

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian yang dilakukan oleh Rachmad Dwi Raharjo, Joko Susila, dan Imam Arifin yaitu membuat desain dan implementasi pengendali PID *gain scheduling*, dimana pengendali akan mengatur nilai dari parameter PID mengalami perubahan yang mengakibatkan hasil *output* dari sistem akan sesuai dengan nilai *setpoint* yang dikehendaki [4].

Penelitian yang dilakukan oleh Bambang Supriyo, Samuel Beta Kuntardjo, dan Sihono yaitu membuat dan menguji kinerja alat peraga pengendali pemanas udara menggunakan arduino uno dengan metode kendali berbasis *Proportional Integral Derivative* (PID) untuk rentang temperatur antara 50°C hingga 70°C. Arduino uno yang telah terprogram untuk mengendalikan dan mengirim data pengendalian secara serial ke komputer melalui *port* USB. Pemilihan parameter awal PID menggunakan gabungan antara metode *Relay Feedback* dan persamaan *Ziegler-Nichols*. Pengendalian yang dihasilkan pada metode: *proportional* (P), *proportional derivative* (PD) dan *Proportional Integral Derivative* (PID) dengan kriteria kinerja pengendalian berdasarkan pada *overshoot* dan *steady state error*. Hasil pengujian memperlihatkan bahwa kendali PID memberikan respon keluaran paling baik dengan *steady state error* mendekati 0°C, sedangkan kontrol P dan PD masih menghasilkan *steady state error* sekitar 3°C. Pada kendali P juga masih memberikan keluaran fluktuatif sekitar 2,5°C-4°C di bawah nilai *set point*, sedangkan kendali PD dan PID mempunyai keluaran fluktuatif yang jauh lebih kecil bahkan mendekati nol [5].

Pada penelitian yang dilakukan Nasrul Z.A, Yonita Putri Roja, dan Novi Sylvia yaitu mengenai sistem pengendali PID yang dilakukan pada reaktor pabrik asam *forminat* yang berkapasitas 100.000 ton/tahun [6]. Dengan adanya penelitian ini, didapatkan respon terbaik dengan menggunakan sistem pengendali PID. Penelitian ini bertujuan untuk mendapatkan respon terbaik kontrol PID dengan menggunakan metode *trial and error* pada nilai parameter PID. Diperoleh nilai

Kc sebesar 8, Ti sebesar 1, dan Td sebesar 0, dengan respon waktu rata-rata yang didapatkan sebesar 0,815 menit.

Dari beberapa penelitian diatas dapat disimpulkan bahwa penelitian dengan menggunakan metode PID dapat menghasilkan respon sistem yang cukup optimal. Maka dari itu, pada penelitian ini digunakan metode PID sebagai sistem pengendalinya. Agar memperoleh hasil respon sistem yang optimal di setiap nilai *setpoint* yang digunakan maka akan digunakan metode *gain scheduling* PID.

2.2 Tinjauan Komponen Penelitian

Pada penelitian ini terdapat tinjauan komponen penelitian yang merupakan penjelasan mengenai beberapa komponen yang digunakan pada penelitian ini. Berikut ini merupakan beberapa komponen yang digunakan :

2.2.1 Reaktor

Reaktor merupakan alat yang mempunyai peran sebagai tempat terjadinya suatu reaksi, salah satunya adalah reaksi kimia. Reaksi dapat terjadi karena adanya perubahan bentuk atau sifat dari suatu bahan, perubahan ini dapat terjadi secara alami ataupun dengan adanya bantuan energi panas. Secara umum reaktor dibagi menjadi dua jenis yaitu reaktor nuklir dan reaktor kimia. Reaktor nuklir adalah suatu alat untuk mengendalikan reaksi fisi berantai dan sekaligus menjaga kesinambungan reaksi fisi tersebut. Reaktor kimia merupakan tempat berlangsungnya suatu reaksi kimia, seringkali dinyatakan sebagai pusat suatu proses kimia [7] untuk mengubah bahan baku menjadi produk. Pada pembuatan reaktor kimia perlu memperhatikan efisiensi dari keluaran produk yang diinginkan. Reaktor kimia yang digunakan pada penelitian ini berbentuk silinder dengan bagian bawah datar serta dilengkapi dengan penutup.

2.2.2 Arduino Mega 2560

Arduino Mega 2560 merupakan *board* mikrokontroler berdasarkan pada ATmega2560. *Board* ini memiliki 54 *pin input/output* digital, dimana 15 *pin* berfungsi sebagai *output Pulse Width Modulation* (PWM), sebanyak 16 *pin analog input*, serta 4 *pin* serial port perangkat keras(UART). *Board* Arduino Mega 2560 juga memiliki *oscillator* 16 Mhz, *port* USB, *power jack* DC, ICSP

header serta tombol untuk *reset* [8]. Tabel 2.1 menunjukkan *pin-pin* yang digunakan pada arduino.

Tabel 2.1 Penggunaan *Pin* Arduino Mega 2560

Komponen	<i>Pin</i> Arduino Mega2560
<i>Solid State Relay</i> (SSR)	<i>Pin</i> 13
	<i>Pin</i> GND
Modul MAX6675	<i>Pin</i> 10
	<i>Pin</i> 11
	<i>Pin</i> 12
	<i>Pin</i> 5V
	<i>Pin</i> GND

2.2.3 *Band Heater*

Band heater merupakan elemen pemanas berbentuk tabung yang berfungsi untuk memanaskan reaktor yang menyerupai bentuk dari reaktor yang akan dipanaskan [9]. Pada penelitian ini digunakan *band heater* dengan dimensi 70×50 mm, kemudian tegangan yang dibutuhkan sebesar 220V, dan daya yang digunakan sebesar 800W. Elemen pemanas ini dibuat dengan menggunakan material *ceramic* sebagai material penghantar panasnya. Gambar 2.1 menunjukkan elemen pemanas yang akan digunakan pada penelitian ini.



Gambar 2.1 *Band heater*

2.2.4 *Sensor Termokopel Tipe-K*

Sensor suhu yang digunakan pada penelitian ini berjenis termokopel tipe-K karena Termokopel tipe-K dapat digunakan untuk mendeteksi atau mengukur suhu

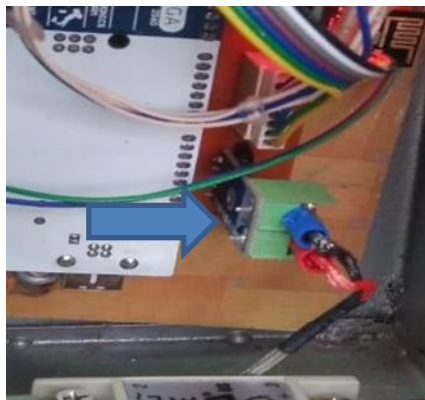
dengan rentang capaian pembacaan suhu sebesar -200°C hingga 1024°C . Sensor ini bekerja untuk mengukur suhu dengan cara menyatukan dua jenis logam konduktor yang berbeda bahan atau materialnya sehingga terjadi efek termoelektrik [10]. Pembacaan sensor termokopel berada pada rentang -200°C hingga 1024°C . Gambar 2.2 menunjukkan sensor suhu termokopel tipe-K.



Gambar 2.2 Sensor suhu termokopel tipe-K

2.2.5 Modul MAX6675

MAX6675 adalah rangkaian yang digunakan sebagai pengganti *cold-junction* pada *Thermocouple* tipe-K. Besar resolusi *output* dengan data yang dihasilkan adalah 12 bit yang mampu bekerja dengan SPI™. Modul ini hanya menggunakan *Format Read Only*. Kelebihan dari modul MAX6675 ini dapat melakukan pembacaan suhu dengan rentang $0,25^{\circ}\text{C}$ - 1024°C . Gambar 2.3 memperlihatkan modul MAX6675.



Gambar 2.3 Modul MAX6675

2.2.6 Relay SSR

Pada sistem CoMoTh MCR membutuhkan suatu perangkat yang dapat menghidupkan dan mematikan elemen *heater* secara otomatis melalui masukan dari mikrokontroler. Saklar yang dibutuhkan adalah saklar bertipe *SSR* dengan semikonduktor yang menggunakan SCR, TRIAC, atau *output* transistor untuk menggantikan saklar kontak mekanik.



Gambar 2.4 *Solid state relay (SSR)*

. *Solid State Relay (SSR)* seperti Gambar 2.4. Untuk spesifikasi khusus komponen relay *SSR* yang dibutuhkan pada *heater* dapat diperlihatkan pada Tabel 2.2.

Tabel 2.2 Spesifikasi *Solid State Relay (SSR)*

Parameter	<i>Solid State Relay (SSR)</i>
Fungsi	Saklar <i>on / off</i> pada elemen pemanas
Tegangan Masuk	3V DC – 32V DC
Tegangan Keluar	24V – 380V AC
Arus	10 <i>Ampere</i>
Deskripsi Kebutuhan	Dapat memutus maupun menghubungkan aliran listrik secara otomatis melalui perintah mikrokontroler.

2.3 Tinjauan Pengolahan Data

Pada penelitian ini terdapat tinjauan pengolahan data yang berisi mengenai penjelasan metode apa yang digunakan pada proses implementasi produk CoMoTh MCR. Berikut ini merupakan metode pengolahan data yang digunakan:

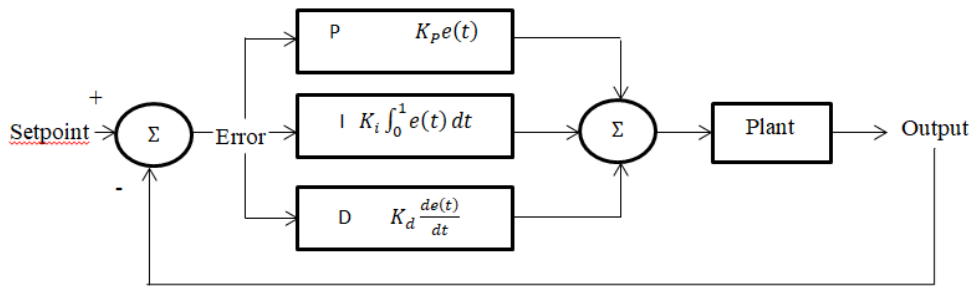
2.3.1 Sistem Kendali PID

Sistem Kontrol PID adalah sebuah metode pengendali untuk menentukan akurasi sistem umpan balik dengan kombinasi dari 3 pengontrol diantaranya *Proportional*, *Integral*, serta *Derivative*. Pengontrol ini akan dirangkai secara paralel dimana setiap pengontrol mempunyai kelebihan dan kekurangannya masing-masing. Pada pengimplementasiannya ketiga pengontrol tersebut dapat bekerja sendiri maupun gabungan dari ketiga parameter tersebut tersebut. Dalam perancangan sistem kontrol PID yang perlu dilakukan adalah mengatur parameter P, I, serta D agar respon dari sinyal keluaran terhadap masukan akan sesuai dengan apa yang diinginkan. Tabel 2.3 menunjukkan karakteristik PID dari masing-masing parameter nilai *gain* PID.

Gambar 2.6 memperlihatkan kontrol dari PID dan pada Persamaan 2.1 menunjukkan hubungan antara nilai PID serta identifikasi keluarannya sebagai $u(t)$. Dari Persamaan 2.1 dapat dirumuskan menjadi bentuk digital diskrit yang dapat dilihat pada Persamaan 2.2. Ketika diterapkan dalam bahasa pemrograman menjadi Persamaan 2.3 .

Tabel 2.3 Karakteristik PID

No	Tanggapan Loop Tertutup	Waktu Naik	<i>Overshoot</i>	Waktu Turun	Kesalahan Keadaan Tunak
1	Proporsional (Kp)	Menurun	Meningkat	Perubahan Kecil	Menurun
2	Integral (Ki)	Menurun	Meningkat	Meningkat	Hilang
3	Derivatif (Kd)	Perubahan Kecil	Menurun	Menurun	Perubahan Kecil



Gambar 2.5 Diagram blok sistem kendali PID

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (2.1)$$

Dimana:

$u(t)$ = Sinyal *output* pengendali PID

$e(t)$ = Sinyal kesalahan (*error*)

K_p = Konstanta *proportional*

K_i = Konstanta *integral*

K_d = Konstanta *derivative*

$$u(k) = K_p e_k + K_i T \sum_0^k e_k + \frac{1}{T} K_D (e_k - e_{k-1}) \quad (2.2)$$

$$u(t) = K_p \times K_i \times (error + last_error) \times T_s + \frac{K_d}{T_s} \times (error - last_error) \quad (2.3)$$

Dimana T_s merupakan *time sampling*, error merupakan kesalahan, serta last error merupakan nilai kesalahan yang terbaca sebelumnya. Selisih atau deviasi antara parameter proses (PV) dengan nilai yang ingin dicapai atau *setpoint* (SP) disebut dengan error sehingga dapat dirumuskan pada Persamaan 2.4 [12]. Nilai kesalahan atau *error* dari pembacaan sensor suhu akan dijadikan nilai acuan perhitungan dari pengendali PID.

$$error = SP - VP \quad (2.4)$$

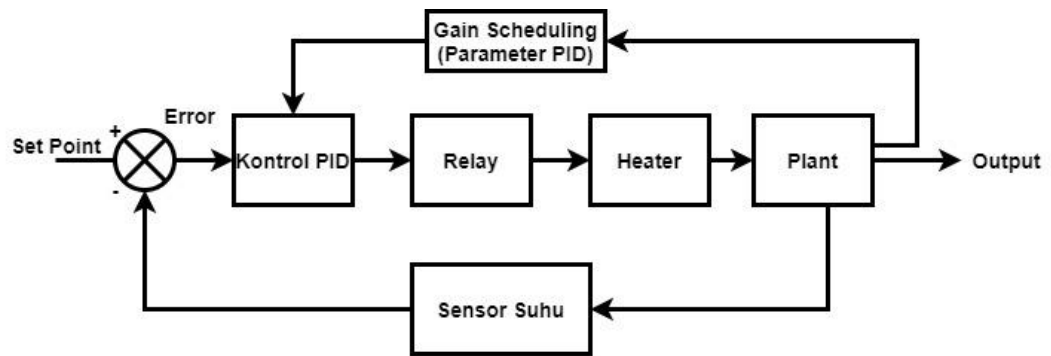
Pada perancangan sistem kontrol PID yang perlu dilakukan adalah mengatur parameter P, I, serta D agar tanggapan sinyal keluaran sistem terhadap masukan tertentu sesuai dengan yang diinginkan. Ada banyak cara yang dilakukan untuk mendapatkan nilai PID, kebanyakan dilakukan dengan metode *trial and error*. Namun, pada sistem pengendalian kali ini akan dilakukan dengan menggunakan metode *gain scheduling* PID atau penalaan otomatis untuk mendapatkan mengubah nilai parameter PID sesuai dengan *setpoint* yang diinginkan.

$$SSR_{ON_TIME} = PID_{interval} \times PID_{output} \quad (2.8)$$

Untuk mengontrol suhu pada reaktor, komponen *relay* SSR harus diaktifkan sehingga dapat menghubungkan tegangan AC ke *heater*. Waktu *on/off* SSR digunakan untuk mengontrol suhu kemudian akan dikontrol dengan menggunakan metode PID. Persamaan 2.8 merupakan pengendalian SSR dengan menggunakan metode PID [13].

2.3.2 Gain Scheduling PID

Seperti yang sudah dijelaskan pada poin sebelumnya bahwa pengendali PID banyak digunakan karena mudah dalam perancangan serta pengimplementasiannya. Namun, pada kondisi tertentu yang mana nilai dari *setpoint* dapat berubah-ubah, dengan ini parameter dari kontrol PID harus dapat diubah secara berkala agar menghasilkan kinerja yang baik. Ada banyak teknik yang dapat digunakan untuk dapat *men-tuning* nilai parameter kendali PID salah satunya adalah metode *proportional integral derivative gain scheduling*. *Gain scheduling* adalah salah satu metode kendali yang mana nilai parameter *proportional*, *integral*, dan *derivative* yang digunakan oleh kendali akan dijadwalkan atau diubah secara otomatis berdasarkan nilai beban atau suhu yang terbaca oleh sensor [13].



Gambar 2.6 Diagram blok sistem kendali *gain scheduling*

Gambar 2.7 memperlihatkan diagram blok sistem kendali dengan menggunakan metode *gain scheduling*. Langkah awal yang harus dilakukan untuk menggunakan metode ini adalah menentukan nilai variabel PID untuk setiap kondisi atau suhu yang digunakan [14].