

# Ecosonda Android

***Autores:*** Polvorín, Gonzalo Adrián  
Valle, Víctor Manuel

***Directores:*** Ing. Gemin, Walter  
Ing. Rivera Raul

***Carrera:*** Ingeniería Electrónica

***Año:*** 2016



RINFI se desarrolla en forma conjunta entre el INTEMA y la Biblioteca de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata.

Tiene como objetivo recopilar, organizar, gestionar, difundir y preservar documentos digitales en Ingeniería, Ciencia y Tecnología de Materiales y Ciencias Afines.

A través del Acceso Abierto, se pretende aumentar la visibilidad y el impacto de los resultados de la investigación, asumiendo las políticas y cumpliendo con los protocolos y estándares internacionales para la interoperabilidad entre repositorios



Esta obra está bajo una [Licencia Creative Commons Atribución-  
NoComercial-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/).

# Índice

Resumen .....	2
1. Introducción.....	2
1.1. Origen del proyecto y problemáticas encontradas .....	5
1.2. Especificaciones preliminares .....	6
2. Anteproyecto .....	6
2.1. Bloque de Control .....	10
2.2. Interfaz Visual.....	10
2.3. Transductor y Amplificador .....	10
3. Proyecto .....	11
3.1. Transmisor/receptor .....	11
3.2. Interconexión Control-Tx/Rx .....	13
3.3. Control.....	14
3.4. Funcionamiento .....	15
3.5. Módulo bluetooth .....	18
3.6. Interfaz visual .....	18
3.7. Gráficos .....	22
3.7.1. Tipo 1 .....	22
3.7.2. Tipo 2 .....	24
3.8. Alimentación .....	25
4. Manual de operación .....	26
4.1. Encender/Detener.....	29
4.2. Tamaño .....	30
4.3. Ganancia.....	30
4.4. Filtro .....	31
4.5. Frecuencia de repetición .....	31
4.6. Ancho de pulso de transmisión .....	32
4.7. Profundidad máxima .....	33
4.8. Tipos de gráficos.....	33
4.9. Persistencia .....	35
4.10. Guardar datos .....	36
5. Mejoras Futuras .....	37
6. Conclusiones .....	38
7. Bibliografía .....	39
Apéndice A. Código fuente del microcontrolador PIC.	
Apéndice B. Código fuente de la Aplicación Android.	
Apéndice C. Hojas de Datos	

## Resumen

Este proyecto consiste en el diseño y construcción de una ecosonda con interfaz Android, orientada al uso en pesca deportiva, a profundidades de no más de 50 metros. El dispositivo cuenta con un transductor ultrasónico, que es la única parte del desarrollo que entra en contacto con el agua. El dispositivo se comunica, mediante una conexión bluetooth de manera totalmente inalámbrica, con cualquier dispositivo que posea un sistema operativo Android (teléfonos celulares, tablets, etc.), para lo que se desarrolló además una aplicación compatible con cualquiera de los dispositivos mencionados.

La parte electrónica del desarrollo se basó en un microcontrolador PIC18F2550 de la firma Microchip, el cual se comunica con un integrado especialmente desarrollado para el procesamiento de señales ultrasónicas, el PW0268, de la firma Pro-Wave, y con el dispositivo Android que se esté utilizando.

El objetivo final de este proyecto, es brindar al pescador deportivo que utiliza embarcaciones de poco porte, una solución simple, con una interfaz visual amigable, de manejo sencillo e intuitivo.

## 1. Introducción

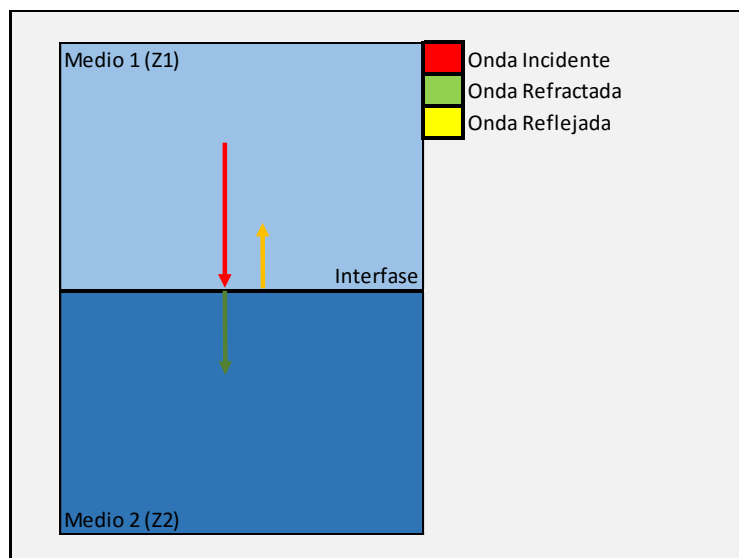
En la pesca deportiva, se utilizan diversos tipos de implementos, y en la pesca deportiva que se realiza en embarcaciones, la electrónica es una herramienta que se utiliza desde hace ya muchos años. Es común encontrar en estas embarcaciones, distintas formas de localizar cardúmenes de peces, entre ellas la ecosonda.

Una ecosonda, es un instrumento utilizado para la emisión y detección acústica. Su función principal es medir la distancia vertical que existe, entre el casco de una embarcación y el fondo marino, identificando cualquier obstáculo que se encuentre entre estos, como por ejemplo, cardúmenes de peces.

El principio de funcionamiento de la ecosonda se conoce como pulso-eco. Este consiste en la emisión de un pulso, en el medio que se desea analizar, y la posterior recepción del eco para su estudio.

La frecuencia a utilizar para el pulso de emisión en un equipo de ultrasonido, es su característica principal. Una frecuencia elevada permite obtener gran detalle del medio bajo estudio, pero no ofrece gran penetración (alcance) en éste, a diferencia de una onda de baja frecuencia que permitirá cubrir grandes distancias, pero obteniendo un detalle (definición de la imagen) mucho menor. Esto se debe a que la frecuencia es inversamente proporcional a la longitud de onda de la señal, y cuanto más chica resulta ésta, la onda incidente es capaz de interactuar con partículas de menor tamaño (pequeñas partículas en suspensión, por ejemplo) y éstas provocan que la onda pierda potencia, disipándose y dispersándose con mayor rapidez.

Un concepto importante que debe tenerse en cuenta es el de la impedancia acústica. Esta recibe su nombre por comparación con la impedancia eléctrica que se define en circuitos eléctricos. Conceptualmente puede definirse como el cociente entre la presión sonora y la velocidad de las partículas del material. El eco se produce cuando la onda incidente enfrenta un cambio en la impedancia acústica del medio. Suponiendo que el cambio en la impedancia se debe a un cambio de medio, una fracción de la onda será reflejada en la interfaz, produciendo un eco, y otra refractada al nuevo medio. Esta situación está representada en la Figura 01.



**Figura 01. Comportamiento de las ondas en una interfaz entre medios.**

Cuanto mayor sea la diferencia entre las impedancias características de los medios, mayor será el porcentaje de la onda reflejada, y menor el de la onda refractada.

El escenario anteriormente descrito, es una idealización del concepto, basandose en la premisa de que las ondas viajan por medios completamente homogéneos y sin pérdidas en su

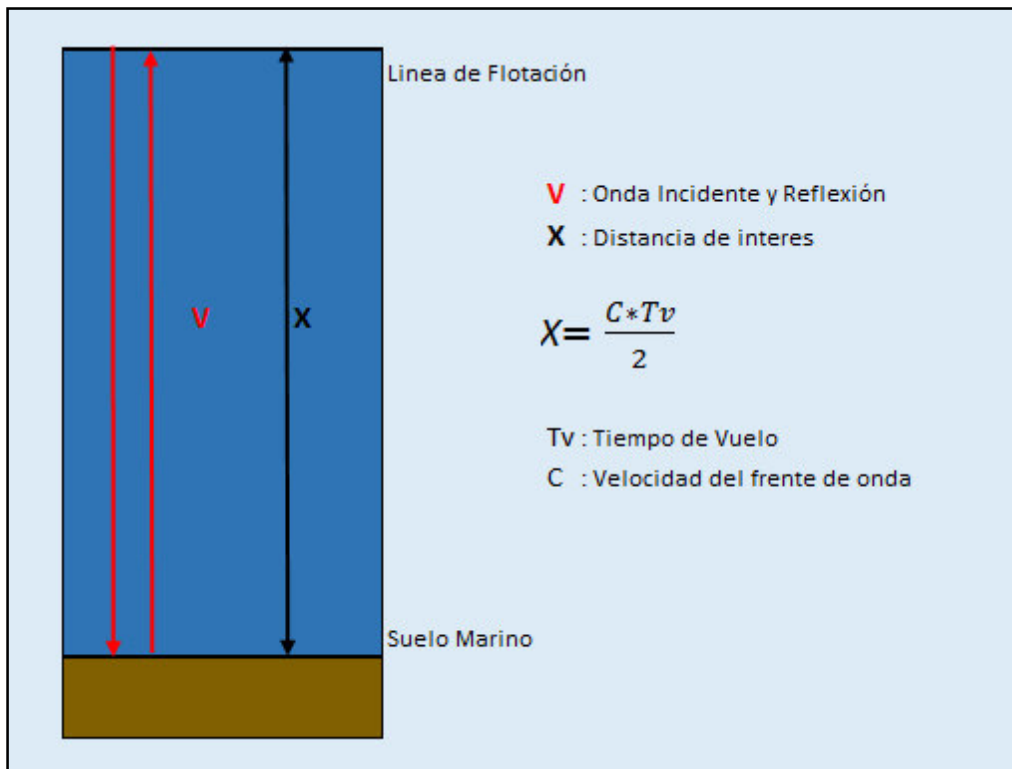
trayecto. Esto en la práctica no es necesariamente cierto, más aún en el medio de interés en este proyecto, el agua marina.

El mar, es en sí mismo un sistema heterogéneo. Características tales como la salinidad del agua y las partículas en suspensión entre otras, generan la formación de múltiples interfaces lo que a su vez generará múltiples ondas reflejadas. Incluso las diferencias de temperatura en el agua, que si bien por definición no representan una interfaz ni un cambio de medio, provocan que la impedancia del medio cambie, lo que para esta aplicación, se traduce en menor penetración o pérdida de potencia en la onda incidente.

En este tipo de aplicaciones, son dos las reflexiones de las cuales es necesario extraer información, las producidas por los peces, y las producidas por el fondo marino. Este último, mucho más compacto y duro que el agua en sí, será generalmente quien produzca la mayor reflexión. De esta se obtendrá la dimensión de la columna de agua sobre la que se encuentra la embarcación. Los peces por su lado, generaran una reflexión de menor intensidad, pero igualmente detectable para el transductor de la ecosonda.

Es importante mencionar que en general, no es el cuerpo del pez, quien genera la reflexión de interés, si no su vejiga natatoria. Este es un órgano que poseen muchos peces y les permite controlar su flotabilidad variando la cantidad de gas que ésta contiene. Este gas representa un cambio grande en la impedancia acústica del medio, lo que genera una reflexión de considerable intensidad.

Una vez recibido el eco ultrasónico, se analiza su intensidad, lo que da una idea de la dimensión y/o composición del obstáculo, y el tiempo de vuelo de la onda. Suponiendo que lo que se desea conocer es la posición del suelo marino, respecto de la línea de flotación de la embarcación, se mide el tiempo entre que se emite el pulso de transmisión y se recibe el eco de la reflexión. Considerando que la onda recorrió el camino de ida y vuelta en el agua y siendo conocida la velocidad con la que se mueve el frente de onda en ésta, puede calcularse la distancia, utilizando la expresión mostrada en la Figura 02.



**Figura 02. Cálculo del Tiempo de Vuelo**

### 1.1. Origen del proyecto y problemáticas encontradas

El principal obstáculo para incorporar la electrónica a la pesca, casi siempre es económico. El valor de una ecosonda comercial, supera al valor de los instrumentos convencionales utilizados para llevar adelante la actividad, y es por este motivo que generalmente no se utilizan ecosondas en embarcaciones de poco porte o embarcaciones que se dedican exclusivamente a la pesca deportiva.

Repasando las partes principales de una ecosonda convencional, encontramos, al transductor, única parte del instrumento que entra en contacto con el agua, al cuerpo del instrumento, donde se aloja la circuitería, el display y la botonera del mismo y a la batería, necesaria para proveer de energía a todo el equipo. Todas éstas partes, son sensibles a la corrosión, el cual es el principal problema a enfrentar, sobre todo si el instrumento se utiliza en embarcaciones abiertas, expuestas constantemente a salpicaduras, como pueden ser gomones o kayaks. Generalmente el deterioro es observado en las conexiones entre el cable conductor y el transductor o la propia ecosonda, las conexiones de la ecosonda con la batería, y claro está, el cuerpo mismo del dispositivo, si este llegara a mojarse o directamente a sumergirse en el agua. Surgió entonces la idea de

desarrollar un instrumento, minimizando las posibilidades de que este se deteriore producto de la corrosión, siendo una de las posibilidades, eliminar o minimizar las conexiones de éste.

Durante el desarrollo y construcción del dispositivo, surgieron diversas problemáticas, sobre todo a la hora de conseguir los componentes electrónicos necesarios para llevar adelante el proyecto. Estas serán tratadas en detalle más adelante.

## 1.2. Especificaciones preliminares

El proyecto consta del diseño y construcción de una ecosonda para embarcaciones de pesca deportiva.

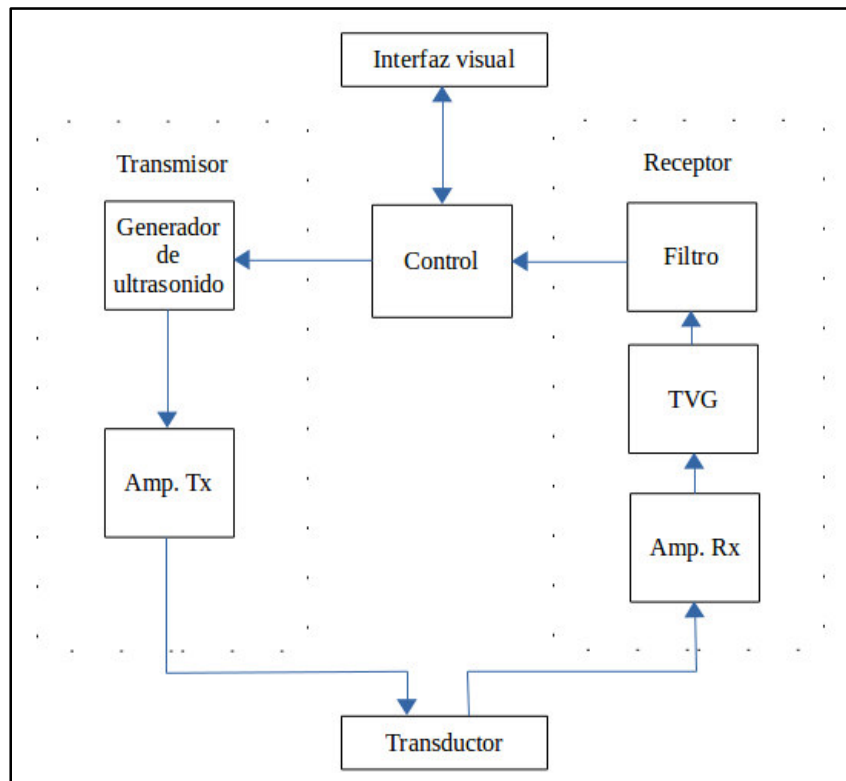
El sistema consta del hardware para la generación y adquisición de las señales de ecosonda y el software de procesamiento y graficación en entorno Android. El objetivo principal es la medición de profundidad y la detección de presencia de peces bajo la embarcación.

Para el desarrollo del proyecto se utilizará un transductor comercial, el hardware comprenderá los diseños de las etapas, analógica de transmisión y adquisición de datos, la etapa digital en la que se realizará un pre-procesamiento de los datos obtenidos y la comunicación inalámbrica con un dispositivo móvil. El software abarca la programación (firmware) del microcontrolador de la interfase y la aplicación en el dispositivo móvil Android, para presentar los datos e instrumentar, de manera virtual, los controles necesarios para operar el sistema.

## 2. Anteproyecto

Una vez analizada la problemática y fijados los objetivos principales del proyecto, se inició a la tarea de decidir la forma de implementación del mismo. En principio, se planteó el diseño mostrado en la Figura 03.



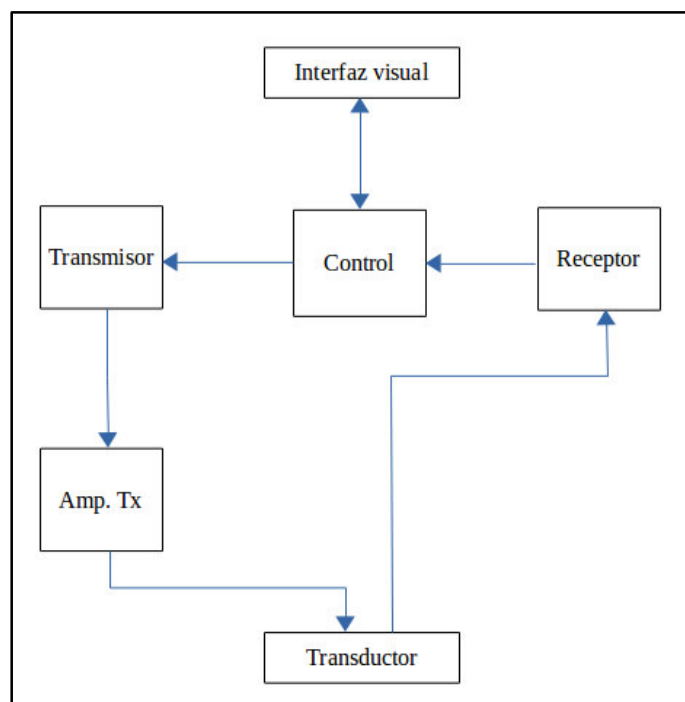


**Figura 03. Diagrama en bloques.**

- Interfaz visual: es la comunicación entre el usuario y la ecosonda. Consiste en la pantalla donde se muestran los datos obtenidos y los controles necesarios para operar la ecosonda;
- Control: es el bloque encargado de generar todas las señales de control y de procesar todos los datos recibidos antes de enviarlos a la interfaz visual;
- Generador de ultrasonido: se encarga de generar una señal senoidal de la frecuencia necesaria (200KHz);
- Amplificador de transmisión (Amp. Tx): se encarga de amplificar y acondicionar la señal proveniente del generador de ultrasonido para excitar al transductor;
- Amplificador de recepción (Amp. Rx): se encarga de amplificar los ecos recibidos para luego ser procesados;
- Ganancia variable en el tiempo (TVG): se encarga de compensar la atenuación producida por la distancia recorrida por la onda, haciendo que la señal recibida dependa sólo de las propiedades acústicas del blanco y no de la distancia al mismo;
- Filtro: se utiliza para eliminar señales interferentes, que pueden producir errores en las lecturas, asegurándose que solo lleguen a procesarse aquellas señales que corresponden a los ecos de la señal transmitida;

- Transductor: es el elemento en contacto con el agua. Cuando la ecosonda está en modo de transmisión, transforma una señal eléctrica en una onda de presión que dirige hacia el fondo del mar. Cuando la ecosonda está en modo de recepción, hace el trabajo inverso, recibe las ondas de presión provenientes de los ecos y las transforma en señales eléctricas que envía al receptor.

En un principio se analizó la posibilidad de diseñar y/o construir cada uno de los bloques de la Figura 03 con el fin de realizar un producto completamente original y propio. Esta idea se cambió, debido a que un proyecto de tal magnitud, por un lado, se extendería demasiado en el tiempo, y por otro lado, diseñar y construir el transductor sería demasiado complejo escapando a nuestros conocimientos y competencias. Por este motivo se centraron los esfuerzos en el diseño de dos bloques claves, como son: el control y la interfaz visual. Se realizó una investigación con el fin de encontrar circuitos y desarrollos de terceros que pudieran simplificar el resto de los bloques y transformar el diagrama de la Figura 03 en el de la Figura 04.



**Figura 04. Diagrama en bloques simplificado A.**

Se encontró un circuito integrado del fabricante Pro-Wave Electronics Corp. (Taiwan), Ultrasonic Sonar Ranging IC – PW0268, el cual está diseñado específicamente para la transmisión y recepción y pre-procesamiento de ultrasonidos.

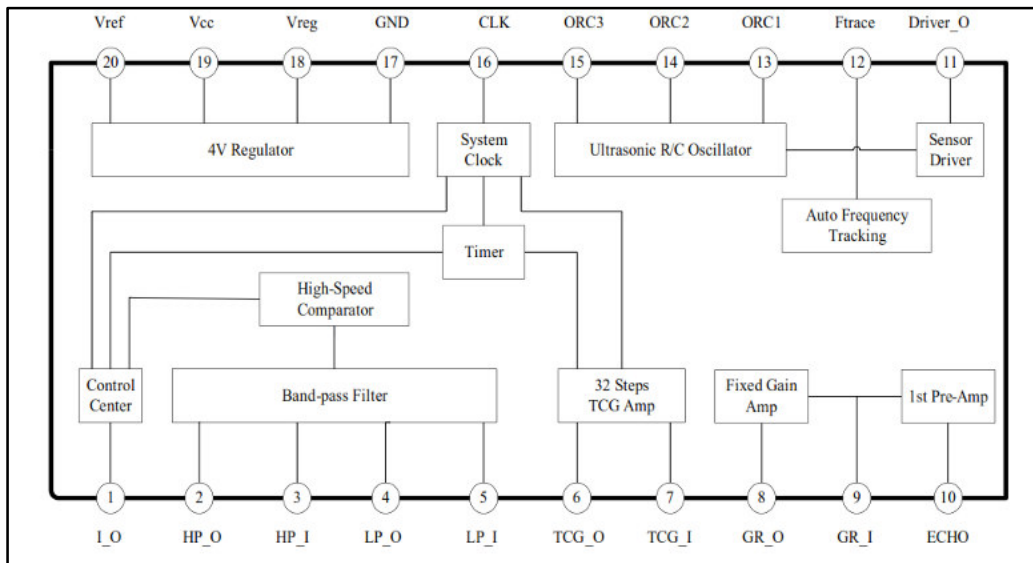


Figura 05. Diagrama en bloques del IC PW0268

Con la elección de este componente el diagrama en bloques se simplifica aún más llegando así al diagrama de la Figura 06.

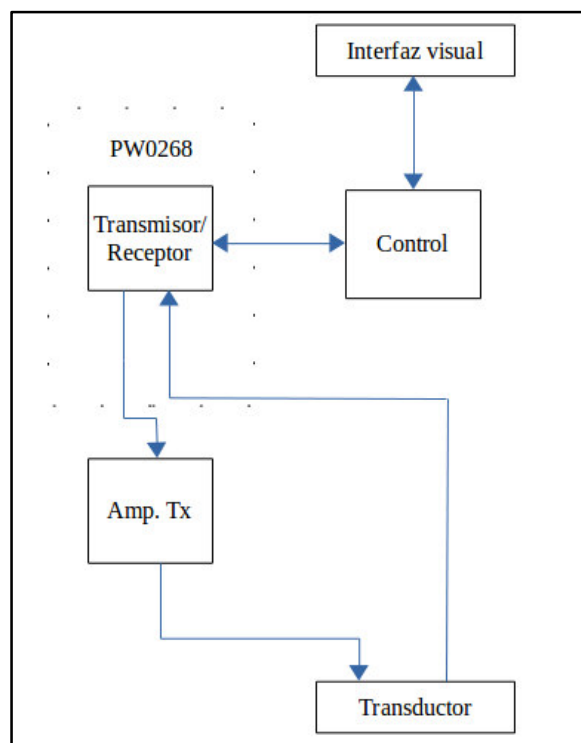


Figura 06. Diagrama en bloques simplificado B.

## 2.1. Bloque de Control

Para el bloque de control se decidió utilizar un microcontrolador PIC18F2550. Este se encuentra disponible comercialmente en nuestro país y además cumple con los requisitos necesarios para el desarrollo, dejando abierta de esta manera la posibilidad de ampliar las funcionalidades del dispositivo sin la necesidad de realizar grandes cambios a nivel de hardware.

## 2.2. Interfaz Visual

Se optó por utilizar un dispositivo Android (Smartphone, Tablet u otro). Esto, además de ser original, representa un ahorro importante si se decide producir el dispositivo de forma comercial, ya que el usuario puede operar e interactuar con el dispositivo directamente desde, por ejemplo, su teléfono celular, lo que hace que no sea necesario incluir display ni botones de control.

## 2.3. Transductor y Amplificador

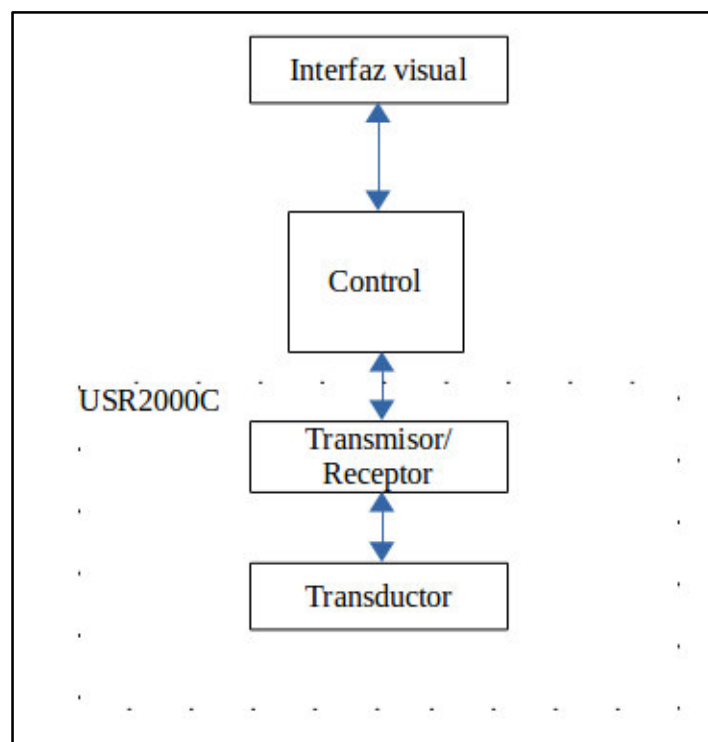
El último paso consistió en la elección del transductor y el amplificador para alimentarlo. En principio, se intentó adquirir en el mercado nacional los componentes recomendados por el fabricante del IC PW0268, esto es, el transductor (200LM450) y el bloque amplificador. Debido a las políticas de importaciones vigentes, esto no fue posible. Fue necesario entonces buscar reemplazos a estos componentes en el mercado local, pero los componentes disponibles en el mercado argentino son repuestos de ecosondas comerciales, de los cuales se desconocen sus especificaciones y modos de funcionamiento, lo que dificulta en gran medida la construcción de un amplificador compatible con el desarrollo. La única posibilidad de adquirir los componentes electrónicos necesarios, junto con sus especificaciones de diseño, fue contactar con fabricantes internacionales que los producen, como FURUNO o RESON, pero los costos asociados a estos resultaron muy por encima del presupuesto disponible. Se decidió entonces adquirir una ecosonda comercial marca GARMIN, con el objetivo de utilizar su transductor, y su bloque amplificador.

Se construyó el circuito transmisor/receptor basado en el IC PW0268, según las especificaciones dadas por el fabricante y se inyectó la señal de control directamente al bloque amplificador de la ecosonda GARMIN. Esta solución no resultó satisfactoria, debido a que los circuitos y bloques funcionales de cada una de las placas no resultaron compatibles.

Se decidió entonces contactar directamente con el fabricante del IC PW0268, para plantear la posibilidad de adquirir directamente de ellos el transductor y el bloque amplificador. El fabricante respondió que era posible que nos envíen el transductor solicitado, pero que no contaba con el bloque amplificador necesario. No obstante, ofreció la posibilidad de fabricar un módulo completo (USR2000C) incluyendo el transductor y el bloque amplificador integrados en él mismo, poniendo a disposición, además, las hojas técnicas de éste. Se analizaron las especificaciones de dicho módulo y al encontrarlas compatibles con las necesidades del proyecto se decidió adquirirlo. Finalmente, el diagrama en bloques definitivo del desarrollo resultó como se lo muestra en la Figura 07.

### 3. Proyecto

El diagrama en bloques del dispositivo se muestra de la Figura 07, el siguiente paso fue diseñar y/o construir cada uno de los bloques allí mostrados.



**Figura 07. Diagrama en bloques**

#### 3.1. Transmisor/receptor

El bloque transmisor/receptor está integrado dentro del módulo USR2000C. El circuito del mismo

se muestra en la Figura 08.

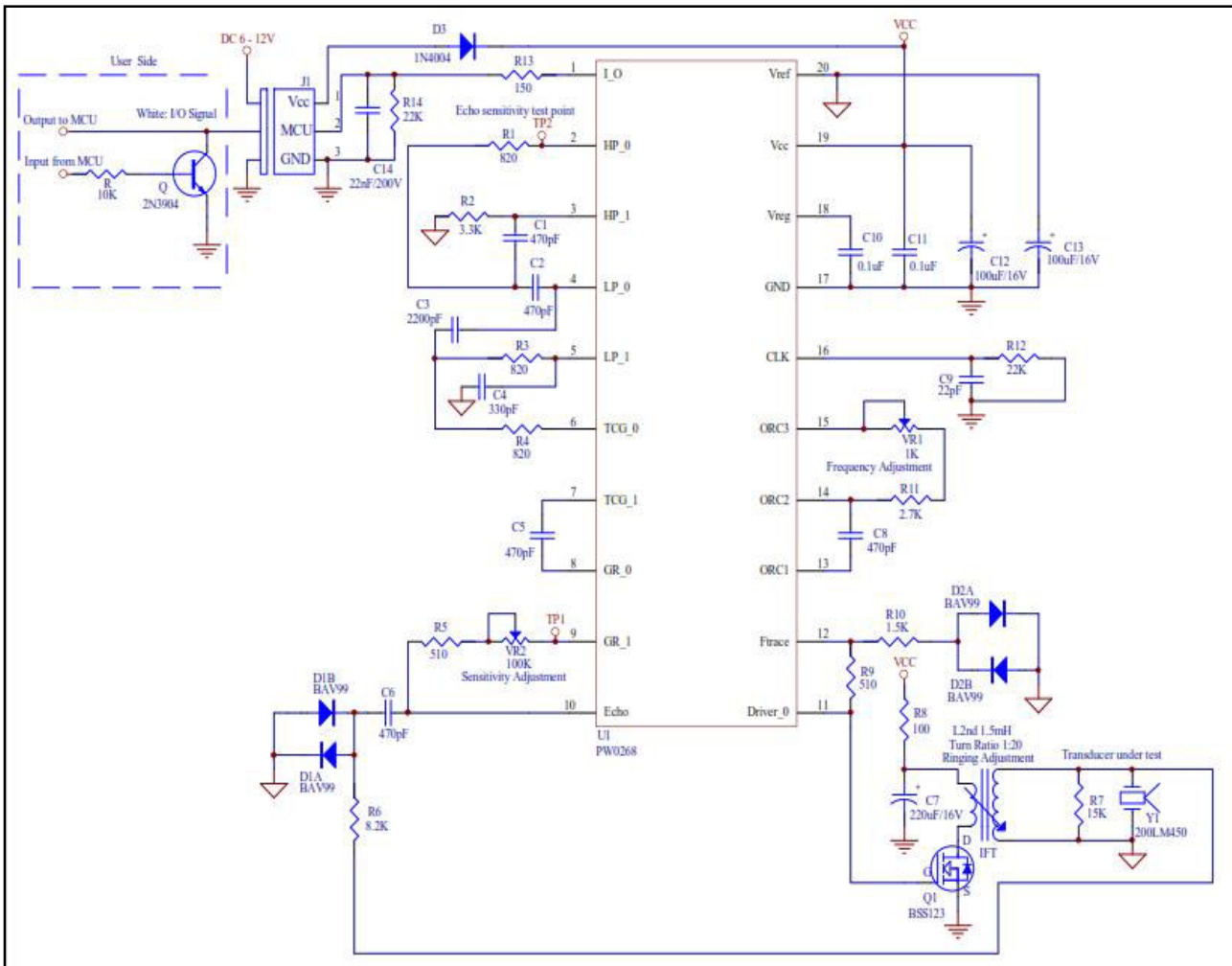


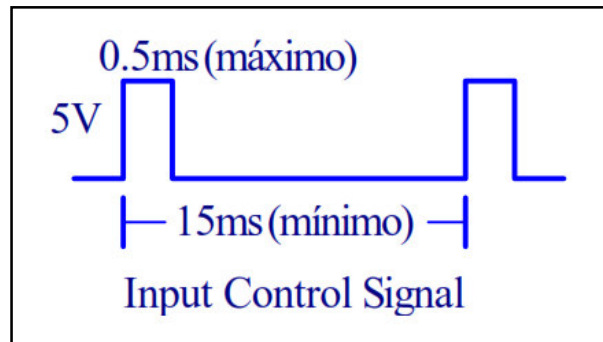
Figura 08. Esquemático módulo USR2000C.

El circuito funciona de la siguiente forma: se envía externamente un pulso de transmisión activo bajo al pin 1 el IC (que opera como puerto de entrada salida). Este genera un tono de 200Khz en el pin 11, el cual es enviado al transductor, previa amplificación. Después del pulso de transmisión el pin 1 volverá a presentar un pulso activo bajo, esta vez generado internamente por el mismo módulo, cuando una señal de eco sea detectada. El ancho de este pulso será proporcional a la potencia de la señal de eco recibida.

El fabricante del USR2000C recomienda realizar la conexión entre dicho módulo y un microcontrolador genérico a través del transistor 2N3904, que opera como llave como se muestra en la Figura 08. Un pulso activo alto generado por el microcontrolador, enciende el transistor mandando a masa el colector del mismo, generando el pulso en bajo que necesita el pin 1 del

transmisor/receptor. De esta manera, los pulsos de recepción activo bajos pueden ser leídos directamente en la entrada del microcontrolador.

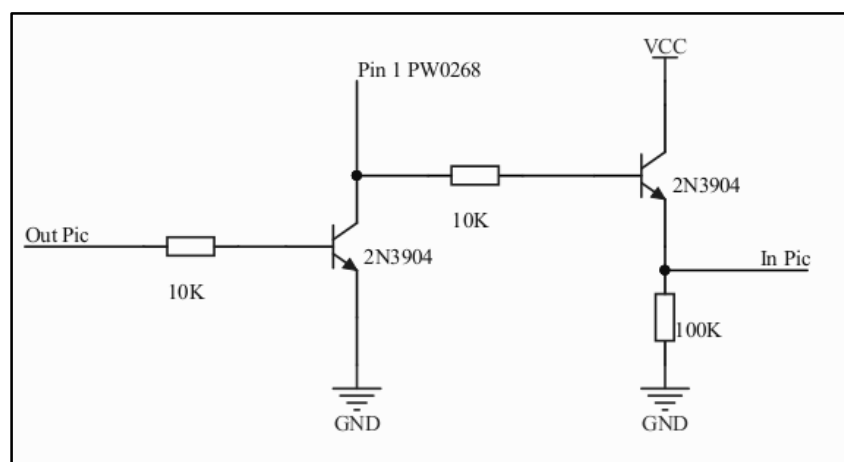
La señal de control generada por el microcontrolador tiene ciertas limitaciones impuestas por propio el módulo transmisor/receptor. Para el circuito utilizado en este diseño, los límites de estos valores resultan ser los mostrados en la Figura 09.



**Figura 09. Señal de Control.**

### 3.2. Interconexión Control-Tx/Rx

El módulo USR2000C utiliza una tensión de alimentación mayor a la que utiliza el microcontrolador PIC18F2550, por lo que fue necesario realizar una adaptación, asilando la entrada del PIC de la tensión presente en el USR2000C. El circuito que se utilizó, se muestra en la Figura 10.

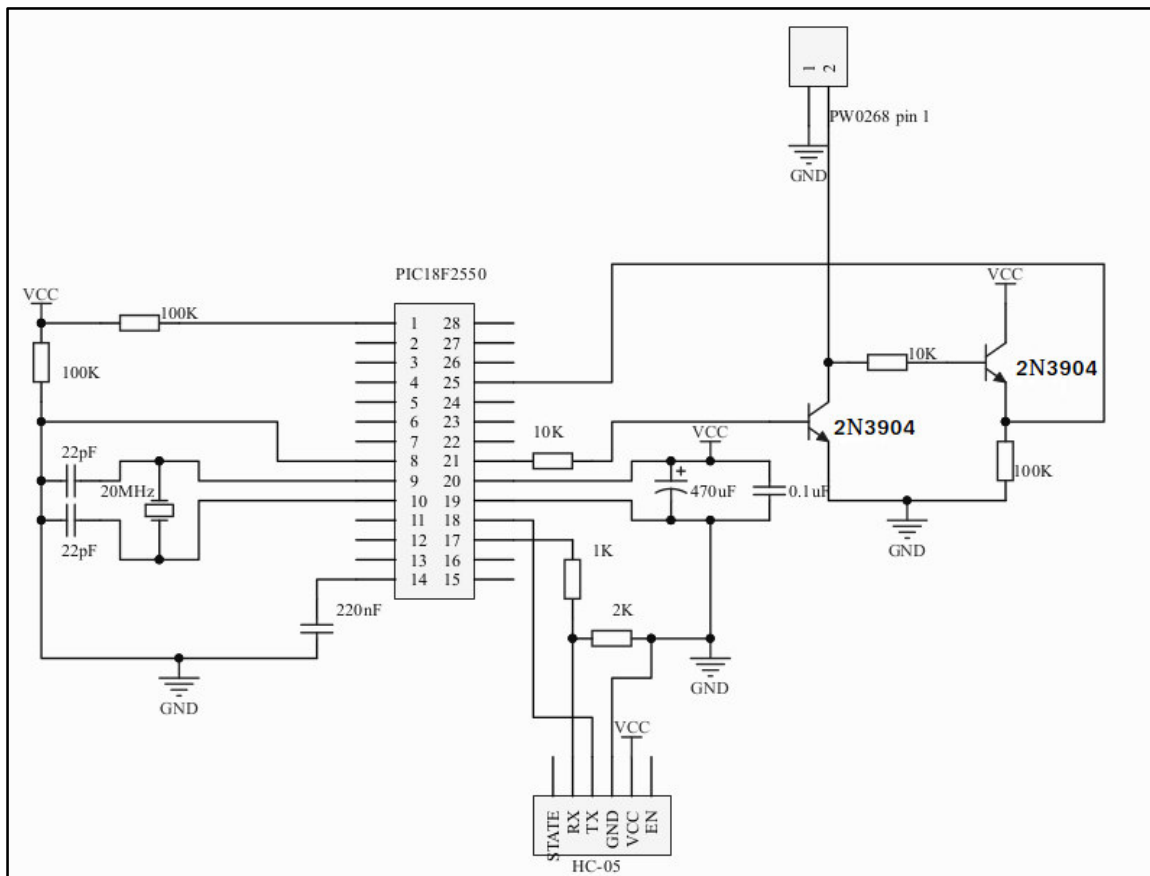


**Figura 10. Interconexión PIC-USR2000C**

### 3.3. Control

Este bloque se diseñó con un microcontrolador PIC18F2550, que trabaja por medio de comandos recibiendo las órdenes dadas por el usuario a través de la interfaz visual Android. Se encarga de generar la señal de control, de recibir y procesar las señales de ecos recibidos, así como de enviar la información sobre esos ecos a la interfaz visual.

En la Figura 11 se muestra el circuito utilizado para el funcionamiento del microcontrolador, el cual cuenta con una única señal digital de entrada y una única señal digital de salida, un cristal de 20MHz que es el encargado de generar las señales de reloj del dispositivo, la conexión al módulo bluetooth HC-05 mediante el puerto serie y distintos capacitores utilizados como filtros para la alimentación del microcontrolador.



**Figura 11. Circuito del bloque de control**

El microcontrolador está programado de manera que el usuario puede variar los parámetros de la señal de control, siempre dentro de los límites mostrados en la Figura 09, según su conveniencia. Dichos parámetros son el ancho del pulso de transmisión y el período de repetición



del mismo.

El ancho del pulso es importante, y es el parámetro que debe modificarse cuando se busca variar el alcance del dispositivo, debido a que cuanto más ancho es este pulso, mayor es la cantidad de energía que se inyecta al medio y mayor es la penetración que se logra en éste. Se debe tener en cuenta, además, que al utilizar un único transductor para la transmisión y recepción de las señales, estos procesos no pueden realizarse en forma paralela, por este motivo, el dispositivo será incapaz de detectar un eco, mientras se encuentra transmitiendo.

En cuanto al período de repetición, debe garantizarse que el mismo no sea menor al tiempo que demoraría en llegar al dispositivo una señal de eco producida a la máxima profundidad posible. Es decir, si se espera por ejemplo, recibir una señal de un objeto ubicado a una profundidad de 10 metros, el período de repetición debe ser tal que permita al transductor, detectar el eco, antes de volver a re-transmitir, evitando de esta manera errores en las mediciones de distancia.

Es necesario destacar, además, que ambos parámetros influyen en el consumo de energía del dispositivo, razón por la cual se programaron por defecto valores de dichos parámetros que intenten maximizar el tiempo de funcionamiento del dispositivo. Por defecto la señal de control se programó con un período de repetición de 1seg y un ancho de pulso de 0.1mseg.

El código fuente de la programación del microcontrolador PIC, puede consultarse en el Apéndice A.

### 3.4. Funcionamiento

Una vez encendido el dispositivo, el microcontrolador se encuentra en reposo a la espera de la llegada de un comando por medio del puerto serie, que le indique que el dispositivo debe comenzar a funcionar. Una vez recibida esta orden, el microcontrolador generará un pulso de transmisión de 0.1mseg que será enviado al módulo transmisor/receptor, iniciando al mismo tiempo un contador, con el objetivo de poder calcular la distancia al objeto que produzca una señal de eco. En este punto el sistema entrará en modo de recepción durante 1 segundo, superado ese tiempo volverá a generar un nuevo pulso de transmisión y reiniciará el contador. Cada vez que esto sucede, el microcontrolador enviará por medio del puerto serie, un aviso de que un nuevo pulso de transmisión ha sido enviado.

En el modo de recepción el sistema estará a la espera de la aparición de un flanco

descendente, que indicará el inicio de un pulso de recepción. Una vez detectado dicho flanco, se guarda en memoria el valor que posee en ese instante el contador iniciado en la transmisión. Luego de la detección del flanco descendente, el sistema esperará la aparición de un flanco ascendente que indicará el final del pulso de recepción. Al momento de recibir este flanco, se guarda en memoria nuevamente el valor del contador, para luego compararlo con el valor que tenía al momento del flanco descendente, permitiendo de esta manera calcular el ancho del pulso de recepción. Una vez que se tienen los valores del tiempo transcurrido entre la transmisión y la recepción y del ancho del pulso de recepción el sistema calcula, la distancia a la cual se encuentra el objeto que produjo el eco y a estima la potencia de dicho eco.

Para calcular la distancia a la cual se encuentra el objeto que produjo el eco, se mide el tiempo transcurrido desde el envío del pulso de transmisión hasta la recepción del eco (tiempo de vuelo), luego se divide este valor por 2, debido a que la onda recorre el camino dos veces y por último se lo multiplica por la velocidad de la onda de ultrasonido en el agua (1500m/s) para obtener así la distancia. Esta situación se muestra en la Figura 02.

Para calcular la potencia del eco recibido, se mide el ancho del pulso de recepción, ya que el mismo es directamente proporcional a la potencia del eco.

Una vez que se obtienen estos dos valores (la distancia en centímetros y la potencia como ancho de pulso en microsegundos) se envían a través del puerto serie, hacia un módulo bluetooth el cual los enviará de forma inalámbrica hacia la interfaz visual. Luego de enviar estos valores el sistema queda nuevamente a la espera de un nuevo pulso de recepción. En la Figura 12 se observa en detalle, el diagrama de flujo detallado de este proceso.

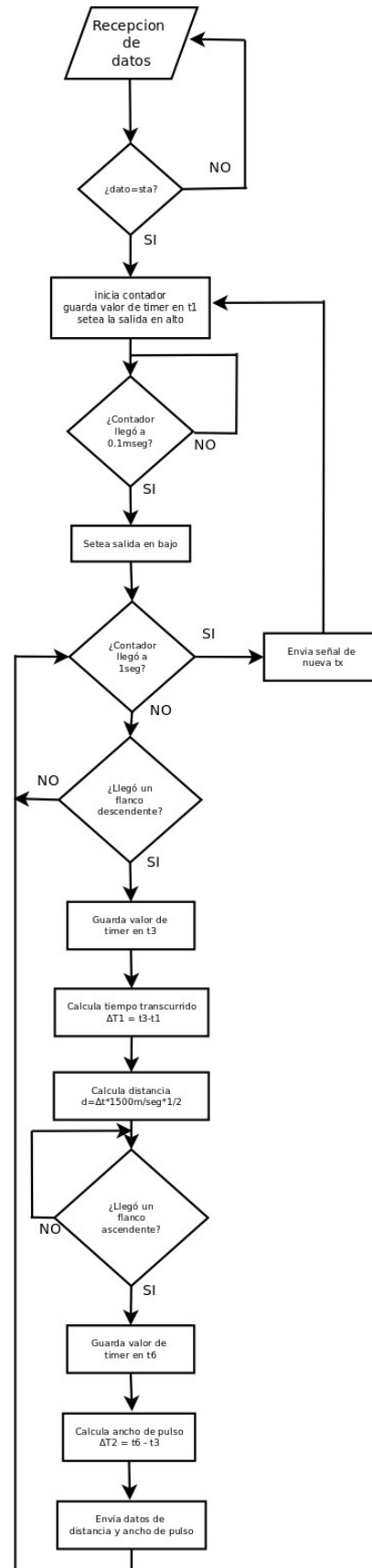


Figura 12. Diagrama de Flujo, bloque de control.

### 3.5. Módulo bluetooth

Para la conexión entre el bloque de control y la interfaz visual se optó por la implementación de una comunicación inalámbrica por medio del protocolo bluetooth, el cual es compatible con las interfaces de comunicación serie, lo que significa que se puede conectar directamente al puerto serie del PIC sin la necesidad de ningún cambio en el software del mismo. Para ello se utilizó un módulo bluetooth modelo HC-05. Este debe ser configurado para coincidir con la configuración utilizada por el puerto serie del microcontrolador PIC y para ser detectado correctamente por la interfaz visual. Esta configuración se realiza por medio de los llamados comandos AT y debe ser la siguiente:

Role: Slave;

Baud Rate: 115200;

Parity: none;

Data Bits: 8;

Stop Bits: 1;

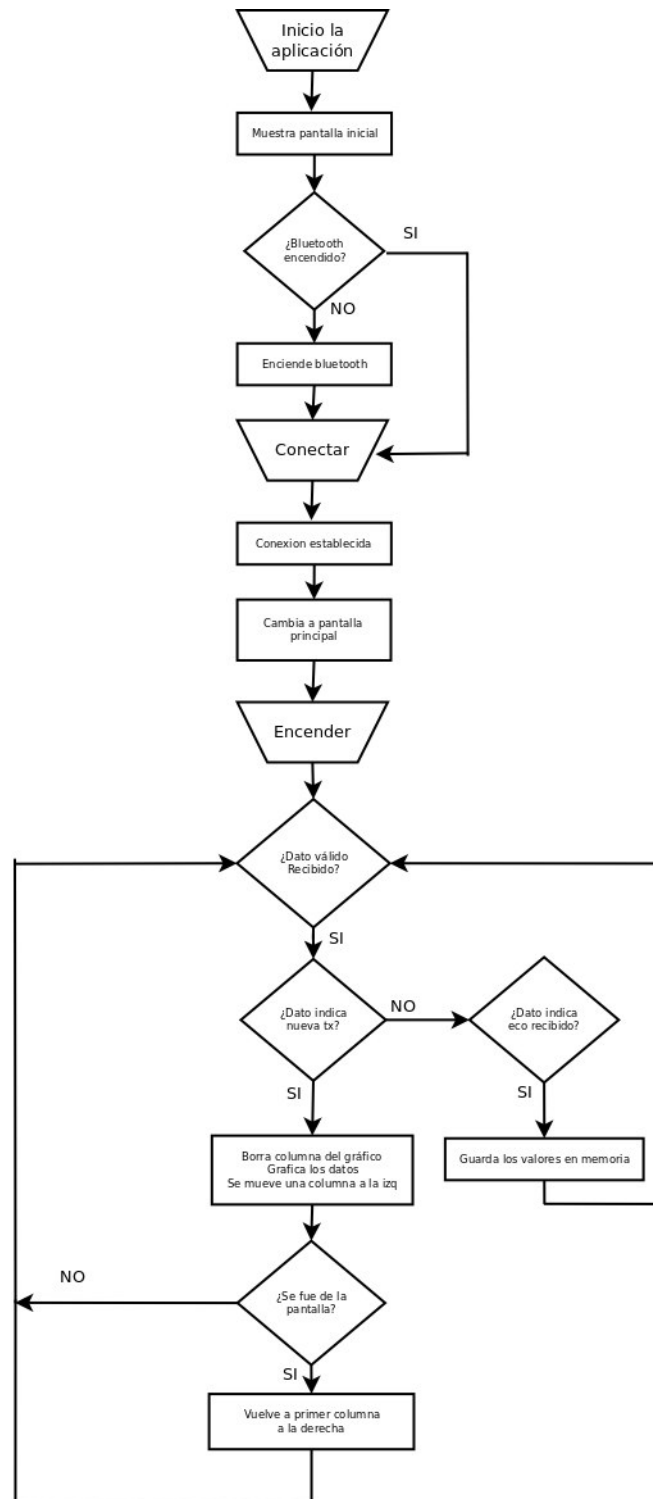
Nombre: ecosonda;

Password: 1234;

Para más información sobre el módulo bluetooth y su forma de configuración, remitirse a la bibliografía.

### 3.6. Interfaz visual

Para la interfaz visual se optó por la creación de una aplicación para dispositivos basados en el sistema operativo Android. Las pruebas realizadas durante el proyecto, se llevaron a cabo con una Tablet de 7" con Android 4.2.2 (Jelly Bean), pero la aplicación es compatible con cualquier dispositivo basado en sistemas operativos Android, y se adapta a distintos tamaños de pantalla de manera automática. En la Figura 13, se observa el diagrama de flujo del funcionamiento de la aplicación.



**Figura 13. Diagrama de flujo, aplicación Android.**

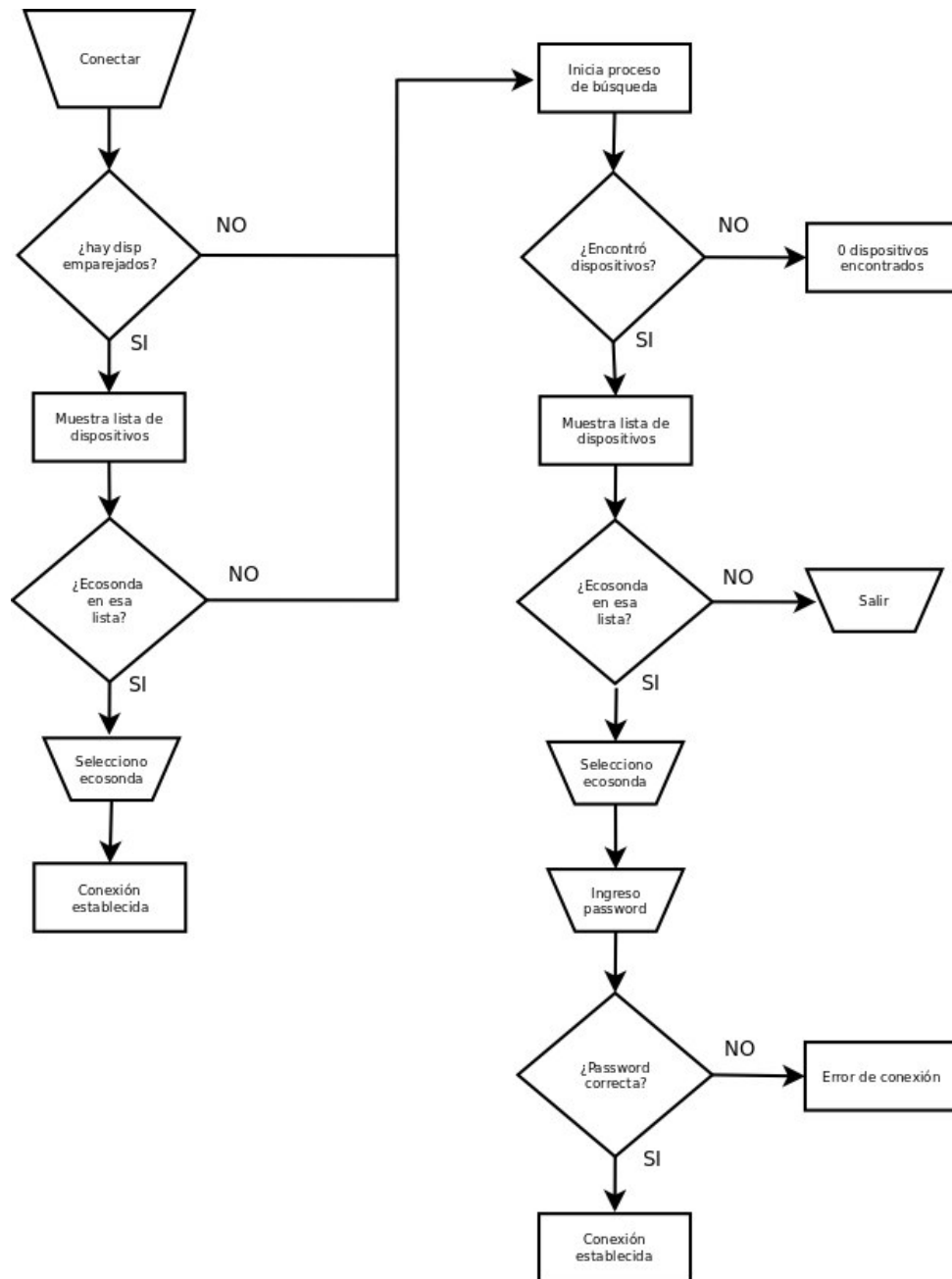
La aplicación consta de dos pantallas distintas. Una pantalla inicial o de bienvenida, mostrada en la Figura 14, y una pantalla principal, mostrada en la Figura 15. Al iniciar la aplicación se muestra la pantalla de bienvenida, y de manera automática se enciende la interfaz Bluetooth del dispositivo, si esta no se encuentra activa. Esta pantalla consta de un conciso mensaje y tres botones. Se debe

elegir entre uno de los dos tipos de gráfico y luego se presiona el botón “Conectar”. En caso de no elegir ningún gráfico, la aplicación iniciará por defecto, utilizando el gráfico “Tipo 1”.



Figura 14. Pantalla inicial.

Al presionar el botón “Conectar”, se inicia el proceso de conexión entre el dispositivo y el módulo bluetooth HC-05. El proceso de conexión se muestra en la Figura 15.



**Figura 15. Proceso de conexión bluetooth.**

Una vez establecida la conexión bluetooth la aplicación pasará a la pantalla principal. En este punto la aplicación ya está conectada al PIC pero éste aún está en estado de reposo. Para poner en funcionamiento el sistema se debe presionar el botón “Comenzar”, ubicado en la esquina superior izquierda de la pantalla, al hacerlo la aplicación envía por medio de la conexión bluetooth el comando que espera el PIC para iniciar su funcionamiento. En este punto la aplicación comienza a recibir los datos enviados por el PIC, los cuales pueden ser de dos tipos:

- ✓ Tipo A, datos numéricos que indiquen la distancia y la potencia de un eco recibido;
- ✓ Tipo B, un carácter que indique que se ha enviado un nuevo pulso de transmisión.

Dependiendo el tipo de gráfico que se haya seleccionado, el procesamiento de estos datos es distinto.

El código fuente de la aplicación Android, puede consultarse en el Apéndice B.

### 3.7. Gráficos

#### 3.7.1. Tipo 1

Este es el gráfico comúnmente utilizado en la mayoría de las ecosondas comerciales. La pantalla está dividida en columnas, correspondiendo cada una de ellas a una transmisión. Cada eco recibido se grafica por medio de un punto de color que indica la potencia, mientras que la profundidad está representada por la distancia del punto al margen superior del gráfico, que representa la línea de flotación de la embarcación.

En caso de recibirse un dato del tipo A, los valores se guardan en memoria y se espera por la recepción de un nuevo dato. Si se recibe un dato del tipo B, la aplicación graficará los valores anteriormente guardados en memoria y luego se moverá una columna hacia la izquierda, a la espera del próximo dato de tipo B. En caso de estar posicionada en la última columna, se vuelve a la primer columna de la pantalla, se grafica el nuevo valor, y de esta manera se refrescan los datos columna por columna.

El color del punto graficado, hace referencia a la potencia del eco recibido, representado el color amarillo (código #FFFF00) la mínima potencia, y con el color rojo (código #FF0000) la máxima potencia. De este código de colores, pueden obtenerse 256 posibilidades, representando cada uno de ellos, una potencia diferente, como se muestra en la Figura 16. En la Figura 17, se observa una captura de pantalla, de la ecosonda funcionando con el gráfico de Tipo 1.



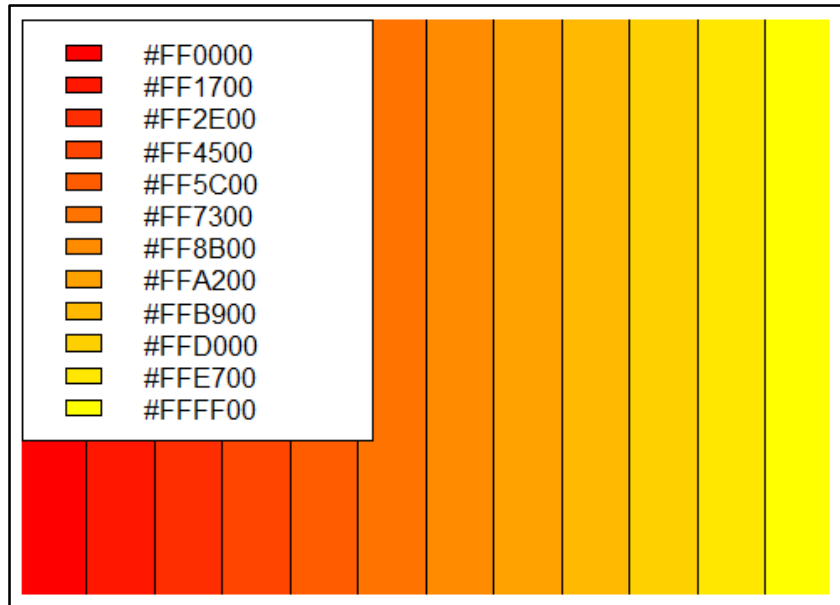


Figura 16. Código de colores.

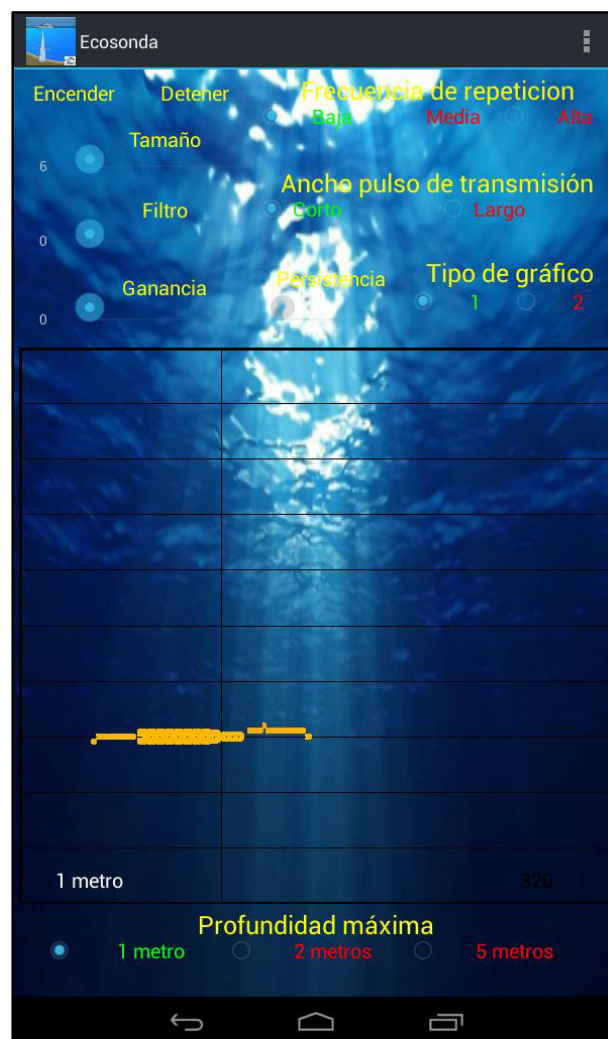


Figura 17. Gráfico convencional, o Tipo 1.

### 3.7.2. Tipo 2

Éste gráfico, se conoce técnicamente como “Scan A”. No existe en ecosondas comerciales, pero es ampliamente utilizado en equipos de inspecciones ultrasónicas. En el eje horizontal del gráfico se setea el “campo” o alcance en distancia que se desea observar en pantalla, y el eje vertical indica la potencia de la reflexión. Para este desarrollo, al estar este dato digitalizado, ésta información está codificada en barras de diferentes alturas. Este tipo de gráfico se incorporó al desarrollo solamente con fines de diseño, ya que facilita la interpretación de los datos al dar información semicuantitativa, y no solo cualitativa como el gráfico de Tipo 1. En la Figura 18 se observa una captura de pantalla, de la ecosonda funcionando con el gráfico de Tipo 2. El campo (eje horizontal) se configuró en 5 metros, y las barras verticales indican los ecos correspondientes a los distintos ecos de fondo sucesivos. El dispositivo, estaba siendo testeado en una columna de agua de aproximadamente 1 metro de altura.

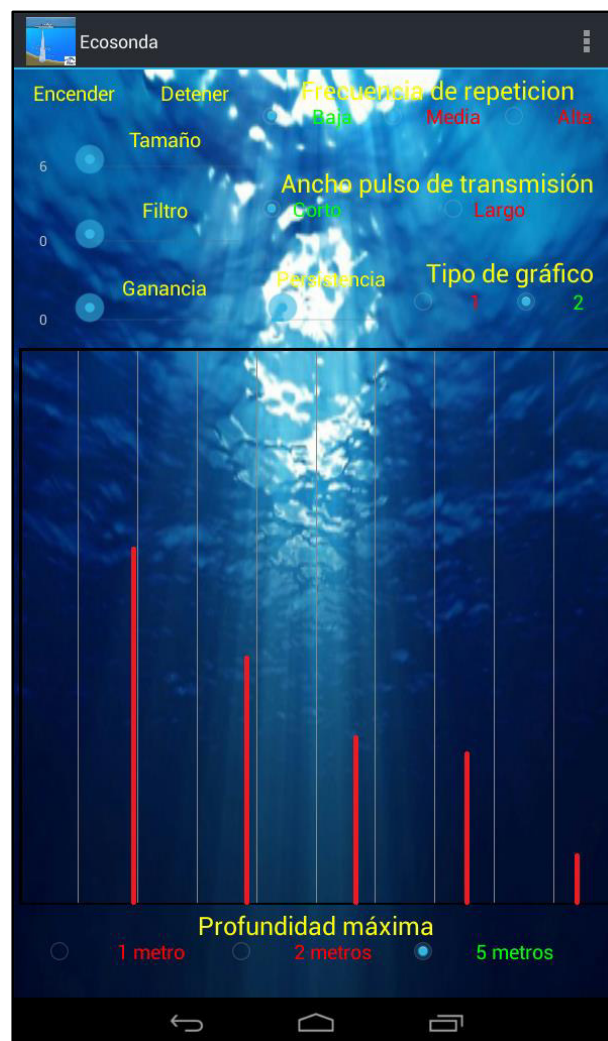


Figura 18. Scan-A, o Gráfico Tipo 2.

Éste gráfico muestra la información recolectada en la última transmisión. En caso de necesitar observar más de una transmisión de manera simultánea, se puede recurrir a modificar el control de “persistencia”, para evitar que la información se borre (refresque en pantalla) luego de cada transmisión.

Si se quiere detener el dispositivo, bastará con presionar el botón “detener” ubicado junto al botón “comenzar”, en la pantalla principal. Al hacerlo la aplicación enviará por medio de la conexión bluetooth un comando que le indicará al PIC que debe detener su funcionamiento.

La pantalla principal posee, además, distintos botones, cuyo funcionamiento será detallado en el manual de operación de la ecosonda.

### 3.8. Alimentación

Debido a que el módulo de control y el módulo transmisor/receptor utilizan diferentes tensiones de alimentación (5 y 9 V, respectivamente) se optó por utilizar una única batería, y dos convertidores DC-DC para alimentar cada módulo. La batería utilizada, de 12V y 7AH, es comúnmente utilizada en las ecosondas comerciales y los convertidores DC-DC (Step-Down, LM2596) tienen la ventaja, además de ser económicos, de permitir variar su voltaje de salida mediante un preset, permitiendo de esta manera, utilizar otras fuentes de alimentación sin la necesidad de realizar ningún cambio a nivel de hardware en el dispositivo.

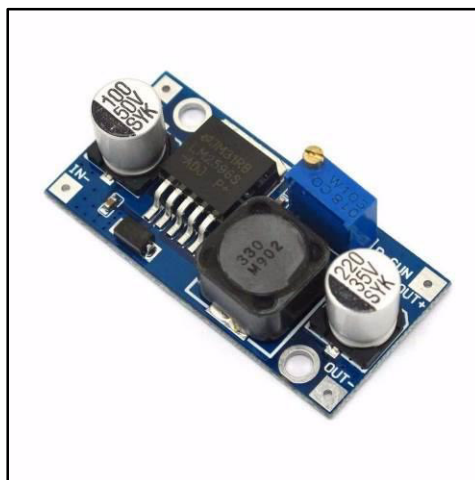


Figura 19. Convertidor DC-DC LM2596

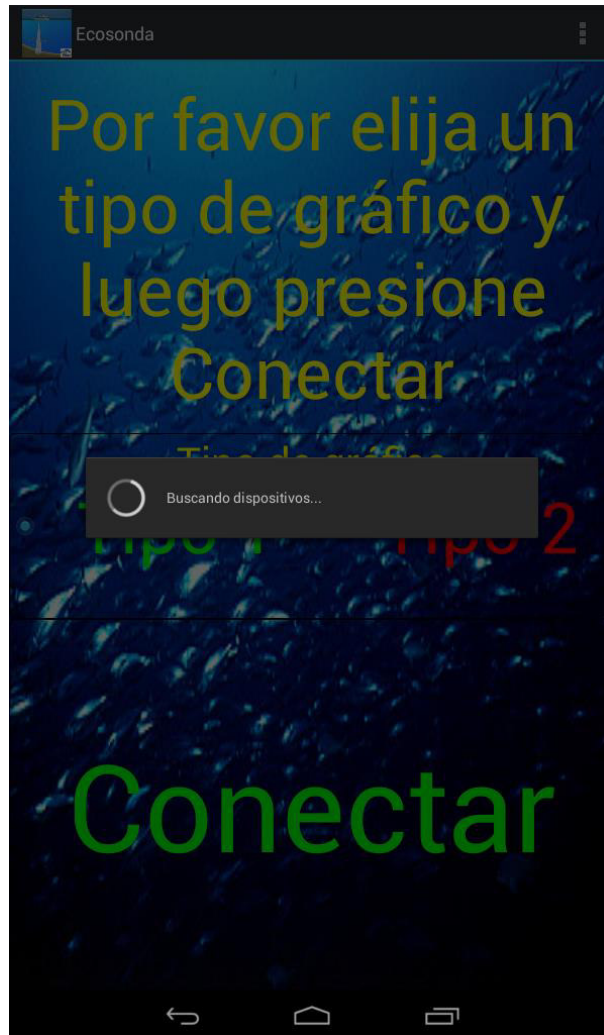
## 4. Manual de operación

Al iniciar la aplicación se muestra la pantalla de bienvenida, encendiendo al mismo tiempo la interfaz bluetooth del dispositivo Android. Esta pantalla consta de un conciso mensaje y tres botones.



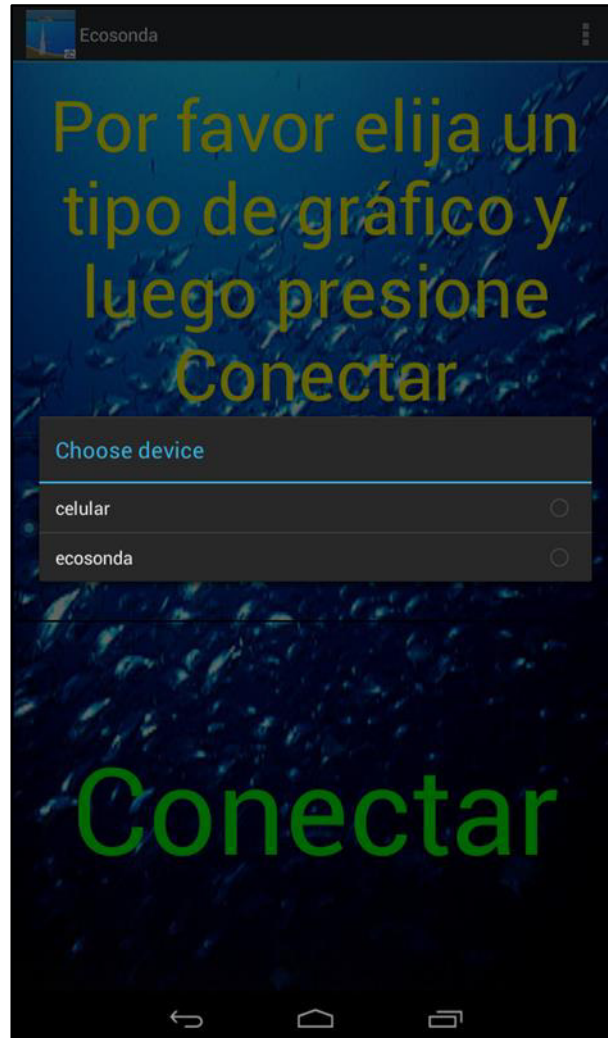
Figura 20. Pantalla inicial. Aplicación Android.

Se debe elegir entre ambos tipos de gráfico y luego presionar el botón Conectar. En caso de no elegir ningún gráfico, la aplicación seleccionará por defecto el gráfico “Tipo 1”. Al presionar el botón Conectar se inicia el proceso de búsqueda de dispositivos bluetooth. Durante este proceso, se mostrará el mensaje “Buscando dispositivos...”, como se muestra en la Figura 21.



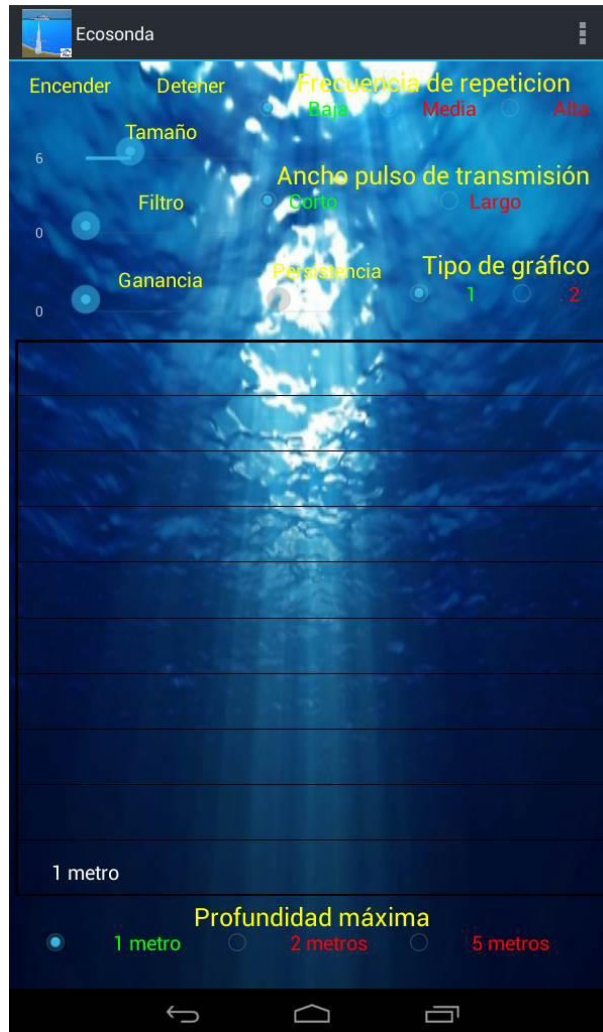
**Figura 21. Proceso de búsqueda bluetooth.**

Una vez terminada la búsqueda, se muestra en pantalla la lista de los dispositivos encontrados. El módulo bluetooth de la ecosonda se identifica a sí mismo como "ecosonda", y es con este dispositivo con cual debe emparejarse la terminal Android que se esté utilizando para poder comenzar a operar la ecosonda, como se muestra en la Figura 22. En caso de ser la primera vez que se emparejen, la aplicación solicitará el ingreso de una contraseña para autenticarse con la ecosonda. Dicha contraseña es "1234".



**Figura 22. Lista de dispositivos encontrados.**

Una vez establecida la conexión bluetooth, la aplicación automáticamente pasará a la pantalla principal, la cual se muestra en la Figura 23.



**Figura 23. Pantalla principal. Aplicación Android.**

En esta pantalla principal, se ubican todos los controles necesarios para la operación de la ecosonda. A continuación, se detalla la función de cada uno de ellos.

#### 4.1. Encender/Detener

En la esquina superior izquierda de la pantalla se ubican los botones correspondientes al encendido y apagado de la ecosonda. Los mismos se verán como en la Figura 24-a, cuando la ecosonda esté apagada, y se mostrarán como en la Figura 24-b, cuando esta se encuentre activa y encendida.



Figura 24-a. Ecosonda apagada.

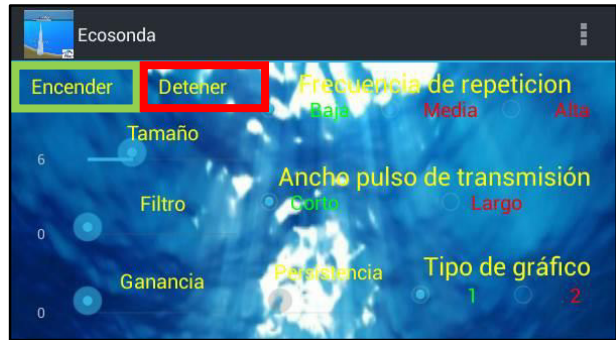


Figura 24-b. Ecosonda encendida.

#### 4.2. Tamaño

Se encuentra disponible un control para aumentar o disminuir el tamaño del punto a graficar. Si bien aumentar el tamaño del punto conlleva una reducción en la cantidad de datos que pueden mostrarse en pantalla, visualmente resulta más fácil de ver, sobre todo si la ecosonda se utiliza con un terminal Android con una pantalla reducida en tamaño.

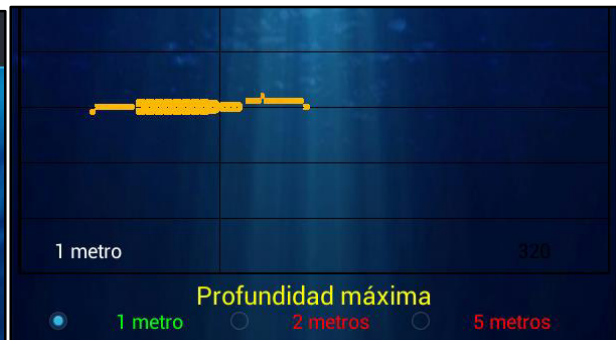
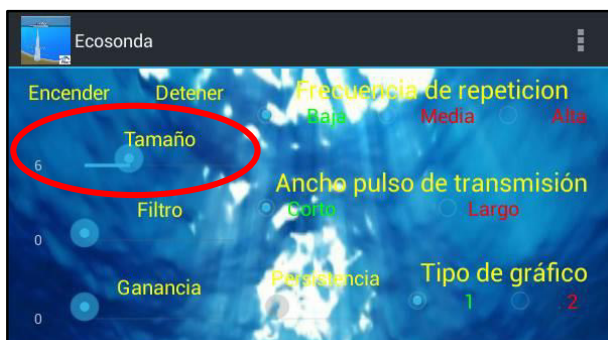
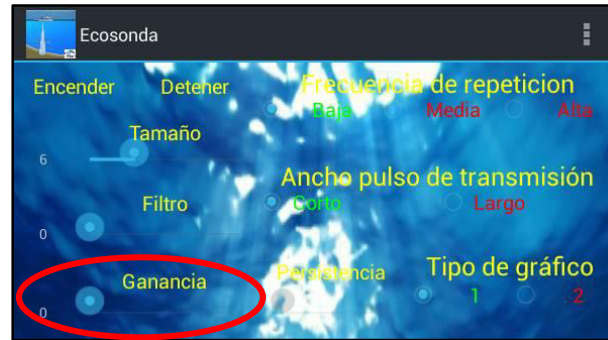


Figura 25-a. Selección del tamaño del punto. Figura 25-b. Distintos tamaños de puntos.

#### 4.3. Ganancia

La aplicación posee un control de ganancia que amplifica por software los ecos recibidos con el fin de poder distinguir con mayor precisión la diferencia de potencia de los mismos. Este control opera únicamente sobre la visualización de los datos, no modifica de ninguna forma la sensibilidad de recepción del dispositivo.

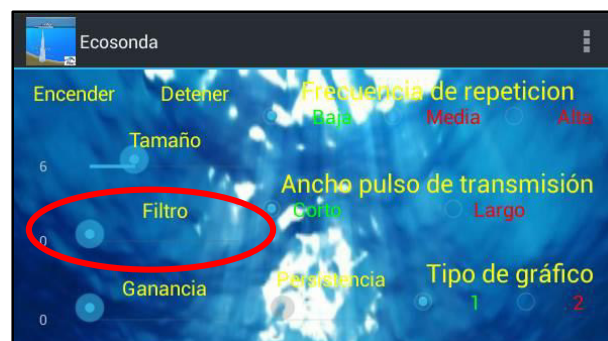




**Figura 26. Control de Ganancia.**

#### 4.4. Filtro

El control de filtro permite seleccionar la potencia mínima que debe poseer un eco para que éste sea graficado, es decir, todo eco cuya potencia sea menor a la seleccionada, no aparecerá en pantalla. Este control resulta útil en situaciones en las que existen ecos no deseados, debidos a la presencia partículas en suspensión, algas, o cardúmenes de peces pequeños que no son de interés, pero que de graficarse provocan una pantalla con puntos que no resultan útiles. Estos ecos indeseados pueden filtrarse utilizando este control, de forma de “limpiar” la pantalla de reflexiones no significativas.



**Figura 27. Control de Filtro.**

#### 4.5. Frecuencia de repetición

La frecuencia de repetición define el tiempo que transcurre desde que termina una transmisión hasta que comienza la siguiente. Cuanto mayor es la frecuencia de repetición, menor es el tiempo transcurrido entre una transmisión y la siguiente, mayor la velocidad de barrido del gráfico y mayor el consumo de energía de la ecosonda.



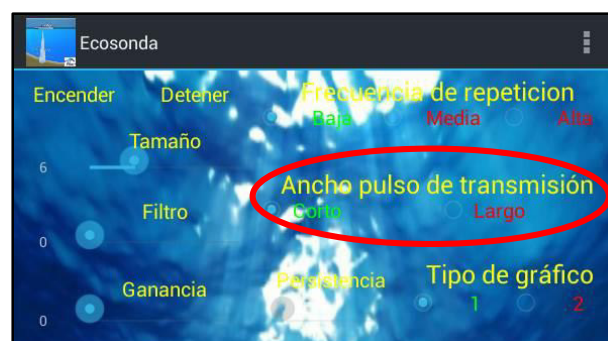
**Figura 28. Frecuencia de Repetición.**

Este control posee 3 valores posibles:

- ✓ Baja: 1 segundo entre una transmisión y la siguiente;
- ✓ Media: 0.5 segundos entre una transmisión y la siguiente;
- ✓ Alta: 0.1 segundo entre una transmisión y la siguiente.

#### 4.6. Ancho de pulso de transmisión

Define el tiempo durante el cual se transmite la señal de ultrasonido. Cuanto mayor es el ancho de pulso, mayor será la energía transmitida, logrando mayor alcance. Se debe tener cuidado al operar sobre este control debido a que, por un lado, aumentar la energía transmitida reduce la autonomía del dispositivo, y por otro lado no es posible detectar ecos producidos mientras el propio pulso se está transmitiendo. Esto se debe a que el mismo transductor se utiliza tanto para emitir, como para recibir las señales, y estas dos funciones no pueden realizarse en forma paralela.



**Figura 29. Ancho del pulso de transmisión.**

Este control posee 2 valores posibles:

- ✓ Corto: ancho de pulso de 0.1mseg. Distancia mínima de recepción (teórica) de 7.5cm;

- ✓ Largo: ancho de pulso de 0.3mseg. Distancia mínima de recepción (teórica) de 22.5cm.

#### 4.7. Profundidad máxima

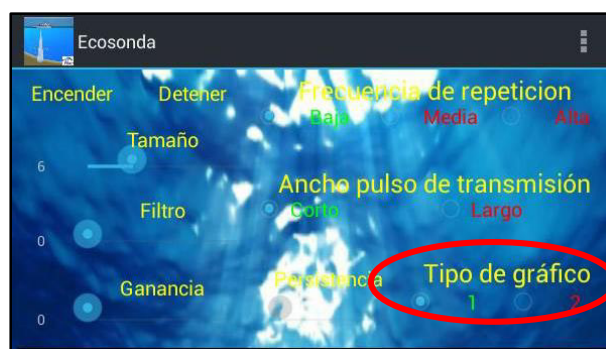
Este control, ubicado en la parte inferior de la pantalla principal, permite seleccionar la máxima profundidad a ser graficada. Cualquier eco que provenga de una distancia mayor a la especificada como profundidad máxima será ignorado. Pueden seleccionarse tres valores distintos de profundidad máxima y estos son: 1 metro, 2 metros y 5 metros. La profundidad máxima teórica que puede lograr la ecosonda, es de aproximadamente 50 metros.



**Figura 30. Control de profundidad.**

#### 4.8. Tipos de gráficos

Este control permite cambiar la forma en la que se visualizan los ecos en la pantalla de la ecosonda. se puede intercambiar entre los 2 tipos de gráfico existentes.



**Figura 31. Control de tipo de gráfico.**

El gráfico de Tipo 1, se encuentra dividido en renglones horizontales, representando estos, distintos niveles de profundidad de la columna de agua que se encuentra bajo la embarcación. Si el control de profundidad es elegido correctamente, la ecosonda mostrará en la parte inferior de la pantalla, una línea que representará el perfil del fondo marino. Cualquier pez, o cardúmenes de

peces que se encuentren bajo la embarcación, serán representados en forma de puntos. Estos puntos varían su color desde el amarillo, hasta el rojo, representando reflexiones de menor o mayor intensidad respectivamente. Generalmente, la reflexión de mayor intensidad será la producida por el perfil del suelo marino.

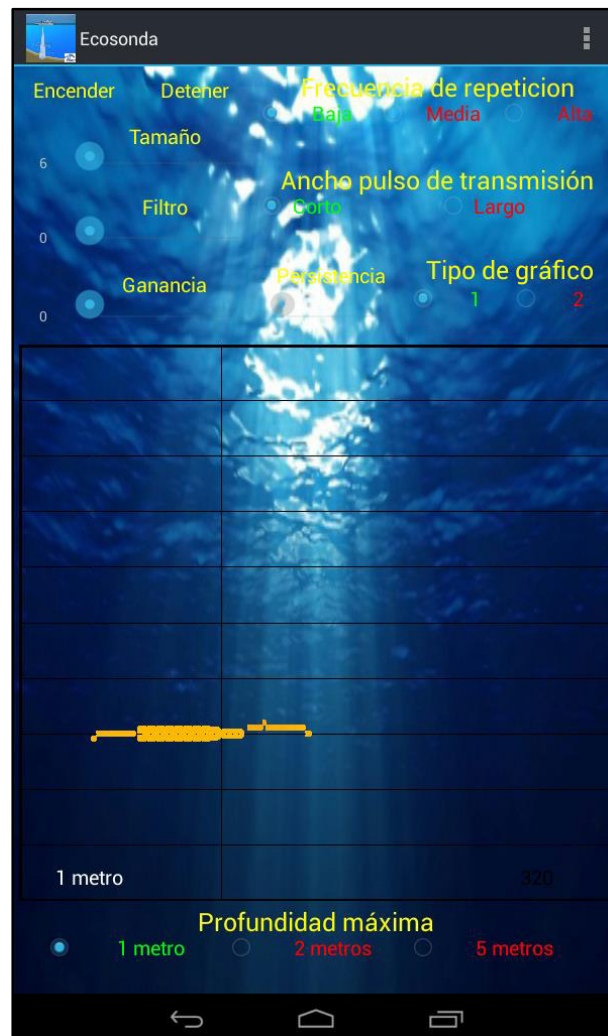


Figura 32. Gráfico Tipo 1.

El gráfico de Tipo 2, también conocido como Scan-A, es una representación en barras, de los distintos ecos recibidos. Las diferencias en la potencia de las reflexiones, se codifican en la altura de las barras. Si el control de profundidad es elegido correctamente, la barra de mayor altura se ubicará en la parte derecha de la pantalla, mostrando la ubicación en distancia del fondo marino, y las diferentes reflexiones producidas por los peces o cardúmenes de peces, se mostrarán como barras de menor altura a la izquierda de la barra del eco de fondo.

La utilización e interpretación de éste tipo de gráfico, resulta un tanto compleja, y solo se

recomienda su uso a usuarios con conocimientos avanzados en ultrasonido.

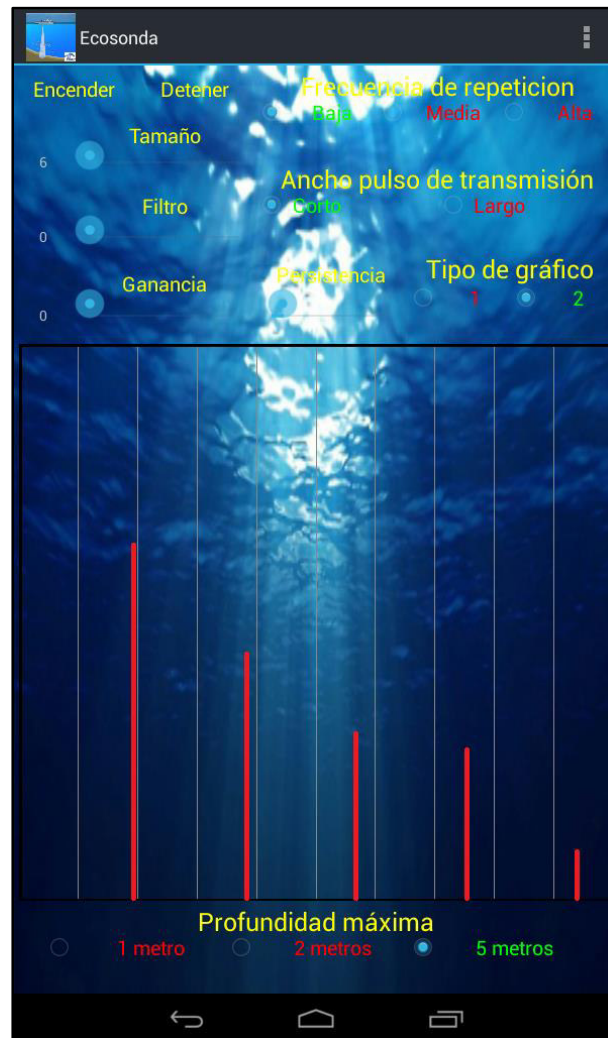


Figura 33. Gráfico Tipo 2.

#### 4.9. Persistencia

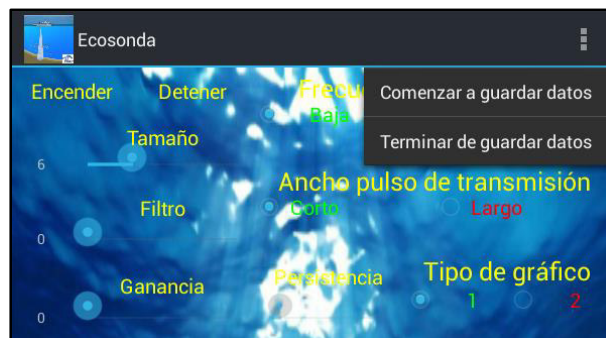
Este control se habilita únicamente cuando se está trabajando con el gráfico Tipo 2 o "Scan A". Por defecto se muestra información de una única transmisión, refrescándose completamente la pantalla en la transmisión posterior. Este control permite configurar la cantidad de transmisiones a ser graficadas en una misma pantalla antes de que ésta sea borrada.



**Figura 34. Control de Persistencia en Gráfico Tipo 2.**

#### 4.10. Guardar datos

La aplicación ofrece la posibilidad de guardar los datos obtenidos de las distintas reflexiones, para un posterior análisis. Esta utilidad, si bien no es necesaria para realizar la actividad de pesca, se incorporó con fines científicos y/o educativos.



**Figura 35. Control de guardado de datos.**

Esta opción debe seleccionarse presionando los tres puntos que aparecen en el borde superior derecho, en la pantalla principal de la ecosonda. Si el dispositivo Android posee instalada una tarjeta de memoria, al seleccionar la opción de “Comenzar a guardar datos” la aplicación crea en la misma, un archivo de texto cuyo nombre está formado por la fecha y hora del momento en que empezó a guardarse los datos. Dentro de ese archivo de texto se registran los datos recibidos de distancia y potencia, así como también la hora de dicha medición. Todos los datos se muestran separados por un punto y coma (“;”), de forma tal de poder exportarse directamente a una planilla Excel para su posterior análisis o representación. Los datos se registran de forma continua, hasta que se presiona el botón “Terminar de Guardar Datos”. En la Figura 36, se muestra un ejemplo de un archivo de texto generado mediante esta funcionalidad.

```
*27720161811.txt x
Hora ; distancia en cm ; potencia (max = 500); pulso de tx al que pertenece el dato
18:11:38;80;307;416
18:11:38;169;285;416
18:11:38;255;270;416
18:11:38;340;249;416
18:11:38;429;244;416
18:11:38;516;217;416
18:11:38;604;200;416
18:11:38;691;170;416
```

Figura 36. Ejemplo de datos registrados.

## 5. Mejoras Futuras

Durante el desarrollo del proyecto, se identificaron mejoras, a nivel de hardware y software, que podrían incorporarse a la ecosonda en futuras actualizaciones del diseño.

A nivel de hardware, sería interesante incorporar al diseño un sensor de temperatura, puesto que este dato, muchas veces resulta importante a la hora de elegir un buen sitio de pesca.

Sin lugar a dudas, las mejoras futuras más interesantes para este proyecto, son a nivel de software, y se detallan a continuación:

- ✓ Aprovechando que el módulo utilizado cuantifica la potencia del eco recibido, podría utilizarse esta información para determinar el tipo suelo marino, es decir, la posibilidad de diferenciar entre un suelo rocoso de uno de arena;
- ✓ Si el dispositivo Android que se esté utilizando cuenta con un módulo GPS, éste podría ser utilizado por la aplicación, por ejemplo, para llevar un control de la velocidad a la que se desplaza la embarcación o para guardar las coordenadas GPS de un punto de pesca determinado, para poder volver a visitarlo en futuras excursiones o simplemente para compartirlo con otros pescadores;
- ✓ La posibilidad de tener acceso a internet desde el dispositivo Android, abre un abanico muy grande de posibilidades, pudiendo incorporarse a la aplicación, por ejemplo, reportes del clima, conexión directa con redes sociales, entre otras.

## 6. Conclusiones

Después de diseñar y desarrollar este proyecto, se llegó a las siguientes conclusiones finales:

- ✓ Si bien se presentaron diversas problemáticas durante el desarrollo de este proyecto, éstas pudieron ser sorteadas, y muchas veces las soluciones adoptadas terminaron enriqueciendo la versión final del dispositivo;
- ✓ El hecho de que se adoptara una conexión totalmente inalámbrica entre el cuerpo electrónico de la ecosonda y la interfaz visual y de control por parte del usuario, representa, además de una innovación no existente en el mercado, una minimización de las conexiones físicas. Esto reduce significativamente el daño de tipo corrosivo, intrínseco en el ambiente marítimo;
- ✓ Utilizar un dispositivo móvil Android, como parte central del desarrollo, representa uno de los puntos más fuertes del mismo, permitiendo incorporar al proyecto funciones como GPS, conexión a internet, entre otras;
- ✓ Unos de los principales problemas que tienen las ecosondas comerciales, y limitan en gran parte su utilización en embarcaciones que no posean motor, es la duración de las baterías que utilizan. Teniendo en cuenta que el cuerpo electrónico de la ecosonda desarrollada no posee pantalla de ningún tipo, la duración de la batería es intrínsecamente superior a la de sus pares comerciales;

Se considera que los objetivos del proyecto fueron alcanzados con éxito, y al no existir en el mercado un dispositivo que ofrezca similares características, se deja abierta la posibilidad de comenzar a producirlo con fines comerciales.



## 7. Bibliografía

1. Introducción al uso de sistemas de sonar para la estimación de la biomasa de peces.  
*FAO, Documentos técnicos de pesca N°191;*
2. Postprocessing system for echo sounder data.  
*Kenneth G. Foote, Hans Petter Knudsen, and Rolf J. Korneliussen. Institute of Marine Research, 5024 Bergen, Norway. 20 March, 1991;*
3. Variations in echosounder–transducer performance with water temperatura.  
*David A. Demer and Josiah S. Renfree. ICES Journal of Marine Science, 65: 1021–1035. 2008.*
4. Garmin Transducer Selection Guide.  
*Información de catálogo, 2015;*
5. Nondestructive Material Testing with Ultrasonics.  
*Michael Berke. Krautkramer;*
6. Como Interpretar las Imágenes de la Ecosonda.  
*Material propio de Escuela Nacional de Pesca - Comandante Luis Piedra Buena;*
7. Fundamentals\_of\_Acoustical Oceanography  
*Herman Medwin, Clarence S. Clay, 1998;*
8. Fisheries Acoustics.  
*Simmons McLennan, Blackwell Science 2005;*
9. Principles of Marine Bioacoustics.  
*Whitlow, Hastings. Ed. Springer 2008;*
10. Manual de Configuración Modulo HC-05.  
[http://www.naylampmechatronics.com/blog/24\\_Configuración--del-módulo-bluetooth-HC-05-usa.html](http://www.naylampmechatronics.com/blog/24_Configuración--del-módulo-bluetooth-HC-05-usa.html);

# APÉNDICE A

## Código Fuente del Microcontrolador PIC

```
#include "C:\Users\Gonzalo\Desktop\codigo PIC Tesis\Tesis.h"
#include <stdio.h>
#include <stddef.h>
#include <stdlib.h>
#include <stdlibm.h>
#include <string.h>

char * buffer;
char * comando;
int8 j,i,N,B4a,B4,yapaso,asc,repeticion,cantidad_ecos,contador, auxiliar;
int16 int_tx,int_10mseg,distancia_minima;
int32 t1,Tfd,Ttx,valor,overflow_potencia,overflow_distancia;
signed int32 * distancia;
signed int32 * potencia;

// Prueba si los caracteres recibidos son un comando válido
int8 comparar_comandos (void)
{
    int8 valor = 1,i;

    for (i = 0;i < 3; i++)
    {
        if (comando[i] != buffer[i])
        {
            valor = 0;
        }
    }
    return(valor);
}

// Calcula la distancia y la potencia del eco recibido
void calculo(void)
{
    //Detección de flanco descendente
    if ((B4 == 0)&&(B4a == 1))
    {
        if (asc == 1)
        {
            Tfd = t1;
            // 65535 son los ciclos de reloj, 167nseg el período del mismo, y dividido por 1000 para obtener el valor en useg
            distancia[cantidad_ecos] = ((t1+(65535*overflow_distancia)-Ttx)*167)/1000;
            B4a = B4;
            overflow_potencia = 0;
            asc = 0;
        }
    }
    else
    {
        //Detección de flanco ascendente
        if ((B4 == 1)&&(B4a == 0))
        {
            if (asc == 0)
            {
                // 65535 son los ciclos de reloj, 167nseg el período del mismo, y dividido por 1000 para obtener el valor en useg
                potencia[cantidad_ecos] = ((t1+(65535*overflow_potencia)-Tfd)*167)/1000;
                B4a = B4;
                asc = 1;
                cantidad_ecos++;
            }
        }
    }
}
```

```

    }
  }
  return;
}

#int_TIMER2
void TIMER2_isr(void)
{
  //Cada 3 interrupciones envió un dato, para darle tiempo suficiente a la transmisión
  if (auxiliar == 3)
  {
    auxiliar = 1;
    if ( contador < cantidad_ecos)
    {
      //Para evitar errores, ignora los valores menores a la mínima distancia que es capaz de medir
      if (distancia[contador] > distancia_minima)
      {
        //Envía los valores de distancia y potencia en useg
        printf("%06ld;%06ld\n\r", distancia[contador], potencia[contador]);
      }
      contador++;
    }
    else
    {
      disable_interrupts(INT_TIMER2);
    }
  }
  auxiliar++;
}

#int_TIMER3
void TIMER3_isr(void)
{
  overflow_potencia++;
  overflow_distancia++;
}

#int_RB
void RB_isr(void)
{
  //Guarda el valor del timer al momento de la interrupción, toma el valor de la entrada digital y luego llama a la
  //función calculo
  t1 = get_timer3();
  B4 = input_state(PIN_B4);
  calculo();
}

#int_RDA
void RDA_isr(void)
{
  //Toma el caracter que llega por RS232
  buffer[j] = getch();
  j++;
  //Cuando llega el caracter de fin de línea comienza a comparar los datos recibidos con los comandos programados
  if (buffer[j - 1] == 0x0A)
  {
    //Deshabilita todas las interrupciones y detiene la transmisión y recepción de datos
    strcpy(comando, "stp");
    if (comparar_comandos() == 1)
    {

```

```
disable_interrupts(INT_RB);
disable_interrupts(INT_CCP1);
disable_interrupts(INT_TIMER2);
//Apaga el led indicador de encendido
output_low(PIN_B3);
}
//Setea el período de repetición deseado en 1seg
strcpy(comando, "frb");
if (comparar_comandos() == 1)
{
    repeticion = 93;
}
//Setea el período de repetición deseado en 0.5seg
strcpy(comando, "frm");
if (comparar_comandos() == 1)
{
    repeticion = 47;
}
//Setea el período de repetición deseado en 0.2seg
strcpy(comando, "fra");
if (comparar_comandos() == 1)
{
    repeticion = 20;
}
//Setea el ancho del pulso de transmisión en 10useg
strcpy(comando, "pte");
if (comparar_comandos() == 1)
{
    int_tx = 60;
}
//Setea la distancia mínima que puede medir para este ancho de pulso
distancia_minima = 100;
}
//Setea el ancho del pulso de transmisión en 50useg
strcpy(comando, "ptl");
if (comparar_comandos() == 1)
{
    int_tx = 300;
}
//Setea la distancia mínima que puede medir para este ancho de pulso
distancia_minima = 200;
}
//Identificación del dispositivo
strcpy(comando, "ide");
if (comparar_comandos() == 1)
{
    printf("Ecosonda Android 2016 By Polvorin & Valle\n\r");
}
//Setea los valores por defecto y enciende el dispositivo
strcpy(comando, "sta");
if (comparar_comandos() == 1)
{
}
//Enciende el led indicador de encendido
output_high(PIN_B3);
repeticion = 93;
N = 0;
int_tx = 60;
CCP_1 = int_tx;
setup_ccp1(CCP_COMPARE_RESET_TIMER);
enable_interrupts(INT_CCP1);

}
```

```

    j = 0;
  }

}

#int_CCP1
void CCP1_isr(void)
{
  //Envia el pulso de transmisión
  //En la primer interrupción se pone la salida en alto
  if (yapaso == 0)
  {
    //Guarda el valor del timer al momento de la transmisión
    Ttx = get_timer3();
    output_high(PIN_B0);
    //Inicia el contador de overflows del timer para el cálculo de la distancia
    overflow_distancia = 0;
    yapaso++;
  }
  else
  {
    //En la segunda interrupción se pone la salida en bajo
    if (yapaso == 1)
    {
      //Setea las interrupciones del timer cada 10mseg
      CCP_1 = int_10mseg;
      output_low(PIN_B0);
      yapaso++;
    }
    //Una vez terminada la transmisión se habilita la recepción
    enable_interrupts(INT_RB);
  }
}
N++;
//Aprox 200mseg después de la transmisión(equivalente a 150m de distancia) habilita interrupciones para la
//transmisión de datos
if (N == 21)
{
  enable_interrupts(INT_TIMER2);
}
//Al llegar al período de repetición seteado deshabilita la recepción de ecos y comienza una nueva transmisión
if ((N >= repeticion)&&(asc == 1))
{
  disable_interrupts(INT_RB);
  N = 0;
  CCP_1 = int_tx;
  //Envia 2 caracteres que indican el comienzo de un nuevo pulso de transmisión
  printf("\n\n\r");
  yapaso = 0;
  cantidad_ecos = 0;
  contador = 0;
}
}

void main()
{
  setup_adc_ports(AN0_TO_AN3|VSS_VDD);
  setup_adc(ADC_CLOCK_DIV_64);
  setup_spi(SPI_SS_DISABLED);

```

```
setup_wdt(WDT_OFF);  
setup_timer_0(RTCC_INTERNAL);  
setup_timer_1(T1_INTERNAL|T1_DIV_BY_2);  
setup_timer_2(T2_DIV_BY_16,252,16);  
setup_timer_3(T3_INTERNAL|T3_DIV_BY_2);  
setup_comparator(NC_NC_NC_NC);  
setup_vref(FALSE);
```

```
enable_interrupts(INT_TIMER3);  
enable_interrupts(INT_RDA);  
enable_interrupts(GLOBAL);
```

#### //Definición de vectores

```
buffer = malloc(8);  
comando = malloc(4);  
distancia = calloc(50,4);  
potencia = calloc(50,4);
```

#### //iniciacion de variables

```
//-----  
output_high(PIN_B1); // enciende un led para indicar que el PIC esta funcionando correctamente  
yapaso = 0;  
i = 0;  
j = 0;  
int_10mseg = 65500; //valor para setear interrupciones de ccp1 cada 10mseg  
overflow_distancia = 0;  
B4a = 1; //valor anterior de la entrada  
B4 = 1; //valor actual de la entrada  
Tfd = 0; //tiempo del flanco descendente  
asc = 1; //indica cual fue el último flanco detectado 1=ascendente, 0=descendente  
overflow_potencia = 0;  
cantidad_ecos = 0;  
contador = 0;  
distancia_minima = 100;  
auxiliar = 1;  
//-----  
while(true)  
{  
  
}
```

# APÉNDICE B

Código fuente de la Aplicación Android



```
#Region Project Attributes
#ApplicationLabel: Ecosonda
#VersionCode: 1
#VersionName:
#SupportedOrientations: portrait
#CanInstallToExternalStorage: true
#End Region

#Region Activity Attributes
#FullScreen: true
#IncludeTitle: True
#End Region

Sub Process_Globals

    Dim guardando As Boolean
    Dim admin As BluetoothAdmin
    Dim serial1 As Serial
    Dim foundDevices As List
    Type NameAndMac (Name As String, Mac As String)
    Dim nm,connectedDevice As NameAndMac
    Dim paired,i,j,jmax,escala,minimo,pmax,conectar,grafico,tx As Int
    Dim fondo_potencia As Long
    Dim fondo_distancia As Long
    Dim recibidos As String
    Dim awake As PhoneWakeState
    Dim tamaño As Float

End Sub

Sub Globals

    Dim TextWriter1 As TextWriter

    Dim btn3,btn4,btnsearch As Button
    Dim Panell,pnlscanb As Panel
    Private pnlgrafico As Panel
    Dim cnvgrafico,cnvscanb As Canvas

    Private Rbtn10 As RadioButton
    Private Rbtn20 As RadioButton
    Private Rbtn50 As RadioButton
    Private frec1 As RadioButton
    Private frec2 As RadioButton
    Private frec3 As RadioButton
    Private ptc As RadioButton
    Private ptl As RadioButton
    Private Classicview As RadioButton
    Private ScanA As RadioButton
    Private grafico1 As RadioButton
    Private grafico2 As RadioButton

    Private tmpts As SeekBar
    Private filtro As SeekBar
    Private ganancia As SeekBar
    Private persistencia As SeekBar

    Dim lbl1 As Label
    Private lbl2 As Label
```

```

Private lbl3 As Label
Private lblganancia As Label
Private lblganancia2 As Label
Private lblinfo As Label
Private lblf As Label
Private lblpt As Label
Private lblprof As Label
Private lbltmpts As Label
Private lblfiltro As Label
Private lblgraficos As Label
Private lblpersistencia As Label
Private lblgrafico As Label
Private lblfondo As Label
End Sub

Sub Activity_Create(FirstTime As Boolean)
    Activity.LoadLayout("main")
    Activity.AddMenuItem("Comenzar a guardar datos","guardar")
    Activity.AddMenuItem("Terminar de guardar datos","terminar")
    Activity.Title = "Ecosonda"
    If FirstTime Then
        paired = 0
        conectar = 0
        tx = 0
        StartService(rctx)
        admin.Initialize("admin")
        serial1.Initialize("serial1")
        If admin.IsEnabled = False Then
            admin.Enable
        End If
    End If
    awake.KeepAlive(True)
    If pnlgrafico.Width < 600 Then
        Classicview.TextSize = 30
        ScanA.TextSize = 30
        lblinfo.TextSize = 30
        lblgrafico.TextSize = 15
        btnsearch.TextSize = 60
    End If
    Classicview.TextColor = Colors.Green
End Sub

Sub Activity_Resume
    paired = 0
    i = 0
    conectar = 0
    tx = 0
    StartService(rctx)
    rctx.ast.Write("stp" & Chr(10))
    admin.Initialize("admin")
    serial1.Initialize("serial1")
    If admin.IsEnabled = False Then
        admin.Enable
    End If
    awake.KeepAlive(True)
End Sub

Sub Activity_Pause (UserClosed As Boolean)
    If serial1.IsEnabled = True Then
        rctx.ast.Write("stp" & Chr(10))
    End If

```

```

        If admin.IsEnabled = True Then
            admin.Disable
        End If
        StopService(rxtx)
        awake.ReleaseKeepAlive
        serial1.Disconnect
        admin.Disable
End Sub

Sub Admin_StateChanged (NewState As Int, OldState As Int)
    If conectar = 1 Then
        'Busca dispositivos bluetooth sincronizados previamente
        Dim PairedDevices As Map
        PairedDevices = serial1.GetPairedDevices
        If PairedDevices.Size <> 0 Then
            paired = 1
            Dim l As List
            l.Initialize
            For i = 0 To PairedDevices.Size - 1
                'Agrega el nombre del dispositivo sincronizado a una lista
                l.Add(PairedDevices.GetKeyAt(i))
            Next
            Dim res As Int
            'Muestra una lista de dispositivos sincronizados
            res = InputList(l, "Choose device", -1)
            If res <> DialogResult.CANCEL Then
                'Si se selecciona un dispositivo de la lista se conecta al mismo
                serial1.Connect(PairedDevices.Get(l.Get(res)))
            Else
                'Si no se selecciona uno comienza una búsqueda de dispositivos bluetooth

                foundDevices.Initialize
                If admin.StartDiscovery = False Then
                    ToastMessageShow("Error al iniciar el proceso de
búsqueda.", True)
                Else
                    ProgressDialogShow("Buscando dispositivos...")
                End If
            End If
        Else
            'Si no hay disp sincronizados previamente comienza una búsqueda de dispositivos

            foundDevices.Initialize
            If admin.StartDiscovery = False Then
                ToastMessageShow("Error al iniciar el proceso de
búsqueda.", True)
            Else
                ProgressDialogShow("Buscando dispositivos...")
            End If
        End If
    End If
End Sub

Sub Serial1_Connected (Success As Boolean)
    ProgressDialogHide
    If Success Then
        ToastMessageShow("Conexion Completa", False)

        rxtx.ast.Initialize(Me, "ast", serial1.InputStream, serial1.OutputStream)
        'Una vez conectado al disp bluetooth se carga la siguiente pantalla
        Activity.RemoveAllViews
    End If
End Sub

```

```

        Activity.LoadLayout("main2")
        definiciones
    Else
        MsgBox(LastException.Message, "Error de conexion.")
        If paired = 1 Then
            foundDevices.Initialize
            If admin.StartDiscovery = False Then
                ToastMessageShow("Error al iniciar el proceso de búsqueda.",
True)
            Else
                ProgressDialogShow("Buscando dispositivos...")
                End If
            End If
        End If
    End Sub

Sub btnsearch_Click
    conectar = 1
    Admin_StateChanged(admin.STATE_ON,0)
End Sub

'Dispositivos bluetooth encontrado
Sub Admin_DeviceFound (Name As String, MacAddress As String)
    Log(Name & ":" & MacAddress)
    Dim nm As NameAndMac
    nm.Name = Name
    nm.Mac = MacAddress
    foundDevices.Add(nm)
    ProgressDialogShow("Buscando dispositivos (~ dispositivo
encontrado)...".Replace("~", foundDevices.Size))
End Sub

'Fin de proceso de búsqueda de dispositivos bluetooth
Sub Admin_DiscoveryFinished
    ProgressDialogHide
    If foundDevices.Size = 0 Then
        ToastMessageShow("Ningun dispositivo encontrado.", True)
    Else
        Dim l As List
        l.Initialize
        For i = 0 To foundDevices.Size - 1
            Dim nm As NameAndMac
            nm = foundDevices.Get(i)
            l.Add(nm.Name)
        Next
        Dim res As Int
        res = InputList(l, "Seleccione a que dispositivo conectarse", -1)
        If res <> DialogResult.CANCEL Then
            connectedDevice = foundDevices.Get(res)
            ProgressDialogShow("Tratando de conectar con : " &
connectedDevice.Name & " (" & connectedDevice.Mac & ")")
            serial1.Connect(connectedDevice.Mac)
        End If
    End If
End Sub

'Elección de tipo de gráfico
Sub ScanA_CheckedChange(Checked As Boolean)
    grafico = 2
    ScanA.TextColor = Colors.Green
    Classicview.TextColor = Colors.Red

```

```

End Sub
Sub Classicview_CheckedChange(Checked As Boolean)
    grafico = 1
    ScanA.TextColor = Colors.Red
    Classicview.TextColor = Colors.Green
End Sub

'Definiciones correspondientes a la segunda pantalla de la aplicación
Sub definiciones
    cnvgrafico.Initialize(Panell1)
    cnvscanb.Initialize(pnlscanb)
    If grafico = 2 Then
        Panell1.Visible = False
        pnlscanb.Visible = True
        grafico2.Checked = True
        persistencia.Enabled = True
        bordes

    Else
        persistencia.Enabled = False
        grafico1.Checked = True
        grilla

    End If
    j = 0
    i = Panell1.Width - 1
    jmax = 1
    fondo_potencia = 0
    fondo_distancia = 0
    tamaño = 3
    lbl2.Text = filtro.Value
    lbl3.Text = 6
    escala = 100
    lbl1.Text = "1 metro "
    lblganancia.Text = ganancia.Value
'Máximo valor de potencia que espera recibir el dispositivo
    pmax = 1000
    If pnlgrafico.Width < 600 Then
        Rbtn10.TextSize = 12
        Rbtn20.TextSize = 12
        Rbtn50.TextSize = 12
        frec1.TextSize = 12
        frec2.TextSize = 12
        frec3.TextSize = 12
        lbl1.TextSize = 12
        btn3.TextSize = 12
        btn4.TextSize = 12
        lblf.TextSize = 15
        lblpt.TextSize = 15
        lblprof.TextSize = 15
        lbltmpts.TextSize = 12
        lblfiltro.TextSize = 12
        lblgraficos.TextSize = 15
        lblganancia2.TextSize = 12
        lblpersistencia.TextSize = 12
        ptc.TextSize = 12
        ptl.TextSize = 12

    End If
    Rbtn10.TextColor = Colors.Green
    frec1.TextColor = Colors.Green
    ptc.TextColor = Colors.Green
    Activity.Title = "Ecosonda"
End Sub

```

```
'Arma cuadrícula para gráfico tipo 1
Sub grilla
    cnvgrafico.DrawColor(Colors.Transparent)
'Grilla horizontal
    cnvgrafico.DrawLine(0,0.1*Panel1.height,Panel1.width,0.1*Panel1.height,Colors.RGB(0x00,0x00,0x00),1dip)
    cnvgrafico.DrawLine(0,0.2*Panel1.height,Panel1.width,0.2*Panel1.height,Colors.RGB(0x00,0x00,0x00),1dip)
    cnvgrafico.DrawLine(0,0.3*Panel1.height,Panel1.width,0.3*Panel1.height,Colors.RGB(0x00,0x00,0x00),1dip)
    cnvgrafico.DrawLine(0,0.4*Panel1.height,Panel1.width,0.4*Panel1.height,Colors.RGB(0x00,0x00,0x00),1dip)
    cnvgrafico.DrawLine(0,0.5*Panel1.height,Panel1.width,0.5*Panel1.height,Colors.RGB(0x00,0x00,0x00),1dip)
    cnvgrafico.DrawLine(0,0.6*Panel1.height,Panel1.width,0.6*Panel1.height,Colors.RGB(0x00,0x00,0x00),1dip)
    cnvgrafico.DrawLine(0,0.7*Panel1.height,Panel1.width,0.7*Panel1.height,Colors.RGB(0x00,0x00,0x00),1dip)
    cnvgrafico.DrawLine(0,0.8*Panel1.height,Panel1.width,0.8*Panel1.height,Colors.RGB(0x00,0x00,0x00),1dip)
    cnvgrafico.DrawLine(0,0.9*Panel1.height,Panel1.width,0.9*Panel1.height,Colors.RGB(0x00,0x00,0x00),1dip)
'Bordes
    cnvgrafico.DrawLine(0,0*Panel1.height,Panel1.width,0*Panel1.height,Colors.RGB(0x00,0x00,0x00),5dip)
    cnvgrafico.DrawLine(0,1*Panel1.height,Panel1.width,1*Panel1.height,Colors.RGB(0x00,0x00,0x00),5dip)
    cnvgrafico.DrawLine(0*Panel1.Width,0*Panel1.height,0*Panel1.width,1*Panel1.height,Colors.RGB(0x00,0x00,0x00),5dip)
    cnvgrafico.DrawLine(1*Panel1.Width,0*Panel1.height,1*Panel1.width,1*Panel1.height,Colors.RGB(0x00,0x00,0x00),5dip)
    Panel1.Invalidate
End Sub

'Arma cuadrícula para gráfico tipo 2
Sub bordes
    cnvscanb.DrawColor(Colors.Transparent)
'Grilla vertical
    cnvscanb.DrawLine(0.1*pnlscanb.Width,0*pnlscanb.height,0.1*pnlscanb.width,1*pnlscanb.height,Colors.Gray,1dip)
    cnvscanb.DrawLine(0.2*pnlscanb.Width,0*pnlscanb.height,0.2*pnlscanb.width,1*pnlscanb.height,Colors.Gray,1dip)
    cnvscanb.DrawLine(0.3*pnlscanb.Width,0*pnlscanb.height,0.3*pnlscanb.width,1*pnlscanb.height,Colors.Gray,1dip)
    cnvscanb.DrawLine(0.4*pnlscanb.Width,0*pnlscanb.height,0.4*pnlscanb.width,1*pnlscanb.height,Colors.Gray,1dip)
    cnvscanb.DrawLine(0.5*pnlscanb.Width,0*pnlscanb.height,0.5*pnlscanb.width,1*pnlscanb.height,Colors.Gray,1dip)
    cnvscanb.DrawLine(0.6*pnlscanb.Width,0*pnlscanb.height,0.6*pnlscanb.width,1*pnlscanb.height,Colors.Gray,1dip)
    cnvscanb.DrawLine(0.7*pnlscanb.Width,0*pnlscanb.height,0.7*pnlscanb.width,1*pnlscanb.height,Colors.Gray,1dip)
    cnvscanb.DrawLine(0.8*pnlscanb.Width,0*pnlscanb.height,0.8*pnlscanb.width,1*pnlscanb.height,Colors.Gray,1dip)
    cnvscanb.DrawLine(0.9*pnlscanb.Width,0*pnlscanb.height,0.9*pnlscanb.width,1*pnlscanb.height,Colors.Gray,1dip)
'Bordes
    cnvscanb.DrawLine(0,0*pnlscanb.height,pnlscanb.width,0*pnlscanb.height,Colors.RGB(0x00,0x00,0x00),5dip)
    cnvscanb.DrawLine(0,1*pnlscanb.height,pnlscanb.width,1*pnlscanb.height,Col
```

```

ors.RGB(0x00,0x00,0x00),5dip)
    cnvscanb.DrawLine(0*pnlscanb.Width,0*pnlscanb.height,0*pnlscanb.width,1*pn
lscanb.height,Colors.RGB(0x00,0x00,0x00),5dip)
    cnvscanb.DrawLine(1*pnlscanb.Width,0*pnlscanb.height,1*pnlscanb.width,1*pn
lscanb.height,Colors.RGB(0x00,0x00,0x00),5dip)
    pnlscanb.Invalidate
End Sub

'Enciende la ecosonda
Sub btn3_click
    rxtx.ast.Write("sta" & Chr(10))
    btn3.Color = Colors.Green
    btn4.Color = Colors.Red
    pt1.TextColor = Colors.Red
    ptc.TextColor = Colors.Green
    frec1.TextColor = Colors.Green
    frec2.TextColor = Colors.Red
    frec3.TextColor = Colors.Red
End Sub

'Detiene la ecosonda
Sub btn4_click
    rxtx.ast.Write("stp" & Chr(10))
    btn3.Color = Colors.Transparent
    btn4.Color = Colors.Transparent
End Sub

'Procesamiento de datos recibidos
Sub ast_NewText(Text As String)
    Dim distancia As Long
    Dim potencia As Long

    recibidos = Text
    If Text.Length <> 0 Then
'Cada 100 transmisiones se grafica una linea vertical
        If ((i*tamaño) Mod 100 = 0) Then

            cnvgrafico.DrawLine(i,0,i,Panel1.Height,Colors.Black,1dip)
            End If
            If i <= 1 Then
                i = Panel1.Width
            End If
'La llegada del string "nl" indica el comienzo de una nueva transmisión

            If recibidos.CompareTo("nl") = 0 Then
'Muestra en pantalla la distancia al eco de mayor potencia(eco de fondo) en cm

                lblfondo.Text = fondo_distancia
                fondo_potencia = 0
                j = j + 1
                i = i - tamaño
                tx = tx + 1
                If grafico = 1 Then
                    Panel1.Invalidate
' BORRA LA COLUMNA DONDE SE VA A GRAFICAR

                cnvgrafico.DrawLine(i,0,i,Panel1.Height,Colors.Transparent,tamaño)
' VUELVE A GRAFICAR EL PEDAZO DE CUADRICULA BORRADO ANTERIORMENTE

                cnvgrafico.DrawLine(i,0.1*Panel1.Height,i+tamaño,0.1*Panel1.Height,Colors.

```

```

Black, 1dip)

    cnvgrafico.DrawLine(i, 0.2*Panel1.Height, i+tamaño, 0.2*Panel1.Height, Colors.
Black, 1dip)

    cnvgrafico.DrawLine(i, 0.3*Panel1.Height, i+tamaño, 0.3*Panel1.Height, Colors.
Black, 1dip)

    cnvgrafico.DrawLine(i, 0.4*Panel1.Height, i+tamaño, 0.4*Panel1.Height, Colors.
Black, 1dip)

    cnvgrafico.DrawLine(i, 0.5*Panel1.Height, i+tamaño, 0.5*Panel1.Height, Colors.
Black, 1dip)

    cnvgrafico.DrawLine(i, 0.6*Panel1.Height, i+tamaño, 0.6*Panel1.Height, Colors.
Black, 1dip)

    cnvgrafico.DrawLine(i, 0.7*Panel1.Height, i+tamaño, 0.7*Panel1.Height, Colors.
Black, 1dip)

    cnvgrafico.DrawLine(i, 0.8*Panel1.Height, i+tamaño, 0.8*Panel1.Height, Colors.
Black, 1dip)

    cnvgrafico.DrawLine(i, 0.9*Panel1.Height, i+tamaño, 0.9*Panel1.Height, Colors.
Black, 1dip)

    cnvgrafico.DrawLine(0, 0*Panel1.height, Panel1.width, 0*Panel1.height, Colors.
RGB(0x00, 0x00, 0x00), 5dip)

    cnvgrafico.DrawLine(0, 1*Panel1.height, Panel1.width, 1*Panel1.height, Colors.
RGB(0x00, 0x00, 0x00), 5dip)
        Else If grafico = 2 Then
            If j > jmax Then
                bordes
                j = 0
            End If
        End If
    Else
'Separa de los datos recibidos los valores de distancia y potencia, expresados
en useg
        distancia = recibidos.SubString2(0, 6)
        potencia = recibidos.SubString(7)
'Convierte la distancia a cm
        distancia = ((distancia*1500*100)/2000000)
        If distancia < escala Then
            If guardando = True Then

                TextWriter1.WriteLine(DateTime.Time(DateTime.Now) & ";" & distancia & ";"
& potencia & ";" & tx)
            End If
'Busca el eco de mayor potencia
            If potencia > fondo_potencia Then
                fondo_potencia = potencia
                fondo_distancia = distancia
            End If
            If grafico = 1 Then
'Adecua los valores de distancia de acuerdo a la escala seleccionada en el
gráfico tipo 1
                distancia = (distancia *
Panel1.Height)/escala
                potencia = (potencia * 255)/pmax

```



```

        If potencia > 255 Then
            potencia = 255
        End If
        If potencia > minimo Then

            cnvgrafico.DrawCircle(i,distancia,tamaño,Colors.RGB(255,255-
potencia,0),True,1dip)

                End If
            Else If grafico = 2 Then
                'Adecua los valores de distancia de acuerdo a la escala seleccionada en el
                gráfico tipo 2
                distancia = (distancia *
pnlscanb.Width)/escala
                potencia = (potencia * pnlscanb.Height)/pmax
                If potencia > pnlscanb.Height Then
                    potencia = pnlscanb.Height
                End If

                cnvscanb.DrawLine(distancia,pnlscanb.Height,distancia,pnlscanb.Height -
potencia,Colors.Red,tamaño)
                    pnlscanb.Invalidate
                End If
            End If
        End If
    End Sub

Sub ast_Terminated
    Log("Connection terminated")
End Sub

'Selecciona la escala de distancia deseada
Sub Rbtn50_CheckedChange(Checked As Boolean)
    If Checked = True Then
        escala = 500
        lbl1.Text = "5 metros "
        grilla
        bordes
    End If
    Rbtn50.TextColor = Colors.Green
    Rbtn20.TextColor = Colors.Red
    Rbtn10.TextColor = Colors.Red
End Sub

Sub Rbtn20_CheckedChange(Checked As Boolean)
    If Checked = True Then
        escala = 200
        lbl1.Text = "2 metros "
        grilla
        bordes
    End If
    Rbtn20.TextColor = Colors.Green
    Rbtn50.TextColor = Colors.Red
    Rbtn10.TextColor = Colors.Red
End Sub

Sub Rbtn10_CheckedChange(Checked As Boolean)
    If Checked = True Then
        escala = 100
        lbl1.Text = "1 metro "
        grilla
        bordes
    End If

```

```

Rbtn10.TextColor = Colors.Green
Rbtn20.TextColor = Colors.Red
Rbtn50.TextColor = Colors.Red
End Sub

'Selecciona la frecuencia de repetición deseada
Sub frec3_CheckedChange(Checked As Boolean)
    rxtx.ast.Write("fra" & Chr(10))
    frec3.TextColor = Colors.Green
    frec2.TextColor = Colors.Red
    frec1.TextColor = Colors.Red
End Sub
Sub frec2_CheckedChange(Checked As Boolean)
    rxtx.ast.Write("frm" & Chr(10))
    frec2.TextColor = Colors.Green
    frec1.TextColor = Colors.Red
    frec3.TextColor = Colors.Red
End Sub
Sub frec1_CheckedChange(Checked As Boolean)
    rxtx.ast.Write("frb" & Chr(10))
    frec1.TextColor = Colors.Green
    frec2.TextColor = Colors.Red
    frec3.TextColor = Colors.Red
End Sub

'Selecciona el tamaño del punto a graficar
Sub tmpts_ValueChanged(Value As Int, UserChanged As Boolean)
    If Value <= 6 Then
        tamaño = 3
        lbl3.Text = 6
    Else
        tamaño = Value/2
        lbl3.Text = Value
    End If
End Sub

'Setea la potencia mínima necesaria de un eco para ser graficado
Sub filtro_ValueChanged(Value As Int, UserChanged As Boolean)
    minimo = Value
    lbl2.Text = Value
End Sub

'Selecciona el ancho del pulso de transmisión
Sub ptl_CheckedChange(Checked As Boolean)
    rxtx.ast.Write("ptl" & Chr(10))
    ptl.TextColor = Colors.Green
    ptc.TextColor = Colors.Red
End Sub
Sub ptc_CheckedChange(Checked As Boolean)
    rxtx.ast.Write("ptc" & Chr(10))
    ptc.TextColor = Colors.Green
    ptl.TextColor = Colors.Red
End Sub

'Selecciona el tipo de gráfico
Sub grafico2_CheckedChange(Checked As Boolean)
    Panell.Visible = False
    pnlscanb.Visible = True
    grafico = 2
    persistencia.Enabled = True
    grafico2.TextColor = Colors.Green

```

```

        grafico1.TextColor = Colors.Red
    bordes
End Sub
Sub grafico1_CheckedChange(Checked As Boolean)
    Panell1.Visible = True
    pnlscanb.Visible = False
    grafico = 1
    persistencia.Enabled = False
    grafico1.TextColor = Colors.Green
    grafico2.TextColor = Colors.Red
    grilla
End Sub

'Setea la ganancia
Sub ganancia_ValueChanged (Value As Int, UserChanged As Boolean)
    pmax = pmax - Value
    lblganancia.Text = Value
End Sub

'Setea la persistencia en el gráfico tipo 2
Sub persistencia_ValueChanged (Value As Int, UserChanged As Boolean)
    jmax = Value
End Sub

'Setea cuando comenzar y terminar de guardar los datos en un archivo de texto
Sub guardar_click
    If File.ExternalWritable = False Then
        MsgBox("Imposible escribir en la memoria.", "Error")
    Else
        guardando = True
        Dim fecha As String
        fecha = DateTime.GetDayOfMonth(DateTime.Now) &
DateTime.GetMonth(DateTime.Now) & DateTime.GetYear(DateTime.Now) &
DateTime.GetHour(DateTime.Now) & DateTime.GetMinute(DateTime.Now)
        TextWriter1.Initialize(File.OpenOutput(File.DirRootExternal, fecha &
".txt", True))
        TextWriter1.WriteLine("Hora ; distancia en cm ; potencia ; pulso de
tx al que pertenece el dato ")
    End If
End Sub
Sub terminar_click
    TextWriter1.Close
    guardando = False
End Sub

```

# APÉNDICE C

## Hojas de Datos

## Ultrasonic Sonar Ranging IC - PW0268



SSOP20

### Features:

- Operating Voltage: 6 – 10Vdc single source
- Operating Frequency: broadband output ranging up to 250KHz
- Variable R/C Oscillator: compensates for transducer resonate frequency drift due to temperature.
- High Gain Amplifier: varies with time over 32 steps
- Integrated Band Pass Filter: reduces external component count
- Bi-direction I/O Pin: simplifies the control function for transmitting a pulse and receiving an echo
- An adjustable System Clock: enables the control of, the number of pulses transmitted, the slope of the variable gain amplifier, and the pulse repetition rate.
- The PW0268 IC is suitable for use in car reversing aids, electronic tape measures and other sonar ranging applications.

### Description:

The PW-0268 ultrasonic sonar ranging IC is ideally suited for echo ranging systems. This chip has many design features to enhance its performance and ease of use in this application.

The externally tunable RC Oscillator automatically tracks and compensates for the shift of the resonate frequency of the transducer due to temperature changes.

The Fix Gain Preamplifier can be tailored to compensate for varying transducer sensitivities. The 32-step Time Controlled Variable Gain Amplifier slope can be modified by adjusting the frequency of the system clock.

An onboard Comparator converts the analog signal of the returning echo to a TTL level digital signal for use with an external microprocessor.

The integrated Band Pass Filters can be adjusted for custom applications by changing a few external components. The frequency of the System Clock can be adjusted to control other operating parameters of the chip including the transmit pulse width and sample rate.

The I<sub>O</sub> pin, (pin 1) is a bi-direction pin and is designed as an open collector connection with an internal pull high resistor. When the I<sub>O</sub> pin is being pulled low by an external transistor, the RC oscillator generates a tone burst signal at DRIVER<sub>0</sub> (pin 11), the output driver stage for the transducer. After the transmit pulse, the I<sub>O</sub> pin, (pin 1) will again go low if a valid echo signal is detected.

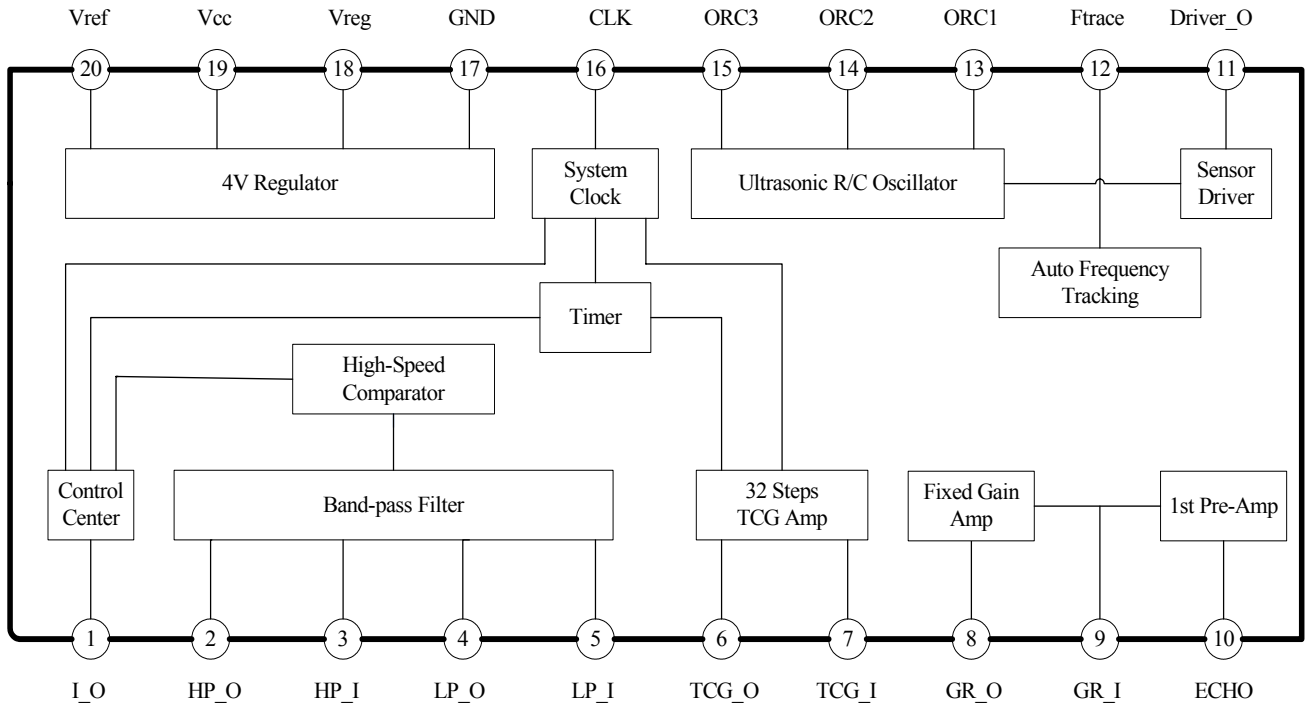
The reflected echo signal is presented to the first stage pre-amplifier through ECHO (pin 10). The gain of pre-amplifier can be adjusted to accommodate transducers with varying sensitivities by changing an external resistor between ECHO (pin 10) and GR<sub>I</sub> (pin 9).

The 32 steps time controlled variable gain amplifier input TCG<sub>I</sub> (pin7) and output TCG<sub>O</sub> (pin 6) is synchronized to start incrementing at the end of control pulse signal I-O, (pin 1) and is reset at the beginning of the next control pulse.

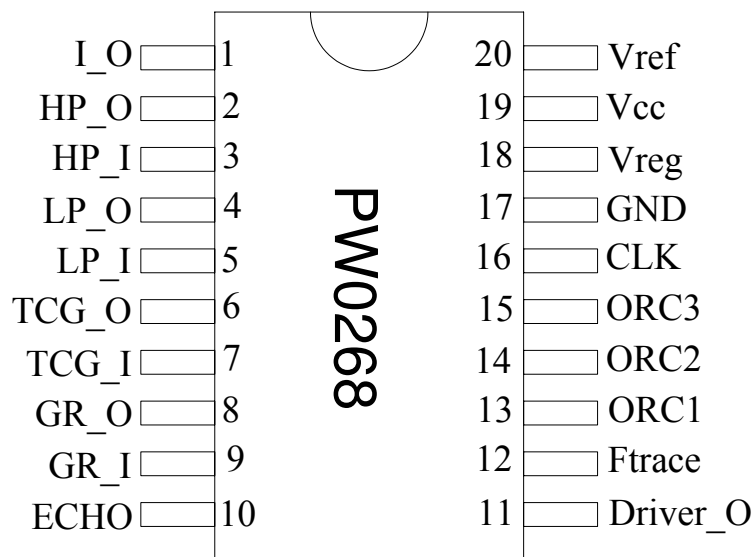
Only a few passive components are needed for the active band pass filter. There are two stages, a low pass, LP<sub>I</sub> (pin 5) and LP<sub>O</sub> (pin 4) and upper band pass, HP<sub>I</sub> (pin 3) to HP<sub>O</sub> (pin 2). The center frequency and bandwidth of the filter are chosen based on the type of ultrasonic transducer being used and the specific application. The amplified echo signal after being filtered is routed to a comparator, which shapes and converts the analog echo signal into digital signal outputted at I<sub>O</sub> (pin 1) for further  $\mu$ P handling.

The unique temperature compensating Ultrasonic R/C Oscillator circuitry tracks the resonant frequency drift of the transducer that is caused by environment temperature changes. Simply adding dual diodes and one resistor between DRIVER<sub>O</sub> (pin 11) and Ftrace (pin 12) is all that is needed to complete this function.

## Block Diagram



## Pin Assignment



### Specifications:

Unless otherwise specified, all data measured under  $V_{cc} = 9V$ ,  $F = 40KHz$

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Supply Voltage	$V_{cc}$	$V_{reg} = 4V$	5.5		11	V
Supply Current	$I_{cc}$	$V_{cc} = 6 \sim 10V$	8	11	14	mA
Regulated Voltage	$V_{reg}$	$V_{cc} = 6 \sim 10V$	3.8	4	5	V
Stability of $V_{reg}$	$V_{lr}$	$V_{cc} = 6 \sim 10V, \pm 3\%$	-3.0	0	+3.0	%
Reference Voltage	$V_{ref}$	$V_{cc} = 6 \sim 10V, R_L > 2K\Omega$	0.4	0.44	0.5	$V_{reg}$
Op-Amp Slew Rate	SR	$V_{in} = 3V_{pp}$	5	-	-	$V/\mu S$
Comparator Trigger Level	$T_{comp}$	Over $V_{ref}$	300	350	400	mV
System Clock Frequency	CLKf	$R=33K\Omega, C=22pF$	610	660	710	KHz
System Clock Frequency Range	CLKr		0.001	-	1500	KHz
Ultrasonic Oscillation Frequency	Fosc f	$R=5.6K, C=1000pF$	38	40	42	KHz
Ultrasonic Oscillation Frequency Range	Fosc r		0.001	-	500	KHz
2 <sup>nd</sup> Amp Gain	GR		29	30	31	dB
Time Controlled Gain Amplifier	TCGain	Min(1x, 0dB)	-1	0	+1	dB
		Max(58x, 35.2dB)	34	35	36	dB
Bandwidth of 2nd Amp	GRbw	Gain = 50dB	150	170	200	KHz
Driving Current	$I_{drv}$	Driver $\bar{O}$	-	20	40	mA
	$I_{sink}$	Driver $\bar{O}$	-	-20	-80	
Input Voltage Level	$I_{OVIH}$		-	0.3	0.4	$V_{cc}$
	$I_{OVIL}$		0.15	0.2	-	
Output Voltage Level	$I_{OVOH}$		-	0.9	1	$V_{cc}$
	$I_{OVOL}$		0	0.05	-	
Input Low Level Current	$I_{OIOL}$		-	-10	-20	mA
$I_{O}$ Internal Pull Up Resistance	$R_{up}$		3.5	5	6.5	$K\Omega$

### Absolute Maximum Ratings

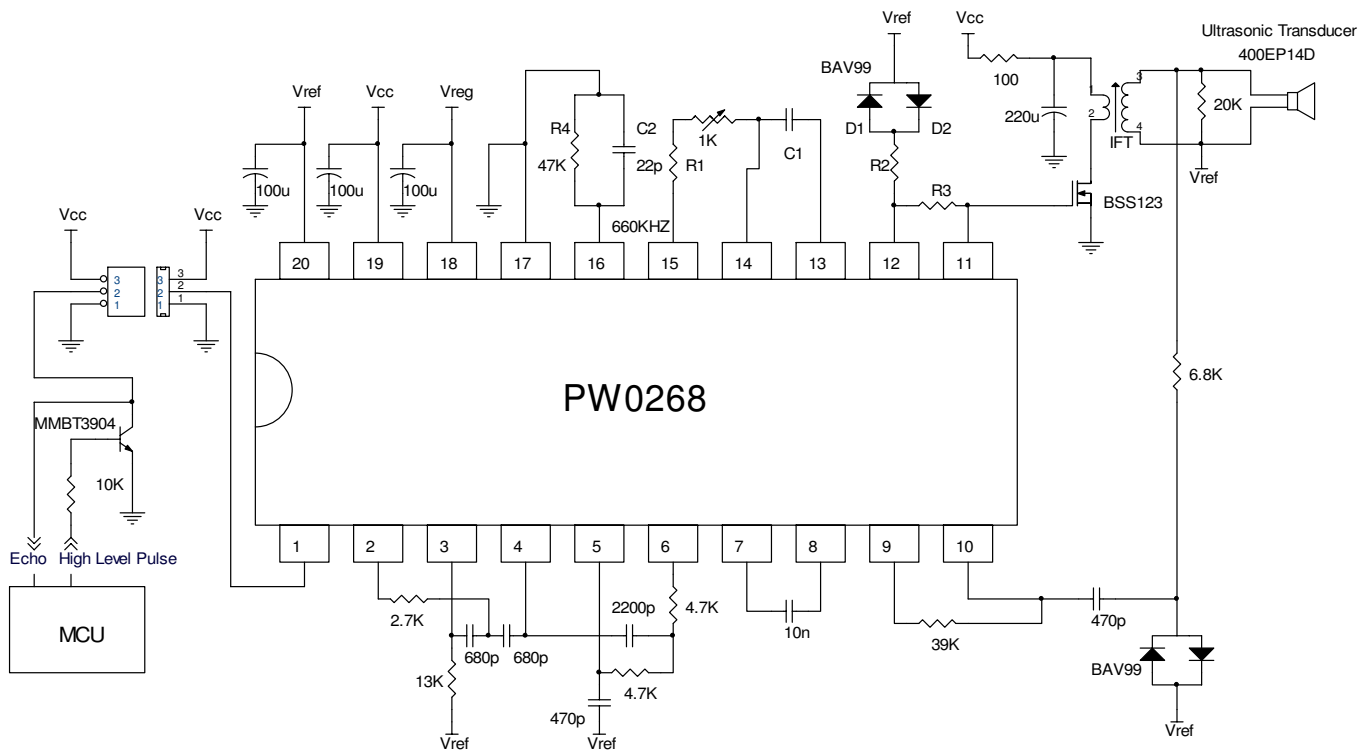
Description	Symbol	Condition	Min.	Max.	Unit
Supply Voltage	$V_{cc}$		0	12	V
Operation Temperature	$T_{opr}$		-40	+85	$^{\circ}C$
Storage Temperature	$T_{stg}$		-65	+150	$^{\circ}C$
Max. Pin Input Voltage	$V_{imax}$	$I_{O}, V_{cc}$	-0.3	$V_{cc}+0.3$	V
		Others	-0.3	$V_{reg}+0.3$	
Max. Input Current	$I_{imax}$	*	-10	+10	mA

\*To prevent latch up, the instantaneous input current should be no large than 100mA for each pins.

**Pins Description:**

Pin	Name	Description	Pin	Name	Description
1	I O	Input/Output	11	Driver_O	Transducer driving output
2	HP_O	High pass filter output	12	Ftrace	Frequency tracing input
3	HP_I	High pass filter input	13	ORC1	RC oscillator: terminal 1
4	LP_O	Low pass filter output	14	ORC2	RC oscillator: terminal 2
5	LP_I	Low pass filter input	15	ORC3	RC oscillator: terminal 3
6	TCG_O	Time controlled gain output	16	CLK	System clock
7	TCG_I	Time controlled gain input	17	GND	Ground
8	GR_O	External adjustable gain output	18	Vreg	Regulated voltage for internal analogue devices
9	GR_I	External adjustable gain input	19	Vcc	Power supply
10	ECHO	Receiving echo input	20	Vref	Reference voltage output

**Application Circuit:** for car reversing aids (values should be changed for other applications)





## Application Note

The circuit shown on page 4 is a typical circuit for car reversing aids. The RC Oscillator generates a tone burst when a low level pulse is applied to the I\_O pin, (pin 1).

To accommodate tolerance variations of transducers during manufacturing, a 1K-ohm variable resistor (R1) is provided to trim the output operating frequency. The range of adjustment is from 38.0 – 42.0 KHz. and allows for a better match of the drive signal to the resonate frequency of the transducer.

The active burst number (number of pulses transmitted) is controlled by the pulse width of the low level signal applied to the I\_O pin, (pin 1).

The tone burst output, Driver\_O (pin 11), drives the transducer through a MOSFET transistor and impedance matching transformer IFT. The inductance in the secondary winding of the transformer is designed to tune out the reactance of the parallel capacitance of the transducer.

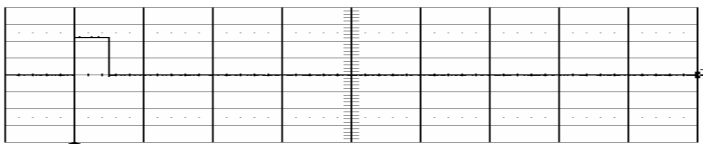
The high output voltage of the tone burst is snubbed by two diodes and the returning echo signal is passed on to the first stage pre-amplifier. The signal is then passed on to the second stage fix gain amplifier and finally to the third stage 32-step time controlled variable gain amplifier. The gain of the pre-amplifier should be properly set to meet the sensitivity needs of the transducer and application requirements.

The center frequency of band-pass filter should be chosen to exactly match the frequency of the RC Oscillator and considerations for the width of pass-band filter should be made based on actual application requirements.

If the amplified echo signal from the output of the band pass filter exceeds  $0.35V + V_{ref}$ , the comparator will output a low pulse to the I\_O pin, (pin 1). The width of the low level pulse is proportional to the echo signal strength.

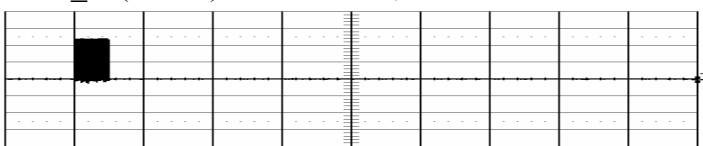
The above description is summarized in the signal timing charts illustrated below.

MCU output: H: 1ms/Div., V: 2.0V/Div.



The RC oscillator will be enabled in the duration of input pulse. The maximum pulse width is  $396/F$  and any time longer than this upper limit will be ignored.

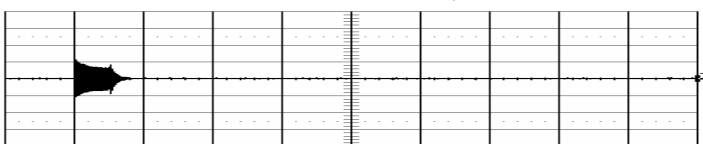
Driver\_O (Pin 11): H: 1ms/Div., V: 2.0V/Div.



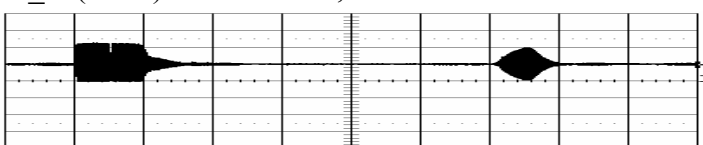
The next input pulse will be ignored if the pulse repetition rate is shorter than  $9900/F + \text{pulse width}$ .

F: Frequency of system clock

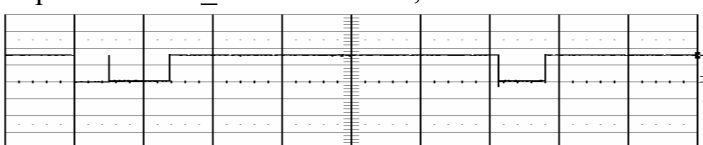
Transducer Oscillation: H: 1ms/Div., V: 50V/Div.



HP\_O (Pin 2): H: 1ms/Div., V: 2.0V/Div.



Output at Pin 1 I\_O: H: 1ms/Div., V: 5.0V/Div.



The resonate frequency of ultrasonic piezo transducers varies in an inversely proportional relationship to temperature. The lower the temperature, the higher the resonate frequency, the higher the temperature the lower the resonate frequency.

This property of piezo transducers may cause a mismatch between transducer and drive signal with changes in ambient temperature and reduces efficiency of the system when the frequency of the drive circuit remains constant and does not track the resonate frequency shift of the transducer.

Therefore it is desirable to have the output frequency of the drive source track the resonate frequency of the transducer with changes in ambient temperature. The Auto Frequency Tracking circuitry between Ftrace, (pin 12), and Drive\_O, (pin 11) is used to accomplish this task.

The voltage change at Ftrace (pin 11) varies in proportion to the forward bias voltage change across diodes D1, D2. This change is caused by the negative temperature coefficient of the diodes and the ratio of the resistor circuit R2/R3.

A lower temperature increases the voltage drop across the diodes. This intern accelerates the charge rate of an internal integrator circuit controlling the R/C Oscillator, ORC3, (pin15). The net result is the adjustment to the R/C Oscillator increases the resonate frequency of the output, Drive\_O, (pin11).

Conversely, a higher temperature decreases the voltage drop across the diodes. This slows the charge rate of the internal integrator circuit controlling the R/C Oscillator. The net result of this adjustment is to decrease the resonate frequency of the output Drive\_O, (pin 11).

Choose values for the components R1, R2, R3 and C1 that will best track the characteristic resonate frequency shift curves due to temperature for a specific transducer.

Recommended values for the following transducers are listed below.

Used Transducer	R1(Ohm)	R2(Ohm)	R3(Ohm)	C1(pF)
400EP14D	3,300	1,500	511	2,200
400EP18A	3,300	1,500	604	2,200
235AC130	2,000	0	2,100	220

For a fixed output of 40KHz at Drive\_O (pin 11) simply remove D1, D2 and R2 and set R1 = 4,500 Ohm, C1 = 2,200 pF, and R3 = 511 Ohm.

The system clock CLK (pin 16) controls the maximum input pulse width, the slope of time controlled gain amplifier and pulse repetition rate.

For example, as illustrated in the block diagram, if the system clock is set to 660KHz (C2: 22pF, R4: 47K Ohm), then:

- (1) The maximum input pulse width is  $396/F = 396/660K = 0.6$  ms and any duration longer than 0.6ms will be ignored.
- (2) The step duration of the 32 step time controlled gain amplifier is equal to  $220/F = 0.333$  ms, starting from the end of the pulse on the I\_O pin, (pin 1).
- (3) The minimum pulse repetition rate is  $9900/F + \text{pulse width} = 9900/F + 0.5$  ms (20 bursts of 40KHz) =  $9900/660K + 0.5 = 15.5$  ms.

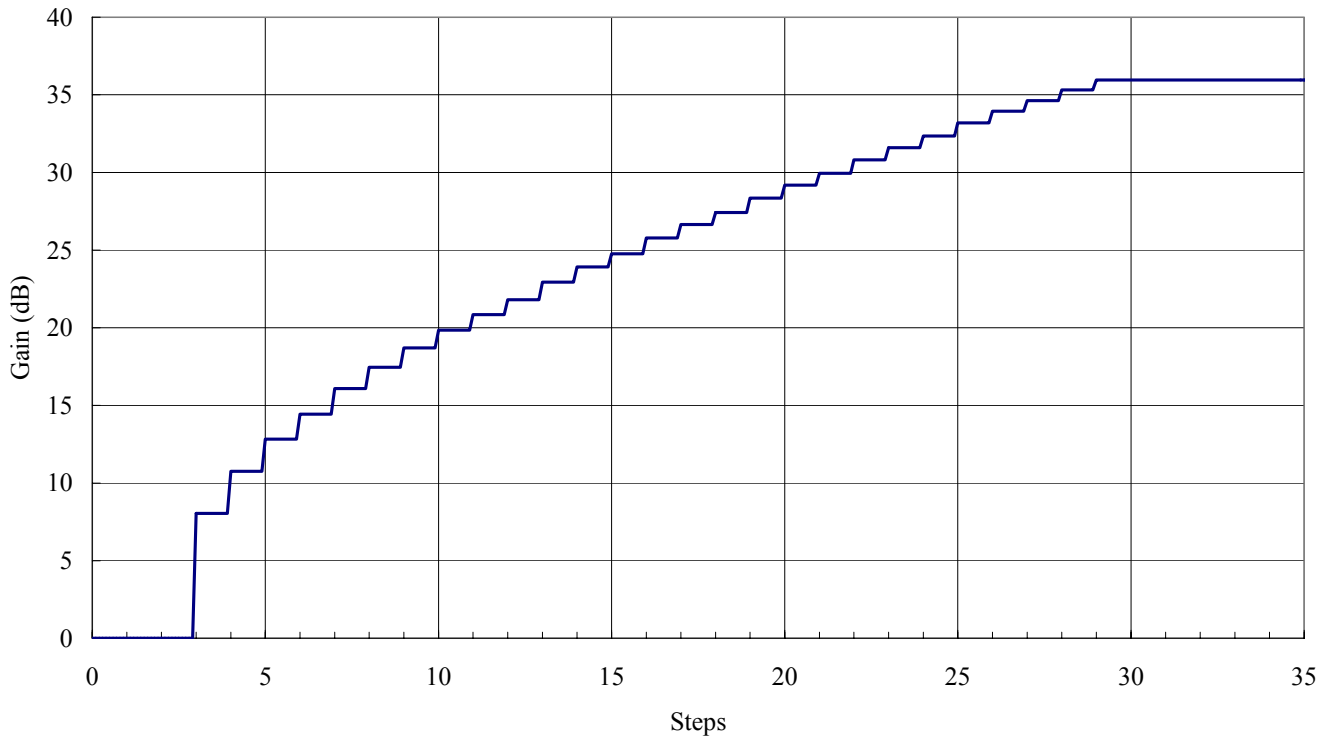
For long distance measurements of 18 meters (one way distance), the system clock should be set as follows:

Min. Pulse Repetition Rate =  $9900/F + 0.75 = 166$  mS (30 bursts of 40KHz)

Frequency of System Clock F = 60 KHz

For additional information about an 18-meter tape measure circuit, please consult with the factory.

### Time Controlled Gain Amplifier

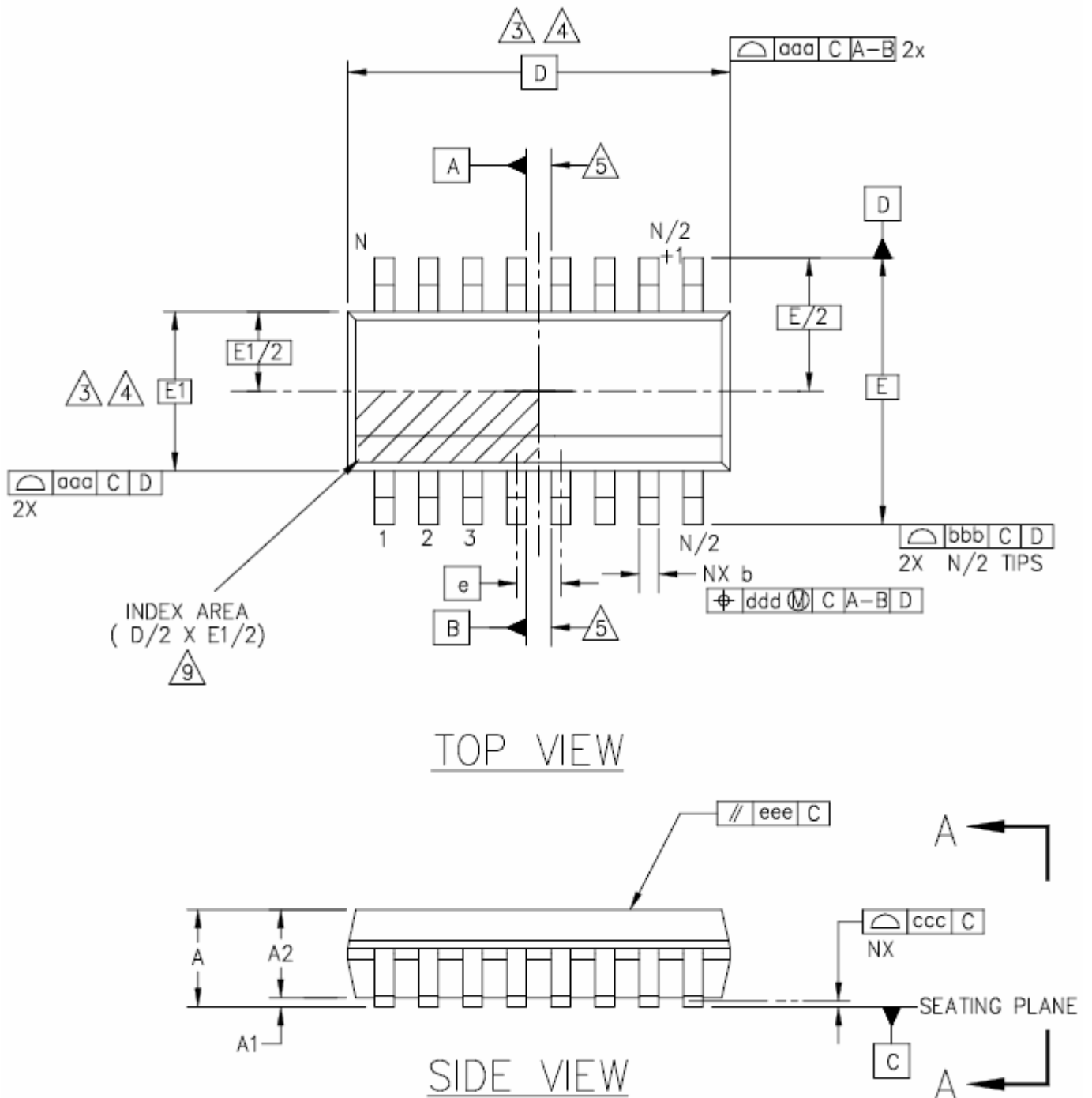


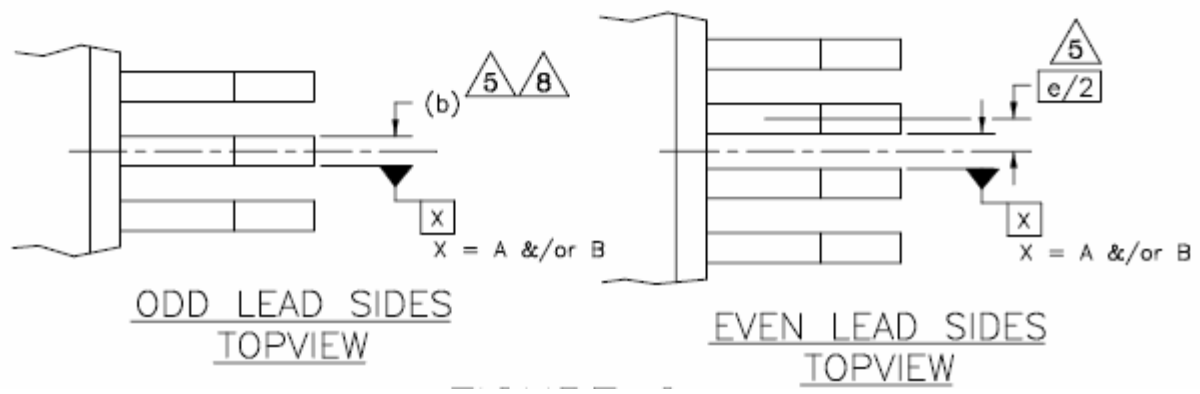
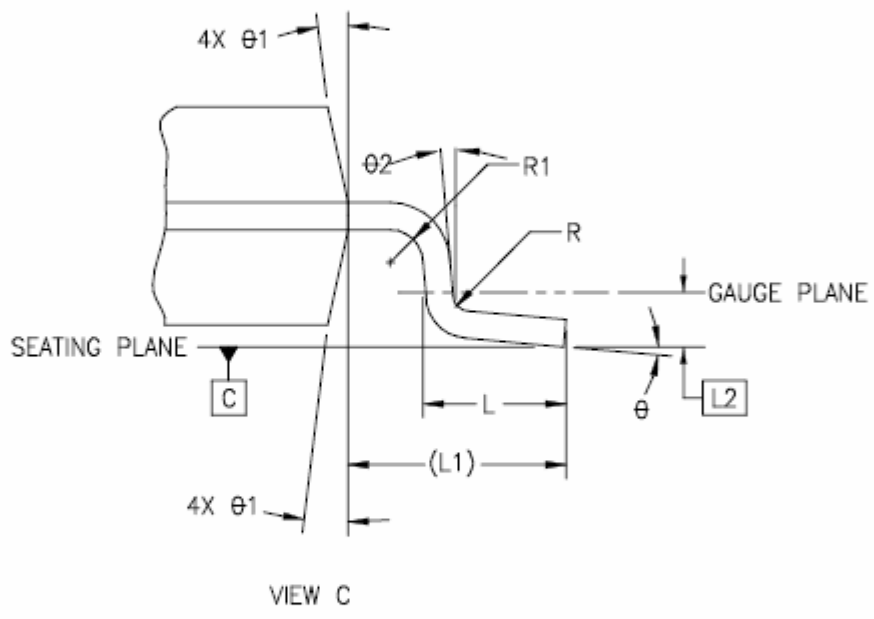
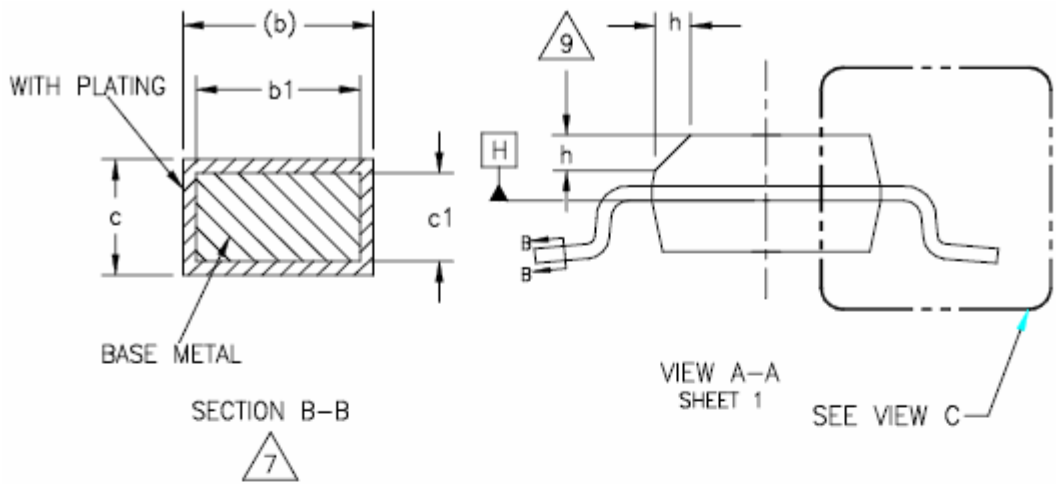
The time controlled gain amplifier is stepping up once the input pulse falling. The time duration can be calculated as:

$$T = 220/F \quad \text{F: Frequency of System Clock}$$

**Package and Dimensions:**

20 Pins, SSOP (150mil)





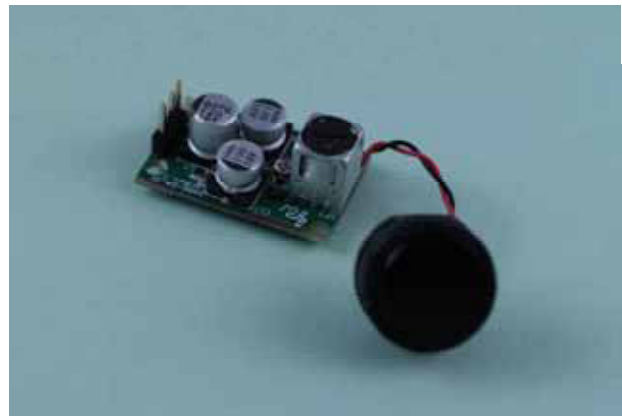
Symbol	Min.	Nom.	Max.
A	0.053	-	0.069
A1	0.004	-	0.010
A2	0.049	-	0.065
b	0.008	-	0.012
b1	0.008	0.010	0.011
c	0.006	-	0.010
c1	0.006	0.008	0.009
D	0.341 BSC		
E	0.236 BSC		
E1	0.154 BSC		
e	0.025 BAS		
L	0.016	-	0.050
L1	0.041 REF		
L2	0.010 BAS		
R	0.003	-	-
R1	0.003	-	-
$\theta$	0°	-	8°
$\theta 1$	5°	-	15°
$\theta 2$	0°	-	-
aaa	0.004		
bbb	0.008		
ccc	0.004		
ddd	0.007		
eee	0.004		

**Notes:**

1. Dimensioning and tolerancing per ANSI Y14.5M-1982.
2. Dimensions in inches (angles in degrees)
3. Dimension D does not include mold flash, protrusions or gate burrs. Mold flash, protrusions or gate burrs shall not exceed 0.006" per end. Dimension E1 does not include interlead flash or protrusions. Interlead flash or protrusions shall not exceed "0.006" per side. D1 and E1 dimensions are determined at datum H.
4. The package top may be smaller than the package bottom. Dimensions D and E1 are determined at the outermost extremes of the plastic body exclusive of mold flash, tie bar burrs, gate burrs and interlead flash, but including any mismatch between the top and bottom of the plastic.
5. Datum A and B to be determined at datum H.
6. N is the maximum number of terminal position. (N=20)
7. The dimensions apply to the flat section of the lead between 0.004 to 0.010 inches from the lead tip.
8. Dimension b does not include dambar protrusion. Allowable dambar protrusion shall be 0.004" total in excess of b dimension at maximum material condition. The dambar can not be located on the lower radius of the foot.
9. Refer to JEDEC MO-137 variation AD.

JEDEC is the registered trademark of JEDEC SOLID STATE TECHNOLOGY ASSOCIATION

The SRM400 is a sonar ranging module utilizing our new developed Sonar Ranging IC, PW-0268, which can work with all our PT or EP type transducers. SRM400 provides as a shortcut to develop car reversing systems or some other distance measurement systems for design engineers who are not very familiar with analog circuit and/or the operation of ultrasonic transducers. By using this module engineers can focus firstly on the other fields of digital circuit and software designs as well as some other mechanical issues. After first stage then you can either design your own analog circuit based on the module construction or consult with factory for making your own module for your special needs.



**Features:**

- Operating Voltage: 6 – 10Vdc single source
- Operating Frequency: broadband output ranging up to 250KHz
- Built-in variable RC oscillator matching transducers with different frequencies
- High Gain Amplifier: varies with time over 32 steps
- Integrated Band Pass Filter: reduces external component count,
- Bi-direction I/O Pin: simplifies the control function for transmitting a pulse and receiving an echo
- An adjustable System Clock: enables the control of, the number of pulses transmitted, the slope of the variable gain amplifier, and the pulse repetition rate
- Board size: 27.9 \* 18 mm (L\*W)

**Specification:**

Operation voltage	DC6 - 10V
Operation current	<20 mA @DC10V
Oscillation frequency	Variable RC oscillator
Amplifier gain	
Pre-Amplifier	14 dB
2 <sup>nd</sup> Stage Amplifier	30 dB
Time controlled 32 steps main amplifier	35 dB max.
Bandpass filter	Fc: 38 KHz
	Bandwidth: 20KHz
	Insertion loss: 1 dB
Driving voltage (no load)	130Vpp; pulse width 0.5ms
Bi-directional I/O	
Input signal	Open collector pull low
Output	0..05*Vcc to 0.9*Vcc digital echo signals
Measuring distance	25 – 150 cm

**SRM400** includes:

1. Module board
2. 400EP14D enclosed type transducer of asymmetrical beam patterns, see detail specification of 400EP14D.
3. Detail electrical schematic



**S. Square Enterprise Company Limited**  
**Pro-Wave Electronics Corporation**

P.O. Box 1-70 Chung Ho, Taiwan, ROC; E-mail: [prowave@ms3.hinet.net](mailto:prowave@ms3.hinet.net); Tel: 886-2-22465101(5 lines), 22459774; Fax: 886-2-22465105

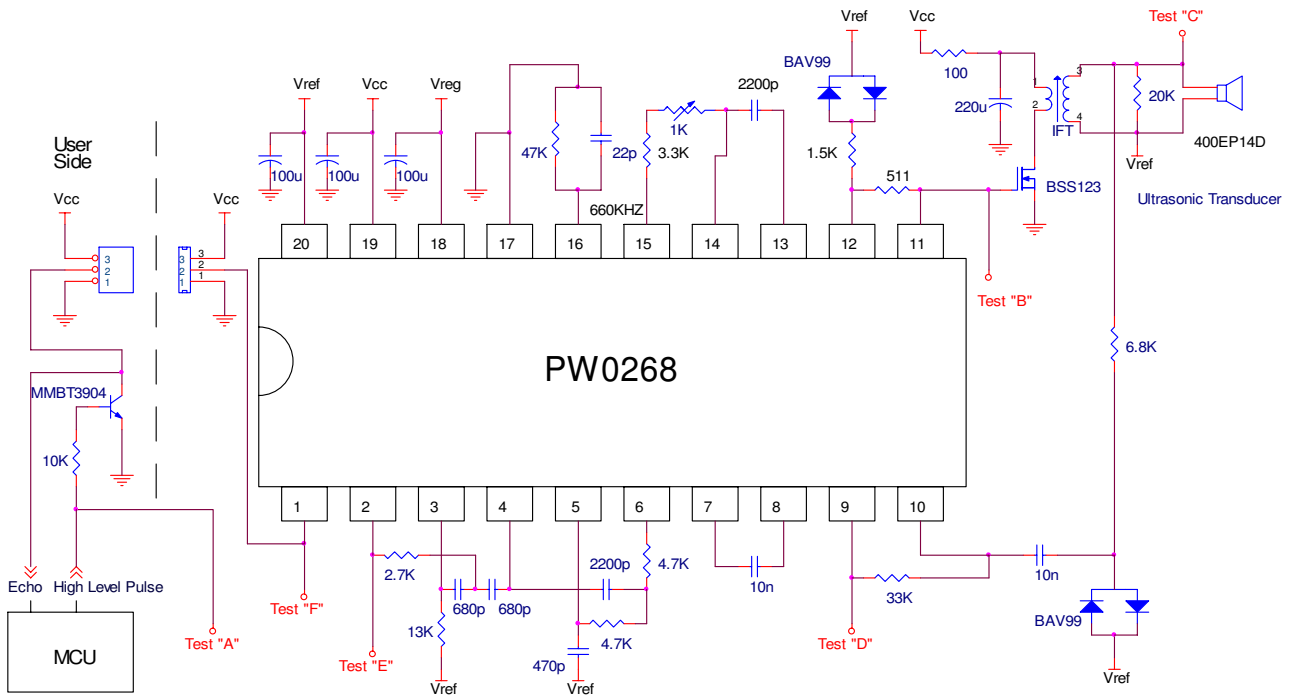
<http://www.s2.com.tw> ; <http://www.prowave.com.tw>



# Sonar Ranging Module

## Electronic Circuit Diagram

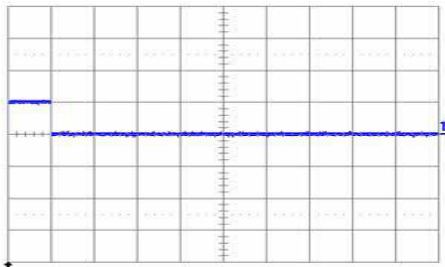
SRM400



### Waveforms at different test points:

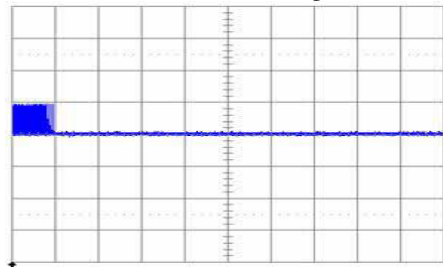
works with transducer model 400EP14D against a hard target of size of 20cmL\*20cmW\*1cmT at distance of 50cm

**“A” Point:** Control Pulse (from MCU)



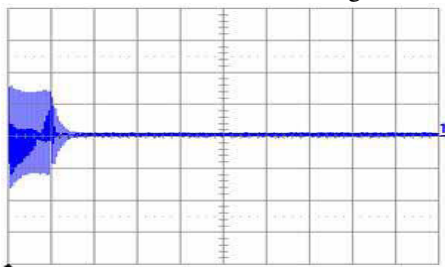
H: 0.5ms/div  
V: 5V/div

**“B” Point:** Tone bursts Signal



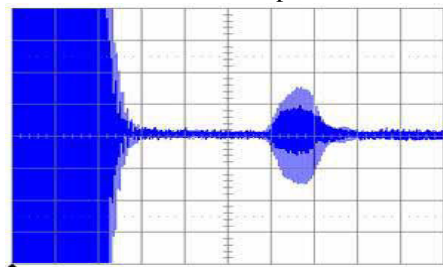
H: 0.5ms/div  
V: 5V/div

**“C” Point:** Transducer loading



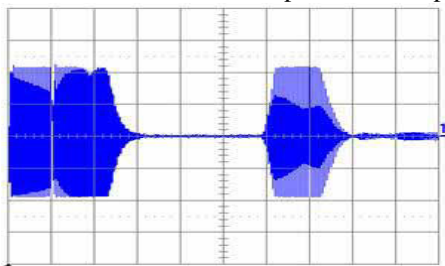
H: 0.5ms/div  
V: 50V/div

**“D” Point:** 1<sup>st</sup> Pre-Amplifier



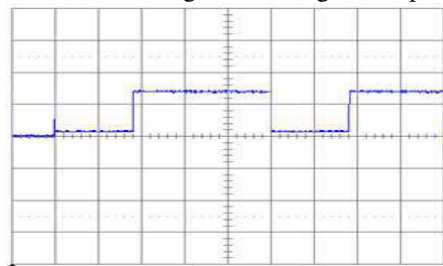
H: 0.5ms/div  
V: 20mV/div

**“E” Point:** Main 32 Steps TCG Amplifier



H: 0.5ms/div  
V: 1V/div

**“F” Point:** Digital Echo signal Output

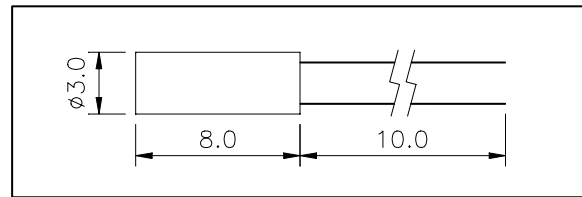


H: 0.5ms/div  
V: 5V/div

Refer to [PW-0268 Sonar Ranging IC](#) for detail information.

## Quartz Crystals & Matching Transformers

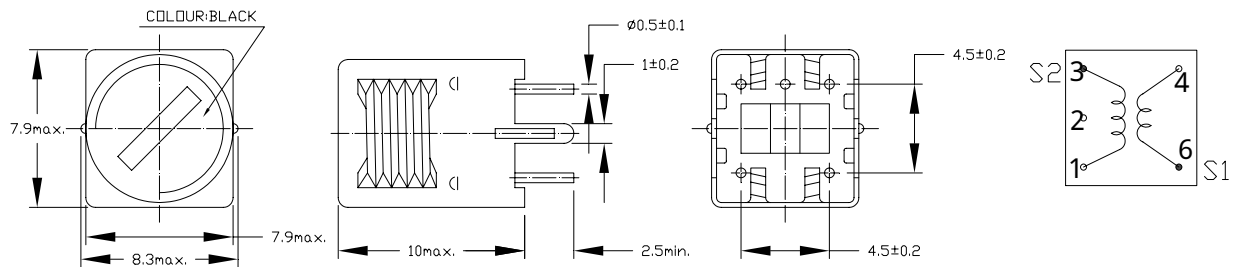
### Miniature Tuning Fork Quartz Crystals



### Specification

Model Number	Nominal Frequency Hz	Tolerance at 25°C PPM	Temperature Stability -10°C to +70°C PPM	Load Capacitance pF	Series Resistance Ohm	Shunt Capacitance pF	Drive Level mW
S40000	40,000	± 60	± 45	12.5	35,000	2.3	0.001
S32768	32,768	± 20	± 30	12.5	35,000	2.3	0.001

### Matching Transformers



### Specification

Parts Number	K4000001	K4000002	K4000003	K4000004
Operating Frequency	40.0 KHz	40.0 KHz	40.0 KHz	40.0 KHz
Variable Inductance (min.)	10.6 mH± 6%	10.6 mH± 6%	10.6 mH± 6%	10.6 mH± 6%
Unloaded Q (min.)	70	100	25	47
Turn Ratio	1:10	1:10	1:10	1:10
Matching Transducer	400EP14D	400EP14D (Temperature Compensated Type)	235SR130	400EP18A



S. Square Enterprise Company Limited  
Pro-Wave Electronics Corporation

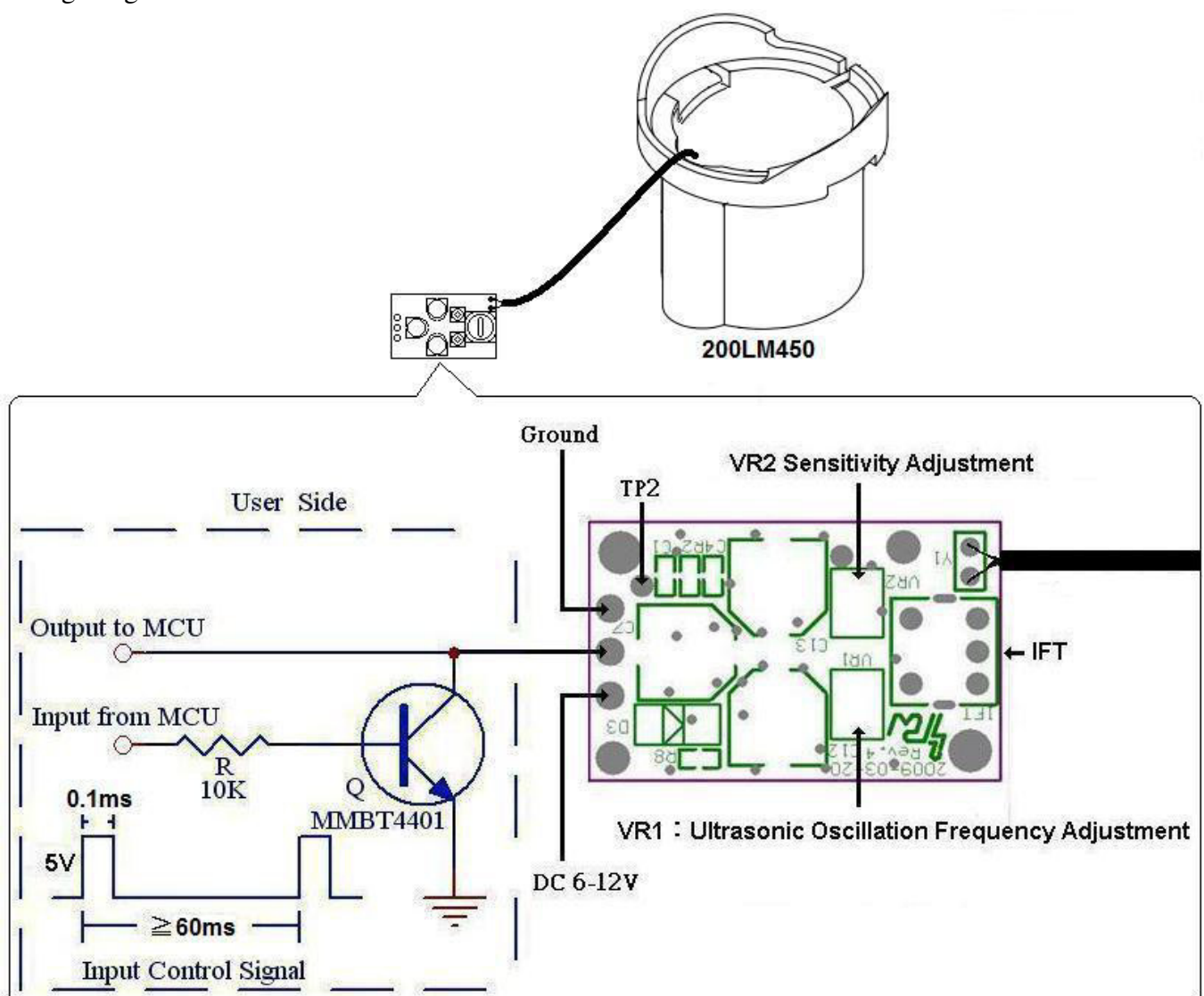
P.O. Box 1-70 Chung Ho, Taiwan, ROC; E-mail: [prowave@ms3.hinet.net](mailto:prowave@ms3.hinet.net); Tel: 886-2-22465101(5 lines), 22459774; Fax: 886-2-22465105  
<http://www.s2.com.tw>; <http://www.prowave.com.tw>

Model Number: USR2000C

Specification:

Operation voltage	DC6 - 12V
Operation current	<30 mA @ DC10V 、 pulse width 0.5ms 、 $\geq 15ms$ Repetition rate
Input signal	Open collector pull low
Output signal	0.2*Vcc to 1*Vcc digital echo signals

Wiring Diagram :



**Test Conditions:**

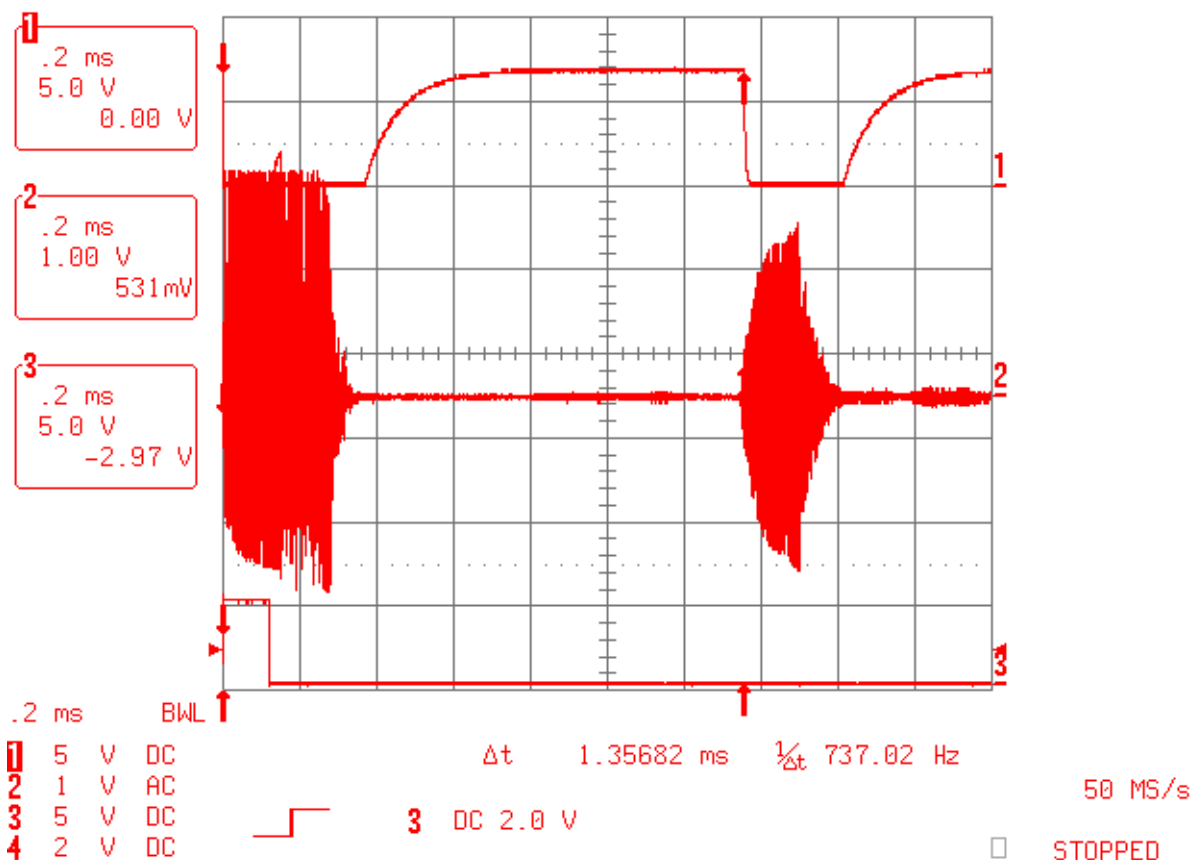
1. DC Voltage: 9V 、 Pulse Width: 0.1ms ◦
2. Reflector: 30\*30\*1cm(L\*W\*T) Aluminum Plate ◦

**Oscilloscope Test setup :**

CH1	VOLTS / DIV : DC 5V	Output to MCU
CH2	VOLTS / DIV : AC 1V	TP2
CH3	VOLTS / DIV : DC 5V	Input from MCU

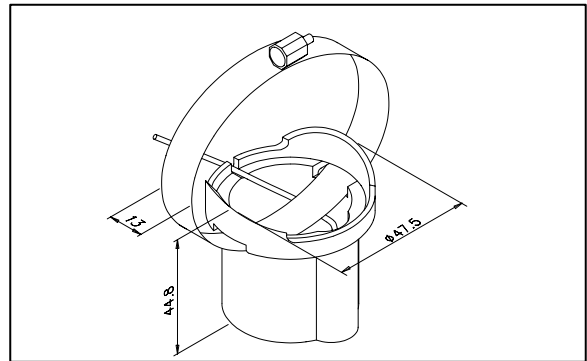
Distance: 100cm (TIME / DIV 0.2ms)

10-Feb-15  
 19:26:34





**Dimensions:** dimensions are in mm



**Specification**

<b>200LM450</b>	Transceiver
<b>Center Frequency</b>	200±10.0Khz
<b>Bandwidth (FOM -6dB)</b>	25Khz
<b>Transmitting Sound Pressure Level</b>	160dB min.
0dB re 1µPa per 1Vrms at 100cm	
<b>Receiving Sensitivity</b>	-180dB min.
0dB = 1 volt/µPa	
<b>Submerged Impedance (Ohm)</b>	200
<b>Capacitance at 1Khz ±20%</b>	2000 pF
<b>Input Power (Pulse Drive)</b>	50 Watts
<b>Total Beam Angle -6dB</b>	20°
<b>Cable Length</b>	4.5 m
<b>Molded Connector</b>	RCA Phono plug 90°
<b>Housing Material</b>	Plastic resin

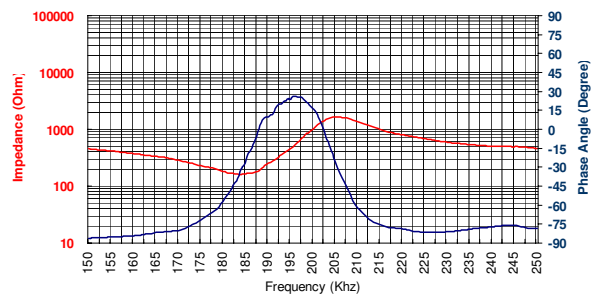
Closer frequency tolerance, shorter ringing and wider bandwidth models can be supplied upon request.

Model available:

1	200LM450	Plastic Housing
---	----------	-----------------

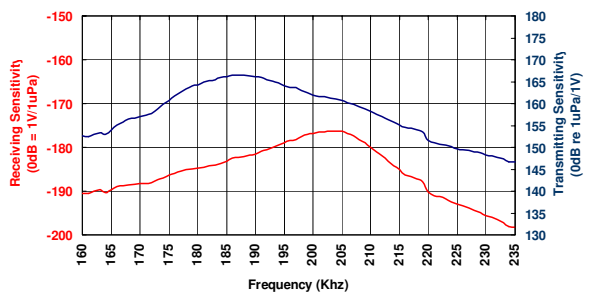
**Submerged Impedance/Phase Angle vs. Frequency**

Tested under 1Vrms Oscillation Level



**Receiving /Transmitting Sensitivity**

Tested at distance of 100cm



**Figure of Merit**

(Receiving Sensitivity + Transmitting Sensitivity)

