Congr. Int. en Ing. Electrónica. Mem. ELECTRO, Vol. 39, pp. 22-26, Oct 2017, Chihuahua, Chih. , México
http://electro.itchihuahua.edu.mx/memorias_electro/MemoriaElectro2017.zip
ISSN 1405-2172

SISTEMA DISTRIBUIDO CON ARDUINO, WIFI Y LABVIEW, UTILIZANDO EL PROTOCOLO UDP.

Hernández R. Sergio I., Espinoza L. Frances J., Aldaz A. Kassandra G., Juzaino Z. Stephanie R., Hernández J. María G., Pelayo M. Vanessa M.

Instituto Tecnológico de Nogales
Ave. Instituto Tecnológico Núm. 911, Nogales, Sonora. C.P. 84065
Teléfono (631) 123- 9703.
sergio.hernandez@cicsa-net.com.mx

RESUMEN.

El presente trabajo de investigación se realizó en el Instituto Tecnológico de Nogales y contempla el estudio e implementación de un sistema de comunicación distribuido, basado en el protocolo UDP (User Datagram Protocol). Se desarrolló una aplicación en el leguaje de programación gráfica LabView, la cual se comunica con la tarjeta de expansión Shield Wifi para Arduino, utilizando el protocolo ya mencionado. La aplicación puede ser aplicada a diferentes ambientes o áreas de trabajo, como el entorno industrial, el sector de bienes y servicios entre otros, para un uso en específico. En este caso se realizó el estudio desarrollando una aplicación orientada al área de las tecnologías de la información, comunicación y electrónica con el fin de valorar el rendimiento de este protocolo con la tarjeta de expansión Wifi Shield de Arduino.

Palabras Clave: Wifi, UDP, Arduino, LabView.

ABSTRACT.

The present research work was carried out at the Instituto Tecnológico de Nogales and contemplates the study and implementation of a distributed communication system, based on the UDP protocol (User Datagram Protocol). An application was developed in the LabView graphical programming language, which communicates with the Shield Wifi expansion card for Arduino, using the aforementioned protocol. The application can be applied to different environments or areas of work, such as the industrial environment, the sector of goods and services, among others, for a specific use. In this case, the study was carried out by developing an application oriented to the area of information, communication and electronic technologies in order to evaluate the performance of this protocol with the Arduino Wifi Shield expansion card.

Keywords: Wifi, UDP, Arduino, LabView.

1. INTRODUCCIÓN.

Las primeras redes que comunicaban computadoras se instalaron en EE.UU. en los años 60. En aquella época prácticamente cada red estaba implementada con una tecnología distinta e incompatibles unas con otras. A principios de los 70 surgió un nuevo concepto: la interconexión de redes. Ya no basta con comunicar computadoras sino que había que comunicar redes de computadoras distintas [1]. Actualmente la tendencia a la globalización exige que los sistemas y aplicaciones informáticos, cuenten con herramientas potentes que le permitan una comunicación entre todos los niveles [2]. Estos sistemas se denominan: sistemas distribuidos y sus principales ventajas son:

- Rendimiento.
- Fiabilidad.
- Escalabilidad.

Las aplicaciones distribuidas son aquellas construidas sobre sistemas distribuidos, donde los aspectos de complejidad como: localización, replicación, fallos y concurrencia, sean transparentes al programador de aplicaciones. A continuación se presenta la implementación de un sistema distribuido utilizando el protocolo UDP, LabVIEW y la interface Arduino[3].

2. MARCO DE REFERENCIA.

2.1. Arduino.

Arduino es una plataforma de desarrollo de computación física (physical computing) de código abierto, basada en una tarjeta con un microcontrolador y un entorno de desarrollo para crear software (programas). Arduino se utiliza para crear aplicaciones interactivas, obteniendo datos de una gran variedad de interruptores y sensores, para el control de diferentes dispositivos [4]. Los proyectos con Arduino pueden ser autónomos o comunicarse con un programa (software) que se ejecute en una computadora. El hardware está disponible para realizarlo uno mismo o se puede comprar ya listo para su uso, el software de desarrollo es abierto y se descarga sin ningún costo [1]. El lenguaje de programación de Arduino es una implementación de Wiring, una plataforma de computación física, que a su vez se basa en Processing, un entorno de programación multimedia.

Figura 1. Tarjeta Arduino Uno.

Para la conexión se utilizó el dispositivo Wifi Shield, el cual permite conectar la tarjeta Arduino a la red mediante wifi. Este dispositivo se comunica con Arduino a través del bus SPI (Serial Peripheral Interface) mediante los pines ICSP y usa los pines 4 y 10 al igual que la Shield Ethernet. Provee una pila de red IP (Internet Protocol) utilizando los protocolos TCP (Transmission Control Protocol) y UDP.

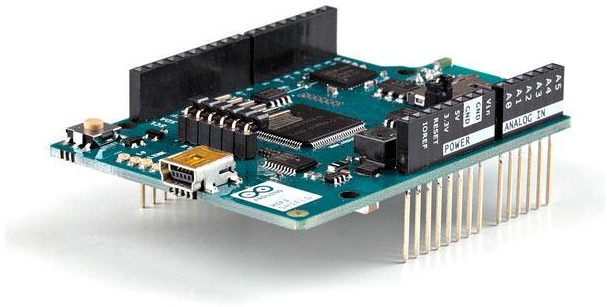

Figura 2. Wifi Shield.

2.2. UDP.

El protocolo UDP, definido en la RFC 768. No añade fiabilidad, control de flujo o recuperación de errores a IP cuando funciona sobre él, simplemente trabaja como un multiplexor/demultiplexor para enviar y recibir datagramas, usando los puertos para dirigir los datagramas. Los puertos permiten la comunicación entre las aplicaciones (procesos), el campo del puerto tiene una longitud de 16 bits, por lo que el rango de valores válidos es de 0 a 65535. En la figura 3 se muestra la estructura UDP, por ejemplo el proceso 1 es una aplicación desarrollada, donde se especifica el número de puerto de destino, el número de puerto origen y la dirección IP a utilizar para la comunicación. El demultiplexado en el dispositivo receptor pasa los segmentos recibidos por el puerto indicado al socket correcto, el multiplexado en el dispositivo emisor colecta los datos envolviéndolos con una cabecera que posteriormente será utilizada para demultiplexar.

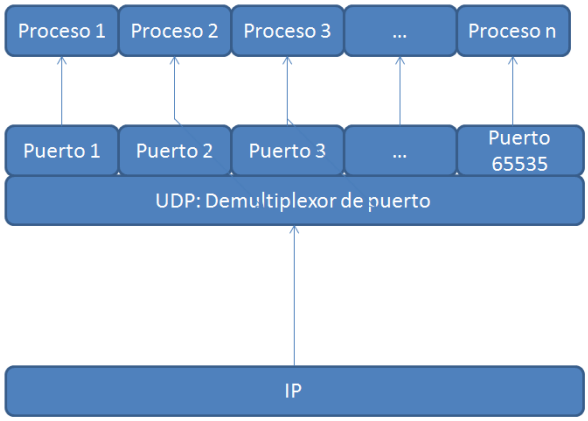

Figura 3. Estructura UDP.

En IP hay direcciones que sirven para encaminar la información de un host a otro a través de la red, en UDP hay puertos que sirven para identificar qué aplicación está asociada a esa información. Por tanto para establecer una conexión entre dos dispositivos se necesitan estos dos datos básicos: puertos y dirección IP. A esta pareja de datos se le llama socket [5].

2.3. LabView.

LabView es el acrónimo de Laboratory Virtual Instrument Engineering Workbench. Es un lenguaje y a la vez un entorno de programación gráfica en el que se pueden crear aplicaciones de una forma rápida y sencilla. National Instruments es la empresa desarrolladora y propietaria de LabView, que comenzó en 1976 en Austin, Texas y sus primeros productos eran dispositivos para el bus de instrumentación GPIB. LabView es una herramienta de programación gráfica. Originalmente este programa estaba orientado a aplicaciones de control de instrumentos electrónicos, usadas en el desarrollo de sistemas de instrumentación, lo que se conoce como instrumentación virtual [6].

Figura 4. LabView.

3. DESARROLLO.

Se desarrolló e implementó un sistema distribuido basado en el protocolo UDP utilizando la Shield Wifi de Arduino, para la transferencia de información entre la computadora y un dispositivo Arduino, el sistema se enfocó en estudiar el rendimiento de la mencionada tarjeta en relación a la distancia-potencia (fig. 5).

Figura 5. Sistema de comunicación.

UDP es un protocolo basado en el envío y recepción de datagramas, este servicio es sin conexión, es decir no se establece una sesión entre los dispositivos, de tal manera que UDP no garantiza ni confirma la entrega de la información, por tal razón la aplicación desarrollada debe proporcionar la confiabilidad necesaria para el transporte de los datos de una manera correcta. Es necesario asignarle una dirección IP y un puerto para indicar el destino de la información a transmitir.

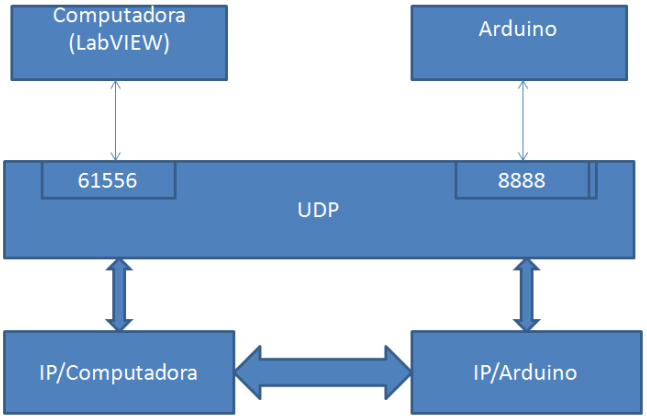

Figura 6. Estructura de la aplicación.

La tarjeta Arduino se conecta a la red a través de la Shield Wifi de Arduino, se puede configurar como un servidor al aceptar conexiones entrantes o como cliente que realiza conexiones salientes. La Shield Wifi utiliza el protocolo SPI para la comunicación con la tarjeta Arduino, se utilizó el sketch proporcionado como ayuda en la IDE de arduino llamado WifiUdpSendReceiveString[1].

La aplicación para la computadora se realizó en LabView, como se mencionó anteriormente el protocolo UDP no garantiza, ni confirma la entrega de la información, razón por la cual fue necesario desarrollar e implementar un método para asegurar la confiabilidad del sistema y verificar el transporte de los datos para que lleguen de manera correcta. Este método fue probado anteriormente en [3] y utilizado en el presente trabajo con buenos resultados.

Figura 7. Aplicación en LabView.

En la figura 7, se muestra la interface de la aplicación, donde se puede observar que se debe configurar los puertos del dispositivo local como remoto para establecer la comunicación. Para verificar el funcionamiento del sistema se utilizó el puerto analógico A0 de la tarjeta Arduino UNO, para enviar información a la computadora simulando la lectura de algún sensor analógico.

Cuando Arduino envía la información, esta puede ser monitoreada mediante la aplicación desarrollada. Un fragmento del código se muestra a continuación.

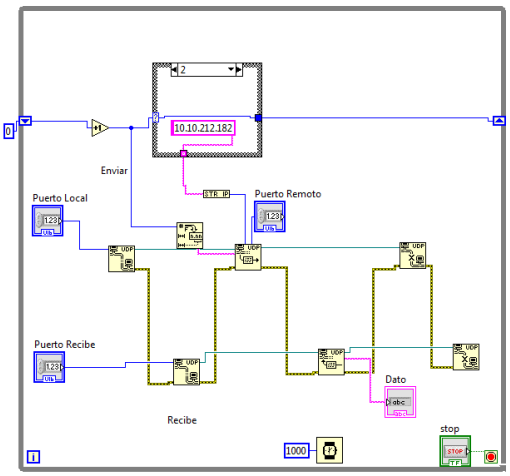

Figura 8. Código en LabVIEW.

Se utilizó las funciones de la paleta UDP, disponibles en diagrama de bloques de LabView, el cual necesita configurar el puerto donde será creado el enlace UDP (socket). Se le tiene que indicar la dirección de red del dispositivo a escuchar, es necesario si se desea comunicar sólo con un dispositivo, si no se especifica la dirección de red LabView escuchará en todas las direcciones.

4. RESULTADOS.

La aplicación en la computadora colecta los datos que provienen de los sockets, envolviéndolos con una cabecera (que posteriormente será utilizada para demultiplexar). La aplicación en el dispositivo recolecta los datos y pasa los segmentos recibidos al socket indicado (ver figura 9). Las capas internas del protocolo realizan esta operación, únicamente se le asigna el número de puerto a utilizar por la aplicación. El socket UDP se identifica por: dirección IP origen, número de puerto origen, dirección IP destino y número de puerto destino. Cuando se recibe un segmento UDP, se verifica el puerto destino y conduce al segmento UDP hacia el socket con ese puerto.

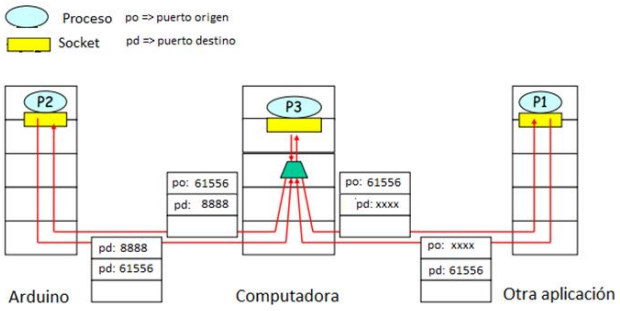

Figura 9. Multiplexado/demultiplexado sin conexión.

El problema de la pérdida de datos es especialmente difícil, debido a que la pérdida puede ocurrir a la mitad de la red, incluso cuando los dispositivos de origen y destino no detectan problema alguno. Tomando en consideración el párrafo anterior y basados en las especificaciones del protocolo UDP, el cual no contempla conexión para el aseguramiento de la información, se tuvo la necesidad de que la aplicación desarrollada fuera diseñada con ciertas características, configuraciones y temporizadores extras para manejar el problema de pérdidas de información. De ser necesario el transmisor retransmite los datos en tres ocasiones la menos que llegue una confirmación antes indicando que la información fue recibida correctamente antes de que expire el temporizador.

En relación a la potencia y distancia se presentan los resultados obtenidos en la medición en la siguiente tabla:

Tabla 1. Relación distancia-potencia.

Distancia (cm)	Potencia (dBm)
50	- 32
100	- 45
150	- 41
300	- 48
600	- 45
1200	- 75
1800	- 77

5. CONCLUSIONES.

Uno de los principales objetivos del presente trabajo fue el de analizar el desempeño de la tarjeta de expansión Wifi para Arduino en relación de potencia-distancia, indicando los resultados en la etapa anterior.

El trabajo presenta de manera general el contexto y los requerimientos mínimos necesarios para el funcionamiento y comunicación del sistema, Aunque el protocolo UDP no presenta propiedades para el aseguramiento de la información, la aplicación desarrollada cumplió satisfactoriamente con los puntos de confiabilidad, control de flujo y multiplexado. Haciendo que los aspectos de complejidad como: localización, replicación, fallos y concurrencia, sean transparentes para el usuario.

Unas de las razones por la cual se decidió utilizar el protocolo UDP en lugar de utilizar TCP, es que a futuro se pretende

integrar al sistema un módulo GPS y una cámara, para ser colocados en un automóvil, aprovechando que es un protocolo de transporte muy ligero y no se pierde tiempo en establecer una conexión.

6. REFERENCIAS.

[1] http://www.arduino.cc/es/.
[2] Sergio I. Hernández R., Pedro Ochoa M., and Julio C. Ramírez V. "Bluetooth and OPC (OLE for Process Control) for the Distributed Data Integration". Fourth Congress of Electronics, Robotics and Automotive Mechanics. IEEE, 2007.
[3] Hernández R. Sergio I., Ortiz N. Bertha L., Meranza C. Manuel O., Bonifaz Estefany, Gutiérrez Aarón, Duarte Rubí, "Sistema Distribuido con Arduino y LabView, utilizando el protocolo UDP". Congreso Internacional en Ingeniería Electrónuica Electro 2014, Vol. 36, pp 352-355, ISSN 1405-2172.
[4] Evans Brian, "Beginning Arduino Programming", writing code for the most popular microcontroller board in the world, Technology in Action, 2011, ISBN: 978-1-4302-3778-5.
[5] Herrera Enrique, "Tecnologías y Redes de Transmisión de Datos", Ed. Limusa, 2003, ISBN: 968-18-6383-6.
[6] Lajara José, Pelegrí José, "LabVIEW, Entorno Gráfico de Programación", Ed. Alfaomega, 2007, ISBN: 978-970-15-1133-6