**Procedimiento de control de calidad**

**Sistema electrónico**

**Carga de firmware**

**Tarjeta HMI:** En esta tarjeta cargue el script “configurationTest.ino” a través de la plataforma <<**Arduino**>>

**Funciones**:

Este Script esta diseñado para poner a prueba el funcionamiento de los periféricos del sistema,

**Pantalla LCD**: Escribirá sobre el display LCD el siguiente mensaje

Task run son core 1

PRUEBA GIBIC NEUMA

CONTEO = 0

(En esta línea cambiara el tipo de mensaje dependiendo del botón presionado)

Al mover el encoder el numero al frente de conteo debe cambiar, en horaria debe aumentar, en dirección anti horaria debe disminuir. Al presionar el botón del encoder, las luces roja, amarilla y verde deben cambiar de estado entre encendido y apagado.

Al presionar cualquiera de los botones pulsadores, su led interno debe cambiar de estado entre encendido y apagado, cambiara el estado de las luces verde, amarilla y roja, y el buzzer sonara por un instante corto, además debe aparecer un mensaje haciendo referencia de que botón se presiono en el display LCD.

Mensajes a mostrar:

Standby Pressed

Stability Pressed

Silence Pressed

Interface Serial, este estará enviando un mensaje por el serial 2 hacia el microcontrolador CONTROL y esperará una respuesta, una vez recibe la respuesta, escribirá por el serial USB “mensaje recibido”

**Tarjeta Control**, microcontrolador Control: En esta tarjeta cargue el script “controlTest.ino” a través de la plataforma <<**Arduino**>>

**Funciones**:

Imprimirá por el serial USB un mensaje de inicialización

"Inicializacion CONTROL"

Encenderá las electroválvulas en orden de 1 a 6 con tiempos de 100 ms

Luego iniciara el muestreo de los canales ADC se 1 a 6 e imprimirá el muestreo cada 50 ms

“Hola, soy CONTROL, ADC1, ADC2, ADC3, ADC4, ADC5, ADC6”

El sistema espera un mensaje proveniente del microcontrolador HMI, al cual responderá a través del serial 2 y escribirá en el serial USB “dato recibido”

**Tarjeta Control**, microcontrolador Monitor: En esta tarjeta cargue el script “MonitorTest.ino” a través de la plataforma <<**Arduino**>>

**Funciones**:

Muestrea las temperaturas dadas por los sensores de temperatura al mismo tiempo que evalúa el cambio de canal con el multiplexor,

Enciende el ventilador

Muestra la temperatura y cada 15 segundos envía un comando de reset para los otros dos microcontroladores