ZDP14x0开发环境及配套工具使用

GZLG Technology Corp.,Ltd

芯片与智能物联解决方案供应商

目录

01 ZDP14x0开发环境搭建

02 ZDP14x0开发资料获取与资料说明

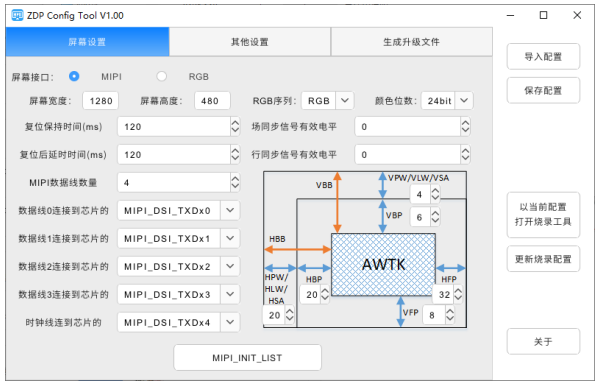
03 ZDP配套工具使用

04 屏幕适配示例

05 协议解析器&虚拟串口使用示例

06 UI交叉编译与升级

开发流程简介

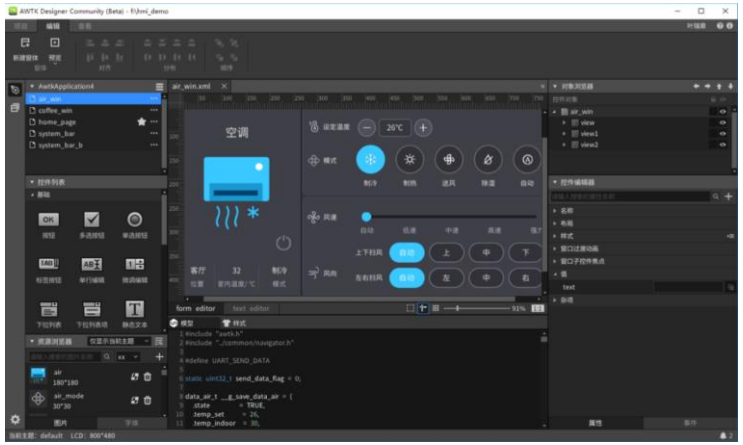


1

屏幕适配：ZDP14x0系列已内置AWTK GUI驱动引擎，只需通过参数配置上位机配置屏幕参数，点击下载即可点亮屏幕。

UI及应用逻辑开发：用户UI和功能开发过程：基于AWTK Designer进行拖拽式UI开发，UI设计完成后，打包导入芯片的应用工程进行编译。

2



3

固件升级：基于参数配置上位机打包编译好的UI固件和素材，可以通过U盘和SD卡进行拷贝升级。

AWTK Designer下载与安装

AWTK开发环境AWTK Designer：专门用来制作AWTK应用程序UI界面的实用工具。只要通过拖拽和点击就可以完成复杂的界面设计，操作简单；可以随时预览效果，所见即所得。

AWStudio安装包下载链接为https://awtk.zlg.cn/awstudio/download.html。

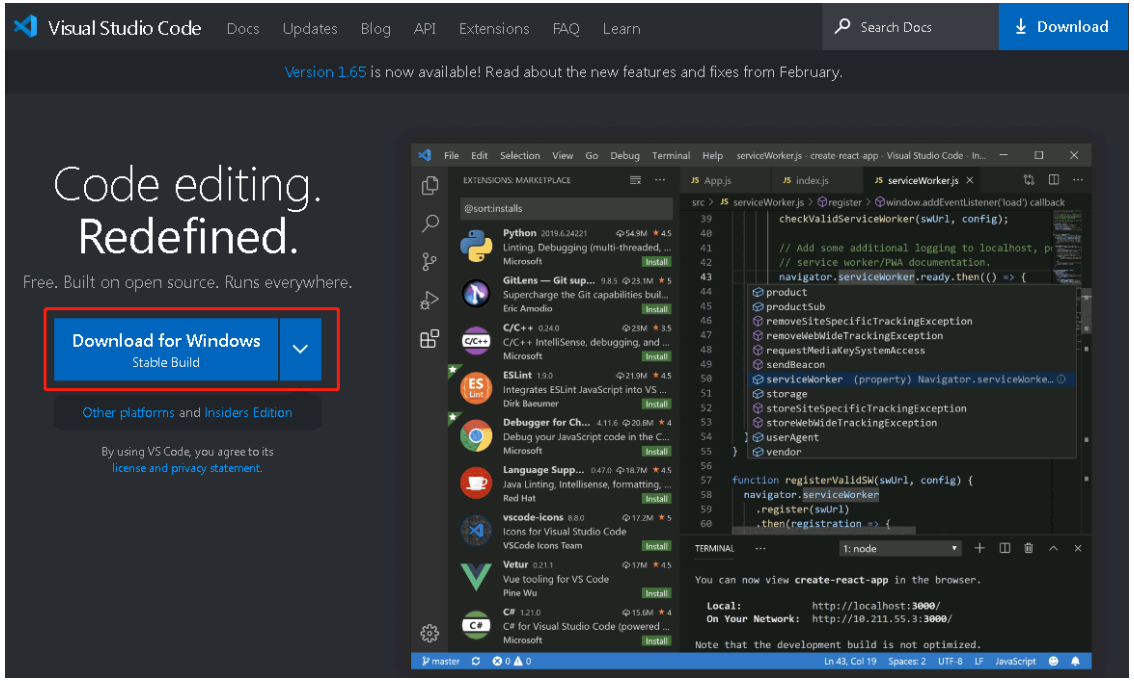
AWTK开发环境检测与安装

AWTK Designer设计完成UI界面，在PC端编译模拟运行需要依赖如下环境。 可以打开AWTK开发环境检测工具，直接一键安装，若出现安装失败，查看对应的安装指南手动安装。

VSCode下载与安装

VSCode全称Visual Studio Code，是微软出的一款轻量级代码编辑器，免费、开源而且功能强大。它支持几乎所有主流的程序语言的语法高亮、智能代码补全、自定义热键、括号匹配、代码片段、Diff、GIT等特性，支持插件扩展，支持Win、Mac、以及Linux。

VSCode官方网站https://code.visualstudio.com/ ，进入后点击【Download for Windows】开始下载安装包。



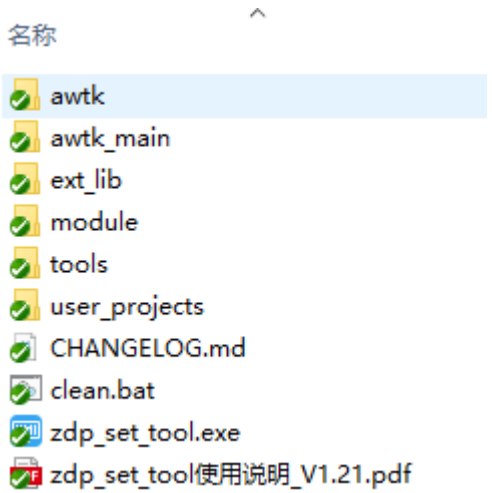
资料发布地址

ZDP14x0的软硬件资料已在Gitee开源发布，点击https://gitee.com/zlgmcuopen/HMI_ZDP14x0D 链接进入，资料如下所示，选择克隆或下载即可获取资料。

资料目录	资料说明
01.快速入门手册	快速入门手册，介绍开发流程和芯片特点
02.串口屏应用开发手册	详细介绍从开发环境搭建到应用开发，提供详细指导
03.UI_build_project	提供参数配置上位机，支持一键UI源码编译及打包
04.UI示例固件	UI示例固件，点亮屏幕后可以用于验证硬件
05.硬件设计参考	提供芯片外围电路设计的硬件设计参考，原理图库
06.芯片手册	芯片手册
07.相关技术笔记	芯片相关搭配的技术笔记与文档

目录结构

路径	
awtk	AWTK源码存放路径
awtk_main	芯片硬件接口函数
ext_lib	库文件
module	命令解析器等模块
tools	编译工具
user_projects	UI存放路径
CHANGELOG.md	修改记录
clean.bat	清除编译临时文件
zdp_set_tool.exe	上位机
zdp_set_tool使用说明_V1.21.pdf	上位机使用说明



ZDP14x0硬件函数接口

SDK工程提供了硬件接口函数，在03.UI_build_project\awtk_main目录的awtk_func.h文件中声明。
部分接口函数如下：

硬件接口函数	函数原型
数据发送	int awtk_data_send(const uint8_t* p_data, uint32_t nbytes);
蜂鸣器鸣叫	void beep_on_ms(uint32_t nms);
背光亮度调节	void adjust_backlight(unsigned long arg);
RTC配置初始化	int rtc_cfg_init(void);
RTC时间设置	int rtc_set_time(systime_t sys_time);
RTC时间获取	int rtc_get_time(systime_t *sys_time);
音频文件播放	int audio_play_file(char* audio_file);
电阻屏触摸校准	void ts_calibrate_restart(void);

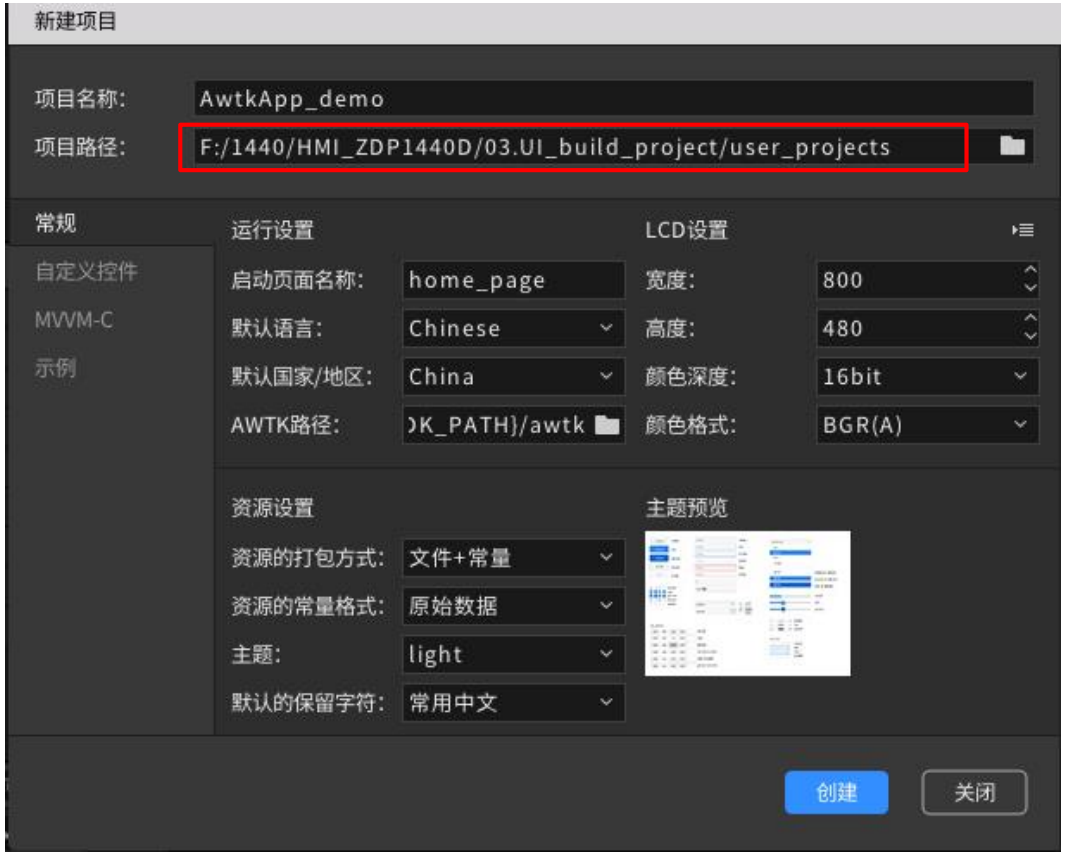
ZDP14x0硬件函数接口使用示例

硬件接口函数使用示例：

函数功能	函数使用示例
发送data数据	awtk_data_send(data, sizeof(data));
蜂鸣器鸣叫50ms	beep_on_ms(50);
背光亮度设置为50%	adjust_backlight(50);
RTC初始化	rtc_cfg_init();
设置RTC时间	rtc_set_time(sys_time);
获取RTC时间	rtc_get_time(&sys_time);
播放bin路径下的audio.mp3音频文件	audio_play_file(“bin/audio.mp3”);
进入电阻屏触摸校准界面	ts_calibrate_restart();

新建UI指定工程路径

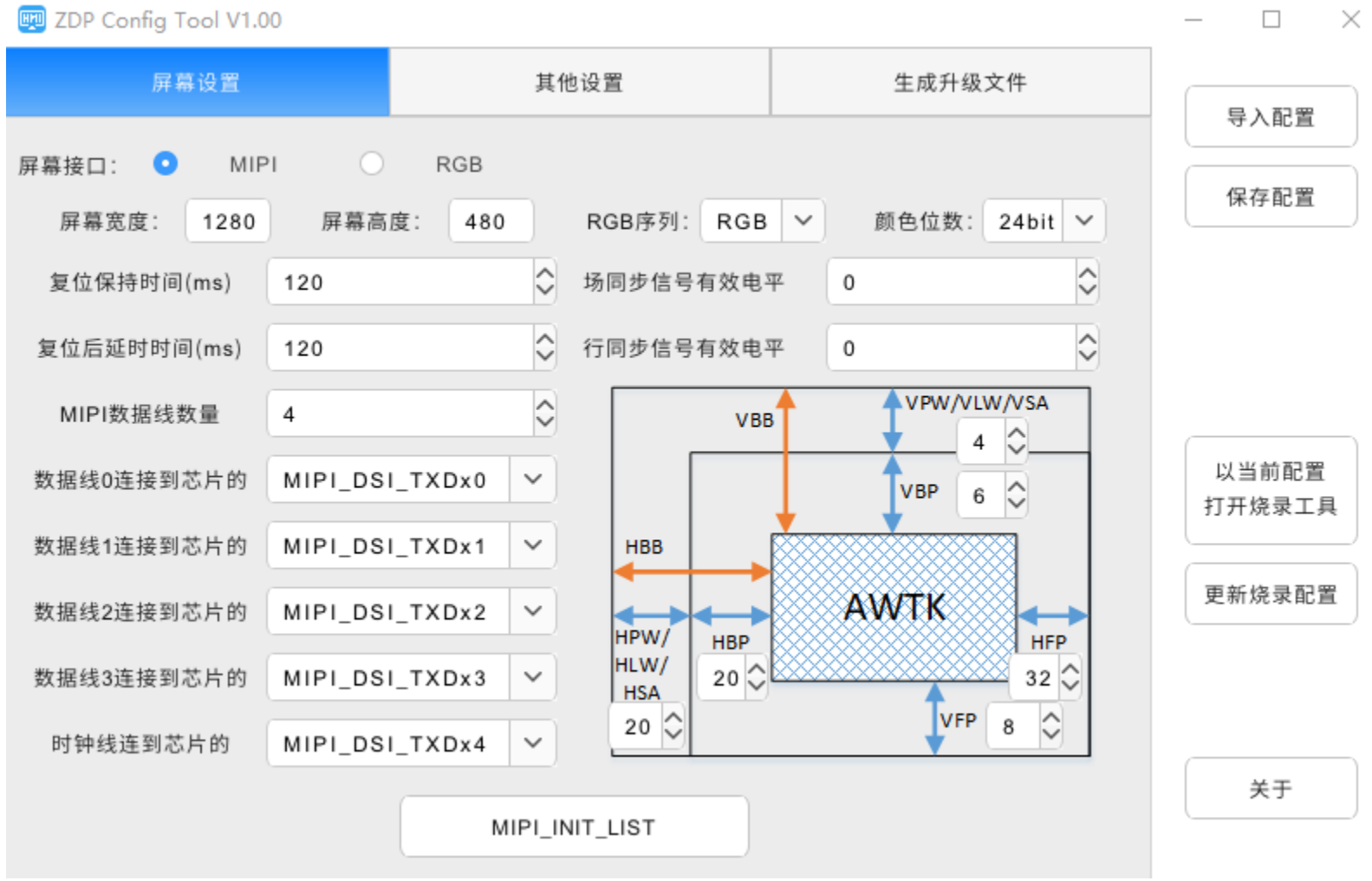
打开AWTK Designer新建UI工程，设置项目名称，指定项目路径为user_projects目录，根据屏幕分辨率配置参数。



ZDP参数配置专用上位机使用说明

ZDP14x0P128D配套专用的参数配置上位机，用户可根据各自外围设计和所选屏幕等进行参数配置，参数配置支持导入和导出，配置完成后可一键下载到HMI板子，接好电源和屏幕，即可点亮屏幕。

在“屏幕设置”页面，上位机支持RGB和MIPI两种屏幕参数配置，如图为打开上位机，在MIPI屏幕设置参数页，可根据屏幕手册填写相关参数，mipi初始化序列填写参考参数配置文档，按规则填写即可。



ZDP14x0 RGB屏幕配置示例

RGB屏幕配置：

根据屏幕手册描述，在上位机填写如下配置参数：

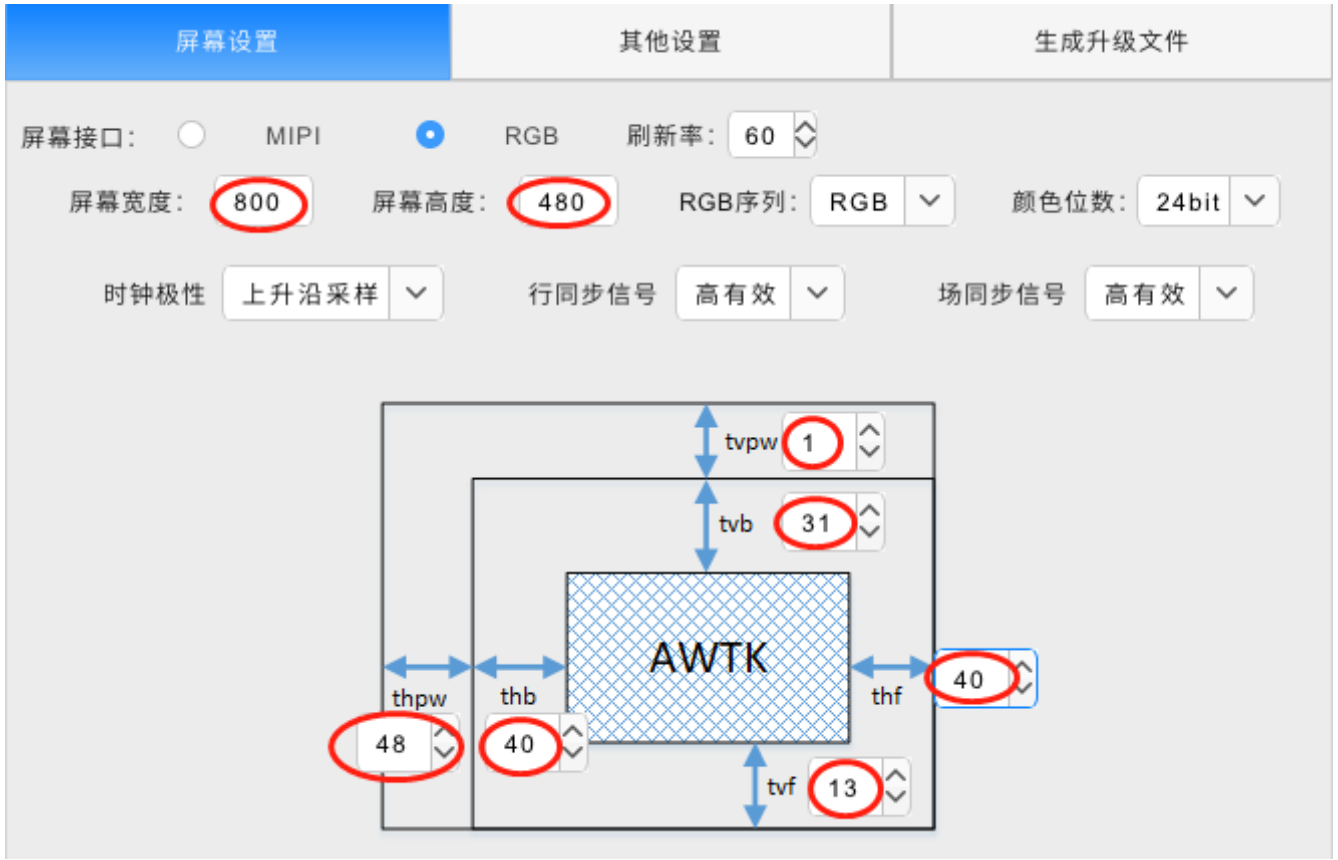
- 根据分辨率设置**宽度**和**高度**；
- RGB序列可根据接线顺序修改；
- 颜色位数选择16bit为RGB565，选择24bit为RGB888；
- 屏幕时序参数tvpw、tvb、tvf、thpw、thb、thf按屏幕手册填写，如下示意图：

Vertical input Timing

Parameter	Symbol	Min.	Typ.	Max.	Unit	Note
Vertical display area	tvd		480		H	
VSD period time	tv	517	525	712	H	tvpw+tvb=32H is fixed
VSD pulse width	tvpw	1	1	3	H	
VSD Back Porch (Blanking)	tvb	31	31	29	H	
VSD Front Porch	tvfp	5	13	200	H	

Horizontal input Timing

Parameter	Symbol	Value			Unit	Note
Horizontal display area	thd		800		DCLK	
DCLK frequency	fclk	Min.	Typ.	Max		
		20	33.3	50	MHz	
1 Horizontal Line	th	908	928	1088	DCLK	thb+thpw=88 DCLK is fixed.
HSD pulse width	thpw	1	48	87		
HSD Back Porch (Blanking)	thb	87	40	1		
HSD Front Porch	thfp	20	40	200		



ZDP14x0 MIPI屏幕配置示例

MIPI屏幕配置：

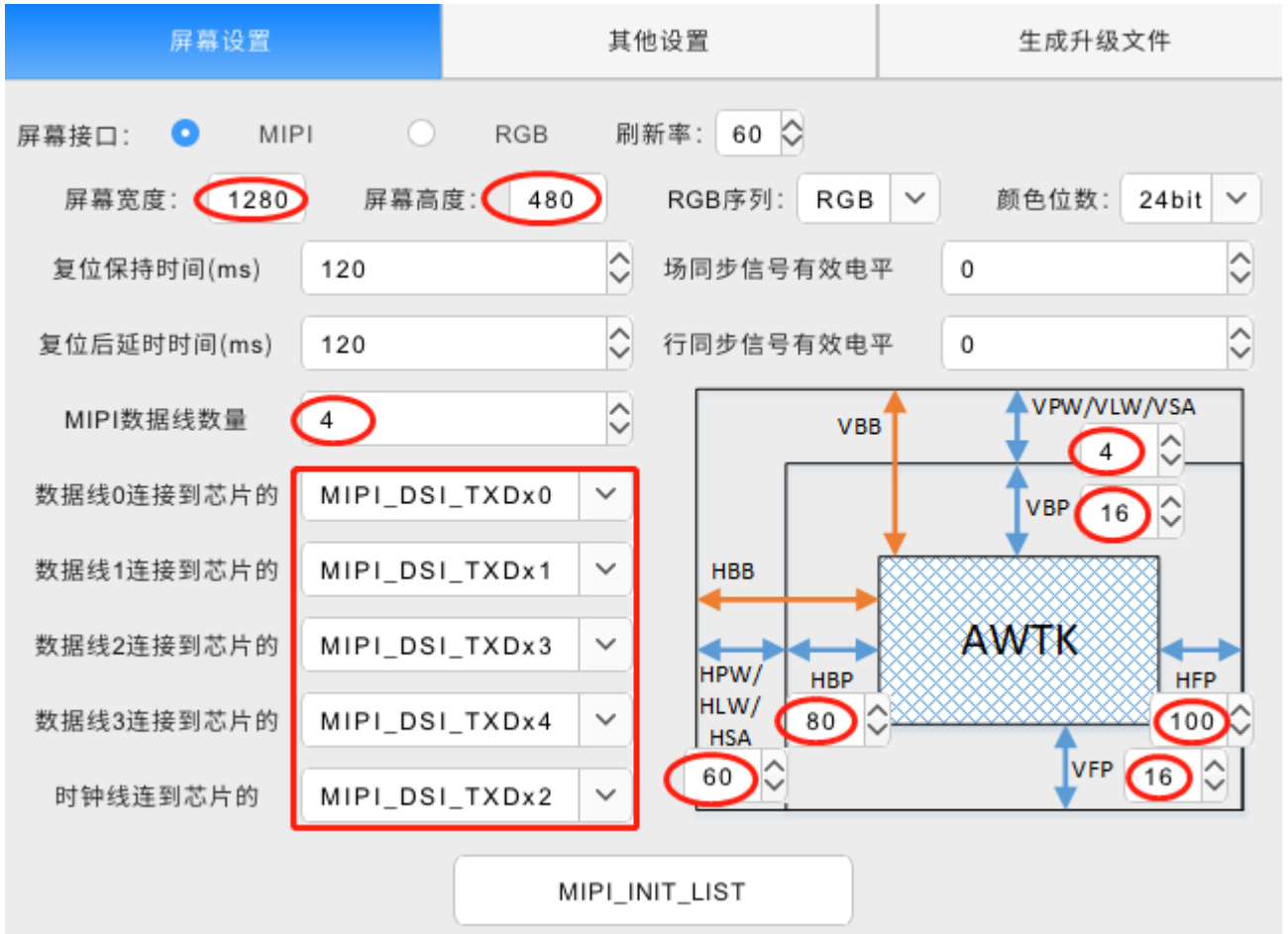
根据屏幕手册描述，在上位机填写如下配置参数：

- 根据屏幕分辨率设置**宽度**和**高度**；
- RGB序列可修改颜色顺序；
- 颜色位数选择16bit为RGB565，选择24bit为RGB888；
- 复位保持时间及复位后延时时间参照屏幕的复位时序；
- MIPI数据线数量及对应连接顺序需要根据硬件修改；
- 屏幕时序参数tvpw、tvb、tvf、thpw、thb、thf按屏厂参数填写，一般屏幕厂家提供的初始化代码中有，如下示例；

```
#define Width 480
#define Height 1280

#define VFP 16
#define VBP 16
#define VSA 4

#define HFP 100
#define HBP 80
#define HSA 60
```



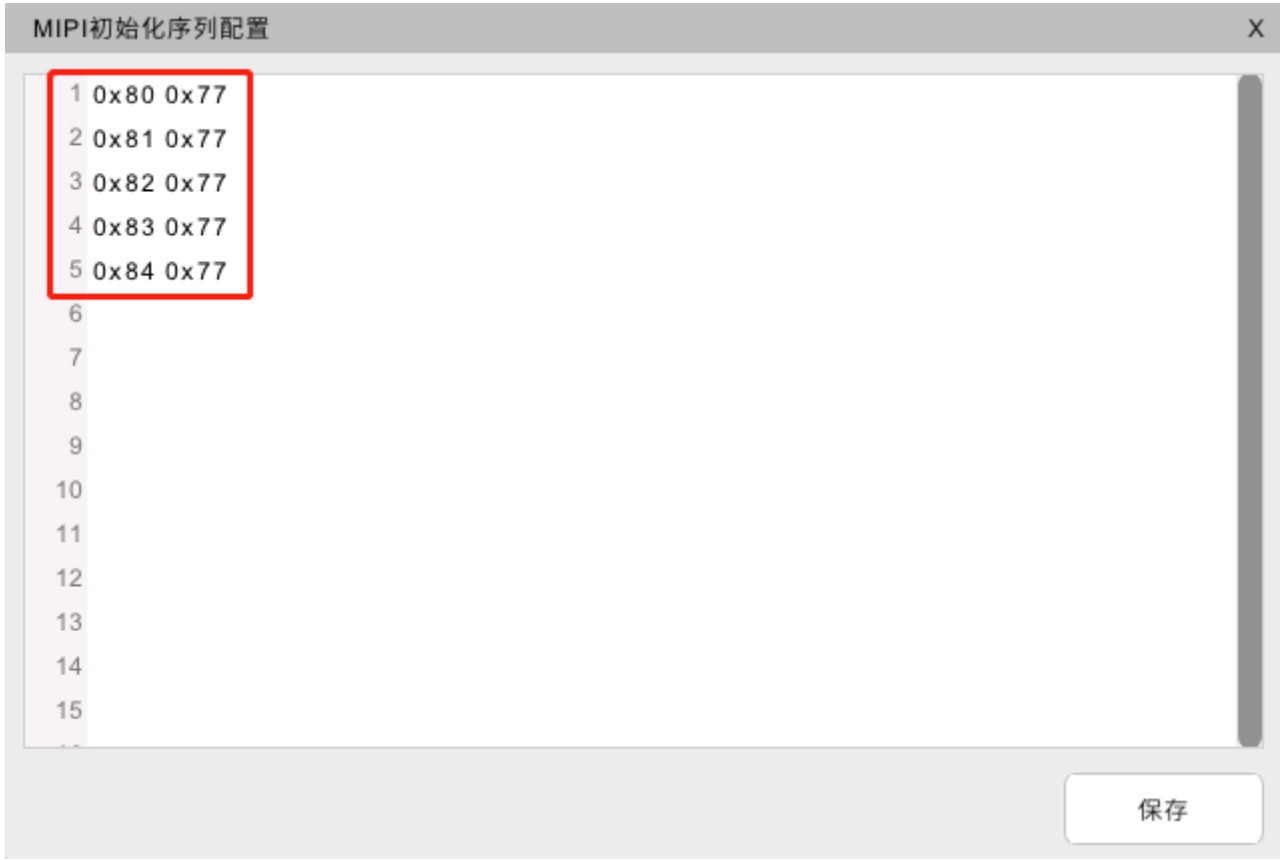
ZDP14x0 MIPI屏幕配置示例

MIPI屏幕初始化序列：

屏幕厂家一般提供两类初始化序列，DCS命令格式，也可能提供代码格式。

DCS命令格式：

```
panel-init-sequence = [
        15 00 02 80 77
        15 00 02 81 77
        15 00 02 82 77
        15 00 02 83 77
        15 00 02 84 77
];
15 00 02 80 77
 |  |  |  |  |
 |  |  |  |  数据
 |  |  |  寄存器地址
 |  |  数据长度
 |  延时
 命令类型（0x05: 单字节数据 0x15: 双字节数据 0x39: 多字节数据）
```



ZDP14x0 MIPI屏幕配置示例

MIPI屏幕初始化序列：

屏幕厂家一般提供两类初始化序列，DCS命令格式，也可能提供代码格式。

代码格式：

```
SPI_WriteComm(0xFF);
SPI_WriteData(0x77);
SPI_WriteData(0x01);
SPI_WriteData(0x00);
SPI_WriteData(0x00);
SPI_WriteData(0x00);

SPI_WriteComm(0x11);

Delay(120);

SPI_WriteComm(0x29);
```

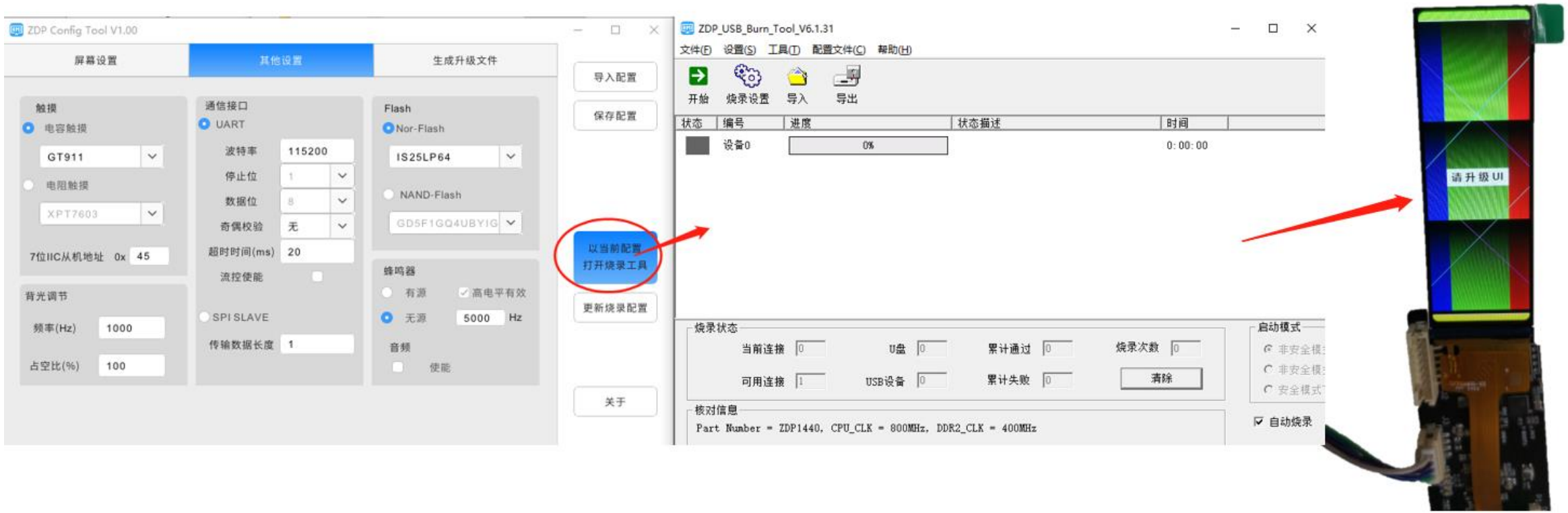
ZDP参数配置专用上位机使用说明

在“其他设置”页面，用户可以根据器件选型以及硬件设计情况，调整对应的设置以配置其相关参数。

在其他参数页面，支持触摸芯片选择，通信接口速率设置，SPI flash型号，蜂鸣器，背光参数配置等。

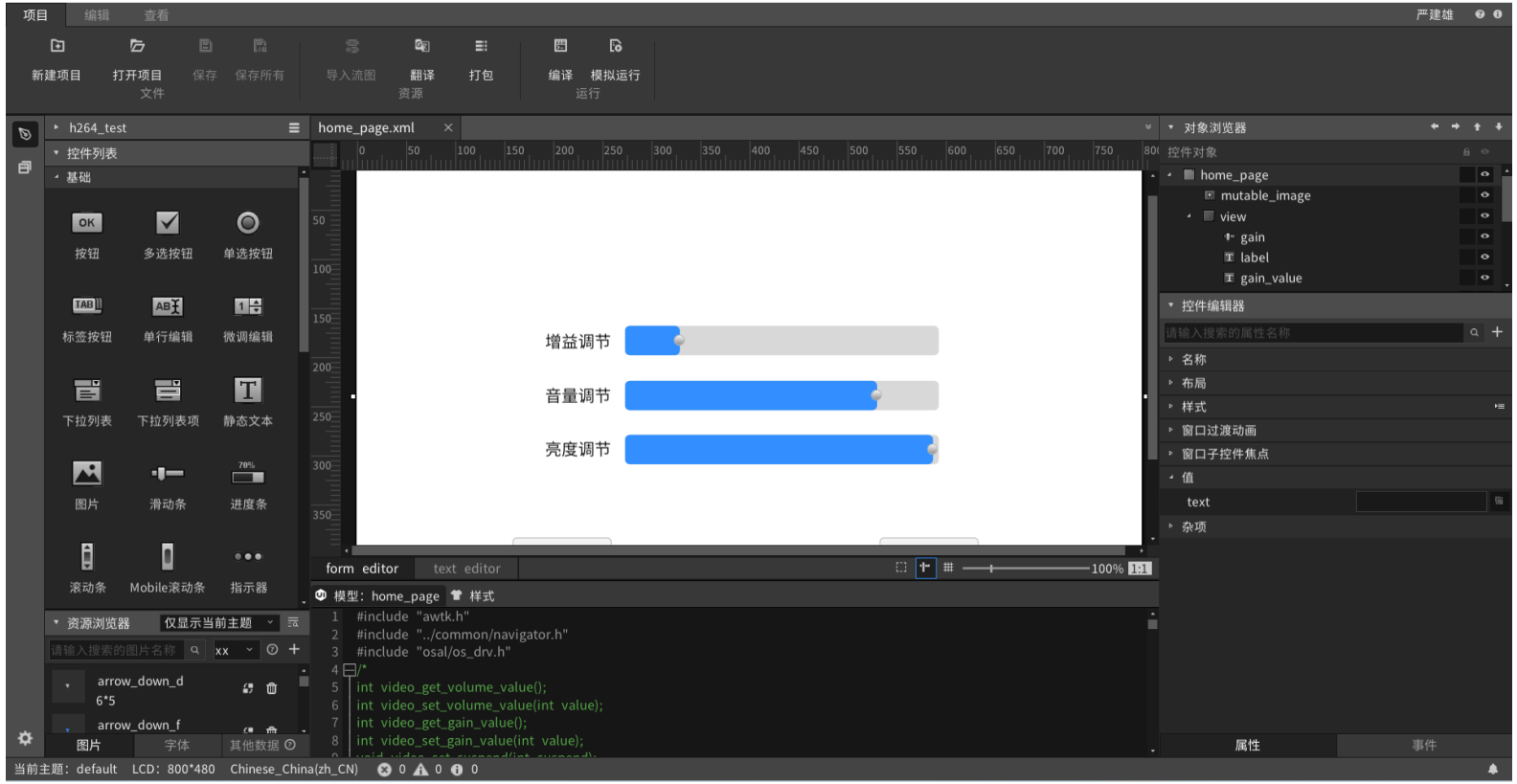
ZDP参数配置专用上位机使用说明

将屏幕设置页面及其他设置页面配置好，确认无误后点击“以当前配置打开烧录工具”按钮，此时，用 USB 线连接 PC 与ZDP14x0 PCB，按下 “Reset”和“BOOT”按钮，随即松开“Reset”，保持“BOOT”按下三秒钟后再松开，即开始烧录，烧录完成后，**屏幕就可以点亮。**



UI设计开发

打开AWTK Designer创建一个UI工程，导入图片素材，拖拽UI控件，开始UI设计；



ZDP参数配置专用上位机使用说明

PC虚拟串口调试、命令解析器

点击【添加PC端虚拟串口&命令解析器文件】，向UI工程添加相应代码；

勾选使能PC端虚拟串口，可以在无硬件的情况下调试UI工程；

使能命令解析器，可以使用自带的协议解析器，支持字符串命令和16进制命令；不使能则自行添加协议解析代码。

通信方式和数据解析

ZDP14x0支持UART和SPI从机的方式与其他设备进行通信，可以通过上位机勾选所需的通信方式和进行相关配置。不管UART或SPI从机方式通信，都采用统一的数据处理接口（ cmd_parsing_example.c文件内 ），如下所示，支持用户自定义协议解析或直接使用内置解析器。


```
/**
 * \brief  指令解析接口
 */
void cmd_parsing_port(const char* rx_data, uint8_t data_len)
{
#if (ENABLE_CMD_PARSER == 1)
    // 使用默认的协议解析器
    if (data_len >= PARSE_CMD_LEN) {
        cmd_len = data_len;
        cmd_with_data = rx_data;
        /* 解析【命令】格式的指令 */
        zdp_cmd_parsing(rx_data);
    }
#else
    // 用户自定义协议解析

}
```

用户自定义数据解析

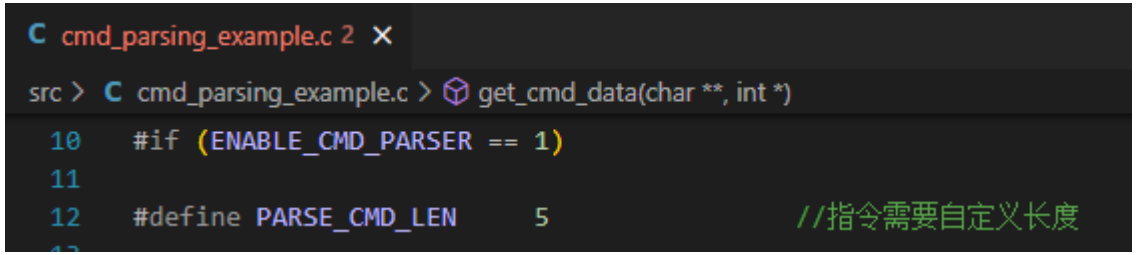
不管UART或SPI从机方式通信，都采用统一的数据处理接口（ cmd_parsing_example.c文件内 ），以用户自定义协议解析为例，假设用户协议的帧头为0xA5 0x5A，第三个字节为用户数据或命令，最后2个字节为帧尾0x0D 0x0A，将接收的数据取出传递给相关变量：

```
/**
 * \brief  指令解析接口
 */
void cmd_parsing_port(const char* rx_data, uint8_t data_len)
{
    if ((rx_data[0] == 0xA5) && (rx_data[1] == 0x5A) && (rx_data[3] == 0x0D) && (rx_data[4] == 0x0A)) {
        switch(rx_data[2]) {
        case 0x00:
            __g_sys_will_set.backlight -= 10;
            break;
        case 0x01:
            __g_sys_will_set.backlight += 10;
            break;
        default:
            break;
        }
    }
}
```

内置协议解析器使用

上面介绍了如何自定义协议解析，使用内置的协议解析器插件，可以在上位机勾选使能命令解析器后，即可使用SDK中提供的命令解析器。

该命令解析器仅能解析固定长度的串口指令，指令长度PARSE_CMD_LEN可在cmd_parsing_example中修改。



内置协议解析器使用

1. 使用ZDP_REGISTER_CMD注册字符串命令

   仅针对字符串格式的指令；第一个参数为指令字符串（不需要双引号），第二个参数为收到指令需调用的函数指针。

   按图配置后，通过串口发送指令，程序就会自动调用注册的回调函数。

   但是这种方式仅能针对字符串，有时指令还会存在0x00、或者>0x80的情况，由此便引出第二种添加指令的方式。

```
static void home_add_pro_ba()
{
    if (__g_homepage_set.progress_bar_data < 100) {
        __g_homepage_set.progress_bar_data += 5;
    }
}
static void home_dec_pro_ba()
{
    if (__g_homepage_set.progress_bar_data >= 5) {
        __g_homepage_set.progress_bar_data -= 5;
    }
}
ZDP_REGISTER_CMD("home_add_proba", home_add_pro_ba);
ZDP_REGISTER_CMD("home_dec_proba", home_dec_pro_ba);
```

内置协议解析器使用

2. ZDP_REGISTER_CMD_16注册十六进制命令

可添加非字符串格式的指令，第一个参数为存放指令的数组，第二个参数为收到指令需调用的函数指针。

按图配置后，通过串口发送A5 5A 00 0D 0A指令，程序就会自动调用backlight_ dec_cmd函数。

```c
#define PARSE_CMD_LEN     5


/**
 * \brief  backlight_dec命令回调函数：背光亮度减少10%
 */
static void backlight_dec_cmd(void)
{
    __g_sys_will_set.backlight -= 10;
    if(__g_sys_will_set.backlight <= 0) {
        __g_sys_will_set.backlight = 0;
    }
}


/* 注册 backlight_dec 协议指令 */
const uint8_t bldec[PARSE_CMD_LEN] = {0xA5, 0x5A, 0x00, 0x0D, 0x0A};
ZDP_REGISTER_CMD_16(bldec, backlight_dec_cmd);
```

指令长度 = PARSE_CMD_LEN

虚拟串口使用示例

选中对应的UI，在已经添加PC端虚拟串口文件的情况下，勾选使能PC端虚拟串口，在PC端模拟运行时，即可为模拟串口屏选择对应的串口；该串口的配置可在03.UI_build_project\module\virtual_serial_port\virtual_serial_port.h中修改。

HMI_ZDP1440D\03.UI_build_project\module\virtual_serial_port\virtual_serial_port.h

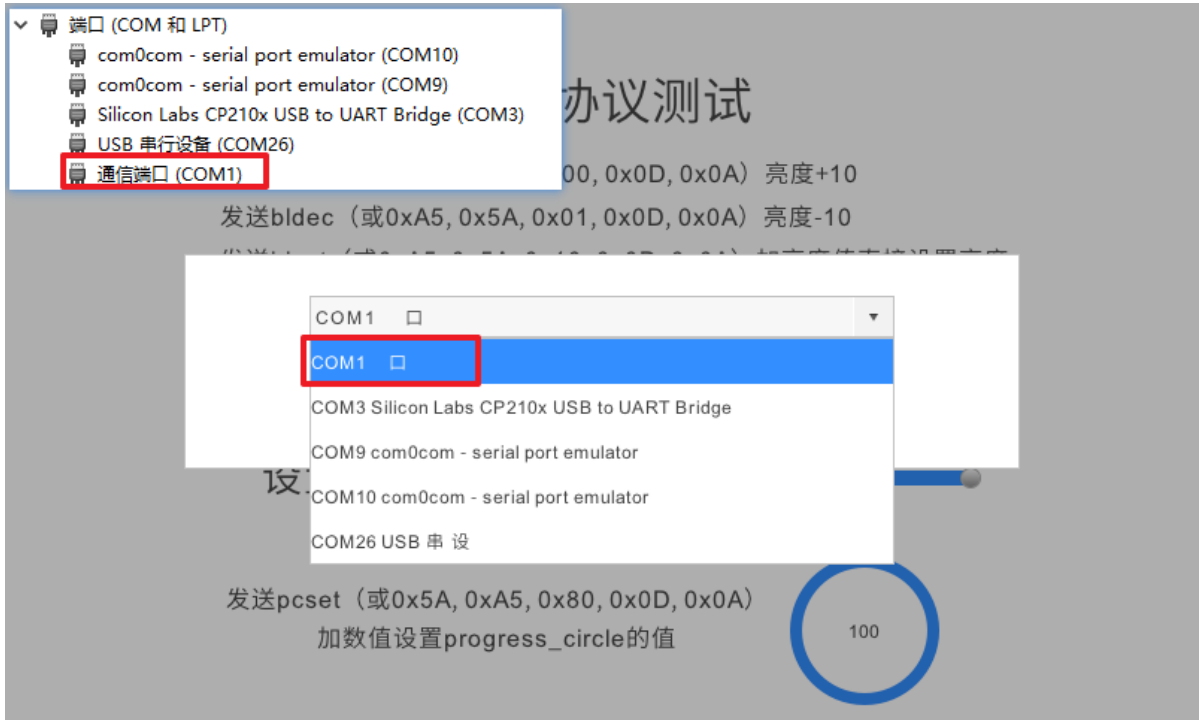
virtual_serial_port.h

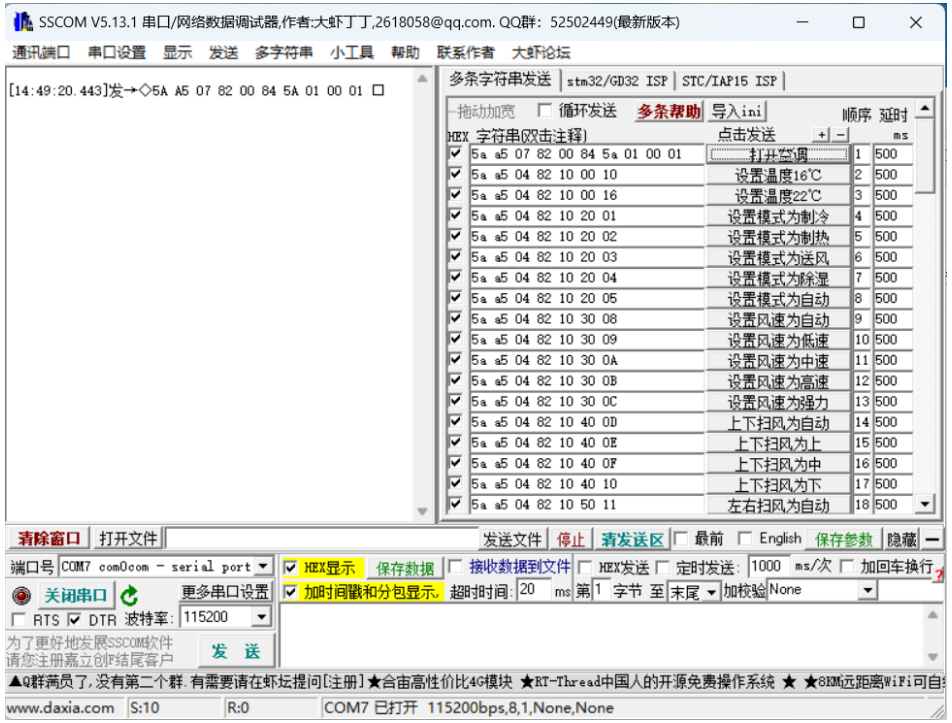
```
8
9   #define PC_UART_BAUDRATE        115200
10  #define PC_UART_BYTESIZE        8
11  #define PC_UART_PARITY          0
12  #define PC_UART_STOPBITS        1
13  #define PC_UART_FLOWCONTROL     0
14
```

虚拟串口使用示例

使能虚拟串口调试后，在AWTK Designer模拟UI运行时，弹出串口选择界面，如图所示，可选择对应的调试端口进行通信。

如果出现汉字显示不全的情况，这是由于被裁减的字库中没有对应的文字所致，不会影响程序运行。





UI固件交叉编译与升级

UI与应用逻辑设计完成后，进入上位机的生成升级文件页面，选中对应的UI后，点击生成固件按钮，即可开始交叉编译生成固件，文件存放于配置工具同级路径。

- 在其他设置中的 Flash 设置选择 Nor-Flash 的情况下，点击生成一个名为 ui_nor.bin 的文件；
- 若 Flash 设 置选择 Nand-Flash的情况下，点击生成一个名为 ui_nand.bin的文件；
- 将此文件拷贝到U盘或SD卡插入板子，即可升级UI。

Dream come true with professionalism and dedica

专业·专注成就梦想

测试
方案

干货
文章

精彩
活动

行业
热点

THANKS!

广州立功科技股份有限公司
GZLG Technology Corp., Ltd.

广州市天河区思成路43号ZLG致远电子大厦
Hotline：400-888-2705
www.zlgmcu.com