

[Home](#)[Beginners](#)[Projects](#)[Tutorials](#)[Articles](#)[Reviews](#)[Software](#)

STARTING ELECTRONICS

Electronics for Beginners and Beyond

[Blog](#)[YouTube](#)[Donate](#)

[Home](#) ▶ [Tutorials](#) ▶ [Arduino](#) ▶ [Ethernet Shield Web Server Tutorial](#) ▶ [AJAX Read Switches Analog](#)

Analog Inputs and Switches using AJAX

Created on: 20 February 2013

Part 9 of the Arduino Ethernet Shield Web Server Tutorial

Updating the status of more than one switch that is interfaced to the Arduino web server, as well as showing the value of one of the analog inputs.

JavaScript is used to make AJAX calls to request the switch status and analog value from the web server.

This video shows the switches and analog input

updated on the web page without flicker. Only parts of the web page are updated using AJAX.

Donate to Starting Electronics

Donate



Arduino Ethernet Shield Tutorial

Tutorials

Arduino

**Part 1:
Ethernet
Shield Tutorial
Introduction
and Hardware**

**Part 2: Basic
Arduino Web
Server**

**Part 3: HTML
Web Page
Structure**

Reading switches and analog input using...



Watch on  YouTube

Can't see the video? [View on YouTube →](#)

Circuit Diagram

The circuit diagram below shows how the switches are interfaced to the Arduino (with Ethernet shield plugged into it). A potentiometer is interfaced to analog input A2 so that the value on A2 can be changed and updated on the web page.

Part 4:
Arduino SD
Card Web
Server

Part 5:
Arduino Web
Server LED
Control

Part 6:
Reading a
Switch

Part 7:
Reading a
Switch using
AJAX

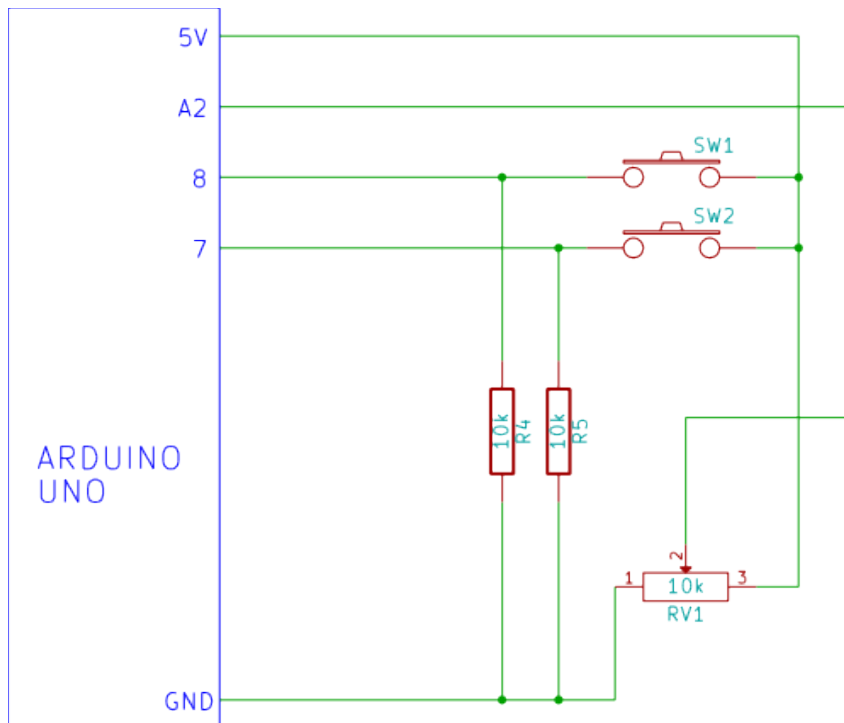
Part 8:
Reading a
Switch
Automatically
using AJAX

Part 9:
Reading an
Analog Input
and Switches
using AJAX

Part 10:
Linking Web
Pages

Part 11: Web
Page Images

Part 12: CSS
Introduction



Part 13:
Reading a
Switch with
SD Card Web
Server and
Ajax

Part 14:
Reading
Inputs with
Ajax and XML

Part 15:
Analog Value
Displayed on
Gauge

Part 16: Inputs
and Outputs
(I/O)

The Sketch

The Arduino sketch is a modified version of the sketch from the [previous tutorial](#).

/*-----

Program: eth_websrv_AJAX_IN

Description: Uses Ajax to update the state of two s
and an analog input on a web page. The
web server hosts the web page.
Does not use the SD card.

Hardware: Arduino Uno and official Arduino Ether
shield. Should work with other Arduino
compatible Ethernet shields.

Software: Developed using Arduino 1.0.3 software
Should be compatible with Arduino 1.0

References: - WebServer example by David A. Mellis

Part 17:
Accessing
HTML Tags
with CSS and
JavaScript

Part 18: CSS
for
Positioning,
Sizing and
Spacing

Summary and
Conclusion



Web Page Code

The above sketch produces the following HTML code:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Arduino Web Page</title>
    <script>
      function GetSwitchAnalogData() {
        nocache = "&nocache=" + Math.random() * 1000000;
        var request = new XMLHttpRequest();
        request.onreadystatechange = function() {
          if (this.readyState == 4) {
            if (this.status == 200) {
              if (this.responseText != null) {
                document.getElementById("sw_an_data").innerHTML = this.responseText;
              }
            }
          }
        };
        request.open("GET", "ajax_switch" + nocache, true);
        request.send(null);
        setTimeout('GetSwitchAnalogData()', 1000);
      }
    </script>
  </head>
  <body onload="GetSwitchAnalogData()">
    <h1>Arduino AJAX Input</h1>
    <div id="sw_an_data">
    </div>
  </body>
</html>
```

[HTML Code Produced by Arduino Sketch - click for a bigger image](#)

Modifications to the Sketch

Arduino pins 7 and 8 are both configured as inputs in the **setup()** part of the sketch.

The JavaScript function that handles the AJAX call has been renamed. The Arduino function that responds to the AJAX call has also been renamed.

A HTML <div> has been created below the H1 header in the HTML code and given the id "sw_an_data". The div is invisible on the page, but it serves as a place for the JavaScript to put the information (switch and analog values) sent back from the Arduino.

Sending the Request for Data from the Browser

The JavaScript function **GetSwitchAnalogData()** is called every second. Every second, it sends a GET request to the Arduino web server.

Receiving and Processing the AJAX Request on the Arduino

When the Arduino receives the AJAX request, it runs the **GetAjaxData()** function. This function reads the state of the two switches and sends the switches' statuses (ON or OFF)

back to the web browser. The function also reads the value on the A2 analog pin and sends the value back to the browser.

Displaying the New Data in the Web Browser

When the web browser receives the data requested from the Arduino, it simply inserts it into the div that has the ID `sw_an_data`.

[← Go back to Part 8](#)

[Go to Part 10 →](#)

Comments

[Login](#)

There are no comments posted yet. [Be the first one!](#)

Post a new comment

Enter text right here!

Comment as a Guest, or login:

Name

Displayed next to your comments.

Email

Not displayed publicly.

Website (optional)

If you have a website, link to it here.

Subscribe to

Submit Comment

[Arduino](#)

[Pinout](#)

[About](#)

[Contact](#)

[Donate](#)

[Privacy Policy](#)

[Amazon Adverts](#)

© 2012 – 2021, Starting Electronics