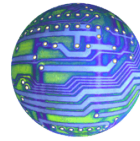




UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA

DEPARTAMENTO DE ELECTRÓNICA



PROYECTO

ELO-322

"FAN MONITORING"

Nombre: Gustavo Sazo S.

Rol: 2990004-3

Profesor: Agustín Gonzalez

Ayudante: Evandry Ramos

Fecha: 23 de Agosto 2013

INTRODUCCIÓN

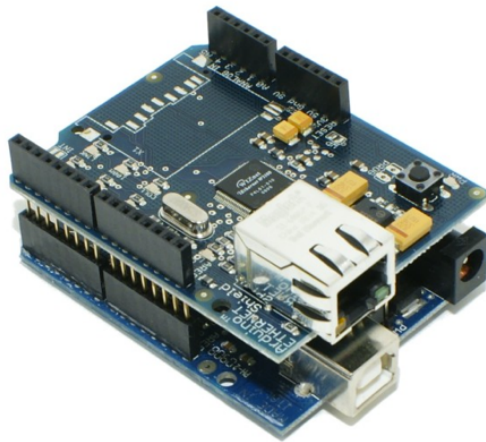
El auge de las telecomunicaciones que ha sido experimentado en los últimos años, en conjunto a internet, ha traído a la sociedad una nueva forma de interactuar con el entorno en el que se vive. Sistemas computacionales, telefónicos e inalámbricos, como nunca antes se habían imaginado, traen consigo la necesidad de crear e innovar dispositivos o iniciativas cada vez más potentes y al alcance de la población.

Es esta necesidad la que motivó la iniciativa del presente proyecto, interactuar remotamente con dispositivos a través de internet. Mas aún, con la tecnología actual de teléfonos inteligentes, es posible aprovechar al máximo la potencialidad de los dispositivos que utilizan internet. En base a esto nace **FAN MONITORING**, hardware y software creado para interactuar con ventiladores en cualquier parte del mundo y en cualquier minuto que se requiera.

RESUMEN

Fan Monitoring busca configurar un servidor web en una placa *Arduino* en conjunto a un dispositivo *Shield Ethernet*, con el propósito de manejar ventiladores a través de la web.

En el ámbito de implementación de hardware se utilizaron las placas **Arduino UNO**, **Arduino Ethernet Shield** y un ventilador de PC. Con respecto a la implementación de software fue necesario adentrarse en la programación de los dispositivos mencionados, estudiando manuales y aprovechando la gran cantidad de información que existe en la web acerca de estos dispositivos. A la vez, fue necesario estudiar cómo poder confeccionar un entorno gráfico necesario para que el cliente pueda ingresar y ajustar los parámetros requeridos de los ventiladores.



Arduino Ethernet Shield montado sobre Arduino

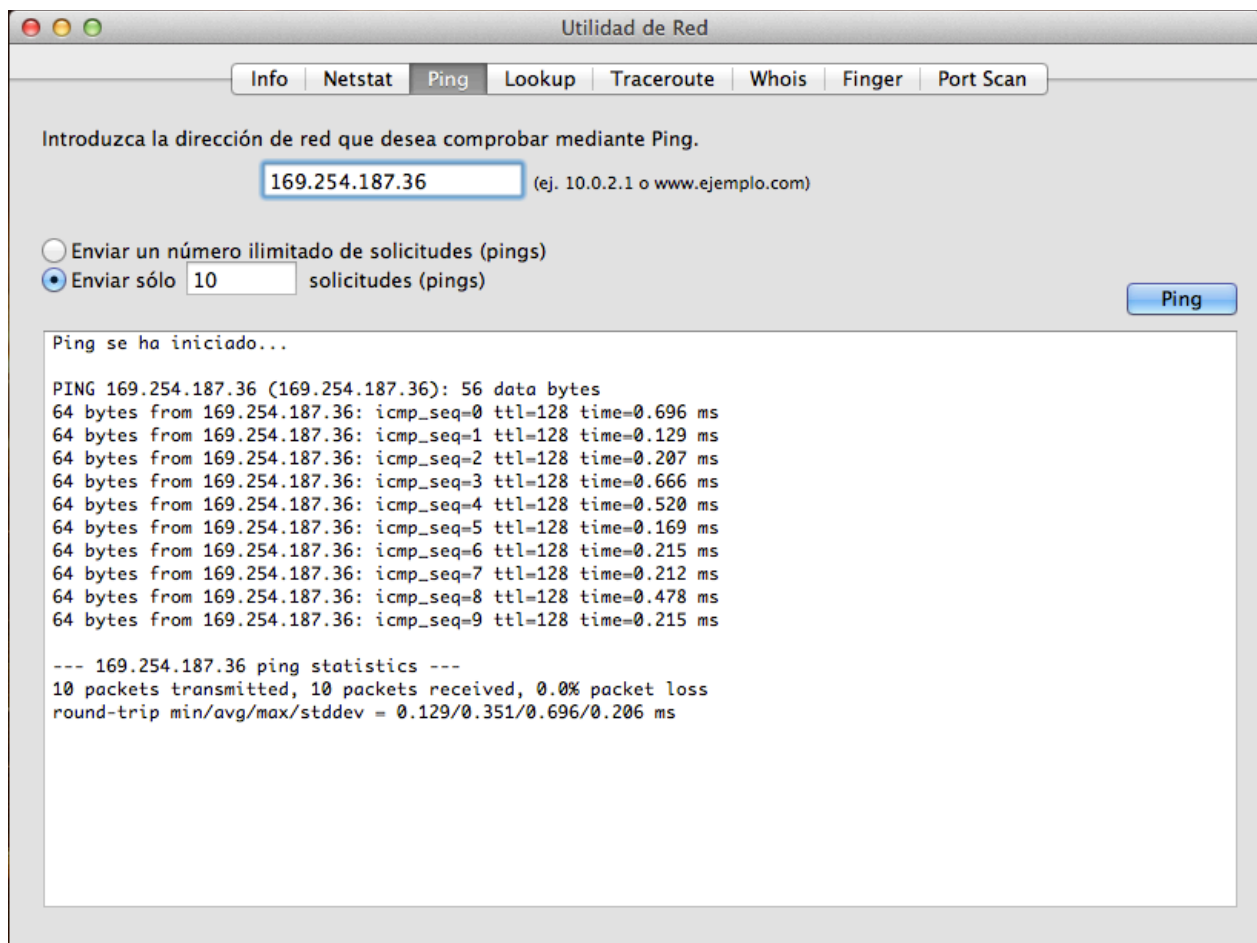


Ventilador de PC

DESARROLLO

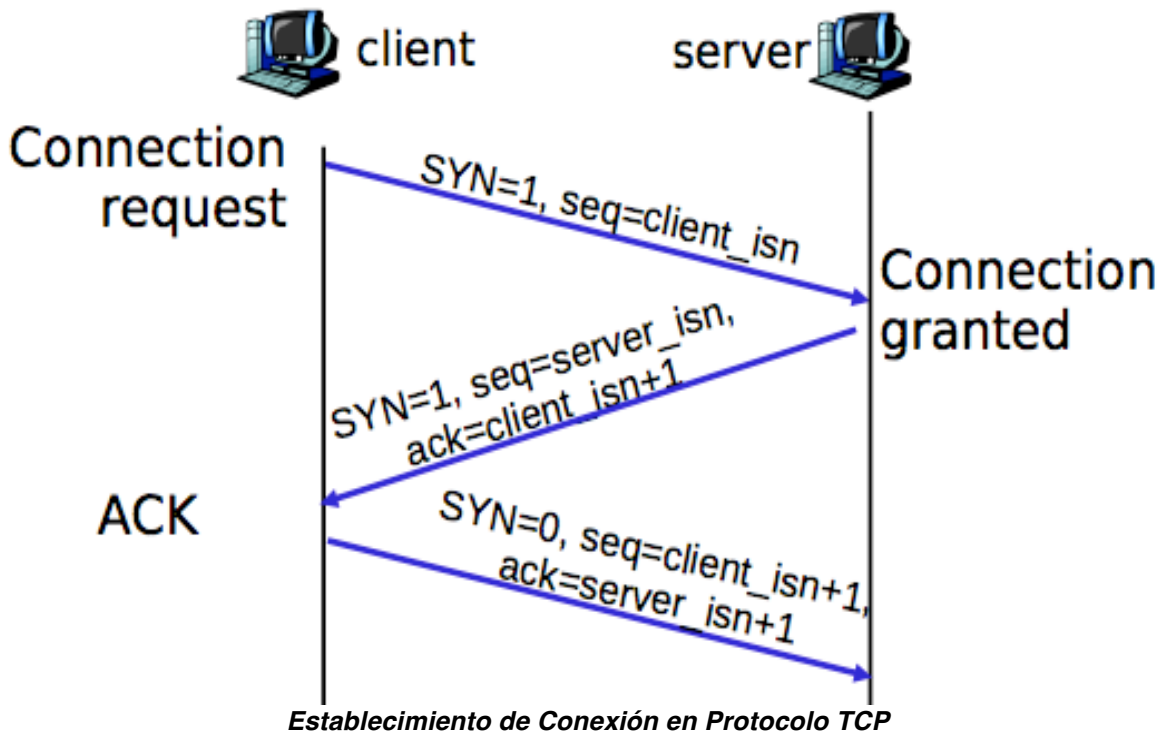
Para el desarrollo del presente proyecto fue necesario realizar pruebas de comunicación entre router (computador) y el conjunto **Arduino - Ethernet Shield**, para comprobar la correcta conexión se envían paquetes, mediante **Ping**, entre los dispositivos y se corrobora que los paquetes no sufrieran pérdidas. Una vez comprobado que los protocolos de transmisión se encuentran correctamente configurados, concluyendo que sólo deben estar conectados a una misma red (red wifi creada con un computador personal), es posible comenzar con el desarrollo del código necesario para la ejecución del sistema.

A continuación se presenta una captura de imagen del programa utilizado para realizar el ping de comprobación de envío de datos:



Captura de Imagen de Comprobación Mediante Ping

La conexión entre cliente y servidor es realizada internamente por el dispositivo, mediante la librería **Ethernet.h** la cual utiliza el protocolo **TCP** para establecer una conexión con transmisión de datos confiable. A continuación se presenta un diagrama del establecimiento de conexión entre un cliente y servidor utilizando el protocolo de transporte TCP:



Una vez que se tiene conexión entre ambos dispositivos, se procede al desarrollo del código que permitirá controlar el ventilador a través de una interfaz web, de muy bajo nivel debido a que se trabaja con una memoria de 32 KB. Para el desarrollo de la interfaz fue imprescindible trabajar con el protocolo de la capa de aplicación **HTTP**, el cual tiene un modelo *cliente - servidor* como el que requerido en el proyecto. Necesario también fue adquirir datos como la dirección **IP** y **MAC** a utilizar para poder configurar correctamente el código en el dispositivo *Arduino*.



Captura de Imagen de la Interfaz Web de FAN MONITORING

A continuación se presenta el código utilizado para la implementación de **FAN MONITORING**:

```
#include <SPI.h>
#include <Ethernet.h> // Librería que realiza la comunicación Cliente-Servidor
byte mac[] = {0x90, 0xA2, 0xDA, 0x0E, 0x85, 0x0F}; // MAC Arduino Ethernet Shield
byte ip[] = {169,254,187,36}; // Asignacion de direccion IP, basado en la red que se tiene
EthernetServer server(80); // Definición puerto. Se utiliza el Puerto 80 (HTTP)
int FAN_1 = 6; // Declaración variables a utilizar en "pines" int FAN_1 = 6;

//Inicio Ethernet
void setup() {
  Ethernet.begin(mac, ip);
  server.begin();
  pinMode(FAN_1, OUTPUT); // Se definen los "pines" a utilizar
}

//Buffer Obtencion de Datos
#define BUFFER 100
char clientline[BUFFER];
int index = 0; //Index Cliente
void loop(){ // Función recursiva que ejecuta el código
  index=0; //Resetea Index Cliente
  EthernetClient client = server.available();

  //Conexión activa Cliente-Servidor
  if (client) {
    boolean currentLineIsBlank = true; while (client.connected()) {
      if (client.available()) { char c = client.read();
        if(index<BUFFER) //Solo agrega datos si el Buffer NO esta lleno
        {
          clientline[index]=c;
          index++; }
      }
```

```
//Codigo HTML que permite activar los diversos "pines"
if (c == '\n' && currentLineIsBlank) {
client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println(); //Se debe dejar una linea en blanco

// Código HTML para Interfaz Web
client.println("<h1><center>PROYECTO ELO322</h1></center>");
client.println("<h1><center>FAN MONITORING</h1></center>");
client.println("<h2><center>VENTILADOR</h2></center><center><form method=get action=?><input
type=checkbox name=L3 value=1>Encendido<br><input type=checkbox name=L3
value=0>Apagado<br><input type=submit value=Enter</form></center>");
break;
}
if (c == '\n') {
currentLineIsBlank = true; }
else if (c != '\r') { currentLineIsBlank = false;
}

//Control Ventilador
if(strstr(clientline,"/?L3=0")!=0) { //Busca comando para apagar el Ventilador
digitalWrite(FAN_1, 0); //Apaga el Ventilador
} else if(strstr(clientline,"/?L3=1")!=0) { //Busca comando para encender el Ventilador
digitalWrite(FAN_1, 1); //Enciende el Ventilador
}
}}
delay(10); // Retardo para reiniciar el ciclo
client.stop(); }
}
```

CONCLUSIONES

En este proyecto fue posible aterrizar la cantidad de opciones que se pueden realizar con hardwares como *Arduino*, dónde la limitación económica deja de ser un problema y lo único que se necesita son las ganas de crear e imaginar soluciones a problemas que aún niquera se piensan. Por otra parte, fue posible incursionar en temas del ramo tales como la designación de puertos, comprender a mayor cabalidad protocolos como *HTTP* y *TCP*, adentrarse más en el tema de conexión entre cliente - servidor y conocer a grandes rasgos la programación básica de un ambiente gráfico web.

El proyecto inicialmente contempló el uso de un ventilador de PC, sin embargo, al requerir éste mayor potencia de la que es capaz de entregar el dispositivo utilizado es que se cambió por un motor de menor capacidad que fuese capaz de ser accionado por la interfaz *Arduino*. Una forma de solucionar este problema es utilizando amplificadores operacionales, para poder aumentar la tensión de salida del dispositivo.

REFERENCIAS

- *Apuntes de Clases ELO-322*, A. Gonzalez
- <http://www.arduino.cc>
- *"Computer Networking: A top-Down Approach Featuring the Internet"* James F. Kurose and Keith W. Ross
- *"Beginning Arduino"*, Michael McRoberts
- *"Beginning Arduino Programming"*, Brian Evans
- *"Practical Arduino, Cool Projects For Open Source Hardware"*, Jonathan Oser & Hugh Blemings