

```

<!DOCTYPE HTML>
<html>
<head>
  <meta charset="utf-8">
</head>
<body>
  <div style="font-size:24pt;text-align: center;">Мой сайт</div>
  <div style="text-align:center;">
    <table style="margin:auto;">
      <td>
        
      </td>
      <td>
        <form><input value="Relay 1 ON" type="button" id="Relay_1_ON"
onclick="Relay(1,1);"><br/><br/></form>
        <form>
          <input value="Relay 1 OFF" type="button" id="Relay_1_OFF"
onclick="Relay(1,0);"><br/><br/>
        </form>
        <form>
          <input value="Relay 1 PULSE" type="button" id="Relay_1_SET"
onclick="Relay(1,2);"><br/><br/>
        </form>
      </td>
    </table>
  </div>
  <script type="text/javascript">
    /*
    Для управления реле при нажатии кнопки вызывается функция
    Relay(Number,Action)
    Number - номер реле, над которым производится действие
    Action - действие, которое требуется совершить:

```

```

    0 - выключить;

```

1 - включить;

2 - подать импульс (длительность настраивается в web-интерфейсе)

Далее выполняется асинхронный запрос по сгенерированному адресу

После окончания запроса вызывается функция пост обработки, которая определяет какая кнопка вызвала запрос и красит её в зеленый цвет, если в ответ на запрос устройство вернуло "Success" иначе в красный цвет 500 мс

```
*/  
  
var IP_ADDR = "172.16.0.150"  
  
function Relay(Number, Action)  
{  
    //Формируем запрос вида http://[IP]/rbXn.cgi (включить реле X), либо http://[IP]/rbXf.cgi  
(выключить реле X)  
    switch (Action)  
    {  
        case (0) : { SendHttpAsyncRequest("http://" + IP_ADDR + "/rb" + (Number-1).toString() +  
"f.cgi"); break; }  
        case (1) : { SendHttpAsyncRequest("http://" + IP_ADDR + "/rb" + (Number-1).toString() +  
"n.cgi"); break; }  
        case (2) : { SendHttpAsyncRequest("http://" + IP_ADDR + "/rb" + (Number-1).toString() +  
"s.cgi"); break; }  
        default : { alert("Incorrect Action in function Relay(..., ...)"); }  
    }  
}  
  
function RelayPostHandler(Request, Status)  
{  
    var TrimmedRequest = Request.replace("http://" + IP_ADDR + "/rb", "").replace(".cgi",  
"").split("");  
    var Suffix = "";  
    switch ( TrimmedRequest[1] )  
  
    {  
  
        case ("n") : { Suffix = (parseInt(TrimmedRequest[0])+1).toString() + "_ON"; break; }  
        case ("f") : { Suffix = (parseInt(TrimmedRequest[0])+1).toString() + "_OFF"; break; }  
        case ("s") : { Suffix = (parseInt(TrimmedRequest[0])+1).toString() + "_SET"; break; }  
    }  
}
```

```

        default: { return; }
    }

    var MyButton = document.getElementById("Relay_" + Suffix);
    if ((Status != "") && (Status === "Success")) { MyButton.style.backgroundColor = "Lime"; }
    else { MyButton.style.backgroundColor = "Red"; }
    setTimeout(() => MyButton.style.backgroundColor = "", 500);
}

function SendHttpRequest(Request)
{
    var req;
    if (window.XMLHttpRequest) { req = new XMLHttpRequest(); }
    else if (window.ActiveXObject) { req = new ActiveXObject("Microsoft.XMLHTTP"); }
    if (req) {
        req.open("GET", Request, true);
        req.timeout = 2000;
        req.onreadystatechange = function (e) {

            if (req.readyState == 4)
            {
                if (req.status == 200) { RelayPostHandler(Request, req.responseText); }
                else { RelayPostHandler(Request, ""); }
            }
        };

        req.onerror = function (e) { RelayPostHandler(Request, ""); }
        req.ontimeout = function (e) { RelayPostHandler(Request, ""); }
        req.send();
    }
}
</script>
</body>
</html>

```