

## Design and Build a Telegram – Based Infusion Droplet Control and Monitoring System

### Rancang Bangun Sistem Kontrol dan Monitoring Tetesan Infus Berbasis Telegram

Aloysius Gonzaga Kristianto Juje, Adjie Ramadhan, Syahri Muharom  
{aloygonzaga13@gmail.com, ramadhan@gmail.com, syahrimuharom@itats.ac.id}

Institut Teknologi Adhi Tama Surabaya, Jl. Arief Rachman Hakim no 100, 031-5997244

**Abstract.** *Infusion is an important part of the process of care and healing of patients who are in the hospital, however, the errors often occur in setting the drip or replacing the infusion that has run out, this can cause the patient to die. From these problems, the researcher has an idea by creating a system that can monitor and control the infusion, so that later it can give warnings to nurses based on Telegram. Several sensors are used for the designed system, including load cell, load cell sends data on the weight of infusion fluids to Arduino, then Arduino sends weight information to Telegram to be monitored, infusion drip control system is carried out using servo and optocoupler sensors. From the tests carried out, the servo and optocoupler systems can adjust the infusion drops according to the patient's needs based on orders from the telegram, with an average error of 1.3%, and the system can provide information to the telegram to identify that the infusion fluid is almost empty. From this study, it is expected that the system can assist nurses in monitoring the availability of infusion fluids in patients and can control infusion drops based on patient needs.*

**Keywords** – Arduino; Control; Infusion; Monitoring; Telegram

**Abstrak.** *Infus adalah bagian penting dari proses perawatan dan penyembuhan pasien yang berada di rumah sakit, tetapi sering juga terjadi kesalahan dalam pengaturan tetesan ataupun penggantian infus yang sudah habis, kelalaian ini dapat menimbulkan pasien meninggal dunia. Dari permasalahan tersebut, peneliti mempunyai sebuah terobosan dengan membuat sebuah sistem yang dapat mengatur tetesan infus, dan memberikan peringatan ke perawat dikala infus hampir habis dengan menggunakan telegram. Beberapa sensor digunakan untuk sistem yang dirancang, diantaranya loadcell, loadcell mengirimkan data berat cairan infus ke arduino, selanjutnya arduino mengirim informasi berat ke telegram untuk dimonitor, sistem pengatur tetesan infus dilakukan dengan menggunakan servo dan sensor optocoupler. Dari pengujian yang dilakukan, sistem servo dan optocoupler dapat mengatur tetesan infus sesuai dengan kebutuhan pasien berdasarkan perintah dari telegram, dengan rata-rata error sebesar 1,3%, dan sistem dapat memberikan informasi ke telegram untuk mengidentifikasi bahwa cairan infus sudah hampir habis. Dari penelitian ini diharapkan sistem dapat membantu perawat dalam memonitor ketersediaan cairan infus pada pasien dan dapat mengontrol tetesan infus berdasarkan kebutuhan pasien.*

**Kata Kunci** – Arduino; Infus; Kontrol; Monitoring; Telegram

## I. PENDAHULUAN

Potensi kesalahan pemberian infus yang ada di Indonesia semakin meningkat, mengingatnya pemberian infus sebagian besar dilakukan secara manual atau metode gravitasi. Beberapa kesalahan - kesalahan berdasarkan pengalaman *empiris* di Rumah Sakit selain faktor kesalahan dalam pengaturan tetesan infus juga dapat disebabkan oleh:

1. Pasien/Keluarga pasien, kesalahan tersebut biasanya dilakukan oleh keluarga pasien dalam mengatur sendiri laju tetesan infus,
2. Peralatan, ketinggian tiang infus dapat mempengaruhi laju tetesan infus, termasuk juga *viskositas* cairan,
3. Vena, spasme vena dapat menurunkan laju tetesan infus.

Hal ini dikarenakan oleh infus yang bersuhu rendah, atau disebabkan oleh infus yang bersifat mengiritasi. Metode-metode pemasangan infus pada dunia medis terbagi menjadi 3, metode gravitasi, metode elektronik dan metode kateter. Secara keseluruhan dari ketiga metode tersebut metode gravitasi memiliki keunggulan tingkat efisiensi yang tinggi dan pemasangan yang cukup cepat serta tidak terikat pada sumber daya listrik [1].

Kegiatan yang paling sering dilakukan oleh perawat adalah pemasangan infus pada pasien. Sebuah kejadian bayi yang meninggal dibangka belitung pada usia 4 hari terduga diakibatkan kelalaian perawat dalam mengganti infus yang

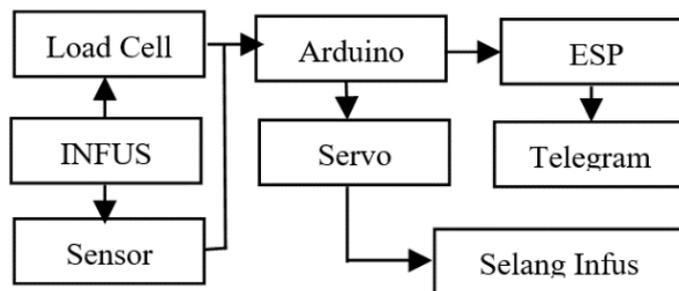
sudah habis, hal tersebut diperkuat dengan temuan botol infus yang telah kosong masih melekat pada tubuh si bayi [2]. Sebanyak 90% pasien di Rumah Sakit menerima berbagai pengobatan melalui *infuse* [3]. Saat ini penggunaan infus di rumah sakit masih secara manual dimana kesalahan - kesalahan seperti pemberian obat, pengaturan tetesan infus masih sering terjadi. Sebuah unit perawatan intensif baru-baru ini menemukan 47% efek samping adalah karena pengobatan dan dosis yang salah termasuk dalam mengatur tetesan infus [4]. Pada saat keterlambatan penggantian botol infus yang sudah kosong bisa menyebabkan masuknya sejumlah besar udara kedalam pembuluh darah dan dapat menimbulkan kematian [5].

Sebuah sistem kontrol dan sistem monitoring diperlukan untuk mengatasi permasalahan mengenai infus, dimana sistem kontrol sering digunakan dalam bidang otomatisasi dan lain-lain [6-9]. Sedangkan monitoring juga sering digunakan dalam banyak bidang [10], [11]. Dari permasalahan yang telah dikemukakan maka terdapat rumusan masalah, dimana bagaimana membuat sebuah sistem yang dapat mengontrol dan memonitoring tetesan infus pasien, berdasarkan diagram kebutuhan pasien. Dari itu peneliti membuat sebuah terobosan dengan membuat sebuah sistem yang dapat mengontrol dan memonitoring tetesan infus berdasarkan diagram kebutuhan pasien menggunakan Telegram.

## II. METODE

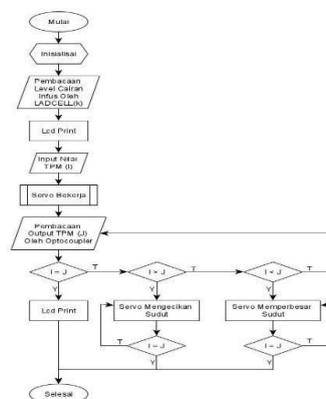
### A. Blok diagram dan flowchart sistem

Pada bagian ini menjelaskan perancangan sistem yang berupa blok diagram. Pada gambar 1 terdapat perancangan sistem yang dikerjakan.



Gambar 1. Blok diagram sistem

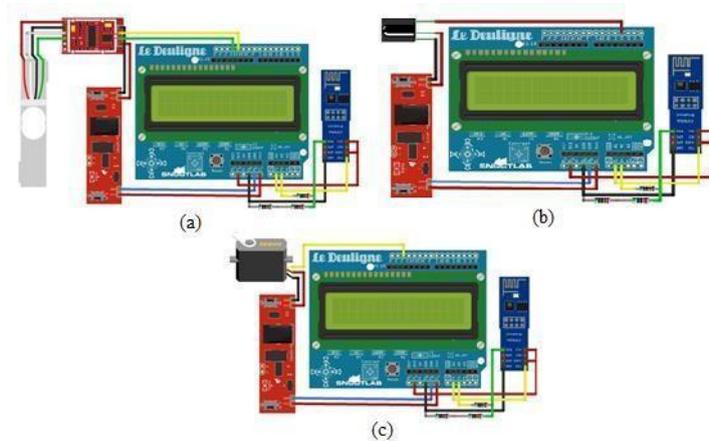
User melakukan input TPM secara manual melalui LCD keypad shield, servo secara otomatis akan bekerja mengubah sudut dengan interval  $3^\circ$  dan delay 10s setiap berubahannya hingga jumlah TPM sudah sesuai dengan yang di *input*-kan. Servo akan terus bekerja, hal ini difungsikan untuk menekan *error* pada nilai *output* TPM. Monitoring sisa cairan infus dan nilai TPM dapat dilihat secara berkala, dan sistem *alarm* akan bekerja ketika loadcell memberikan *input* sisa cairan kurang dari 100 cc, untuk lebih jelasnya secara alur kerja sistem dapat dilihat flowchart gambar 2.



Gambar 2. Flowchart sistem

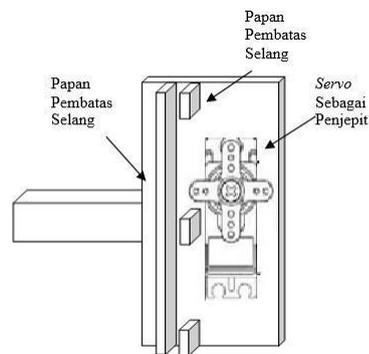
## B. Perancangan hardware

Pada bagian ini membahas tentang perancangan hardware, dimana perancangan hardware ini dibagi menjadi 3 bagian diantaranya, perancangan load cell, perancangan sensor optocoupler dan perancangan servo penjepit selang infus pasien. Pada gambar 3(a) adalah perancangan sensor load cell.



**Gambar 3.** (a) Desain hardware *load cell*, (b) Desain hardware Opto Coupler, (c) Desain hardware Servo

Dari gambar 3(a) dapat dijelaskan bahwa inputan sensor *load cell* berasal dari infus, setelah sensor *load cell* melakukan pembacaan, keluaran dari sensor ini berupa data analog sehingga harus dikonversikan menggunakan modul HX711 agar dapat diproses oleh mikrokontroler. Hasil pembacaan berat botol infus dikonversikan menjadi ml dengan Arduino Uno dan ditampilkan pada LCD dan Telegram. Tiang infus akan dimodifikasi pada pengait botol infus dimana pengait asli akan diganti dengan sebuah besi pen yang telah dilubangi sesuai dengan lubang pada sensor *load cell*. Selanjutnya adalah perancangan sensor optocoupler yang dapat dilihat pada gambar 3(b). Perhitungan jumlah tetesan dilakukan menggunakan sensor *Optocoupler* yang diletakan lurus diantara ruang tetes. Pembacaan perhitungan oleh mikrokontroler berupa logika *low – high* menurut *Photodiode*. Hasil perhitungan akan ditampilkan pada lcd. Selanjutnya adalah perancangan motor servo sebagai pengatur tetesan infus dapat dilihat pada gambar 3(c). Pada sistem kontrol ini menggunakan metode gravitasi sehingga fungsi *roller clamp* pada infus digantikan oleh motor servo. Untuk memperjelas proses perancangan penjepitan selang infus dapat dilihat pada gambar 4.



**Gambar 4.** Desain Penjepit

Penggunaan motor servo sebagai pengatur jumlah TPM dimana cara kerja alat ini berdasarkan data masukan sensor optocoupler. Posisi awal motor servo akan menjepit selang hingga tidak ada tetesan, selanjutnya servo akan bergerak memperbesar sudut hingga terdapat tetesan. Langkah selanjutnya pergerakan servo dari memperbesar sudut atau memperkecil sudut terikat pada data masukan yang diberikan oleh sensor optocoupler.

## III. HASIL DAN PEMBAHASAN

Pada bagian ini dilakukan pengujian dan analisa setelah *hardware* dan *software* sudah diselesaikan. Kegiatan ini bertujuan untuk mengetahui kinerja dari *hardware* dan *software*, sehingga dapat memberikan hasil berupa data dimana data tersebut akan dianalisa.

**A. Pengujian sensor load cell**

Pengujian Sensor Berat (*Load Cell*) ini dilakukan agar mendapatkan hasil kalibrasi yang tepat berdasarkan acuan berat sesungguhnya. Komponen yang digunakan dalam pengujian ini adalah Sensor *load cell*, neraca digital dan Arduino Uno. Sedangkan untuk Variabel kontrolnya menggunakan buah timbangan dengan berat 50gr, 100gr, 200gr, 500gr, dan 1000gr. Maka selanjutnya dilakukan kalibrasi untuk sensor *load cell* agar didapat *factory reset* yang lebih tepat dan dilakukan pengukuran kembali untuk setiap buah timbangan yang sudah ditentukan. Tabel 1 adalah hasil kalibrasi dari sensor *load cell*.

**Tabel 1.** Data kalibrasi *load cell*

No	Berat Benda (gr)	Timbangan digital (gr)	Loadcell (gr)	Selisih (%)
1	50	50	50	0
2	100	100	100	0
3	200	200	200	0
4	500	500	479	4.2
5	1000	1000	986	1.4
6	50	50	51	2
7	100	100	99	1
8	200	200	201	0.5
9	500	500	487	2.6
10	1000	1000	993	0.7
11	50	50	50	0
12	100	100	99	1
13	200	200	200	0
14	500	500	490	2
15	1000	1000	985	1.5
<b>Selisih Rata-rata</b>				1,12%

Pada proses kalibrasi ini sensor Loadcell menggunakan modul penguat sinyal berupa HX711 yang juga berfungsi sebagai ADC untuk meng-konversi sinyal analog ke sinyal digital. Sehingga proses kalibrasi ini terfokus pada *adjustment* nilai *factory reset* yang memiliki acuan objek buah timbangan yang sebelumnya telah diukur satuan beratnya menggunakan neraca digital. Proses penyesuaian nilai *factory reset* terus dilakukan hingga didapat nilai *output* sama dengan nilai *input*. Pada awal sebelum dilakukan kalibrasi kondisi *Load cell* memiliki nilai *factory reset* sebesar 1000.00, selanjutnya dikarenakan selisih pengukuran bernilai positif maka nilai *factory reset* dikurangi. Proses kalibrasi ini menggunakan buah timbangan seberat 50 gr, sehingga tahapan kalibrasi harus mendapatkan nilai hasil pengukuran sebesar 50 gr dan ketika beban dilepaskan harus mendapatkan nilai sebesar 0 gr.

Setelah dilakukan pengurangan nilai *factory reset* hingga kedua kondisi terpenuhi didapat angka *factory reset* sebesar 688.80. Data pada tabel 1 didapatkan data rata - rata *error* sebesar 1.12%, adanya nilai *error* ini didapat setelah dilakukan pengujian berat buah timbangan masing – masing sebanyak 3 kali dengan urutan sesuai pada tabel 4.2. Nilai *error* paling tinggi didapat saat melakukan pengukuran buah timbangan 500gr dan 1000gr, sehingga menghasilkan kesimpulan sementara tingkat akurasi paling baik dari sensor ini dibawah 500 gr. Gambar 5 adalah saat pengujian dilakukan.



**Gambar 5.** Proses kalibrasi *load cell*

**B. Pengujian sensor optocoupler**

Pada pengujian ini akan dibandingkan antara jumlah pembacaan tetesan secara manual dengan pembacaan oleh sensor Optocoupler, serta menguji secara singkat pembacaan besaran nilai TPM berdasarkan sensor Optocoupler. Pengujian dilakukan pada cahaya sekitar sensor Optocoupler yang diatur sebesar 100 lux menggunakan senter tangan yang dikaitkan pada tripod. Besaran sudut Servo dilakukan secara berurutan yaitu dari 50°, 55°, dan 60°. Dimana hasilnya akan dianalisa guna perbandingan akurasi sensor Optocoupler terhadap intensitas cahaya ruang sekitar. Pada pengujian ini jarak senter tangan diatur berhadapan dengan ruang tetes, dan diatur tingkat kecerahannya dengan menggeser jarak tripod lalu melakukan pengukuran intensitas cahaya menggunakan lux meter. Tabel 2 adalah data pengujian sensor optocoupler.

**Tabel 2.** Data pengujian optocoupler

<i>Sudut Servo</i>	<i>Durasi (s)</i>	<i>Tetes</i>		<i>Tetes Per Menit</i>		<i>Error (%)</i>
		<i>Aktual</i>	<i>Terbaca</i>	<i>Aktual</i>	<i>Terbaca</i>	
50	60	15	14	15	15	0
55	60	32	32	32	33	3.1
60	60	33	32	33	34	0
50	60	11	10	11	11	0
55	60	29	28	29	28	3.4
60	60	32	31	32	32	0
50	60	14	13	14	14	0
60	60	28	27	28	28	0
50	60	21	20	21	21	0
55	60	23	22	23	23	0
60	60	33	32	33	34	2.9
50	60	12	11	12	13	8
55	60	29	29	29	28	3.4
60	60	33	32	33	36	9
50	60	15	14	15	15	0
<b>Rata-rata Error</b>						2,9

Pada pengujian ini, dimana cahaya sekitar sensor diatur hingga 100 lux memiliki nilai 2.9% untuk *error* pembacaan nilai TPM. *Error* yang timbul didapati dari tetes pertama yang tidak terbaca. Seperti pada pengujian pertama, pada data serial monitor dapat dilihat jeda waktu yang cukup panjang, tetes pertama yang jatuh pada detik ke-dua setelah servo terbuka sebesar 50° tidak ter-counter oleh sensor Optocoupler. Perolehan nilai *error* pada jumlah tetesan diperoleh dari faktor yang sama, yaitu tidak ter-counter-nya tetesan pertama pada pengujian ini. gambar 6 adalah proses pada saat pengambilan data.



**Gambar 6.** Proses Pengujian Optocoupler

### C. Pengujian motor servo

Pengujian motor servo ini dilakukan dengan cara memerintahkan servo bergerak dari 0° - 180° dengan interval 10° yang diperintahkan melalui serial monitor. Hal ini untuk melihat tingkat akurasi dan kinerja servo terhadap program yang telah dibuat pada sistem ini. Proses pengujian ini dilakukan dengan motor Servo MG90, Arduino Uno, Kertas busur. Dikarenakan torsi yang dimiliki motor servo yang hendak diuji, maka pada pengujian ini kertas busur direkatkan pada kayu penyangga yang sudah direkatkan ke lubang baut motor Servo MG90 dan dikunci menggunakan mur dan baut ukuran m4. Hal ini dilakukan untuk meminimalisir Bergeraknya kertas busur disaat motor servo bekerja, sehingga hasil pengujian dapat dianalisa secara tepat. Hasil pengujian servo dapat dilihat pada tabel 3.

**Tabel 3.** Data pengujian servo

<i>Sudut</i> (°)	<i>Terbaca</i> (°)	<i>Selisih</i> (%)	<i>Sudut</i> (°)	<i>Terbaca</i> (°)	<i>Selisih</i> (%)
10	10	0	20	20	0
30	30	0	40	40	0
50	50	0	60	60	0
70	70	0	80	80	0
90	90	0	100	100	0
110	110	0	120	122	1,6
130	132	1,5	140	142	1,4
150	152	1,3	160	160	0
170	170	0	180	180	0

**Rata-rata selisih = 0.3**

Pada pengujian ini *output* sudut memberikan hasil yang tepat hingga pada sudut 110°. Sudut 120° - 150° memiliki nilai *error*. Saat diujikan perintah 120°, Servo mengalami hentakan sehingga berhenti pada 122°. Namun saat diperintahkan 130° servo bergerak tepat 10°, hingga berhenti pada sudut 132°. Hal ini terjadi berulang hingga diperintahkan 150° dimana Servo berhenti pada 152°, saat diperintahkan pada sudut 160°, motor Servo bergerak dan bertambah sebesar 8° sehingga ujung mata servo berhenti tepat pada 160°. Selanjutnya diperintahkan 170°, servo bergerak tepat pada sudut 170° lalu terakhir diperintahkan bergerak ke sudut 180° dan Servo bergerak tepat di sudut 180°.

### D. Pengujian keseluruhan sistem

Pada bagian ini akan diujikan keseluruhan sistem dengan pemberian infus dengan level cairan yang berbeda - beda dan pemberian nilai TPM yang berbeda - beda. Pada saat pengujian akan diambil beberapa *sampling* data dengan rentan waktu acak berupa sisa cairan infus dan jumlah TPM saat ini. Pengujian ini juga dilakukan hingga level infus dibawah 100cc, hal ini untuk mengetahui apakah fungsi alarm bekerja sesuai fungsinya. Perhitungan *error* akan dilakukan dari selisih antara data LCD dan data Bot Telegram. Lalu perhitungan *error* selanjutnya dihitung dari perbandingan selisih antara kolom *input* dengan *output* Bot Telegram. Tabel 4 adalah hasil dari pengujian keseluruhan sistem.

**Tabel 4.** Data pengujian sistem keseluruhan

<i>Input</i>		<i>Output LCD</i>		<i>Output Telegram</i>		<i>Error (%)</i>	
<i>Infus</i> (cc)	<i>TPM</i>	<i>Infus</i> (cc)	<i>TPM</i>	<i>Infus</i> (cc)	<i>TPM</i>	<i>infus</i>	<i>TPM</i>
471	10	471	12	471	12	0	0
337	20	337	20	337	20	0	0
280	30	280	30	279	30	0.35	0
195	40	193	40	185	40	4.1	0
170	50	167	47	167	47	0	0
153	10	153	10	153	8	0	20
247	20	247	20	247	20	0	0
321	30	321	28	321	28	0	0
439	40	439	39	440	39	0.22	0
530	50	530	48	528	48	0.37	0
544	10	544	10	543	10	0.18	0
460	20	460	21	460	21	0	0

338	30	338	30	338	30	0	0
237	40	237	41	237	41	0	0
128	50	127	51	128	51	0	0
<b>Rata – rata Error =</b>						<b>0.34%</b>	<b>1.3%</b>

Dari data yang diamati, *error* paling banyak didapat pada hasil pengukuran Level infus sebesar 0.34%, sedangkan untuk nilai TPM keseluruhan memiliki nilai *error* sebesar 1.3%. Data *error* pada pengukuran level infus disebabkan karena adanya pergerakan angka pengukuran *load cell* secara *signifikan* dimana hasil *output*-nya selalu berpindah – pindah  $\pm 1$  angka. Sehingga data pada tabel pengukuran level infus selalu memiliki selisih satu angka, terkecuali pada pengujian ke-4 selisih mencapai 4 angka dikarenakan adanya keterlambatan pengiriman data oleh Esp8266-01 ke *Bot* telegram. Hal ini dikarenakan *router* telah mencapai kondisi *maximal user* sehingga ada *delay* pada Esp8266-01 untuk mengirim data. Dalam arti lain, pengambilan data pada lcd dilakukan saat servo masih terbuka dengan sudut yang kecil, sedangkan data yang dikirimkan oleh Esp8266-01 ke *Bot* Telegram yaitu saat infus sudah menetes dengan jumlah yang lebih besar sehingga menimbulkan selisih angka yang cukup besar.

Secara keseluruhan dari 15 kali pengujian didapati data *error* sebanyak 5, dengan selisih  $\pm 1$  angka. Berbeda dengan *error* pengukuran nilai TPM, dimana dari 15 kali pengujian didapati satu data *error* yang cukup besar yaitu 20%. Namun besarnya data *error* tersebut masih dalam *range* sistem, dimana sistem memberikan toleransi sebesar 3 angka. Data *error* tersebut di-*input*-kan sebesar 10 TPM, namun sistem mengatur sudut servo dan membaca 8 TPM maka sistem mengolah data tersebut bernilai benar karena masih berada dalam *range* yang telah ditentukan.

Pengujian sistem alarm berupa pesan telegram juga bekerja dengan baik. Sistem ini mengirim pesan peringatan yang dikirim melalui Esp8266-01 namun tidak terikat pada *delay* pengiriman data TPM dan Level infus. Dapat dilihat pada Lampiran 4.9.1 ketika level infus berada di bawah 100cc maka secara konstan setiap 30s sistem mengirim pesan ke *Bot* telegram yang juga diterima semua pengguna telegram yang tergabung pada *Bot* telegram yang telah dibuat. Fungsi kontrol juga dapat bekerja sesuai fungsinya, ketika level infus dibawah 100cc diujikan mematikan tetesan infus menggunakan menu *Bot* telegram. Pada serial monitor ditampilkan respon Esp8266-01 lalu memerintahkan “mematikan infus” dimana perintah ini sama dengan perintah “*Stop*” pada *Lcd Keypad Shield* yang memerintahkan servo bergerak ke sudut minimal yang telah ditentukan yaitu 10°.

Pada saat pengujian, didapati nilai TPM selalu bertambah setelah Esp8266-01 mengirim data ke *Bot* Telegram. Saat diteliti Servo selalu melakukan hentakan disaat Esp8266-01 mengirim data ke *Bot* Telegram, servo bergerak memperbesar sudut lalu kembali lagi pada sudut semula sebelum Esp8266-01 mengirim data. Faktor – faktor yang dapat mempengaruhi antara lain adanya kesamaan power dari satu sumber. Jika dilihat pada *output* serial monitor, setiap kali Esp8266-01 memberikan *output* pada serial monitor maka nilai TPM berubah meskipun sebelumnya nilai *output* telah sesuai dengan *input* yang diberikan.

## V. KESIMPULAN

Pada penelitian ini, terdapat pengujian sistem dimana dengan pembagian antara pengujian level cairan infus, pengujian nilai TPM, dan pengujian sistem *alarm* dengan hasil, dimana sistem kontrol mampu bekerja mengatur nilai TPM dengan tingkat keberhasilan sebesar 98.7%. Sistem *alarm* mampu bekerja secara aktual berdasarkan *input* load cell yang memiliki nilai *error* sebesar 0.34%. Sistem Monitoring dapat bekerja sesuai tujuan penelitian dimana dengan acuan pengambilan data melalui *bot* telegram dapat dibalas secara aktual. Berdasarkan hasil tersebut maka sistem ini sudah sesuai dengan tujuan penelitian, dan harapan kedepan agar sistem ini mampu memberikan informasi kepada perawat saat infus hampir habis.

## REFERENSI

- [1] A. Febrianti, “Evaluasi Pelaksanaan Identifikasi Pasien Dalam Pemberian Terapi, Transfusi, Pemeriksaan Penunjang Terhadap Insiden Di Rumah Sakit,” *J. Health Sains*, vol. 2, no. 1, Art. no. 1, Jan. 2021, doi: 10.46799/jhs.v2i1.73.
- [2] “Diduga Akibat Perawat Lalai, Bayi 4 Hari Tewas : Okezone News.” <https://news.okezone.com/read/2009/09/24/340/259679/diduga-akibat-perawat-lalai-bayi-4-hari-tewas> (accessed Apr. 21, 2022).
- [3] T. Kusuma and M. T. Mulis, “Perancangan Sistem Monitoring Infus Berbasis Mikrokontroler Wemos D1 R2,” *Konf. Nas. Sist. Inf. KNSI 2018*, no.0, Art. no. 0, Mar. 2018, Accessed: Apr. 21, 2022. [Online]. Available: <http://jurnal.atmaluhur.ac.id/index.php/knsi2018/article/view/549>
- [4] S. Purwanto, M. Mulya, and S. Sembiring, “MONITORING INFUS BERDASARKAN WAKTU TETESAN,” *Proceeding Semin. Nas. Keperawatan*, vol. 3, no. 1, Art. no. 1, Apr. 2018.

- [5] “Sistem Monitoring Infus dengan Human Machine Interface secara Wireless | NATALIANA | ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika.” <https://ejournal.itenas.ac.id/index.php/elkomika/article/view/2581> (accessed Apr. 21, 2022).
- [6] S. Adi, A. A. Kunto, T. Suheta, and S. Muharom, “Pengaturan Tingkat Suhu Dan Kelembaban Pada Mesin Penetas Telur Burung Puyuh,” *Sinar Fe7*, vol. 2, no. 1, Art. no. 1, Jul. 2019.
- [7] S. Muharom, I. Masfufah, R. A. Firmansyah, A. Hamid, and S. Oetomo, “Implementasi Kontrol Suhu Menggunakan Metode PID Pada Aplikasi Inkubator Infant Warmers,” *CYCLOTRON*, vol. 4, no. 1, Art. no. 1, Feb. 2021, doi: <http://dx.doi.org/10.30651/cl.v4i1.5275>.
- [8] S. Muharom, H. Suseno, and S. A. Setyawan, “Rancang Bangun Sistem Penyiram Tanaman Bawang Merah Secara Otomatis,” *Pros. Semin. Nas. Sains Dan Teknol. Terap.*, vol. 1, no. 1, pp. 385–390, Sep. 2019.
- [9] B. D. C. Ximenes and S. Muharom, “Rancang Bangun Sistem Pemberian Pakan Ikan Otomatis Pada Aquarium Berbasis Arduino Uno,” *Sinar Fe7*, vol. 3, no. 1, Art. no. 1, Sep. 2020.
- [10] E. Alfianto, S. Agustini, S. Muharom, F. Rusydi, and I. Puspitasari, “Design Monitoring Electrical Power Consumption at Computer Cluster,” *J. Phys. Conf. Ser.*, vol. 1445, p. 012027, Jan. 2020, doi: [10.1088/1742-6596/1445/1/012027](https://doi.org/10.1088/1742-6596/1445/1/012027).
- [11] I. S. Saputra, A. Ramadhan, and S. Muharom, “Sistem Kontrol dan Monitoring Penerangan Lampu Taman Berbasis Website,” *Pros. Semin. Nas. Tek. Elektro Sist. Inf. Dan Tek. Inform. SNESTIK*, vol. 1, no. 1, Art. no. 1, Apr. 2022, doi: [10.31284/p.snestik.2022.2793](https://doi.org/10.31284/p.snestik.2022.2793).