



Universidad Veracruzana

# MEDIDOR DE PULSO CARDIACO (Manual Técnico)

Facultad de Ingeniería  
en Instrumentación Electrónica

Microprocesadores y Microcontroladores

Violeta Margarita Fernández Salinas  
Luis Gerardo Salazar Aguilar





## Contenido

.....	0
<b>Introducción.....</b>	<b>2</b>
<b>Objetivos.....</b>	<b>3</b>
<b>Contenido técnico.....</b>	<b>4</b>
Diagramas de flujo de datos .....	4
Modelo de datos físico.....	4
Plataforma de usuario.....	5
Áreas de aplicaciones y/o alcance de los procedimientos.....	5
<b>Responsable.....</b>	<b>7</b>
<b>Glosario.....</b>	<b>11</b>



## Introducción

Este documento contiene toda la información sobre los recursos utilizados por el proyecto, llevando descripciones detalladas sobre las características físicas y técnicas de cada elemento.

Este manual técnico va dirigido para aquellas personas que tenga conocimientos técnicos sobre programación, en este caso será lenguaje C. En este se lograrán identificar los aspectos y características que forman parte del Medidor de Pulso Cardíaco, pues podrán conocer aquellas ventajas y desventajas, características y funcionalidades, funciones y ventajas, así como costos y/o beneficios que implican el desarrollo de este proyecto.

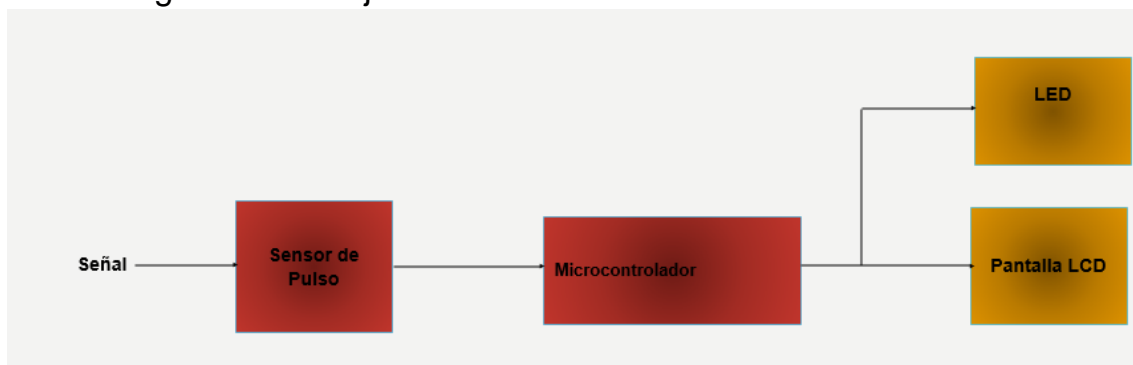


## Objetivos

- Identificar toda la información acerca del proyecto.
- Aplicar las mejoras en las diversas áreas, utilizando un método de monitoreo para asegurar que están se estén llevando a cabo de la forma correcta.
- Comparar el funcionamiento de los dispositivos que funcionan y los que no.
- Conocer las funcionalidades de los distintos aparatos.

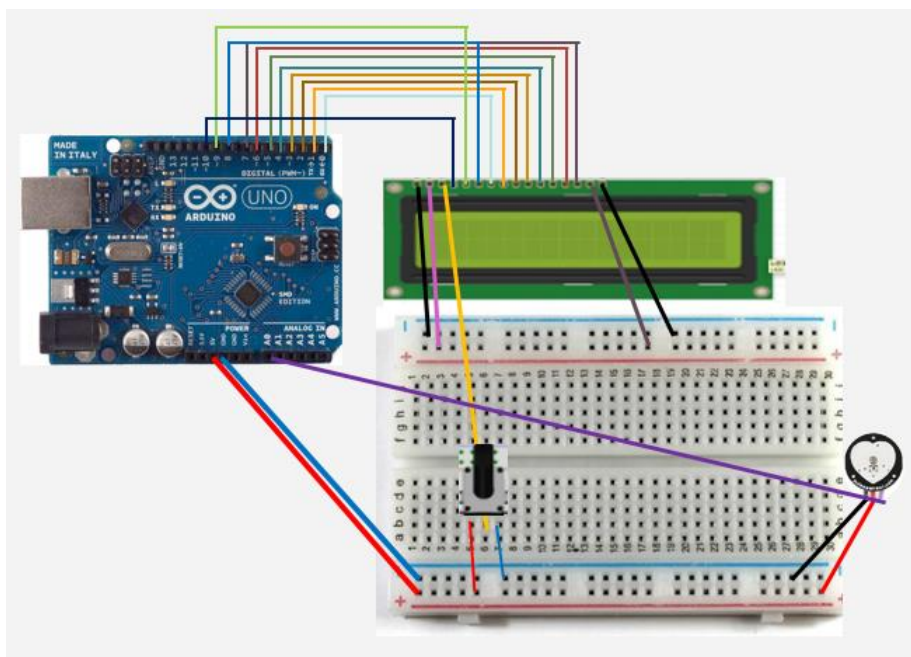
## Contenido técnico

### Diagramas de flujo de datos



El proceso para entender el funcionamiento de un Sensor de Pulso es sencillo, primero entrará una señal (en este caso nuestro dedo la transmitirá

### Modelo de datos físico



Para la construcción del Medidor de Frecuencia es necesario el uso de diferentes dispositivos para su funcionamiento tales como:

Tarjeta Arduino UNO. - se encargará de hacer que el sistema funcione, es decir, reciba y mande información, en este caso, recibirá la información del sensor de pulso se pasará al lenguaje que entiende el microcontrolador y

Pantalla LCD. – Transmite la información del sensor de pulso, nos muestra el valor de BPM.

Sensor de pulso. – Encargado de emitir las señales de pulso, en este proyecto se usó un sensor de pulso Xd-58c.



Nota: Para más información acerca del sensor de pulso vaya al documento “sensor de pulso”, esto para comprender mejor el funcionamiento.

LED. – Servirá de guía de acuerdo al estado del sensor.

### Plataforma de usuario.

Para el uso correcto de este proyecto, es necesario tener los requerimientos mínimos de las partes que constituyen al hardware y al software.

Por parte del hardware es indispensable conocer como es que funciona cada una de las partes que constituyen el Medidor de Frecuencia Cardiaca. Pues como se sabe, cada componente tiene su funcionalidad, para ello se requiere de conocimientos de la conexión de cada uno, pues se pueden llegar a dañar, lo que lleva a un mal funcionamiento y por consiguiente resultados totalmente erróneos.

De igual manera se debe de tener conocimiento sobre que tipo de cable utilizar para la circuitería, esto es debido a que existen cables que son muy delgados y pueden romperse creando mala funcionalidad, o por el contrario ser demasiados gruesos provocando que no embonen bien en los pines de algún componente como el caso de la tarjeta Arduino (placa basada en un microcontrolador ATMEL, en donde un microcontrolador es un conjunto de circuitos integrados en donde se pueden grabar diversas instrucciones, las cuales pueden ser escritas en lenguaje de programación, estas permiten crear programas para interactuar con otros dispositivos electrónicos).

Mientras que por parte del software es necesariamente el conocimiento de lenguaje C en los programas de Atmel Studio y Arduino IDE.

Es importante el lenguaje de Atmel Studio porque es aquí donde se realiza todo lo que hace funcionar al microcontrolador Arduino, donde se usan los puertos de entrada, salida y todas las series de instrucciones para hacer lo que uno desea. Sin embargo, se debe de saber cómo se escribe el código, las librerías a utilizar, etcétera.

Incluso el saber como unir Atmel Studio con la placa de Arduino o incluso Atmel con programa de Arduino IDE, pues en algunos casos será necesario, como el caso de el Medidor de Frecuencia Cardiaca, donde se puede observar las gráficas de la señal de pulso y los valores de dicha señal.

### Áreas de aplicaciones y/o alcance de los procedimientos

Para saber si su placa Arduino uno funciona correctamente se deben de seguir los siguientes pasos:



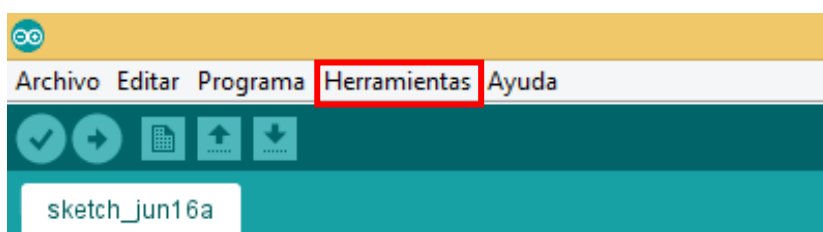
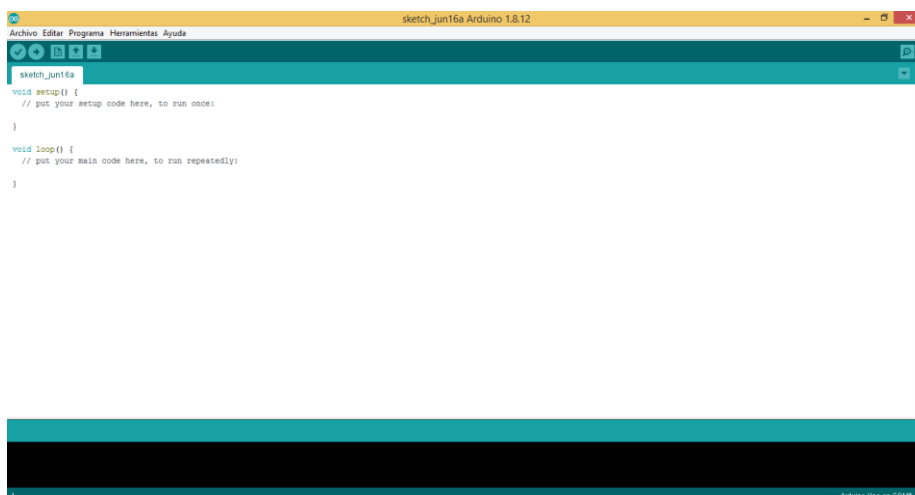


Asegúrese de que su placa Arduino uno se encuentre bien conectada a su computadora.



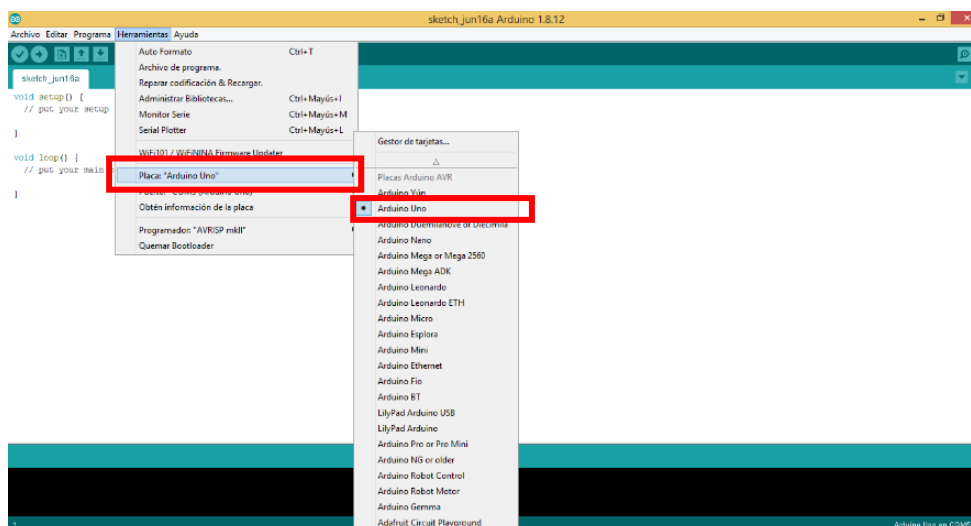
De igual forma asegúrese que el led del arduino del prenda, pues si no lo hace, significara que no funciona.

Abra el programa de Arduino IDE, para poder localizar el puerto al que se encuentra conectado su tarjeta de arduino uno.

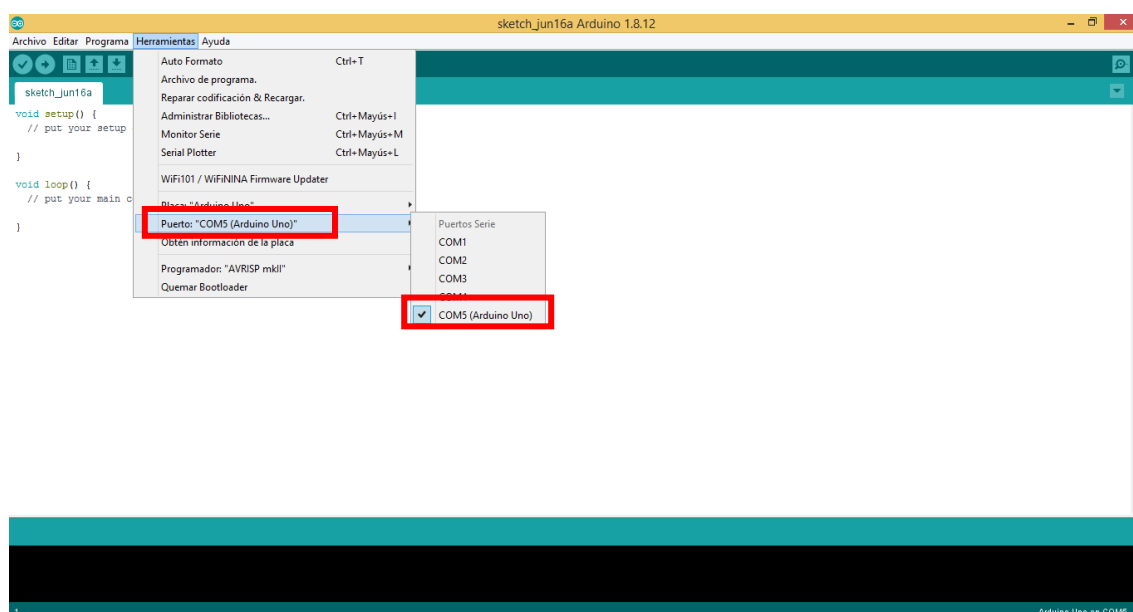


Vaya a la opción de herramientas.

Seleccione la opción de placa: “Arduino uno” y después Arduino uno

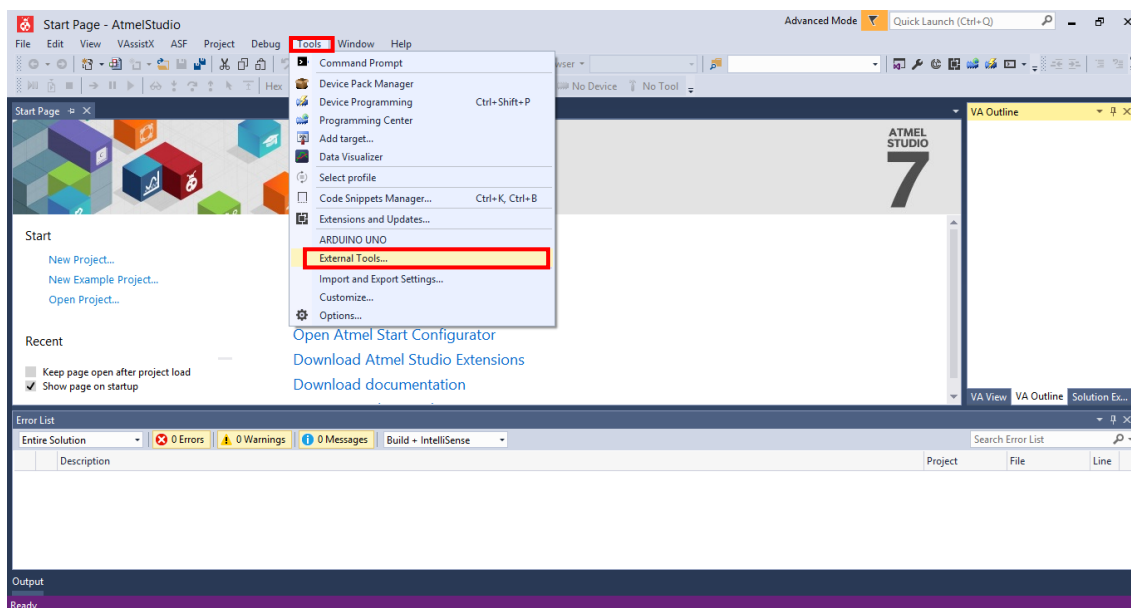


Ahora elija la opción puerto, pues es aquí donde se podrá ver en que puerto serie se encuentra la tarjeta de Arduino, en este caso será COM5



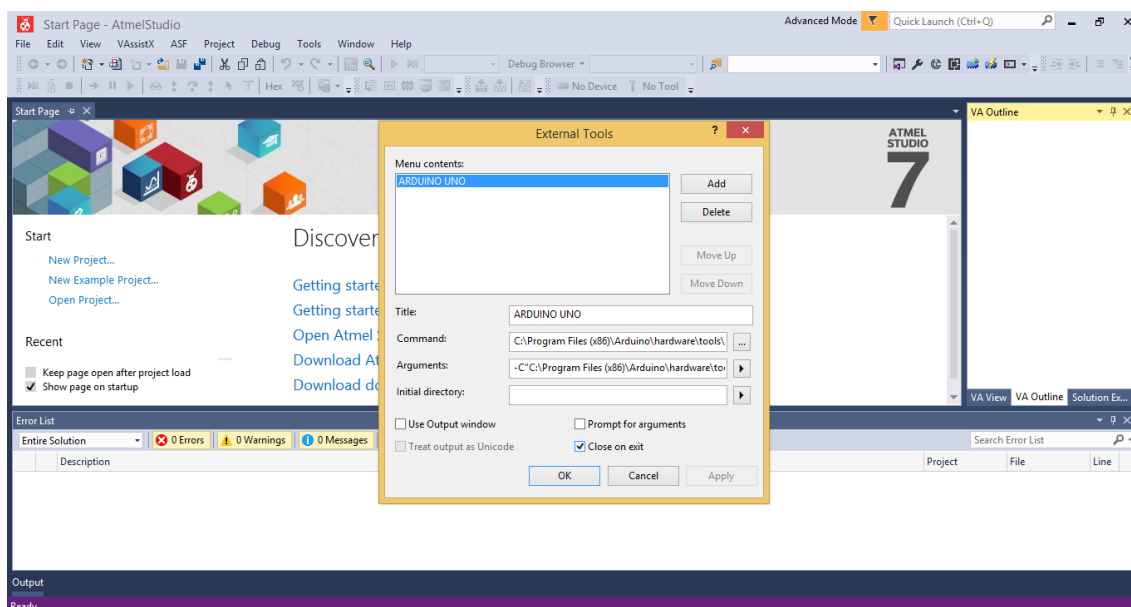
Una vez observando que, si funciona el Arduino y detecta la computadora el puerto, se pasa a Atmel Studio.  
Primero se selecciona Tools, continuando con la opción de External Tools...





A continuación, saldrá una nueva pantalla, es aquí donde será necesario crear la conexión entre el código creado en Atmel Studio y la placa de Arduino.

Primero se le da un nombre, por ejemplo: ARDUINO UNO, después en la opción de command, se agrega la ruta de avrdude.exe, esta puede variar de acuerdo a donde se tenga guardado, después se establecen la opción de arguments: -C\"(ruta de avrdude.exe\"-p atmega328p -c arduino -P (Puerto) -b 115200 -U flash:w:\"\$(ProjectDir)Release\\\$(TargetName).hex\"-p atmega328p -c arduino -P COM3 -b 115200 -U flash:w:\"\$(ProjectDir)Release\\\$(TargetNa

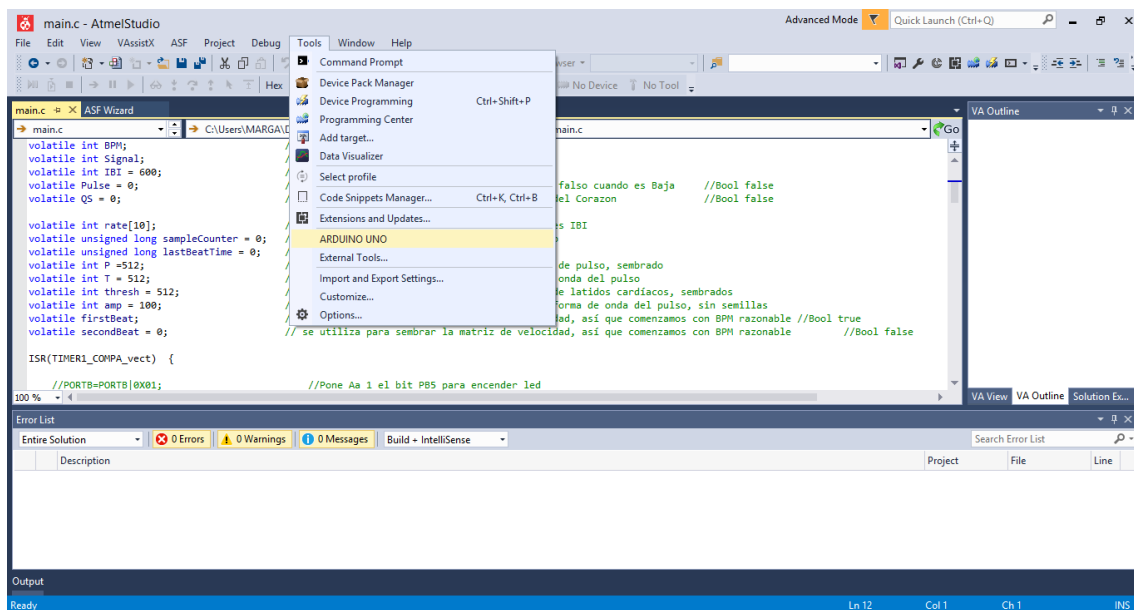


Una vez configurando lo anterior correctamente. Se dirige al código de Atmel Studio

Seleccionando el nombre que se le dio anteriormente en la configuración, esto hará que el código se ejecute en la placa de Arduino. Para saber si es este se

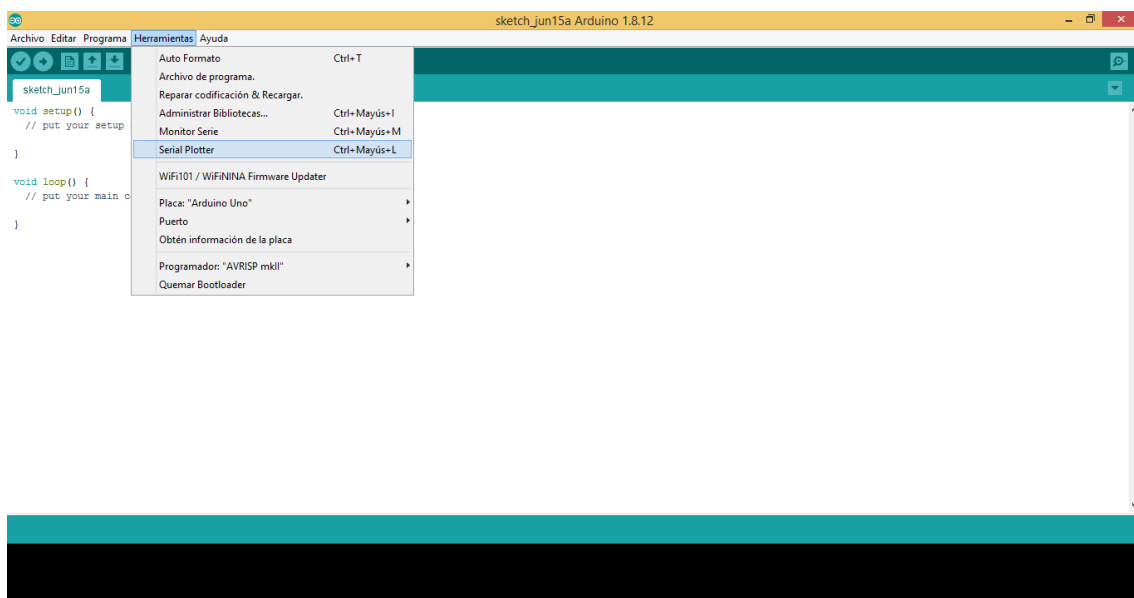


subió, la placa, cuenta con led, el cual debe de parpadear y hasta que este deje de hacerlo significará que ya esta listo para juntarlo con los demás componentes.



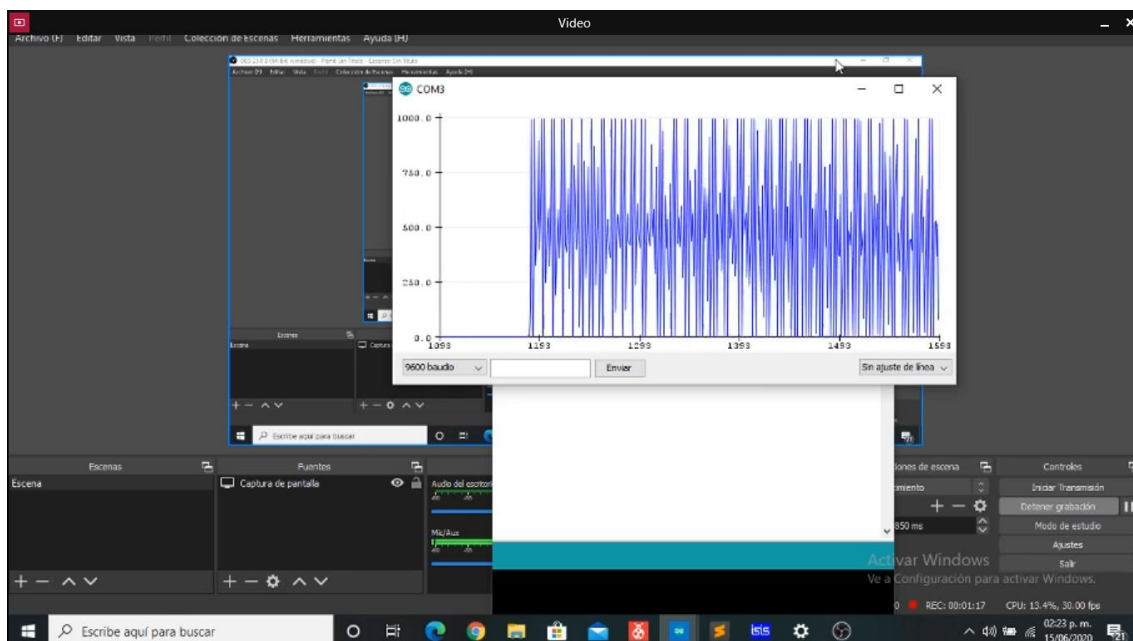
Después de esto y teniendo el circuito completo ya listo (sección de modelos de datos físicos).

Una vez conectados, el resultado que manda el puso se podrá ver en una gráfica, en la opción de herramientas y después serial plotter.

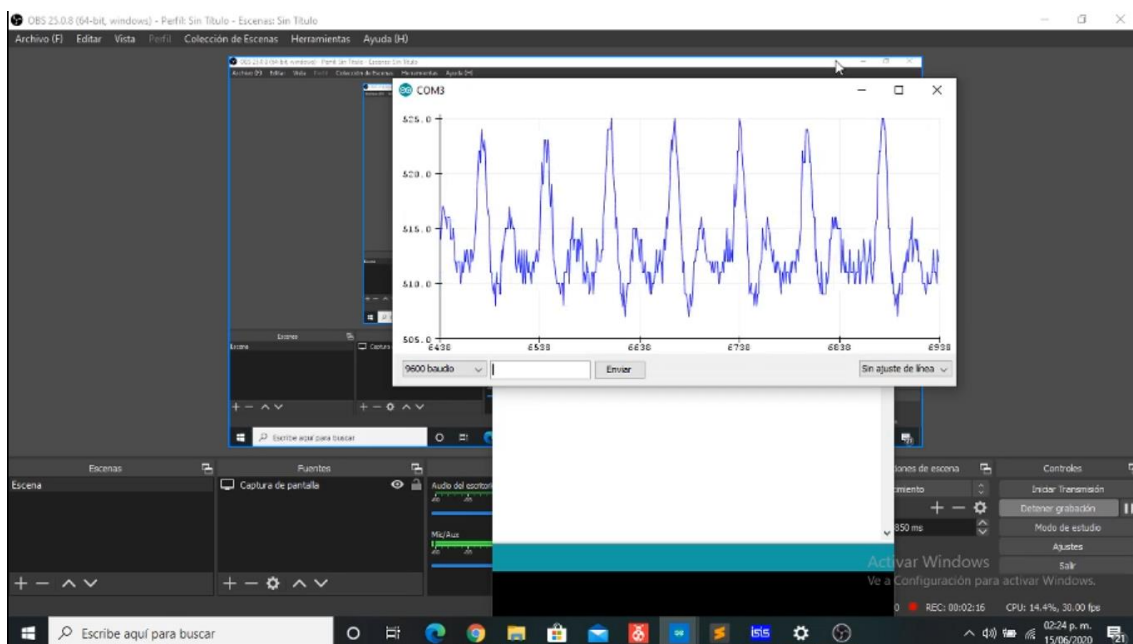


Nos saldrá una pantalla como la siguiente imagen en donde saldrán los datos graficados. Cuando el sensor esta en el aire, es decir no se pone el dedo para que se detecte el pulso, tiene a saturarse como en la siguiente imagen.





Pero si se coloca el dedo en la parte trasera de donde se encuentra los pines después de un tiempo se estabilizará, provocando una mejor señal como se muestra en la siguiente imagen.



Y si se elige la opción de monitor serie, saldrán los valores de los pulsos. Como en la siguiente imagen:

```

0
0
0
0
373
362
352
342
333
323
314
305
296
282
255

void loop(){
  int pulso = analogRead(A0);           //Lee el valor del pulsometro conectado al puerto Analog A0
  if (pulso >= 530) {                   // Enciende led 13 cuando el pulso pasa de un valor (debe ajustarse)
    digitalWrite(13, HIGH);
  }
  else{
    digitalWrite(13, LOW);
  }
  lcd.setCursor(1,0);                   //Muestra en el LCD el valor de BPM
  lcd.print("BPM= ");
  lcd.print(BPM);
  lcd.print(" ");
  //Serial.print("BPM = "); Serial.println(BPM); //Habilitar estas lineas para ver BPM en el monitor serial pero deshabilitar la siguiente
  Serial.print(pulso);                  // envia el valor del pulso por el puerto serie (desabilitarla si habilita la anterior linea)
  Serial.println(pulso);
  if (QS == true){                      // Bandera del Quantified Self es verdadera cuando el Arduino busca un pulso del corazon
  }
}

```

Monitor serie no disponible mientras plotter este abierto

Arduino Uno en COM5

Al principio saldrán valores muy variados, pues la señal debe de estabilizarse, sin embargo, después de 40 segundos aproximadamente ya será estable. Obteniendo valores mayores a 520 (estos varían de acuerdo con la frecuencia de cada persona y del estado en que se encuentre, puede estar agitada). Si los datos que dan no son mayores a 530 o varían por mucho, significará que el sensor no funciona. Y es necesario cambiarlo.

Ahora por parte de los dispositivos tangibles, tales como el LED y la pantalla LCD. La pantalla LCD, mostrara los valores de nuestra BPM, los cuales irán cambiando de acuerdo con los latidos de nuestro corazón. (Asegúrese de primero cargar el código en la tarjeta Arduino y después conectar la pantalla LCD, para que esta logre funcionar correctamente).

El LED al principio prendera (cuando no este colocado su dedo), sin embargo, al momento de que se coloque el dedo se apagará, esto nos estará avisando si el sensor de pulso esta detectando las señales que recibe.

## Responsable

Queda en responsabilidad del usuario el uso que le de los cambios que realice. Si se tiene problemas con resultados totalmente malos, es recomendable llevarlo al lugar donde lo adquirió, donde le darán el correcto mantenimiento de sus funcionalidades, siempre y cuando se compruebe que no ha existido intervención alguna por el usuario.

## Glosario

- Hardware: *Conjunto de elementos físicos.*
- Software: *Conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas.*
- Arduino IDE: *Entorno de desarrollo integrado, llamado IDE (sigla en inglés de integrated development environment), es un programa informático compuesto por un conjunto de herramientas de programación. Puede dedicarse en exclusiva a un solo lenguaje de programación o bien puede utilizarse para varios.*
- Atmel Studio: *es la plataforma de desarrollo integrado (PDI) para el desarrollo y la depuración de Atmel ARM® Cortex® -M y Atmel AVR® microcontrolador (MCU) aplicaciones basadas. El 6 IDP Atmel Studio le da un ambiente sin costuras y fácil de usar para escribir, crear y depurar sus aplicaciones escritas en C / C++ o código ensamblador.*  
<https://forum.arduino.cc/index.php?topic=347290.0>
- Arduino: *Placa de microcontrolador de código abierto basado en el microchip ATMEGA 328p*
- LED: Diodo Emisor de Luz.
- Microcontrolador: *Circuito integrado el cual contiene una unidad central de procesamiento (CPU).*
- BPM: *El ritmo cardíaco es el número de latidos del **corazón** por la unidad del tiempo. El ritmo cardíaco se expresa generalmente como batidos por minuto (BPM). Los batidos de **corazón** a proveer oxigenaron sangre limpia del ventrículo izquierdo a los vasos sanguíneos de la carrocería vía la aorta.*  
[https://www.news-medical.net/health/What-is-Heart-Rate-\(Spanish\).aspx#:~:text=El%20ritmo%20card%C3%ADaco%20es%20el,la%20carrocer%C3%ADa%20v%C3%ADa%20la%20aorta.](https://www.news-medical.net/health/What-is-Heart-Rate-(Spanish).aspx#:~:text=El%20ritmo%20card%C3%ADaco%20es%20el,la%20carrocer%C3%ADa%20v%C3%ADa%20la%20aorta.)