

# ELECTRONOOBS



Temperature PID controller - Arduino

(<https://www.electrogeek.com/electrogeek/pid/>)

(<https://twitter.com/ELECTRONOOLS>)

## PART 3 PID control

So now that we know how to read the real temperature, let's mount this next schematic and control the power applied to the heating element with a MOSFET. I mount the circuit on a breadboard once again and upload the next code. This second code has the PID algorithm already created. We read the temperature, calculate the error, sum the PID values and create a PWM signal on digital pin D3 that will be applied to the MOSFET. I set the desired temperature at 100 degrees and use the LCD to print the set value and the real temperature.

## ADVERTISE

PCBWay — Custom PCB Service —

**Only \$5**  
for 10 boards

Rogers, HDI, aluminum and rigid-flex PCB are available now

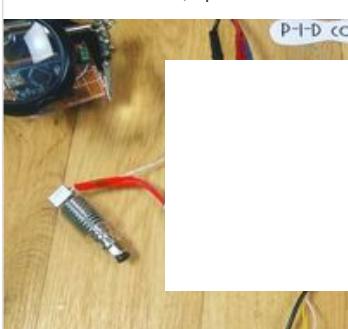
Production time 24 hours

(<https://www.pcbway.com?from=electrooobs.com>)

# PID controller for temperature



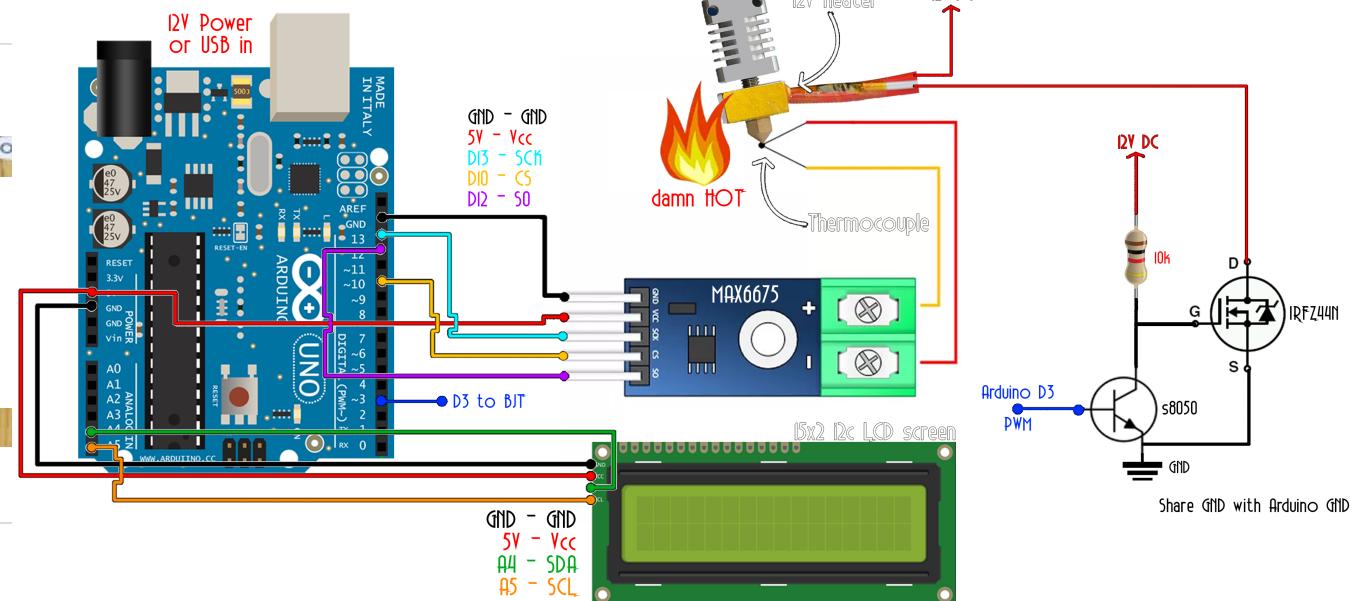
electronoob...  
Barcelona, Spain



[到 Instagram 查看更多內容](#)

457 個讚

新增留言.....



Downlaod schematic here:  [DOWNLOAD](#) (eng\_arduino\_tut24\_sch2.php)

協作型機器人是否適合  
本電子書將  
協助您一探究竟

您是否缺少  
部署協作型機器人的  
完整規劃？  
了解成功部署之道

Code for fixed PID control

Ok so the code below is a bit large. But don't worry. It is very easy. We set a variable setpoint at 100 degrees for this example. Then we read the thermocouple real temperature value as in the past example. Then we use 3 constants and calculate the PID sum. Depending on that value we create a PWM signal on pin D3 and apply it to the MOSFET gate using a BJT driver.

Download PID code without control here: [!\[\]\(4729e517bc6a7cd81c8025b9646574fb\_img.jpg\) DOWNLOAD](#)  
(eng\_arduino\_tut24\_code2.php)

```
/*
 * Max6675 Module ==> Arduino
 * CS ==> D10
 * SO ==> D12
 * SCK ==> D13
 * Vcc ==> Vcc (5v)
 * Gnd ==> Gnd */
//LCD config (i2c LCD screen, you need to install the Liqui
```



([https://s.click.aliexpress.com/e/\\_9jqPR0?  
bz=300\\*250](https://s.click.aliexpress.com/e/_9jqPR0?bz=300*250))



(<https://bit.ly/2ZNWgqy>)



([https://s.click.aliexpress.com/e/\\_AF7EeE?](https://s.click.aliexpress.com/e/_AF7EeE?))



As you can see the temperature stays at that value. But that is after trying a lot of PID constants and that is the tricky part of this project. So, what you will have to do is try your own values till you get the correct ones. I advise you to start with the I and D values equal 0 and then increase those values slowly till you get good results.

Here on my oscilloscope I have the PWM signal of the MOSFET connected.

bz=500\*500)

協作型機器人是否適合您的生產

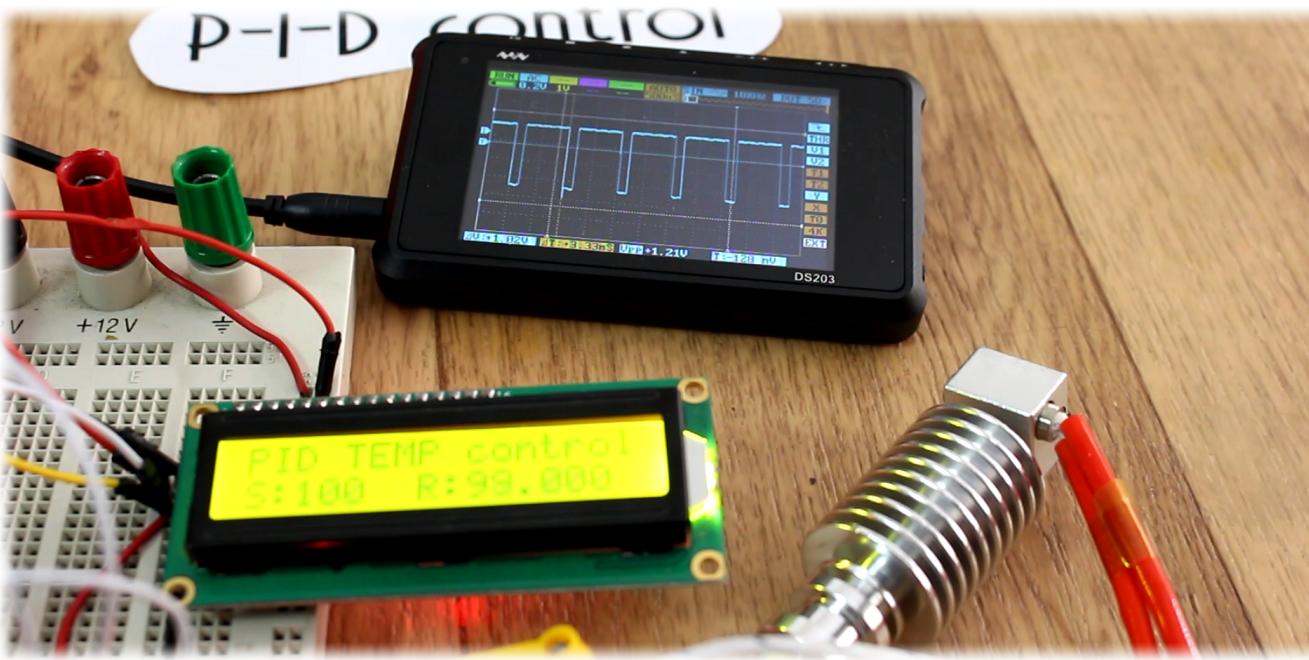
本電子書將  
協助您一探究竟



改善製造業難題困難重重嗎

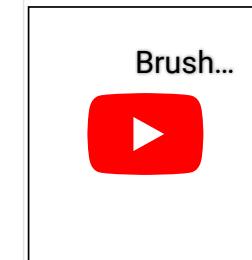
協作型機器人提供的  
5個解決之道



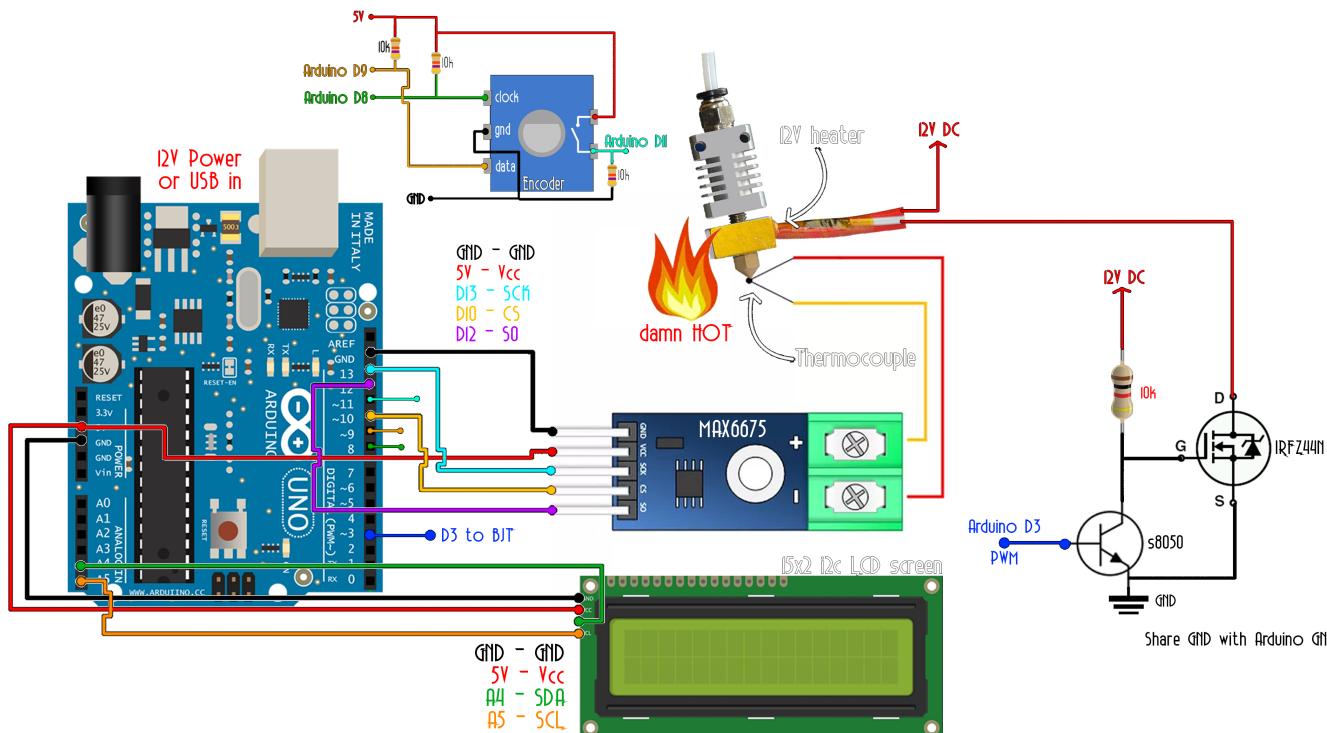


At the beginning, till the system reaches the desired value the pulse has a small width since I use a BJT to activatthe the N channel gate, so the mosfet is activated with a LOW value in this case. Once the set value is reached it starts to wamble around and by that maintaining the temperature. As you can see, if I try to cool down the heating element by blowing air with this tube, the PWM signal width get's lower in order to keep the same value. So, the control works.

## Part 4 PID + rotary control



Now, all this system needs is some sort of control together with the LCD screen in order to view and also be able to set the desired temperature value as this commercial PID controller has the set and up and down buttons. For that I'll use the rotary encoder. It has a push button integrated so I can use it to enter the setpoint menu and increase or decrease the value.



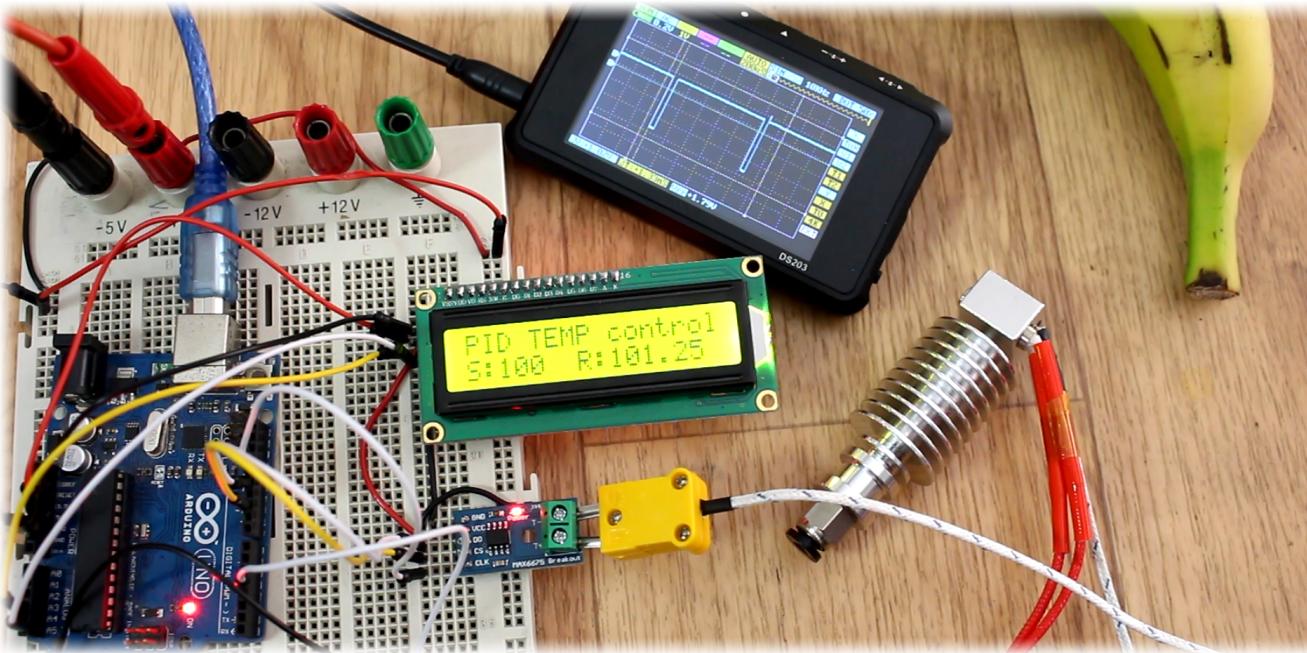
(eng arduino tut24 sch3.php)

This above is the final schematic of this project. We have an LCD screen to print the values, the rotary encoder with push button insede for control, the thermocouple with the MAX 66 75 module, the MOSFET and the BJT as a driver circuit that will control the

power and the heating element. Make sure that the thermocouple is touching the heating element in order to know the real value. I mount everything on the breadboard and now let's test the new code, which by the way, you can also download from a below.

Download final code here:  [DOWNLOAD](#) (eng\_arduino\_tut24\_code3.php)

The default value is now 0 degrees. Press the set button of the rotary encoder Rotate it to the left in order to increase or decrease the temperature value. Press the set button once again and now you can set the P constant for the PID control. Press once again and select the I value. Finally, press the button again and select the D value. Now press the button and exit the menu and the new settings are stored. I set it to 100 degrees and now the real read starts increasing till it reaches the desired value and it stays there. When we reach the desired value you can see the PWM wobbling in order to maintain that value.



There you go our PID of temperature works. I could 3D print a case for this project just as the commercial one has, but, since I'll use this project for my soldering station in a future tutorial, I won't do that now. In a future project, I'll use this 12V soldering iron with a thermocouple inside and make my own soldering station so stay tuned for that guys.

Have in mind that today's project is for **only DC power control**, so only DC heating elements will work. I'll also make a tutorial on a temperature PID control, but for AC 220V voltage using a TRIAC control as in the past TRIAC tutorial.

If my videos/tutorials help you and you would like to help my projects, I have a Patreon campaign (<https://www.patreon.com/ELECTRONOOBS>). I would really appreciate that guys and it will help my workshop for other new projects and keep this channel/webpage going. BTW, thanks to all my Patreons.

I hope that you've enjoyed this small tutorial on PID control. If you have any question about this video or any other, just leave it in the comment section below or on my Q&A page ([eng\\_preguntas.php](#)). Also, don't forget to subscribe and watch all of my other great tutorials. Remember, if you consider helping my projects check my Patreon page as well.



## FPC test jig

Use FPC connector when you test LCD module or something repeat

YOKOWO

O



**BACK**

([eng\\_arduino\\_tut24.php](#))

**18 comments**Sort by **Oldest**

Add a comment...

**Serkan Özkan**

Hello. I am trying the codes now and after every loop of menu finishes, all values are increased by 1 automatically. I inspected the codes and at the end of the codes there are  $kp=kp+1$ ,  $ki=ki+1\dots$  I deleted the but then the menu was invisible. Please could you fix this?

[Like](#) · [Reply](#) · 4y**Santosh Babu**

I AM ALSO FACING SAME PROBLEM DID U GOT ANY SOLUTION

[Like](#) · [Reply](#) · 3y**Waseem Abbas**

Santosh Babu did problem had solved?

[Like](#) · [Reply](#) · 2y**Lavish Kumar**

Sir can you build GSM control car

[Like](#) · [Reply](#) · 4y**Robert Bushell**

"I'll also make a tutorial on a temperature PID control, but for AC 220V voltage using a TRIAC control as in the past TRIAC tutorial"  
Any progress?

[Like](#) · [Reply](#) · 3y

ELECTRONOOPS 2022  
Keep up you guys!