

网页控制基本原理简介（一）—— CGI 和 SSI 简介

TI 例程中的 Demo2 采用的就是这种方法。

CGI：Common Gate Interface，简称 CGI。在物理上是一段程序，运行在服务器上，提供同客户端 HTML 页面的接口。这样说大概还不好理解。那么我们看一个实际例子：现在的个人主页上大部分都有一个留言本。留言本的工作是这样的：先由用户在客户端输入一些信息，如名字之类的东西。接着用户按一下“留言”（到目前为止工作都在客户端），浏览器把这些信息传送到服务器的 CGI 目录下特定的 cgi 程序中，于是 cgi 程序在服务器上按照预定的方法进行处理。在本例中就是把用户提交的信息存入指定的文件中。然后 cgi 程序给客户端发送一个信息，表示请求的任务已经结束。此时用户在浏览器里将看到“留言结束”的字样。整个过程结束。

SSI：将内容发送到浏览器之前，可以使用服务器端包含 (SSI) 指令将文本、图形或应用程序信息包含到网页中。例如，可以使用 SSI 包含时间/日期戳、版权声明或供客户填写并返回的表单。对于在多个文件中重复出现的文本或图形，使用包含文件是一种简便的方法。将内容存入一个包含文件中即可，而不必将内容输入所有文件。通过一个非常简单的语句即可调用包含文件，此语句指示 Web 服务器将内容插入适当网页。而且，使用包含文件时，对内容的所有更改只需在一个地方就能完成。

因为包含 SSI 指令的文件要求特殊处理，所以必须为所有 SSI 文件赋予 SSI 文件扩展名。默认扩展名是 .stm、.shtm 和 .shtml

简单来说有以下几个关键词：

运行在服务器端：CGI 的处理函数运行在服务器端，说白了点现在就是 LM3S8962 开发板。

浏览器发送 uri：通过提交 HTML 表单的方式来提交请求 CGI 函数处理的必要信息。

CGI 处理函数：当 Http 服务器收到了 uri 之后，分析它是否是 CGI 处理结构体中的某一个函数，如果有的话，就调用处理函数去处理，如果没有，就去寻找其它的网页。

返回网页：可以返回任何整个的网页，就返回 index.shtm，或者是其它 *.ssi, *.shtml 之类的网页文件，这些文件里边一般都有 SSI 标签，但这些标签是注释了的，正常情况下是不会显示出来的，需要服务器在发送数据之前把它们替换掉。所以这样就有了在服务器 SSI 处理函数。

有关 CGI 和 SSI 的知识需要我们逐步去积累和学习，但 TI 提供的这个例程是一个很好的基础教程，有了它你可以很快地入门，我也会在把这个例程进行分解，实现一个最简单的功能，其实很多复杂的功能都是这些简单功能的有机组合。把杂糅在一起的功能给它分开，这样就可以逐个击破，再综合运用，给初学者出点力。

我们在看一个简单的表单的例子，这是我提供的简单例程中的一部分：

```
<form method="get" action="settxt.cgi" name="settxt"> // 请求服务器执行的 CGI 函数
<table border="1" cellpadding="2" cellspacing="2" width="100%">
<tbody>
<tr align="center">
<td valign="top"><b> 请输入不超过 20 个的英文字符：<br>
```

```

</b><input maxlength="20" size="20" name="DispText">
<input
    name="Display" value="    发送文 本" type="submit"></td>
</tr>
</tbody>
</table>
</form>

```

其实和编写普通的网页文件没有太大的区别，这点和 Keil RL TCPnet 中所提供的按行来处理的 CGI 方法是不太一样的，个人觉得花点时间掌握这个之后更灵活地应用。

接下来，在对这两种基本方法做一个简单地总结吧，有疑问可以在论坛帖子下留言，我们共同讨论：

绝大多数的 CGI 程序被用来解释处理来自表单的输入信息，并在服务器产生相应的处理，或将相应的信息反馈给浏览器。CGI 程序使网页具有交互功能。

总之，这两种方法都是为了使网页具有交互功能，而不只再满足于服务器端提供网页数据，客户端被动的显示数据。有了这两种技术，客户端可以要求服务端按照客户端的各种特殊情况提供不同的网页数据。这样我们就可以利用这两种技术来实现网页对设备的发送控制信息，并让设备返回执行控制命令之后的状态变化。

它们的异同点是，大体上都是这样的三个步骤，

1.客户端发送信息（XMLHttpRequest 采用的是 XMLHttpRequest 的 open 方法，CGI 采用的是提交表单的方式）

2.服务器端根据接收到的信息调用相应的程序（XMLHttpRequest：方式采用的是一般读取网页数据的方法，采取这种方法的原因是，XMLHttpRequest 的 open 方法中使用了 GET 方法，而我们知道获取网页文件一般也是发送了一个 GET 方法，两者之间就有了一致点。只需要用一个 switch（uri）case 语句就可以区分开来自客户端的各种不同的请求信息，实际上如果是发送控制信息的话，我们最后并不读取网页文件，而只是借用一下这个“壳”，而是判断之后，执行控制程序，把执行结果（当然是动态生成的存于数组里的信息）返回到文件指针；CGI：在客户端（浏览器）同 XMLHttpRequest 一样也是发送了一个 GET 请求，后面跟的 uri。这个 uri 是 form 的 action 值，参数是该 form 中各种类型的输入控件的 name 和 value，这个参数的个数可以在服务器中进行设置。

3 返回的控制信息到了客户端，客户端要显示出来。这两者的显示方法是大大不同的，XMLHttpRequest 是 Ajax 技术的核心，最大的优点就是在不刷新整个页面的前提下由 JavaScript 在后台默默地与服务器交换数据。而 CGI 返回的是整个页面，这样一来就相当于刷新了整个页面，如果只是要反馈少量信息的话，这样每次刷新页面就会给人一种延时很大的感觉，而且对网络的利用率也不高。对于 XMLHttpRequest 来说，具体在编写 HTML 语言的时候就是要在需要的位置放置一些 <div></div>。