Desarrollar un sistema permite el control de distintos sensores, actuadores y periféricos en Java

Jhordy Bayas[1,2][L00365906], Joshua Reyes[1,3][L00372677], and Roger Espinoza[1,4][L00366058]

[1] Universidad de las Fuerzas Armadas E.S.P.E, Sangolqui, Ecuador
[2] jhordyb3@gmail.com
[3] jsreyes@espe.edu.ec
[4] rjespinoza@espe.edu.ec

Abstract. In this document, we present the project of serial connection through Arduino and Java applying different models of sensors and actuators. Currently, technology has been constantly developing pending the information they send or receive such as machine-human interaction, for this reason software has been developed that is responsible for providing and receiving information, through the use of interfaces Java GUI For this reason, the distribution of the data is presented by the actuators, in addition to the LCD that is evaluated in real time, temperature, humidity, CO2 and proximity, with this by applying the corresponding means to analyze them.

Keywords: Arduino · nodeMCU · Temperature.

1 Introducción

Con el avance constante de la tecnología, es posible que una o mas computadores estén conectados de manera serial a través del RS232, esto a su vez puede realizar interaciones de las distintos sensores y actuadores que se pueda dar, entre maquina-usuario, con ello realizan distintos tipos de funciones con la intención de mostrar datos precisos y veraces dependiendo si el usuario los solicita. En la actualidad multitud de plataformas se han venido desarrollando tanto de manera de hardware y software con un formato libre, además de un coste reducido, gran documentación y una comunidad bastante activa, la plataforma Arduino es una de ellas esta a su vez dispone de suficientes librerías y una comunidad

activa que facilitan la interconexión del Arduino con diferentes módulos de sensado, facilitando de esta forma la realización de proyectos deseados, entre ellas las conexiones con otros tipos de lenguaje como es el caso de Java.

2 Objetivos

2.1 Objetivo General

Desarrollar un sistema permite el control de distintos sensores, actuadores y periféricos en Java

2.2 Objetivos específicos

- Especificar las variables para el desarrollo del sistema para los distintos sensores y actuadores.
- Identificar los distintos sensores de CO2, Humedad, Temperatura y de proximidad a controlar mediante Java.
- Deducir las distintas aplicaciones mediante la ulizacación del cable serial, además de las interfaces GUI de Java.
- Diseñar un prototipo de software para que cada uno de los sensores está asociado a un relé para controlar un ventilador, una niquelina, un foco y un LCD para el despliegue de información del sistema.

3 Estado del Arte

Al revisar distintos ambitos en donde se esta llegando a aplicar los distintos módulos, se puede encontrar grandes temas, por el cual se llegó al artículo realizado por J.Chandramohan, R.Nagarajan, K. Satheeshkumar, N.Ajithkumar, A.Gopinath, S.Ranjithkumar los cuales en su paper " Intelligent Smart Home Automation and Security System Using Arduino and Wi-fi" donde se resume una aplicación de un sistema de control y monitoreo del hogar flexible y de bajo costo con la ayuda de un equipo integrado.[**ChandramohanNagarajanSatheeshkumarAjithkumarGopinathR**
Al observar el artículo "Intelligent multi-sensor control system based on innovative technology integration via ZigBee and Wi-Fi networks" de los autores Kuang-Yow Lian, Sung-Jung Hsiao y Wen-Tsai Sung, donde se descrbe que una red de transmisión de datos es uno de los problemas más difíciles de resolver en los sistemas de redes de sensores inalámbricos . ZigBee y Wi-Fi pertenecen a diferentes protocolos de red . Si un sistema de red debe usar ZigBee y Wi-Fi al mismo tiempo para transmitir datos, se presenta un desafío considerable. Este artículo presenta un nuevo método de hardware que integra ZigBee y Wi-Fi. El método propuesto se basa en el módulo portátil de Arduino ZigBee y el concepto de Ethernet. Este estudio construye un sistema inteligente de control de electrodomésticos utilizando la red ZigBee. Este sistema de control inteligente utiliza una arquitectura de red integrada ZigBee y Wi-Fien la casa. Nuestro estudio envía los mensajes del sensor ZigBee a una base de datos en la nube a través

de la red de protocolo TCP / IP que contiene la red física y las líneas de dispositivos de red inalámbrica . El control deacceso a la gestiónse logra mediante teléfonos inteligentes. El método propuesto es muy simple y fácil de implementar utilizando circuitos Arduino. La efectividad del método propuesto se verifica mediante la simulación ylos resultados experimentales. Loscomponentes de hardwareincluyen el controlador Arduino, elmódulo de comunicacióninalámbrica XBee Serie 2ylossensores deldispositivo final. Loslenguajes de programación de-Android yJava. Se utilizan para escribir el teléfono inteligente y los programas de reconocimiento del servidor.[**Lian&Hsiao&Sung2013**]
Viendo otros usos que se pueden dar las utilizaciones del módulo se puede encontrar el articulo "Taking Arduino to the Internet of Things: The ASIP programming model" realizado por Gianluca Barbona, Michael Margolis, Filippo Palumbo, Franco Raimondi, Nick Weldin nos meciona que los microcontroladores como Arduino son ampliamente utilizados por todo tipo de fabricantes en todo el mundo. La popularidad se debe a la simplicidad de uso de Arduino y la gran cantidad de sensores y bibliotecas disponibles para ampliar las capacidades básicas de estos controladores. La última década ha sido testigo de una oleada de soluciones de ingeniería de software para "Internet of Things", pero en varios casos estas soluciones requierenrecursos computacionalesque son más avanzados que los microcontroladores simples y con recursos limitados.En el presenten documento presentan el modelo de Programación de Interfaz de Servicio de Arduino (ASIP), un nuevo modelo que aborda los problemas anteriores proporcionando una abstracción de "Servicio" para agregar fácilmente nuevas capacidades a los microcontroladores, y brindando soporte para tableros en red que utilizan una variedad de estrategias, que incluyen conexiones de socket , dispositivos de puente, publicación basada en MQTT, mensajería de suscripción, servicios de descubrimiento, etc. Ofrecemos una implementación de código abierto del código que se ejecuta en las placas Arduino y las bibliotecas de clientes en Java, Python, Racket y Erlang. Mostramos cómo ASIP permite el rápido desarrollo de aplicaciones no triviales (coordinación de entrada / salida en tableros distribuidos e implementación de un algoritmo de seguimiento de línea para un robot remoto) y evaluamos el rendimiento de ASIP de varias maneras, tanto cuantitativas como cualitativas. [**BarbonaMargolis2016**]
Como se puede observar el tema de la investigación tiene diversas investigaciones, artículos que poseen cierta relación con lo que se esta investigando, pero este trabajo se diferencia de los demás en que se quiere integrar Arduino de manera serial mediante la conexión entre 2 PC. Se profundizará en varios temas que no se han investigado juntos y se orientara más a las distintas ventajas que ofrece la conexión serial, partiendo de que en el arduino se puede realizar tanto conexiones en paralelo como en serie y convertir a su vez la salida como tener actuadores, es decir, los distintos actuadores, como es el ventilador, la niquelina, un foco, y la apreciación de los datos mediante la pantalla LCD, la misma que s epuede encontrar comunmente las libreías en Arduino para ocuparlas. Por lo que nuestro trabajo unirá los beneficios de plataformas gratuitas que ofrezcan servi-

cios especiales con las características del Arduino, la interfaz de Java y además de una aplicación para demostrar el potencial que puede tener el uso de estos.

4 Marco Teórico

4.1 Comandos para la Comunicación Serial

Serial.begin(rate) Abre el puerto serie y fija la velocidad en baudios para la transmisión de datos en serie. El valor típico de velocidad para comunicarse con el ordenador es 9600, aunque otras velocidades pueden ser soportadas.
Nota: Cuando se utiliza la comunicación serie los pines digitales 0 (RX) y 1 (TX) no pueden utilizarse para otros propósitos.

Serial.println(data) Imprime los datos en el puerto serie, seguido por un retorno de carro y salto de línea. Este comando toma la misma forma que Serial.print (), pero es más fácil para la lectura de los datos en el Monitor Serie del software.

5 Diagramas

Diagrama:

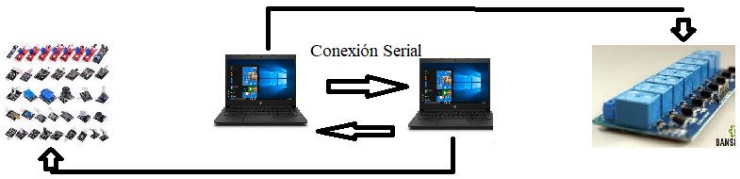

Fig. 1. Monitores Serie

6 Código Fuente

```
#include <dht.h>

#define RLOAD 22.0
#define RZERO 879.13
#include "MQ135.h"
dht DHT;
#define dht_temp A3
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,20,4);

long tiempo;
int disparo = 2;
int echo = 3;
float distancia;
MQ135 gasSensor = MQ135(A0);
int val;
int sensorPin = A0;
int sensorValue = 0;


void setup() {
        lcd.init();
    lcd.backlight();
   lcd.setCursor(10, 0);
  Serial.begin(9600);
    pinMode(disparo, OUTPUT);
  pinMode(echo, INPUT);
  digitalWrite(disparo, LOW);
  pinMode(sensorPin, INPUT);
}

void loop() {
  val = analogRead(A0);
  float ppm = gasSensor.getPPM();
  Serial.print ("ppm: ");
  Serial.println (ppm);
  delay(1000);
  digitalWrite(disparo, HIGH); //Se envia el pulso de activacion del sensor
  delayMicroseconds(10);
  digitalWrite(disparo, LOW);
  tiempo = pulseIn(echo, HIGH); // Obtengo el tiempo del sensor
  distancia = 0.0343*tiempo/2; //Calculo la distancia
  Serial.print(distancia);
```

```
  Serial.println("cm");//Muestro la distancia en el monitor serie
  delay(1000);

  //TEMPERATURA//
   DHT.read11(dht_temp);
  Serial.print("Temperatura = ");
  Serial.print(DHT.temperature);
  Serial.println(" C");
  //HUMEDAD RELATIVA//
  Serial.print("Humedad = ");
  Serial.print(DHT.humidity);
  Serial.println(" %");
  delay(1000);
 sensorValue = map(analogRead(A1), 0, 1023, 0, 100);
  Serial.print("Umbral de Ruido :");
  Serial.print(sensorValue);
   Serial.println(" %");
     delay(1000);
  lcd.setCursor(0,0);
   lcd.print("d:");
      lcd.print(distancia);
        delay(100);
        lcd.setCursor(6,0);

   lcd.print("G:");
      lcd.print(ppm);
  delay(100);
      lcd.setCursor(12,0);

   lcd.print("R:");
      lcd.print(sensorValue);
       delay(100);
         lcd.setCursor(0,1);

   lcd.print("T: ");
      lcd.print(DHT.temperature);
      lcd.setCursor(8,1);
      lcd.print("H: ");
      lcd.print(DHT.humidity);
      lcd.print(" %");
        delay(100);
}
```

En el codigo anterior se explica el funcionamiento de los sensores y de como va a enviar los datos serialmente el Arduino y que datos nomas va a enviar.

Como se puede ver vamos a tener datos de humedad, temperatura, distancia, ruido y CO2.

7 Conclusiones

- Arduino es muy versátil para el desarrollo de proyectos ya que es fácil de trabajar y acoplar sus módulos con lo cual nos da una amplia capacidad para realizar sistemas, aplicaciones en casi cualquier tema.
- La conexión serial entre placas Arduino se lo puede realizar de manera sencilla con los comandos "Serial" que nos ofrece Arduino.
- Con la facilidad que se realiza las conexiones seriales se podría para un trabajo futuro realizar sistemas mas complejos donde se podría enviar los datos también por internet a un servidor y añadirle a las computadoras un programa externo que envié datos al Arduino.
- Las distintas conexiones seriales de arduino, pueden conectarse indistintamente en cualquier tipo de placa basada en el mismo, como nodeMCU o un sensor de temperatura, se puede concluir que la ventaja de arduino es poder incluir este tipo de uniones, y también encontrar dichas librerías para una óptima utilización de la programación realizada.

8 Recomendaciones

- Tener conocimientos básicos de lo que es el código arduino y lo que representa las conexiones seriales.
- Comprender la naturaleza de su concepto y luego introducirnos en las características que incorpora este nuevo lenguaje.
- Estudiar el diagrama de conexiones del arduino, y las normas básicas para que el mismo funcione.
- Comprender el funcionamiento básico, pines de conexión y alimentación del módulo ESP-8266.
- Se recomienda verificar detalladamente la estructura del prototipo a diseñar, teniendo ideas claras, y el enfoque a donde va dirigido.