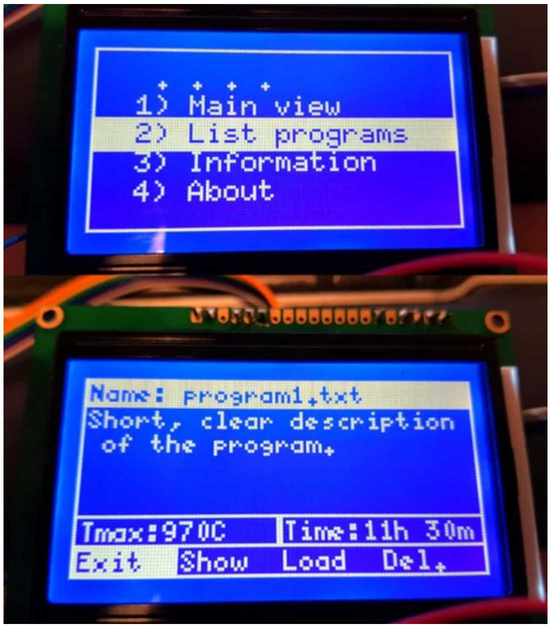
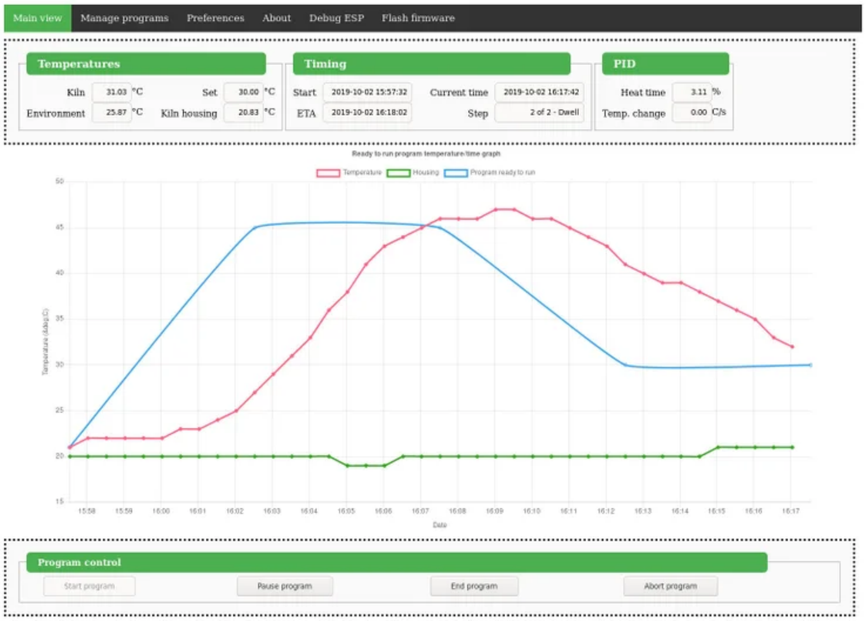
PIDKiln - PID Temperature Controller to Ceramic/Glass/ Metal Kiln

By Adrian Siemieniak in CircuitsMicrocontrollers

Introduction: PIDKiln - PID Temperature Controller to Ceramic/Glass/Metal Kiln





Temperatures

Kiln 31.03 °C Set 30.00 °C
Environment 25.87 °C Kiln housing 20.83 °C

Timing

Start 2019-10-02 15:57:32 Current time 2019-10-02 16:17:42
ETA 2019-10-02 16:18:02 Step 2 of 2 - Dwell

PID

Heat time 3.11 %
Temp. change 0.00 C/s

Program control

This is "yet another" PID temperature controller. I've made this one, because bought one was crappy and I needed something solid. There are two or three other DIY controllers "on the market" but this one, at least I see it this way :), is much more robust and has all features I needed.

Key features:

- interface accessible both from LCD screen and WWW webpage
- unlimited (only by storage) kiln programs number,
- program file size limited to 10KiB (but this is artificial limit - can be extended)
- internal ESP storage for programs, data, logs (perhaps later SD - but I'm not sure yet)
- local preferences on disk, editable with Web interface
- online monitoring, program management, editing, graphs and kiln controll
- build in clock synchronised with NTP servers (if Internet connected)
- safety features build in (temperature run out protection, probe failure, SSR failure, kiln insulation failure)
- simply cool and cheap (comparing to commercially available products) all in one solution

Supplies

- ESP32-Wrover board
- MAX31855 breakout board (or two)
- K-type thermocouple
- DC->AC solid state relay

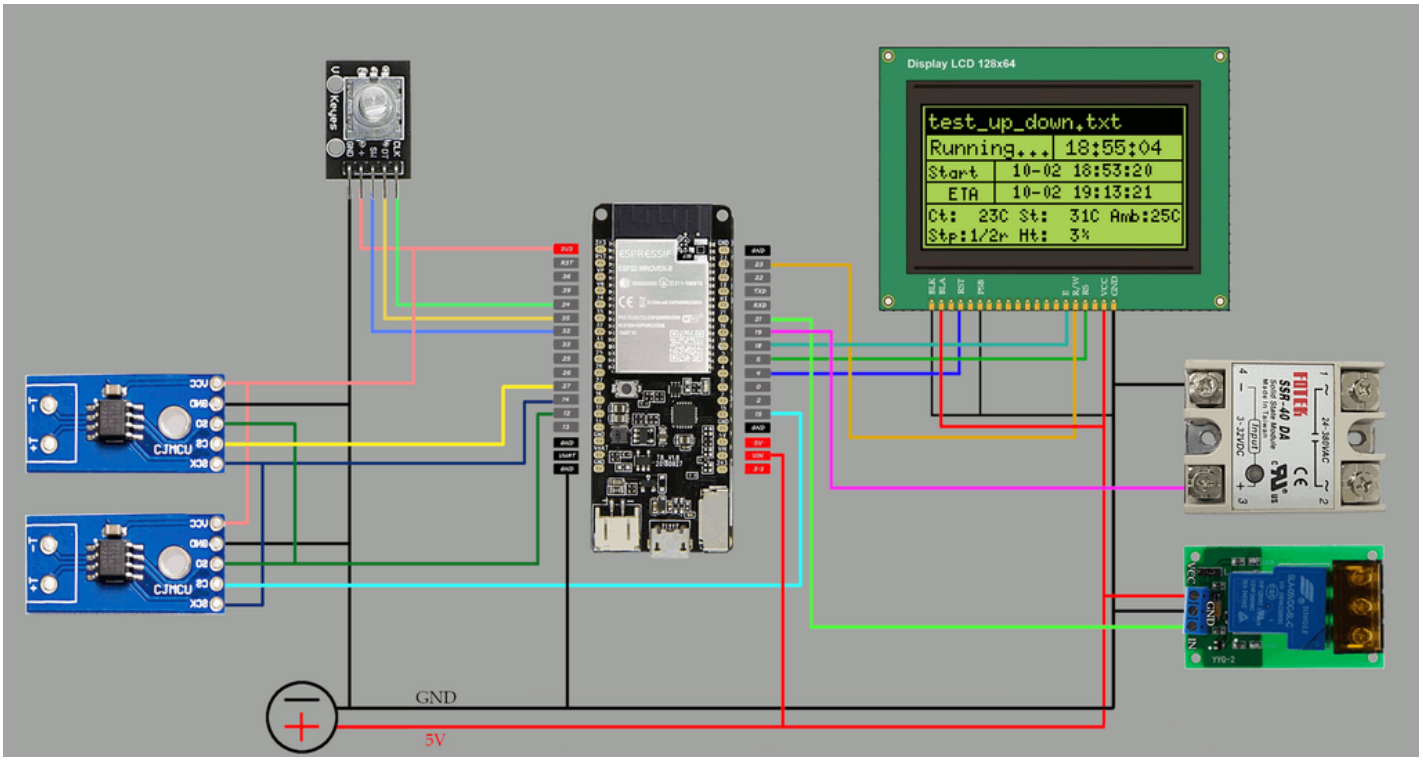
Kind of optional, but recommended:

- 128x65 dot matrix LCD 12864B v2
- Rotary encoder with button

Optional:

- DC/AC secondary relay - like SLA-05VDC-SL-C (240V/30A) mechanical relay
- Additional MAX31855 board with K-type thermocouple for housing temperature measuring
- Perhaps a kiln :)

Step 1: Wiring



LCD

Connected to one of three SPI on ESP32 - called VSPI (MOSI-23, MISO-19, CLK-18, CS-5)

ESP32	LCD
+3.3V	BLA (this can be also +5V if you wish)
GND	BLK
4	RST
GND	PSB
+5V	VCC (This should be – for ESP sake – 3,3V, but my LCD doesn't work with lower voltage. Try first with 3,3V)
GND	GND
5	RS
18	E
23	R/W

Encoder

ESP32	Encoder
+3.3V	5V/VCC
GND	GND
32	Key
34	S2
35	S1

MAX31855

Connected to one of three SPI on ESP32 - called HSPI (MOSI-13, MISO-12, CLK-14) CS-15/27

EPS32	MAX31855 A
+3.3V	VCC
GND	GND
12	SO/DO (slave output/data output)
14	SCK (clock)
15	CS (chip select)

```
EPS32   | MAX31855 B
--------|---------
+3.3V   | VCC
GND     | GND
12      | SO/DO (slave output/data output)
14      | SCK (clock)
27      | CS (chip select)
```

Relays

```
ESP32   | SSR
--------|-------
GND     | GND
19      | IN
```

```
ESP32   | EMR (SLA-05VDC-SL-C)
--------|----------------
GND     | GND
21      | IN
5-48V*  | VCC
```

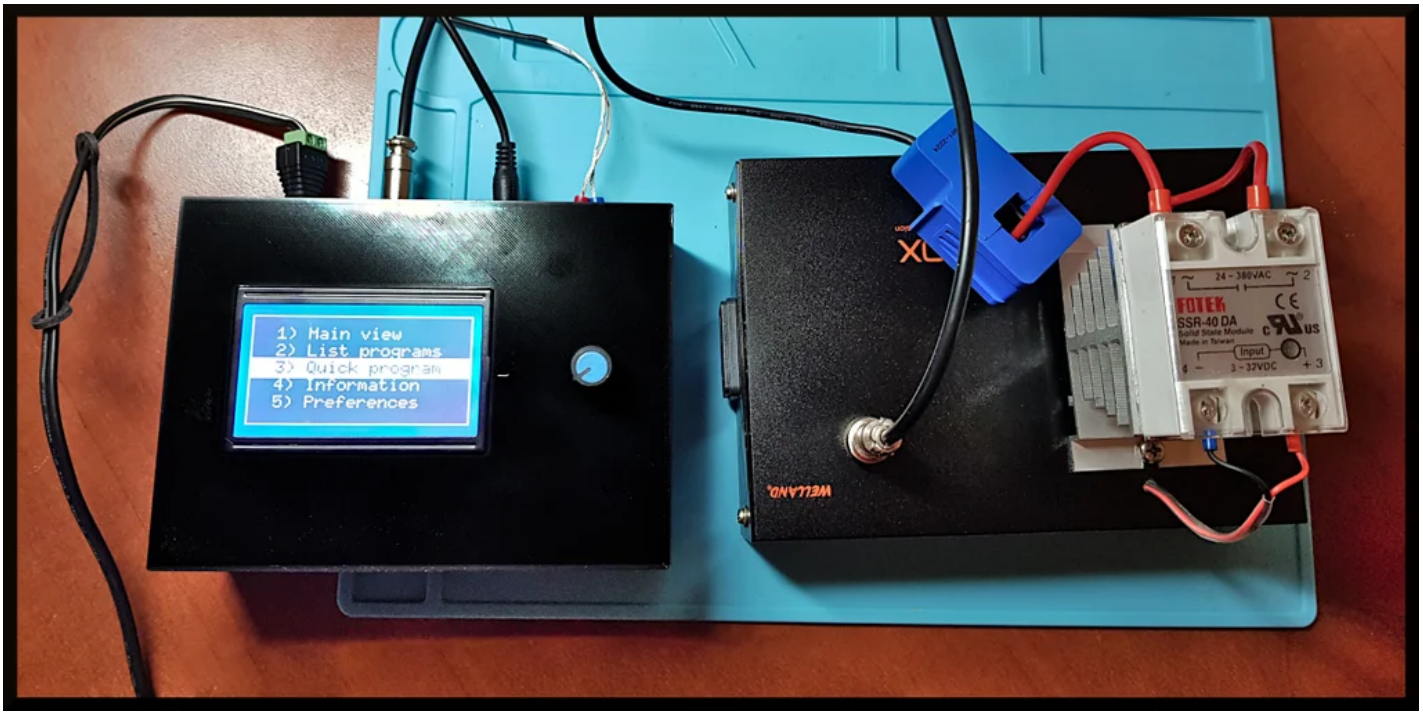
* Do not connect 5V from ESP - use external source. This can be any 5V-48V power supply with around 1W power.

Required source code is part of the Github: https://github.com/Saur0o0n/pidkiln

Some more information you can find on my webpage: https://adrian.siemieniak.net/portal/tag/PIDKiln/

3d printable case: https://www.thingiverse.com/thing:3907643

Step 2: Assembled Controller



This is fully assembled controller - on the left, and remote relay box (with power meter) - on the right. I've split those two elements because SSR relay can get quite hot and I wanted to have mains voltage away from logic boards.

PIDKiln is in 3d printed box, for relays I've used old 3,5" aluminium disk enclosure and some 3d printed sides. I wanted this to be metal as an additional heatsink for SSR.