**🛠️ Manual de Uso (Operador)**

**1. Pantalla Principal y Controles**

Al encender el termostato, verás la pantalla de control principal con la siguiente información:

* **Temp:** La temperatura actual que está midiendo el sensor.
* **Set:** El punto de ajuste (setpoint), que es la temperatura deseada.
* **Estado:** Indica si el sistema de control está ACTIVO o APAGADO.
* **Calef:** Muestra si el termostato está CALENTANDO (activando la salida) o está OFF.
* **Salida:** Muestra si la salida está configurada para un SSR o para un MOSFET.

**2. Ajuste de Temperatura (Setpoint)**

Para cambiar la temperatura deseada, simplemente **gira el encoder**. El valor del **Set** cambiará en la pantalla.

* Gira a la derecha: Aumenta la temperatura.
* Gira a la izquierda: Disminuye la temperatura.

ℹ️ **Consejo:** Si giras el encoder rápidamente, el valor cambiará en incrementos de 10°C para agilizar el ajuste.

**3. Ingreso al Menú Principal**

Para acceder a todas las configuraciones, **presiona el botón del encoder** una vez. Esto te llevará al **Menú Principal**.

**4. Navegación en los Menús**

Una vez dentro de un menú, la navegación es muy simple:

* **Gira el encoder:** Para moverte entre las opciones del menú. El símbolo > indicará la opción seleccionada.
* **Presiona el botón:** Para seleccionar la opción marcada y entrar en el siguiente submenú.
* **Mantén presionado el botón (1 segundo):** Para volver al menú anterior o salir de una configuración, guardando los cambios si aplica.

**5. Opciones del Menú Principal**

* **Activar / Desac.:** Permite encender o apagar el control de temperatura. Cuando está apagado, la salida no se activa.
* **Opciones Avanzadas:** Te da acceso a la configuración de PID, ON/OFF, alarmas y tipo de salida.
* **Modo Manual:** Te permite controlar la potencia de salida directamente, sin un control de temperatura.
* **Calibración:** Sirve para ajustar la lectura de temperatura del sensor.
* **Prueba Salida:** Activa la salida de forma manual para probar el calentador.
* **Acerca de:** Muestra información sobre el software.
* **Reiniciar:** Restablece todas las configuraciones a sus valores de fábrica.
* **Salir:** Regresa a la pantalla de control principal.

**🧑‍💻 Manual Técnico (Integrador)**

**1. Conexiones del Hardware**

El circuito se basa en un microcontrolador Arduino con los siguientes componentes principales:

* **Termocupla MAX6675:** Para medir la temperatura.
  + SCK (pin 13)
  + SO (pin 12)
  + CS (pin 10)
* **Encoder Rotativo:** Para la interfaz de usuario.
  + PIN A(clk) (pin 2)
  + PIN B (dt) (pin 3)
  + Botón (pin 4)
* **Salida de Control:** Para activar el sistema de calentamiento.
  + MOSFET o SSR (pin 9, salida PWM)
* **Buzzer:** Para las alarmas sonoras.
  + Buzzer(el código está preparado para buzzer pasivo (pin 8)
* **Pantalla OLED I2C:** Para la visualización.
  + SDA(a4) y SCL(a5) (pines del bus I2C)  
      
    ***Los componentes usados son:***  
    \*Pantalla OLED azul de 128 x 64 píxeles, 1.3 pulgadas  
    \*Driver Dual Mosfet De Potencia 5-36v 15a 400w Dc  
    \*Sensor Max6675(con termocupla +-800c°)  
    \*Placa Experimental Perforada 5x7 Doble capa  
    \*Placa Experimental Perforada 6x4 Doble capa  
    \*Modulo rele ssr 2 salidas(si es que queres usarlo con rele por que por ejemplo manejas algo de 220vAC) (cosa que el mosfet acepta hasta 36v DC)  
    \*Encoder Rotativo 20 Pasos Vuelta   
    \*Arduino Pro mini de 5v”16mhz” (no el de 3.3- 8 MHz)  
      
    ***Librerias usadas***  
    \*U8x8lib(la básica no la 2 por q se come el 60% de la ram del arduino pro mini y no entra el code)”osea texto en plano sin posibilidad de hacer dibujos y mostrar otras cosas q no sea texto”.  
    \*Max6675 DE GIVER  
    \*Encoder.h  
    \*PiD\_v1  
    \*EEPROM

**2. Configuración de Salida**

Este termostato puede funcionar con dos tipos de salida:

* **MOSFET (Salida PWM):** Utiliza una señal PWM (modulación por ancho de pulso) en el pin 9 para regular la potencia de forma gradual. Es ideal para calentadores que pueden ser controlados de esta manera.
* **SSR (Relé de Estado Sólido):** Envía una señal digital (HIGH/LOW) para activar o desactivar el relé. Esta opción es para calentadores que solo requieren un simple encendido/apagado, y la modulación la realiza el PID de forma interna.

La selección de la salida se realiza en el menú Opciones Avanzadas. Esto cambia la lógica de la función updateOutput(), que ajusta el código para el tipo de hardware que estés utilizando.

**3. Lógica del PID y ON/OFF**

* **Modo ON/OFF:** Control simple que enciende o apaga la salida en función de la temperatura y la histéresis configurada. Si la temperatura cae por debajo del setpoint - histéresis, la salida se enciende; si la temperatura sube por encima del setpoint, la salida se apaga. Es útil para sistemas lentos o que no requieren gran precisión.
* **Modo PID:** Es un algoritmo de control más avanzado que ajusta la potencia de salida de manera proporcional, integral y derivativa (Kp, Ki, Kd). Esto permite una regulación más precisa y rápida, evitando el sobrecalentamiento. Los valores de Kp, Ki, y Kd se pueden ajustar en el menú Config PID para optimizar el rendimiento del sistema de calentamiento.

**4. Gestión de Errores y Protecciones**

El código incluye varias protecciones de seguridad:

* **Error de Sensor:** Si la lectura del sensor es inválida (NaN o -9999), el sistema entra en estado de error, apaga la salida y muestra un mensaje en pantalla. Para salir de este estado, debes presionar el botón del encoder.
* **Alarma de Alta Temperatura:** Se activa una alarma sonora y visual si la temperatura excede el highTempAlarm configurable.
* **Límite de Seguridad (OVERSHOOT\_LIMIT):** Es un valor de seguridad fijo (250°C) que, si se supera, apaga la salida y entra en estado de error. Este límite no se puede cambiar desde el menú(hay q reprogramar).
* **Detección de Falla de Calentamiento:** Si la salida está activa pero la temperatura no aumenta 5°C en tres minutos, el sistema asume una falla y entra en estado de error.  
    
  Todo realizado en C++ por “The Hacker”de Byte-ElectronicLab

Todo el código fue escrito desde 0, no es un proyecto “robado de internet”. esta fácilmente editable, ordenado por módulos , el mismo se puede adaptar a otros usos, el proyecto es de código libre, pero siempre aclarando quien es el creador  
P.D: si Tenes dudas del código cárgaselo a Google Gemini y te va ayudar a editarlo