Sensor de aparcamiento por ultrasonido con conectividad RS232 (Asíncrono).

Juan Carlos Noguera, Jhon Kevin Muñoz. Escuela de Ingeniería Eléctrica y Electrónica Universidad del Valle

Abstract—El presente documento pertenece a la tercera y fase final "sensor de aparcamiento por ultrasonido con conectividad RS485". A diferencia de la anterior fase la conectividad implementada en hardwared es utilizando el circuito MAX232 que utiliza la interfaz de comunicación RS232. A parte de la comunicación se integra el sistema de luces accionado por medio de la señal de salida proporcionado por dispositivo de ultrasonido HC-SR04. El microcontrolador utilizado es el ATmega32 y la comunicación con este ultimo es utilizando un PC. El software utilizado para el envió y recepción de tramas en PC es Docklight. Se mencionan también los cambios realizados con respecto a los diseños y planteamientos expuestos en el informe de la primera y segunda fase. Se utilizo Atmel Studio en la versión 6.0.

Palabras Clave—Microprocesador, Modulo, Polling, RS232, Sensado, Transceptor, Ultrasonido.

I. Introducción.

El sistema a desarrollar esta conformado por el sensor de ultrasonido capaz de proporcionar la señal necesaria para la obtención de la distancia al excitar el sensor con un voltaje de entrada TTL, dichas señales son proporcionadas y a la vez actúan sobre el microprocesador Atmega32, a su vez de acuerdo al tiempo calculado se proporciona a los led una señal alto o bajo para determinar la presencia de un automóvil en una zona de estacionamiento. Se tiene un led verde para determinar que la zona esta libre y un led rojo para cuando esta ocupada. Otro modulo que pertenece al sistema es la interfaz de comunicación con su debido protocolo. La interfaz RS232 permite conexión maestro/esclavo y es de tipo Full-Duplex al utilizar el circuito integrado MAX232, aunque por medio de software se configuro para que la operación fuera Half-Duplez o solo teniendo respuesta del microcontrolador solo cuando se envía una trama valida desde el master. La dirección en el código permite al maestro reconocer a cual de los dispositivos esclavos es con el cual se esta comunicando. El sistema esta configurado como esclavo y debe estar en la capacidad de proporcionar el estado actual del sistema según lo requiera el maestro. La técnica para el desarrollo del software para los diferentes módulos que pertenecen al sistema es el polling. Para integrar ambos módulos se utilizo el kernel de tiempo real FreeRTOS. El reloj por el cual esta operando el ATmega32 es un cristal externo de 16MHz. La primera sección de este documento se conoce como: Descripción del algoritmo y diagrama de los módulos donde se se tienen los diagramas esquemáticos de cada modulo, el funcionamiento de cada pin del microcontrolador y los registros necesarios para la conexión con los periféricos. Los resultados y su respectiva

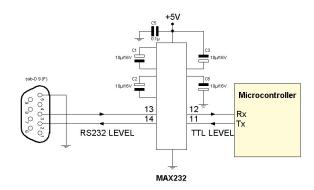


Fig. 1. Diagrama esquemático para comunicación entre ATmega32 y PC.

discusión conforman la siguiente sección donde se exponen las señales medidas mediante osciloscopio.

II. DESCRIPCIÓN DEL ALGORITMO Y DIAGRAMA DE LOS MÓDULOS.

A. Interfaz de comunicación.

Para utilizar la interfaz de comunicación RS232 Fig 1 es necesario configurar los registros del microcontrolador para utilizar la comunicación serial asíncrona (UART), habilitando como salida el pin TX para transmitir y RX como pin de entrada para recibir. La velocidad en baud es 9600 y se escoge el modo asíncrono y formato de frame de 8 bits de datos, 1 bit de stop y sin paridad. El circuito integrado por el cual se pasa de señales TTL a señales tipo serie es el MAX232. Para transmitir es necesario cargar en el registro UDR el dato. Para recibir se descarga el registro UDR en una variable de 1 byte. El PC es usado como maestro y desde el cual se envían las tramas. Aunque debido a que la PC carece de puerto DB9, es necesario el uso del adaptador USB a puerto serial de referencia BF-810. En el software Docklight se configura de igual manera la misma velocidad, formato de frame, bit de stop y paridad configurado en el microcontrolador. La trama a enviar y a recibir esta divida en tres partes Cabezera (Header), Tipo de Trama (FRAME_TYPE) y dato (DATA). La cabezera esta constituido por 3 byte donde cada uno tiene el hexadecimal 55. El tipo de trama define si esta detectando la presencia del dispositivo, se esta preguntando por el estado del estacionamiento o configurando la mínima ò máxima distancia. El dato esta compuesto por 5 byte donde el ultimo es usado como la dirección del dispositivos. Los 4 bytes restantes del

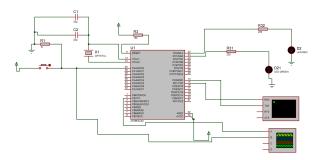


Fig. 2. Diagrama esquemático para sensado y activación de luces led.

dato son usados para representar un punto flotante, para lo cual se utiliza una unión y así poder operar dentro del algoritmo. El microcontrolador al reconocer la trama actúa con respecto a dicha información, alterando variables si es necesario y devolviendo una respuesta de 5 byte donde los 3 primeros byte sirven de cabezera, un byte de tipo de trama y un ultimo byte para el dato.

B. Modulo de luces y sensado.

En Fig 2 el botón representa el pin ECHO del HC-SR04 debido a que la señal es de tipo TTL. Mediante un retardo mediante la función delay se genera la señal de ancho de $12\mu s$ para excitar el pin de entrada TRIG del sensor lo cual provoca internamente en el HC-SR04 una señal de 8 ciclos de 40kHz para después asignar en el pin ECHO un nivel lógico alto que es recibido como señal de entrada en el pin PA0. Mientras el pin PA0 este en alto se habilita el temporizador TIMER0 con un preescalado de $clk_{I/O}/8$. Al terminar lo anterior se guarda el valor del registro TCNT0 en una variable flotante, se inhabilita el temporizador nuevamente y se carga el valor TCNT0 con un valor de 0. Con el valor guardado en la variable se procede a calcular la distancia mediante.

Para mayor estabilidad en el comportamiento del modulo se considera adquirir 10 muestras de distancias y hallar el promedio. El rango comprendido entre 3.8m a 0.3m se considera como plaza ocupada (PC0 en alto) y para los demás rangos como plaza libre (PC1 en alto). El uso del TIMER0 y no del TIMER1 utilizado en la anterior fase se debe a que se esta utilizando FreeRTOS, y este ultimo utiliza el TIMER1 para conmutar entre tareas. Debido a lo anterior existe una restricción del TIMER0 ya que tiene la capacidad de realizar 255 cuentas, lo cual no alcanza para medir la distancia máxima de 3.8m. La solución para lo anterior fue utilizar una variable capaz de aumentar en 1 cada vez que se alcanzaba el desbordamiento (overflow) en el registro TCNT0.

III. RESULTADOS Y DISCUSIÓN.

En la Fig 3 corresponde al montaje con todos los módulos que pertenecen "Sensor de aparcamiento por ultrasonido con conectividad RS232 (Asíncrono)", donde esta conglomerado el modulo de sensado y encendido de luces y la comunicación. Mediante el software Docklight es posible observar que solo se tiene respuesta del ATMega32 si se envía una trama desde

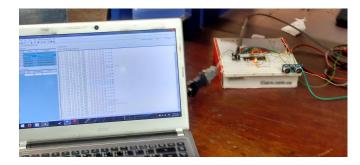


Fig. 3. Montaje con los módulos de comunicación y sistemas de encendido de luces.

el master o PC. A su vez se tiene el caso cuando se detecta obstáculo en el rango configurado de 1.0m a 1.5m.

A. Interfaz de comunicación.

En la Fig 4 se tiene la señal de salida del microcontrolador al enviar desde el PC la trama correspondiente a detectar presencia y Fig 5 la señal de salida por el pin TX al enviar la trama para configurar la máxima distancia a 1.5m. Debido a la velocidad de transmisión no es posible determinar a través de la medición por osciloscopio que es la trama correcta, pero si es posible determinar que la comunicación es a través del UART.

B. Modulo de luces y sensado.

La Fig 6a corresponde al montaje cuando se tiene la plaza libre o no esta dentro del rango de 0.3m a 3.8m y Fig 6b corresponde al caso contrario. La Fig 6c es la señal de voltaje en el pin Echo, el ancho de tiempo a un nivel de voltaje lógico alto cuando se tiene la distancia mínima de 0.3m y Fig 6d corresponde cuando la distancia es máxima o de 3.8m.

IV. DIAGRAMA DE FLUJO.

El diagrama de flujo Fig 7 considera dos programas ejecutados de manera concurrente y en paralelo mediante el uso de FreeRTOS donde se tiene una tarea para sensar y encender luces de acuerdo al estado de la zona de estacionamiento y la tarea para comunicación entre maestro y esclavo. Ademas de considerar para la tarea de sensar y encender luces el promedio

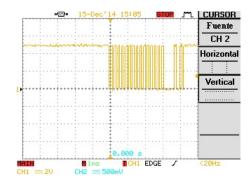


Fig. 4. Señal de salida del pin TX del ATmega32 al recibir del PC la trama detectar presencia .

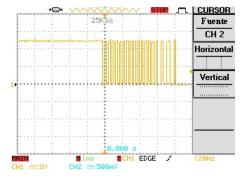


Fig. 5. Señal de salida del pin TX del ATmega32 al recibir del PC la trama configurar máxima distancia 1.5m.

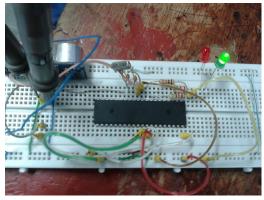
de 10 muestras de distancias y el calculo de esta ultima. Para el modulo de comunicación en la fase actual se considero primero obtener la trama y luego preguntar por cada byte que lo compone.

V. OBSERVACIONES.

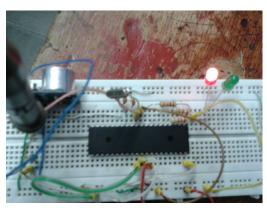
- El modulo correspondiente al sistema de luces presento cambios en software al utilizar el TIMERO.
- La interfaz utilizada para comunicar entre el PC y el microcontrolador fue RS232.

REFERENCES

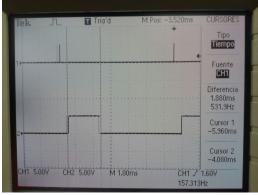
- [1] Jatuporn Chinrungrueng, Udomporn Sunantachaikul, Satien Triamlumlerd Smart Parking: an Application of optical Wireless Sensor Network Proceedings of the 2007 International Symposium on Applications and the Internet Workshops (SAINTW'07)
- [2] Ho Gi Jung, Member, IEEE, Young Ha Cho, Pal Joo Yoon, and Jaihie Kim Scanning Laser Radar-Based Target Position Designation for Parking Aid System
- [3] S. V. Srikanth, Pramod P. J, Dileep K. P, Tapas S, Mahesh U. Patil, Sarat Chandra Babu Design and Implementation of a prototype Smart PARKing (SPARK) System using Wireless Sensor Networks 2009 International Conference on Advanced Information Networking and Applications Workshops
- [4] Alberto López Esteban, Diseño Y desarrollo de un módulo de conexión a canopen de un sensor comercial fuerza/par, Pag 47. [Online] Disponible http://e-archivo.uc3m.es/bitstream/handle/10016/11854/PFC_ Alberto_Lopez_Esteban.pdf?sequence=2
- [5] Datasheet MAX232 [Online] Disponible http://www.ti.com/lit/ds/ symlink/max232.pdf



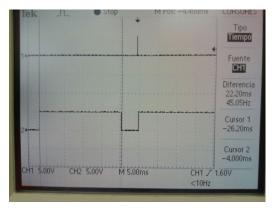
(a) Montaje sistema de luces y sensado plaza libre.



(b) Montaje sistema de luces y sensado plaza libre.



(c) Voltaje en el pin ECHO del HC-SR04 para el valor mínimo de 0.3m.



(d) Voltaje en el pin ECHO del HC-SR04 para el valor máximo de 3.8m.

Fig. 6. Resultados obtenidos para el modulo de luces y sensado.

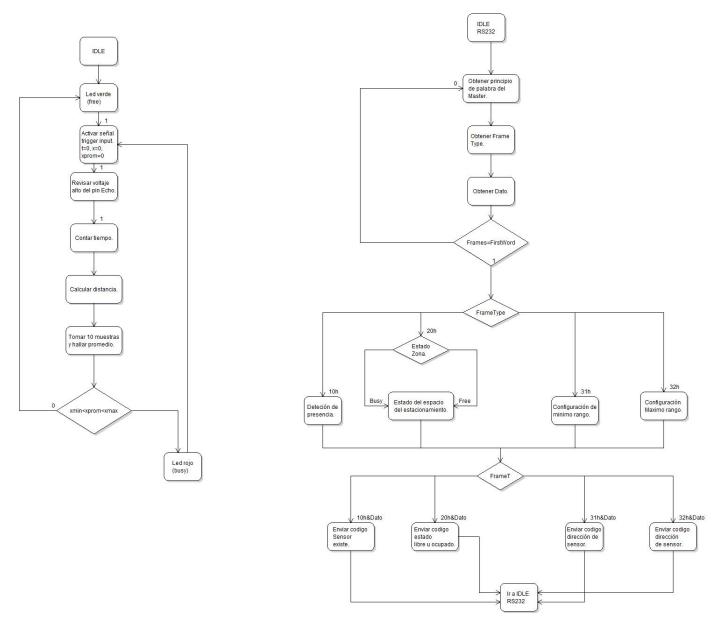


Fig. 7. Diagrama de flujo sensor de aparcamiento por ultrasonido con conectividad RS232.