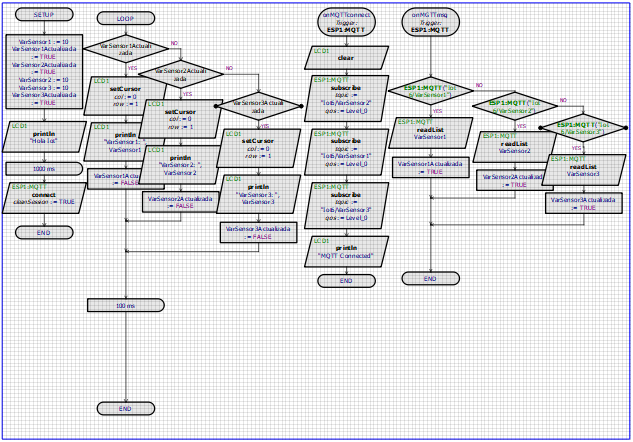
1. *PROTEUS*

FIGURA 1. CÓDIGO FUENTE IMPLEMENTANDO 3 SENSORES.

Como prototipo de nuestra propuesta, utilizamos la herramienta Proteus con el objetivo de simular los procesos que los sensores realizarán a la hora de colocarlos en las zonas determinadas. Como ya lo mencionamos con anterioridad decidimos implementar sensores de luminosidad (Sensor 1), sonido (Sensor 2) y temperatura (Sensor 3), en la imagen se observa el código fuente en formato de diagrama de flujo utilizado para realizar dicha simulación. Lamentablemente tuvimos problemas a la hora de compilar la simulación en Proteus por lo que  se decidió utilizar sólo 2 sensores para demostrar el funcionamiento de la simulación.

A continuación, se explicará el funcionamiento del código con los tres sensores. En el setup, se declaran todas las variables de control para poder recibir los datos de los sensores y la conexión con un cliente MQTT. En el loop, se asignan las posiciones de cómo se va a mostrar los datos en una pantalla LCD, además de recibir constantemente las lecturas de los sensores de luminosidad y de sonido. Estos sensores reciben señales análogas por lo que constantemente se están actualizando las variables de control. Después esta información la manda al cliente de MQTT para publicar la información y finalmente imprimir en la pantalla del LCD las lecturas de los sensores. Esta rutina se repite cada 100 m

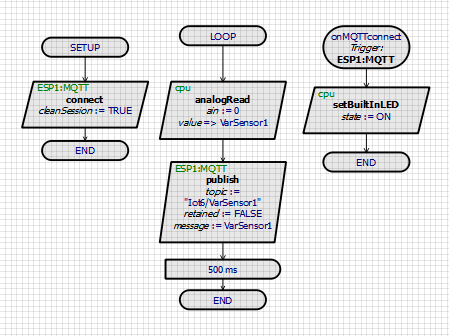
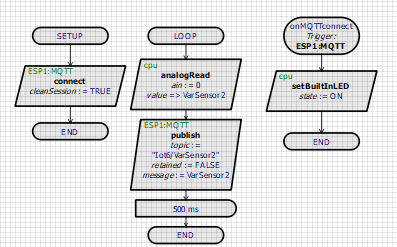
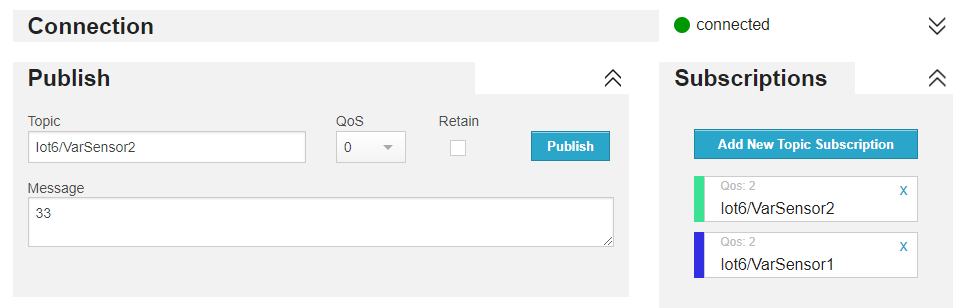
 

FIGURA 3. SENSOR 1. FIGURA 4. SENSOR 2.

Dentro de la ilustración 2 se puede ver código fuente utilizado para el resultado final de la simulación en donde se tiene como Sensor 1 al sensor de luminosidad, y al Sensor 2 como el sensor de detección de ruido. El código es bastante parecido al mostrado en la figura 1, sin embargo en este tenemos bloques condicionales solamente con respecto a los sensores previamente mencionados.

Para poder llevar a cabo la impresión en tiempo real de los valores que arrojan los sensores en nuestra pantalla LCD, es necesario implementar el protocolo MQTT. El funcionamiento del protocolo MQTT está descrito por la interacción y comunicación “De máquina a máquina”, en donde se transmiten mensajes a través de un proceso manejado por un publicador y un subscriptor.

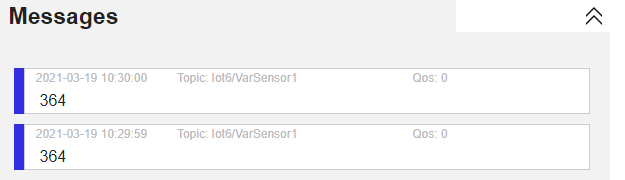
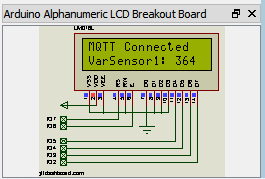
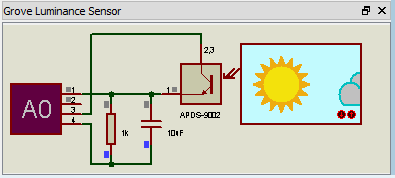
Dentro del código en el bloque de evento “onMQTTconnect” lo que estamos haciendo es que nuestra pantalla esté suscrita al broker con el tópico “Iot6/VarSensor1” para recibir la información del sensor de luminosidad, e “Iot6/VarSensor2” para leer los datos del sensor de ruido.

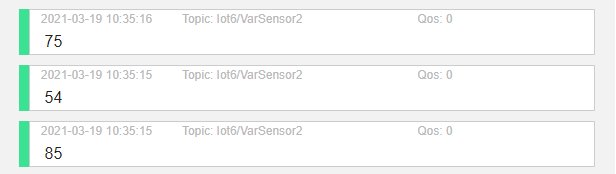
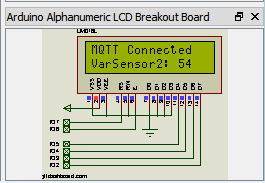
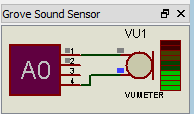
  
FIGURA 5. BROKER.

Este servidor central, llamado broker nos permite filtrar los mensajes y hacer que los dispositivos se encuentren conectados. Como se puede ver en las imágenes 3 y 4, los sensores están enlazados como publicadores haciendo que solamente envíen información a la central, para que después esta envíe estos números a todos aquellos dispositivos suscritos, como lo es el caso de la pantalla LCD.

Aquí, los tópicos permiten organizar los mensajes jerárquicamente. El tópico empleado para dividir la información del sensor 1 es “Iot6/VarSensor1”, mientras que para el 2 es “Iot6/VarSensor2”. Eso nos permite dividir la información en dos bloques distintos haciendo que como se ve en el código fuente 2, al tener bloques condicionales que revisan qué información está llegando, muestren en pantalla la información que es requerida medir.

En el apartado de subscriptions en la figura 5 podemos observar que se encuentran los tópicos que caracterizan a los sensores, y que son estos los que se están enviando datos a los dispositivos suscritos, como se mencionó anteriormente.

  
FIGURA 6. PRUEBA DE DESEMPEÑO SENSOR 1.

  
FIGURA 7. PRUEBA DE DESEMPEÑO SENSOR 2.

Como se puede observar en las figuras 6 y 7, ambos sensores trabajan correctamente y muestran a tiempo real la comunicación que hay a través del broker de MQTT.