

## Purwarupa Sistem Kendali PID: Studi Kasus Kendali Suhu Ruang

Dyah Aruming Tyas<sup>\*1</sup>, Raden Sumiharto<sup>2</sup>

<sup>1</sup>Prodi Elektronika dan Instrumentasi, FMIPA UGM, Yogyakarta

<sup>2</sup>Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: <sup>\*</sup>[dyah.at09@gmail.com](mailto:dyah.at09@gmail.com), <sup>2</sup>[r\\_sumiharto@ugm.ac.id](mailto:r_sumiharto@ugm.ac.id)

### Abstrak

*Sistem pengendalian otomatis banyak memberikan keuntungan bagi manusia. Selain dapat mempercepat waktu kerja, pengendalian otomatis juga dapat mengurangi kesalahan yang dilakukan oleh manusia (human error) dan meningkatkan efektifitas kerja. Salah satu aplikasi pengendalian otomatis adalah pengendalian suhu dengan tujuan memperoleh suhu yang diinginkan dalam waktu yang relatif singkat serta dapat mempertahankan suhu dalam kondisi stabil meski terdapat gangguan.*

*Sistem pengendalian suhu dapat diimplementasikan menggunakan board Arduino dan LabVIEW. Pada sistem pengendalian suhu ini digunakan tiga buah sensor suhu LM35. Ketiga sensor suhu tersebut diletakkan pada posisi yang berbeda di dalam plant untuk mengetahui penyebaran suhu yang terjadi pada plant. Salah satu sensor tersebut selain untuk mengetahui persebaran suhu juga digunakan sebagai feedback. Data dari sensor kemudian masuk ke board Arduino, yang pada sistem pengendalian suhu ini berfungsi sebagai akuisisi data. Untuk menjaga suhu di dalam plant, digunakan heater untuk menaikkan suhu jika suhu plant dibawah set point dan kipas ventilasi untuk menurunkan suhu jika suhu di dalam plant melebihi set point. Untuk mengendalikan heater dan kipas ventilasi ini digunakan kendali PID yang diimplementasikan melalui LabVIEW. Pada kendali PID diperlukan tiga parameter yang bisa didapatkan dengan penalaan Ziegler-Nichols.*

*Hasil dari penelitian menunjukkan bahwa respon plant adalah osilasi kontinyu dan bisa diatasi dengan menggunakan penalaan osilasi kontinyu Ziegler-Nichols. Dengan menggunakan kendali PID pada LabVIEW dan penalaan parameter menggunakan metode Ziegler-Nichols, untuk set point suhu 40°C didapatkan respon plant sebesar  $(40 \pm 0,5)$  °C. Dengan kata lain pengendalian suhu menggunakan kendali PID dan metode Ziegler-Nichols memiliki error dibawah 2% yang menunjukkan bahwa sistem kendali suhu ini layak untuk digunakan.*

**Kata kunci**— suhu, kendali PID, LabVIEW, board Arduino, Ziegler-Nichols, feedback

### Abstract

*Automatic control system provides many benefits for humans. Besides being able to speed up work time, automatic control can also reduce the errors made by humans (human error) and increase work effectiveness. One application is the automatic control of temperature control in order to obtain the desired temperature in a relatively short time and can maintain the temperature in a stable condition despite the presence of interference.*

*Temperature control system can be implemented by arduino and LabVIEW. The temperature control system sensor uses three LM35. Three temperature sensors were placed at different positions within the plant to determine the temperature distribution that occurred in the plant. One of the sensors was used determine the temperature distribution and used as feedback. Temperature from the sensors were processed in the Arduino board, at which this temperature control system serves as a data acquisition. To keep the temperature inside the plant, the temperature control system used heater to raise the temperature when the temperature was below the set point and the plant ventilation fan to lower the temperature if the*

temperature inside the plant was above the set point. To control the heater and ventilation fan PID controller was implemented through LabVIEW. PID control takes three parameters that can be obtained using the Ziegler-Nichols tuning.

Results from the study show that the response of the plant is a continuous oscillation and it can be overcome by using continuous tuning Ziegler-Nichols oscillation. The LabVIEW PID control and the tuning parameters using Ziegler-Nichols method set to 40°C set point plant responses result in  $(40 \pm 0,5)$  °C. In other words, temperature control using PID control and Ziegler-Nichols method has an error rate of 2%, which indicates that the temperature control system is feasible to be used.

**Keywords**— Temperature, PID control, LabVIEW, Arduino board, Ziegler-Nichols, feedback

## 1. PENDAHULUAN

Sistem kendali merupakan bagian yang terintegrasi dari sistem kehidupan modern saat ini. Sebagai contoh kendali pesawat, robot, suhu ruang, mesin cuci, dan lain sebagainya. Dengan sistem kendali memungkinkan adanya sistem yang stabil, akurat dan tepat waktu. Sistem kendali dapat dikatakan sebagai hubungan antara komponen yang membentuk sebuah konfigurasi sistem, yang akan menghasilkan tanggap sistem yang diharapkan. Sistem kendali dibedakan menjadi kendali manual dan otomatis [1].

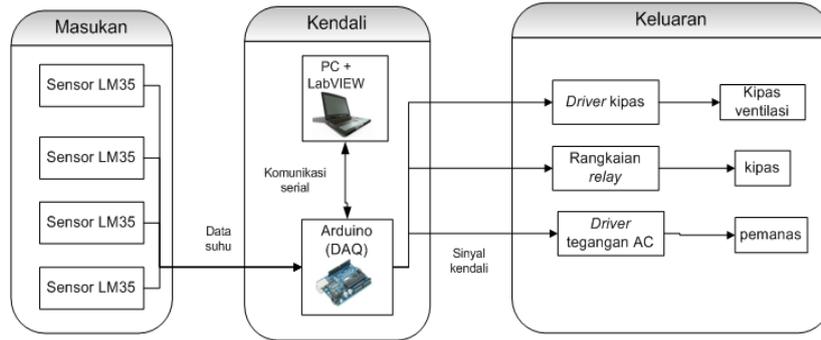
Sistem pengendalian otomatis banyak memberikan keuntungan bagi manusia. Selain dapat mempercepat waktu kerja, pengendalian otomatis juga dapat mengurangi kesalahan yang dilakukan oleh manusia (*human error*) dan meningkatkan efektifitas kerja. Salah satu aplikasi pengendalian otomatis yang dapat kita jumpai dalam kehidupan sehari-hari adalah pengendalian suhu. Suhu sebuah ruangan perlu dikendalikan dan tetap terjaga agar dapat dimanfaatkan untuk keperluan tertentu.

Pada sistem pengendalian suhu, proses pemantauan perubahan suhu dalam ruangan sangatlah penting. Untuk memudahkan operator atau *user* dalam memantau suhu diperlukan sebuah HMI. Salah satu perangkat lunak untuk membuat HMI adalah LabVIEW. Dengan software LabVIEW dapat dirancang sistem pengendalian sekaligus beserta *user interface* atau HMI [2].

## 2. METODE PENELITIAN

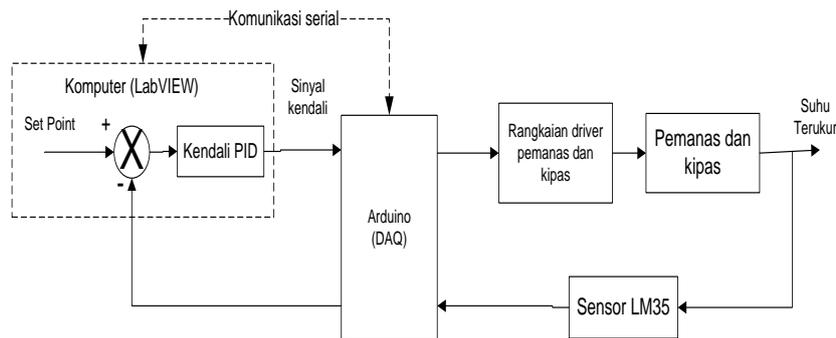
### 2.1. Rancangan Sistem Secara Keseluruhan

Sistem kendali PID berbasis LabVIEW ini bertujuan untuk dapat mengendalikan suhu yang berada dalam suatu ruangan, baik dalam kondisi bebas dari gangguan ataupun terdapat gangguan. Untuk dapat mengetahui suhu dalam ruang digunakan sensor suhu LM35 [3] yang terhubung dengan *board* Arduino [4]. Gambar 1 menunjukkan diagram balok perangkat keras yang digunakan. Digunakan empat buah sensor suhu pada sistem. Satu buah sensor untuk mengukur suhu dari pemanas dan ketiga sensor lainnya diletakkan pada beberapa tempat yang berbeda dalam maket ruangan. Ketiga posisi penempatan sensor dipilih untuk mewakili pembacaan suhu pada daerah yang dekat pemanas, daerah depan pemanas dan bagian atas pemanas, sehingga diperoleh gambaran penyebaran suhu dalam maket ruangan tersebut.



Gambar 1 Blok diagram rancangan sistem secara keseluruhan

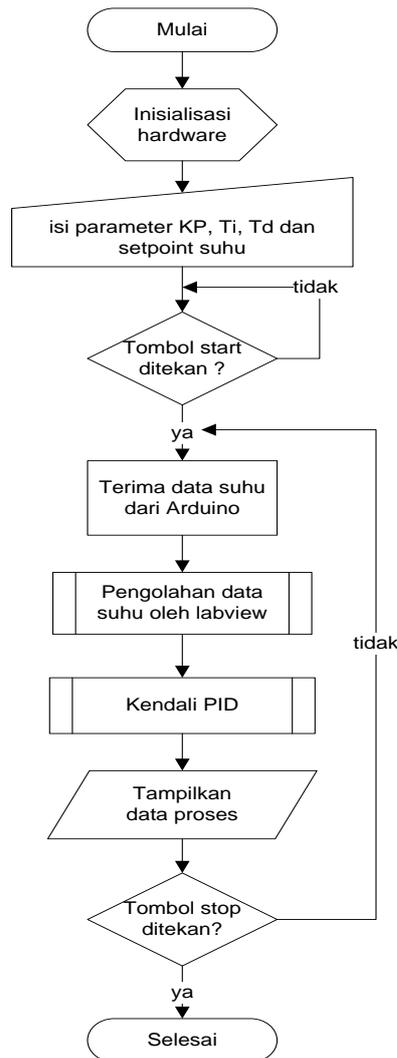
Keluaran dari sensor suhu yang berupa data analog akan dikonversi oleh *board* Arduino menjadi data digital. Data tersebut selanjutnya dikirim ke LabVIEW dengan menggunakan komunikasi serial untuk diolah [5]. Berdasarkan selisih antara suhu terukur dengan suhu yang diinginkan, kendali PID pada LabVIEW akan menghasilkan sinyal kendali digital yang kemudian dikirim ke pin-pin PWM pada *board* Arduino. Dari pin-pin tersebut rangkaian driver (pengendali) tegangan AC untuk pemanas dan rangkaian *driver* kipas pembuang udara panas mendapatkan masukan untuk mengendalikan tegangan dan arus yang diperoleh pemanas dan kipas. Berdasarkan masukan tersebut, maka pemanas dan kipas akan aktif bekerja. Sistem kendali PID berbasis LabVIEW secara keseluruhan dalam bentuk diagram balok ditampilkan pada Gambar 2.



Gambar 2 Diagram blok sistem kendali

Gambar 2 menunjukkan diagram balok sistem pengendalian suhu ruang secara keseluruhan. Sebuah nilai suhu yang diinginkan (*set point*) diberikan oleh *user* melalui antarmuka (HMI) pada komputer. Nilai tersebut kemudian dibandingkan dengan suhu terukur hasil pembacaan sensor suhu LM35 yang telah dikonversi oleh *board* Arduino lalu dikirimkan melalui komunikasi serial ke komputer dan diolah oleh labVIEW untuk memperoleh nilai *error* yang akan diterima oleh kendali PID.

Keluaran berupa sinyal kendali selanjutnya dimanfaatkan oleh *plant* yang berupa pemanas dan kipas untuk menghasilkan suhu ruangan yang diinginkan. Karena tegangan dan arus yang dihasilkan oleh sinyal kendali sangat terbatas dan tidak dapat memenuhi tegangan yang dibutuhkan oleh pemanas maka dibutuhkan rangkaian driver pemanas dan kipas. Rangkaian ini menggunakan *pulse width modulations* (PWM), dimana *duty cycle* PWM ini diatur oleh sinyal keluaran kendali PID. Dengan demikian dapat dikatakan bahwa suhu ruangan yang ada dapat dikendalikan oleh kendali PID.

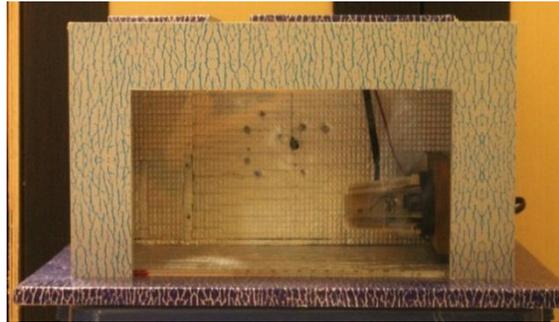


Gambar 3 Diagram algoritma sistem

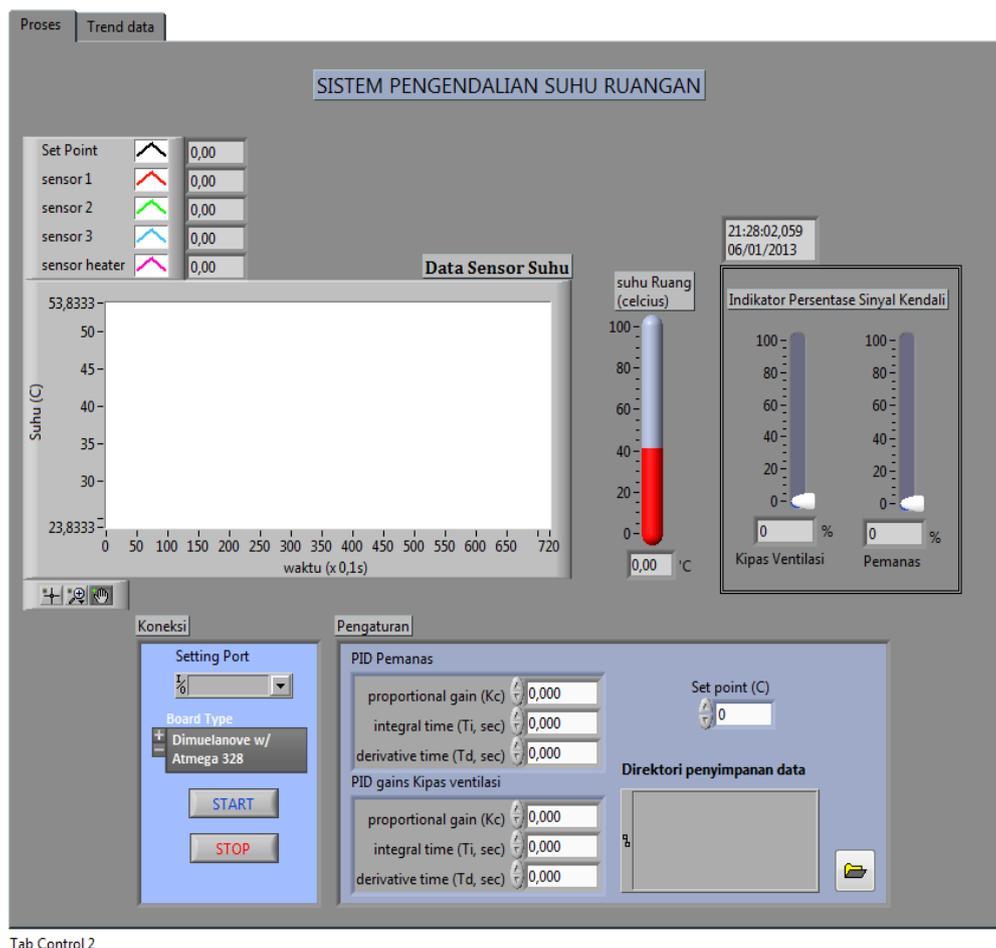
## 2.2. Implementasi

Sistem yang telah dirancang pada bab sebelumnya, terdiri dari bagian perangkat keras (*hardware*) dan perangkat lunak (*software*). Implementasi perangkat keras terdiri dari implementasi *plant* dan implementasi elektronik sedangkan implementasi perangkat lunak meliputi pembuatan program dengan menggunakan LabVIEW pada komputer.

Implementasi *plant* ini telah dibuat sesuai dengan apa yang telah dirancang sebelumnya. Gambar 4 menunjukkan implementasi *plant* untuk sistem pengendalian suhu dengan pemanas berupa *hairdryer* (AC 220V, 50Hz, 400Watt) yang ada di pasaran, dan kipas pembuang udara panas (DC 12 volt, 0,15A) dan sensor suhu LM35. Untuk mendeteksi suhu digunakan empat buah sensor yang diletakkan dalam *plant*. Satu buah sensor diletakkan menempel pada elemen pemanas dan ketiga sensor lainnya (sensor 1, sensor 2 dan sensor 3) diletakkan pada dinding *plant* yang secara berurutan di belakang atas pemanas, samping tengah pemanas dan depan atas pemanas.

Gambar 4 Implementasi *plant*

Tampilan antarmuka atau *front panel* pada LabVIEW dibuat sesuai dengan rancangan sistem yang terdiri dari pengaturan port, pengaturan parameter kendali, trend suhu, lokasi penyimpanan data serta indikator proses. Terdapat dua bagian antarmuka pada sistem, yaitu halaman pertama yang ditunjukkan pada Gambar 5. Gambar 5 merupakan halaman antarmuka yang menunjukkan berlangsungnya proses pengendalian suhu ruangan sedangkan halaman kedua berisi trend data suhu ruangan dan trend data sinyal keluaran kendali selama proses berlangsung. Dari antarmuka yang ditunjukkan pada Gambar 5 operator dapat menentukan *set point* suhu yang diinginkan, parameter kendali yang akan digunakan serta lokasi penyimpanan data. tombol Start dan Stop untuk mengawali dan mengakhiri proses ditampilkan pada bagian ini.



Gambar 5 Implementasi HMI

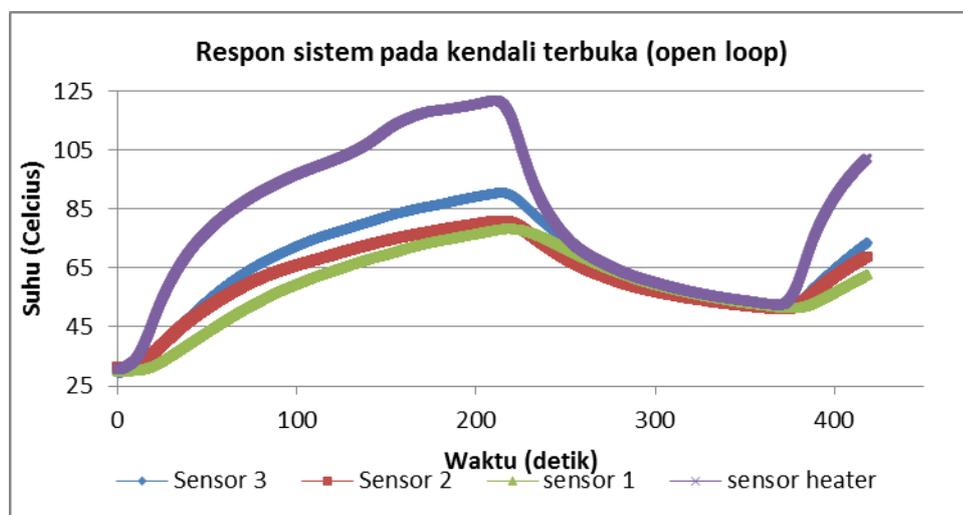
### 2.3. Penentuan parameter PID menggunakan metode Ziegler-Nicols

Melihat respon kendali hidup-mati yang terus berosilasi kontinyu maka dilakukan penalaan menggunakan metode osilasi kontinyu. Dalam penggunaan metode penalaan Ziegler Nichols ini, langkah yang harus dilakukan adalah pengujian respon *plant* dengan nilai  $K_p$  mulai dari nilai minimum hingga memperoleh nilai  $K_u$  (ultimate gain) yaitu  $K_p$  yang dapat menghasilkan respon osilasi kontinyu pada *set point*[1]. Nilai parameter integrator diatur tak berhingga dan parameter *derivative* diatur nol ( $T_i = \infty$  dan  $T_d = 0$ ). Dalam sistem yang dibuat, terdapat dua kendali PID yang masing-masing akan mengendalikan pemanas dan kipas pembuang udara panas. Oleh karena itu diperlukan penalaan parameter pada tiap kendali PID. Dalam sistem ini Sensor 3 merupakan sensor yang digunakan sebagai umpan balik (variabel proses) pada kendali PID yang digunakan, sedangkan sensor *heater* merupakan sensor yang menunjukkan suhu pada elemen pemanas.

## 3. HASIL DAN PEMBAHASAN

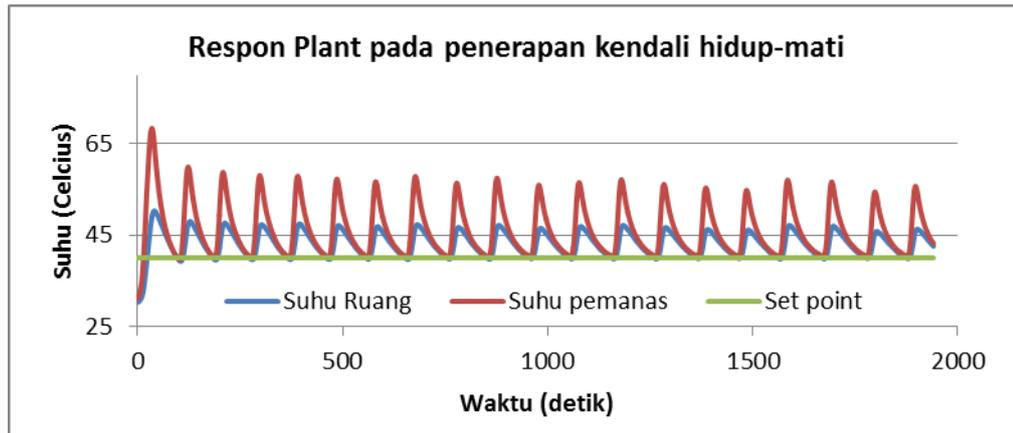
### 3.1. Hasil pengujian

Pengujian dilakukan terhadap pembacaan sensor suhu LM35 dan proses pengendalian sistem secara keseluruhan menggunakan parameter pengendalian, sehingga diperoleh parameter pengendalian yang terbaik. Terdapat empat sensor yang digunakan sistem yang dibuat, yaitu sebuah sensor diletakkan pada pemanas (warna ungu) dan tiga sensor lainnya (Sensor 1 warna hijau, Sensor 2 berwarna merah dan Sensor 3 berwarna biru) diletakkan pada dinding maket dengan posisi yang berbeda. Gambar 6 menunjukkan data pembacaan sensor menggunakan kendali terbuka dengan sinyal kendali untuk pemanas yang diatur 80% dari nilai maksimumnya dan sinyal kendali untuk kipas pembuangan udara panas 0% (kondisi off).



Gambar 6 Hasil pembacaan keempat sensor dengan kendali kalang terbuka

Percobaan kendali hidup mati (*on-off*) merupakan percobaan penerapan kendali hidup-mati dalam sistem untuk mengetahui kinerja sistem ketika digunakan kendali hidup-mati. Hasil percobaan penerapan kendali ini ditunjukkan pada Gambar 7.



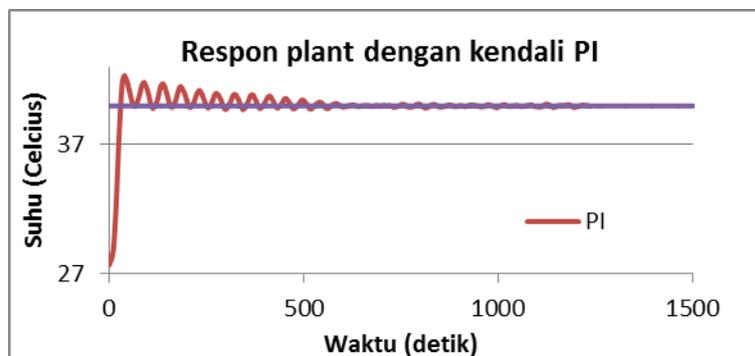
Gambar 7 Grafik penerapan kendali hidup mati

Dari hasil penalaan parameter yang telah dilakukan, maka dipilih hasil penalaan dengan nilai parameter pemanas yaitu  $K_p=2$  dan parameter kipas pembuang udara panas yaitu  $K_p=-5$ . Untuk melakukan penalaan terhadap respon *plant* dengan osilasi kontinu, maka dibutuhkan nilai ultimate gain ( $K_u$ ) dan waktu dari puncak ke puncak ( $P_u$ ) pada respon *plant* dengan kondisi osilasi kontinyu. Didapatkan nilai  $P_u=56$  detik dan nilai  $K_u$  adalah nilai  $K_p$  pemanas yang digunakan yaitu  $K_p=2$ , maka diperoleh hasil penalaan parameter seperti yang ditunjukkan pada Tabel 1.

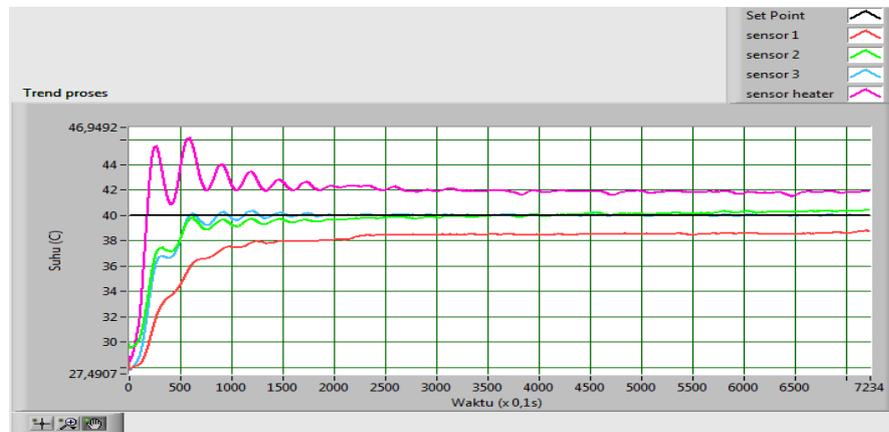
Tabel 1 Parameter hasil penalaan dengan metode osilasi Ziegler-Nichols

Kendali Pemanas	$K_p$	$T_i$ (s)	$T_d$ (s)
P	1	0	0
PI	0,9	46,66	0
PID	1,2	28	7

Setelah memperoleh parameter kendali, maka dilakukan pengujian terhadap kendali PI dengan menggunakan parameter untuk pemanas  $K_p=0,9$  dan  $T_i=46,66$  detik dan parameter kipas pembuang udara panas  $K_p=-5$ . Respon *plant* dengan menggunakan kendali PI untuk pemanas dan kendali P untuk kipas pembuang udara panas ditunjukkan pada Gambar 8. Dari respon tersebut diketahui waktu naik sistem adalah 29 detik dan mengalami puncak pertama hingga  $42,4^\circ\text{C}$  atau 6% dari set point lalu turun ke suhu  $40^\circ\text{C}$  dan kembali naik ke suhu  $41,8^\circ\text{C}$  lalu kembali turun lagi. Secara keseluruhan respon mengalami osilasi teredam yang mencapai kondisi *steady* pada detik ke-606 dengan nilai *error steady state* 1,47%.

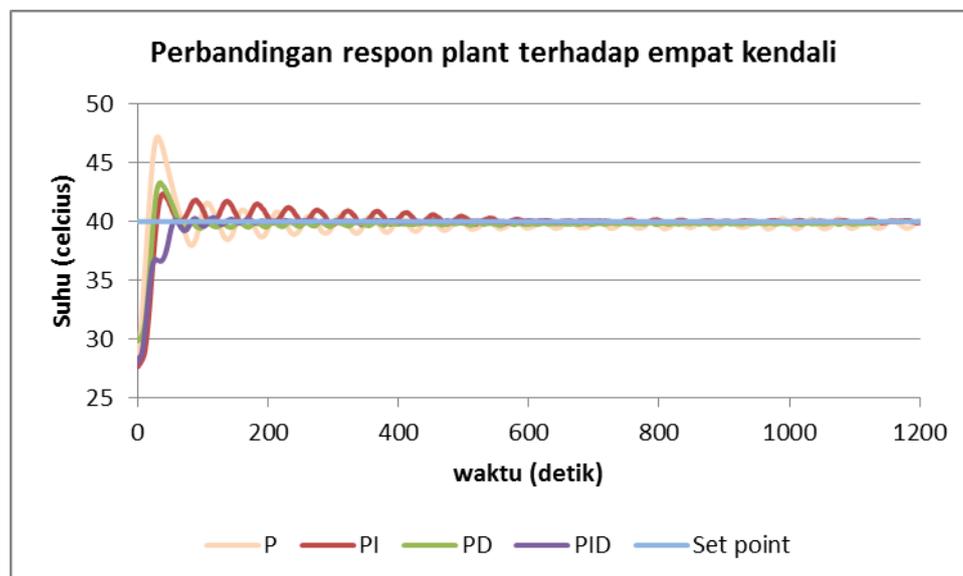
Gambar 8 Respon *plant* menggunakan kendali PI untuk pemanas dan kendali P untuk kipas pembuang udara panas pada *set point*=  $40^\circ\text{C}$

Hasil dari kendali PI masih memiliki *overshoot* sebesar 6% dan waktu untuk mencapai kondisi *steady* cukup lama. Oleh sebab itu dilakukan pengujian terhadap respon *plant* dengan kendali PID. Dari tabel hasil penalaan, parameter kendali PID pemanas yang akan digunakan adalah  $K_p=1,2$  ;  $T_i = 28$  detik dan  $T_d= 7$  detik dan kendali P untuk kendali kipas pembuang udara panas dengan  $K_p= -5$ . Gambar 9 menunjukkan respon pengujian parameter yang telah didapat.



Gambar 9 Respon *plant* menggunakan kendali PID untuk pemanas dan P untuk kipas pembuang udara panas pada setpoint= 40 °C

Hasil dari kendali PI masih memiliki *overshoot* sebesar 6% dan waktu untuk mencapai kondisi *steady* cukup lama. Oleh sebab itu dilakukan pengujian terhadap respon *plant* dengan kendali PID. Dari tabel hasil penalaan, parameter kendali PID pemanas yang akan digunakan adalah  $K_p=1,2$  ;  $T_i = 28$  detik dan  $T_d= 7$  detik dan kendali P untuk kendali kipas pembuang udara panas dengan  $K_p= -5$ . Gambar 10 menunjukkan respon pengujian sistem kendali P, PI, PD dan PID yang telah didapat.



Gambar 10 Grafik perbandingan respon *plant* terhadap empat kendali yang diterapkan pada *plant*

Keempat jenis kendali menghasilkan respon sistem yang berbeda. Waktu naik tercepat diperoleh dengan kendali P, sedangkan yang terlama pada kendali PID, namun *error steady*

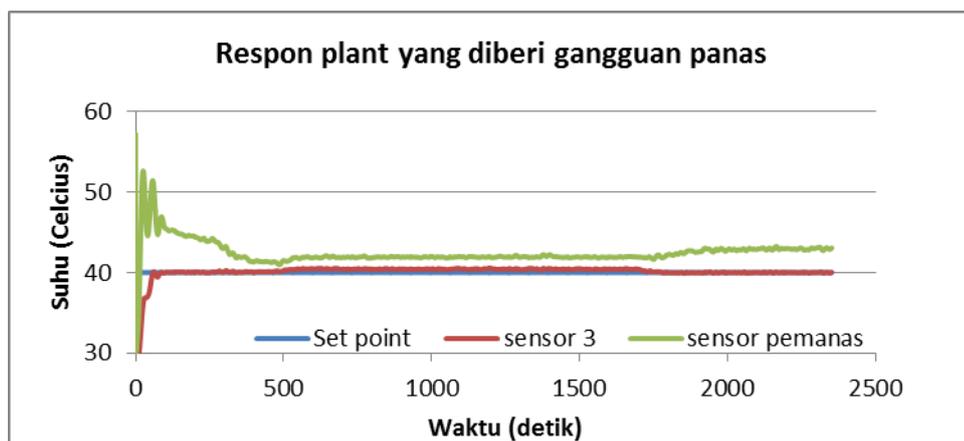
*state* terkecil diperoleh dari kendali PID dan terbesar dari kendali P. Dari perbandingan respon keempat kendali tersebut, maka pengguna dapat menentukan jenis kendali yang akan digunakan sesuai dengan kebutuhan aplikasi. Tabel 2 menunjukkan data perbandingan respon sistem terhadap keempat kendali.

Tabel 2 Perbandingan empat sistem kendali

Kendali	Tr (detik)	Ts (detik)	Overshoot (%)	Error steady state (%)
P	16,60	500	17,8	1,47
PI	29,00	606	6,00	0,50
PD	24,00	400	7,35	0,40
PID	58,00	175	0,15	0,25

Dari tabel 2 ditunjukkan bahwa dengan rise time tercepat dicapai oleh kendali proporsional namun memperoleh *overshoot* 17,8% dan *error steady state* sebesar 1,47%. Dengan *error steady state* yang diinginkan berada dibawah 0,5% dan memiliki waktu yang cepat untuk mencapai kondisi stabilnya, maka jenis kendali yang memenuhi syarat tersebut adalah kendali PID. Pemilihan jenis kendali ini dapat disesuaikan dengan kebutuhan akan hasil *plant* yang diinginkan sebelumnya.

Untuk membuktikan kehandalan sistem kendali yang dibuat, maka dilakukan pengujian terhadap beberapa gangguan yang tidak bisa dikendalikan berupa gangguan panas dari sumber lain yaitu solder. Gambar 11 menunjukkan gambar respon dari *plant* dengan gangguan panas. Gangguan panas diaktifkan pada menit ke-4 dan dinon-aktifkan pada menit ke-28. Dengan set point 40 °C dari gambar tersebut terlihat bahwa ketika gangguan panas diaktifkan, maka terjadi peningkatan panas secara perlahan hingga mencapai 40,6°C dan sistem kendali dapat mempertahankan suhu *plant* pada suhu 40,5°C yang artinya kesalahan kondisi steady sebesar 0,5°C atau 1,25% dari setpoint yang ditentukan. Karena nilai kesalahan kondisi *steady* masih di bawah 2%, performa sistem kendali dapat dikatakan baik dalam mengatasi gangguan panas yang diberikan.



Gambar 11 Grafik respon *plant* yang diberi gangguan panas dari sumber lain dengan set point 40 °C

#### 4. KESIMPULAN

1. Telah berhasil dibuat purwarupa sistem kendali PID berbasis LabVIEW dengan menggunakan akuisisi data *board* Arduino pada studi kasus kendali PID suhu ruang.
2. Metode kedua Ziegler-Nichols dapat digunakan untuk penalaan parameter PID pada sistem yang memiliki respon osilasi kontinu.
3. Hasil penalaan parameter kendali sesuai metode osilasi kontinu Ziegler-Nichols adalah ketika nilai  $K_p$  untuk kendali pemanas = 2 dan  $K_p$  untuk kendali kipas ventilasi = -5.
4. Hasil penalaan parameter kendali sesuai rumus empiris yang disarankan metode Ziegler-Nichols untuk kendali PID pemanas adalah  $K_p=1,2$  ,  $T_i= 28$  detik dan  $T_d = 7$  detik dengan perolehan *rise time* 58 detik, *settling time* 175 detik dan *steady state error* 0,25%.
5. Hasil penalaan menggunakan metode Ziegler-Nichols untuk kendali PI memiliki *steady state error* sebesar 0,5% sedangkan dengan penalaan hasil perhitungan analitis sebesar 2,5%.

#### DAFTAR PUSTAKA

- [1] Ogata, Katsuhiko, 1997, *Teknik Kontrol Automatik*, Jilid 1 dan 2, Edisi Kedua, Erlangga: Jakarta.
- [2] National Instrument, 2009, *Introduction to LabVIEW 8.6 in 3 Hours*, <http://www.ni.com/white-paper/5247/en> diakses tanggal 11 Juni 2012.
- [3] Semiconductor, N., 2000, LM35 Precision Centigrade Temperature Sensor, America: National Semiconductor Corporation.
- [4] Arduino, 2009, Arduino Duemilanove, <http://arduino.cc/en/Main/arduinoBoardDuemilanove> diakses tanggal 10 September 2012.
- [5] Artanto, D., 2012, *Interaksi Arduino dan LabVIEW*, PT Elex Media Komputindo: Jakarta.