

Desain dan Simulasi *Chiller Sequencing Control* dengan Metode *Chilled Water Return Temperature (T-Based) Control*

Alexander Irvan Kaware¹, Handry Khoswanto², Roche Alimin³

Program Studi Teknik Elektro, Universitas Kristen Petra

Jl.Siwalankerto 121-131, Surabaya 60236, Indonesia

Email: irvan.alexander28@gmail.com¹, handry@petra.ac.id², ralimin@petra.ac.id³

Abstrak — Pada proyek ini akan dibahas pembuatan program *chiller sequencing control* dengan metode *chilled water return temperature (T-Based) control* beserta HMI-nya dengan menggunakan kontroler OPEN EMS 600 I32 pada PT. XYZ yang memiliki sebuah *multiple chiller plant* yang terdiri dari 4 *chiller plant*. Selain itu, akan dibahas juga pembuatan simulator *chiller plant* dengan Arduino Mega 2560 untuk mewakili perilaku *chiller plant* yang nantinya dapat berkomunikasi dengan kontroler OPEN EMS 600 I32 melalui komunikasi Modbus RTU.

Berdasarkan hasil pengujian, program *chiller sequencing control* berhasil mengontrol 4 buah *chiller plant* dengan metode *chilled water return temperature (T-Based) control*, dimana urutan sequencing tiap komponen-komponen utamanya (*chiller, chilled water pump, condenser water pump, dan cooling tower*) berdasarkan *running time* per 8 jam dari tiap-tiap komponen tersebut, HMI mampu menampilkan keadaan *chiller plant* baik dalam keadaan normal atau eror, dan Arduino Mega 2560 mampu mengirimkan data dengan tepat ke kontroler OPEN EMS 600 I32 melalui komunikasi Modbus RTU.

Kata Kunci — *multiple chiller plant, chiller sequencing control, DEOS OPEN EMS 600 I32, Arduino Mega 2560, HMI, Modbus*

I. PENDAHULUAN

Pengondisian udara (air conditioning) merupakan salah satu kebutuhan dasar dari sebuah bangunan gedung untuk memberikan kenyamanan termal dan menyediakan lingkungan dalam gedung yang sehat [1]. Pada gedung-gedung besar, pengondisian udara seringkali dilakukan dengan menggunakan *centralized air conditioning system*, karena dapat menghemat biaya instalasi dan memiliki COP (*Coefficient of Performance*) yang tinggi, antara 3-7 [2]. Namun, permasalahan utama dari penggunaan *centralized air conditioning system*, adalah pengoperasiannya membutuhkan konsumsi energi yang tinggi. Di Indonesia, sekitar 40%-70% penggunaan listrik di gedung-gedung besar dialokasikan untuk keperluan *air conditioning* yang didominasi oleh *multiple-chiller-plant* [3]

Hal ini memunculkan sebuah kebutuhan untuk menjamin bahwa konsumsi energi oleh *multiple-chiller-plant* efisien. Oleh karena itu, *chiller sequencing control* dipilih untuk menjawab kebutuhan tersebut. *Chiller sequencing control* berfungsi untuk menentukan kapan dan *chiller plant* mana yang akan menyala atau mati berdasarkan kondisi suatu parameter.

Dengan adanya *chiller sequencing control*, sistem pendinginan dapat bekerja dengan COP yang tinggi, sehingga dapat mendinginkan panas dari sebuah gedung dengan efisien, serta mencegah *chiller* untuk bekerja diatas kapasitas pendinginannya [4].

Melihat besarnya manfaat dari penerapan *chiller sequencing control* pada *multiple-chiller-plant*, maka muncul sebuah kebutuhan akan *chiller sequencing control* pada sebuah bangunan yang memiliki *multiple-chiller-plant*. Maka dari itu, proyek ini akan membahas mengenai pembuatan desain dan simulasi *chiller sequencing control* pada *multiple-chiller-plant* pada PT.XYZ dengan menggunakan metode *chilled water return temperature-based (T-based) control* beserta HMI-nya.

II. PERENCANAAN DAN IMPLEMENTASI

Secara garis besar, sistem *chiller sequencing control* yang akan dibuat terdiri dari :

- Kontroler OPEN EMS 600 I32 yang berfungsi untuk menerima input dari potensiometer 10 KOhm dan mikrokontroler Arduino yang kemudian akan diproses oleh program *chiller sequencing* yang outputnya kemudian akan dikirim ke HMI.
- Potensiometer 10 KOhm yang berfungsi sebagai input manual temperatur air pada header pipa *chilled water return (CHWR)* dengan nilai minimum 9°C, dan maksimum 12.5°C.
- Arduino Mega 2560 sebagai simulator *chiller plant* yang akan berkomunikasi dengan kontroler OPEN EMS sebagai *slave* dengan menggunakan komunikasi serial RS 485 dengan protokol Modbus RTU.
- HMI (*Human Machine Interface*) yang berfungsi untuk visualisasi *chiller plant* PT. XYZ dan pemberian nilai *input* parameter.

Untuk pendesainan program *chiller sequencing control* beserta HMInya digunakan aplikasi DEOS FUP XL. Selanjutnya, agar HMI program *chiller sequencing control* dapat diakses melalui internet, digunakan aplikasi OPENView.

Untuk *chiller sequencing control* digunakan parameter *chilled water return temperature based (T-Based) control* yang menggunakan temperatur air pada header pipa *chilled water return (CHWR)* sebagai parameter untuk menyala atau

mematikan sebuah *chiller*. *Chiller sequencing control* hanya berfungsi untuk mengontrol *sequencing chiller* dan komponen-komponen utamanya. Cara kerja parameter *chilled water return temperature based (T-Based) control* adalah sebagai berikut [5]:

- *Chiller* n+1 akan menyala apabila jumlah *chiller* yang sedang menyala adalah n dan temperatur CHWR (T_{rtn}) nilainya lebih besar dibandingkan dengan switch on threshold dari n *chiller* ke n+1 *chiller* selama periode yang telah ditentukan.
- *Chiller* n+1 akan mati apabila jumlah *chiller* yang sedang menyala adalah n+1 dan temperatur CHWR (T_{rtn}) nilainya lebih kecil dibandingkan dengan *switch off threshold* dari n+1 *chiller* ke n *chiller* selama periode yang telah ditentukan.

A. Menentukan Nilai Switch On dan Switch Off Threshold T_{rtn}

Untuk mendapatkan nilai *switch on* dan *switch off threshold* T_{rtn} (temperatur air pada *return pipe*) untuk *T-based chiller sequencing*, pertama-tama perlu ditentukan parameter temperatur *chilled water supply*, kapasitas maksimum pendinginan *chiller*, *chilled water flow*, persentase *full load* tiap *chiller*, dan *deadband* yang didapatkan dari datasheet *chiller* Daikin CUWD230CS5Y. Untuk tabel parameternya dapat dilihat pada Tabel 1.

Tabel 1 Tabel Parameter

PARAMETER (Satuan)	Nominal
Jumlah Chiller (Unit)	4
Kapasitas Maksimum Pendinginan(Qr) per Chiller (kW)	755
Temperatur Chilled Water Supply (°C)	7°C
Chilled Water Flow (L/s)	36.06
n (Persentase Full Load Chiller) (%)	95%
d (Deadband) (%)	4%
c_w (Chilled water specific heat capacity)(kJ/kg°C)	4,2

Setelah menentukan parameter-parameternya, kita perlu mencari nilai *switch on* dan *switch off threshold* Q (beban pendinginan instan) terlebih dahulu dengan menggunakan persamaan berikut, yang hasilnya dapat dilihat pada Tabel 2:

$$Q_z^{on} = n \times Qr (z + d) \tag{1}$$

$$Q_z^{off} = n \times Qr (z - d) \tag{2}$$

Dimana:

$$Q_z^{on} = Q \text{ switch on threshold}$$

$$Q_z^{off} = Q \text{ switch off threshold}$$

$$n = \text{persentase kapasitas full load chiller (\%)}$$

$$Qr = \text{kapasitas pendinginan ideal chiller (kW)}$$

$$z = \text{jumlah chiller}$$

$$d = \text{deadband (\%)}$$

Tabel 2 Tabel Nilai Switch On dan Switch Off Threshold Q

Switch On Thresholds	Switch Off Threshold
From 1 chiller to 2 chillers: $Q > 745.94 \text{ kW}$	From 2 chillers to 1 chiller: $Q < 688.56 \text{ kW}$
From 2 chillers to 3 chillers: $Q > 1463.19 \text{ kW}$	From 3 chillers to 2 chillers: $Q < 1405.81 \text{ kW}$
From 3 chillers to 4 chillers: $Q > 2180.44 \text{ kW}$	From 4 chillers to 3 chillers: $Q < 2123.06 \text{ kW}$

Keterangan: Q => Beban pendinginan instan

Setelah didapatkan nilai Q *switch on* dan *switch off threshold*, didapatkan nilai *switch on* dan *switch off threshold* T_{rtn} (temperatur air pada *chilled water return pipe*) dengan menggunakan persamaan berikut, yang hasilnya dapat dilihat pada Tabel 3:

$$T_{rtn} = T_{sup} + \frac{Q}{c_w m_p} \tag{3}$$

Dimana:

$$T_{rtn} = \text{temperatur air pada header pipa CHWR}$$

$$T_{sup} = \text{temperatur air pada header pipa CHWS}$$

$$Q = \text{beban pendinginan instan}$$

$$c_w = \text{chilled water specific heat capacity}$$

$$m_p = \text{chilled water flow}$$

Tabel 3 Tabel Nilai Switch On dan Switch Off Threshold T_{rtn}

Switch On Thresholds	Switch Off Threshold
From 1 chiller to 2 chillers: $T_{rtn} > 11.93 \text{ }^\circ\text{C}$	From 2 chillers to 1 chiller: $T_{rtn} < 9.27 \text{ }^\circ\text{C}$
From 2 chillers to 3 chillers: $T_{rtn} > 11.83 \text{ }^\circ\text{C}$	From 3 chillers to 2 chillers: $T_{rtn} < 10.10 \text{ }^\circ\text{C}$
From 3 chillers to 4 chillers: $T_{rtn} > 11.79 \text{ }^\circ\text{C}$	From 4 chillers to 3 chillers: $T_{rtn} < 10.50 \text{ }^\circ\text{C}$

Keterangan: T_{rtn} = Suhu air pada header pipa Chilled Water Return (CHWR)

B. Desain Pemipaan Multiple Chiller Plant

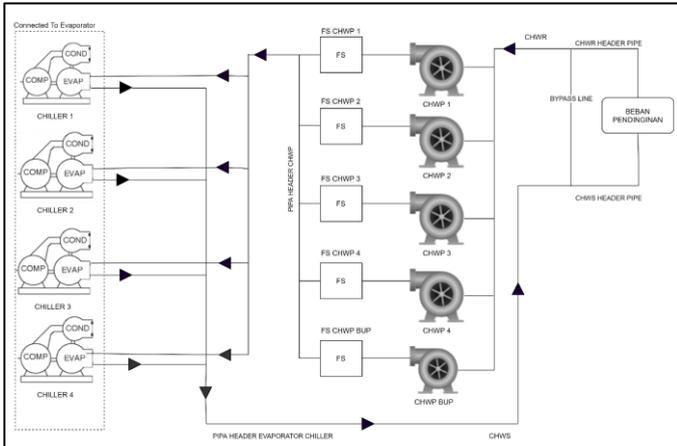
Multiple chiller plant terdiri dari 4 buah *water-cooled-chiller* dengan merek Daikin CUWD230CS5Y, 5 pompa evaporator (CHWP) beserta *flow switch*nya, dimana 4 merupakan pompa utama, dan 1 merupakan pompa *backup* yang baru akan menyala apabila salah satu pompa utama rusak, 5 pompa kondensor (CWP) beserta *flow switch*nya, dimana 4 merupakan pompa utama, dan 1 merupakan pompa *backup* yang baru akan menyala apabila salah satu pompa utama rusak, dan 4 buah *cooling tower* beserta *flow switch*nya.

Gambar 1 menunjukkan desain pemipaan CHWP terhadap evaporator-evaporator *chiller*. Untuk desain pemipaan pompa CHWP ke *chiller-chiller* digunakan metode manifold, dimana pompa-pompa CHWP menerima air dari header pipa CHWR (*chilled water return*), yang kemudian output dari pompa-pompa tersebut dijadikan satu di sebuah header, baru kemudian akan diteruskan ke evaporator-evaporator *chiller*, lalu output dari evaporator-evaporator *chiller* berupa CHWS (*chilled water supply*) akan dijadikan satu juga di sebuah header, lalu baru diteruskan ke AHU-AHU pada gedung.

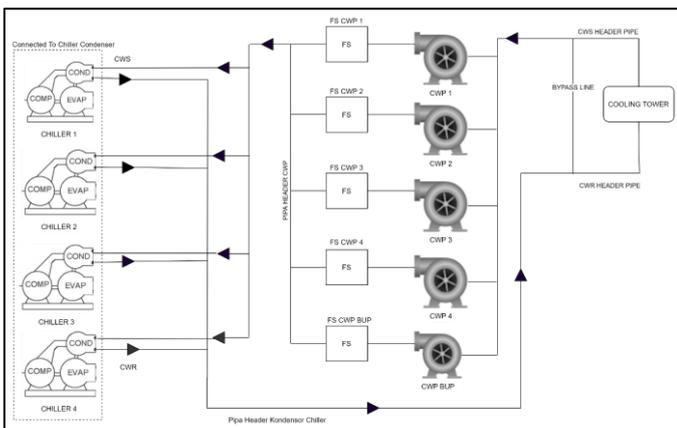
Gambar 2 menunjukkan desain pemipaan CWP terhadap kondensor-kondensor *chiller*. Untuk desain pemipaan pompa CWP yang menghubungkan *cooling tower* ke kondensor *chiller-chiller* digunakan metode manifold juga, dimana pompa-pompa CWP akan mengalirkan air CWS (*cooling water supply*) dari *cooling tower* menuju kondensor tiap *chiller*, output dari pompa-pompa tersebut dijadikan satu di sebuah header, baru kemudian akan diteruskan ke kondensor tiap *chiller*, lalu output CWR (*cooling water return*) dari kondensor

dari tiap *chiller* akan dijadikan satu juga di sebuah *header*, lalu baru diteruskan *cooling tower* untuk didinginkan kembali.

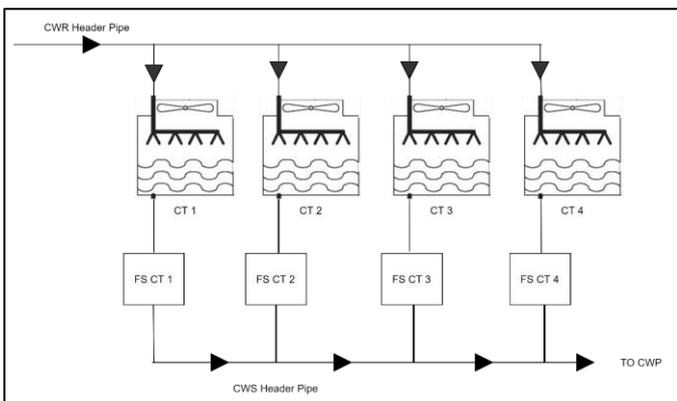
Gambar 3 menunjukkan desain pemipaan *cooling tower*. Untuk desain pemipaan *cooling tower* didesain dengan metode manifold, dimana *cooling tower* akan menerima air dari *header* pipa CWR. Lalu output dari semua *cooling tower* akan dijadikan satu di *header* pipa CWS, yang kemudian akan dipompa oleh pompa CWP menuju input kondensator *chiller*.



Gambar 1 Desain pemipaan CHWP



Gambar 2 Desain pemipaan CWP



Gambar 3 Desain pemipaan *cooling tower*

C. Desain Program Chiller Sequencing Control

Secara garis besar, program *chiller sequencing control* terdiri dari 6 buah FUP *page* yang berjalan secara paralel, dimana tiap subprogram memiliki fungsi-fungsi untuk dijalankan. Untuk penjelasan fungsi dari tiap FUP *page* adalah sebagai berikut :

- `mainprog.f`
Berfungsi untuk membaca temperatur air pada *header* pipa CHWR, jumlah *chiller plant* yang sedang beroperasi, dan jumlah *chiller plant* maksimum yang dapat beroperasi, lalu menentukan berapa jumlah *chiller plant* yang diperlukan.
- `chwpsq.f`
Berfungsi untuk mengurutkan CHWP berdasarkan waktu operasional per 8 jam dari yang paling rendah, menyalakan atau mematikan sebuah CHWP, dan menentukan jumlah CHWP maksimum yang dapat beroperasi.
- `cwpsq.f`
Berfungsi untuk mengurutkan CWP berdasarkan waktu operasional per 8 jam dari yang paling rendah, menyalakan atau mematikan sebuah CWP, dan menentukan jumlah CWP maksimum yang dapat beroperasi.
- `ctseq.f`
Berfungsi untuk mengurutkan CT berdasarkan waktu operasional per 8 jam dari yang paling rendah, menyalakan atau mematikan sebuah CT, dan menentukan jumlah CT maksimum yang dapat beroperasi.
- `chseq.f`
Berfungsi untuk mengurutkan *chiller* berdasarkan waktu operasional per 8 jam dari yang paling rendah, menyalakan atau mematikan sebuah *chiller*, dan menentukan jumlah *chiller* maksimum yang dapat beroperasi.

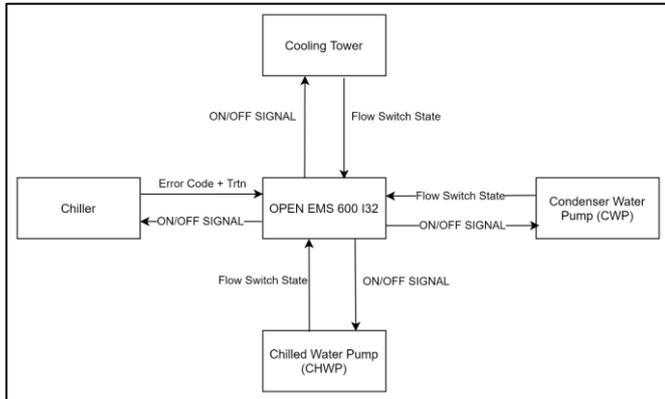
D. Desain Simulator Chiller Plant

Simulator *chiller plant* berfungsi untuk mensimulasikan sebuah *multiple chiller plant*. Secara garis besar, sistem simulator dibuat dengan menggunakan kontroler DEOS OPEN EMS 600 I32 yang dimana kontroler tersebut berfungsi untuk mengirimkan perintah untuk menjalankan atau mematikan sebuah komponen pada *chiller plant* berdasarkan input yang diterima oleh kontroler, yakni nilai T_{rn} dan kondisi dari tiap komponen-komponen utama pada *chiller plant*. Untuk diagram sistemnya dapat dilihat pada Gambar 4.

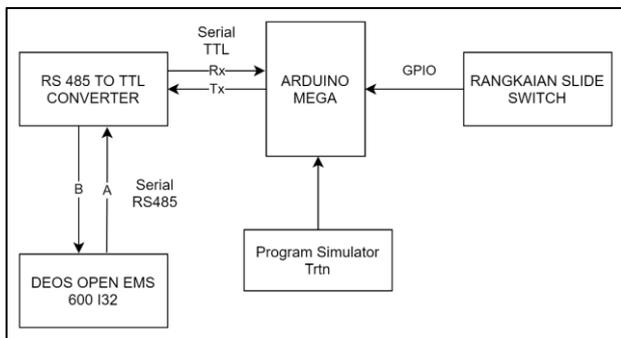
Untuk CHWP, CWP, dan CT, *flow switch state* disimulasikan dengan cara memberikan input di HMI dari program *chiller sequencing control*, sedangkan untuk *chiller*, kode eror dan nilai temperatur air pada *header* pipa *chilled water return* akan disimulasikan dengan menggunakan sebuah Arduino Mega yang kemudian data-data tersebut masing-masing akan disimpan di dalam sebuah *register* yang kemudian *register-register* tersebut akan dikirim ke kontroler OPEN EMS 600 I32 sebagai *input* untuk program *chiller sequencing control* melalui protokol Modbus dengan metode RS-485.

Untuk nilai temperatur air pada *header* pipa *chilled water return* selain dapat disimulasikan melalui Arduino Mega, dapat disimulasikan juga dengan melalui potensiometer 10 KOhm.

Hardware yang digunakan dalam pembuatan simulator *chiller plant* ini adalah RS485 to TTL converter, *slide switch*, dan mikrokontroler Arduino Mega 2560 yang koneksinya dapat dilihat pada Gambar 5.



Gambar 4 Diagram sistem simulator *chiller plant*



Gambar 5 Diagram koneksi *hardware* simulator *chiller plant*

Slide switch berfungsi untuk mensimulasikan kode-kode eror yang mungkin terjadi dalam sebuah *chiller*, dalam proyek ini, kode-kode eror yang akan disimulasikan mengacu pada kode alarm yang terdapat pada MicroTech III milik Daikin, yang fungsinya sebagai kontroler protokol informasi *chiller* Daikin. Berikut adalah kode-kode alarm yang akan disimulasikan:

1. Tekanan tinggi terdeteksi pada kondensor (kode alarm : 37159), yang akan disimulasikan dengan nilai 101.
2. Tekanan rendah terdeteksi pada evaporator (kode alarm : 39207), yang akan disimulasikan dengan nilai 102.
3. Temperatur kompresor terlalu tinggi (kode alarm : 34599), yang akan disimulasikan dengan nilai 103.

Slide switch akan dihubungkan ke Arduino Mega sebagai input dengan menggunakan resistor *pull-up* yang terdapat pada Arduino Mega. Dimana satu kaki akan dihubungkan dengan pin GPIO Arduino Mega, dan satu kaki akan dihubungkan ke *ground* dari Arduino Mega. Untuk tabel daftar input *slide switch* yang akan digunakan dapat dilihat pada Tabel 4.

Tabel 4 Tabel Input *Slide Switch*

PIN GPIO	KODE EROR
22	Tekanan tinggi terdeteksi pada kondensor pada Chiller 1
23	Tekanan rendah terdeteksi pada evaporator Chiller 1
24	Temperatur kompresor terlalu tinggi pada Chiller 1
25	Tekanan tinggi terdeteksi pada kondensor pada Chiller 2
26	Tekanan rendah terdeteksi pada evaporator Chiller 2
27	Temperatur kompresor terlalu tinggi pada Chiller 2
28	Tekanan tinggi terdeteksi pada kondensor pada Chiller 3
29	Tekanan rendah terdeteksi pada evaporator Chiller 3
30	Temperatur kompresor terlalu tinggi pada Chiller 3
31	Tekanan tinggi terdeteksi pada kondensor pada Chiller 4
32	Tekanan rendah terdeteksi pada evaporator Chiller 4
33	Temperatur kompresor terlalu tinggi pada Chiller 4

E. Desain HMI

HMI untuk program *chiller sequencing* ini terdiri dari 7 halaman. Untuk penjelasan fungsi dari tiap halaman HMI adalah sebagai berikut :

- Halaman “*Chiller Plant Overview (Text-Based)*”, yang berfungsi untuk menampilkan informasi mengenai status *multiple-chiller-plant* dalam bentuk teks.
- Halaman “*Setting Parameter*”, yang berfungsi untuk memberikan *input* nilai periode *threshold* dan periode *delay* pematian tiap komponen
- Halaman “*Chiller Plant Overview*”, yang berfungsi untuk menampilkan informasi mengenai status *multiple-chiller-plant* dalam bentuk gambar.
- Halaman Status CHWP, CWP, CT, dan *Chiller*, yang berfungsi untuk menampilkan informasi mengenai *runtime*, *lifetime*, urutan *sequencing*, dan *alarm* dari tiap komponen.

III. PENGUJIAN DAN HASIL

Pada bab ini akan dilakukan beberapa pengujian untuk mengetahui fungsionalitas dari potensiometer sebagai input manual untuk nilai T_{rn} , program *chiller sequencing control*, HMI, dan komunikasi antara Arduino Mega dengan Kontroler OPEN EMS 600 I32.

A. Pengujian Konfigurasi Potensiometer

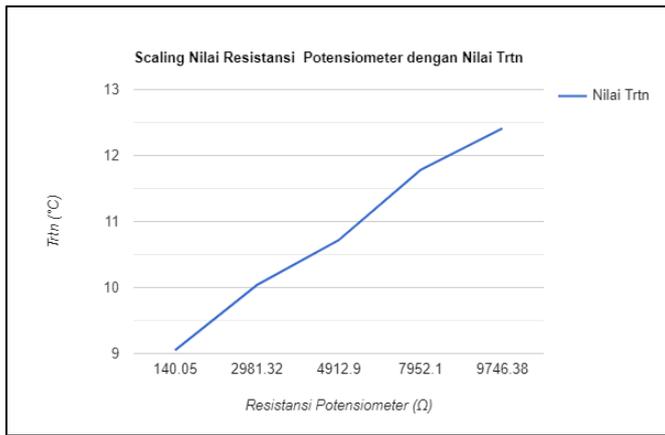
Pengujian ini dilakukan untuk mengetahui apakah potensiometer mampu mensimulasikan nilai T_{rn} dengan nilai minimum 9°C, dan maksimum 12.5°C dengan linear.

Terlihat pada Gambar 6, potensiometer sudah mampu menjalankan fungsi tersebut dengan persamaan seperti berikut :

$$Trtn = \frac{3.5}{10000} \times \text{Nilai Resistansi Potensio} + 9^{\circ}\text{C}$$

B. Pengujian Program Chiller Sequencing Control

Pengujian ini berguna untuk mengetahui apakah program *chiller sequencing control* telah mampu melakukan *sequencing chiller plant* dengan benar dengan cara mengoperasikan komponen-komponen dengan *running time* per 8 jam yang terendah. Pengujian dilakukan dengan cara menjalankan program selama 48 jam dengan nilai temperatur air pada *header* pipa CHWR disimulasikan dengan menggunakan input manual, lalu setiap 2 jam dilihat komponen mana yang sedang dalam keadaan *running* dan *not running* dan periode *threshold* yang digunakan adalah 30 menit dan *running time* awal tiap komponen adalah 0.



Gambar 6 Scaling nilai Trtn terhadap nilai potensimeter

Pada Gambar 7, terdapat hasil pengujian *sequencing* untuk *chiller*. Dari gambar tersebut, dapat terlihat bahwa program mampu mendeteksi jumlah *chiller plant* yang dibutuhkan berdasarkan parameter T_{rtn} . Selain itu, program juga mampu mengurutkan *chiller-chiller* berdasarkan dari yang memiliki *running time* per 8 jam terendah hingga ke yang memiliki *running time* per 8 jam tertinggi. Apabila terdapat 2 komponen atau lebih yang memiliki *running time* per 8 jam yang sama, maka program akan memberikan prioritas sebagai berikut, *chiller 1* -> *chiller 2* -> *chiller 3* -> *chiller 4*. Pengujian juga dilakukan pada CHWP, CWP, dan CT, dan hasilnya juga sama seperti Gambar 7.

C. Pengujian HMI

Pengujian ini bertujuan untuk mengetahui apakah HMI sudah mampu memvisualisasikan keadaan pada *multiple-chiller-plant* dengan tepat atau belum. Pengujian dilakukan dalam 2 skenario, yaitu ketika semua komponen *multiple-chiller-plant* berjalan dengan normal, dan ketika ada komponen yang eror. Ketika semua komponen berjalan dengan normal, maka tampilan halaman HMI juga normal. Gambar 8 hingga Gambar 10 merupakan tampilan halaman HMI ketika terdapat sebuah komponen yang mengalami eror, pada kasus ini, *chiller 2* mengalami eror. Ketika hal tersebut terjadi, maka pada

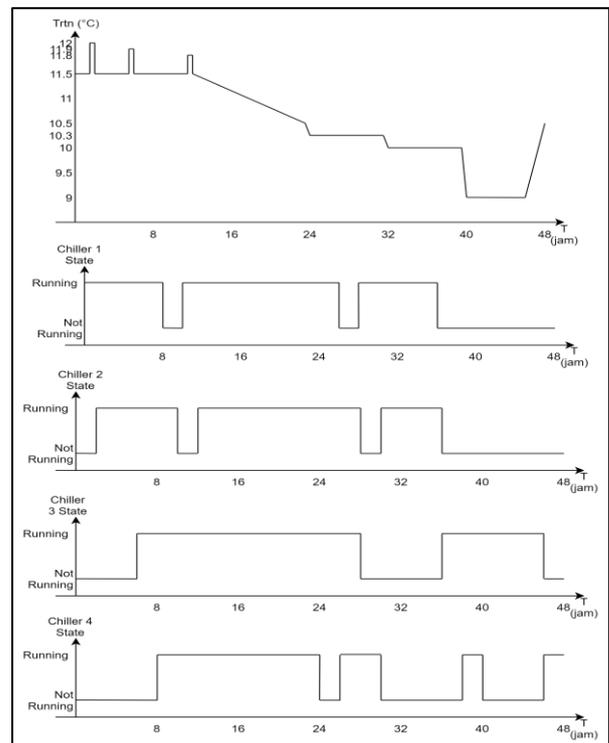
halaman “*Chiller Plant Overview (Text-Based)*”, “*Chiller Plant Overview*”, dan halaman status *chiller* akan muncul tanda eror pada *chiller 2*. Lalu, pada *alarm list* di halaman “*Chiller Plant Overview (Text-Based)*” akan muncul informasi bahwa *chiller 2* sedang mengalami eror.

D. Pengujian Komunikasi Modbus antara Arduino Mega dengan Kontroler OPEN EMS 600 I32

Pengujian ini dilakukan untuk mengetahui apakah data-data yang dikirimkan oleh Arduino Mega, nilai T_{rtn} yang diterima dan kode-kode eror tiap *chiller*, tepat.

Tabel 5 menunjukkan hasil pembacaan nilai T_{rtn} oleh OPEN EMS 600 I32 dari Arduino Mega. Dapat dilihat bahwa nilai T_{rtn} akan terus naik setiap 15 detik sekali sebesar 0.2°C hingga nilai T_{rtn} melebihi nilai batas atas, yakni 12.6°C , setelah itu nilai T_{rtn} akan terus turun setiap 15 detik sekali sebesar 0.2°C hingga nilai T_{rtn} beada di bawah nilai batas bawah, yakni 9°C .

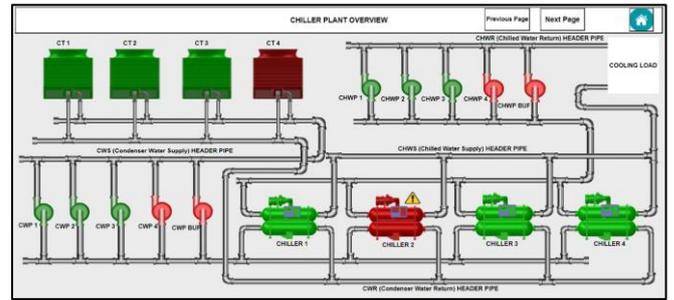
Tabel 6 menunjukkan hasil pembacaan kode eror pada tiap *chiller* oleh OPEN EMS 600 I32 dari Arduino Mega. Dapat dilihat bahwa setiap pin yang *ditrigger* hanya akan memberikan input kode eror yang unik ke sebuah *chiller*.



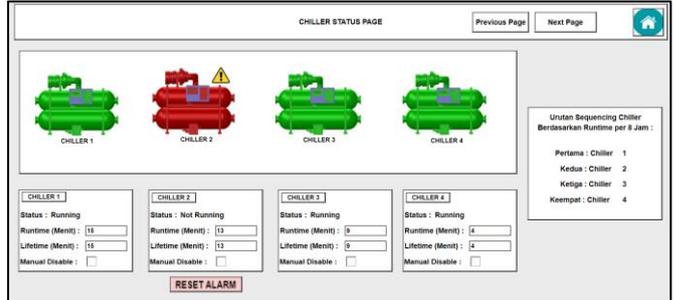
Gambar 7 Hasil pengujian *sequencing* untuk *chiller*

Tabel 5 Tabel Hasil Pembacaan Nilai Trtn yang Diterima Oleh OPEN EMS 600 I32

t(s)	Trtn (°C)	t(s)	Trtn (°C)
0	11.8	210	11
15	12	225	10.8
30	12.2	240	10.6
45	12.4	255	10.4
60	12.6	270	10.2
75	12.8	285	10
90	12.6	300	9.8
105	12.4	315	9.6
120	12.2	330	9.4
135	12	345	9.2
150	11.8	360	9
165	11.6	375	8.8
180	11.4	390	9
195	11.2	405	9.2



Gambar 9 Tampilan HMI untuk halaman "Chiller Plant Overview", ketika ada komponen yang error



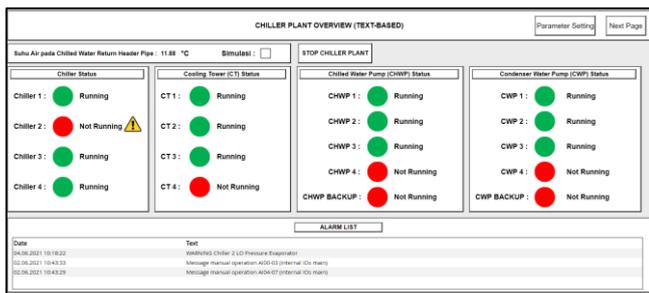
Gambar 10 Tampilan HMI untuk halaman status chiller ketika ada komponen chiller yang error

Tabel 6 Tabel Hasil Pembacaan Kode Error pada tiap Chiller yang Diterima Oleh OPEN EMS 600 I32

Trigger Pin Arduino	Kode Error			
	Chiller 1	Chiller 2	Chiller 3	Chiller 4
22	101	0	0	0
23	102	0	0	0
24	103	0	0	0
25	0	101	0	0
26	0	102	0	0
27	0	103	0	0
28	0	0	101	0
29	0	0	102	0
30	0	0	103	0
31	0	0	0	101
32	0	0	0	102
33	0	0	0	103

Keterangan :

101 -> Error Tekanan pada Kondensor Terlalu Tinggi
 102 -> Error Tekanan pada Evaporator Terlalu Rendah
 103 -> Error Suhu pada Kompresor Terlalu Tinggi



Gambar 8 Tampilan HMI untuk halaman "Chiller Plant Overview (Text-Based)" ketika ada komponen yang error

IV. KESIMPULAN

Dari perencanaan, implementasi, dan pengujian dari sistem chiller sequencing control ini, didapatkan kesimpulan sebagai berikut:

1. Program chiller sequencing control yang telah dibuat mampu untuk mengontrol 4 buah chiller plant dengan metode chilled water return temperature (t-based) control, yang memanfaatkan temperatur air pada header pipa chilled water return sebagai parameter untuk menentukan mati-nyalanya sebuah chiller plant, dimana urutan sequencing-nya tiap komponen-komponen utamanya (chiller, chilled water pump, condenser water pump, dan cooling tower) berdasarkan running time per 8 jam dari tiap-tiap komponen tersebut.
2. Berdasarkan pengujian yang dilakukan, HMI (Human Machine Interface) mampu menampilkan keadaan pada multiple-chiller-plant pada poin 1 baik pada saat multiple-chiller-plant sedang dalam keadaan normal dan pada saat ada salah satu atau lebih komponen utama yang sedang dalam kondisi error.
3. Telah berhasil dibuatnya simulator chiller plant dengan menggunakan Arduino Mega 2560, beserta komunikasinya dengan kontroler OPEN EMS 600 I32 dengan metode komunikasi Modbus RTU, dimana Arduino Mega dapat mengirimkan data kode error tiap chiller dan nilai simulasi suhu air pada header pipa chilled water return ke kontroler OPEN EMS 600 I32 dengan tepat, dan kontroler dapat menerima data-data tersebut dengan tepat.

V. DAFTAR PUSTAKA

[1] Y. Deng, Z. Feng, J. Fang and S. Cao, "Impact of ventilation rates on indoor thermal comfort and energy efficiency of ground-source heat pump

- system", *Sustainable Cities and Society*, vol. 37, pp. 154-163, 2018. Available: 10.1016/j.scs.2017.11.014.
- [2] Department of Climate Change and Energy Efficiency of Australia, "Boiler Efficiency," *energy.gov.au*. [Online]. Available: <https://www.energy.gov.au/sites/default/files/hvac-factsheet-boiler-efficiency.pdf>. [Accessed: 4-Jun-2021].
- [3] Balai Besar Teknologi Energi BPPT, "Perencanaan Energi dan Elastisitas Energi 2012", *Balai Besar Teknologi Konversi Energi*, 2012. [Online]. Available: <http://b2tke.bppt.go.id/images/Documents/PPID/SetiapSaaat/R%20-%20Publikasi%20Ilmiah/Perencanaan%20Efisiensi%20Dan%20Elastisitas%20Energi%202012.pdf>. [Accessed: 6-Jun-2021].
- [4] Wei, G. Xu and A. Kusiak, "Modeling and optimization of a chiller plant", *Energy*, vol. 73, pp. 898-907, 2014. Available: 10.1016/j.energy.2014.06.102.
- [5] Y. Liao, G. Huang, Y. Sun and L. Zhang, "Uncertainty analysis for chiller sequencing control", *Energy and Buildings*, vol. 85, pp. 187-198, 2014. Available: 10.1016/j.enbuild.2014.09.037.