Laporan Praktikum Alat Pemanas Filamen



Disusun oleh:

Deyan triyanda (22106020034)

Program Studi Fisika Fakultas Sains Dan Teknologi Universitas Islam Negeri Sunan Kalijaga Yogyakarta

2024

1. Alat dan Bahan

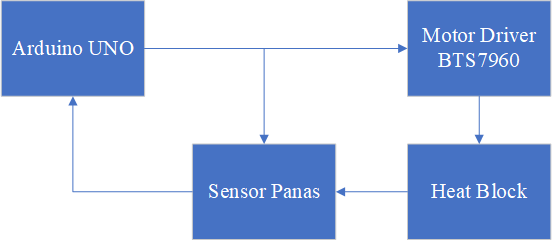
Berikut merupakan alat dan bahan yang kami gunakan pada saat merangkai alat pemanas filamen

Tabel 1.1 Alat dan Bahan

|  |  |
| --- | --- |
| Alat dan bahan | Fungsi |
| Arduino uno | Memprogram dan mengontrol perangkat elektronik. |
| Breadboard | Menghubungkan komponen elektronik tanpa menyolder. |
| Power supply | Sumber daya listrik untuk memberikan tegangan dan arus ke rangkaian |
| Laptop | Untuk memprogram Arduino dan memonitor sistem yang dibuat. |
| Motor driver BTS7960 | Mengatur atau membatasi arus yang masuk ke elemen pemanas (*heat block*) |
| *Heat Block* | Alat untuk memanaskan filamen |
| Resistor 100k Ohm | Komponen yang membatasi aliran arus listrik dalam rangkaian. |

1. Diagram blok pemanas filamen

Berikut merupakan diagram block pada rangkaian alat pemans filamen



Gambar 1.1 Diagram Blok

Penjelasan dari Gambar 1.1 Diagram Blok diatas

1. Pengukuran Suhu: Sensor panas membaca suhu dari *heat block* dan mengirimkan informasi tersebut ke Arduino.
2. Pengolahan Data: Arduino memproses data suhu dan memutuskan apakah perlu mengubah daya yang dikirim ke *heat block.*
3. Pengendalian Daya: Arduino mengirimkan sinyal ke motor driver BTS7960 untuk mengatur keluaran daya ke *heat block*, sehingga suhu dapat dikendalikan secara efisien.
4. Koefisien Umpan Balik: Sistem ini menggunakan mekanisme umpan balik di mana sensor panas terus memantau suhu, dan Arduino menyesuaikan daya yang dikirim ke heat block berdasarkan pengukuran tersebut.
5. Proses Perangkaian Alat

Berikut merupakan prosen perangkaian alat pemanas filamen

1. Alat dan bahan disiapkan
2. Tentukan diagram blok yang akan digunakan
3. Pin 9 digital pada arduino disambungkan pada pin motor driver BTS790
4. Pin A0 pada arduino disambungkan pada sensor panas
5. Pada breadboard ditambahkan resistor 100k Ohm sebagai penghambat tegangan yang masuk ke sensor panas
6. Sensor panas dimasukan *heat block*
7. Pengujian risetime dan overshoot sistem kendali pemanas dengan variasi *Kp, Ki, Kd*

Tabel 1.2 Data pengujian

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Percobaan | *Kp* | *Ki* | *Kd* | *Rise time* | *Over shoot* | *Steadystate eror* |
| 1 | 1 | 0 | 0 | 0 | 0 |  |
| 2 | 1 | 0.008 | 0 | 271, 97 detik | 2, 63 Celcius |  |
| 3 | 1 | 0.008 | 0,001 | 296, 05 detik | 2, 06 Celcius |  |

Percobaan 1: Nilai *Kp, Ki, dan Kd* semuanya 0. Ini berarti tidak ada yang dilakukan pada sistem, sehingga tidak ada perubahan (semua hasil tetap 0).

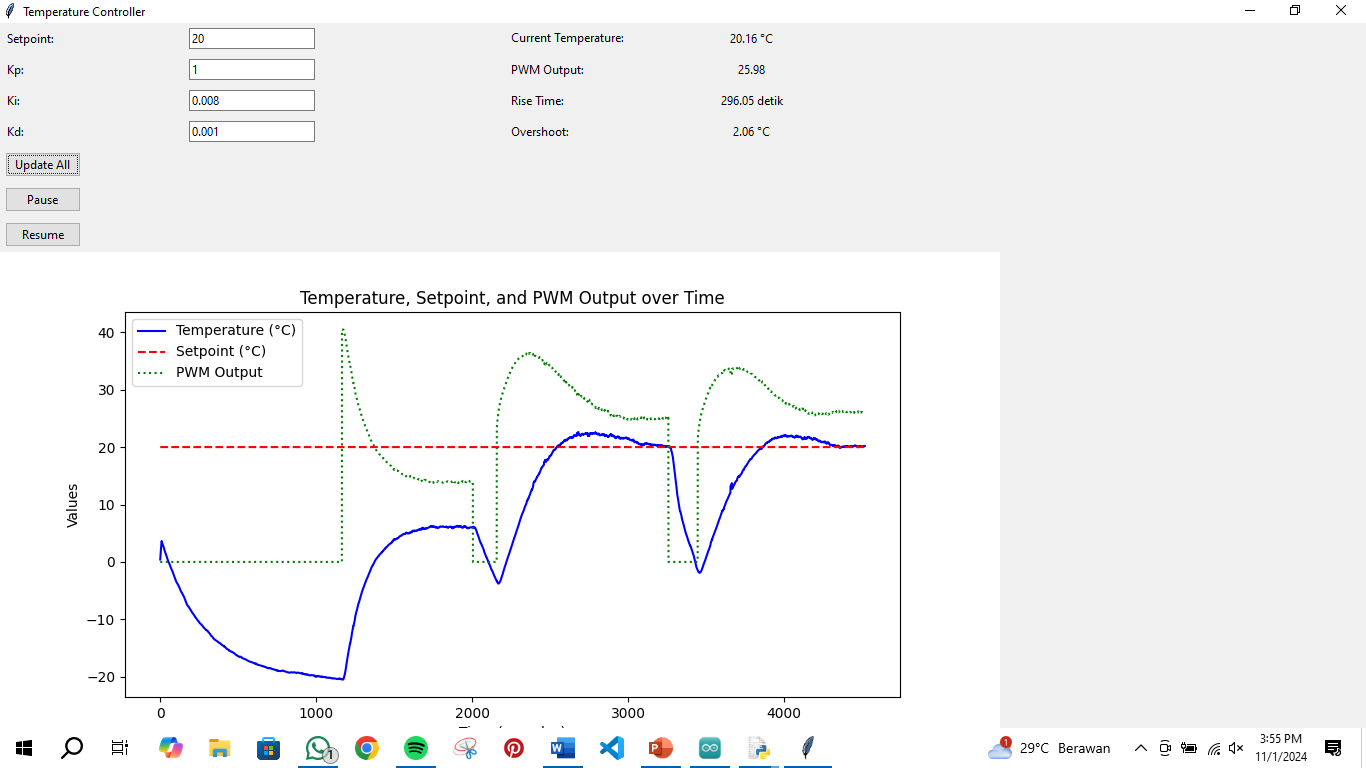
Percobaan 2: Nilai *Kp* adalah 1, *K*i adalah 0.008, dan *Kd* adalah 0. Ini adalah konfigurasi kendali PI (Proportional-Integral) tanpa pengaruh diferensial. Pada percobaan ini:

1. Rise time atau waktu naiknya adalah 271,97 detik.
2. Overshoot atau lonjakan melewati setpoint sebesar 2,63°C.

Percobaan 3: Nilai *Kp* adalah 1, *Ki* adalah 0.008, dan *Kd* adalah 0.001. Merupakan konfigurasi kendali PID (Proportional-Integral-Derivative) . Pada percobaan ini:

1. Rise time lebih lambat dibanding percobaan 2, yaitu 296,05 detik.
2. Overshoot sedikit lebih kecil, sebesar 2,06°C.

Dari perbandingan percobaan 2 dan 3, bahwa penambahan *Kd* mengurangi nilai Overshoot namun memperlambat Rise time.



Gambar 1.2 Grafik termperatur

Grafik pada gambar 1.2 Grafik temperatur menunjukkan bagaimana sistem koefisien PID bekerja dalam mengatur suhu (Temperature) agar mendekati setpoint yang diinginkan, dengan memodulasi output PWM (Pulse Width Modulation). Rise Time pada grafik menunjukkan waktu yang dibutuhkan suhu untuk mulai meningkat ke arah setpoint setelah sistem diaktifkan. Rise time dalam grafik ini tercatat sekitar 296,05 detik. Overshoot pada grafik terlihat bahwa suhu melebihi setpoint sebesar 2,06°C, menunjukkan bahwa ada energi berlebih yang masuk ke dalam sistem sebelum suhu mulai stabil. Pada grafik ada osilasi yang menunjukkan bahwa sistem koefisien PID perlu penyetelan untuk memperbaiki stabilitas dan mengurangi overshoot.

Pada grafik, penggunaan *Ki* (Integral) digunakan ketika sistem tidak bisa mencapai setpoint, dan ada kesalahan tetap (steady-state error) yang tidak bisa dihilangkan, tetapi dengan koefisien proporsional (*Kp*). Misalnya suhu sudah mulai mendekati setpoint tapi tetap tidak sampai ke target, *Ki* bisa untuk mengoreksi dan mengatur suhu ke setpoint.

Sementara itu, penambahan nilai *Kd* (Derivative) dipakai untuk mengurangi osilasi dan overshoot. Menggunakan *Kd* jika sistem sering osilasi berlebihan atau overshoot, di mana suhu atau parameter lain sering melewati setpoint dan memantul kembali. *Kd* berguna untuk sistem yang responsif, terutama jika perubahan mendekati setpoint bisa menyebabkan fluktuasi yang besar.

Kesimpulannya, Untuk meningkatkan stabilitas dan mengurangi osilasi yaitu dengan cara Mengurangi *Kp* yang dapat mengurangi koefisien dan menurunkan osilasi dengan cara meningkatkan *Kd* untuk meredam osilasi dan membuat sistem lebih stabil. Mengatur nilai *Ki* Jika terlalu tinggi dapat menyebabkan osilasi lebih, Jika terlalu rendah, akan memperlambat respons terhadap kesalahan steady-state. Mengatur parameter PID diperlukan untuk mencapai keseimbangan antara kecepatan respons dan stabilitas sistem.