```
// zoomkat's meta refresh data frame test page 5/25/13
// use http://192.168.1.102:84 in your brouser for main page
// http://192.168.1.102:84/data static data page
// http://192.168.1.102:84/datastart meta refresh data page
// for use with W5100 based ethernet shields
// set the refresh rate to 0 for fastest update
// use STOP for single data updates
#include <SPI.h>
#include <Ethernet.h>
const int analogInPin0 = A0;
const int analogInPin1 = A1;
const int analogInPin2 = A2;
const int analogInPin3 = A3;
const int analogInPin4 = A4;
const int analogInPin5 = A5;
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; //physical mac address
byte ip[] = { 192, 168, 1, 102 }; // arduino ip in lan
byte gateway[] = { 192, 168, 1, 1 }; // internet access via router
byte subnet[] = { 255, 255, 255, 0 }; //subnet mask
EthernetServer server(84); //server port
unsigned long int x=0; //set refresh counter to 0
String readString;
//////////////////////
void setup(){
Serial.begin(9600);
// disable SD SPI if memory card in the uSD slot
pinMode(4,OUTPUT);
digitalWrite(4,HIGH);
Ethernet.begin(mac, ip, gateway, gateway, subnet);
server.begin();
Serial.println("meta refresh data frame test 5/25/13"); // so I can keep track of
what is loaded
}
void loop(){
EthernetClient client = server.available();
if (client) {
while (client.connected()) {
if (client.available()) {
char c = client.read();
if (readString.length() < 100) {
readString += c;
}
//check if HTTP request has ended
if (c == '\n') {
//check get atring received
Serial.println(readString);
//output HTML data header
client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println();
//generate data page
if(readString.indexOf("data") >0) {  //checks for "data" page
x=x+1; //page upload counter
client.print("<HTML><HEAD>");
//meta-refresh page every 1 seconds if "datastart" page
if(readString.indexOf("datastart") >0) client.print("<meta http-equiv='refresh'
content='1'>");
//meta-refresh 0 for fast data
```

```
if(readString.indexOf("datafast") >0) client.print("<meta http-equiv='refresh'
content='0'>");
client.print("<title>Zoomkat's meta-refresh test</title></head><BODY><br>");
client.print("page refresh number: ");
client.print(x); //current refresh count
client.print("<br><br>");
//output the value of each analog input pin
client.print("analog input0 is: ");
client.print(analogRead(analogInPin0));
client.print("<br>analog input1 is: ");
client.print(analogRead(analogInPin1));
client.print("<br>analog input2 is: ");
client.print(analogRead(analogInPin2));
client.print("<br>analog input3 is: ");
client.print(analogRead(analogInPin3));
client.print("<br>analog input4 is: ");
client.print(analogRead(analogInPin4));
client.print("<br>analog input5 is: ");
client.print(analogRead(analogInPin5));
client.println("<br></BODY></HTML>");
}
//generate main page with iframe
else
{
client.print("<HTML><HEAD><TITLE>Zoomkat's frame refresh test</TITLE></HEAD>");
client.print("Zoomkat's Arduino frame meta refresh test 5/25/13");
client.print("<BR><BR>Arduino analog input data frame:<BR>");
client.print("  <a href='http://192.168.1.102:84/datastart'
target='DataBox' title=''yy''>META-REFRESH</a>");
client.print("    <a href='http://192.168.1.102:84/data'
target='DataBox' title=''xx''>SINGLE-STOP</a>");
client.print("    <a href='http://192.168.1.102:84/datafast'
target='DataBox' title=''zz''>FAST-DATA</a><BR>");
client.print("<iframe src='http://192.168.1.102:84/data' width='350' height='250'
name='DataBox'>");
client.print("</iframe><BR></HTML>");
}
delay(1);
//stopping client
client.stop();
//clearing string for next read
readString="";
}
}
}
}
}
```