

Aplikasi untuk Memonitor PLC Pada Mesin Filling dan Capping

Gregorius Andry Laksmana, Petrus Santoso, Felix Pasila
Program Studi Teknik Elektro, Universitas Kristen Petra
Jl. Siwalankerto 121-131, Surabaya 60236, Indonesia

E-Mail: G_andry93@yahoo.ac.id ; Petrus@peter.petra.ac.id ; Felix@peter.petra.ac.id

Abstrak— Seiring berkembangnya teknologi industri, suatu sistem otomasi industri membutuhkan teknologi PLC (*Programmable Logic Controller*) untuk memenuhi tuntutan adanya fleksibilitas. Kebutuhan untuk pengawasan kendali proses dan mesin oleh PLC dan pengolahan data selama proses berlangsung dan dapat dikontrol jarak jauh melalui internet, membawa lahirnya konsep HMI dan aplikasi yang mendukung adalah visual studio C#.

Paper ini dirancang untuk mengatur sistem pengisian cairan ke dalam botol kemasan dan dijalankan melalui aplikasi C#. Algoritma kontrol yang digunakan pada mesin ini adalah *on/off*. Semua penerapan ini dilakukan menggunakan PLC Siemens S7-1200. Semua sistem dapat dikontrol secara otomatis melalui PC dengan menggunakan TIA Portal v12 sebagai HMI *software*. Dan kontrol jarak jauh melalui laptop menggunakan visual studio C# sebagai aplikasi. Dalam aplikasi C# yang terkoneksi ke PLC menggunakan library Snap7.

Pengujian yang dilakukan dengan membandingkan jalannya *plant* dengan tampilan/HMI pada komputer serta koneksi PLC Siemens S7-1200 dengan aplikasi dari visual studio C#. Mesin ini memiliki kapasitas yang dapat diatur sesuai produksi botol yang diminta dengan rata-rata botolnya berisi 8,4 ml. Aplikasi C# dapat mengontrol PLC S7-1200 yang dihubungkan pada *plant*.

Kata Kunci— TIA Portal v12, PLC (*Programmable Ladder Controller*), Visual Studio (C#), *Filling* dan *Capping Machine*.

I. PENDAHULUAN

PADA zaman yang semakin berkembang ini, Aplikasi *desktop* merupakan salah satu hal yang memiliki potensi untuk berkembang dengan sangat pesat. Hal ini dikarenakan laptop memiliki peran yang sangat krusial yaitu di bidang informasi dan entertainment. Di lain sisi, penggunaan aplikasi untuk mendukung hobi maupun dunia kerja masih sangat minim dikembangkan. Hal ini sangat memungkinkan untuk dimanfaatkan karena aplikasi *desktop* dapat berkontribusi untuk menunjang pekerjaan di bidang *engineering*. Adanya kebutuhan yang cepat dan efisien dalam menangani PLC maka dari itu digunakan secara aplikasi *desktop* untuk mengatur proses PLC tersebut.

Di tengah pengembangan aplikasi *desktop* untuk mendukung pekerjaan *engineering*, terdapat beberapa aplikasi *desktop* di bidang teknik elektro yaitu “Visual Studio”, “Visual Basic”, “Delphi”, “Java”, dan beberapa aplikasi *desktop* lainnya[5]. Dari beberapa aplikasi *desktop* yang ada, pengembangan yang dilakukan masih sedikit. Maka dari itu banyak *engineering* masih belum mengenal

atau belum paham penggunaan dengan aplikasi *desktop* tersebut.

Beberapa aplikasi *desktop* yang telah dikembangkan seperti Visual Basic. Salah satunya dari aplikasi Visual Basic ini hanya menampilkan koneksi dan informasi data yang berupa input dan output dari PLC. Maka dari itu dalam paper ini akan dikembangkan lagi dengan mengontrol PLC.

Software-software pendukung untuk desain proses PLC meliputi “WinCC”, “STEP7”, “HMI”, “C#” dan lain sebagainya. Dari *software* tersebut dapat mendukung *engineering* untuk memonitor dengan mudah[1,2,6,7]. Aplikasi *desktop* yang didesain ini menggunakan bahasa pemrograman C#, dari aplikasi *desktop* ini dapat dikoneksikan dan memonitor PLC dengan lebih mudah digunakan tanpa membuka program seperti Wonderware, TIA Portal, dan lain-lainnya[7].

Dengan adanya kebutuhan *monitoring* proses PLC dengan cepat, efisien dan aplikasi di lapangan, maka perlu dilakukan pengembangan aplikasi *desktop* di bidang *automation industry* terutama dalam desain proses PLC yang *compact* sehingga *user* dapat menggunakan aplikasi tersebut dengan mudah, dan efisien.

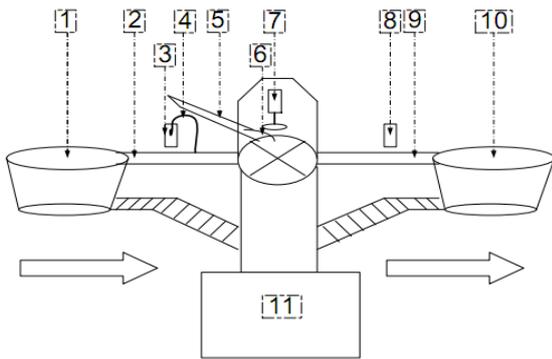
II. PERANCANGAN SISTEM

A. Desain Plant

Secara umum, mesin *filling* dan *capping* otomatis ini adalah suatu mesin yang dikontrol secara sistem SCADA yang berfungsi untuk mengisi cairan kedalam sebuah botol kemasan dan kemudian akan melewati proses penutupan tutup botol tersebut. Volume cairan yang diisikan dapat diatur dengan caramengatur letak sensor *read switch* yang berada pada *body* silinder *pneumatic* yang menjadi aktuator dari piston pengisi cairan. Mesin *filling* dan *capping* otomatis ini dikontrol dengan menggunakan metode *on-off*. Kontroler yang digunakan mesin *filling* dan *capping* otomatis ini menggunakan sebuah PLC Siemens S7-1200[1]. Mesin ini juga menggunakan sebuah mikrokontroler AT Mega 8 yang digunakan untuk mengontrol kecepatan motor.

Cara kerja dari sistem ini adalah menggunakan sensor posisi pada titik pengisian yang akan memberikan input pada PLC Siemens S7-1200 pada saat botol telah berada pada titik pengisian. Pada saat botol berada pada titik pengisian, konveyor akan berhenti dan botol diisi cairan. Botol yang telah terisikemudian akan ditutup dan dihitung jumlah produksinya pada setiap harinya.

Berikut pada gambar 1 merupakan sketsa dari mesin *filling* dan *capping* otomatis.



Gambar. 1. Sketsa mesin *filling* dan *capping* otomatis

Keterangan gambar sketsa mesin *filling* dan *capping* otomatis:

- *Round table in*
- Konveyor *in*
- Photo sensor untuk proses *filling*
- Pipa pengisian cairan
- Jalur antrian tutup botol
- *Limit switch* untuk proses *capping*
- Silinder *pneumatic* untuk proses *capping*
- Photo sensor untuk mengetahui jumlah produksi
- Konveyor *out*
- *Round table out*
- Panel

B. Desain Hardware

Perancangan hardware meliputi perancangan jalur konveyor *in* ke *round table middle*, mengubah sensitifitas sensor posisi, penggantian PLC (*Programmable Logic Control*), dan penambahan *photosensor* pada bagian konveyor *out*, yang berfungsi untuk mengakuisisi data yaitu menghitung jumlah produksi dan mematikan *plant* pada saat hasil produksi terpenuhi.

Pada paper ini, mekanik yang digunakan yaitu melanjutkan mekanik tugas akhir dari Yohanes Stevie Budion dengan judul “Mesin *filling* dan *capping* otomatis untuk botol kemasan essens” dengan no Tugas akhir 02010910/ELK/2010. Proses yang dilakukan selanjutnya adalah perancangan jalur konveyor *in* ke *round table middle*. Pada perancangan mekanik yang baru ini, proses botol di konveyor *in* dipindahkan ketika botol berada di *round table middle*, sebelum botol masuk ke proses *capping*. Hal ini dilakukan sebab pada saat botol berada di konveyor *in* botol seolah-olah ditahan di *round table middle* sehingga botol mudah terguling. Pada gambar 2 merupakan sistem mekanis perpindahan botol dari konveyor *in* ke *round table middle*.



Gambar. 2. Sistem mekanis perpindahan dari *conveyor in* ke *round table middle*

Pada mesin *filling* dan *capping* otomatis ini dilengkapi dengan sensor posisi yang berfungsi untuk mengetahui posisi botol kemasan yang akan diisi cairan, bila posisi botol telah mencapai titik *filling*, maka sensor akan memberikan sinyal pada PLC Siemens S7-1200 untuk melakukan proses pengisian air.

Pada desain sistem, Sensor posisi ini menggunakan produk dari perusahaan ANLY dengan tipe *photosensor* PS-10DP. Dari tipe sensor tersebut dapat dilihat karakteristik dan jenis dari sensor tersebut.

P: *Photo type*

S: *Screw type*

10: Jarak maksimum sensor untuk mendeteksi adalah 10 cm

D: *Diffuse Reflective*

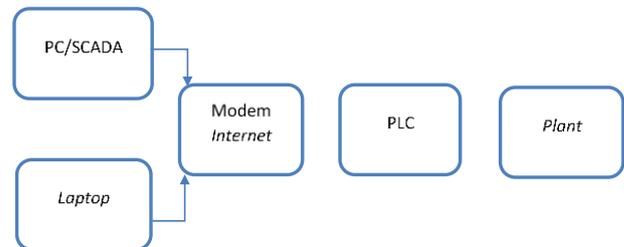
P: *PNP DC mode*



Gambar. 3. *Photosensor* PS-10D-P

C. Desain Sistem

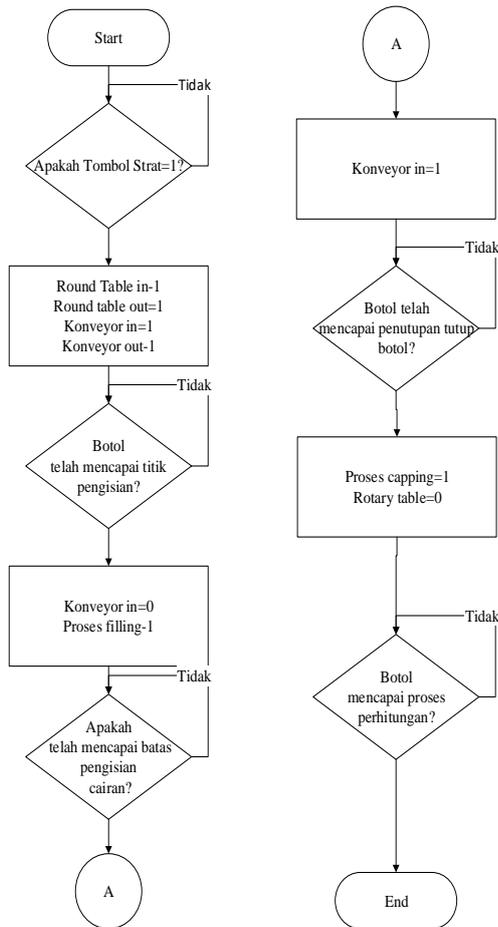
Sistem yang didesain adalah sistem *plant filling* dan *capping machine* yang di kontrol oleh aplikasi desktop dimana sistem ini dari botol kosong diisi melalui sistem *filling* dan setelah itu botol akan di *capping* melalui sistem *capping*. *Plant* ini disebut *packaging*. Sistem ini menggunakan PLC sebagai kontrol utama pada sistem. Program pada PLC akan mengelolah data input dari sensor dan output dari motor/*drives* yang akan diaktifkan. Berikut ini merupakan blok diagram dari sistem *filling* dan *capping machine*.



Gambar. 4. Blok diagram sistem

D. Desain Software TIA Portal V12

Alur pemrograman pada PLC untuk mesin *filling* dan *capping* otomatis dapat dilihat pada flow chart berikut:



Gambar. 4. Flow Chart pemrograman PLC

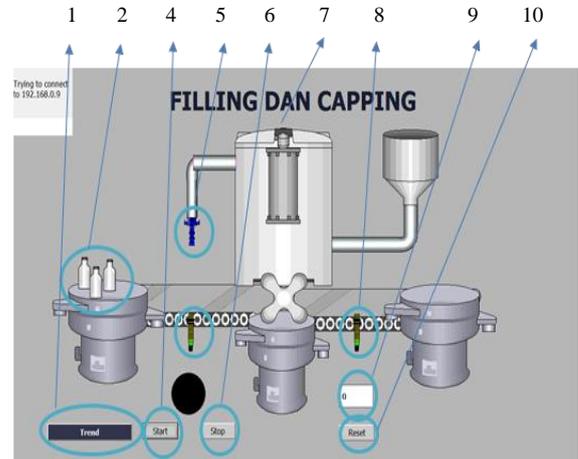
Proses *flow chart* di atas yang dimulai dengan penekanan tombol *start* kemudian proses yang dilakukan akan berlangsung hingga selesai. Botol yang masih kosong diletakkan pada *round table input* secara acak, setelah itu *round table input* botol akan berputar dan secara tidak langsung akan tertata satu per satu kemudian masuk pada *konveyor in*.

Setelah ada pada jalur *konveyor in*, botol yang dibawa oleh *konveyor in* ini akan berhenti pada titik pengisian. Hal itu terjadi ketika PLC Siemens mendapat inputan dari sensor posisi. Setelah itu *solenoid valve* piston akan aktif dan berfungsi untuk mendorong air keluar dari piston menuju botol kemasan yang telah berada pada titik pengisian. *Solenoid valve* akan tetap aktif untuk menyuplai udara *pneumatic* sampai *reed switch* pada bagian depan bodi *pneumatic* aktif, setelah itu kondisi *solenoid valve* akan mati, dan *pneumatic* mundur kembali pada posisi awal. Disaat yang bersamaan mundurnya piston menyebabkan air yang ada pada tampungan tersedot menuju piston. Setelah piston terisi air maka piston siap untuk mengisi botol selanjutnya.

Ketika posisi botol telah terisi, maka *konveyor* akan aktif dan botol akan bergerak menuju pada *round table middle*. Botol tersebut secara otomatis akan mendapat tutup botol ketika lewat di bawah filter tutup botol. Botol yang telah mendapat tutup akan masuk ke proses *capping* sehingga tutup botol akan terpasang dengan sempurna. Botol yang masuk ke *konveyor out* akan secara otomatis dihitung

melalui *photo sensor* yang terhubung dengan input PLC Siemens.

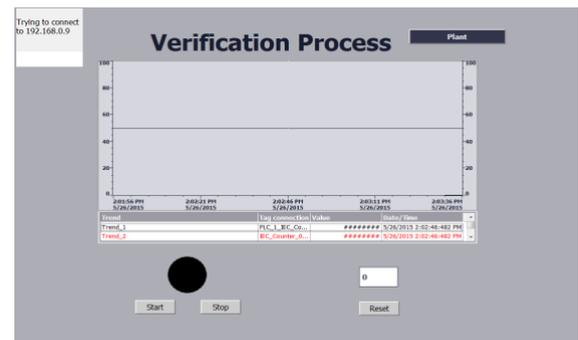
Desain *layout* diperlukan untuk menata tampilan main screen pada WinCC Runtime Advanced[1,4]. Pada desain *layout* WinCC Runtime Advanced untuk sistem SCADA mesin *filling* dan *capping* otomatis memiliki 2 main screen, yaitu halaman kontrol *Filling* dan *Capping* otomatis dan halaman *Trend*[1,4,6,7]. Pada gambar 5 dan gambar 6 merupakan desain *layout* pada WinCC Runtime Advanced.



Gambar. 5. Desain layout main screen *filling* dan *capping*

Keterangan gambar desain main screen *filling* dan *capping* di WinCC RT Advanced:

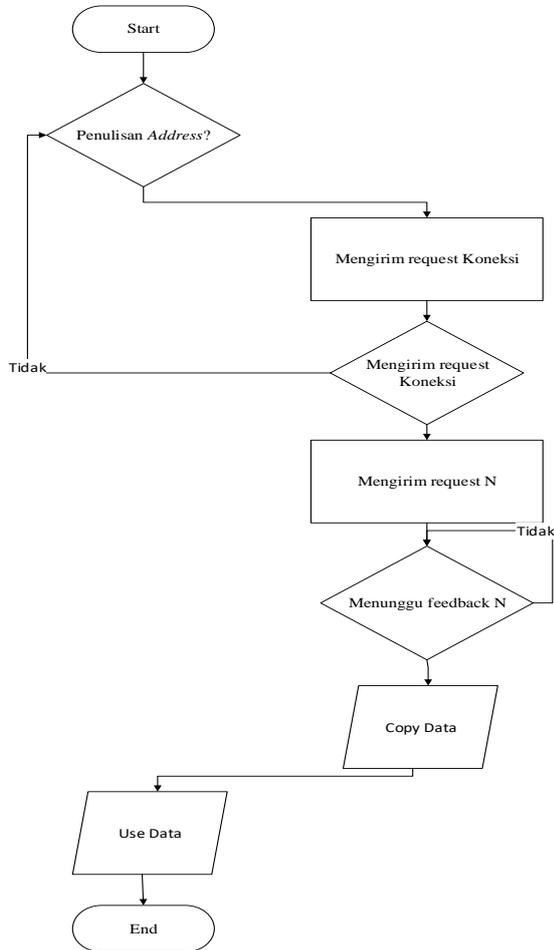
1. Halaman *Trend* berfungsi untuk mengganti halaman/main screen *filling* dan *capping* menjadi halaman *trend*.
2. Indikasi botol berfungsi untuk menampilkan botol pada awal animasi/visualisasi di WinCC RT Advanced.
3. Tombol *start* berfungsi untuk mengaktifkan proses *filling* dan *capping* pada saat *run* program WinCC RT Advanced.
4. *Filling* berfungsi untuk indikator pada saat botol diisi air/proses *filling*.
5. Tombol *stop* berfungsi untuk *off* proses *filling* dan *capping* ada saat *run* program WinCC RT Advanced.
6. *Capping* berfungsi untuk indikator pada saat botol dalam proses *capping*.
7. Sensor hasil produksi berfungsi untuk mendeteksi botol saat telah melalui proses *capping*.
8. Sensor *filling* berfungsi untuk mendeteksi botol pada saat *start on*.
9. Indikator hasil produksi berfungsi untuk mengeluarkan indikator hasil produksi dengan angka.
10. Tombol *reset* berfungsi untuk mengulang proses *filling* dan *capping*.



Gambar. 6. Desain layout main screen *Trend*

E. Desain Aplikasi PLC S7-1200 dengan C# (Visual Studio)

Alur koneksi pada C#(visual studio) ke PLC untuk mesin *filling* dan *capping* otomatis dapat dilihat pada *flow chart* berikut[5]:



Gambar. 7. Flowchart koneksi PLC ke C# (Visual Studio)

Proses *flow chart* pada gambar 7 dimulai dengan pada aplikasi C#, dimana aplikasi C# yang dikoneksikan ke PLC membutuhkan IP address yang terdapat di TIA Portal v12 terutama IP dari PLC. IP address yang akan ditulis itu masih didalam *user program* atau dapat dikatakan di aplikasi C#. Setelah itu pengiriman *request* dari aplikasi C# melalui *library* Siemens PLC menggunakan Snap7. C# mengirim *request* lalu menunggu *feedback* untuk koneksi. Dari dalam PLC mengirim *request* untuk koneksi dengan PLC dan menunggu *feedback* untuk mengambil data block. Setelah itu PLC *copy* data ke aplikasi C#, sekarang C# dapat terkoneksi dengan PLC dan dapat menggunakan atau memakai data yang terdapat dari PLC.

Pada desain dari koneksi aplikasi C# ke PLC menggunakan komputer, gambar 8 adalah tampilan dari *main windows* aplikasi C#[5].

Keterangan gambar desain main window aplikasi C#:

1. IP Address berfungsi untuk menuliskan IP yang ingin di koneksi dengan aplikasi C#.
2. Tombol *connect* berfungsi untuk meminta koneksi ke PLC.
3. Tombol *disconnect* berfungsi untuk memutuskan koneksi ke PLC.
4. Tombol *start* berfungsi untuk menjalankan data yang terdapat di PLC.
5. Indikator berfungsi untuk indikasi *start* pada saat *on* atau *stop* pada saat *off*.
6. Tombol *reset* berfungsi untuk me-reset indikator *on* dari *filling*, *capping* dan hasil produksi.
7. Tombol *stop* berfungsi untuk tidak mematikan data yang terdapat di PLC.

Tombol Data Block berfungsi untuk menjalankan *filling*, *capping* dan hasil produksi secara manual pada PLC.

III. PENGUJIAN SISTEM

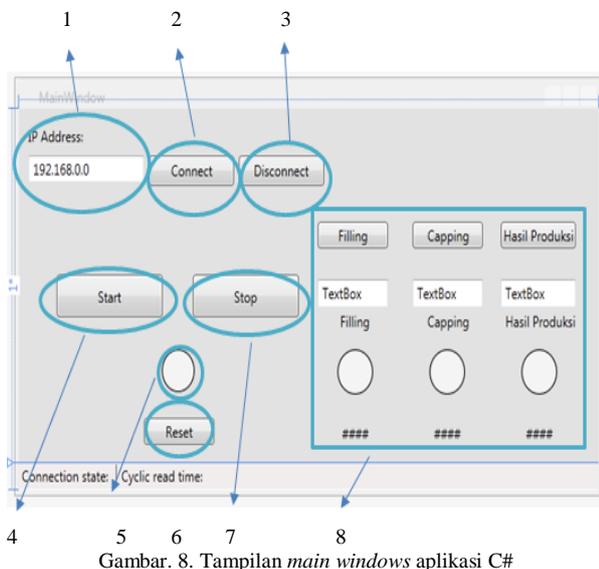
A. Pengujian Volume Piston Dengan Menggunakan Air

Pengujian yang dilakukan berguna untuk mengetahui ketepatan *volume* cairan diisikan ke botol dengan menggunakan *piston*. Cairan yang digunakan dalam pengujian ini adalah air. *Volume* cairan minuman yang dapat disedot oleh *piston* ini adalah 5 ml dan *volume* maksimumnya adalah 43 ml.

Pada saat dilakukan pengujian dengan jumlah volume yang diambil secara acak didapatkan data sebagai berikut:

Percobaan ke-	Volume air
1	8 ml
2	7,8 ml
3	8 ml
4	8,1 ml
5	8,2 ml
6	8,9 ml
7	8,9 ml
8	8,9 ml
9	8,8 ml
10	8,4 ml
Volume rata-rata	8.4 ml

Gambar. 9. Tabel Pengujian jumlah Volume air.



Gambar. 8. Tampilan main windows aplikasi C#

Dalam pengujian ini metode yang digunakan adalah dengan cara menuangkan air dari hasil botol yang diisi air dalam proses *filling* ke dalam tabung suntik. Dari tabung suntik tersebut dapat dilihat berapa ml setiap hasil botol yang didapatkan. Dan bisa dilihat dari data gambar 9 dapat diasumsikan dengan tidak ada kebocoran pada jalur cairan. Asumsi yang lain adalah tidak adanya udara yang masuk pada *piston* yang berfungsi untuk menyedot cairan dari tampungan menuju botol kemasan yang telah dipersiapkan.

B. Pengujian TIA Portal V12 WinCC RT Advanced

Proses pengambilan data *delay* yang dilakukan ketika mesin sedang melakukan proses *filling* dan *capping*. Berikut ini beberapa pengujian untuk melihat perbedaan waktu pada saat aktif dari tampilan *software* WinCC RT Advanced dengan *plant*:

- *User* mengaktifkan *plant* dengan *start*. Maka pada proses pengujian di WinCC RT Advanced dengan *plant* yang dijalankan bersamaan dapat dilihat ketika terjadi proses di *filling* adalah:

Percobaan ke-	WinCC RT Advanced	<i>Plant</i>	<i>Error</i>
1	3,82 s	3,82 s	0 s
2	4,68 s	3,82 s	0,84 s
3	4,19 s	3,64 s	0,55 s
4	4,08 s	3,50 s	0,42 s
5	3,64 s	3,09 s	0,55 s
6	3,79 s	3,37 s	0,42 s
7	3,26 s	3,48 s	0,22 s
8	3,69 s	3,69 s	0 s
9	3,28 s	3,19 s	0,09 s
10	3,36 s	3,08 s	0,28 s
		RMS _e	0,1341 s

Gambar. 10. Tabel Pengujian *delay* Visualisasi dengan *Real (Plant)*

Pada gambar 10 dapat dilihat bahwa input dan output yang ditampilkan sesuai pada dengan keadaan *plant* yaitu pada saat proses *filling* sedang berlangsung. Pada proses ini terdapat perbedaan waktu antara visualisasi dengan *plant*. Perbedaan ini dapat dilihat pada RMS_e dari pengujian pada gambar 10 terlihat bahwa terdapat *error* dengan perbedaan waktu 0,1341 s. hasil ini didapatkan dengan rumus RMS_e. Maka dapat disimpulkan bahwa pada visualisasi terdapat *delay* pada saat menjalankan proses *filling*.

- *User* mengaktifkan *plant* dengan *start*. Maka pada proses pengujian di WinCC RT Advanced dengan *plant* yang dijalankan bersamaan dapat dilihat ketika terjadi proses *capping* adalah:

Percobaan ke-	WinCC RT Advanced	<i>Plant</i>	<i>Error</i>
1	8,56 s	8,33 s	0,23 s
2	8,55 s	8,11 s	0,44 s
3	8,15 s	8,78 s	0,63 s
4	8,22 s	8,74 s	0,52 s
5	8,58 s	8,58 s	0 s
6	7,88 s	7,54 s	0,34 s
7	8,07 s	9,39 s	0,32 s
8	8,51 s	9 s	0,49 s
9	8,37 s	8,69 s	0,32 s
10	8,64 s	9,21 s	0,37 s
		RMS _e	0,1269 s

Gambar. 11. Tabel Pengujian *delay* Visualisasi dengan *Real (Plant)*

Percobaan ke-	WinCC RT Advanced	<i>Plant</i>	<i>Error</i>
1	16,04 s	17,4 s	1,36 s
2	15,65 s	15,01 s	0,64 s
3	15,37 s	15,37 s	0 s
4	15,9 s	15,7 s	0,2 s
5	15,88 s	15,7 s	0,18 s
6	14,57 s	14,76 s	0,19 s
7	15,43 s	15,31 s	0,12 s
8	14,51 s	14,23 s	0,28 s
9	15,38 s	15,2 s	0,18 s
10	15,23 s	15,36 s	0,13 s
		RMS _e	0,1584 s

Gambar. 12. Tabel Pengujian *delay* Visualisasi dengan *Real (Plant)*

Pada gambar 11 dapat dilihat bahwa input dan output yang ditampilkan sesuai pada dengan keadaan *plant* yaitu pada saat proses *capping* sedang berlangsung. Pada proses ini terdapat perbedaan waktu antara visualisasi dengan *plant*. Perbedaan ini dapat dilihat pada RMS_e dari pengujian pada gambar 11 terlihat bahwa terdapat *error* dengan perbedaan waktu 0,1269 s. hasil ini didapatkan dengan rumus RMS_e. Hal ini dikarenakan *plant* pada saat proses *capping* botol tersangkut/menunggu lubang dari *round table middle* mencapai posisi pengambilan botol. Maka dapat disimpulkan bahwa visualisasi tidak terdapat *delay* pada saat menjalankan proses *capping*.

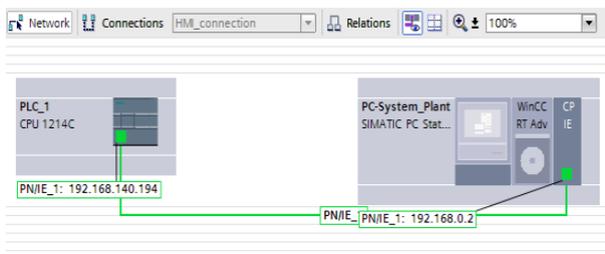
- *User* mengaktifkan *plant* dengan *start*. Maka pada proses pengujian di WinCC RT Advanced dengan *plant*

yang dijalankan bersamaan dapat dilihat ketika terjadi proses hasil produksi adalah:

Pada gambar 12 dapat dilihat bahwa input dan output yang ditampilkan sesuai pada dengan keadaan *plant* yaitu pada saat proses hasil produksi sedang berlangsung. Pada proses ini terdapat perbedaan waktu antara visualisasi dengan *plant*. Perbedaan ini dapat dilihat pada RMSe dari pengujian pada gambar 12 terlihat bahwa terdapat *error* dengan perbedaan waktu 0,1584 s. hasil ini didapatkan dengan rumus RMSe. Hal ini dikarenakan *plant* setelah proses *capping* botol tidak terhambat/tidak tersangkut lubang dari *round table middle*. Maka dapat disimpulkan bahwa visualisasi terdapat *delay* pada saat menjalankan proses hasil produksi.

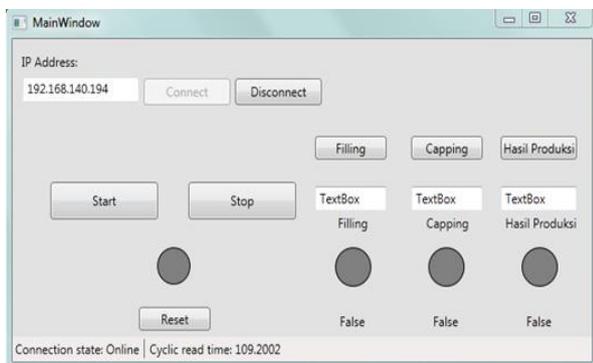
C. Pengujian Aplikasi PLC S7-1200 Dengan C#(Visual Studio)

Pada desain software ini menggunakan jaringan *local area network* (LAN), PLC akan disambungkan via LAN pada modem Wifi Petra. Pertama-tama IP PLC akan diatur terlebih dahulu. Cara mengatur IP PLC melalui TIA Portal v12. Dari *default* PLC (192.168.0.0) akan diatur menjadi 192.168.140.194/21. Terlihat pada gambar 13.



Gambar. 13. Tampilan IP PLC pada TIA Portal

Pengujian yang dilihat jalannya aplikasi pada C# yang telah terkoneksi dengan PLC S7-1200 melalui wifi Lab Robotika. ketika tombol *connect on* dan koneksinya akan terputus ketika tombol *disconnect on*. Aplikasi C# akan mengirim data ketika tombol *connect* ditekan dan tombol *disconnect* ditekan. Hal ini diperlukan untuk melihat koneksi aplikasi C# dengan PLC S7-1200 yang sedang beroperasi. Berikut tampilan menu aplikasi C# *connection*.



Gambar. 14. Tampilan aplikasi C# *connection online*

Pada gambar 14 *connection* aplikasi C# berhasil koneksi. Jika koneksi tersambung maka pada *connection state* akan keluar tulisan *online*. Itu berarti koneksi dari aplikasi C# dapat mengirim dan menerima data dari PLC S7-1200. Hal ini disebabkan IP *address* yang dimasukkan sesuai dengan

IP pada PLC S7-1200. Tetapi ada *error* yang terjadi seperti pada saat tombol *connect* ditekan koneksi pada aplikasi C# tidak langsung terkoneksi dan kadang-kadang *not responding* ini disebabkan koneksi wifi yang tersambung pada laptop tidak maksimal jadi menyebabkan hal tersebut.



Gambar. 15. Tampilan aplikasi C# pada saat *Start*

Terlihat pada gambar 15 dari aplikasi C# pada saat ditekan *start*. Yang terjadi pada aplikasi ini *led* yang terdapat di bawah tombol *start* berwarna hijau yang berarti *plant on* atau *plant* mulai berjalan. Pada gambar 15 dari aplikasi C# pada saat ditekan *stop*. Yang terjadi pada aplikasi ini *led* yang terdapat di bawah tombol *start* dan *stop* berwarna merah yang berarti *plant off* atau *plant* mulai berhenti. Itu berarti koneksi pengiriman data dari aplikasi C# ke PLC S7-1200 berhasil.

IV. KESIMPULAN

Kesimpulan yang dapat diambil dari hasil pengerjaan paper ini adalah sebagai berikut:

- Rata-rata air pada botol yang dihasilkan adalah 8,4 ml.
- Delay yang terjadi antara *plant* dan visualisasi pada proses *filling* 0,1341 s, pada proses *capping* 0,1269 s dan pada proses hasil produksi 0,1584 s.
- Bahwa aplikasi pada C# dapat digunakan untuk koneksi dan mengontrol PLC dengan menggunakan *library* Snap7.

DAFTAR PUSTAKA

- [1] Advanced, W. et al., 2012. Engineering for all HMI-systems WinCC comfort TIA AS S SUP FA SIMATIC WinCC V12 WinCC Advanced WinCC Professional. , pp.1-40.
- [2] Advanced, W.R. et al., 2011. HMI software in the TIA Portal.
- [3] Ag, S., 2013. WinCC advanced v12.0. , pp.1-4050.
- [4] Hmi, S., 2013. Premium add-ons for SIMATIC WinCC training for industry.
- [5] Nardella, D., 2015. Snap7 reference manual. , pp.1-286.
- [6] Panels, C. & Advanced, R., 2013. Comfort panels , runtime advanced - advanced configuration options WinCC V12 comfort panels , runtime advanced - advanced configuration options. , pp.1-84.
- [7] Portal, I.A., 2011. SIMATIC WinCC in the Totally Integrated Automation Portal SIMATIC WinCC in the Totally Integrated Automation Portal. , (April).