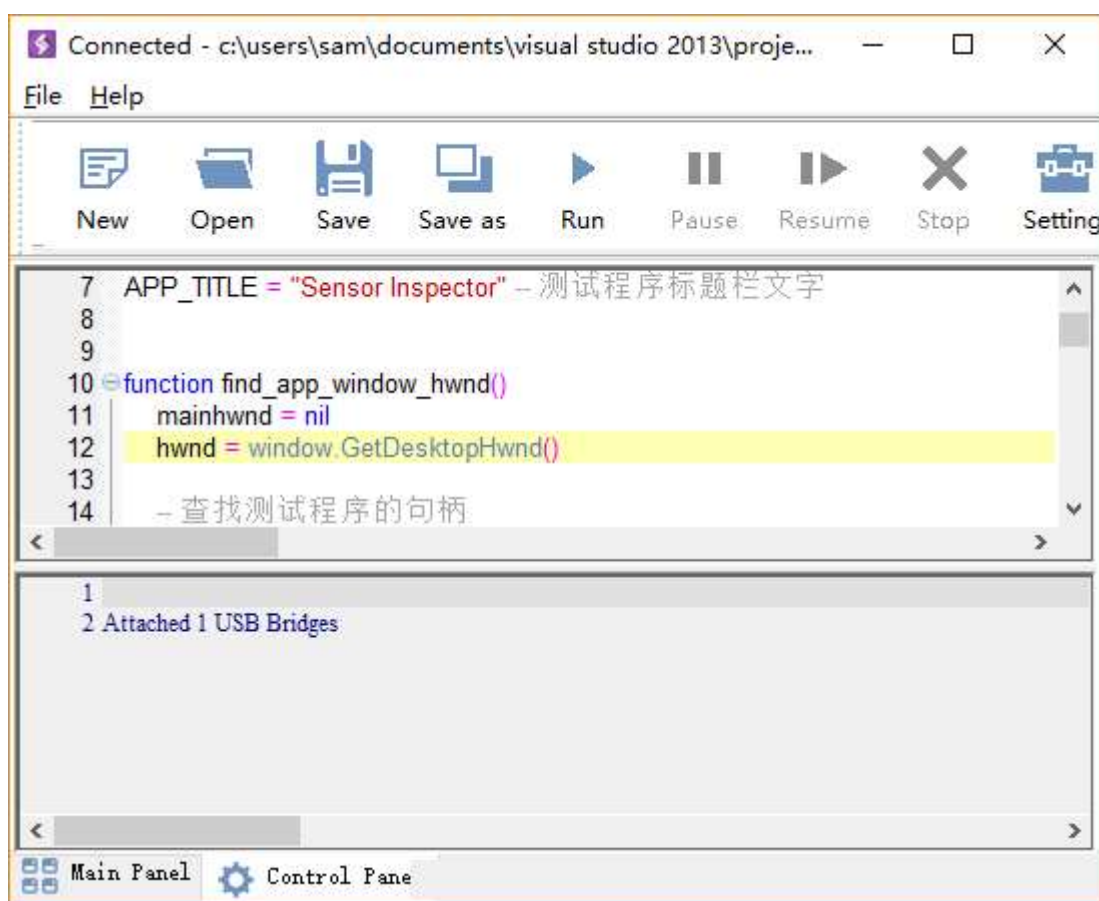


快速验证使用说明

- (1) 先将桥接板连接到电脑的 USB 口，待自动安装完驱动绿色指示灯亮即可开始使用。
- (2) 将一个 LED 等接在第一脚和第四脚之间。第一脚是 GND，第四脚是 GPIO 口输出脚。
- (3) 打开  TestCompanionsV1.0.0.1.exe 软件，载入专为 Atmel 而写的程序 atmel_monitor.lua



- (4) 将源程序 21~29 行，49~63 行的注释去掉。打开 Atmel 原厂测

试程序。先不要点击

Execute Test

按钮！

将第 47 行程序更改为 `Device Connect` 一样的文字 `Device Connect`（注意区分大小写）

```
f,x=window.EnumControl(windowhwnd,"STATIC","Device  
Connect")
```

运行程序，此时下面的状态信息会有输出，需要检查 `Text` 字段是否有相同的 `Device Connect` 字符，此时需要记下 `HWND`。

```
1  
2 HWND = 00641344; Class Name = WindowsForms10.Window.8.app.0.141b42a_r6_ad1; TextLength = 25; Text = Sensor Inspector Ver...  
3 mainhwnd = 00641344  
4 app_hwnd = 00641344  
5 \HWND = 003715DC; Class Name = WindowsForms10.STATIC.app.0.141b42a_r6_ad1; TextLength = 14; Text = Device Connect  
6  
7 Return Code =0(耗时 1521毫秒)  
8
```

然后将第 59 行中的 `HWND` 替换为查找到的数值。

```
if(0x003715DC == x.Hwnd[i]) then  
    sys.Print("\r\n Index = " ,i)  
end
```

保存程序，重新运行程序，此时状态信息输出为

```
1  
2 HWND = 00641344; Class Name = WindowsForms10.Window.8.app.0.141b42a_r6_ad1; TextLength = 25; Text = Sensor Inspector Ver...  
3 mainhwnd = 00641344  
4 app_hwnd = 00641344  
5 \HWND = 003715DC; Class Name = WindowsForms10.STATIC.app.0.141b42a_r6_ad1; TextLength = 14; Text = Device Connect  
6 Index = 1  
7  
8 Return Code =0(耗时 1996毫秒)  
9
```

此时就找到了测试程序输出控件的序号。

我们只需要把第 64 行的

`textboxhwnd = x.Hwnd[19]` --显示测试结果的控件的句柄

的 19，改为实际检测到的序号即可，这里是 1。

改好以后，就可以运行程序，看到效果。

```
1
2 HWND = 00641344; Class Name = WindowsForms10.Window.8.app.0.141b42a_r6_ad1; TextLength = 25; Text = Sensor Inspector Ver...
3 mainhwnd = 00641344
4 app_hwnd = 00641344
5 \_HWND = 003715DC; Class Name = WindowsForms10.STATIC.app.0.141b42a_r6_ad1; TextLength = 14; Text = Device Connect
6 Index = =1
7 text_hwnd = 003715DC
8 添加监视.....
9 被监测控件的文字 = Running.....
10 被监测控件的文字 = All PASS.....
```

(5) 源代码 function testresult_text_changed(code,str)用来根据文字判断测试结果。

```
1 --文字改变事件调用函数
2 function testresult_text_changed(code str)
3     sys.Print("\n\n 被监测控件的文字 = " str)
4     --重新打开设备，因为Atmel测试程序在测试的时候会软件模拟拔出所有USB设备，需要时刻保证USB连线。
5     usb.OpenByIndex()
6     --如果判断结果是PASS
7     if (str == "All PASS") then
8         usb.SetResetPinStatus(0) --设置设备的GPIO口为高电平
9         event.SyncCallBack(3000,low_the_gpio) --异步调用low_the_gpio函数，三秒钟后执行
10    end
11 --[[ 如果要判断是FAIL的话，使用这块程序
12    if (string.sub(str,0,4) == "FAIL") then
13        usb.SetResetPinStatus(0)--设置设备的GPIO口为高电平
14        event.SyncCallBack(3000,low_the_gpio) --异步调用low_the_gpio函数，三秒钟后执行
15    end
16 ]]
17 end
```

其中下面两行实现了检测后给定时间的 GPIO 口电平变化。

usb.SetResetPinStatus(0) --设置设备的 GPIO 口为高电平

event.SyncCallBack(3000,low_the_gpio) --异步调用 low_the_gpio

函数，3000 毫秒(三秒)钟后执行.