

基于CSS框架的S7-1200Web页面制作『续』

之前通过静态HTML和CSS文件制作的页面，美观度是没有什么问题了，但是为了数据实时更新显示，需要通过一条语句不停的刷新页面来实现，在页面文件比较少的环境下，这种刷新可能无伤大雅，但是当页面很多，或者一个页面上有好多条显示数据，控制按钮，这种刷新就成了无法忍受的硬伤，那么怎么让页面不刷新，数据也能实时更新呢，Javascript脚本就该登台了，具体机制我不做过多介绍，Javascript到底是啥我也不多说了，这里只注重实现，毕竟搞工业自动化又不是专职做网页。

引言

一个页面中有输入的变量，像是表单，按钮等，也有输出变量，像数据或者图形，我在上次的页面基础上增加一个文本框，用于输入变量，页面布局是这样的

DEMO Web Server for S7-1200 Base Bootstrap

基于S7-1200的测试程序

温度 0 湿度 0 大气压力 0

Device Name	Measure	Value	Note
温度传感器	环境温度值	0	-
湿度传感器	环境湿度值	0	-
压力传感器	大气压力	0	-
未知设备	设定值	0	<input type="text" value="输入任意值"/>

按下写入

输出控制

温度传感器	ON	OFF
湿度传感器	ON	OFF
压力传感器	ON	OFF

最后想要实现不刷新页面的情况下，所有数值实时变化，按下按钮后，对应的动作『即时』生效。

准备工作

首先需要准备两个html文件，一个文件放入输入变量，一个文件放入输出变量，js脚本工作时会从输出变量文件中找到需要的变量显示在页面中，或者把页面中得内容写到输入变量页面中得变量里，这里我把两个文件分别命名为IO_Output.html和IO_Input.html

IO_Output.html的内容如下，这是个Json结构，类似于Python中的字典，通过『键』可以访问到冒号后面的『值』，也就是常说的键值对

```
1 {
2   "Device1":":="Bootstrap_DB".Device1:",
3   "Device2":":="Bootstrap_DB".Device2:",
4   "Device3":":="Bootstrap_DB".Device3:",
5   "Temperature":":="Bootstrap_DB".Temperature:",
6   "Humidity":":="Bootstrap_DB".Humidity:",
7   "Pressure":":="Bootstrap_DB".Pressure:",
8   "Tag1":":="Bootstrap_DB".Tag1:"
9 }
```

IO_Input.html的内容如下

```
1 <!-- AWP_In_Variable Name='"Bootstrap_DB".Device1' -->:= "Bootstrap_D
  B".Device1:
2 <!-- AWP_In_Variable Name='"Bootstrap_DB".Device2' -->:= "Bootstrap_D
  B".Device2:
3 <!-- AWP_In_Variable Name='"Bootstrap_DB".Device3' -->:= "Bootstrap_D
  B".Device3:
4 <!-- AWP_In_Variable Name='"Bootstrap_DB".Tag1' -->:= "Bootstrap_DB".T
  ag1:
```

这样两个文件就准备好了，接下来就可以开始在HTML文件中调用它们实现数据实时读写了。

实际的使用中，还是按照之前说的几种操作来举例

- 网页端按下按钮控制PLC的变量
- 在网页端显示PLC的某个变量
- 网页端输入某个数字到PLC内某个变量

不过这次都通过Js脚本来实现。

开始之前，还有一点需要在html文件中做一下修改，就是要给每一个在页面中需要显示或者输入的元素给定一个唯一的id值，因为Js脚本是通过id值找到页面中的元素的，例如第一行的温度，就需要在html文件中温度的元素处给

定一个id值，这里给定的id值是Temperature，**每一个元素的id值必须唯一**

```
1 <li class="nav-item">
2   <button type="button" class="btn btn-primary">
3     温度 <span class="badge badge-light" id="Temperature">0</span>
4   </button>
5 </li>
```

网页端显示PLC的某个变量

首先需要在Head文件中增加Js脚本的调用，通过下面这条实现，具体jQuery是啥，去百度吧，文章大把的

```
1 <script type="text/javascript" src="src/js/jquery-2.0.2.min.js"></scr
  ipt>
```

要实时的读取PLC的变量在网页中显示，就需要Js脚本按照固定的时间间隔去获取IO_Output.html文件中的变量，之后放到合适的位置显示它。需要有一个如下的结构来写Js代码，在setInterval函数里面放入读变量的代码就可以实现数据的实时显示，因为这个函数是按照1000ms的周期循环执行

```
1 <script type="text/javascript">
2   $(document).ready(function () {
3     $.ajaxSetup({
4       cache: false
5     });
6     setInterval(function () {
7
8       //放入代码
9
10      });
11     }, 1000);
12   });
13 </script>
```

那么放入什么代码呢，自然是需要从IO_Output.html文件中把变量读过来，因为是Json格式，所以需要使用getJSON函数，这个函数有两个参数，一个是source，这里就是IO_Output.html，一个是data。通过之前定义的id值访问到网页中温度的元素，之后用data替换元素中的text，这段代码就这个意思

```

1 $.getJSON("IO_Output.html", function (data) {
2     $('#Temperature').text(data
3         .Temperature);
4 });

```

网页中实时显示PLC中数据，就是上面这些代码的复制粘贴，完整的代码如下

```

1 <script type="text/javascript">
2     $(document).ready(function () {
3         $.ajaxSetup({
4             cache: false
5         });
6         setInterval(function () {
7             $.getJSON("IO_Output.html", function (data) {
8                 $('#Temperature').text(data
9                     .Temperature);
10                $('#Humidity').text(data
11                    .Humidity);
12                $('#Pressure').text(data
13                    .Pressure);
14                $('#Temperature_t').text(data
15                    .Temperature);
16                $('#Humidity_t').text(data
17                    .Humidity);
18                $('#Pressure_t').text(data
19                    .Pressure);
20                $('#tag1').text(data
21                    .Tag1);
22            });
23        }, 1000);
24    });
25 </script>

```

控制PLC的某个布尔量

首先还是需要 在按钮的元素中给定一个唯一的id值，这个id值device1ON，id值一定要有意义，不然元素太多根本记不清哪个id对应的什么东西

```
1 <button class="btn btn btn-success" id="device1ON">ON</button>
```

之后需要在脚本中等待鼠标的点击动作，当这个元素上有click的事件发生，就执行POST的动作，把val的值赋给PLC的变量，这里是Bootstrap_DB".Device1

```
1 $("#device1ON").click(function () {
2     url = "IO_Input.html";
3     name = '"Bootstrap_DB".Device1';
4     val = 1;
5     sdata = escape(name) + '=' + val;
6     $.post(url, sdata, function (result) {});
7 });
```

页面中的所有按钮都可以这样处理，下面这段脚本就实现了PLC中一个布尔量的开和关

```
1 <script type="text/javascript">
2     $(document).ready(function () {
3         $.ajaxSetup({
4             cache: false
5         });
6         $("#device1ON").click(function () {
7             url = "IO_Input.html";
8             name = '"Bootstrap_DB".Device1';
9             val = 1;
10            sdata = escape(name) + '=' + val;
11            $.post(url, sdata, function (result) {});
12        });
13        $("#device1OFF").click(function () {
14            url = "IO_Input.html";
15            name = '"Bootstrap_DB".Device1';
16            val = 0;
17            sdata = escape(name) + '=' + val;
18            $.post(url, sdata, function (result) {});
19        });
20    });
21 </script>
```

网页端输入某个数字到PLC内某个变量

这个和按下按钮控制变量的实现机制是一样的，只不过需要有个输入框，输入框也需要有一个唯一的id值

```
1 <input class="form-control" type="text" placeholder="输入任意值" id="setValue">
```

和按钮触发的代码唯一不同的是，val变量处不能给定1或者0，而是要从输入框元素中获取，当然还是靠id值找到输入框

```
1 $("#setWrite").click(function () {
2     url = "IO_Input.html";
3     name = '"Bootstrap_DB".Tag1';
4     val = $('input[id=setValue]').val();
5     sdata = escape(name) + '=' + val;
6     $.post(url, sdata, function (result) {});
7 });
```

图形显示

再来说一种，有些时候希望一个Bool变量不是单纯的在网页端显示1或者0，而是能用图形来进行显示，这个也是可以实现的，当然首先还是你要给img元素一个唯一的id值

```
1 <img id="Lamp3" src='image/Lamp00.png' alt="">
```

那怎么替换图片呢，自然是通过改变src属性的值就可以，可以通过在脚本里面判断一个与图片相关的变量的值，当这个值变化是，对应的图片的src属性也跟着变化，这段代码就是判断按钮按下后，灯泡变量

```
1 if (data.Device3 == true) {
2     $("#Lamp3").attr("src", 'image/Lamp01.png')
3 }
```

基本上面四个方法能实现大多数页面中的效果，其他诸如更复杂的图表，曲线等，大家自己研究吧

最后是这个样子的

DEMO Web Server for S7-1200 Base Bootstrap

基于S7-1200的测试程序

温度 273


湿度 273

大气压力 273

Device Name	Measure	Value	Note
温度传感器	环境温度值	273	-
湿度传感器	环境湿度值	273	-
压力传感器	大气压力	273	-
未知设备	设定值	0	<input type="text" value="输入任意值"/>


[按下写入](#)

输出控制

温度传感器 


ON

OFF

湿度传感器 

ON

OFF

压力传感器 

ON

OFF

[百度网盘链接](#)，提取码：uy96