

## Manipulación de motor Shunt, monitoreo de temperatura y vibración mecánica mediante App Virtuino y módulo Esp8266 vía wifi

LÓPEZ-TOLEDO, Eliut\*†

Recibido 7 de Enero, 2017; Aceptado 16 de Marzo, 2017

### Resumen

Manipulación De Motor Shunt, Monitoreo De Temperatura y Vibración Mecánica Mediante App Virtuino y Módulo Esp8266 Vía Wifi. Realizar el arranque y paro de un Motor Shunt de CC y el monitorear las variables: Temperatura y vibración mecánica utilizando la APP Virtuino de Android. Utilizando una tarjeta Arduino Mega, un módulo Wifi ESP8266, un sensor de Temperatura y un sensor de vibración mecánica, se realiza el hardware, posteriormente la programación del módulo Wifi ESP8266 y de la tarjeta Arduino Mega para la comunicación con la App Virtuino. En la App Virtuino de Google Play se realiza la configuración del servidor para la comunicación con la interface y se establecen los indicadores para el monitorear las variables y los interruptores para activar y desactivar al motor shunt. Permite monitorear las variables: temperatura, corriente eléctrica y vibraciones mecánicas por medio de dispositivos móviles. Permite activar o desactivar el motor por necesidades diversas de manera remota.

**Iot (Internet de las cosas), Monitoreo, APP Virtuino, ESP8266**

### Abstract

Shunt Motor Manipulation, temperature monitoring and mechanical vibration using App Virtuino and module ESP8266. Carry out the start and stop of a DC Motor Shunt and the monitoring of the variables: Temperature and mechanical vibration using Virtuino APP. Using an Arduino Mega board, an ESP8266 Wifi module, a temperature sensor and a mechanical vibration sensor, the hardware is realized, then programming the ESP8266 Wifi module and the Arduino Mega card for communication with the Virtuino App. In the Google Play Virtuino App the configuration of the server is made to communicate with the interface and the indicators are set to monitor the variables and the switches to activate and deactivate the shunt motor. It allows monitoring the variables: temperature, electric current and mechanical vibrations by means of mobile devices. It allows to activate or deactivate the motor by diverse needs of remote way

**IOT (Internet of Things), Monitoring, Virtuino APP, ESP8266**

**Citación:** LÓPEZ-TOLEDO, Eliut. Manipulación de motor Shunt, monitoreo de temperatura y vibración mecánica mediante App Virtuino y módulo Esp8266 vía wifi. Revista de Ingeniería Tecnológica 2017. 1-1:29-38

\*Correspondencia al Autor (Correo Electrónico: elopez@ute.edu.mx)

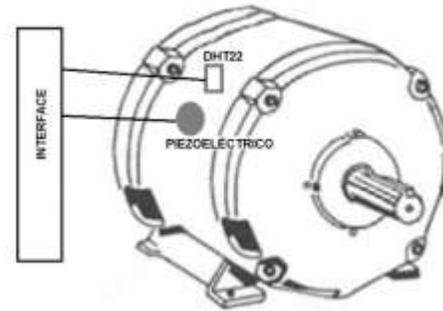
† Investigador contribuyendo como primer autor.

## Introducción

Cuando trabajamos con una herramienta que nos permita saber en todo momento el comportamiento de nuestras variables importantes en un proceso industrial, se pueden reducir los costos que implican los mantenimientos correctivos y preventivos, disminuyendo las paradas por inspección a los equipos y permitiéndonos evitar las averías imprevistas detectando las anomalías que pudieran estar presentes y dándole el seguimiento a su posible evaluación; además apoya la existencia de parámetros funcionales indicadores del estado del equipo.

En este trabajo se realiza el monitoreo de las variables físicas de Temperatura y vibración mecánica en un Motor Shunt DC mediante una tarjeta Arduino Mega, un sensor de temperatura, un sensor piezoeléctrico y un dispositivo móvil a través de la aplicación que se encuentra libre en Play store de Google llamada Virtuino. Debido a la importancia de saber en cualquier momento el estado de estas variables, mediante una conexión wifi y el dispositivo móvil con sistema operativo Android esto es posible.

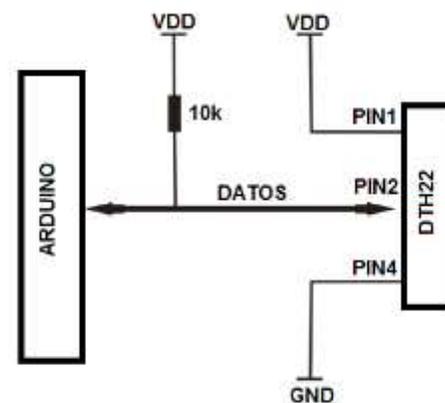
Dado el crecimiento de la transformación digital en los procesos industriales, es por ello la importancia del desarrollo y aplicación de este tipo de sistemas a procesos industriales, que en esta ocasión es destinado a un Motor eléctrico Shunt DC (ver figura 1).



**Figura 1** Motor shunt y sensores DHT22 y Piezoeléctrico

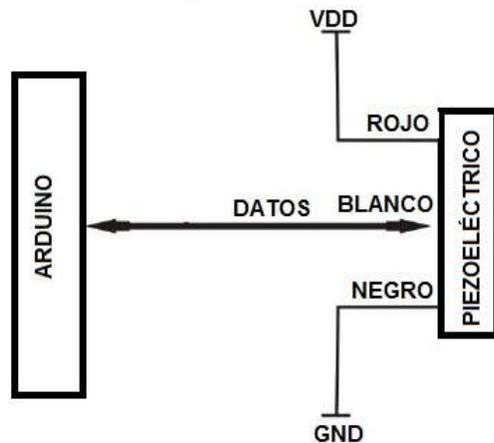
## Metodología y desarrollo

El sensor de temperatura y humedad que se utilizó para este proyecto es el DHT22, que está compuesto por un sensor de humedad capacitivo y un termistor, también cuenta con un circuito integrado básico que realiza la conversión de analógico a digital enviando una señal digital con la temperatura y humedad detectada. La lectura de temperatura es de  $\pm 0.5^{\circ}\text{C}$  de precisión y puede medir de  $-40$  a  $80^{\circ}\text{C}$ . El modo de conexión es el que se muestra en la figura 2.



**Figura 2** Conexión de DHT22 a Arduino

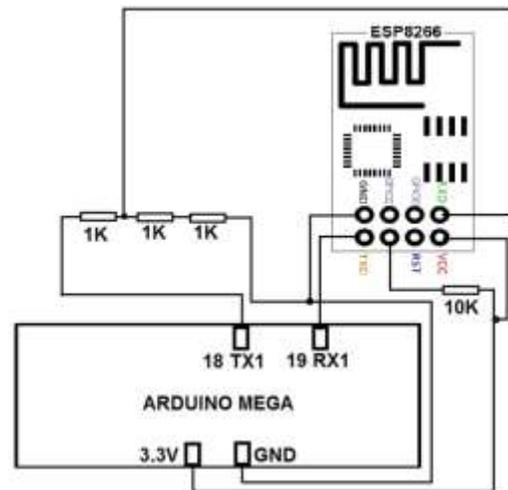
El sensor piezoeléctrico para las vibraciones cuenta con tres cables: rojo, blanco y negro. La conexión es la que se muestra en la figura 3.



**Figura 3** Conexión de sensor piezoeléctrico a Arduino

Primeramente se conecta el módulo ESP8266 con la tarjeta Arduino como se muestra en la figura 4, con el propósito de establecer la comunicación entre ambos componentes utilizando comandos AT (Atención). El código para el Arduino Mega es el siguiente:

```
void setup() {
  Serial.begin(9600);
  Serial1.begin(115200);
}
void loop() {
  while (Serial.available()) {
    char ch = Serial.read();
    Serial1.print(ch);
  }
  while (Serial1.available()) {
    char ch = Serial1.read();
    Serial.print(ch);
  }
}
```



**Figura 4** Conexión de Módulo ESP8266 y Arduino Mega

Una vez verificada la conectividad entre ambos componentes, se instala la librería de Virtuino en el entorno de desarrollo de Arduino y abrimos el ejemplo llamado `webserver_esp01_Example_01` desde el IDE de Arduino, estableciendo el nombre de la red en la que estaremos conectados y la contraseña de la misma; además añadimos el puerto 8000 y una dirección IP válida para establecer la comunicación. Quedando de la siguiente manera:

```
#include "VirtuinoEsp8266_WebServer.h"
VirtuinoEsp8266_WebServer
virtuino(Serial1,115200);
void setup()
{
  virtuino.DEBUG=true;
  Serial.begin(9600);
```

```

virtuino.connectESP8266_toInternet("INFINIT
UMD7A3","werT1988",8000);
virtuino.esp8266_setIP(192,168,1,200);
virtuino.password="1234";
pinMode(13,OUTPUT);
void loop(){
  virtuino.run();
}

```

Es probable que se genere un error en la línea

```

VirtuinoEsp8266_WebServer
virtuino(Serial1,115200); y para corregirlo es
necesario ir a la librería de Virtuino y editar el
archivo "VirtuinoEsp8266_Webserver.h" estable
ciendo como comentario la línea de código:
##define
ESP8266_USE_SOFTWARE_SERIAL

```

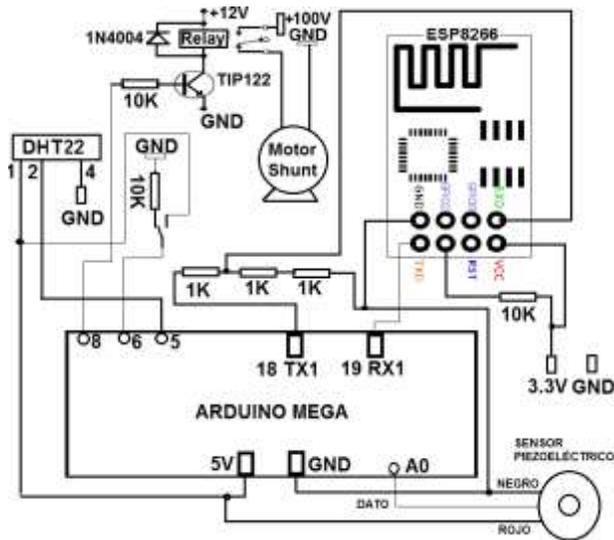
Posteriormente guardar, compilar, cargar a Arduino y verificar que el programa funcione abriendo el monitor del IDE Arduino y que se haya establecido la conexión entre Arduino Mega y la red Wifi por medio del módulo ESP8266.

Para monitorear las variables de temperatura y las vibraciones mecánicas, instalamos la App Virtuino que se descarga de la Play Store para el sistema operativo Android. Al ejecutar la App Virtuino, seleccionamos la opción de Arduino server setting añadiendo la misma dirección IP que utilizamos para establecer la comunicación con el módulo ESP8266, el puerto 8000 y en este seleccionamos la tarjeta Arduino Mega (ver figura 5).



Figura 5 Dirección IP valida en virtuino

Teniendo energizado el hardware y el monitor de Arduino abierto, realizamos la prueba de conectividad que establecerá la comunicación entre la App Virtuino y nuestra tarjeta Arduino Mega. En caso de indicarnos un error de comunicación, debemos instalar un firmware actualizado a nuestro módulo ESP8266. La conexión se establece y guardamos la configuración. En esta ocasión nos interesa trabajar con las variables de temperatura y vibración mecánica, por lo que construimos la interface final como se muestra en la figura 6.



**Figura 6** Interface con todos los componentes del sistema

El siguiente código es el utilizado para realizar la comunicación final entre la interface y la App Virtuino.

```
#include <DHT.h>
#define DHTPIN 5//conecta DHT22 al pin 5
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
#include "VirtuinoEsp8266_WebServer.h"
VirtuinoEsp8266_WebServer
virtuino(Serial1,115200);
void setup()
{
  virtuino.DEBUG=true;
  Serial.begin(9600);

  virtuino.connectESP8266_toInternet("INFINIT
  UMC7A3","werT1988",8000);
  virtuino.esp8266_setIP(192,168,1,200);
  virtuino.password="1234";
  dht.begin();
  pinMode(8,OUTPUT); //Motor ON/OFF
}
long timeStored=0;
void loop(){
  virtuino.run();//commando para comunicación
//con virtuino
//Lectura de sensor DHT22 cada 10 segundos
```

```
if (millis()> timeStored+10000) {
  float t = dht.readTemperature();
  float h = dht.readHumidity();
  if (isnan(t) || isnan(h)) {
    if (virtuino.DEBUG) Serial.println("Failed
to read from DHT");
    virtuino.vMemoryWrite(0,-1000);
    virtuino.vMemoryWrite(1,-1000);
  }
  else {
    if(virtuino.DEBUG)
    Serial.println("Temp="+String(t)+" *C");
    if(virtuino.DEBUG)
    Serial.println("Humidity="+String(h)+" %");
    virtuino.vMemoryWrite(0,t);
    virtuino.vMemoryWrite(1,h);
  }
  timeStored= millis();
}
//pin A0 para el sensor piezoeléctrico
int value= analogRead(A0);
if (value<100) virtuino.vMemoryWrite(2,1);
else if (value>900)
virtuino.vMemoryWrite(2,2);
else virtuino.vMemoryWrite(2,0);
}
```

Antes de compilar el programa .ino, es necesario considerar que se debe instalar y actualizar la librería para el sensor DHT22 de [https://github.com/adafruit/Adafruit\\_sensor](https://github.com/adafruit/Adafruit_sensor). Una vez compilado y cargado el programa en la tarjeta de Arduino Mega, se visualiza primeramente que se establece la comunicación entre la interface y la conexión wifi como se muestra en la figura 7.

```

OK
ets Jan 8 2013,rst cause:2, boot mode:(3,0)

load 0x401AT+GMR
AT version:1.3.0.0(Jul 14 2016 18:54:01)
SDK version:2.0.0(656edbf)
compile time:Jul 19 2016 18:44:22
OK
WIFI DISCONNECT
AT+CWMODE=1

OK
AT+CWJAP="INFINITUM", "1988"
WIFI CONNECTED
WIFI GOT IP

```

Figura 7 Comunicación entre wifi e interface vista en monitor de IDE Arduino

Abrimos la App Virtuino y desarrollamos la interface gráfica para interactuar entre el personal y la interface. Para este proyecto se trabaja sobre dos pantallas, la principal cuenta con un indicador de temperatura y humedad, así como un botón para activar o desactivar al motor (ver figura 8).



Figura 8 Temperatura/humedad y ON/OFF de motor

En la segunda pantalla se muestran las vibraciones mecánicas del motor Shunt con tres niveles que son clasificados como: baja, normal y alta ver (figuras 9, 10 y 11 respectivamente).



Figura 9 Vibración mecánica baja



Figura 10 Vibración mecánica normal



Figura 11 Vibración mecánica alta

Los valores que se muestran a través de la app Virtuino, de igual manera son visibles por medio del monitor que se encuentra en el IDE Arduino (ver figura 12). Esta información puede ser registrada por medio de algún software como Lab View de National Instruments para el análisis de la información a considerar en los mantenimientos predictivos de un motor shunt.

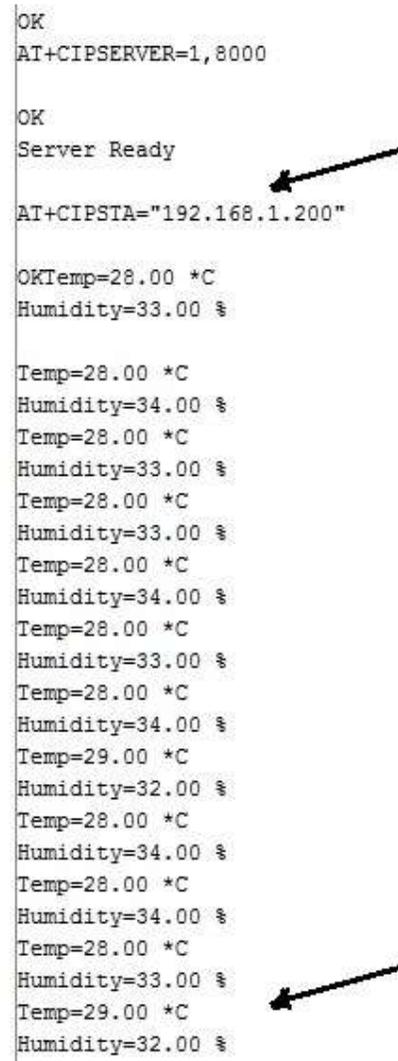


Figura 12 Variable en monitor Arduino

### Conclusiones

Dado que el internet de las cosas crece a pasos acelerados y pretende cambiar al mundo conectándolo todo al internet, el uso de esta tecnología en la industria también es un hecho y los costos para la implementación no son elevados.

En este caso para la construcción de la interface electrónica el costo es relativamente bajo y las aplicaciones de Virtuino así como la Arduino no tienen costo alguno, lo que permite desarrollar aplicaciones para el sensado no únicamente dos variables físicas en un motor shunt y activarlo o desactivarlo, sino que también poder monitorear más variables y controlar los equipos remotamente; así como el uso de Labview de National Instruments que puede utilizarse para el registro de los datos monitoreados y posteriormente realzar el análisis de la información para la toma de decisiones en los mantenimientos predictivos.

En el caso de las pequeñas o micro empresas, donde el personal de mantenimiento requiera del uso de una tecnología cómoda, eficiente y económica; utilizar los principios de este proyecto pueden resultar de gran importancia dados la sencillez de operación y el bajo costo del mismo.

## Referencias

- Félix Cesáreo Gómez de León. (1998). El mantenimiento industrial. En *Tecnología del mantenimiento industrial* (25-28). España: Universidad de Murcia.
- Débora J. Slotnisky. (2016). *Transformación digital: cómo las personas y las empresas deben adaptarse a esta revolución.* Argentina: Digital House.
- Marco Schwartz. (2016). *Internet of Things with Arduino*. Birmingham UK: Packt.
- Google. (2017). Virtuino. 15/Febrero/2017, de Google Sitio web: [https://play.google.com/store/store/apps/details?id=com.virtuino\\_automation.s.virtuino&hl=es](https://play.google.com/store/store/apps/details?id=com.virtuino_automation.s.virtuino&hl=es)
- Agus-Kurniawan.(2017).*Arduino. Programming with .NET and Sketch*. Indonesia: Apress.
- Michael J. McGrath and Clíodhna Ní Scanaill. (2013). *Sensor Technologies*. London UK: Apress.
- Anderson, Rick, Cervo, Dan. (2013). *Pro Arduino. Arduino expert topics and techniques*. United States: Apress.
- Vargas Manuel, Castillo Georgina, Sandoval Juan & Brambila Alfredo. (14 Septiembre 2015 ). *Arduino una Herramienta Accesible para el Aprendizaje de Programación*. *Revista de Tecnología e Innovación*, 2 N0.4, 810-815.
- CANO-LARA, Miroslava, ARMENTA-LOREDO, Miquel, CABAL-YEPEZ, Ernesto y JUÁREZ-RIOS, Higinio. (14 Junio 2016 ). *Diseño y simulación de un sistema de control Smart Home*. *Revista de Aplicación Científica y Técnica*, 2 N0.4, 13-19.
- Lajara Vizcaíno, José Rafael/pelegrí Sebastián, José. (2014). *Sistemas Integrados con Arduino*. México: Alfaomega, Marcombo.
- Núñez González Gerardo, Velázquez Pérez Domingo, Pelayo Cortes Francisco y Barboza Jiménez Pedro. (Diciembre 2016). *Monitoreo de la temperatura y humedad relativa de un aula de cómputo en la Universidad de Guadalajara*. *Revista de Aplicación Científica y Técnica*, Vol.2 No. 6, 16-21.
- Gómez Ramos Marcos Yamir, García Amaro Ernesto y Reyna Ángeles Omar. (Junio 2016). *Aplicación Móvil para Control Escolar*. *Revista de Aplicación Científica y Técnica*, Vol. 2 No. 4, 1-5.
- Cano Lara Miroslava, Armenta Loredo Miguel, Cabla Yepez Ernesto y Juárez Rios Higinio.

(Junio 2016). Diseño y simulación de un sistema de control Smart Home. Revista de Aplicación Científica y Técnica, Vol. 2 No. 4, 13-19.

Arroyo Ledesma Jaime, Contreras Aguilar Luis, Jiménez Betancourt Ramón y Venegas Trujillo Tiberio. (Septiembre 2016). Modelado bilineal de un motor de CD. Revista de Investigación y Desarrollo, Vol. 2 No. 5, 61-70.  
Fernando Reyes Cortes. (2015). Arduino: aplicaciones en robótica, mecatrónica e ingenierías. México: Marcombo.