

1 Ejercicio 8

Se propuso el diseño de un circuito que permita medir distancias utilizando un sensor ultrasónico de distancia (HC-SR04).

El circuito debe cumplir con los siguientes requerimientos:

- *trigger*: pin de entrada que dispara la medición en el flanco positivo.
- *trigger enable*: pin de entrada que habilita la señal de disparo.
- *meas*: 8 bits de salida que indican el tiempo medido en unidades de 100 μ s.
- *meas ready*: pin de salida que indica que la medición ha finalizado.

1.1 Sensor HC-SR04

El sensor de distancia HC-SR04¹ es un sensor de distancia ultrasónico. Posee cuatro terminales: dos de alimentación (V_{CC} y gnd), el *trigger* y *echo*.

El terminal de *trigger* acciona la medición. Para ello se debe cambiar el estado de este terminal a *high* por más de 10 μ s y, de esta manera, el sensor comienza a medir. Luego por el pin de *echo* se devuelve un pulso de ancho T μ s. La duración de este pulso es proporcional a la distancia medida: Distancia = $\frac{1\mu s}{58} [cm]$.

El rango de medición del sensor es de 2cm a 4 metros. Por ende, el ancho del pulso devuelto por *echo* es de entre 116 μ s y 23200 μ s.



Figura 1: Sensor de distancia

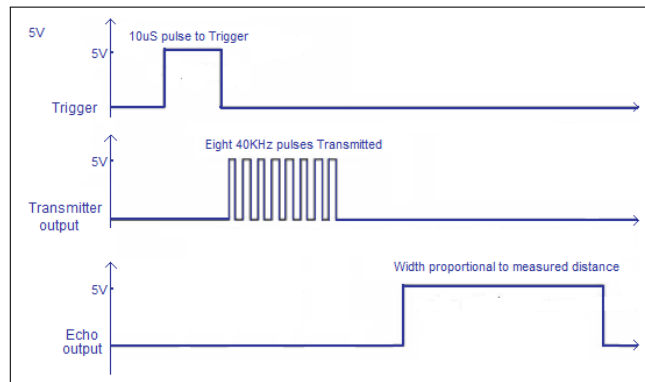


Figura 2: Diagrama temporal

¹Datasheet del sensor: <https://www.mouser.com/ds/2/813/HCSR04-1022824.pdf> (consultado: 13/10/18).

1.2 Circuito Implementado

Se modularizó el circuito de la siguiente manera:

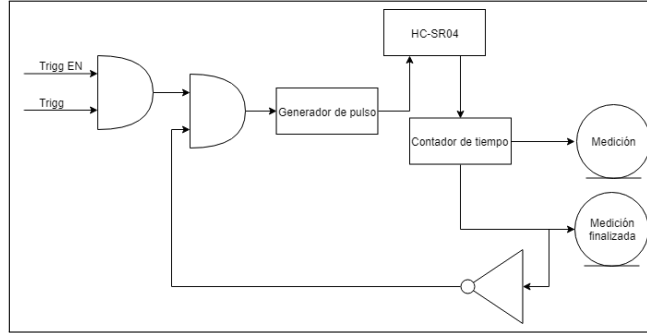


Figura 3: Diagrama de bloques

Los dos módulos principales, tal como se muestra en la imagen 3, son el generador de pulso y el contador de tiempo.

El generador de pulso, cuando recibe un flanco positivo, genera un pulso mayor a $10\mu s$ para que el sensor comience a medir.

El contador de tiempo, por otra parte, mide el ancho del pulso devuelto por *echo* y lo devuelve en 8 bits. También se encarga de indicar que la medición finalizó.

1.2.1 Generador de pulso

El generador de pulso se implementó con un LM555. El circuito utilizado fue el de la figura 4. Este circuito genera un pulso de ancho $t = R_a C \cdot 1.1s$. Para generar el disparo se debe generar un pulso menor que el configurado.

Como el sensor requiere un ancho de pulso mayor de $10\mu s$, se eligieron los componentes para un pulso de $50\mu s$, por ende el capacitor $C = 10nF$ y $R_a = 5k\Omega$

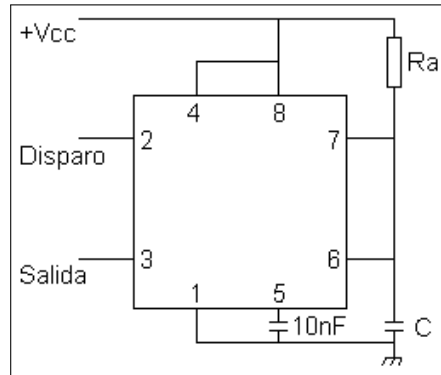


Figura 4: Circuito del generador de pulsos

Para evitar que al generador de pulso le ingrese una señal mayor a $50\mu s$ se le colocó a la entrada (disparo) un RC con tiempo característico de $10\mu s$.

1.2.2 Contador de tiempo

Para medir cuánto tiempo el pulso de *echo* está en *high*, se utilizó un contador de 8 bits y una señal cuadrada de 50% de *duty cycle* como *clock*.

El contador utilizado cuenta flancos positivos de *clock*. Por ende, para contar cuánto tiempo la señal de *echo* estuvo en *high*, se conectó la entrada de clock del contador, a la salida de una *and*. Las entradas de la *and* son la señal de *echo* y una señal cuadrada de $10kHz$. De esta manera, cada vez que ocurre un flanco de *clock* y la señal *echo* esta en *high* el contador incrementa en una unidad. Debido a que la frecuencia de la señal cuadrada es de $10kHz$, cada unidad en el contador representa $100\mu s$.

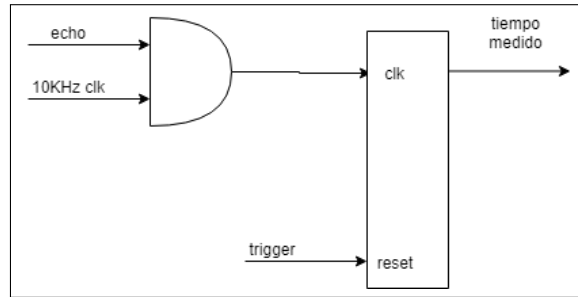


Figura 5: Contador de tiempo

En cuanto al reset del contador, se lo conectó al *trigger* del sensor. De esta manera, cada vez que se dispara una medición el contador vuelve a cero.

1.2.3 Meas ready

En cuanto a la salida de *meas ready*, bastó con negar la señal de *echo*. Mientras no haya medición, la señal de *echo* se mantiene en *low* y por ende *meas ready* se mantiene en *high*. Cuando *echo* está en *high*, en cambio, quiere decir que se está midiendo, y *meas ready* se encuentra entonces en *low*.

1.2.4 Prototipo

Previo al diseño final del circuito, se decidió construir un prototipo del mismo en una *protoboard*, para así probar el correcto funcionamiento de cada etapa y del conjunto.

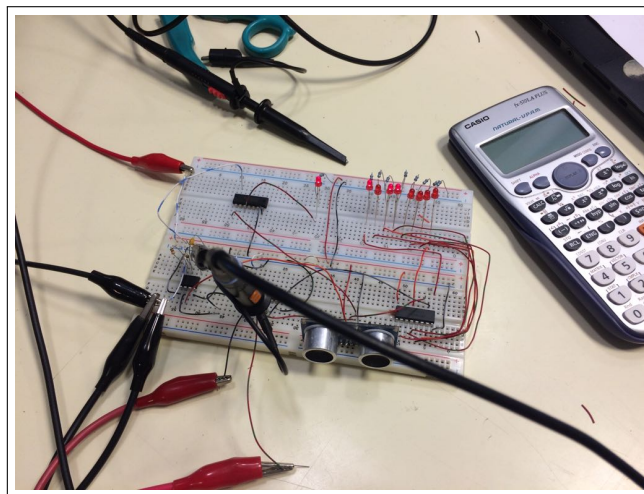


Figura 6: Prototipo del circuito

El prototipo funcionó correctamente, tal como se puede apreciar en [este video](#). Se procedió, pues, al armado de la placa.

1.2.5 Circuito

El circuito final construido posee las siguientes características:

- Entrada de *trigger*, *trigger enable* y *clock*. El *clock* al que se lo debe conectar es de $10kHz$ (señal cuadrada, 5V de amplitud y 50% de *duty cycle*).
- Salida 8 bits que indican el tiempo medido, y 1 bit que indica que la medición finalizó.

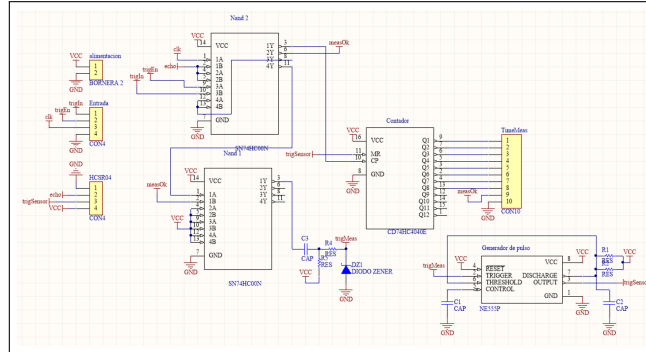


Figura 7: Esquemático

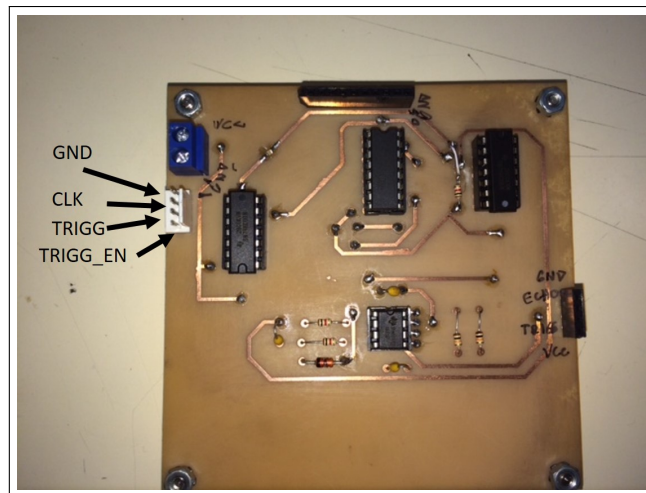


Figura 8: PCB

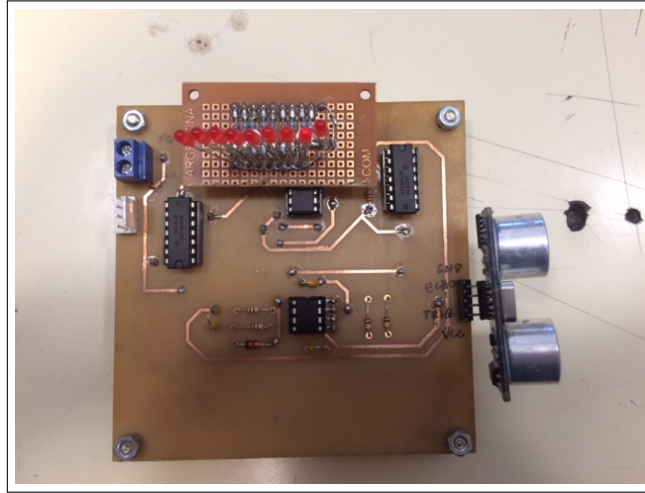


Figura 9: PCB con sensor y led para indicar la salida

1.2.6 Simulación

Se simuló en Iverilog el circuito, y con GTKWave se visualizó su comportamiento en el tiempo.

Los módulos utilizados en la simulación son los mismos que los anteriormente descritos, es decir hay un módulo encargado de generar un pulso y otro de contar el tiempo.

La figura 10 es la simulación temporal del circuito. "meas" es el resultado de la medición, "sEcho" es el pulso devuelto por el sensor, "sTrigger" es el pulso que el circuito le envía al sensor para comenzar la medición, "trigger" es el pulso que recibe el circuito para comenzar a medir, y "meas ready" indica que la medición finalizó.

Tal como se observa en la imagen, primero se dispara el *trigger*, posteriormente el circuito genera en sTrigger un pulso de ancho $50\mu s$. Después sEcho pasa a *high* (comienza a medir el sensor) y *meas ready* pasa a *low*. El contador comienza a contar cuántos pulsos de *clock* la señal de echo estuvo en *high*. Cuando sEcho pasa a *low* (la medición finalizó) y *meas ready* pasa a *high*. Como el ancho del echo es de $500\mu s$, el contador llegó a 5.

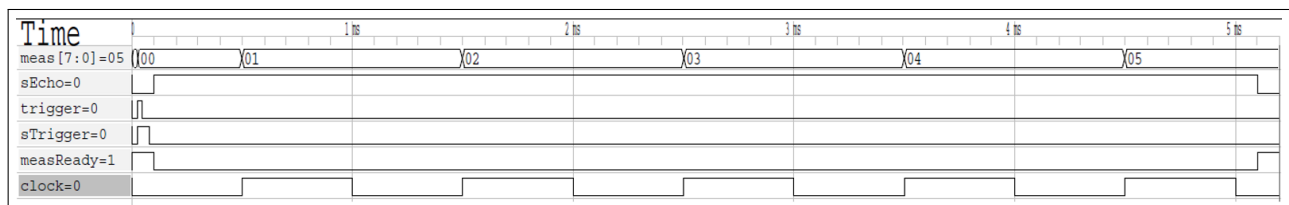


Figura 10: Simulación