

Automasi Sistem Kendali Mesin di Lini Blending 3 Ton PT Djarum Plant OASIS dengan PLC Siemens S7 300 dan Wincc Expoler V7.2

Happy Prakoso Adhi ¹, R. Rizal Isnanto ², Bambang Sudarsono ³

¹Program Studi Pendidikan Profesi Insinyur Fakultas Teknik, Universitas Diponegoro

²Departemen Teknik Komputer Fakultas Teknik, Universitas Diponegoro

³Departemen Teknik Geodesi Fakultas Teknik, Universitas Diponegoro
 Jl. Prof. Soedarto SH, Tembalang, Semarang, Indonesia 50275

^{*)}Coresponding author: happy.adhi87@gmail.com

(Received: February 13, 2023 ; Accepted: March 23, 2023)

Abstract

Along with the development of technology, industrial automation system start to take over the conventional system that has been running before. Human role to do manual setting to the machine start to take over by algorithm to produce tobacco with a good quality. Because the machine is controlled by algorithm, human error that often influence product quality can be minimized. To reach the target, Blending 3 Ton Line on Djarum OASIS plant had developed machine control system which flexible, easy to operate and maintain. For the brain of the control system, we use S7 300 PLC (Programmable Logic Controller), and for monitoring and operating the machine we use Wincc Explorer V7.2 SCADA (Supervisory Control and Data Acquisition). SCADA concept here covers data communication, data management, process monitoring and operation interface for the machine. On this automation system design base on Industrial Ethernet network which had been developed so that all data from machine can be monitored on SCADA and also can be saved on the database.

Keywords: Djarum Plant Oasis, Blending Line, Siemens, Industrial Ethernet, Automation.

Abstrak

Seiring dengan berkembangnya teknologi, sistem automasi industri telah mulai mengambil alih sistem konvensional yang selama ini sudah berjalan. Peranan manusia dalam melakukan setting mesin mulai digantikan dengan algoritma untuk menghasilkan produk yang berkualitas baik. Dengan dikendalikan oleh algoritma yang terstruktur, kesalahan manusia (human error) yang sering mengurangi stabilitas kualitas hasil produksi bisa diminimalkan. Untuk memenuhi target tersebut PT Djarum di Lini Blending 3 Ton Oasis telah mengembangkan sistem kendali mesin baru. Sistem kendali baru yang dikembangkan harus fleksibel, mudah dioperasikan, dan mudah dalam perawatan. Sebagai otak dari sistem kendali menggunakan PLC (Programmable Logic Controller) Siemens S7 300, sedangkan untuk kebutuhan pengawasan dan kendali proses digunakan SCADA (Supervisory Control and Data Acquisition) Wincc Explorer V7.2. Konsep SCADA ini mencakup komunikasi data, manajemen data, pengawasan proses, dan kendali terhadap mesin. Dalam desain sistem automasi ini didasari dengan jaringan komunikasi Industrial Ethernet antarmesin sehingga semua data dari plant dapat dimonitor di dalam SCADA dan disimpan di dalam basisdata.

Kata kunci: Djarum Plant Oasis, Lini Blending, Siemens, Industrial Ethernet, Automasi

How to Cite This Article: Adhi H.P., R. R., Isnanto, B. Sudarsono ., (2023), Automasi Sistem Kendali Mesin di Lini Blending 3 Ton PT Djarum Plant OASIS dengan PLC Siemens S7 300 dan Wincc Expoler V7.2, JPII 1 (4), 139-149

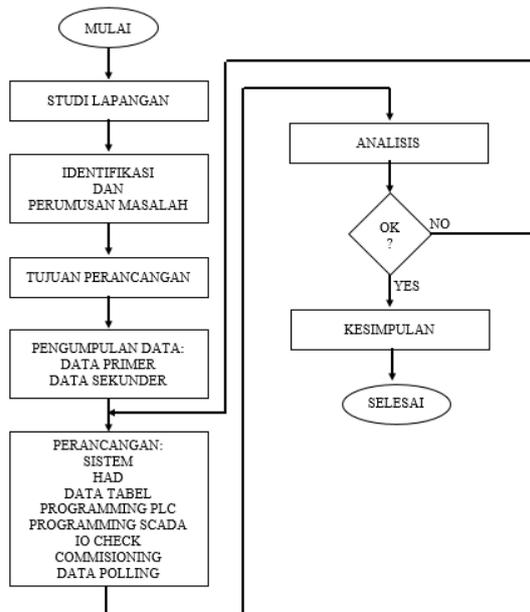
PENDAHULUAN

Automasi dalam industri atau sering disebut dengan *industrial automation* merupakan salah satu realisasi dari perkembangan teknologi yang merupakan alternatif untuk memperoleh sistem kerja yang cepat, akurat, efektif, dan efisien. Dalam sistem kontrol industri konvensional, peranan manusia masih sangat dominan, misalnya dalam memberikan tanggapan (*response*) kepada besaran-besaran proses yang diukur oleh sistem kontrol dengan serangkaian langkah berupa pengaturan panel dan saklar-saklar. Di masa kini peranan manusia yang masih dominan tersebut mulai digeser dan digantikan oleh sistem kontrol otomatis untuk meningkatkan efisiensi dan produktivitas mesin.

PT Djarum merupakan salah satu perusahaan pengolahan hasil tembakau di Indonesia. Perusahaan ini memiliki pusat operasional produksi Di Kudus, Jawa Tengah, yang salah satu plant operasional produksi terbesar yang dimilikinya bernama Oasis. Dengan tuntutan kapasitas produksi yang besar, maka keandalan sistem kontrol mesin menjadi hal yang sangat penting dalam proses produksi untuk mengejar kapasitas mesin dan kualitas produk yang ditargetkan. Oleh sebab itu, PT Djarum mengembangkan sistem automasi industri di semua lini produksinya untuk meningkatkan kapasitas, efisiensi, dan stabilitas kualitas produk yang dihasilkan .

METODE PENELITIAN

Sebelum melakukan desain suatu sistem kendali, kita tentukan terlebih dahulu diagram alir perancangan yang akan kita lakukan. Pelaksanaan perancangan mengacu pada diagram alir seperti dapat dilihat pada Gambar 1



Gambar 1. Gambar Diagram Alir Perancangan

Penulis mengawali penelitian ini dengan menentukan lokasi yang dijadikan subjek penelitian dalam menerapkan suatu perancangan sistem kendali secara otomatis pada lokasi tersebut. Latar belakang

pemilihan lokasi didasari kepada kebutuhan lini produksi yang memerlukan perbaikan sistem kendali dan sistem untuk menghasilkan mesin yang efisien, dapat diandalkan, dan memiliki hasil berkualitas tinggi.

Dalam studi lapangan ini dilakukan pengamatan di PT Djarum Plant OASIS pada Lini *Blending* untuk 6 Ton yang merupakan mesin OEM dari Jerman untuk mengamati lini produksi yang sudah berjalan, cara mengoperasikan mesin, dan standar kualitas yang ditargetkan. Penulis mulai melakukan observasi dan wawancara terhadap operator dan kepala bagian produksi tentang kendala, kesulitan dan masukan untuk perbaikan dalam cara mengoperasikan mesin Lini *Blending* 6 Ton untuk bisa diaplikasikan di *Blending* Lini 3 Ton yang akan dikerjakan.

Berdasarkan hasil studi lapangan, fokus penelitian pada karya ilmiah ini yaitu tentang perancangan sistem kendali otomatis di Lini *Blending* 3 Ton. Sistem kendali otomatis yang dirancang harus mempunyai beberapa kriteria, yaitu:

1. Tidak ada sistem kendali konvensional.
2. Mesin fleksibel sesuai kebutuhan produksi.
3. Mesin dengan efisiensi tinggi yang aman dioperasikan.
4. Mesin yang dapat diandalkan.
5. Mesin memiliki stabilitas produk dengan kualitas yang baik.
6. Mesin dapat dioperasikan dengan SCADA dari ruang kontrol, ada *logging* data proses dan *alarm*.

PERANCANGAN SISTEM

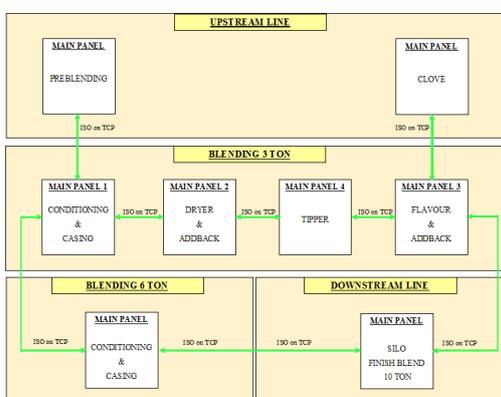
Dalam membuat desain sistem kendali suatu lini produksi, kita melakukan perancangan terlebih dahulu sistem seperti apa yang akan diterapkan. Dalam perancangan ini kita melakukan automasi di Lini *Blending* yang merupakan Lini Proses Tembakau terakhir sebelum tembakau dilinting menjadi batangan rokok. Dalam perancangan di Lini *Blending* 3 Ton OASIS, lini kita bagi menjadi 4 lini utama, yaitu:

1. Lini *Conditioning and Casing*
 Pada lini *Conditioning and Casing* ini merupakan proses pengkondisian *Moisture Content* (MC) tembakau. Setelah tembakau terkondisi pada suhu dan kelembaban tertentu di dalam Cylinder, ditambahkan cairan *Casing* ke dalam tembakau dengan suatu takaran tertentu sehingga didapatkan tembakau dengan spesifikasi yang ditentukan.
2. Lini *Dryer and Addback Prior*
 Pada Lini *Dryer and Addback Prior* ini merupakan proses pengeringan tembakau hasil dari proses *Conditioning*. Pada proses ini tembakau dikeringkan untuk mencapai suatu target kelembaban tertentu. Setelah keluar dari *Dryer Cylinder*, tembakau dicampur dengan *addback* tembakau dengan rasio tertentu, kemudian tembakau masuk ke dalam SILO *Ex Dryer*.
3. Lini *Flavour and Addback Post*
 Pada lini *Flavour and Addback Post* ini merupakan proses terakhir dari lini *Blending*. Disini tembakau ditambahkan *flavour* dan *addback post* dengan ratio tertentu, kemudian dicampur di dalam *Mixing Cylinder* dan kemudian di simpan ke dalam SILO *Finish Blend*.

4. Line Tipper

Pada lini *Tipper* ini merupakan lini yang merupakan lini pendukung di dalam lini Blending. Lini *Tipper* ini bertugas untuk melakukan *Tip* tembakau ke dalam Feeder di dalam Lini Blending.

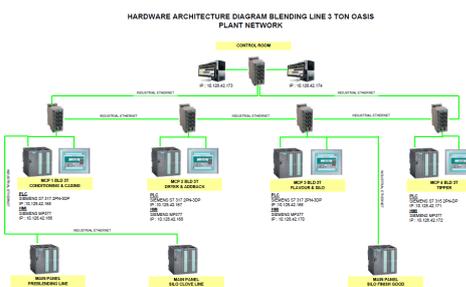
Pembagian lini ini dibagi berdasarkan fungsi dari masing-masing mesin. Jadi dalam perancangan ini dibagi dalam 4 Main Panel dengan 4 PLC sebagai pengendali utamanya. Dalam perancangan sistem ini kita tentukan komunikasi antar panel akan seperti apa, baik itu antar Main Panel di Blending 3 Ton maupun dengan *Upstream* dan *Downstream Line*. Dalam perancangan ini komunikasi antar panel di *Blending 3 Ton*, komunikasi dengan *upstream Line* dan komunikasi dengan *Downstream Line* menggunakan *Industrial Ethernet* dalam Platform *ISO on TCP*. Gambar 2 menggambarkan komunikasi antar panel tersebut.



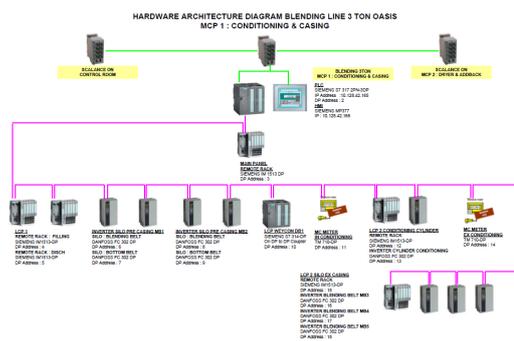
Gambar 2. Gambar Diagram Jaringan Komunikasi Lini Blending 3 Ton

Hardware Architectur Diagram (HAD)

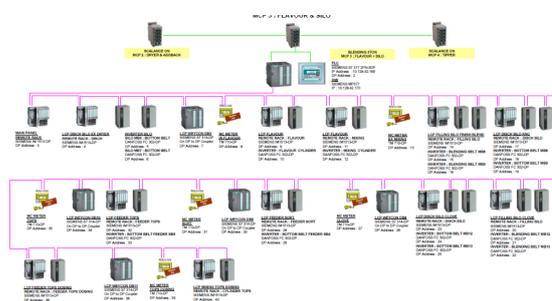
Setelah sistem dan pembagian panelnya ditentukan, kita bisa mulai melakukan desain *Hardware Architecture Diagram* (HAD) untuk jaringan komunikasi antar panel dan juga untuk jaringan komunikasi di masing masing panel dengan *slave*. Pada perancangan ini komunikasi antar main panel menggunakan *Industrial Ethernet* dengan Topologi *Ring* sedangkan untuk komunikasi antara *main panel* dengan *slave* menggunakan *profibus* dengan Topologi *Bus*. Pada gambar 3 dapat dilihat gambar *Hardware Architecture Diagram* Plant Blending 3 Ton Oasis.



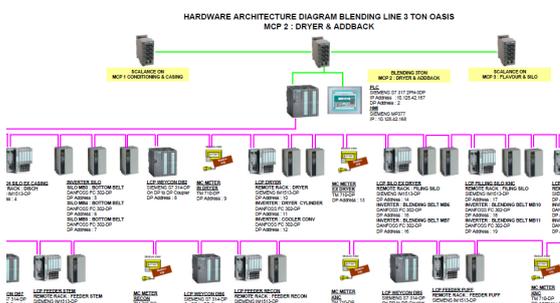
Gambar 3. Gambar HAD Plant Blending 3 Ton OASIS



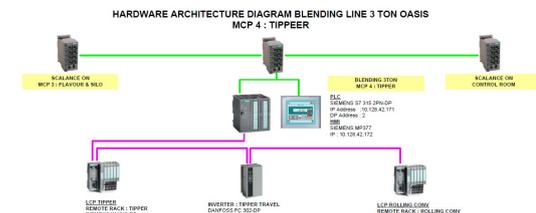
Gambar 4. Gambar HAD Lini Conditioning and Casing



Gambar 5. Gambar HAD Lini Dryer and Add-back Prior



Gambar 6. Gambar HAD Lini Flavour & Add-back Post



Gambar 7. Gambar HAD Lini Tipper

Pada gambar 4, gambar 5, gambar, 6 dan gambar 7 dapat dilihat *Hardware* masing-masing lini di *Blending 3 Ton OASIS*. Pada gambar diatas dapat dilihat komunikasi antar Main Panel dengan *Ethernet*, dan komunikasi antara main panel dengan *slave* dengan *Profibus*.

Data Table

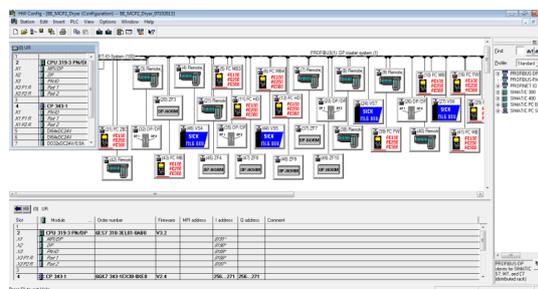
Setelah *Hardware Architecture Diagram* selesai dibuat dan pembagian panel sudah jelas, maka kita bisa mulai membuat data tabel dari program yang akan kita desain. Dalam data tabel tersebut kita menyusun urutan *List Input* dan *Output* dari lini yang akan kita desain. Dalam data tabel itu direncanakan secara mendetail apa saja *input* dan *output* yang kita perlukan dalam program, dari motor, sensor, isolator, *safety* dan *analog* yang kita perlukan dari plant ke PLC.

Setelah data tabel selesai disusun, data tabel dan *Hardware Architecture Diagram* kita kirimkan kepada desainer elektrik untuk dilakukan pengecekan apakah sudah sesuai dengan *Machine List* yang akan kita desain. Apabila sudah tidak ada kekurangan, maka desain elektrik panel siap dibuat berdasarkan *Hardware Architecture Diagram* dan Data Tabel yang sudah kita buat.

Programming PLC

Setelah *Hardware Architecture Diagram* dan Data Tabel sudah selesai dibuat, kita bisa memulai membuat program untuk lini yang akan kita desain. Dalam program kali ini kita gunakan *Software STEP 7 V5.5 SP4* dan kita desain dengan bahasa pemrograman *Ladder diagram* untuk logika sistem dan motor, sedangkan untuk aritmatika kita menggunakan bahasa pemrograman *Statement List (STL)*.

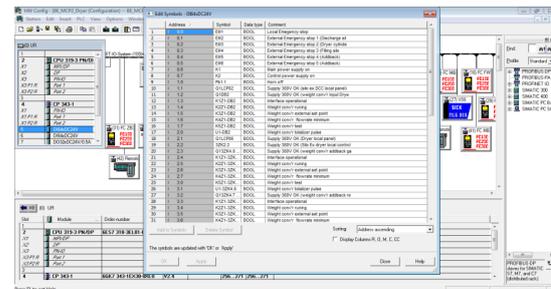
Pada awal melakukan desain program PLC, kita melakukan konfigurasi perangkat keras (*hardware*) sesuai dengan *Hardware Architecture Diagram* dan Data Tabel yang sudah kita buat. Dalam konfigurasi *hardware* ini juga kita memberikan alamat *Ethernet* dan *Profibus* yang sudah kita rencanakan ke dalam PLC. Dalam melakukan konfigurasi *hardware*, kita juga harus melihat barang yang akan kita pasang dikarenakan *firmware* yang kita konfigurasi pada program harus sesuai dengan *firmware device* yang terpasang di panel. Apabila ada perbedaan, maka pada saat didownload akan menghasilkan *error* pada PLC yang ditunjukkan dengan menyalnya indikator SF (*System Fault*). Apabila hal itu terjadi, maka bisa kita buka *Hardware Configuration* pada Step 7 seperti ditunjukkan pada Gambar 8 dan kita klik Ctrl+D (*Diagnostic Buffer*) untuk melihat apakah yang menyebabkan *System Fault* dan *device* mana yang menyebabkan *System Fault*.



Gambar 8 STEP 7 Hardware Configuration

Setelah kita selesai melakukan konfigurasi *hardware*, maka kita bisa mulai memasukan deskripsi pada *Input* dan *Output* yang sudah kita desain dalam

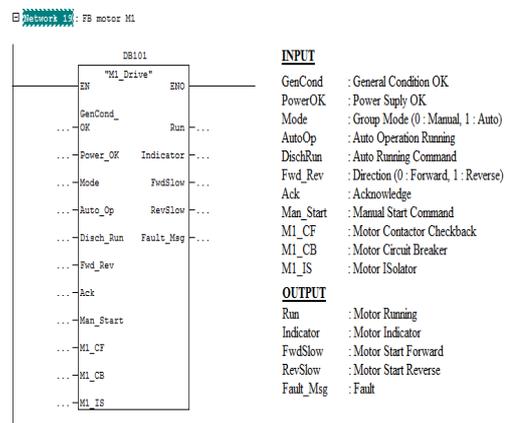
data tabel seperti ditunjukkan pada Gambar 4.11. Desain deskripsi *Input* dan *Output* yang sesuai akan memudahkan kita kedepannya apabila ingin melakukan *trace* dalam program. Apabila dalam program tidak diberikan deskripsi yang benar dan sesuai pada *Input* dan *Output*, maka akan membuat kita kesulitan apabila ingin melakukan pemrograman karena kita harus selalu melihat pada gambar *hardware* atau data tabel secara terus menerus pada saat akan melakukan *trace* program.



Gambar 9. Gambar Hardware Input / Output Description

Dalam membuat program, kita biasanya membuat *User Define Data Type (UDT)*. Seperti terlihat pada Gambar 4.12, UDT ini kita buat untuk memudahkan kita dalam membuat *Structure Tag* yang sering kita gunakan sehingga kita mempunyai standar yang sama dalam membuat *Tag* dalam *Data Block* yang akan kita desain. UDT ini juga memudahkan kita apabila kita akan mendesain Blok Fungsi (*Function Block*) apabila kita ingin membuat *Input* maupun alamat statis (*Static Address*) yang berupa Struktur atau latik (*Array*). *User Define Data Type* ini akan mempermudah apabila kita ingin membuat struktur *Data Block* maupun *Function Block*, karena hanya dengan mengambil satu buah UDT, kita akan mendapatkan *Structure Tag* di dalamnya.

Dalam membuat program ini, kita menggunakan *Function Block* untuk blok program yang sering kita gunakan sehingga program menjadi lebih sederhana dan mempunyai standar yang sama. Dalam desain ini kita ada yang menggunakan *Function Block* bawaan dari Siemens untuk PID, dan ada juga yang kita buat sendiri sesuai dengan kebutuhan untuk motor, *valve* dan aritmatika.



Gambar 10. Gambar Function Block Motor DOL

Pada Gambar 10 terlihat salah satu contoh *Function Block* yang kita desain yaitu *Function Block* untuk Motor DOL. Di dalam *Function Block* tersebut sudah terdapat logika program yang sudah kita desain untuk menjalankan motor, lengkap dengan *fault*, *state* motor dan *running logic* nya. Maka apabila kita ingin membuat program untuk sebuah motor, kita tinggal memanggil *Function Block* tersebut, dan mengisikan alamat pada kaki *input* dan *output* yang sudah tersedia. Maka kita akan mendapatkan kendali suatu motor sesuai dengan standar yang sudah kita tentukan.

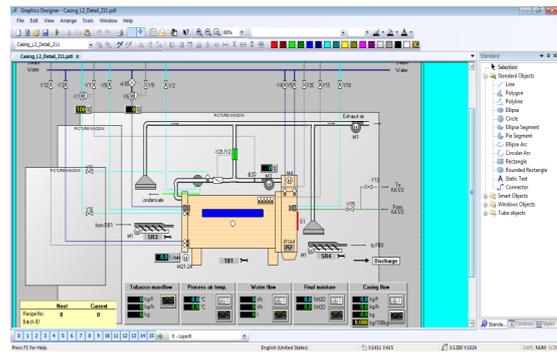
Di dalam program ini juga terdapat *Function Block* untuk PID untuk mengatur suhu *Conditioning Cylinder*, *Dryer*, MC tembakau dan juga *flow* cairan untuk *dossing*. Didalam *Function Block* yang sudah disediakan oleh Siemens kita tinggal mengisikan parameter yang kita perlukan dan kemudian melakukan *tunning* parameter PID sesuai permodelan dan perhitungan yang sudah kita tentukan untuk mencapai target yang kita tentukan. Selain itu di dalam program juga terdapat *Function Block* untuk data statistik yang kita gunakan untuk menyimpan data dan menganalisis secara otomatis di dalam program apakah proses tembakau yang dilakukan di dalam mesin sudah sesuai dengan spesifikasi produk yang ditargetkan.

Setelah semua program selesai dibuat, kita bisa melakukan simulasi program untuk mengurangi kesalahan program pada saat di aplikasikan di lapangan. Simulasi program kita lakukan dengan PLC *virtual* dari Siemens yaitu dengan *Software PLCSIM*. Jadi setelah *PLCSIM* dijalankan, program yang kita sudah desain bisa kita unduh (*download*) ke dalam *PLCSIM* kemudian logika program yang sudah dibuat kita bisa simulasikan dengan *Variable Table* (*VAT*) dalam Step 7. Kita bisa melakukan *force* nilai di dalam *Data Block* untuk melihat logika program apakah sudah berjalan sesuai dengan yang kita inginkan.

Programming SCADA

Setelah program PLC selesai, kita mulai membuat program SCADA yang digunakan untuk mengendalikan mesin. Untuk *software* yang digunakan, kita gunakan *Simatic Wincc Explorer V7.2* untuk melakukan *development* dan *run time* SCADA yang akan kita desain. Pada gambar 4.16 dapat kita lihat tampilan antarmuka *Wincc Explorer V7.2*. Dalam mendesain SCADA yang kita buat, pertama tama kita gambarkan *layout* mesin yang akan kita monitor sesuai dengan gambar *layout* mekanik di lapangan. *Layout* ini nantinya akan diberikan animasi sesuai dengan nomor item mesin yang berjalan di lapangan sehingga dapat mewakili kondisi di lapangan untuk dimonitor oleh operator produksi.

Di dalam SCADA juga digambarkan detail mesin yang kita kendalikan termasuk P&ID diagram, sensor dan detail motor di dalam mesin sehingga apa yang tertampil di HMI nantinya benar benar mewakili apa yang berjalan di lapangan. Pada Gambar 4.18 menunjukkan desain P&ID *Conditioning Cylinder*.

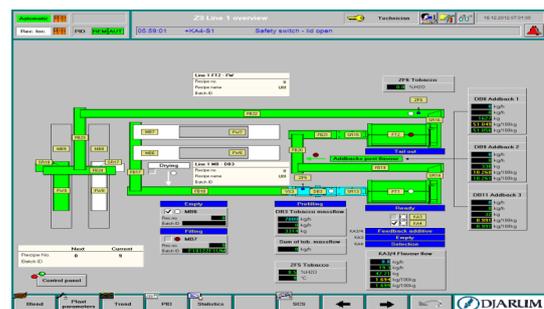


Gambar 11 Desain Gambar P&ID Conditioning

Setelah semua gambar grafik selesai dibuat kita mulai masukan *Tag* PLC sesuai dengan yang akan ditampilkan ke dalam HMI. Tipe data *Tag* dan alamat yang kita buat di dalam HMI harus sesuai dengan alamat dan tipe data di dalam PLC. Kita bisa juga memasukan *Tag* secara lebih cepat dengan menggunakan *tools Export and Import Tag* yang sudah disediakan di dalam *Wincc Explorer*. Untuk lebih memudahkan, *Tag* kita buat dalam *group* sehingga memudahkan kita pada saat *browse Tag* pada saat akan memilih *Tag* pada halaman *Graphic Designer*.

Setelah *Tag* selesai kita *import*, kita mulai bisa membuat *script* untuk menyimpan data proses ke dalam *database*. Disini dalam membuat *script* digunakan bahasa pemrograman *script* Virtual Basic. Disini penyimpanan data akhir proses dan data proses setiap periode tertentu kita simpan ke dalam tabel *database* untuk bisa ditampilkan lagi sesuai dengan kebutuhan. Selain itu kita juga mulai bisa melakukan input *alarm logging* ke dalam *Alarm Tag Management* untuk melakukan penyimpanan data *alarm* dan *warning* ke dalam *internal database Wincc Explorer*.

Setelah semuanya selesai dibuat, kita bisa mulai melakukan *runtime* dengan menghubungkan komputer SCADA ke jaringan *ring* PLC untuk melihat apakah yang ingin kita tampilkan sudah tertampil dengan benar. Disini kita monitor *layout*, detail mesin, data *logging*, *alarm logging*, *trend* dan penyimpanan data proses pada *database* apakah sudah sesuai dengan desain kita. Selain itu kita juga melakukan pengecekan *realtime* pembacaan data apakah ada gangguan yang bertujuan untuk mengecek apakah jaringan *ring Industrial Ethernet* yang sudah kita konfigurasi terjadi gangguan dan interferensi atau tidak.



Gambar 12 HMI Layout Window

IO Check

Setelah semua program PLC selesai dibuat dan dilakukan simulasi dengan PLCSIM dan juga sudah selesai dilakukan koneksi dari PLCSIM ke SCADA untuk melakukan simulasi SCADA, maka semua program siap *download* ke panel di lapangan. Ketika instalasi mekanik dan elektrikal di lapangan selesai dilakukan, maka kita siap untuk *download software* PLC ke panel dan kita siap untuk mulai melakukan *check IO*. Ada beberapa langkah yang dilakukan ketika *check IO*, yaitu:

1. Download PLC program

Langkah pertama yang dilakukan pada saat *check IO* adalah *Power On* panel yang sudah dirangkai. Sebelum panel dinyalakan, kita melakukan *continuity check* terhadap koneksi panel untuk memastikan tidak ada *short*. Setelah semua dicek dan panel dipastikan aman untuk dinyalakan, kita bisa mulai menyalakan panel. Setelah panel menyala, kita *download* program PLC yang sudah kita buat ke dalam PLC.

2. Konfigurasi Rack PLC dan cek komunikasi PLC dengan slave

Setelah PLC selesai diunduh maka PLC akan kembali ke dalam mode *RUN*. Pada awalnya kita biasanya masih melihat lampu SF dan BF di dalam PLC masih menyala. Hal ini biasanya terjadi dikarenakan *rack* yang kita konfigurasi dalam program belum menemukan *device* di dalam jaringan. Yang kita lakukan adalah melakukan pengecekan kabel *profibus* dari *main* panel ke *slave*, melakukan pengecekan *address profibus device* apakah sudah sesuai dengan yang kita konfigurasi di dalam program, melakukan pengecekan terminasi *end* pada jaringan *profibus*. Apabila semua sudah kita lakukan tetapi *main* panel dan *slave* yang dituju masih menyala *blinking* merah, maka kita lakukan *check firmware* dari *device* yang telah kita konfigurasi. Hal ini bisa kita *check* dengan menekan *Ctrl+D* (*Diagnostic Buffer*) dari *Hardware Configuration* Step 7. Dari sini kita bisa melakukan *crosscheck* apa yang menyebabkan *error* terjadi. Setelah semua selesai dan sudah tidak ada lampu indikator led menyala, maka PLC dan jaringan *slave* sudah *ready* dan kita bisa melanjutkan ke tahap selanjutnya.

3. Check IO

Setelah semua konfigurasi *rack* selesai, kita mulai melakukan *check IO* dari masing masing *input* dan *output* yang sudah kita susun di dalam data tabel. Kita cocokkan dan cek *input* dan *output* yang ada di dalam data tabel dengan yang terpasang di lapangan. Bagian yang sudah sesuai kita berikan *check list* untuk menandakan data tabel dan IO sudah selesai kita cek. Selain melakukan pengecekan dari I/O PLC, kita juga melakukan pengecekan dari HMI apakah *input* dan *output* yang kita cek sama dengan indikator di HMI. Sehingga diharapkan apa yang tertampil di HMI mewakili apa yang ada di lapangan.

4. Test manual run mesin

Setelah *Check I/O* selesai, kita melakukan test manual satu per satu mesin. Kita start secara manual item mesin yang akan kita cek dari HMI. Setelah mesin berjalan, kita cek indikator di dalam HMI apakah sudah sesuai, kita cek arah putaran mesin apakah sudah sesuai dan kita lakukan *setting* mesin tersebut

(pengencangan belt, *setting* scrap dan sebagainya). Setelah itu kita juga melakukan *setting* ampere dari MCCB pada saat mesin berjalan tanpa beban. Setelah itu kita melakukan test kita lakukan *fault* pada mesin (*trip*, isolator & *safety*), kemudian kita lihat apakah indikator pada HMI dan *alarm banner* sudah sesuai. Apabila sudah sesuai kita lakukan *Acknowledge* dan kita start manual kembali. Kita lakukan satu per satu pada item mesin sampai semua item selesai di cek.

5. Setting device

Setelah semua item kita lakukan cek *running manual*, kita lakukan *setting device* yang ada pada mesin seperti *MC meter*, *Control Valve*, sensor, *inverter* dan sebagainya. Kita baca datanya pada HMI apakah sudah sesuai *setting* seperti yang kita tentukan.

6. Membuat komunikasi antar panel

Setelah selesai melakukan *check running manual* dan *setting device*, kita buat program untuk komunikasi PLC antar panel dalam *platform ISO on TCP*. Kita cek dalam *NetPro Configuration* pada Step 7, kita *download* konfigurasi sampai dengan pada *NetPro Configuration* komunikasi antar panel dalam kondisi *Established*. Setelah dalam kondisi *Established*, kita bisa mulai membuat programnya. Kita gunakan *Function AG SEND* dan *AG RECEIVE*. Kita atur urutan data yang akan kita kirim dan terima sesuai dengan list data yang sudah kita buat.

7. Test Automatic Dry Run

Setelah komunikasi antar panel selesai dibuat dan data antar panel sudah bisa ditransmisikan kita bisa melakukan *Automatic Dry Run Test*. Disini bertujuan untuk melakukan pengecekan *sequence-sequence* mesin apakah sudah sesuai dengan desain. Kita lakukan *start group* dan cek *sequence-sequence* yang berjalan dalam mesin.

Endurance test

Setelah *dry run* dan simulasi lancar dilakukan, mesin kita lakukan secara berkelanjutan untuk mengecek ketahanan dan keandalan mesin apabila dijalankan secara terus menerus.

Commissioning

Setelah semua rangkaian tahapan dalam *Check IO* selesai dilakukan, kita mulai melakukan *commissioning* mesin menggunakan material. *Commissioning* adalah rangkaian aktivitas dalam rangka pengujian kemampuan / keandalan mesin yang sudah diinstal dan kemudian dioperasikan sesuai dengan desain yang sudah ditetapkan. Dalam *commissioning* ini kita menjalankan mesin sesuai standar operasi yang sudah dibuat dan kemudian kita amati apakah mesin yang kita install mampu mencapai target spesifikasi produk yang ditargetkan. Target spesifikasi produk biasanya dilihat dari suatu nilai set point produk yang telah tercapai dengan nilai *Process Capability Index* (CPK) ≥ 1.33 . apabila bisa mencapai nilai CPK tersebut maka dianggap mesin bisa menghasilkan produk dengan spesifikasi tertentu dengan merata dan stabil.

HASIL DAN PEMBAHASAN

Setelah kita selesai melakukan *commissioning*, kita dapat menarik data proses yang sudah berjalan dari basisdata SCADA. Pada *database* SCADA kita

bisa mengambil data akhir proses maupun data *logging* sepanjang proses. Kemudian data dianalisis untuk mengetahui apakah produk yang dihasilkan mesin sudah sesuai dengan spesifikasi produk yang ditargetkan.

Proses Conditioning and Casing

Dari *database* SCADA, kita mengambil data akhir proses Lini *Conditioning and Casing* untuk kita analisis apakah produk yang dihasilkan mesin sudah sesuai dengan spesifikasi produk yang di targetkan. Data akhir proses Lini *Conditioning and Casing* pada saat *commissioning* dapat dilihat pada Tabel 1.

Tabel 1. Tabel Hasil Proses Conditioning and Casing

PROCESS	AVG MC IN DCC	SD MC IN DCC	SP RATIO CASING	PV RATIO CASING	AVG RATIO CASING
1	13,420	0,120	16,406	16,412	16,411
2	13,420	0,110	16,406	16,414	16,413
3	13,4	0,07	16,406	16,413	16,407
4	11,82	0,06	7,727	7,731	7,725
SD RATIO CASING	AVG MC OUT	SD MC OUT	AVG TEMP OUT	SD TEMP OUT	
0,008	23,18	0,16	58,91	0,72	
0,009	23,17	0,15	59,25	0,69	
0,009	23,12	0,1	59	0,46	
0,006	27,5	0,21	57,01	0,72	

Dari data akhir proses di atas, dapat kita lihat bahwa semua *set point* yang ditargetkan dapat dicapai oleh mesin. Namun untuk melihat apakah pencapaian itu merata kita juga melihat dari nilai Koefisien Variasi (KV) nya. Koefisien Variasi (KV) adalah suatu sistem perbandingan antara simpangan standar dengan nilai rata-rata yang dinyatakan dalam bentuk presentase.

- *Ratio Casing*
Spesifikasi target nilai KV maksimal = 0,2
Proses Pertama

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,008}{16,411} * 100 = 0,0487$$

Proses Kedua

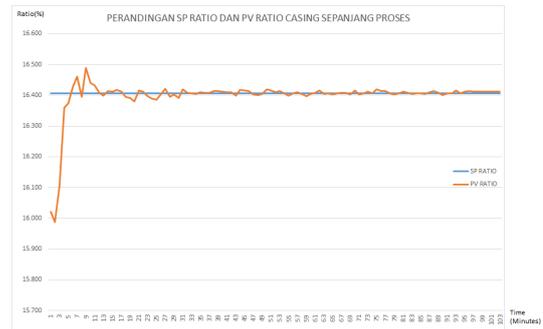
$$KV = \frac{Sd}{Avg} * 100 = \frac{0,009}{16,413} * 100 = 0,0549$$

Proses Ketiga

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,009}{16,407} * 100 = 0,0548$$

Dari hasil perhitungan KV ketiga proses di atas, terlihat bahwa *set point ratio casing* sudah tercapai dan nilai KV sudah memenuhi spesifikasi produk yang ditentukan. Perbandingan antara *Set Point ratio* dan nilai aktual sepanjang proses dapat dilihat pada

Gambar 13.



Gambar 13 Grafik Perbandingan SP dan PV Ratio Casing

- MC Keluaran *Conditioning*
Spesifikasi target nilai KV maksimal = 1,5
Proses Pertama

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,16}{28,18} * 100 = 0,6902$$

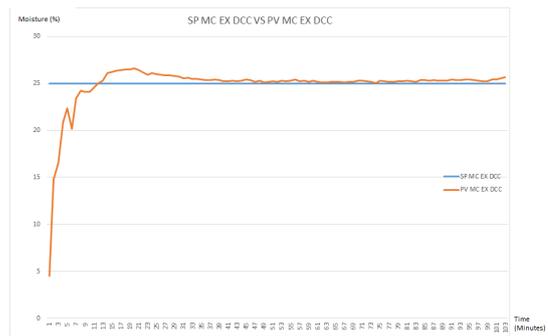
Proses Kedua

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,15}{23,17} * 100 = 0,6473$$

Proses Ketiga

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,1}{23,12} * 100 = 0,4325$$

Dari hasil perhitungan KV ketiga proses di atas, terlihat bahwa *set point MC output Conditioning* sudah tercapai dan nilai KV sudah memenuhi spesifikasi produk yang ditentukan. Perbandingan antara *Set Point MC Output Conditioning* dan nilai aktual sepanjang proses dapat dilihat pada Gambar 14 di bawah ini.



Gambar 14 Grafik Perbandingan SP dan PV MC Output DCC

- Suhu Keluaran *Conditioning*
Spesifikasi target nilai KV maksimal = 2,5
Proses pertama

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,86}{64,73} * 100 = 1,3285$$

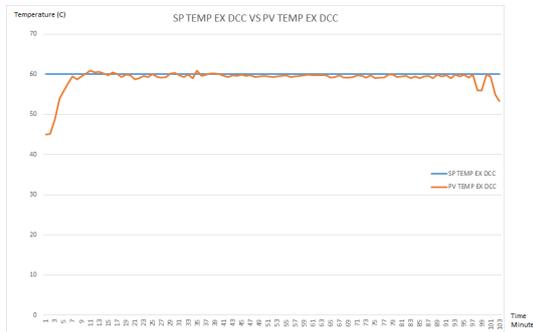
Proses Kedua

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,83}{64,76} * 100 = 1,2816$$

Proses Ketiga

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,79}{64,9} * 100 = 1,2172$$

Dari hasil perhitungan KV ketiga proses di atas terlihat bahwa *set point* suhu keluaran *Conditioning* sudah tercapai dan nilai KV sudah memenuhi spesifikasi produk yang ditentukan. Perbandingan antara *Set Point* Suhu keluaran *Conditioning* dan nilai aktual sepanjang proses dapat dilihat pada Gambar 15.



Gambar 15 Grafik Perbandingan SP dan PV Temperature Output DCC

Proses Dryer and Addback Prior

Dari *database* proses kita ambil data akhir proses *dryer and addback prior* untuk kita analisis apakah produk yang dihasilkan mesin sudah sesuai dengan spesifikasi produk yang di targetkan. Data akhir proses Lini *Dryer and Addback Prior* pada saat *commissioning* dapat dilihat pada Tabel 2

Tabel 2. Tabel Hasil Proses *Dryer and Addback Prior*

PROCESS	SP RATIO ADDBACK 1	PV RATIO ADDBACK 1	AVG RATIO ADDBACK 1	SD RATIO ADDBACK 1
1	23,907	23,921	24	0,019
2	23,907	23,764	23,944	0,022
3	23,907	23,916	23,933	0,022
SP RATIO ADDBACK 2	PV RATIO ADDBACK 2	AVG RATIO ADDBACK 2	SD RATIO ADDBACK 2	AVG MC OUT DRYER
6,461	6,465	6,464	0,005	16,5
6,461	6,421	6,471	0,006	16,5
6,461	6,461	6,467	0,006	16,5
SD MC OUT DRYER	SP TEMP OUT DRYER	AVG TEMP OUT DRYER	SD TEMP OUT DRYER	
0,1	55	53,33	0,27	
0,15	55	53,23	0,32	
0,1	55	53,2	0,27	

Dari data akhir proses di atas dapat kita lihat bahwa semua *set point* yang ditargetkan dapat dicapai oleh mesin. Namun untuk melihat apakah pencapaian itu merata, kita biasanya juga melihat dari nilai Koefisien Variasi (KV) nya.

• *Ratio Addback Prior 1*

Spesifikasi target nilai KV maksimal = 0,2
Proses Pertama

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,019}{24,0} * 100 = 0,0794$$

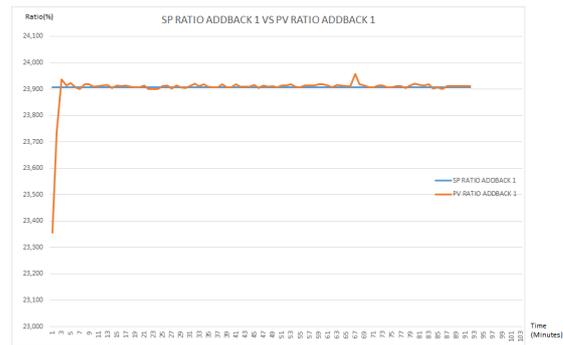
Proses Kedua

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,022}{23,944} * 100 = 0,0918$$

Proses Ketiga

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,022}{23,933} * 100 = 0,0919$$

Dari hasil perhitungan KV ketiga proses di atas terlihat bahwa *set point ratio addback prior 1* sudah tercapai dan nilai KV sudah memenuhi spesifikasi produk yang ditentukan. Perbandingan antara *Set Point ratio addback prior 1* dan nilai aktual sepanjang proses dapat dilihat pada Gambar 16.



Gambar 16 Grafik Perbandingan SP dan PV Ratio Addback Prior 1

• *Ratio Addback Prior 2*

Spesifikasi target nilai KV maksimal = 0,2
Proses Pertama

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,005}{6,464} * 100 = 0,0773$$

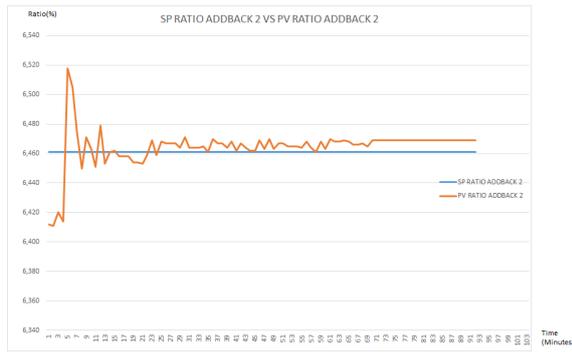
Proses Kedua

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,006}{6,471} * 100 = 0,0927$$

Proses Ketiga

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,006}{6,467} * 100 = 0,0927$$

Dari hasil perhitungan KV ketiga proses di atas terlihat bahwa *set point ratio addback prior 2* sudah tercapai dan nilai KV sudah memenuhi spesifikasi produk yang ditentukan. Perbandingan antara *Set Point ratio addback prior 2* dan nilai aktual sepanjang proses dapat dilihat pada Gambar 17.



Gambar 17 Grafik Perbandingan SP dan PV Ratio Addback Prior 2

• *MC Output Dryer*

Spesifikasi target nilai KV maksimal = 1.5
Proses Pertama

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,1}{16,37} * 100 = 0,6108$$

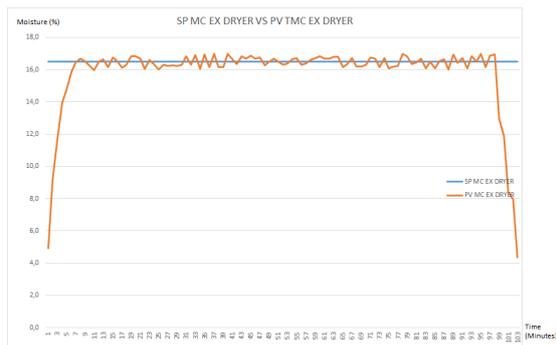
Proses Kedua

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,15}{16,62} * 100 = 0,0925$$

Proses Ketiga

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,1}{16,36} * 100 = 0,6112$$

Dari hasil perhitungan KV ketiga proses di atas terlihat bahwa *set point MC output Dryer* sudah tercapai dan nilai KV sudah memenuhi spesifikasi produk yang ditentukan. Perbandingan antara *Set Point MC output Dryer* dan nilai aktual sepanjang proses dapat dilihat pada Gambar 18.



Gambar 18 Grafik Perbandingan SP dan PV MC Output Dryer

Proses Flavour and Addback Post

Dari *database* proses kita ambil data akhir proses *flavour* untuk kita analisa apakah sudah sesuai dengan spesifikasi yang di targetkan. Data akhir proses Lini *Flavour and Addback Post* pada saat *commissioning* dapat dilihat pada Tabel 3

Tabel 3. Tabel Hasil Proses *Flavour and Addback Post*

NO PROCESS	SP RATIO ADDBACK 1	PV RATIO ADDBACK 1	AVG RATIO ADDBACK 1	SD RATIO ADDBACK 1
1	50,058	50,089	50,075	0,014
2	50,058	50,076	50,102	0,02
3	50,058	50,092	50,132	0,029
SP RATIO ADDBACK 3	PV RATIO ADDBACK 3	AVG RATIO ADDBACK 3	SD RATIO ADDBACK 3	
0,991	0,991	0,992	0,0001	
0,991	0,991	0,992	0,0001	
0,991	0,991	0,992	0,0002	
SP RATIO FLAVOUR	PV RATIO FLAVOUR	AVG RATIO FLAVOUR	SD RATIO FLAVOUR	
1,694	1,694	1,695	0,001	
1,694	1,693	1,695	0,001	
1,694	1,695	1,695	0,001	

Dari data akhir proses di atas dapat kita lihat bahwa semua *set point* yang ditargetkan dapat dicapai oleh mesin. Namun untuk melihat apakah pencapaian itu merata kita biasanya juga melihat dari nilai Koefisien Variasi (KV) nya.

• *Ratio Addback Post 1*

Spesifikasi target nilai KV maksimal = 0.2
Proses Pertama

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,014}{50,075} * 100 = 0,02795$$

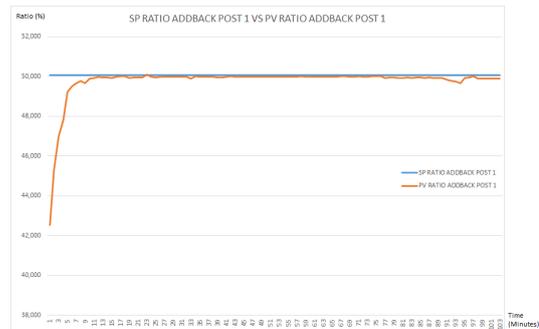
Proses Kedua

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,002}{50,102} * 100 = 0,0391$$

Proses Ketiga

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,029}{50,132} * 100 = 0,0578$$

Dari hasil KV ketiga proses di atas terlihat bahwa *set point ratio addback post 1* sudah tercapai dan nilai KV sudah memenuhi spesifikasi produk yang ditentukan. Perbandingan antara *set point ratio addback post 1* dan nilai aktual sepanjang proses dapat dilihat pada Gambar 19.



Gambar 19 Grafik Perbandingan SP dan PV Ratio Addback Post 1

- *Ratio Addback Post 3*
Spesifikasi target nilai KV maksimal = 0.2
Proses Pertama

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,0001}{0,992} * 100 = 0,01009$$

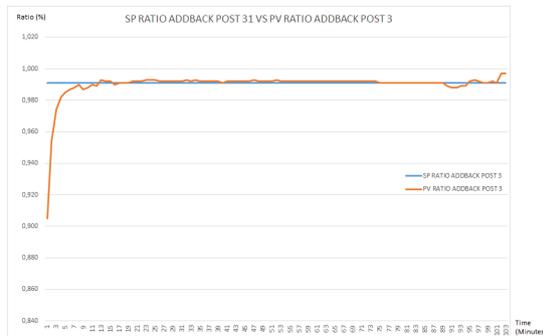
Proses Kedua

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,0001}{0,992} * 100 = 0,01009$$

Proses Ketiga

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,0002}{0,992} * 100 = 0,02018$$

Dari hasil KV ketiga proses di atas terlihat bahwa *set point ratio addback post 3* sudah tercapai dan nilai KV sudah memenuhi spesifikasi produk yang ditentukan. Perbandingan antara *Set Point ratio addback post 3* dan nilai aktual sepanjang proses dapat dilihat pada Gambar 20.



Gambar 20 Grafik Perbandingan SP dan PV Ratio Addback Post 3

- *Ratio flavour*
Spesifikasi target nilai KV maksimal = 0.2
Proses Pertama

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,001}{1,695} * 100 = 0,05899$$

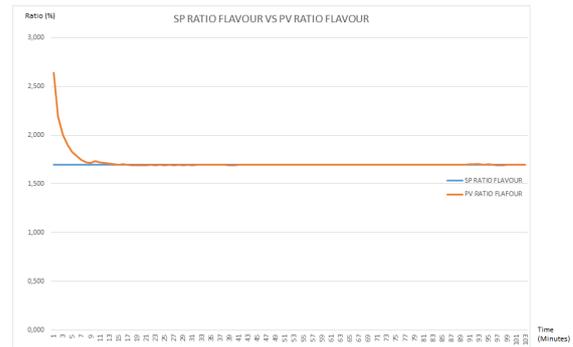
Proses Kedua

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,001}{1,695} * 100 = 0,0589$$

Proses Ketiga

$$KV = \frac{Sd}{Avg} * 100 = \frac{0,001}{1,695} * 100 = 0,05899$$

Dari hasil KV ketiga proses di atas terlihat bahwa *set point ratio flavour* sudah tercapai dan nilai KV sudah memenuhi spesifikasi produk yang ditentukan. Perbandingan antara *set point ratio flavour* dan nilai aktual sepanjang proses dapat dilihat pada Gambar 21.



Gambar 21 Grafik Perbandingan SP dan PV Ratio Flavour

Dari hasil grafik diatas terlihat bahwa spesifikasi yang ditargetkan dapat dicapai oleh sistem yang sudah di desain.

Dari hasil analisis hasil diatas, dapat kita lihat bahwa program mesin yang dirancang sudah bisa memenuhi spesifikasi produk yang ditargetkan oleh bagian produksi, baik itu target spesifikasi MC, rasio, suhu dan nilai KV dari masing masing spesifikasi produk. Langkah berikutnya adalah uji rasa dan uji homogenitas di laboratorium sebelum mesin bisa digunakan untuk produksi secara reguler oleh bagian produksi.

KESIMPULAN

Dari penelitian ini didapatkan beberapa kesimpulan mengenai perancangan sistem kendali otomatis di Lini *Blending 3 Ton OASIS* sebagai berikut.

1. Di dalam perancangan Lini *Blending* dibagi menjadi empat Line, yaitu: Lini *Conditioning and Casing*, Lini *Dryer and Addback Prior*, Line *Flavour and Addback Post* dan Lini *Tipper*.
2. Masing-masing lini dikontrol oleh PLC Siemens, dan antar panel terhubung dengan topologi *ring* dalam *platform* komunikasi *Industrial Ethernet*.
3. Untuk komunikasi dari *main* panel ke *slave* terhubung dalam topologi *bus* dalam *platform* komunikasi *Profibus*.
4. Lini dikendalikan dari ruang kendali pusat dengan SCADA, semua data dari lini dapat terlihat dari SCADA.
5. Data akhir proses, data *logging* proses dan *alarm logging* disimpan ke *database* oleh SCADA.
6. Setelah *commissioning*, dari data proses didapatkan kesimpulan bahwa semua spesifikasi produk dapat dipenuhi oleh mesin.

DAFTAR PUSTAKA

Anonim. (2020). Dasar Sistem Kendali. Universitas Gadjah Mada

R Copeland, Brian. (2008). The Design of PID Controllers Using Ziegler Nichols Tuning

He, Jenkins. (2011). Tunning PID Controllers.

Simenteknindo.com. "Industrial Ethernet" (diakses 05 April 2020)

Simentari Abdibina.S7-300 PLC Training Basic Level.2014

Help on Simatic Manager V5.5 SP4

Help on Wincc Explorer V7.2

Primary Engineering PT Djarum.*Basic of Statistical for Process Control* .2009