Desafío 1

Andrés Romero Capachero

Nikolas Geovanny Ortega Suarez

Informática II

Aníbal guerra

Augusto Salazar

Universidad de Antioquia

Medellín, Antioquia

11 de septiembre de 2024

**Informe desafío #1**

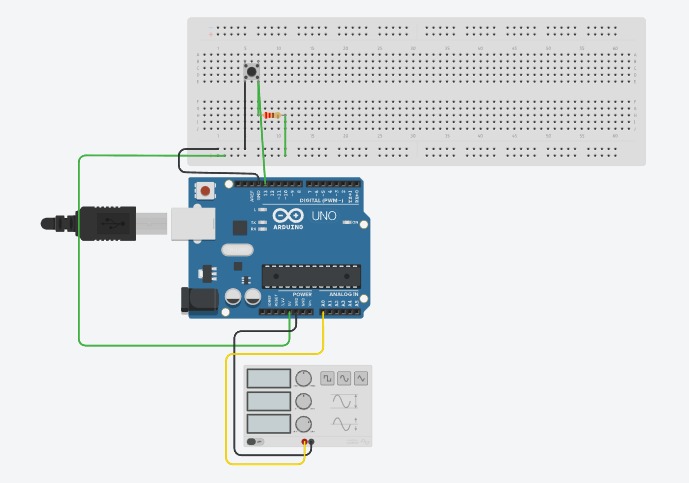
**Análisis del problema:**

El desafío presenta los siguientes retos:

* Implementar un código íntegramente de C++ sin librerías especiales (exceptuando que sean de autoría propia).
* Utilizar la plataforma de simulación tinckercat, específicamente usando Arduino Uno, un generador de señales, pulsadores, resistenecias y una pantalla LCD.
* Crear un proyecto capaz de recibir y traducir en imagen diferentes señales con diferentes formas e intensidades.
* Uso obligatorio de punteros, arreglos y memoria dinámica
* Commits diarios del avance
* Responsable uso de la memoria en vista de las limitaciones de Arduino

En vista de los ítems anteriormente mencionados se plantea la siguiente solución; por al menos dos días nos dedicaremos a la captación de información necesaria para comenzar con el desarrollo, sin conocimientos previos de Arduino seria irreal plantear un plan de acción dando inicio al desarrollo real el día 13 de septiembre, por ende, el compromiso es desarrollar las habilidades en implementación de memoria dinámica y arreglos además de las competencias necesarias en Arduino

Hoy 14/09/2024 se hacen las siguientes actualizaciones al informe, tras tres días de trabajo e investigación se logran avances en codificación teniendo en cuenta el comportamiento de las funciones a evaluar, considerando sus cambios como factores importantes para su reconocimiento, logrando traducir exitosamente señales senoidales y cuadradas, mediante el uso de arreglos y condicionales; Con diferentes valores en el generador de funciones, en cuanto a implementación de punteros y arreglos se han utilizado arreglos durante las pruebas y ya se tiene planteada la manera de poner punteros y memoria dinámica, sin embargo, en la fase de programación actual no se ha requerido de dichos ítems

El apartado electrónico del proyecto se ha hecho hasta ahora con un pulsador, una placa, Arduino Uno y una resistencia, el generador de funciones; aun no se implementa el LCD, todo manejado y supervisado mediante la consola 

Para los próximos días de trabajo se definen como tareas la implementación la librería adafruit, la creación de la lógica para interpretar la señal triangular, el uso de punteros y memoria dinámica, perfeccionamiento del circuito electrónico además de, ser eficientes en el trabajo colaborativo, creación del video explicativo a nivel de electrónica, código y funcionamiento de ambos.

Hoy 18/09/2024 como ultimo informe de proceso y como entrega definitiva del desafio uno scamos las siguientes conclusiones de cada item a evaluar en este informe:

**a-Analisis del problema y consideraciones para la alternativa de solución propuesta:**

en el análisis inicial del problema recononocemos nuestra completa ignorancia sobre aspectos como electrónica, uso de señales, y dispositivos como Arduino, a lo largo del desarrollo se fueron proponiendo soluciones básicas como encontrar lógicas que basadas en el muestreo de una onda y la evaluación de sus valores, den la información suficiente para el reconocimiento de su tipo, amplitud y frecuencias sin que factores como el desface lo afecten, con el pasar de los días dichas soluciones se encontraban con problemas como por ejemplo la incapacidad de diferenciación entre la señal senoidal y triangular, que en un entorno ideal muestran diferencias claras en sus picos y valles sin embargo en el ambiente de desarrollo simulado no cuentan con sus parámetros generales, sino que sufren alteraciones en sus formas, para la señal triangular se intentó integrar una lógica basada en la comparación de las pendientes entre dos puntos consecutivos y además los cambios entre subidas y bajadas, por otro lado también se intentó con derivación, ambas ideas descartadas por su ineficacia en el reconocimiento, además en el caso de la frecuencia de cada uno, las ideas propuestas iban desde temporizadores, conteos en el muestreo, pruebas con delay, comparación entre sus picos, todas y cada una implementadas sin éxito.

En medio del desarrollo estuvo la idea de una función que comparara directamente los valores de la función ingresados en un arreglo que reconociera los picos de la función; comparara la pendiente del punto con el punto anterior y el siguiente al pico, dichos valores en determinados rangos de error podrían decirnos la naturaleza de dicha onda, sin embargo, por falta de tiempo esta alternativa se quedo en el ideal.

Como retos frente a la electrónica del desafío decimos que tomo alrededor de 3 días de indagación e investigación para saber el funcionamiento de Arduino(tanto la placa, sus entradas, y las palabras reservadas para su correcto funcionamiento en el código y sus conexiones a otros dispositivos)

**b-esquema de tareas en el desarrollo del algoritmo:**

Como raíz fundamental todo proceso, una buena esquematización de tareas ayudo al desarrollo del desafío, definiendo en primera instancia el auto reconocimiento de falencias respecto a las necesidades del desafío llevando el siguiente orden lógico:

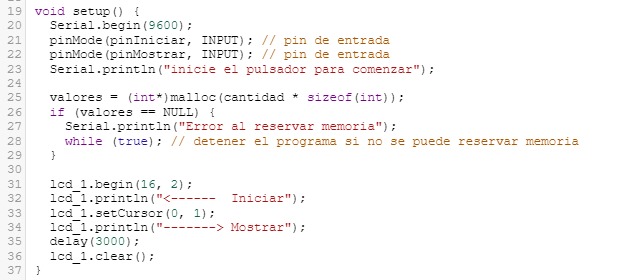
* Tanto Nikolas como Andrés, en la primera fase de desarrollo (días del 1-3) se dividieron la investigación sobre Arduino uno y su parte electrónica, su parte algorítmica y su parte lógica, además la investigación de cada uno a cerca de la librería adafruit y el uso de pantallas LCD y su código implicado, sin olvidar el buen uso de arreglos y memoria dinámica
* Para la segunda fase (días del 3-7) basados en los conocimientos adquiridos en la fase 1 comenzamos con el armado del circuito, en un momento inicial sin LCD solo para pruebas con el Arduino y el generador de funciones, se desarrollan las funciones para el reconocimiento de señales senoidales y cuadradas, además de su amplitud, en una segunda instancia alrededor del día 6 se implementa el LCD al circuito, y sus respectivos ajustes al código, para que los datos analizados de la onda se vean reflejados en el LCD, luego de esta durante el día 6 y 7 se hacen ejecuciones de prueba y error para agregar tanto la función de reconocimiento de la onda triangular como la función capaz de traducir los datos en la frecuencia de cada onda (antes dicho, implementadas sin éxito)
* En una tercera fase de desarrollo ( día 8) se hace uso del día para refinar detalles sobre el código y su funcionamiento hasta donde está definido en busca del perfeccionamiento del trabajo hecho hasta el momento.

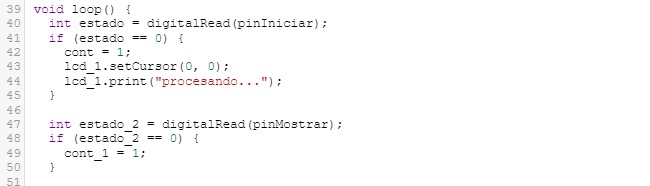
**c-Algoritmos implementados**

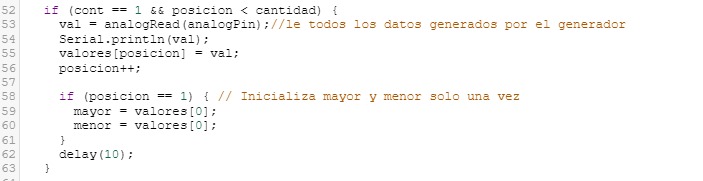
1. Se hace la inicialización de variables necesarias para el trabajo del código, y declara el puntero para el arreglo dinámico, además de la cantidad de valores del arreglo

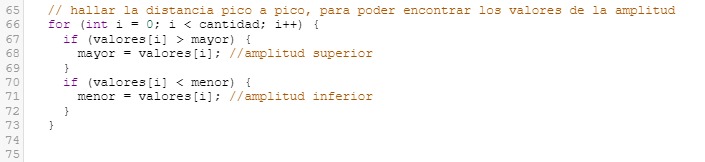
**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

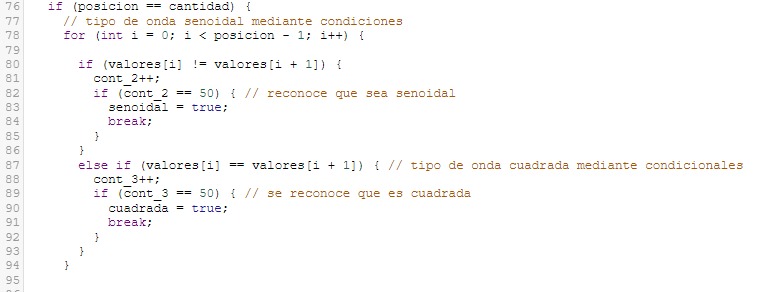
Descripción generada automáticamente**

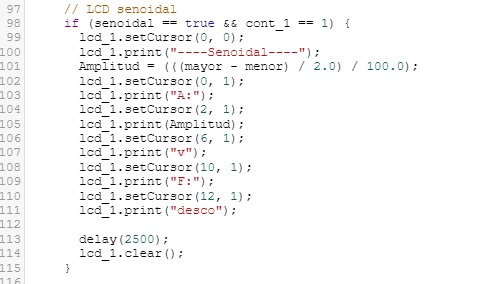
2-establece los pines de entrada y reserva memoria (en caso de no haber suficiente retorna un error), se genera el LCD y se muestra en pantalla cual pulsador es para empezar y cual pide la información 

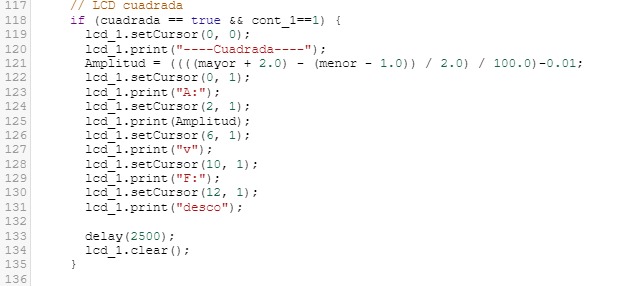
3-define el estado de los pulsadores 

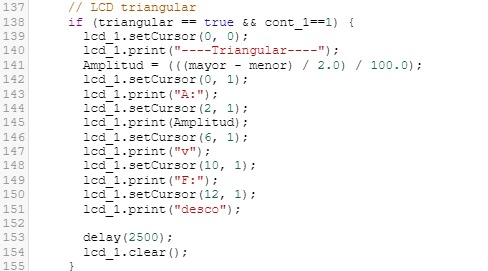
4-se generan los valores y se guardan en el arreglo, tomando el primer valor del arreglo como punto de comparación 

5-haya las crestas y valles de la función para definir la amplitud

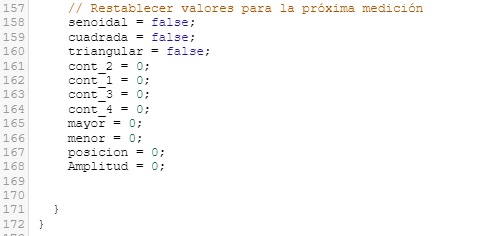
6-mediante el uso de condicionales, se comparan las características de la onda para definir si es senoidal o cuadrada, basado en los cambios en los valores 

7-se configura el LCD para imprimir el tipo de onda y la amplitud de la senoidal 

8- se configura el LCD para imprimir el tipo de onda y la amplitud de la cuadrada

9- se configura el LCD para imprimir el tipo de onda y la amplitud de la triangular(A PESAR DE NO ESTAR IMPLEMENTADA SE TENIA EN CUENTA COMO POSIBLE)

10-Por ultimo se restablecen los valores



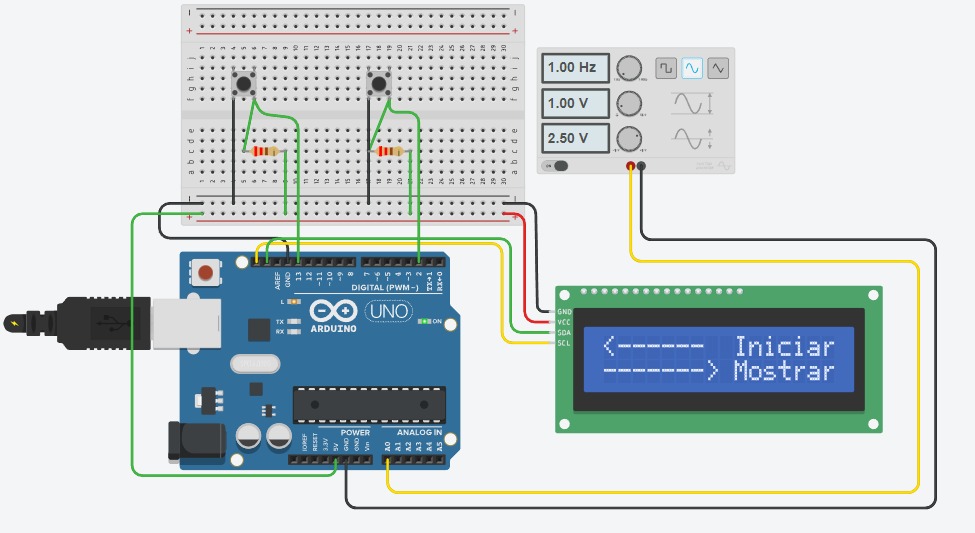
Posibles resultados: 

Imagen que contiene electrónica, circuito

Descripción generada automáticamente

**d-problemas de desarrollo:**

* Falta en conocimientos previos para el desarrollo de la actividad en el tiempo estipulado
* Poca intuitividad en el manejo del simulador
* Tamaño del arreglo (que tan grande debía ser la recepción de datos)
* Velocidad en la captación de datos
* Propiedades de las ondas afectadas por el simulador
* Propiedades del tiempo que son muy distintas entre la realidad y la simulación
* Uso eficiente de la memoria
* Problemas par el reconocimiento preciso de la amplitud, cuestiones del desface
* Disposición horaria de ambos por el cumplimiento de responsabilidades académicas con el laboratorio y otras materias del semestre
* Falta de tiempo( en la entrega)
* Ambigüedad en las “soluciones” propuestas por los profesores en las clases

**e-** **Evolución de la solución y consideraciones para tener en cuenta en la implementación:**

De forma resumida se adquirieron los conocimientos básicos en el uso de Arduino Uno, se mejoró la lógica en la implementación de ecuaciones en el algoritmo, el funcionamiento básico y uso de una pantalla LCD y el reconocimiento de ondas basados en sus características, como consideraciones siempre se tuvo presente el correcto uso de la memoria y la correcta implementación de los recursos como punteros y arreglos, reconociendo los limitantes del ambiente de simulación, el tiempo y el ruido en las ondas