INFORME

El actual informe corresponde al primer afrontamiento lógico que aborda la compresión y abordamiento de la solución del problema algorítmico por tanto está sometido a posibles cambios

En el momento de la discusión afrontamos distintas ideas o puntos base para empezar con el desarrollo algorítmico y entre esos el caso persistente corresponde a varias partes fundamentales:

Análisis y diseño:

**Adquisición y organización de los datos:**

La información procedente del microcontrolador el cual entrega valores aproximados a funciones de ondas por parte de un osciloscopio deben de ser almacenados en estructuras dinámicas. Esa las vamos a implementar mediante una función la cual se encargará de leer los datos de la salida digital del arduino correspondientes a la entrada análoga del osciloscopio, esta función tendrá la tarea de gestionar el volumen del arreglo el cual dinámicamente estará en constante aumento de volumen hasta terminada su llamada por parte del pulsador final. De forma interna asignaremos un arreglo el cual será reemplazado por uno nuevo que contenga sus datos más el nuevo elemento que ingrese por el pin digital y eliminando el bloque anterior para evitar fugas de memoria.

**Caracterización de la información:**

En esta sección se busca identificar las diferentes características de la función como son los valores predeterminados de una onda(frecuencia, amplitud, periodo, etc..) con el uso de los datos añadidos al arreglo.

Para definir estas funciones es fundamental la previa adquisición de los datos, luego adquirir el tiempo por medio de una función como base para obtener los diferentes valores.

Planteamos al ser C++ un lenguaje de ejecución secuencial, la ejecución de la función de reconocimiento de tiempo la cual se definirá con el uso de la función milis() que determinará el tiempo entre la pulsación inicial con la final y retorna valores en milisegundos añadiendo precisión a datos como la frecuencia. Esta función será implementada dentro de la función de adquisición de datos para asegurar una coherencia dentro de la duración de la adquisición, dicha función será fundamental para determinar valores como el periodo o ciclo, fundamentales para caracterizar la onda.

Las diferentes funciones que se encargan de distinguir los diferentes valores serán una simple implementación de igualdades físicas que nos dan dichos valores.

El principal obstáculo en esta porción es la inexactitud de los datos entregados por el microcontrolador los cuales tienen una tendencia a distorsionar la precisión de los valores, en el actual momento la solución o necesidad de esa solución está en discusión, pero hasta ahora algún acercamiento a ella es el aumento del valor de la función de Serial.begin(), a la cual al aumentar sus valores, aumenta la capacidad en baudios para procesar la información del osciloscopio especialmente a frecuencias altas.

Para ello iteramos los datos del arreglo principal, primeramente identificamos el tipo de onda del ciclo hasta que nos encontremos una cresta, la cual la definimos aquel punto en el que los datos dejan de ascender y comienzan a descender, a medida del paso de los datos, vemos si este tipo es constante, lo cual nos debe dar un resultado en el que la onda es de un tipo y mostrandolo en el LCD, en el caso contrario en el que haya varios tipos se mostrará como desconocida.

Para identificar los ciclos tendremos en cuenta 3 casos: triangular , cuadrada y senoidal. En la triangular se busca una relación lineal entre los datos, mientras en la cuadrado se buscará un patrón constante en dos tiempos y por último en la senoidal se busca una relación cambiante entre la diferencia de los datos.

Debido a inexactitudes con los datos este proceso será impreciso por lo que se buscará siempre la aproximación. se definirá límites en las características de las ondas donde se busque un buen porcentaje de fiabilidad. Está en consideración una función para limpiar estos datos siguiendo un rango de valores de acuerdo a los patrones de la inexactitud de los datos que entrega el Arduino.

Impresión de datos:

Se implementará una función que recoja todos los datos obtenidos e imprima los datos en el display LCD conectado al Arduino, utilizando una librería propia de este.