

## Mendoza Meza Manuel Everardo

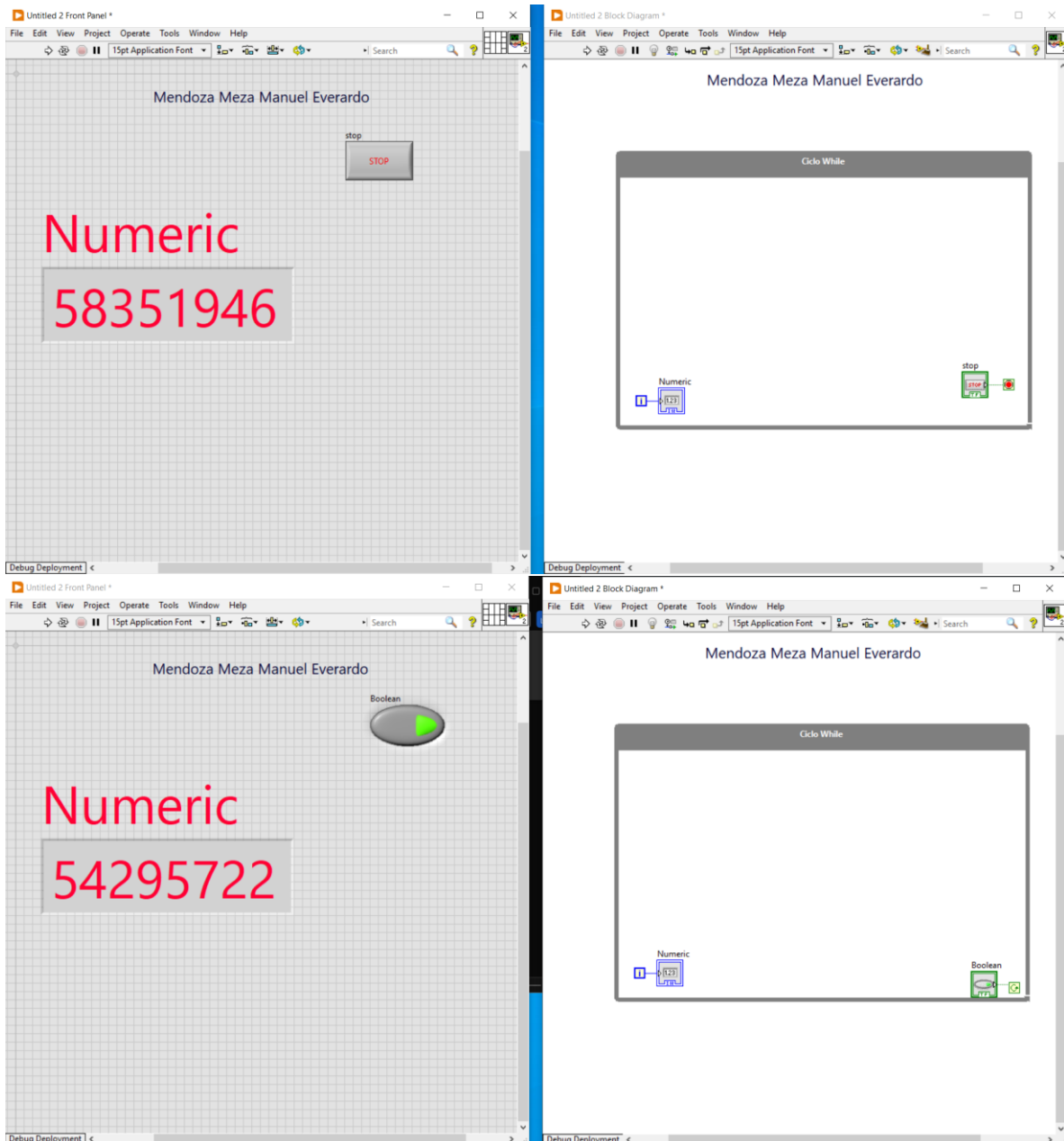


Imagen 1: Cambiando configuración de orden de paro

Dada la imagen 1 podemos observar el comportamiento y uso del ciclo While con las configuraciones de paro, representando "True" y "False" respectivamente, esto nos permite desarrollar sistemas en los que se involucre condiciones de paro, ya sea, por medio de la sentencia lógica 1 o 0, el ejemplo más claro, es el funcionamiento de la compuerta de un horno eléctrico, donde, la compuerta al encontrarse cerrada, genera un 0 lógico para evitar tanto la luz que emite dentro del horno (ya que, si no lo ejerciera, la luz se mantendría encendida todo el tiempo y haría que la tensión recorriese el sistema de forma consistente) a su vez, produce un 1 lógico para evitar que la tensión se disipe.

## Mendoza Meza Manuel Everardo

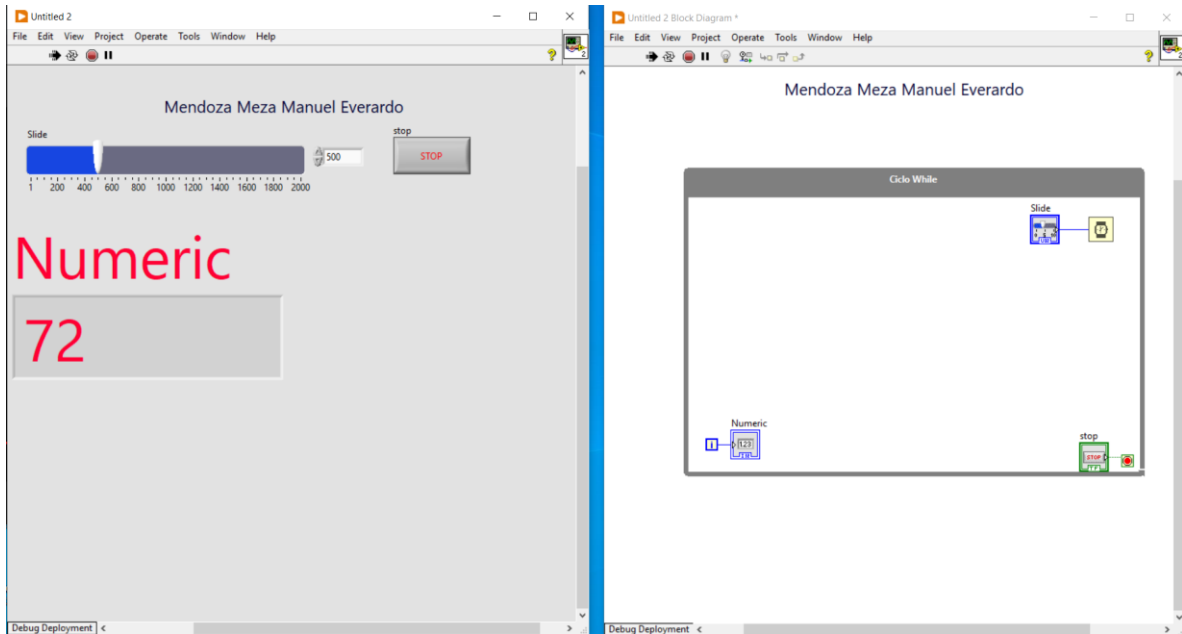


Imagen 2: Ocupación de slider e indicador

Con la imagen 2 podemos observar que aumentan las implementaciones al ciclo While, de tal forma que, el programa realiza un conteo que aumenta de 1 en 1 hasta cierto rango, dado el “slider U32”, la cual nos indica el tamaño de rango que puede llegar a alcanzar, con su respectivo botón de paro “stop”. Teniendo en cuenta este sistema, puede ser implementado para realizar un tipo de cronómetro, un ejemplo de programa es el sistema cronómetro que contienen casi todo sistema operativo dentro de celulares, relojes digitales, laptops y computadoras. Otro sistema puede ser un temporizador, ya que, con su configuración, producen un conteo, que puede detenerse si se presiona el botón de paro, de otra forma continua su conteo.

## Mendoza Meza Manuel Everardo

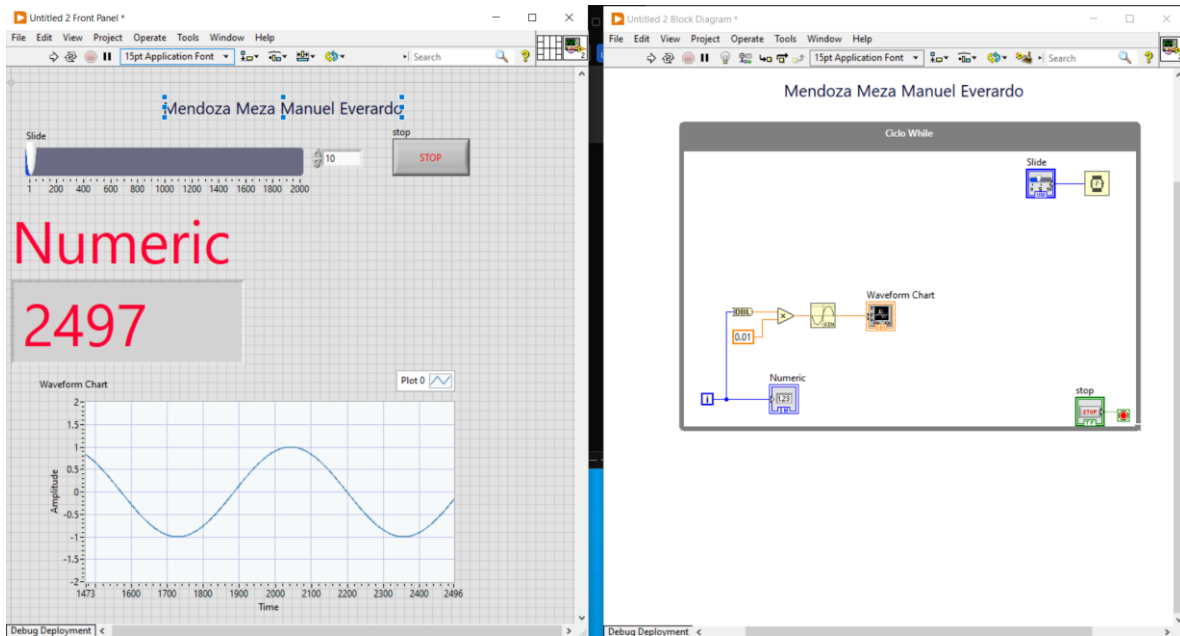


Imagen 3: Agregado de una onda sinusoidal al ciclo While

Con el agregado al ciclo While podemos observar que el comportamiento de la onda sinusoidal (o cualquier otro tipo de onda), esta representada bajo la función "SIN", sin embargo, tenemos que tener en cuenta que la función esta dada por default en radianes, esto implica que el valor que proviene de "i" es de valor entero, provocando que los valores que entran a la función seno sean de tipo entero, resultando valores incorrectos a la salida que es representada por la "waveform" en el panel frontal, por lo que es necesario colocarle una función que convierta el valor entero a double, después aplicamos una multiplicación de este valor convertido con una constante del mismo tipo (double) para hacer que la función se convierta de la misma forma que una señal sinusoidal correcta (imagen 3) Esta aplicación se da de forma notoria en un generador de ondas, ya que el comportamiento es precisamente el mismo,. Otro sistema que puede llevarse a cabo implementando este IV es un comparador de funciones que permita detectar cuando una función realice el mismo comportamiento, que con el que, uno puede ir controlando o variando hasta detectar dicha función.

## Mendoza Meza Manuel Everardo

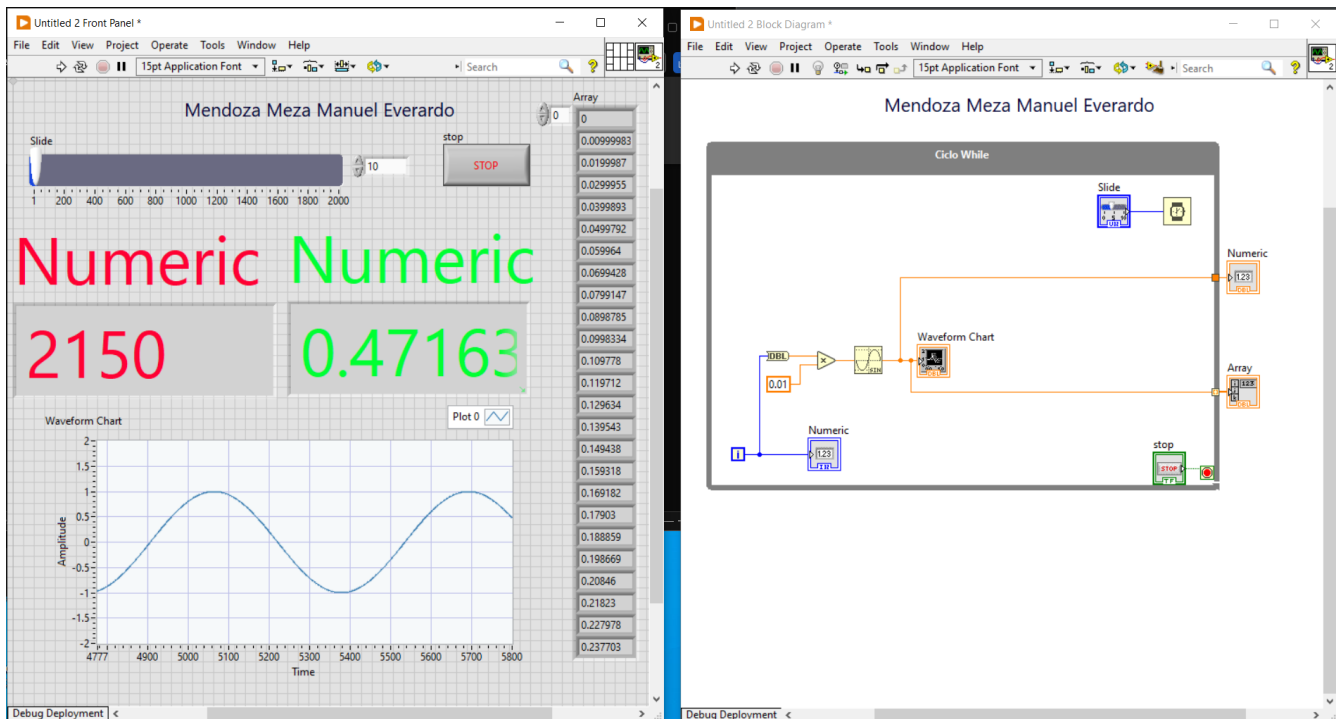


Imagen 4: Agregado "Numeric" para conocer comportamiento al clickear "stop"

Con el comportamiento mostrado en la imagen 4, podemos realizar de la misma forma que en un comportamiento for, generar una señal de salida del sistema While, esto es, una vez que el sistema se haya parado, generará los últimos puntos que obtuvo o por los que pasó el conteo de "i" esta cantidad de datos puede ser obtenida por dos medios, por el "Numeric" o por el arreglo "Array". Este último, es por lo que, se puede realizar un sistema de utilidad, para conocer los últimos puntos de guardado que pueden obtenerse dado una función o rango que esté bajo análisis. Es por ello, que un sistema muy útil, es la memoria de un celular, computador, o cualquier sistema digital que necesite este tipo de configuración, los cuales, constan de una función, o arreglo que permita la obtención de ciertos datos o valores antes que este se apague o se detenga.

## Mendoza Meza Manuel Everardo

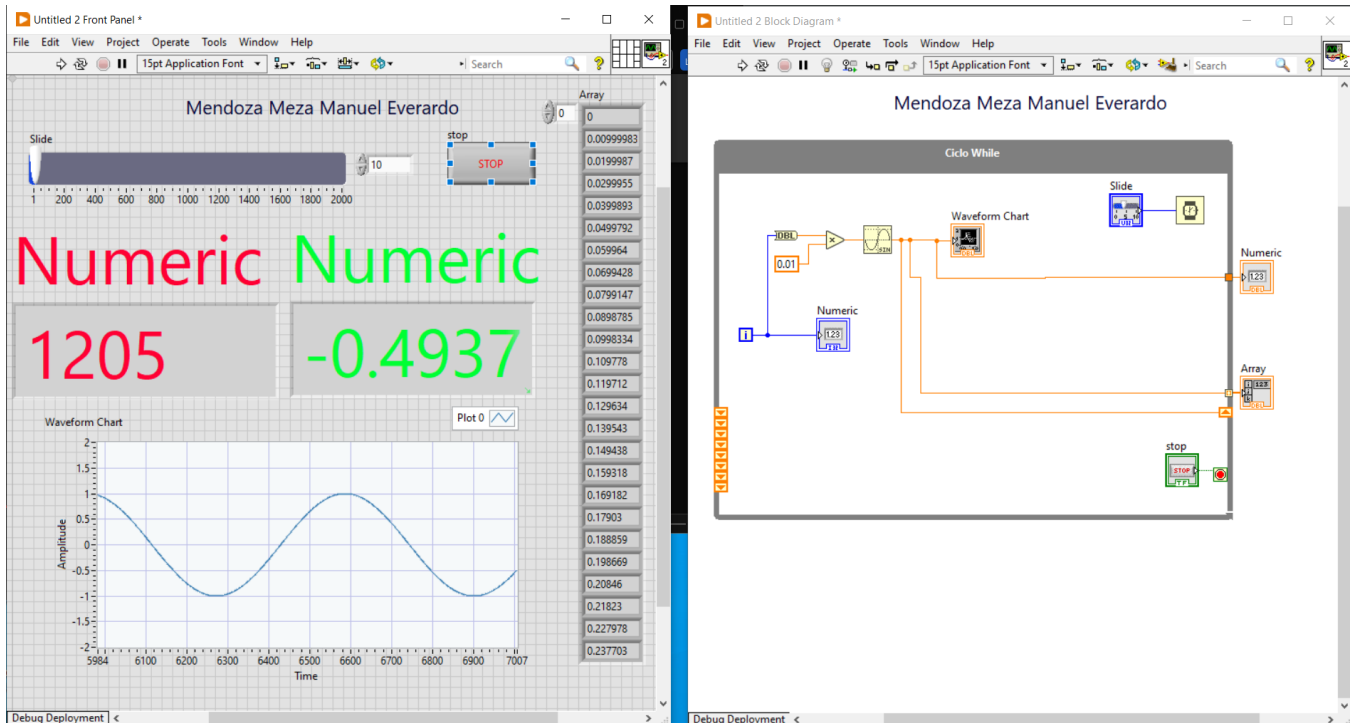


Imagen 5: Ocupando “shift Register”

Esta ultima configuración es importante, ya que con ella, podremos realizar muchas entradas que nos permitan realizar una entrada en específico, o por el caso contrario, podemos generar muchas entradas y salidas que permitan (a través de su propio comportamiento) salirse del ciclo While, este tipo de configuración es de vital utilidad en los sistemas actuales, ya que, a través de ciertos comandos que desee realizar el usuario, puede realizar un “escape” del ciclo While, esto nos muestra que, incluso con operaciones lógicas que pueden cargar con la mayor parte de aprovechamiento del sistema, pueden ser fácilmente salteadas con una simple instrucción del usuario. Esta configuración es normalmente vista en sistemas digitales que requieren de más de un comando a la vez, tales como, vehículos modernos, aviones, sistema de predicción en bolsa de valores y domótica.