

MC51F8114 触摸板及触摸库使用说明

2020/03/10



上海晟矽微电子股份有限公司

Shanghai SinoMCU Microelectronics Co., Ltd.

目录

1	概述.....	4
2	运行环境.....	4
3	硬件准备.....	4
3.1	硬件环境.....	4
3.2	硬件连接.....	6
3.2.1	仿真器 V01 硬件连接.....	6
3.2.2	仿真器 V02 硬件连接.....	7
4	软件准备.....	13
4.1	软件启动.....	13
4.1.1	打开 demo 工程.....	13
4.1.2	编译并烧录程序.....	13
4.1.3	脱机运行程序.....	16
4.2	运行效果.....	16
5	demo 板原理图.....	19
6	触摸库快速入门.....	20
6.1	触摸库占用资源.....	20
6.2	触摸库运行环境.....	20
6.3	触摸库工作原理.....	20
6.3.1	判键触发的 4 个基本概念.....	20
6.3.2	判键触发基本公式.....	21
6.4	触摸库文件结构.....	22
6.5	新工程植入触摸库.....	23
6.5.1	新建工程.....	23
6.5.2	移植触摸库资源.....	30
7	触摸库详细介绍.....	33
7.1	tkm_config.h 配置文件说明.....	33
7.1.1	按键通道设定.....	33
7.1.2	触控参数设定.....	33
7.1.3	寄存器设定.....	35
7.2	函数及常用变量说明.....	36
7.2.1	函数功能说明.....	36
7.2.2	常用变量使用说明.....	36
7.3	TK 库调用.....	37
7.3.1	TK 库调用截图.....	37
7.3.2	TK 库调用说明.....	37
7.4	触摸库调试步骤.....	39
8	TouchKey 上位机简介.....	42
9	软件运行环境.....	42
9.1	硬件连接.....	42
9.2	软件运行.....	43
10	软件使用.....	43
10.1	串口设置.....	44


10.2	开始采样	45
10.3	图表显示设置	45
10.4	数据保存	46
10.4.1	“txt” 文本格式	47
10.4.2	“csv” 文件格式	48
11	demo 板验证测试	50
12	版本及更新记录	51

1 概述

本文档主要指导大家完成对我司 MC51F8114 芯片开发的触摸按键 demo 板程序的烧录操作，演示触摸按键效果；并对如何使用触摸库进行详细说明。

2 运行环境

1. 需安装 KEIL 51 软件。
2. 安装我司 SNLinkTK 插件安装包（在我司提供的“Keil 及仿真器工具 \SNLinkTK_KeilC51Driver_Vx.x.zip”中，如下图红框所示）。

 SNLinkTK_KeilC51Driver_V1.2.zip

3 硬件准备

3.1 硬件环境

1. 配备 MC51F8114 芯片的触摸按键 demo 板一块（以下都以此 demo 板为例演示讲解），如下图所示。

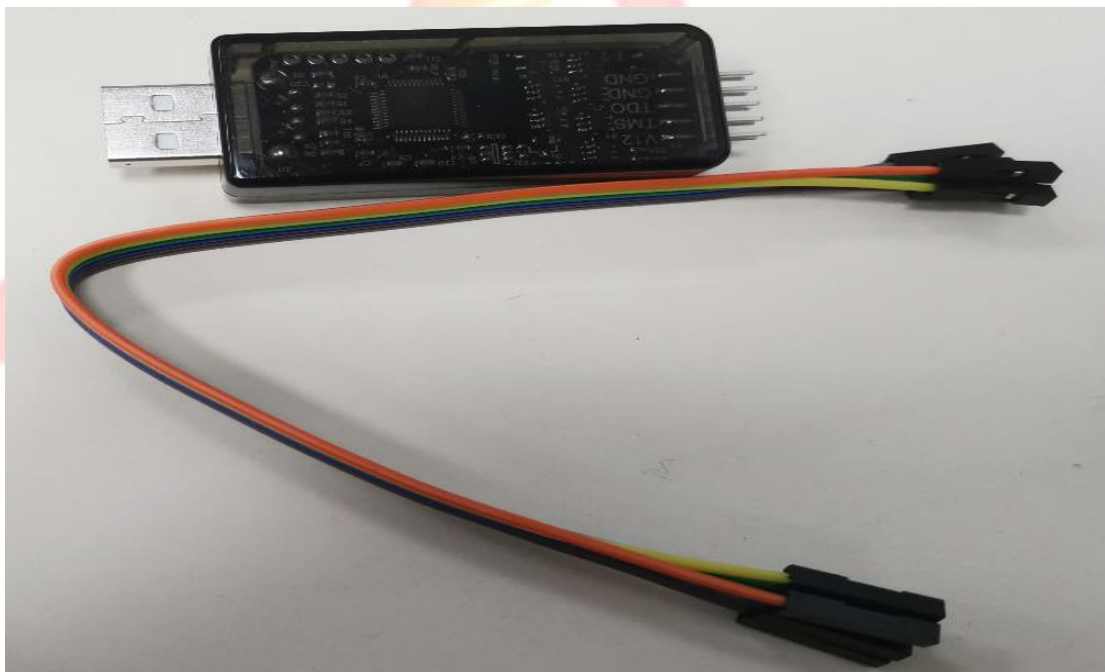


2. 我司目前有两版仿真器：仿真器 V01 和仿真器 V02，两款仿真器硬件接口不同，功能相同。因此有两套下载和仿真程序的设备配置。

1) MC51F8114 仿真器 V01 及配套 mini usb 线（用于下载和仿真程序，以下都以此配置进行演示讲解），如下图所示。



2) MC51F8114 仿真器 V02 及 6 根母对母杜邦线（用于下载和仿真程序），如下图所示。



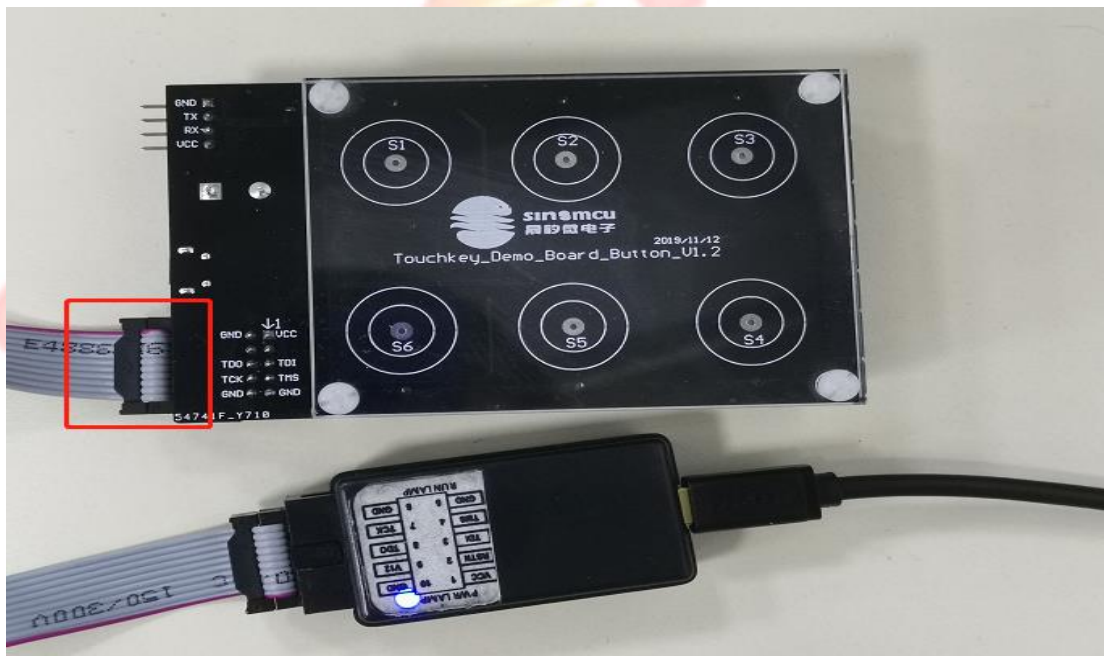
3. micro usb 线一根（供电给 demo 板，以便 demo 板脱机运行程序）。



3.2 硬件连接

3.2.1 仿真器 V01 硬件连接

如下图所示，为下载程序，需将仿真器一端连接至电脑，另一端连接至红框中标注弯牛角内。



3.2.2 仿真器 V02 硬件连接

仿真器正面图示（VCC 所在面）：

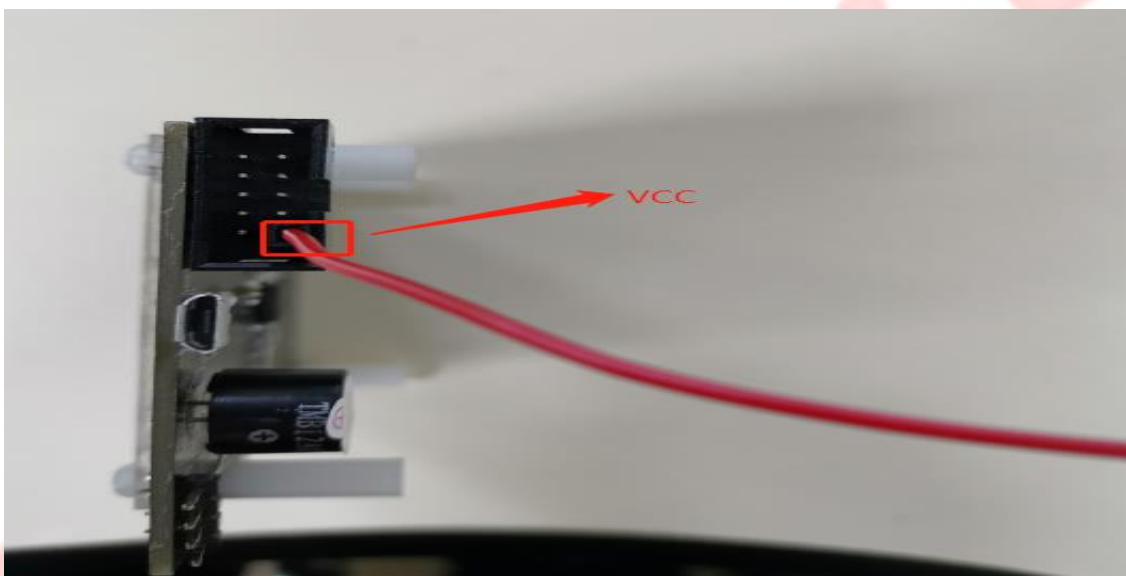


仿真器反面图示：

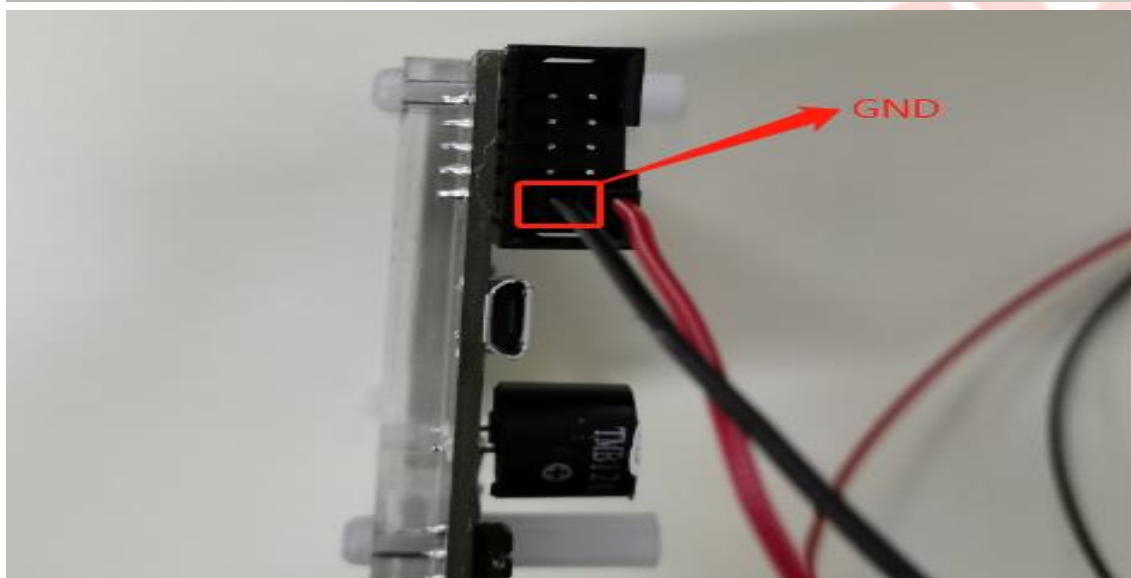
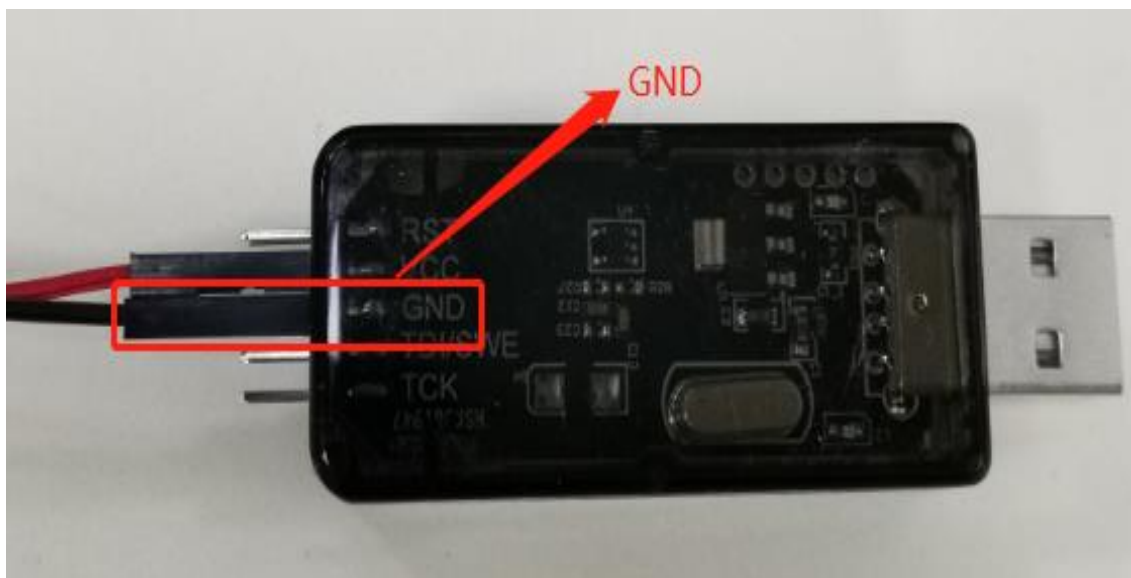


仿真器正面进行程序烧录连线，程序烧录连接步骤如下：

1 连接 VCC。

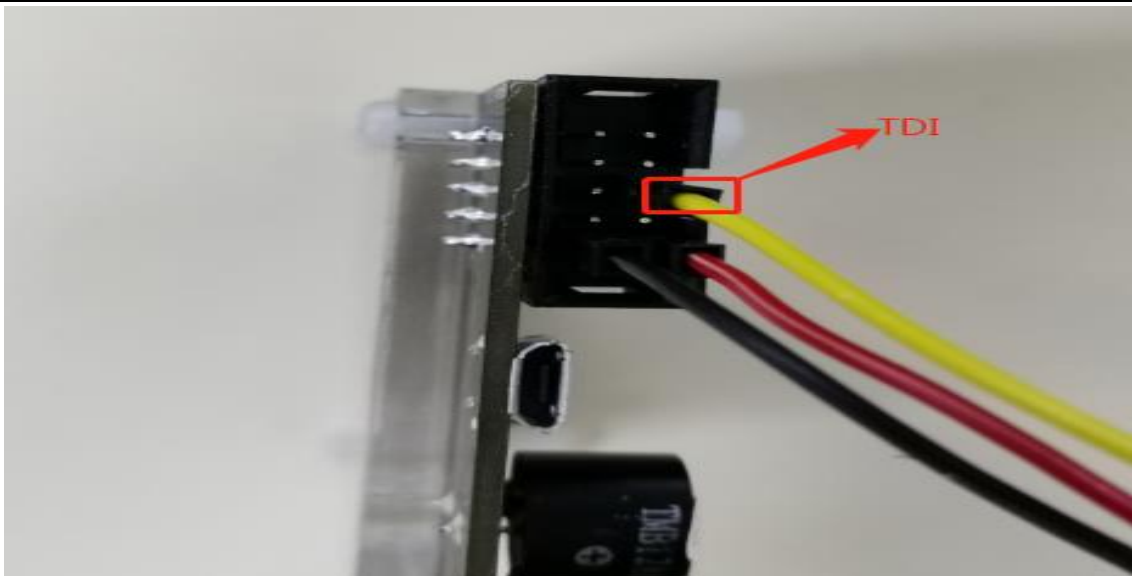


2 连接 GND。

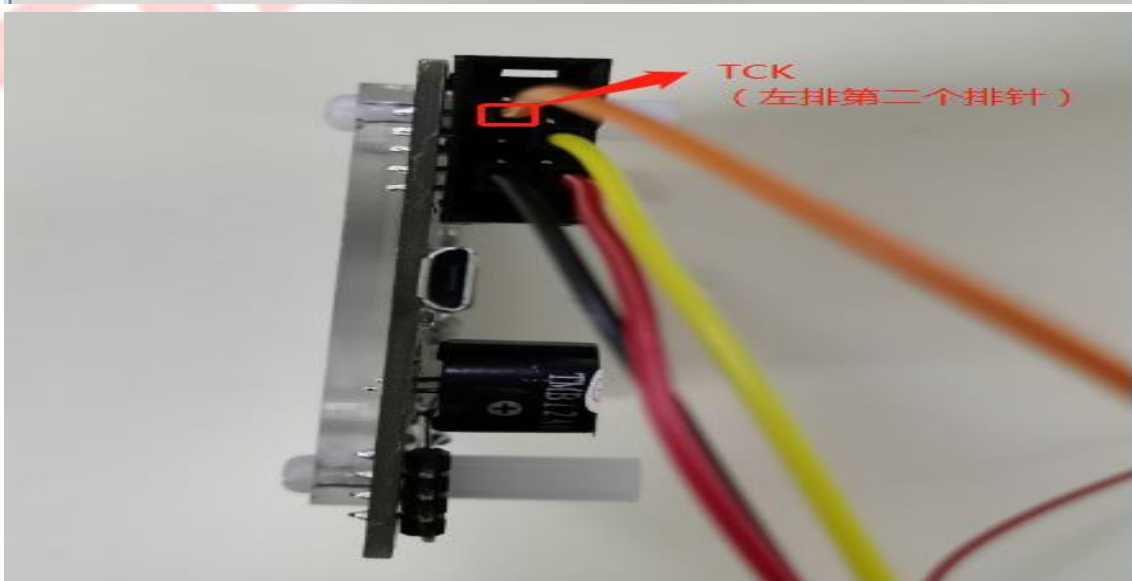


3 连接 TDI。



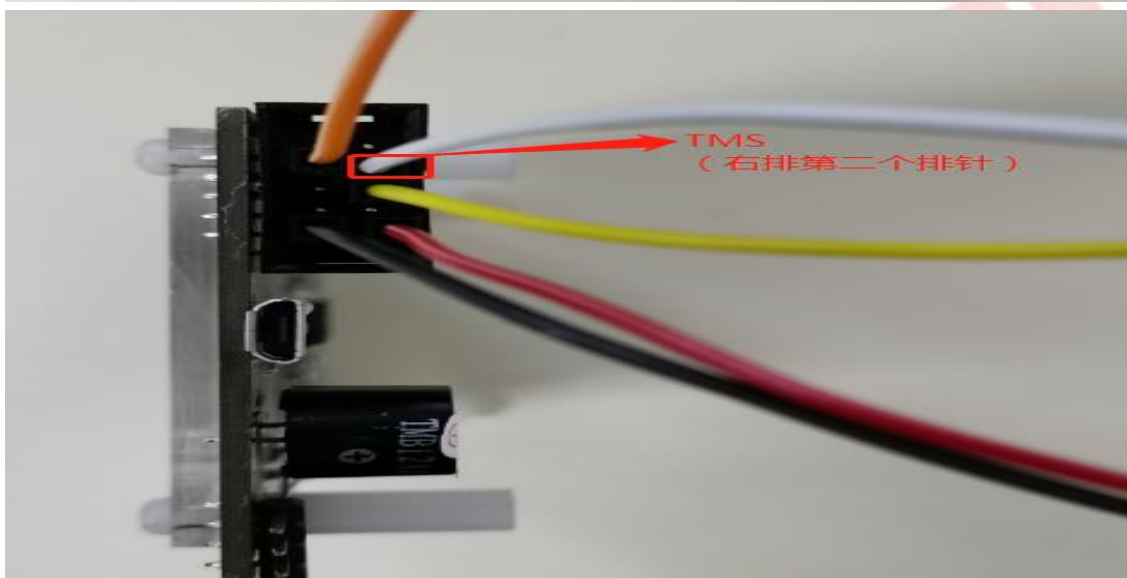


4 连接 TCK。

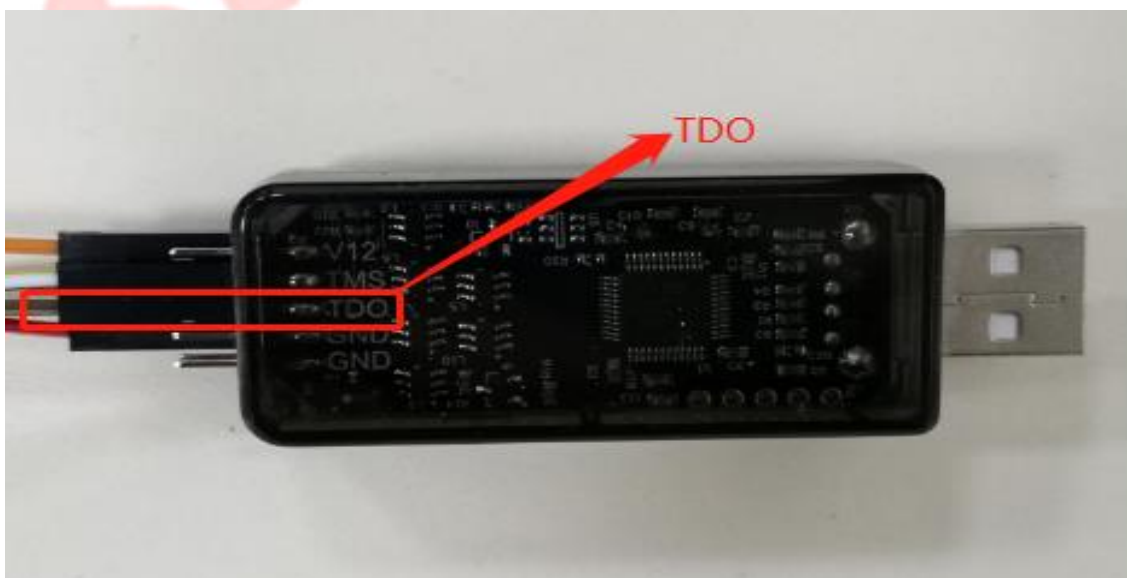


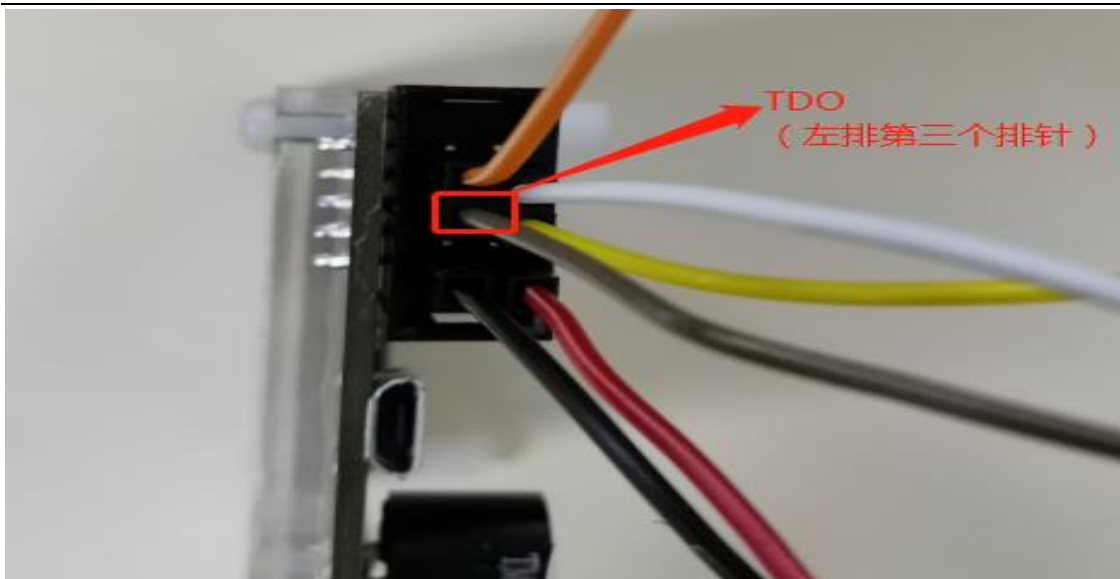
若需进行程序仿真，则还需在仿真器反面进行如下连接：

5 连接 TMS。

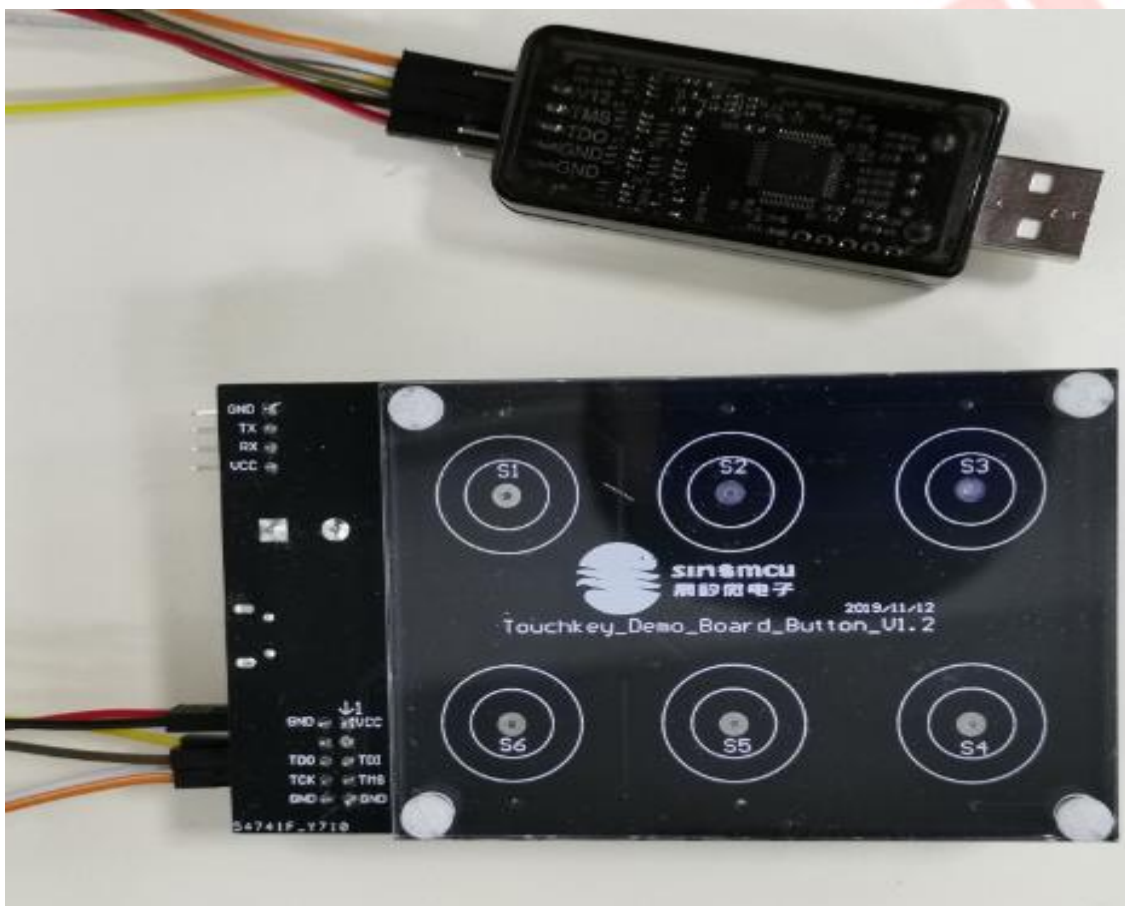


6 连接 TDO。





连接全图如下所示:



4 软件准备





4.1 软件启动

4.1.1 打开 demo 工程

如下图所示，在“demo 程序”文件夹中存有我司的 demo 板样例程序。

 DemoBoard_MC51F8114_Button_V1.1.2.zip	2020/3/3 10:55	WinRAR ZIP 压缩...	82 KB
---	----------------	------------------	-------

解压该文件夹后，内容如下图所示：

 DemoBoard_MC51F8114_Button_弹簧_V1.1.2	2020/3/3 10:50	文件夹	
 DemoBoard_MC51F8114_Button_非弹簧_V1.1.2	2020/3/6 10:17	文件夹	
 DemoBoard_MC51F8114_Button_V1.1.2.zip	2020/3/3 10:55	WinRAR ZIP 压缩...	
 VersionUpdate.txt	2020/3/8 18:06	Text Document	

根据 demo 板硬件（弹簧和非弹簧按键）不同，此文件夹相应包含两个 demo 程序：

- 1) 弹簧按键：DemoBoard_MC51F8114_Button_弹簧_Vx.x.x。
- 2) 非弹簧按键：DemoBoard_MC51F8114_Button_非弹簧_Vx.x.x。

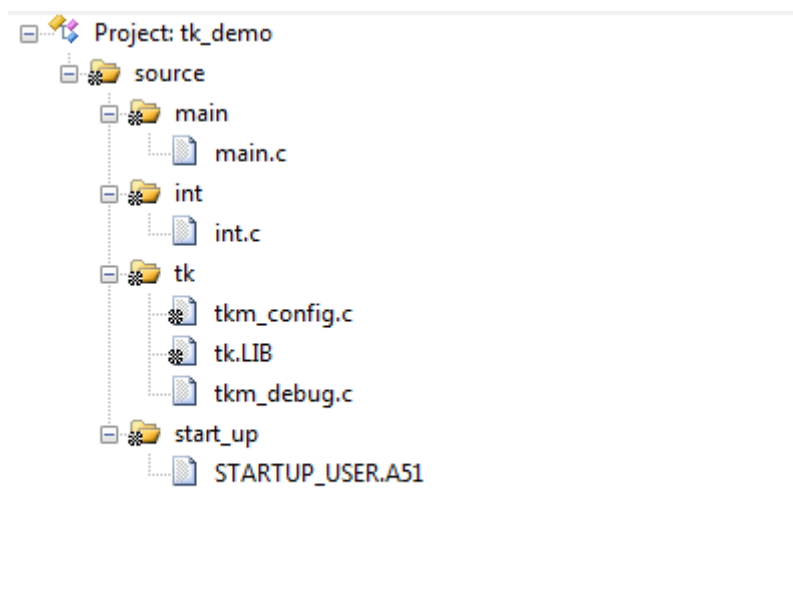
文件夹中 VersionUpdate.txt 中记录了 demo 程序的更新细节。下面以非弹簧的 demo 板及其程序为例进行讲解。

4.1.2 编译并烧录程序

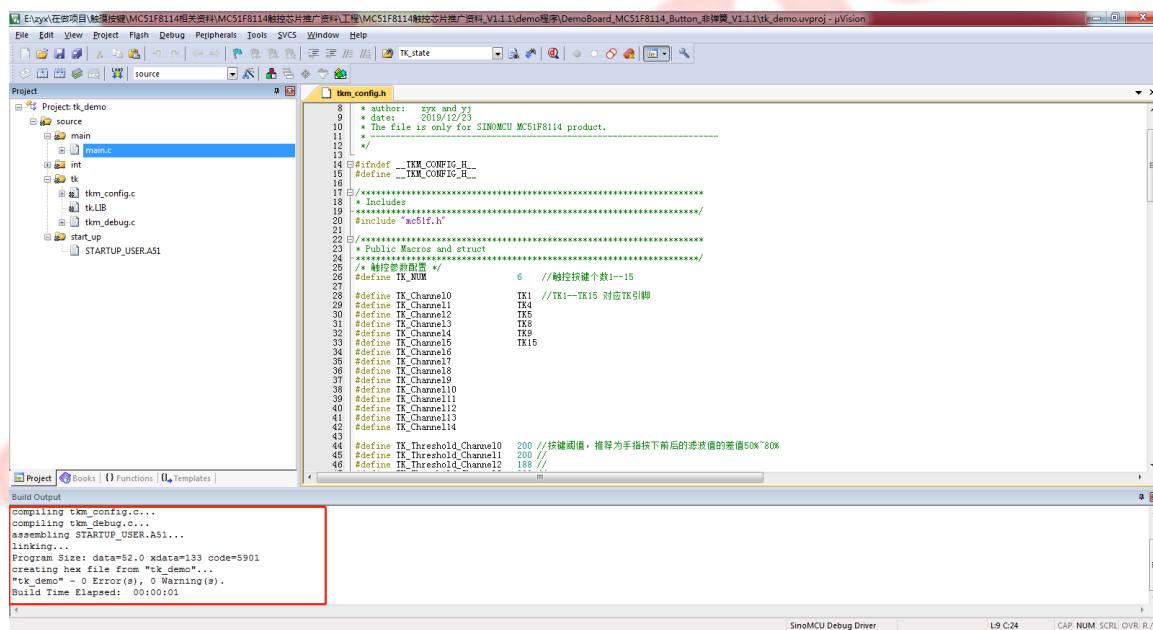
- 1 打开“DemoBoard_MC51F8114_Button_非弹簧_Vx.x.x”的工程文件夹，找到下图所示工程文件并使用 Keil C51 软件打开。

名称	修改日期	类型	大小
 Include	2019/12/12 10:17	文件夹	
 Source	2019/11/28 16:47	文件夹	
 CodeOption.bin	2019/12/12 10:21	BIN 文件	1 KB
 tk_demo.uvopt	2019/12/11 21:01	UVOPT 文件	8 KB
 tk_demo.uvproj	2019/12/12 10:00	Keil uVision4 Project	27 KB

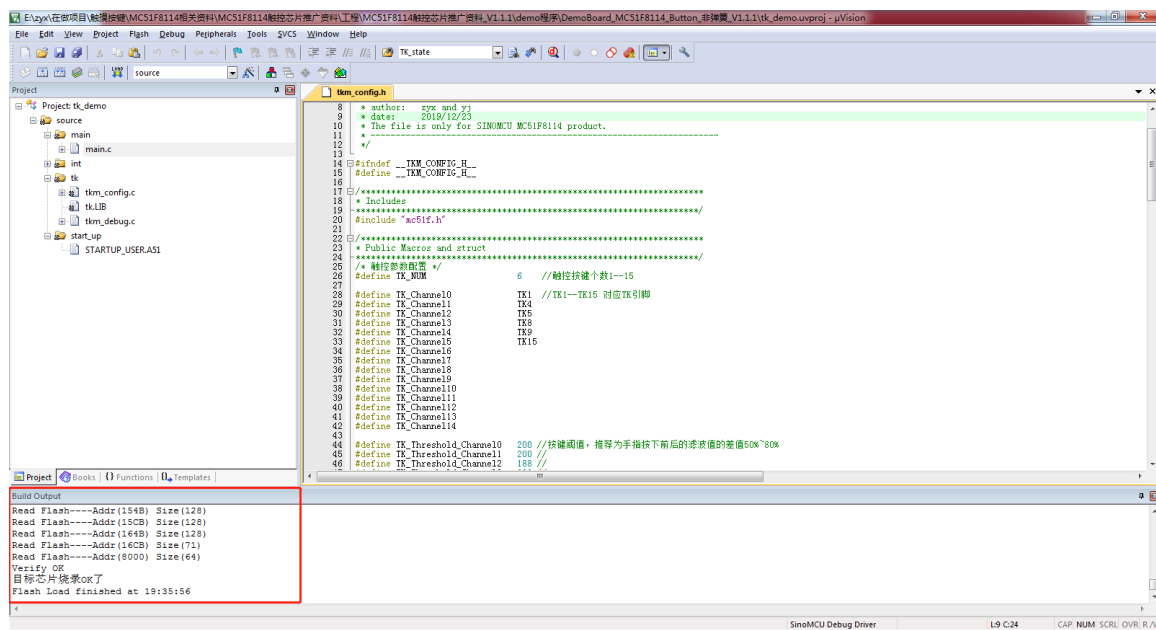
2 打开工程文件后如下图所示。



3 编译并运行程序，编译成功后提示信息如下图所示红框中所示。



4 烧录程序到 demo 板中，烧录成功后提示信息如下图红框中所示。



4.1.3 脱机运行程序

由于按键 S5 和 S6 的 LED 灯管脚和仿真器管脚复用，会导致出现按键触发时 LED 灯未被点亮的现象，建议拔除仿真器，使用 micro usb 线供电给 demo 板。如下图所示。



4.2 运行效果

程序烧录完毕开始运行，蜂鸣器会鸣叫一声作为响应。每个按键被触发时，蜂鸣器鸣叫，对应按键旁边的 LED 灯被点亮。demo 板按键触发时效果如下图所示。

1. 按键 S1 被触发



2. 按键 S2 被触发



3. 按键 S3 被触发



4. 按键 S4 被触发



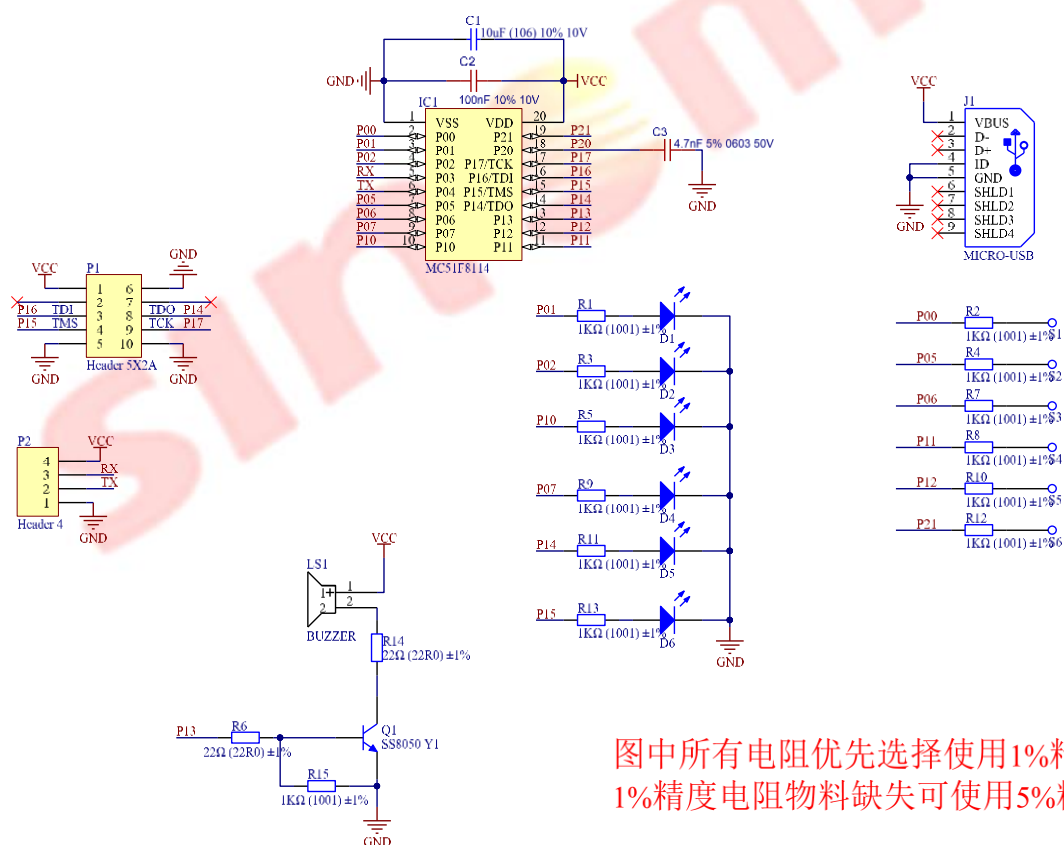
5. 按键 S5 被触发



6. 按键 S6 被触发



5 demo 板原理图



图中所有电阻优先选择使用1%精度
1%精度电阻物料缺失可使用5%精度电阻替换

6 触摸库快速入门

6.1 触摸库占用资源

- 1 ROM 用量：5.5K bytes 左右。
- 2 RAM 用量：180 bytes 左右。

6.2 触摸库运行环境

- 1 需安装 KEIL 51 平台（Keil uVision4 及以上版本）。
- 2 安装 SNLinkTK 安装包：SNLinkTK_KeilC51Driver_Vx.x.zip。

6.3 触摸库工作原理

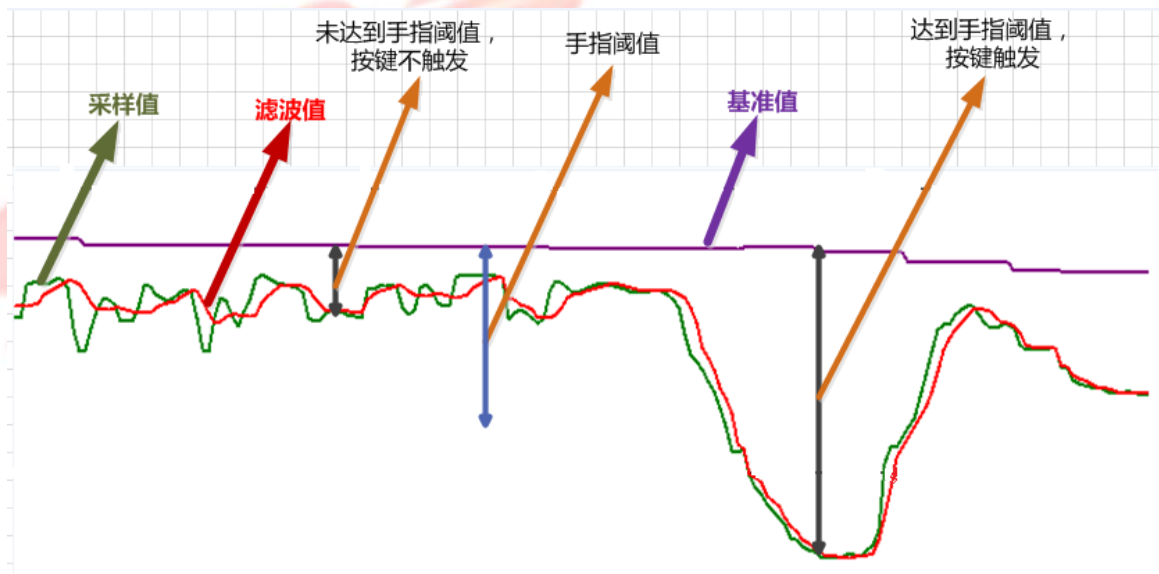
芯片 TK 模块工作过程中，当有按键触发时，TK 采集数值会突然减小，与按键触发前，TK 采集数值形成一定的差值，根据此差值判断按键触发。

由于外界环境的干扰导致 TK 采集数值产生的变化和上述描述基本一致，因此需要对 TK 采集数值进行软件滤波算法处理，还原数据的真实性。

为有效判断按键触发，需要引入 4 个基本概念：采样值、滤波值、基准值、手指阈值。

6.3.1 判键触发的 4 个基本概念

下图为实际触摸曲线图。



1 采样值

芯片 TK 模块工作过程中，TK 模块的采样数值。

2 滤波值

如上图所示，由于电容式触控按键易受外界环境干扰导致 TK 采集数值异常，因此需要将采样值进行滤波处理，排除干扰还原数据的真实性，此数据结果就是滤波值。

3 基准值

为有效判断按键触发，必须确保相对滤波值有一个较为稳定的参考对象（常态环境的数值），因此需要引入一个基准值概念，同时该曲线也需要根据环境进行自适应的变化（不受按键和噪声的影响），如上图所示。

4 手指阈值

如上图所示，通过设定手指阈值，确保每次手指触摸按键时变化的差值范围大于该数值，判断按键触发有效。

6.3.2 判键触发基本公式

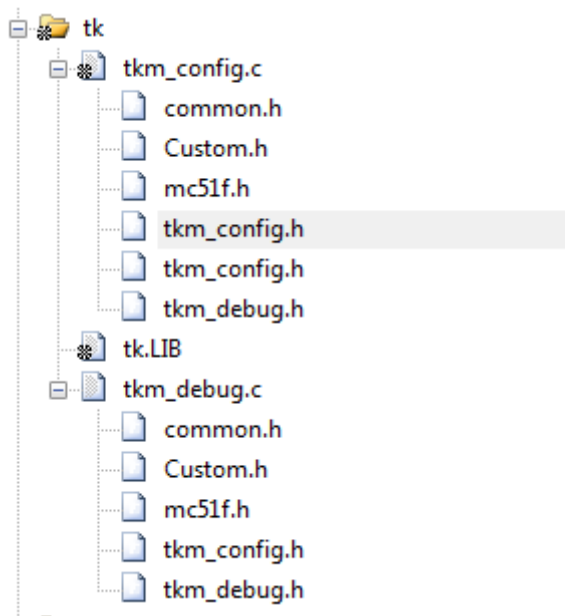
$$\Delta X = T_a - T_f \quad (1)$$

$$\Delta X > T_s \quad (2)$$

式中：

- 1 T_a 为基准值；
- 2 T_f 为滤波值；
- 3 T_s 为手指阈值。

6.4 触摸库文件结构



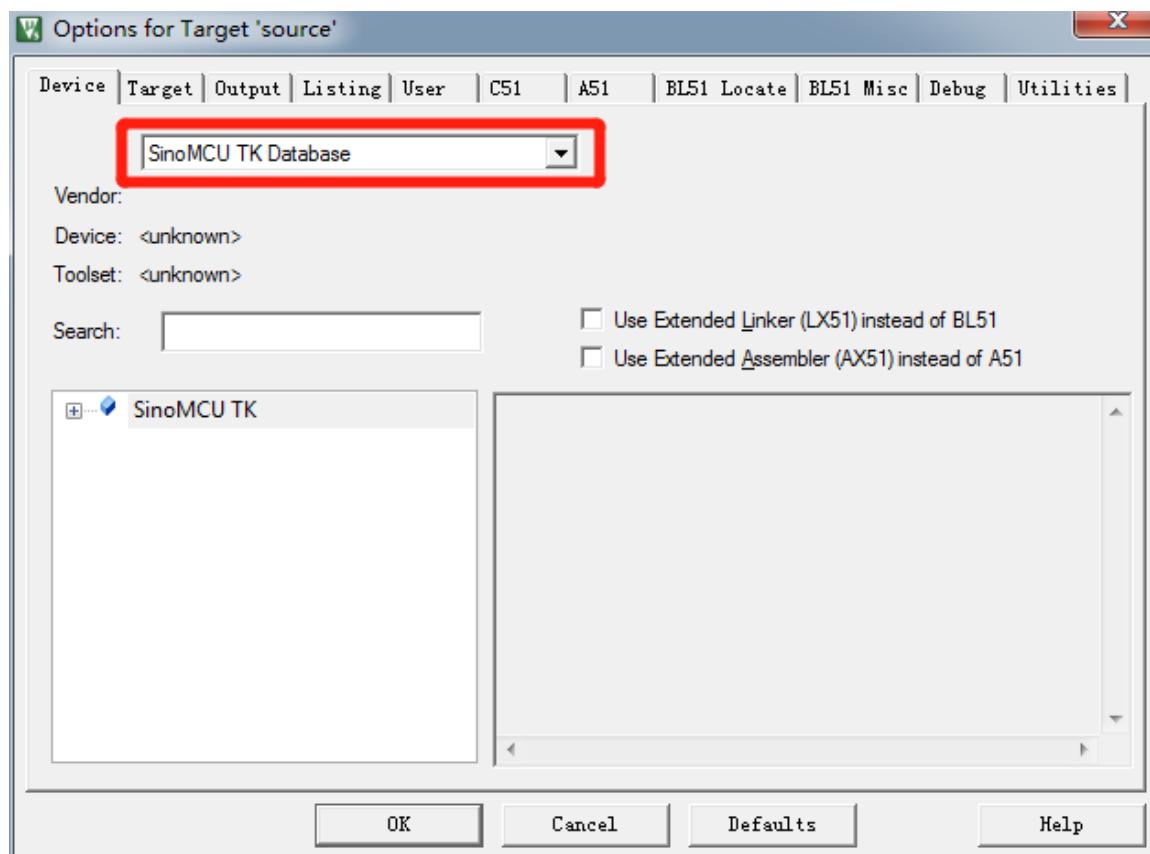
文件类型	文件名	文件功能
Include	tkm_config.h	TK 参数配置文件
	mc51f.h	仅包含 Custom.h 和 common.h 文件
	common.h	MC51F8114 芯片寄存器配置文件
	Custom.h	通用数据类型和宏定义文件
	tkm_debug.h	TK 与上位机工具通信配置文件
Source	tk.LIB	TK 算法库
	tkm_config.c	TK 参数配置的实现文件
	tkm_debug.c	TK 与上位机工具通信程序实现文件（占用 UART 资源）

6.5 新工程植入触摸库

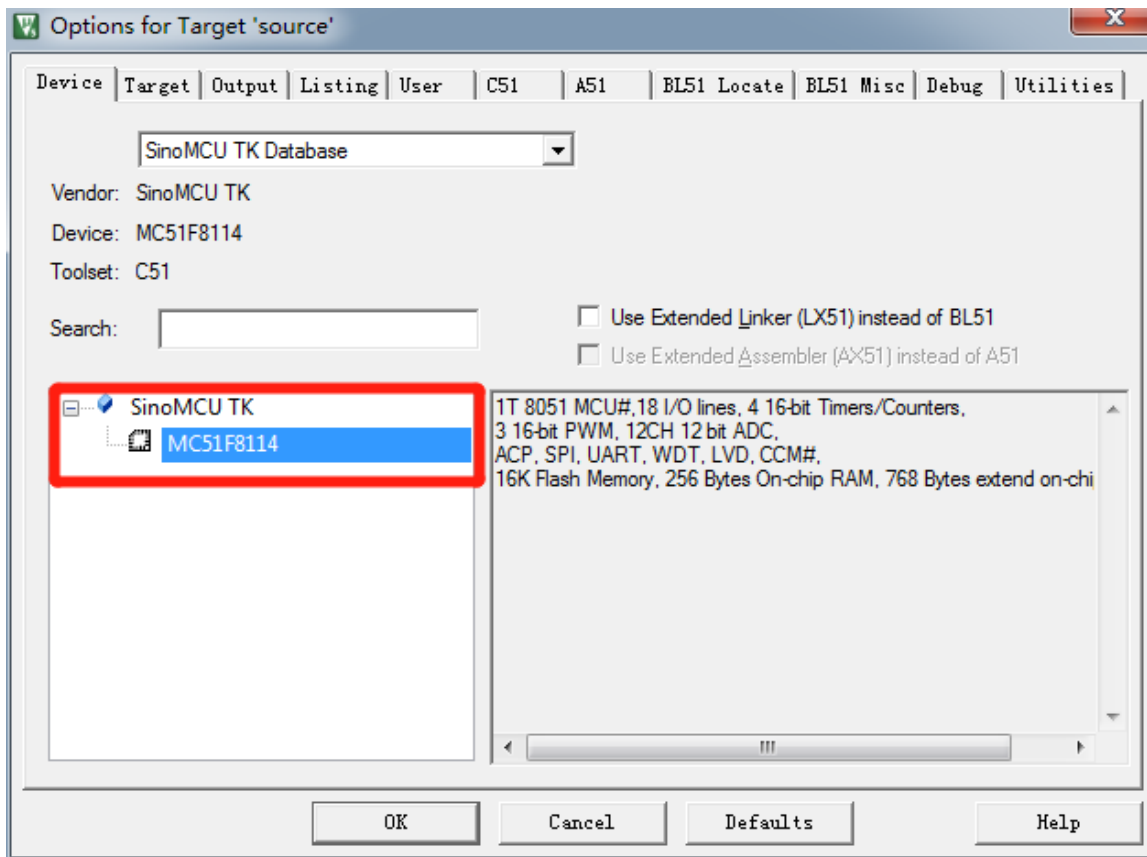
6.5.1 新建工程

1 Device 芯片类型

- 选择 SinoMCU TK Database



➤ 选择具体型号



2 SinoMCU_TK 文件路径检查

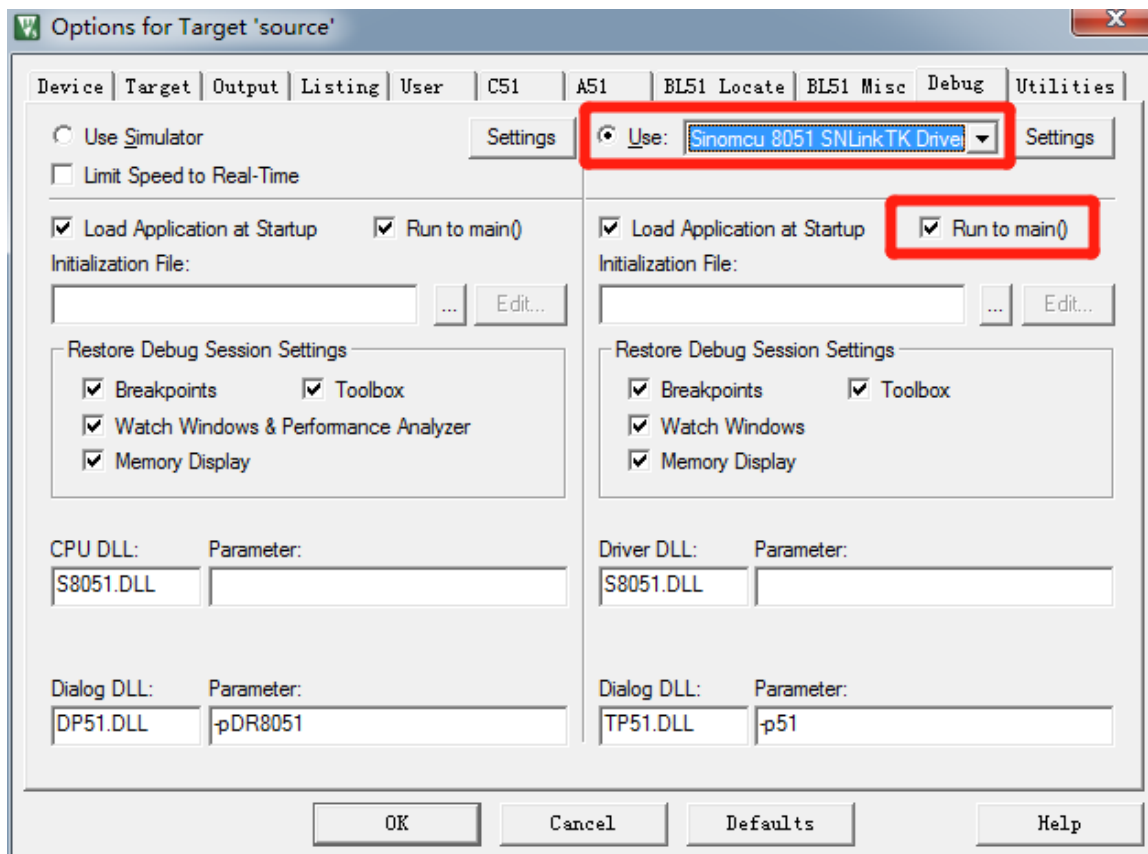
若 1 中“Device 芯片类型”已顺利执行，则可跳过此标题所述操作。若未查找到 Device 芯片，请执行下述操作。

如下图所示，确认 SinoMCU_TK 文件是否添加在 UV4、C51 同一级目录（均应该放在 keil 根目录下），若 SinoMCU_TK 文件不和 UV4、C51 在同一级目录，请将 SinoMCU_TK 文件剪切到与 UV4、C51 同一级目录下。然后请重新执行“Device 芯片类型”操作。

 C51	2019/9/27 14:16	文件夹
 SinoMCU_TK	2020/3/3 10:48	文件夹
 UV4	2020/2/26 13:49	文件夹

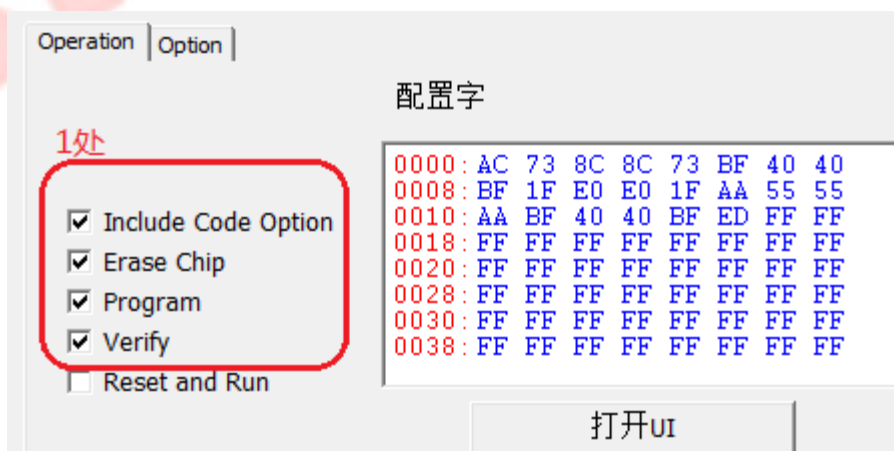
3 Debug 设置

1: 选择调试驱动, 如下图红框中标注, 选择 “Use” 单选按钮和 “Sinomcu 8051 SNLinkTK Driver”。另外如果勾选 “Run to main”, 在 C 语言调试时可直接跳到 main 函数。



2: 选中 “Sinomcu 8051 SNLinkTK Driver”, 后单击 Setting

“” 出现下图, 在 Operation 选项中如下图进行勾选。



3: 单击上图中 Option 选项, 如下图, 可根据实际进行配置, 按确定完成配置。

Setting Table

Operation Option

OSC_SEL 上电为内部时钟, 软件可进行配置 时钟选择

上电延时 128mS 上电延时配置, 可调整

☐ WDT使能控制 000000: AC F3 0C 0C F3 BF 40 40
000008: BF 1F E0 E0 1F FF 00 00
☐ BOR使能控制 000010: FF BF 40 40 BF ED FF FF

BOR电压点 2.1V

外部时钟 16M 外部时钟配置

振荡器电流 16M模式

反馈电阻 16M模式

☐ 调试功能关闭 ☐ 停振检测使能

RSTN复用 IO功能 P00管脚可作为RSTN使用

POR滤波时间 200us滤波 根据实际情况选择内部滤波特性

BOR滤波时间 200us滤波

FLASH第12K~15K IAP可擦除可编程程序可读 FLASH使用IAP时, 保护关键程序

FLASH第8K~11K IAP可擦除可编程程序可读

FLASH第4K~7K IAP可擦除可编程程序可读

FLASH第0K~3K IAP可擦除可编程程序可读

测试模式 测试模式关闭 默认

主时钟滤波时间 20nS

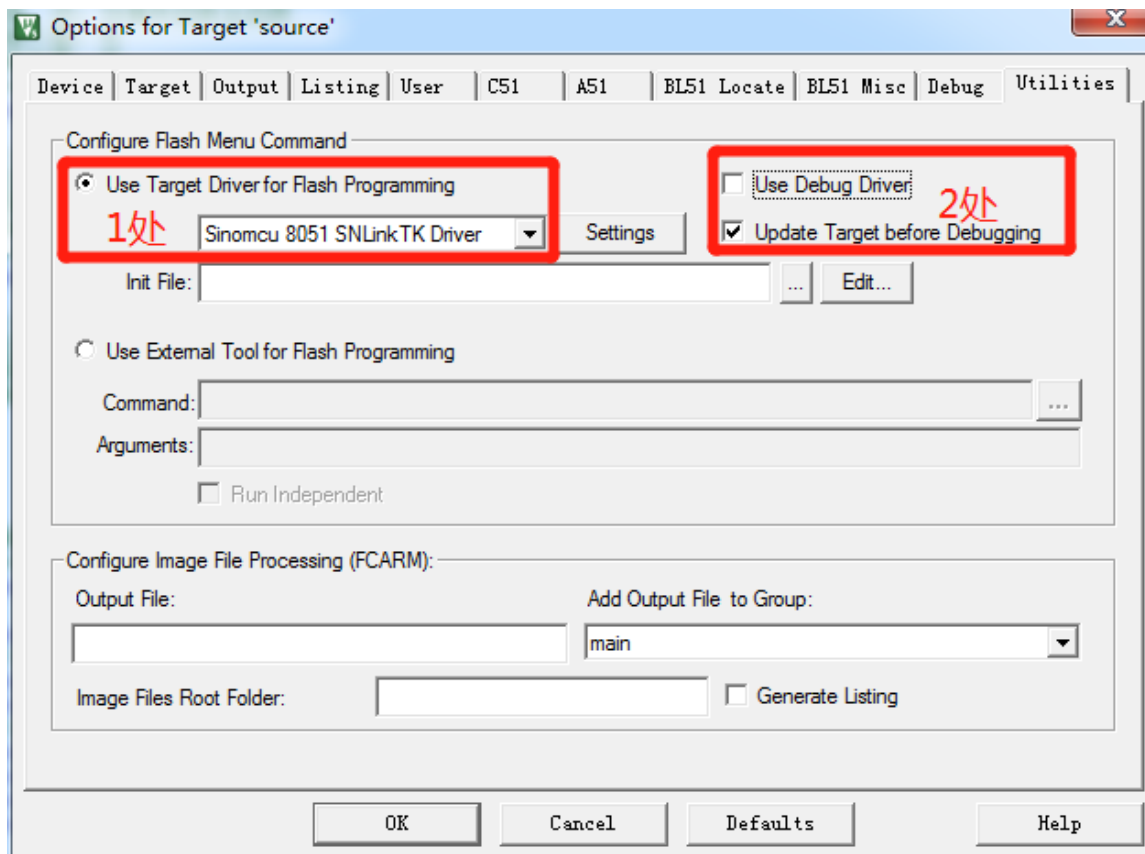
高功耗控制 高功耗模式在IDLE下关闭

校准字load方式 配置字load校准值

确定 取消 缺省值

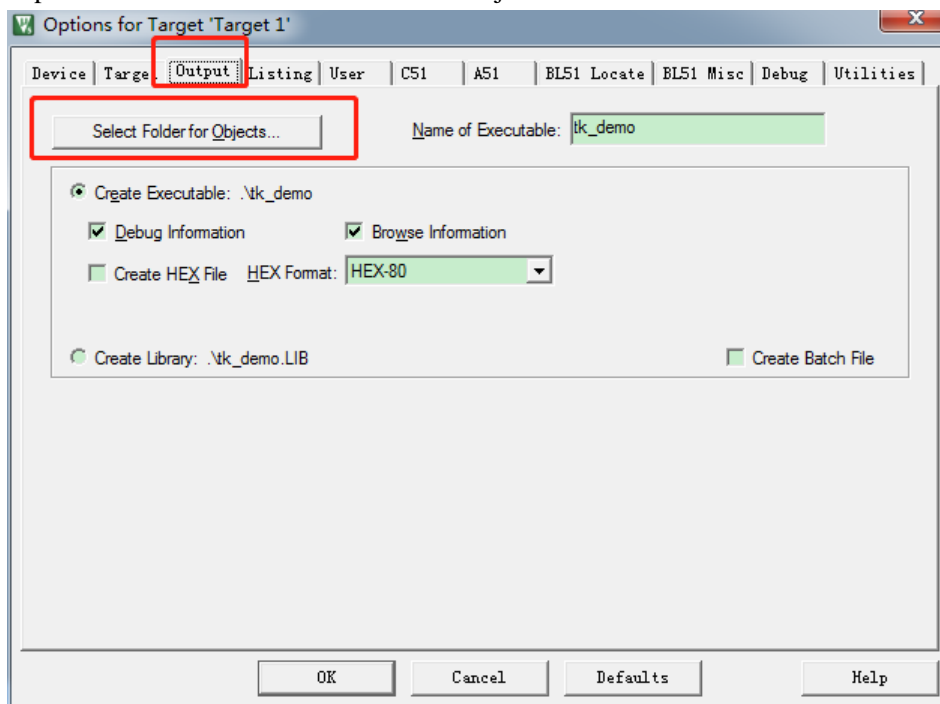
4 Utilities 设置

配置 Utilities, 1 处选择 “Use Target Driver for Flash Programming” 单选按钮和 “Sinomcu 8051 SNLinkTK Driver”; 2 处 “Use Debug Driver” 和 “Update Target before Debugging” 项为必选项。按 OK 完成配置。

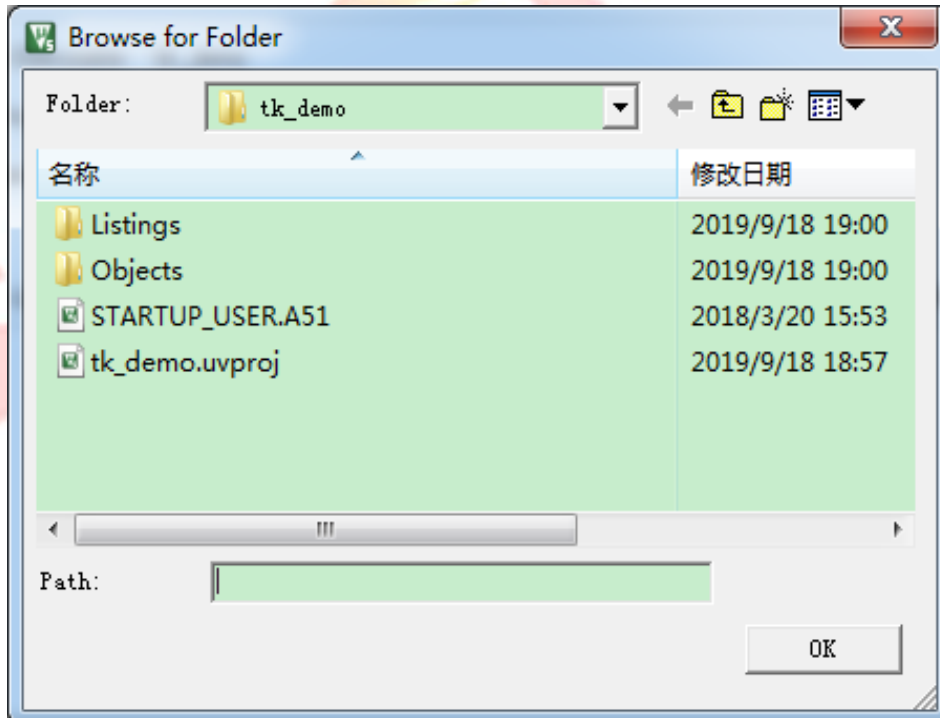


5 定义目标文件输出路径

- 1 选择“Output”选项，点击“Select Folder for Objects”。

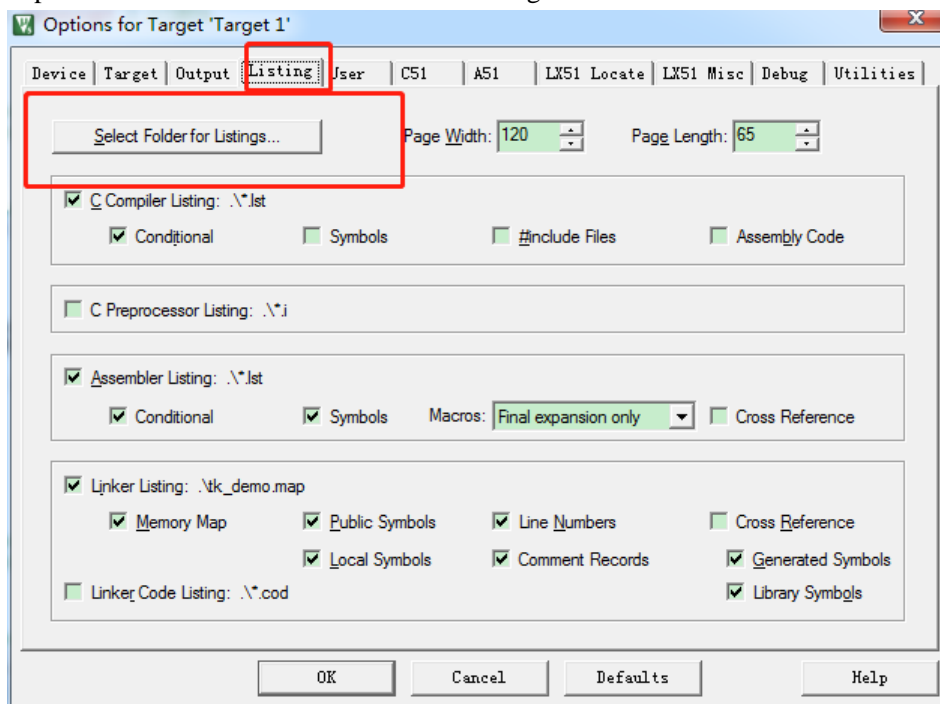


- 2 配置目标文件与 keil project 同一级目录下，否则会导致仿真器无法正常下载配置字，导致芯片无法正常运行。

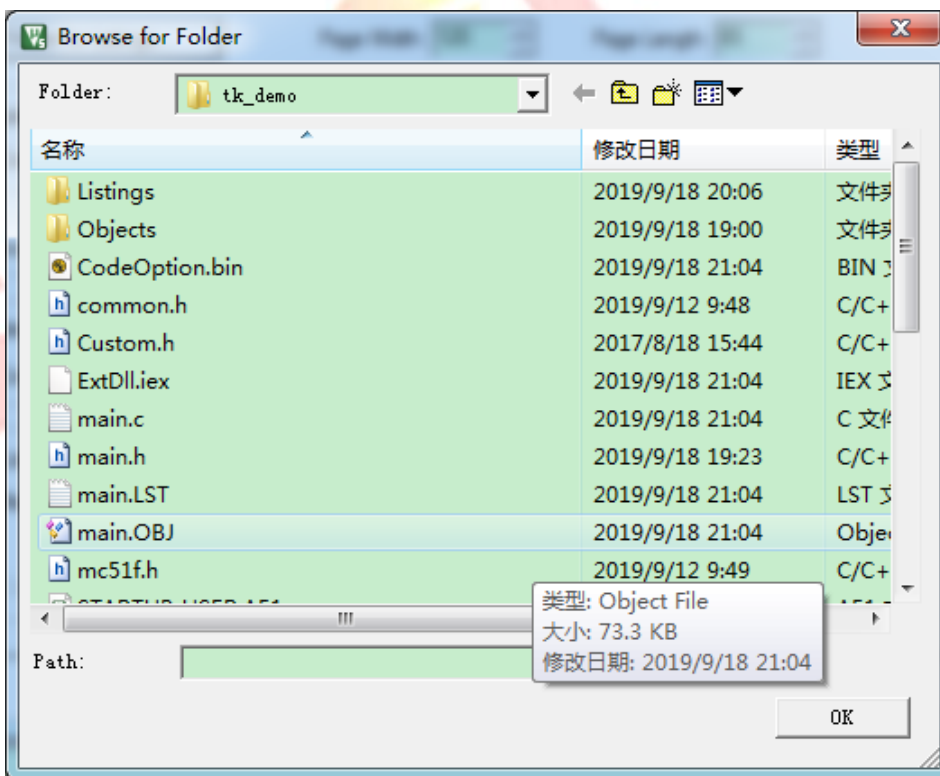


6 定义列表文件路径

- 1 选择“Output”选项，点击“Select Folder for Listings”。



- 2 将列表文件放置与 keil project 文件同一目录。否则会导致仿真器无法正常下载配置字，导致芯片无法正常运行。



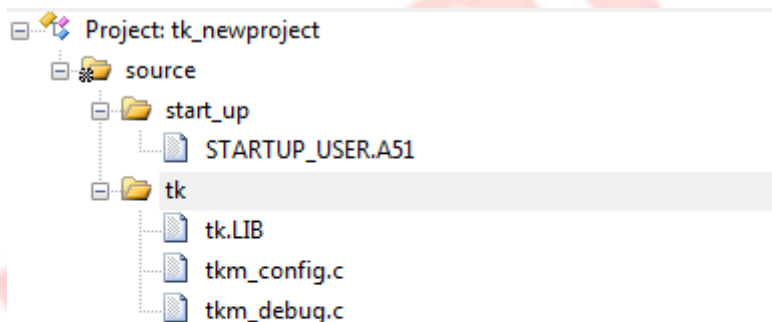
6.5.2 移植触摸库资源

- 1 将 Include、Source\tk 文件夹下所有文件和 main 文件夹下 STARTUP_USER.A51 启动文件（若用户新工程中已有启动文件则无需添加此文件）移植到新工程文件夹中，如下图红框中标注所示。

名称	修改日期	类型	大小
Listings	2019/12/12 13:36	文件夹	
Objects	2019/12/12 13:37	文件夹	
common.h	2019/9/12 9:48	C/C++ Header	55 KB
Custom.h	2017/8/18 15:44	C/C++ Header	10 KB
mc51f.h	2019/9/12 9:49	C/C++ Header	1 KB
STARTUP_USER.A51	2019/11/13 11:04	A51 文件	8 KB
tk.LIB	2019/12/11 18:57	Object File Library	16 KB
tk_newproject.uvproj	2019/12/12 13:36	新建工程文件	0 KB
tkm_config.c	2019/12/12 9:28	C 文件	13 KB
tkm_config.h	2019/12/12 10:16	C/C++ Header	8 KB
tkm_debug.c	2019/12/12 9:30	C 文件	10 KB
tkm_debug.h	2019/12/12 10:17	C/C++ Header	3 KB

建立的新工程文件

- 2 将工程下.c 和 tk.LIB 以及.A51 启动文件（若用户新工程中已添加启动文件则无需添加此文件）添加到工程中，如下图所示。



- 3 建立 main.c 和 main.h 文件，并在 main.h 中包含触摸库相关头文件，如下图所示。

```

1 /*
2 * -----
3 * Copyright (C) 2019 SINOMCU
4 * All Rights Reserved
5 * -----
6 * filename: main.h
7 * version: v1.0.2
8 * author: zyx and yj
9 * date: 2019/12/11
10 * The file is only for SINOMCU MC51F8114 product.
11 * -----
12 */
13
14 #ifndef __MAIN_H__
15 #define __MAIN_H__
16
17 /*****
18 * Includes
19 *****/
20 #include "mc51f.h"
21 #include "tkm_config.h"
22 #include "tkm_debug.h"
23 /*****
24 * Public Macros and struct
25 *****/
26 #define clr_WDT {WDT_OP = 0x5a;}
27
28 /*****
29 * Public variables
30 *****/
31
32
33
34 /*****
35 * Public functions
36 *****/

```

- 4 新工程 main.c 中需要包含如下图所示函数。

- 1 TKInit()函数，进行 TK 模块初始化操作。
- 2 Tk_service()函数，对 TK 按键进行算法处理进行按键的触发判断。
- 3 App_Process()函数，对触发的 TK 按键进行响应，例如点灯或者蜂鸣器鸣叫等。

```

*****/
*****/
函数名: void main(void)
描 述: 主函数
输入值: 无
输出值: 无
*****/
*****/
void main(void)
{
    TKInit();
    while (1)
    {
        clr_WDT;
        Tk_service();
        App_Process();
    }
}

```

- 5 将 TKIntProcess() 放置到 TK 中断服务子程序中如下图所示，进行获取 TK 模块采样值。

```

/*****
函数名: void Int4Interrupt(void)
描 述: TK中断服务子程序
输入值: 无
输出值: 无
*****/
void Int4Interrupt(void) interrupt 9 //INT4
{
    TKIntProcess();
}
    
```

- 6 若用户需要通过 TK 库与上位机工具通讯进行调试，则可在 Uart 服务子程序中添加 ReceivePCData() 和 SendPCData() 如下图所示。

```

/*****
函数名: void Int3Interrupt()
描 述: Uart中断服务子程序
输入值: 无
输出值: 无
*****/
void Int3Interrupt() interrupt 8 //INT3
{
    #if UART_PC_COMMUNICATEFUNCTION == ON
        ReceivePCData();
        SendPCData();
    #endif
}
    
```

- 7 正确编译工程后，如下图所示。

```

Build Output
assembling STARTUP_USER.A51...
compiling tkm_config.c...
compiling tkm_debug.c...
compiling main.c...
compiling int.c...
linking...
Program Size: data=52.0 xdata=133 code=5556
".\Objects\tk_newproject" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:01
    
```


7 触摸库详细介绍

7.1 tkm_config.h 配置文件说明

7.1.1 按键通道设定

```
1  #define TK_NUM                6    // 触控按键个数 1--15
   选择使用的触控按键个数（本文档中使用 6 个触摸按键）。
2  #define TK_Channel0           TK1   // TK1--TK15 对应 TK 引脚
   #define TK_Channel1           TK4
   #define TK_Channel2           TK5
   #define TK_Channel3           TK8
   #define TK_Channel4           TK9
   #define TK_Channel5           TK15
   #define TK_Channel6
   #define TK_Channel7
   #define TK_Channel8
   #define TK_Channel9
   #define TK_Channel10
   #define TK_Channel11
   #define TK_Channel12
   #define TK_Channel13
   #define TK_Channel14
```

根据目标板实际原理图选择要使用的 TK 通道。（本文档配置使用了 TK1、TK4、TK5、TK8、TK9 以及 TK15 共 6 个通道，对于要使用的 TK 通道对应的芯片引脚，请参见 MC51F8114 用户手册）。

7.1.2 触控参数设定

```
1  按键阈值设定
   #define TK_Threshold_Channel0 240 //按键阈值，推荐为手指按下前后的滤波值的差值 50%~80%
   #define TK_Threshold_Channel1 240 //
   #define TK_Threshold_Channel2 240 //
   #define TK_Threshold_Channel3 240 //
   #define TK_Threshold_Channel4 240 //
   #define TK_Threshold_Channel5 240 //
   #define TK_Threshold_Channel6 75
   #define TK_Threshold_Channel7 75
   #define TK_Threshold_Channel8 75
   #define TK_Threshold_Channel9 75
   #define TK_Threshold_Channel10 75
   #define TK_Threshold_Channel11 75
   #define TK_Threshold_Channel12 75
   #define TK_Threshold_Channel13 75
```

```
#define TK_Threshold_Channel14 75
```

定义各通道的门限，门限设定原则为最终产品手触摸滤波值变化量（差值）的 50%~80%。

2 采样次数累加设定

```
#define TK_Samples_perscan 18 // 采样次数累加设定，推荐(10~20),不超过 16 位数据
```

1. 累加设定后 TK 采样结果不超过 16 位数据，否则请减小此数据。
2. 通过采样次数累加对采样数据进行放大（由于某些硬件环境下会导致滤波值和基准值差值过小，导致无法设置合理手指阈值，所以我们需要对数据进行放大，而目前 8114 类芯片没有放大系数的寄存器可供调节，因此需要对采样数值进行时间累加来间接达到放大的效果）。

3 基准值更新次数

```
#define TK_BaseSamples_perscan 30 // 基准值更新次数，推荐(10~30)，可设置(1~60)
```

基准值自动修正以适应环境，数值越大，基准值修正速度越慢。

4 按键按下灵敏度设定

```
#define TK_Threshold_press 10 // 8-12 按键按下灵敏度设定
```

按键响应灵敏度调节，越小，判断按键触发越灵敏。

5 按键松开灵敏度设定

```
#define TK_Threshold_release 10 // 8-12 按键松开灵敏度设定
```

按键离开灵敏度调节，越大，判断按键离开越灵敏。

6 连续按下的触发次数设定

```
#define TK_Debounce_press 3 // 连续按下的触发次数设定，推荐(3~5),可设置(1~10)
```

次数越多，按键的抗干扰能力越强。

7 连续离开的触发次数设定

```
#define TK_Debounce_release 3 // 连续离开的触发次数设定，推荐(3~5),可设置(1~10)
```

次数越多，按键的抗干扰能力越强。

8 滤波系数增量

```
#define TK_Filter_1st_Increase 32 // 滤波系数增量，推荐(4~32),可设置 (1~64)
```

越大，滤波值滤波幅度越小。

9 基准系数增量

```
#define TK_Average_1st_Increase 16 // 基准系数增量，推荐(4~32),可设置 (1~64)
```

越大，基准值滤波幅度越小。

7.1.3 寄存器设定

如下图所示为触摸按键寄存器参数设定。详细说明请查阅 MC51F8114 用户手册。

```

/*TouchKe寄存器配置=====开始=====*/
/*TK模块选择基准电源*/
#define TK_VKFS_VREF          0x40 //VREF2.5V作为TK模块基准电源
#define TK_VKFS_VDD          0x00 //VDD作为TK模块基准电源

/*TK检测阈值选择*/
#define TK_VRFS_2p0V          0x07 //TK检测阈值选择2V
#define TK_VRFS_1p6V          0x06 //TK检测阈值选择1.6V
#define TK_VRFS_1p4V          0x05 //TK检测阈值选择1.4V
#define TK_VRFS_1p2V          0x04 //TK检测阈值选择1.2V
#define TK_VRFS_1p0V          0x03 //TK检测阈值选择1.0V
#define TK_VRFS_0p8V          0x02 //TK检测阈值选择0.8V
#define TK_VRFS_0p6V          0x01 //TK检测阈值选择0.6V
#define TK_VRFS_0p2V          0x00 //TK检测阈值选择0.2V

/*TK 扫描时钟源 TK_SRC 选择*/
#define TK_CKS_PWM0           0x03 //选择 PWM0 波形输出
#define TK_CKS_HRC             0x02 //选择 HRC
#define TK_CKS_LRC             0x01 //选择 LRC
#define TK_CKS_SYS             0x00 //选择系统时钟 SYSCLK

/*TK 扫描时钟分频*/
#define TK_DIV_SEL             0x01 //范围:0~63 公式: FTKCLK = FTK_SRC/(TK_DIV_SEL + 1) × 2 意义: (TK_DIV_SEL+1)*2分频

/*TK滤波时间*/
#define TK_FLT_DIS             0 //关闭TK滤波
#define TK_FLT_SEL             0x03 //范围: 0~15 公式: TFLT = (TK_FLT_SEL + 1)/FTKCLK 意义: 计算TK滤波时间
/*TouchKe寄存器配置=====结束=====*/

```

7.2 函数及常用变量说明

7.2.1 函数功能说明

序号	函数名	函数功能
1	void TKInit(void)	触摸按键初始化子程序
2	void TK_Config_Init(...)	触摸按键配置参数初始化程序（用户不可访问）
3	TK_Interrupt_EN()	TK 中断初始化使能
4	void Tk_getbaselinedata(void)	上电获取触摸按键初始的滤波值和基准值(调用后会强行更新滤波值和基准值，用户不可访问)
5	void Tk_service(void)	TK 按键滤波处理（用户不可访问）
6	TKIntProcess()	TK 中断服务子程序（用户不可访问）
7	void Customer_Keystatejustpress(void)	触控任意通道被刚刚检测到按下状态
8	void Customer_Keystatejustrelease(void)	触控任意通道被刚刚检测到释放状态
9	void Customer_Tkdataready(void)	TK 数据刚刚完成一次更新
10	void Customer_Tkdatareadytoupdate(void)	数据刚刚采样完毕，准备对采样数据进行滤波处理
11	void App_Process(void)	根据触发的按键进行响应操作
12	void ReceivePCData(void)	接收上位机工具发送数据（用户需按上述说明将此函数放至指定位置）
13	void SendPCData(void)	向上位机工具发送数据（用户需按上述说明将此函数放至指定位置）

7.2.2 常用变量使用说明

1 采样数据

采样数据: TK_value_origin[TK_NUM] = {0};

采集 TK 通道上未经处理的数据。

2 滤波数据

滤波数据: TK_value_filter[TK_NUM] = {0};

对采样数据进行软件滤波算法后的数据。

3 基线数据

基线数据: TK_value_average[TK_NUM] = {0};

对滤波数据进行处理后的值，作为是否产生按键事件的基准值。

注: TK_value_average 与 TK_value_filter 对应的按键通道差值大于设置的按键阈值即判定按键触发。

7.3 TK 库调用

7.3.1 TK 库调用截图

```
/******  
函数名: void main(void)  
描 述: 主函数  
输入值: 无  
输出值: 无  
*****/  
void main(void)  
{  
    TKInit();  
    while (1)  
    {  
        clr_WDT;  
        Tk_service();  
        App_Process();  
    }  
}
```

7.3.2 TK 库调用说明

1 main 函数

1.1 调用 TkInit()函数进行 TK 模块初始化。TkInit()函数调用内容如下所示。

```
/******  
函数名: void TKInit(void)  
描 述: 触摸按键初始化子程序  
输入值: 无  
输出值: 无  
*****/  
void TKInit(void)  
{  
    VREF_C = 0xeb;  
#if UART_PC_COMMUNICATEFUNCTION == ON  
    tkm_DebugInit();  
#endif  
    TK_Config_Init(TK_NUM, TK_VTKS_VREF, TK_VRFS_1p6V, TK_CKS_SYS, TK_DIV_SEL, TK_FLT_SEL);  
    TK_Interrupt_EN();  
    Delaytime(2000);  
    Tk_getbaselinedata();  
}
```

- a) tkm_DebugInit()函数进行触摸库与上位机工具通讯初始化操作，用户可通过 tk_debug.h 中对“UART_PC_COMMUNICATEFUNCTION”进行宏体选择确定是否开启此功能，如下图所示。
- ON: 开启通讯功能。
- OFF: 关闭通讯功能。

```
/* TK与PC通讯功能模块（占用UART资源） */  
#define UART_PC_COMMUNICATEFUNCTION OFF // 通过上位机工具将触摸库数据（采样值、滤波值、基准值绘制为波形）ON/ OFF
```

b) TK_Config_Init(...)函数进行 TK 参数配置操作。具体 TK 参数如下。

```

/*TouchKe寄存器配置=====开始=====
/*TK模块选择基准电源*/
#define TK_VTKS_VREF          0x40 //VREF2.5V作为TK模块基准电源
#define TK_VTKS_VDD           0x00 //VDD作为TK模块基准电源

/*TK检测阈值选择*/
#define TK_VRFS_2p0V          0x07 //TK检测阈值选择2V
#define TK_VRFS_1p6V          0x06 //TK检测阈值选择1.6V
#define TK_VRFS_1p4V          0x05 //TK检测阈值选择1.4V
#define TK_VRFS_1p2V          0x04 //TK检测阈值选择1.2V
#define TK_VRFS_1p0V          0x03 //TK检测阈值选择1.0V
#define TK_VRFS_0p8V          0x02 //TK检测阈值选择0.8V
#define TK_VRFS_0p6V          0x01 //TK检测阈值选择0.6V
#define TK_VRFS_0p2V          0x00 //TK检测阈值选择0.2V

/*TK 扫描时钟源 TK_SRC 选择*/
#define TK_CKS_PWM0           0x03 //选择 PWM0 波形输出
#define TK_CKS_HRC             0x02 //选择 HRC
#define TK_CKS_LRC             0x01 //选择 LRC
#define TK_CKS_SYS             0x00 //选择系统时钟 SYSCLK

/*TK 扫描时钟分频*/
#define TK_DIV_SEL             0x01 //范围:0~63 公式: FTKCLK = FTK_SRC/(TK_DIV_SEL + 1)
/*TK滤波时间*/
#define TK_FLT_DIS              0 //关闭TK滤波
#define TK_FLT_SEL              0x03 //范围: 0~15 公式: TFLT = (TK_FLT_SEL + 1)/FTKCLK
/*TouchKe寄存器配置=====结束=====

```

需要手动设置数值

除具体红框标注需用户手动设置数值外，其余均可通过对应宏定义传参。

c) TK_Interrupt_EN()开启 TK 中断使能。

d) Tk_getbaselinedata()上电获取触摸按键初始的滤波值和基准值(调用后会强行更新滤波值和基准值)。

1.2 Tk_service()函数进行 TK 按键滤波处理。

1.3 App_Process()函数根据触发的按键进行响应操作。

如下图所示，while 循环中每次 Tk_service()运行完毕后可通过对 TK_state 变量相应 bit 是否为 1 判断按键触发。例：bit0 为 1 代表配置文件中 TK_Channel0 代表的 TK 按键触发。为 0 则未触发。以此对其它 bit 位进行类推。

```

/*****
函数名: void App_Process(void)
描 述: 根据触发的按键进行响应操作（用户可自定义，建议将TK_state
      赋值给临时变量，通过对临时变量的判断进行按键响应操作
输入值: 无
输出值: 无
*****/
void App_Process(void)
{
    uchar i = 0;
    uint TK_state_temp;
    TK_state_temp = TK_state;
    for (i = 0; i < TK_NUM; i++)
    {
        if (TK_state_temp & ((uint)1 << i))
        {
            /* 根据触发的按键进行APP响应操作 */
        }
    }
}

```

2 TK 中断函数

调用 TKIntProcess()函数扫描触控通道，获取触摸按键采样值。

7.4 触摸库调试步骤

1. 配置按键个数 (TK_NUM)

如图所示在 tkm_config.h 中红框标注设置使用的触摸按键个数。

```

13
14 #ifndef TKM_CONFIG_H
15 #define TKM_CONFIG_H
16
17 /*****
18 * Includes
19 *****/
20 #include "mc51f.h"
21
22 /*****
23 * Public Macros and struct
24 *****/
25 /* 触控参数配置 */
26 #define TK_NUM 6 //触控按键个数1--15
27
28 #define TK_Channel0 TK1 //TK1--TK15 对应TK引脚
29 #define TK_Channel1 TK4
30 #define TK_Channel2 TK5
31 #define TK_Channel3 TK8
32 #define TK_Channel4 TK9
33 #define TK_Channel5 TK15
34 #define TK_Channel6
35 #define TK_Channel7
36 #define TK_Channel8
37 #define TK_Channel9
38 #define TK_Channel10

```

2. 配置使用的触摸按键通道 (TK_Channel(n),n=0~14)

如下图所示，根据 MC51F8114 用户手册设置使用的触控通道，如图本例中使用的第一个触控通道为 TK1 代表的管脚。

```

#define TK_Channel0 TK1 // TK1--TK15 对应TK引脚
#define TK_Channel1 TK4
#define TK_Channel2 TK5
#define TK_Channel3 TK8
#define TK_Channel4 TK9
#define TK_Channel5 TK15
#define TK_Channel6
#define TK_Channel7
#define TK_Channel8
#define TK_Channel9
#define TK_Channel10
#define TK_Channel11
#define TK_Channel12
#define TK_Channel13
#define TK_Channel14

```

3. 配置按键采样次数 (TK_Samples_perscan)

推荐采样值在 9500~10500 之间。用户可根据按键按下前后滤波值变化量进行设置，若按键前后滤波值变化较小（例滤波值变化不超过 100），可增加按键采样次数。按键按下前后阈值变化越大，按键触发判断越准确。

4. 配置按键阈值 (TK_Threshold_Channel(n),n=0~14)

设置按键阈值的方法如下：

- 1) 记录按键按下前，要设置的对应通道滤波值。
- 2) 记录按键按下后，要设置的对应通道滤波值。
- 3) 取按键按下前后两次滤波值差值的 50%~80%作为对应按键通道按键阈值。

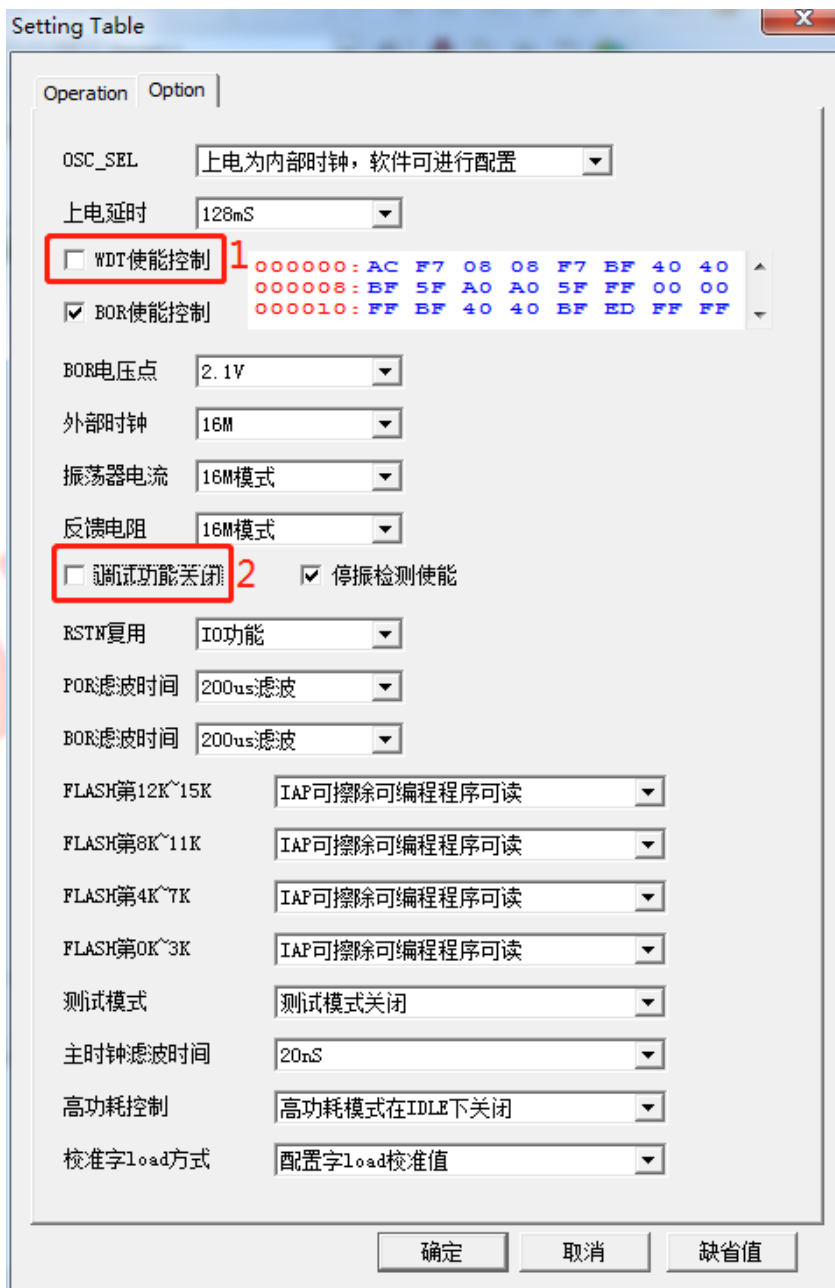
查看 TK 通道滤波值方法有 2 种，分别如下：

1. 通过我司开发 TouchKey 上位机软件，将数据更直观地以波形形式显示出来，具体使用方法，请参见本文档 § 8~10 章节说明。

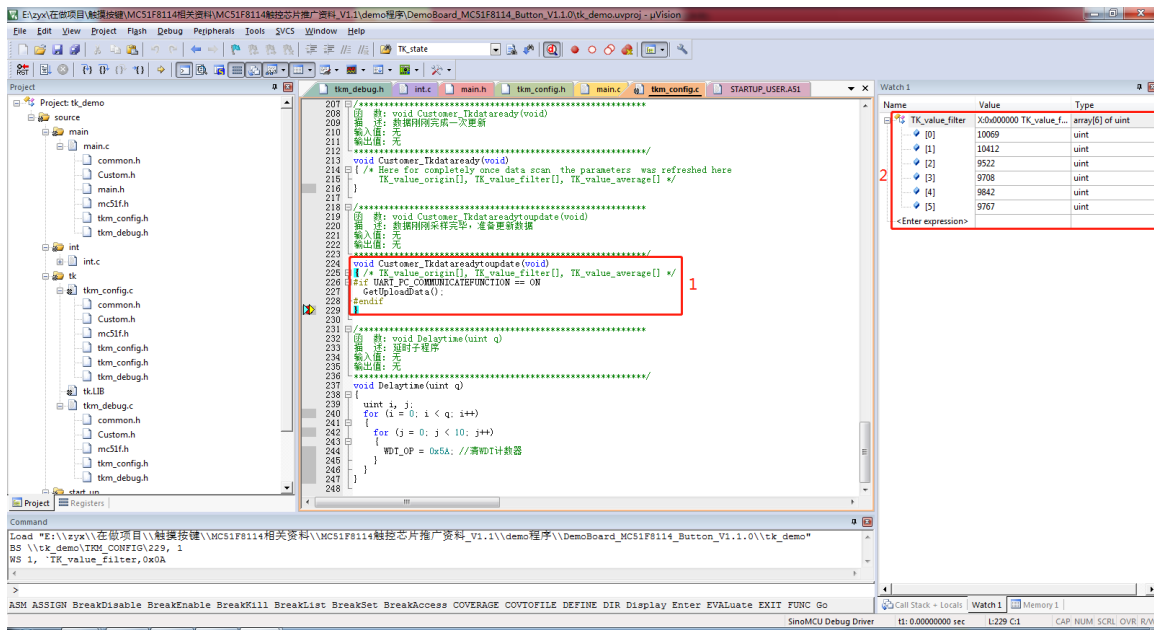
2. 进入仿真模式查看 TK 通道滤波值。具体方法如下：

(1) 进入仿真程序模式

为观察按键触发效果以及触摸按键相关数据，需进入仿真程序模式。若红框中标注选项已选择，请取消选中该选项（若要脱机运行，则需重新选中红框 2 中选项，红框 1 中选项使用户需求而确定是否选择）



(2) 如下图所示，在红框中 1 标注函数 void Customer_Tkdatareadytoupdate(void)中设置断点，查看右图中红框 2 标注按键滤波值 TK_value_filter。



设置后，重新编译下载后脱机运行程序，测试手指触摸按键效果。

5. 按键触摸频率调节

- 1) TK_Debounce_press (连续按下的触发次数设定，推荐 (3~5)，可设置 (1~10))：数值越小，判键按下越快
- 2) TK_Debounce_release (连续离开的触发次数设定，推荐 (3~5)，可设置 (1~10))：数值越小，判键离开越快
- 3) TK_Filter_lst_Increase (滤波系数增量，推荐 (4~32)，可设置 (1~64))：数值越大，判键频率越快，涉及滤波，设置后需测试

6. 按键触摸力度调节

- 1) TK_Threshold_press (8-12 按键按下灵敏度设定)：数值越小按键越灵敏
- 2) TK_Threshold_release (8-12 按键松开灵敏度设定)：数值越大按键越灵敏

7. 按键长按效果调节

TK_BaseSamples_perscan (基准值更新次数，推荐 (10~30)，可设置 (1~60))：数值越大，长按效果越好，同时下次判断按键按下时间会加长

8 TouchKey 上位机简介

我司提供了专门的触控按键电脑界面软件 SinoMCU TK Debugger，方便用户通过一系列的人机交互完成调试工作。用户通过此软件配合 MC51F8114 芯片开发的触摸库，可迅速直观地找到用户 PCB 最合适的触控按键参数，简化对触控按键的开发工作的难度。为方便用户更好使用我司 SinoMCU TK Debugger 软件，本文将对 SinoMCU TK Debugger 软件使用进行简要说明。

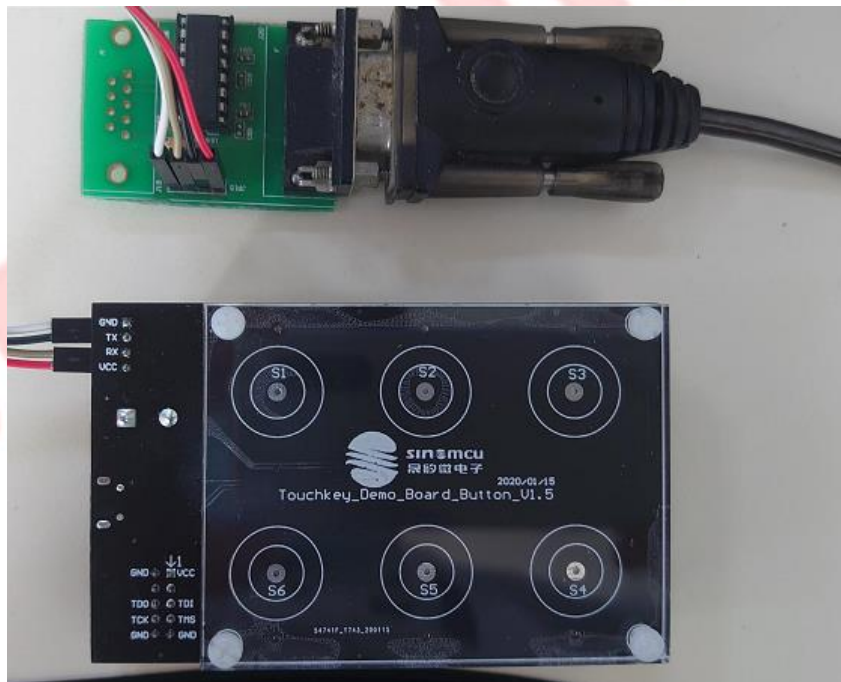
9 软件运行环境

9.1 硬件连接

1 连接 demo 板与串口通讯工具，连接关系如下：

- (1) demo 板 VCC 连接串口工具 VCC；
- (2) demo 板 RX 连接串口工具发送端；
- (3) demo 板 TX 连接串口工具接收端；
- (4) demo 板 GND 连接串口工具 GND。

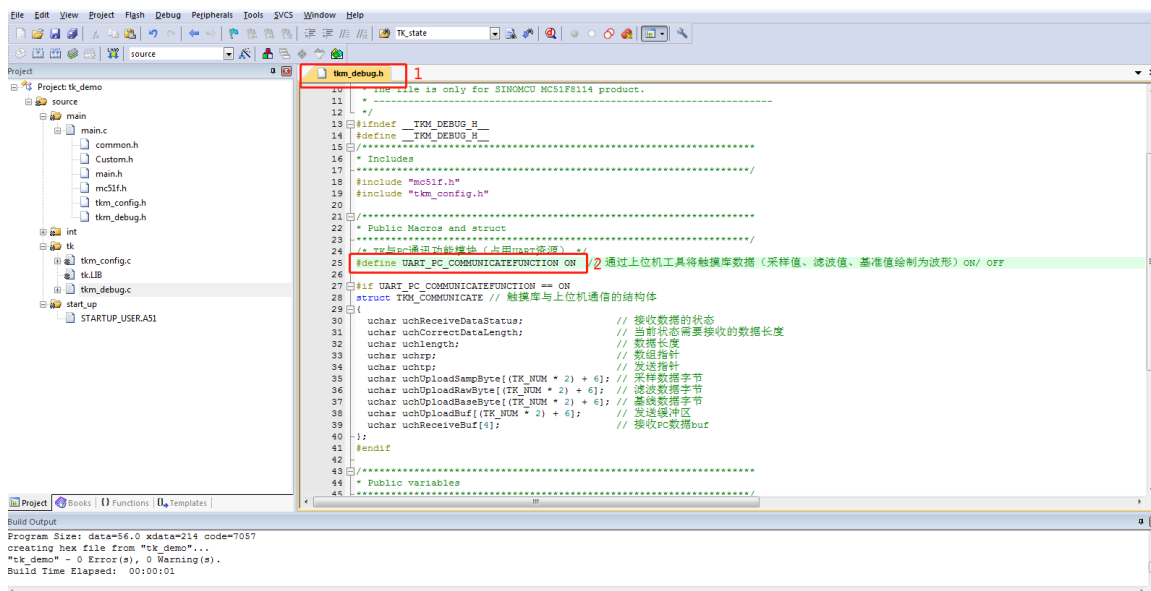
然后将串口通讯工具连接到电脑上。连接图示如下。



2 通过 micro usb 线给 demo 板供电。

9.2 软件运行

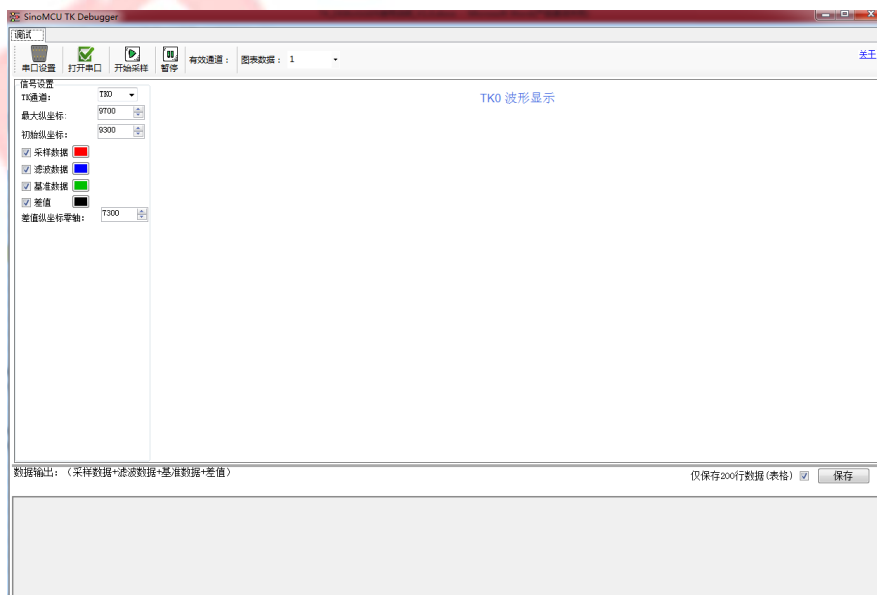
- ◆ 支持的操作系统：Win XP 及以上版本；
- ◆ CPU 主频：1.6GHz，推荐 2.2GHz 以上；
- ◆ PC 软件环境：Net Framework4.0。
- ◆ demo 板程序环境：在我司 demo 工程中打开 tkm_debug.h 文件，打开上下位机通讯开关如下图红框中 2 标注所示。



编译并下载程序到 demo 板中。

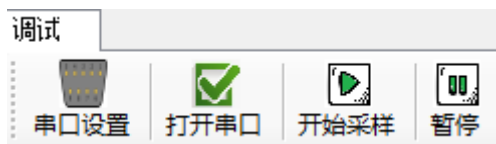
本软件无需安装，直接点击 TK_Debugger.exe 运行应用程序启动软件。

双击应用程序，启动软件，进入如下页面。



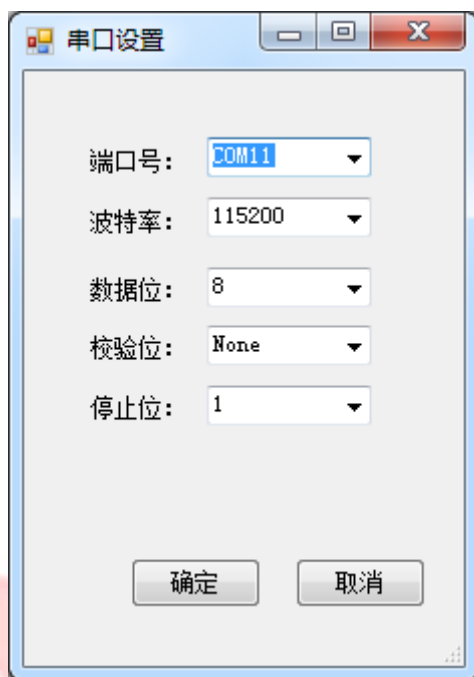
10 软件使用

调试页面有串口设置及调试控制几个按钮。

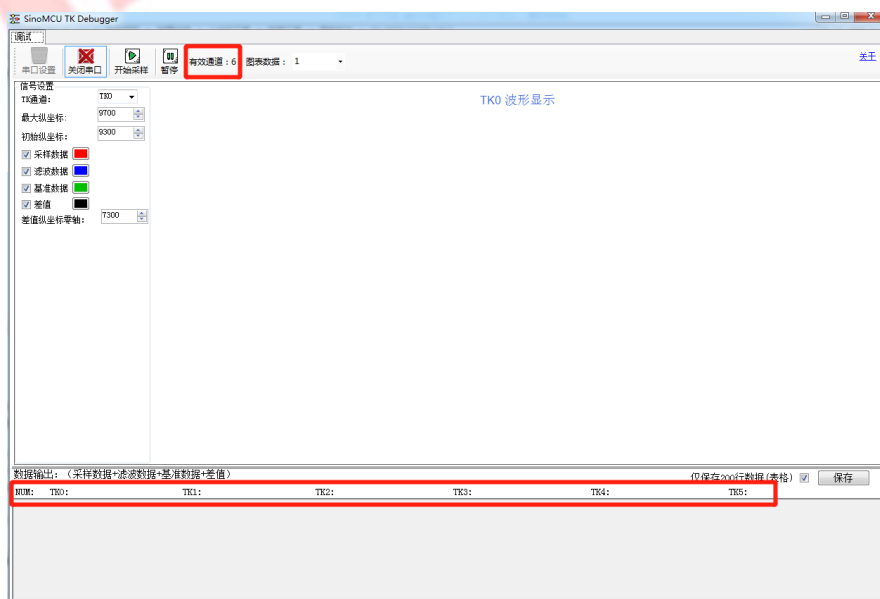


10.1 串口设置

进行调试前先要配置串口。点击“串口设置”按钮打开串口设置窗口，依次设置端口名、波特率、数据位、校验位和停止位，点击确定。再点击“打开串口”按钮以打开串口。串口打开后无法再配置串口，必须先关闭已打开的串口。



“打开串口”成功后将显示用户所配置 TK 通道。因 demo 板中配置使用 6 个 TK 通道，因此显示下图中红框中标注显示有效通道数：6，以及 TK0~TK5 等 6 个有效通道。



10.2 开始采样

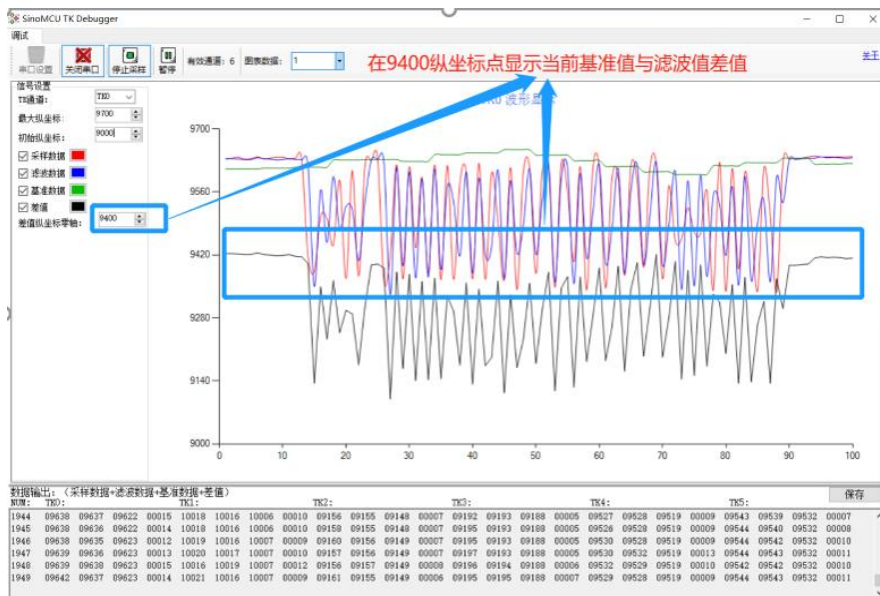
点击“开始采样”后，软件将与固件持续进行通讯，读取当前触控通道状态，并以图表形式显示。

调试时可配置显示不同图表数量：主界面显示的图表数量，支持 1、2、3、4、6、9 等不同个数图表。设置后，相应数量的图表将排列在主界面中，每个图表可以分别显示不同的数据。如下图所示显示 6 个图表。



10.3 图表显示设置

- ◆ **TK 通道：**选择当前需要显示的 TK 通道。可选通道由“打开串口”时，由软件向固件查询而来。
- ◆ **最大纵坐标：**显示数据的上限值。
- ◆ **初始纵坐标：**显示数据的下限数值。可通过最大纵坐标和初始纵坐标放大或缩小当前显示波形。
- ◆ **采样数据：**显示触控通道中的采样值。
- ◆ **滤波数据：**显示触控通道的滤波值。
- ◆ **基准数据：**显示触控通道的基准值。
- ◆ **差值：**显示当前触控通道基准值与滤波值的差值。
- ◆ **差值纵坐标零轴：**如下图所示，通过调节此参数，可以更直观的观察基准值与滤波值差值曲线，从而了解当前触控按键状态。

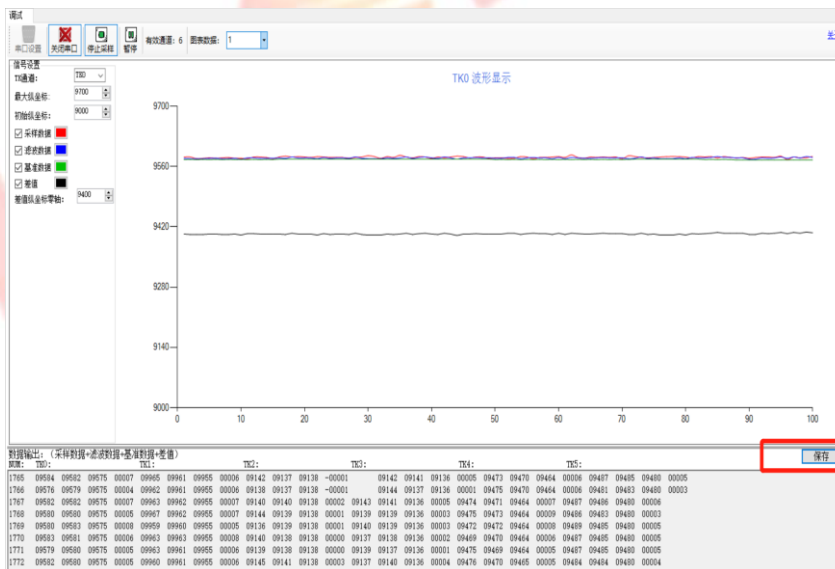


用户可调节图表中的采样数据、滤波数据、基准数据以及差值绘制曲线的颜色，从而设置便于用户观察的绘制曲线。

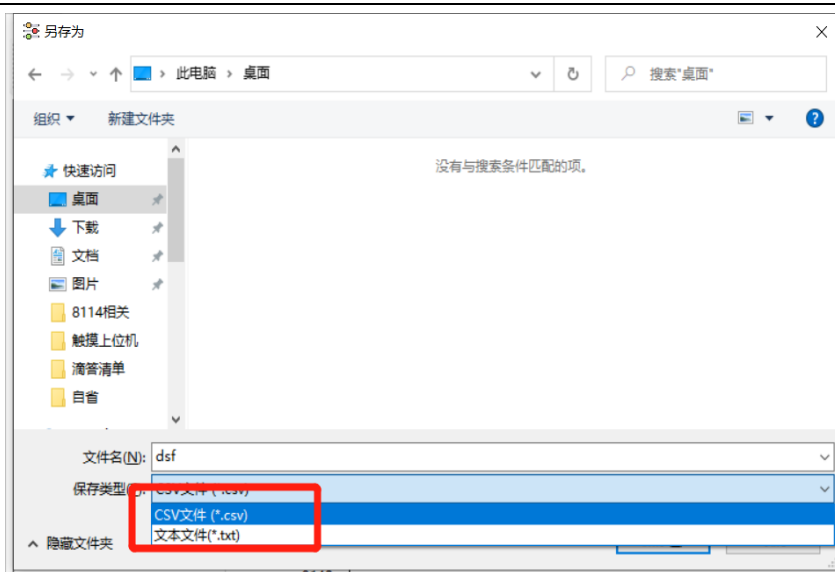
启动后用户可以选择“停止采样”或“暂停”。暂停后再恢复，之前的数据保留；停止后再启动，之前的数据将清除。

10.4 数据保存

如下图所示，点击红框标注“保存”按钮，可对当前数据进行保存。



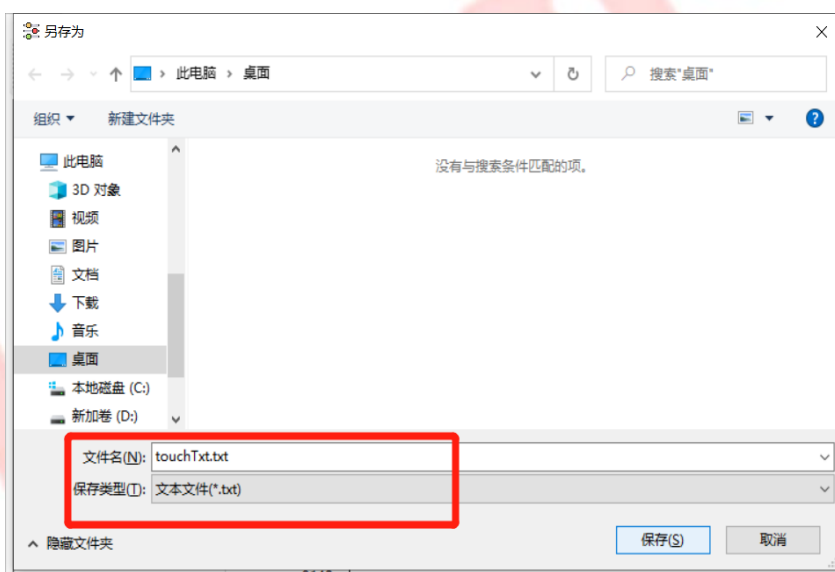
用户可选择“txt”文本格式或者“csv”文件格式保存数据。



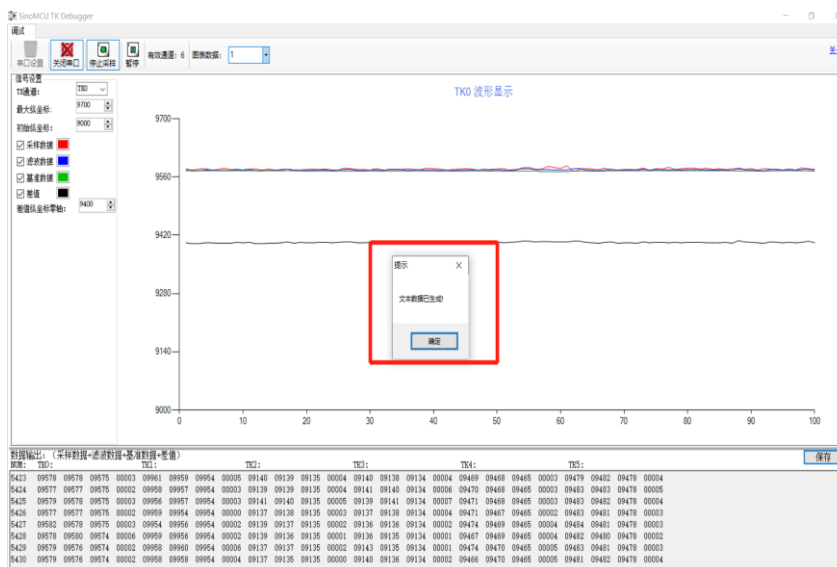
下面将对两种保存数据格式分别介绍。

10.4.1 “txt” 文本格式

1. 选择以“txt”文本格式保存数据。如下图所示。



2. 保存成功后，软件弹窗提示用户“文本数据已生成”。

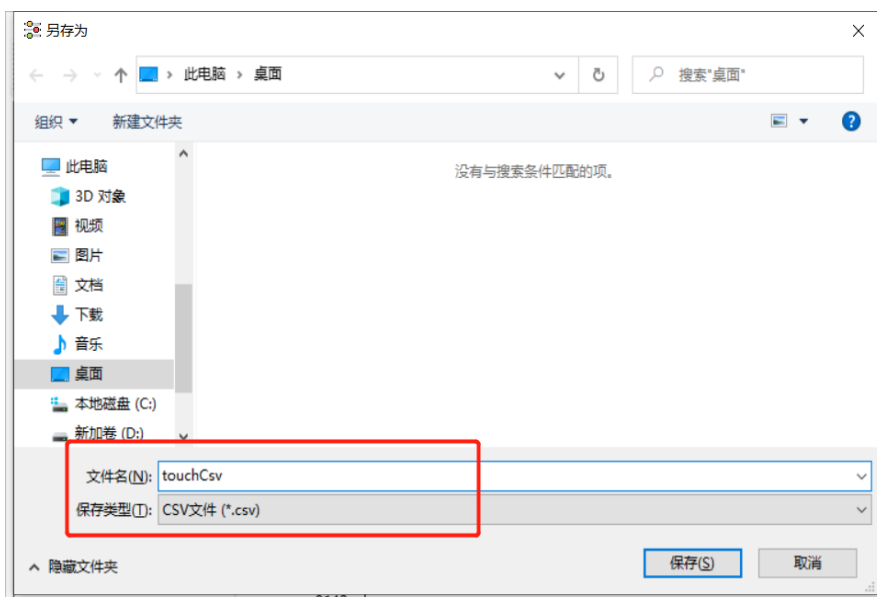


3. 打开文件，保存内容如下图所示。

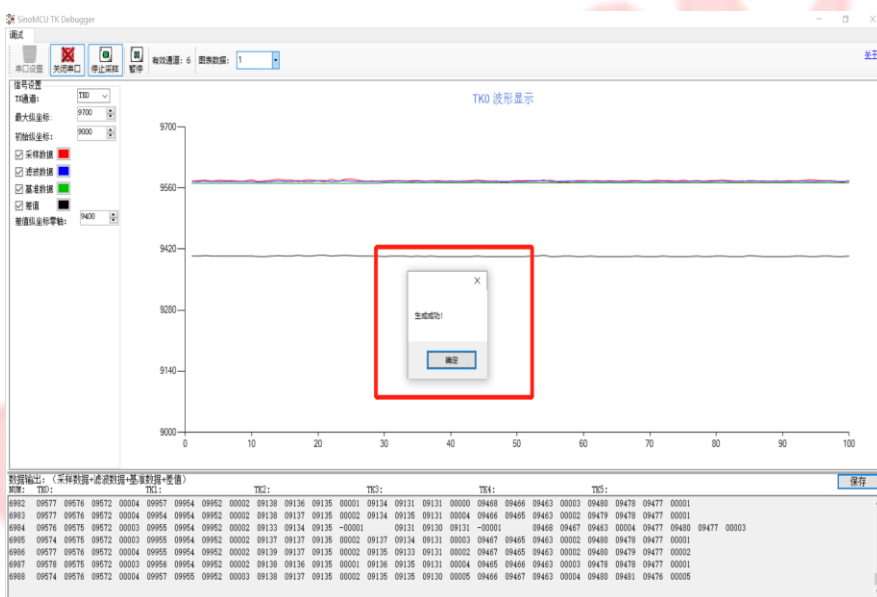
NUM		TK1										TK2										TK3										TK4										航控通函	
NUM		TK1	差值									差值									差值									差值													
0	0915	0915	0915	0910	0005	10001	09998	09993	00005	09174	09170	09171	-00001	09177	09174	09171	00003	09510	09506	09501																							
1	0920	0917	0910	0007	09996	09996	09993	00003	09173	09172	09171	00001	09173	09170	09170	00000	09505	09505	09504	09501																							
2	0920	0918	0910	0008	09996	09995	09993	00002	09173	09172	09171	00001	09173	09170	09170	00000	09505	09505	09504	09501																							
3	09615	09619	09610	00006	10000	09995	09993	00002	09175	09171	09171	00000	09174	09169	09169	-00001	09507	09504	09504	09501																							
4	09615	09616	09610	00006	10000	09997	09994	00004	09174	09170	09171	-00001	09170	09172	09170	00000	09506	09506	09506	09501																							
5	09620	09613	09610	00003	09994	09997	09993	00004	09169	09173	09171	00002	09174	09170	09170	00000	09505	09507	09507	09501																							
6	09619	09617	09610	00007	09995	09994	09993	00001	09180	09172	09171	00001	09176	09170	09170	00000	09506	09506	09506	09501																							
7	09618	09616	09610	00006	10002	10000	09993	00007	09179	09173	09171	00002	09173	09171	09169	00002	09506	09508	09508	09501																							
8	09615	09617	09610	00007	09999	09995	09993	00002	09172	09171	09171	00000	09172	09171	09169	00002	09506	09506	09506	09501																							
9	09614	09617	09610	00007	09995	09996	09993	00003	09171	09172	09171	00001	09170	09171	09169	00002	09506	09509	09509	09501																							
10	09619	09614	09610	00004	10000	09996	09993	00003	09171	09171	09171	00000	09172	09171	09169	00002	09506	09506	09509	09501																							
11	09619	09615	09610	00005	09996	09996	09993	00005	09177	09170	09171	-00001	09173	09171	09169	00002	09507	09507	09507	09501																							
12	09616	09616	09610	00006	09994	09995	09993	00002	09171	09175	09171	00004	09168	09171	09169	00002	09506	09504	09504	09501																							
13	09619	09615	09610	00005	09995	09996	09993	00003	09175	09173	09171	00002	09172	09172	09169	00003	09507	09507	09507	09501																							
14	09613	09617	09610	00007	09995	09997	09993	00004	09173	09174	09171	00003	09170	09169	09169	00000	09503	09503	09503	09501																							
15	09620	09617	09610	00007	09997	09996	09993	00003	09177	09172	09171	00001	09171	09169	09169	00000	09508	09507	09507	09501																							
16	09620	09618	09610	00008	10002	09995	09993	00002	09173	09173	09171	00002	09177	09175	09169	00006	09502	09505	09505	09501																							
17	09619	09619	09610	00009	09999	09997	09993	00004	09173	09170	09171	-00001	09172	09171	09169	00002	09508	09506	09506	09501																							
18	09618	09615	09610	00005	09998	09997	09993	00004	09175	09170	09171	-00001	09173	09169	09169	00000	09505	09505	09505	09501																							
19	09620	09616	09610	00006	09999	09996	09993	00003	09176	09171	09171	00000	09169	09169	09169	00000	09507	09505	09505	09501																							
20	09621	09610	09610	00008	10002	09997	09993	00004	09174	09173	09171	00002	09170	09168	09169	-00001	09512	09506	09506	09501																							
21	09619	09616	09611	00005	10001	09996	09993	00003	09175	09172	09171	00001	09173	09168	09169	-00001	09508	09508	09508	09501																							
22	09617	09616	09611	00007	09998	09997	09993	00004	09175	09171	09171	00000	09172	09170	09169	00001	09506	09505	09505	09501																							
23	09616	09616	09611	00005	09998	09998	09993	00005	09173	09170	09171	-00001	09172	09170	09169	00001	09509	09507	09507	09501																							
24	09617	09616	09611	00005	09999	09995	09993	00002	09172	09173	09171	00002	09174	09173	09169	00004	09505	09507	09507	09501																							
25	09615	09616	09611	00005	09999	09996	09993	00003	09177	09174	09170	00004	09172	09172	09169	00003	09511	09504	09504	09501																							
26	09620	09616	09611	00005	09999	09994	09993	00001	09175	09172	09170	00002	09170	09172	09169	00003	09507	09506	09506	09501																							
27	09622	09615	09611	00004	09998	09997	09993	00004	09170	09173	09170	00003	09173	09172	09169	00003	09506	09506	09506	09501																							
28	09617	09615	09611	00004	09998	09997	09993	00003	09171	09175	09170	00004	09175	09172	09169	00003	09507	09505	09505	09501																							
29	09616	09617	09611	00006	09999	09995	09993	00002	09175	09175	09170	00005	09175	09172	09169	00003	09504	09504	09506	09501																							
30	09616	09614	09612	00002	09999	09999	09993	00006	09175	09174	09169	00005	09175	09173	09169	00004	09507	09506	09506	09501																							
31	09616	09612	09612	00004	09994	09996	09993	00003	09176	09175	09169	00006	09174	09174	09169	00005	09508	09508	09508	09501																							
32	09618	09617	09612	00005	09998	09997	09993	00004	09171	09172	09169	00003	09172	09172	09169	00003	09506	09504	09504	09501																							
33	09618	09617	09612	00005	09998	09997	09993	00004	09171	09172	09169	00003	09172	09172	09169	00003	09506	09504	09504	09501																							
第 1 页, 共 1 页																						100% Windows (Ctrl) - UTF-8																					

10.4.2 “csv” 文件格式

1. 选择以“csv”文件格式保存数据。如下图所示。



- 保存成功后，软件弹窗提示用户数据“生成成功”。



3. 打开文件后，保存内容如下图所示。

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	
1	TK0采样值	TK0滤波值	TK0基址值	TK0差值	TK1采样值	TK1滤波值	TK1基址值	TK1差值	TK2采样值	TK2滤波值	TK2基址值	TK2差值	TK3采样值	TK3滤波值	TK3基址值	TK3差值	TK4采样值	TK4滤波值	TK4基址值	TK4差值	TK5采样值	TK5滤波值	TK5基址值	TK5差值
2	9615	9615	9610	5	10001	9998	9993	5	9174	9170	9171	-1	9177	9174	9171	3	9510	9506	9501					
3	9620	9617	9610	7	9996	9996	9993	3	9173	9172	9171	1	9173	9170	9170	0	9505	9505	9501					
4	9620	9618	9610	8	9996	9995	9993	2	9173	9172	9171	1	9173	9170	9170	0	9505	9504	9501					
5	9615	9616	9610	6	10000	9995	9993	2	9175	9171	9171	0	9174	9169	9170	-1	9507	9504	9501					
6	9615	9616	9610	6	10000	9997	9993	4	9174	9170	9171	-1	9170	9172	9170	2	9506	9506	9501					
7	9620	9613	9610	3	9994	9997	9993	4	9169	9173	9171	2	9174	9170	9170	0	9505	9507	9501					
8	9619	9617	9610	7	9995	9994	9993	1	9180	9172	9171	1	9176	9170	9170	0	9506	9506	9501					
9	9618	9616	9610	6	10002	10000	9993	7	9179	9173	9171	2	9173	9171	9169	2	9506	9508	9501					
10	9615	9617	9610	7	9999	9995	9993	2	9172	9171	9171	0	9172	9171	9169	2	9506	9506	9501					
11	9614	9617	9610	7	9995	9996	9993	3	9171	9172	9171	1	9170	9171	9169	2	9506	9509	9501					
12	9619	9614	9610	4	10000	9996	9993	3	9171	9171	9171	0	9172	9171	9169	2	9506	9509	9501					
13	9619	9615	9610	5	9996	9998	9993	5	9177	9170	9171	-1	9173	9171	9169	2	9507	9507	9501					
14	9617	9616	9610	6	9994	9995	9993	2	9171	9175	9171	4	9168	9171	9169	2	9506	9504	9501					
15	9619	9616	9610	6	9996	9996	9993	3	9175	9173	9171	2	9172	9172	9169	3	9507	9505	9501					
16	9613	9617	9610	7	9995	9997	9993	4	9173	9174	9171	3	9170	9169	9169	0	9503	9508	9501					
17	9620	9617	9610	7	9997	9996	9993	3	9177	9172	9171	1	9171	9169	9169	0	9508	9507	9501					
18	9620	9618	9610	8	10002	9995	9993	2	9173	9173	9171	2	9177	9175	9169	6	9502	9505	9501					
19	9619	9619	9610	9	9999	9997	9993	4	9173	9170	9171	-1	9172	9171	9169	2	9508	9506	9501					
20	9618	9615	9610	5	9999	9997	9993	4	9175	9170	9171	-1	9173	9169	9169	0	9505	9505	9501					
21	9620	9616	9610	6	9998	9996	9993	3	9176	9171	9171	0	9169	9169	9169	0	9507	9505	9501					
22	9621	9618	9610	8	10002	9997	9993	4	9174	9173	9171	2	9170	9168	9169	-1	9512	9506	9501					
23	9619	9616	9611	5	10001	9996	9993	3	9175	9172	9171	1	9173	9168	9169	-1	9508	9508	9501					
24	9617	9618	9611	7	9998	9997	9993	4	9175	9171	9171	0	9172	9170	9169	1	9506	9505	9501					
25	9616	9616	9611	5	9998	9998	9993	5	9173	9170	9171	-1	9172	9170	9169	1	9509	9507	9501					
26	9617	9616	9611	5	9999	9995	9993	2	9172	9173	9171	2	9174	9173	9169	4	9505	9507	9501					
27	9615	9616	9611	5	9999	9996	9993	3	9177	9174	9170	4	9172	9172	9169	3	9511	9504	9501					
28	9620	9616	9611	5	9999	9994	9993	1	9175	9172	9170	2	9170	9172	9169	3	9507	9506	9501					
29	9622	9615	9611	4	9998	9997	9993	4	9170	9173	9170	3	9173	9172	9169	3	9506	9506	9501					
30	9617	9615	9611	4	9997	9995	9993	2	9175	9174	9170	4	9176	9173	9169	4	9509	9504	9501					

11 demo 板验证测试

demo 板经过以下测试：

1. EFT 测试

(1) 5KHz: -4000V~+4000V 测试；

(2) 100KHz: -4000V~+4000V 测试。

2. CS 动态 10V 0.15MHz~80MHz 测试。

12 版本及更新记录

版本号	更新日期	更新内容	人员
V1.0	2019-11-07	初版发布	张玉鑫
V1.1	2019-11-27	1 添加新版仿真器图示 2 添加 demo 板原理图	张玉鑫
V1.2	2019-12-12	1 §4.1.2 添加编译和烧录程序详细操作步骤	张玉鑫
V1.3	2019-12-23	1 §3.2 中添加仿真器 V02 和 demo 板连线步骤说明 2 §4.1.1 中针对 demo 板硬件不同对两个 demo 程序作区分说明	张玉鑫
V1.4	2020-02-27	1 修改§5 中 demo 板原理图	张玉鑫
V1.5	2020-03-03	1 修改§2 中对于 SLinkTK 插件安装包路径的叙述	张玉鑫
V1.6	2020-03-10	1 更新§4.1.1 中插图 2 添加§6 和§7 如何使用触摸库进行介绍 3 添加§8、§9 和§10 对如何使用我司 TouchKey 上位机工具进行说明 4 添加§11demo 板验证测试，对 demo 板性能进行说明	张玉鑫