_j (进入触摸校准功能, 此指令不需要参数) 备注: 所有设备出厂时已经校准过, 一般情况下不需要使用此功能	
8. cle_c	使所有触摸热点失效
cle_c 实例 1: cle_c (此指令不需要参数) 备注: 使用此指令后, 当前页面的所有触摸热点将不在有效, 不在自动进行触摸热区识别, 直到下次执行”page”指令的时候才会重新加载页面上设置的触摸热点。	
9. com_stop	暂停串口指令执行
com_stop 实例 1: com_stop (此指令不需要参数) 备注: 1. 暂停串口指令执行之后设备会继续接受指令, 但是都不会执行, 全部放在指令缓存区, 直到收到”com_star”指令后, 设备会从暂停时的指令开始到当前为止的所有指令全部执行。 2. 使用指令暂停与恢复功能的时候, 请评估您的设备的串口缓存区大小和指令缓存队列的最大数量是否足够支持你需要缓存的指令数目。这两项参数在你购买的设备规格书中的参数表中可以查询到。	
10. com_star	恢复串口指令执行
com_star 实例 1: com_star (此指令不需要参数) 备注: 1. 设备收到此指令之后, 将从暂停时的指令开始到当前为止的所有指令全部执行。 2. 使用指令暂停与恢复功能的时候, 请评估您的设备的串口缓存区大小和指令缓存队列的最大数量是否足够支持你需要缓存的指令数目。这两项参数在你购买的设备规格书中的参数表中可以查询到。	

11. code_c 清空串口指令缓冲区中还没有执行的所有指令

code_c

实例 1: code_c (此指令不需要参数)

立即清空串口指令缓冲区还没有执行的所有指令。

分类二:GUI 绘图指令

注: GUI 绘图指令主要应用在如下场合:

当上位界面编辑软件无法实现您的某些特殊显示要求的时候, 使用 GUI 指令自己绘图来实现自己想要的显示效果。大多数情况下其实是不需要使用这些绘图指令的, 大多数的应用都可以通过界面编辑软件的控制操作来实现。

1. cls 清屏指令

cls color

color:十进制颜色值或颜色代号

实例 1:cls 1024 (用十进制 1024 的颜色值刷屏)

实例 2: cls RED (用代号为 RED 的颜色 (RED 代表红色) 刷屏)

备注:

1. 想得到某个颜色的 10 进制数据可以使用设备配套的界面编辑软件” TJCHMI” 获取, 进入软件菜单栏” 工具” - “取色工具”。
2. 想了解设备支持的颜色代号表请参看本表格后面的” 基本型串口 HMI 颜色代号表”。
3. 本指令表中所有指令中的颜色参数, 全部都可以使用设备支持的颜色代号, 也可以使用 10 进制的颜色值, 请知晓。

2. pic 刷图指令

pic x,y,picid

x:起始点 x 坐标;

y:起始点 y 坐标;

picid:图片 ID;

实例 1:pic 10, 20, 0 (在坐标(10, 20)位置显示资源文件中图片 ID 为 0 的图片)

实例 2:pic 40, 50, 1 (在坐标(40, 50)位置显示资源文件中图片 ID 为 1 的图片)

3. picq 切图指令

格式:picq x,y,w,h,picid

x:起始点 x 坐标;

y:起始点 y 坐标;

w:区域宽度 ;

h:区域高度;

picid:图片 ID;

实例 1:picq 20, 50, 30, 20, 0 (将图片 0(必须是全屏图片)起始坐标 (20, 50) 宽度 30 高度 20 这个区域切到屏幕上显示, 显示坐标也是 (20, 50) 的起始坐标)

备注:

此指令要求图片必须是全屏图片, 否则切出来的图像不是你想要的。图片上的切图区域和屏幕上的显示区是重叠的。

4. xstr**写字指令**

xstr x,y,w,h,fontid,pointcolor,backcolor,xcenter,ycenter,string

x:起始点坐标 x;

y:起始点坐标 y;

w:区域宽度;

h:区域高度;

fontid:字库 ID;

pointcolor:字体颜色;

backcolor:背景色;

xcenter:水平对齐方式(0 为左对齐, 1 为居中, 2 为右对齐);

ycenter: 垂直对齐方式(0 为上对齐, 1 为居中, 2 为下对齐);

string:字符内容;

实例 1:xstr 0,0,100,30,1,RED,BLACK,1,1,中国

实例解释: 使用字库 1 在起始坐标(0,0), 宽度 100,高度 30 这个区域写出”中国”, 字体色为 RED, 背景色为 BLACK(如果不想写背景色可以使用 NULL 即为无背景), 水平对齐方式为居中, 垂直对齐方式也为居中。

备注:

1. 字符写到超过设定的 w 以后将自动换行, 如果换行到 h 之后还有剩下的字符没写完, 将会被忽略。

2. 关于颜色值的说明请参看 cls 指令的备注。

5. fill**区域填充指令**

fill x,y,w,h,color

x:起始点坐标 x ;

y:起始点坐标 y;

w:区域宽度;

h:区域高度;

color:填充颜色;

实例 1:fill 0,0,100,30,RED (在起始坐标(0,0)宽度 100, 高度 30 这个区域填充 RED 颜色)

备注:

关于颜色值的说明请参看 cls 指令的备注。

6. line**画线指令**

line x,y,x2,y2,color

x:起始点坐标 x;

y:起始点坐标 y;

x2:结束点坐标 x ;

y2:结束点坐标 y;

color:画线颜色;

实例 1:line 0,0,100,100,RED (在坐标(0,0)和坐标(100,100)之间画出一条 RED 颜色的线)

备注:

关于颜色值的说明请参看 cls 指令的备注。

7. draw**画矩形**

draw x, y, x2, y2, color

x:起始点坐标 x ;

y:起始点坐标 y;

x2:结束点坐标 x ;

y2:结束点坐标 y;

color:画线颜色;

实例 1:draw 0, 0, 100, 100, RED (画一个矩形, 左上角为 (0, 0), 右下角为 (100, 100), 颜色为 RED)

备注:

1. draw 画出来的是空心矩形, 需要填充实心矩形的话请直接使用 fill 区域填充指令。

2. 关于颜色值的说明请参看 cls 指令的备注。

8. cir**画空心圆**

cir x, y, r, color

x:圆心坐标 x

y:圆心坐标 y

r:半径

color:画线颜色;

实例 1:cir 100, 100, 30, RED 以坐标 (100, 100) 为圆心画一个半径为 30 的空心圆, 颜色为 RED

备注:

关于颜色值的说明请参看 cls 指令的备注。

提示: 本指令表中所有指令中的颜色参数, 全部都可以使用设备支持的颜色代号, 也可以使用 10 进制的颜色值, 请知晓, 想了解设备支持的颜色代号表请参看本表格后面的”基本型串口 HMI 颜色代号表”, 想得到某个颜色的 10 进制数据可以使用设备配套的界面编辑软件”USART HMI”获取, 进入软件菜单栏”工具”-“取色工具”。

基本型串口 HMI 系统变量列表

注：所有变量名称使用小写字母

序号	名称	含义	示例/备注
1	dim	当前背光亮度值(0-100)	1. dim=50 2. dim=dim+10 3. dim=dim-10
2	dims	上电默认背光亮度值(0-100)	1. dims=50 2. dims=dims+10 3. dims=dims-10
3	baud	当前波特率值	baud=9600 设备支持的波特率有:2400 4800 9600 19200 38400 57600 115200
4	bauds	上电默认波特率值	bauds=9600
5	spax	字体显示横向间距(上电默认为 0)	spax=2
6	spay	字体显示纵向间距(上电默认为 0)	spay=2
7	thc	触摸绘图时的画笔色	1. thc=RED 2. thc=1024
8	thdra	触摸绘图功能开关	thdra=0 (关闭) thdra=1 (打开)
9	sendxy	实时发送触摸坐标功能开关	sendxy=0 (关闭) sendxy=1 (打开) 备注: 1 打开发送功能以后,有触摸按下时设备会通过串口发送触摸坐标。 2. 发送坐标的格式请参看本表格后面的”基本型串口 HMI 设备返回数据格式”表格。
10	delay		delay=100 (让设备停顿 100ms) 备注: 执行延时指令后,设备 CPU 不会执行任何指令,但是会继续接受串口指令保存到串口指令缓存区。
11	sleep		sleep=0 (退出休眠) sleep=1 (进入休眠) 备注: 设备从休眠模式唤醒之后,设备会自动重新刷新当前页面,背光亮度也会恢复到进入休眠前的亮度值,所以如果想要退出休眠以后先用指令改变屏幕内容再点亮背光的话,在退出休眠以前先设置 dim 系统变量值为 0,这种情况下设备唤醒以后的默认亮度也就为 0。用户就可以先操作好页面,再设置 dim 变量为正常的亮度值。

基本型串口 HMI 颜色代号表

注：所有代号的书写均为大写

代号	10 进制	所表示的颜色
RED	63488	红色
BLUE	31	蓝色
GRAY	33840	灰色
BLACK	0	黑色
WHITE	65535	白色
GREEN	2016	绿色
BROWN	48192	橙色
YELLOW	65504	黄色

基本型串口 HMI 设备返回数据格式

注：

设备返回数据的结束符为” 0XFF 0XFF 0XFF” 三个字节。

返回数据第一位	含义	格式
0X00	无效指令	0X00+结束符 (当收到用户发来的无效指令时返回此数据)
0X01	指令成功执行	0X01+结束符 (用户发来的指令被成功执行完毕时返回此数据)
0X02	控件 ID 无效	0X02+结束符 (用户发来的指令中包含无效控件 ID 或者无效控件名称时返回此数据)
0X03	页面 ID 无效	0X03+结束符 (用户发来的指令中包含无效页面 ID 或者无效页面名称时返回此数据)
0X04	图片 ID 无效	0X04+结束符 (用户发来的指令中包含无效图片 ID 时返回此数据)
0X05	字库 ID 无效	0X05+结束符 (用户发来的指令中包含无效字库 ID 时返回此数据)
0X11	波特率设置无效	0X11+结束符 (用户发来的波特率配置指令中包含无效波特率参数) 设备支持的波特率有:2400 4800 9600 19200 38400 57600 115200
0X1A	变量名称无效	0X1A+结束符 当串口收到的变量名称为无效名称时返回此数据 注: 控件属性也称为变量, 比如您设置一个控件的属性的时候, 输入的是一个它没有的属性名称, 也会返回此数据。
0X1B	变量运算无效	0X1B+结束符 比如文本控件 t0 的 txt 属性赋值时应该写成 t0.txt=” abc” 如果你写成 t0.txt=abc 就出错了, 再比如进度条 j0 的 val 属性应该是数值, 所以要写成 j0.val=50, 如果写成 j0.val=” 50” 或者 j0.val=abc 也会出错
0X65	触摸热区事件返回	0X65+页面 ID+按键 ID+触摸事件+结束符 (用户创建的热区被按下或弹起时返回此数据) (触摸事件的定义: 按下事件 0x01 弹起事件 0X00) 举例:0X65 0X00 0X02 0X01 0XFF 0XFF 0XFF 含义:页面 0 按钮 2 按下

0X66	当前页面的 ID 号返回	0X66+页面 ID+结束符 (设备收到” sendme” 指令时会返回此数据) 举例: 0X66 0X02 0XFF 0XFF 0XFF 含义: 当前页面 ID 为 2
0X67	触摸坐标数据返回	0X67++坐标 X 高位+坐标 X 低位+坐标 Y 高位+坐标 Y 低位+触摸事件状态+结束符 (当系统变量” sendxy” 为 1 之后, 有触摸事件时返回此数据) (触摸事件的定义: 按下事件 0x01 弹起事件 0X00) 举例: 0X67 0X00 0X7A 0X00 0X1E 0X01 0XFF 0XFF 0XFF 含义: 坐标 (122, 30) 事件: 按下
0X68	睡眠模式触摸事件	0X68++坐标 X 高位+坐标 X 低位+坐标 Y 高位+坐标 Y 低位+触摸事件状态+结束符 (当设备进入睡眠模式后, 有触摸事件时返回此数据) (触摸事件的定义: 按下事件 0x01 弹起事件 0X00) 举例: 0X67 0X00 0X7A 0X00 0X1E 0X01 0XFF 0XFF 0XFF 含义: 坐标 (122, 30) 事件: 按下
0X70	字符串变量数据返回	0X70+变量内容 ASCII 码+结束符 使用 get 指令获取的变量为字符串类型时, 返回此数据. 举例: 0X70 0X61 0X62 0X63 0XFF 0XFF 0XFF 含义: 返回字符串数据:” abc”
0X71	数值变量数据返回	0X71+变量二进制数据 (4 字节小端模式, 低位在前)+结束符 使用 get 指令获取的变量为数值时, 返回此数据. 举例: 0X71 0X66 0X00 0X00 0X00 0XFF 0XFF 0XFF 含义: 返回数值数据: 102
0X88	系统启动成功	设备上电初始化成功之后发送此数据
0X89	开始 SD 卡升级	设备上电检测到 SD 卡之后将发送此数据, 然后进入升级界面