



Ranah Research : Journal of Multidisciplinary Research and Development

+62 821-7074-3613

ranahresearch@gmail.com

<https://jurnal.ranahresearch.com/>



Rancang Bangun Kontrol Kecepatan Cairan Infus Berbasis Arduino Uno

*Sandi Youda*¹, *Juli Sardi*²

¹ Universitas Negeri Padang, Indonesia, sandiyouda@gmail.com

² Universitas Negeri Padang, Indonesia, sardijuli@gmail.com

Corresponding Author: sandiyouda@gmail.com¹

Abstract: *Technological developments can provide benefits if applied properly. Along with this, the need for technology is also very much needed in the medical world. With the condition of a large hospital, the large number of patients and the limited number of medical personnel and the demand for good patient care has always been a problem in every hospital. One of the problems is the provision of infusion fluids, in this research the aim is to make scientific work to realize designs, and programs, and determine the performance of Arduino Uno-based infusion fluid speed control devices, using the Weight Sensor (Load Cell) method as a detector for remaining infusion fluids, and using the NodeMCU8266 as a data transfer medium, as well as controlling the infusion flow rate using the Macro set and Micro set methods. After testing this device, it was found that this device worked well and was suitable for the research subject. This system is expected to make it easier for nurses in hospitals to monitor and replace infusion fluids.*

Keyword: *Infusion speed control, Arduino uno*

Abstrak: Perkembangan teknologi dapat memberikan manfaat bila diterapkan secara tepat. Seiring dengan hal tersebut, maka kebutuhan teknologi juga sangat dibutuhkan dalam dunia medis. Dengan kondisi rumah sakit yang luas, jumlah pasien yang banyak serta keterbatasan tenaga medis dan tuntutan pelayanan pada pasien yang baik selalu menjadi masalah dalam setiap rumah sakit. Salah satu masalahnya adalah pemberian cairan infus, pada penelitian ini bertujuan membuat karya ilmiah untuk Merealisasikan rancangan, program, dan mengetahui unjuk kerja alat Kontrol kecepatan cairan infus berbasis Arduino Uno, dengan memakai metode Sensor Berat (*Load Cell*) sebagai pendeteksi sisa cairan infus, dan menggunakan NodeMCU8266 sebagai media transfer data, serta kontrol kecepatan aliran infus menggunakan metode set Makro dan set Mikro. Setelah dilakukannya uji coba pada perangkat ini, diperoleh hasil bahwa perangkat ini dapat bekerja dengan baik dan juga sesuai subjek penelitian. Sistem ini diharapkan dapat mempermudah para perawat dirumah sakit dalam melakukan monitoring dan pergantian Cairan infus.

Kata Kunci: Kontrol kecepatan infus, Arduino uno

PENDAHULUAN

Semakin berkembangnya kemajuan teknologi, belakangan ini meningkatkan kreasi manusia dalam menciptakan perangkat yang dapat mendukung kinerja manusia dalam melakukan proses pekerjaan agar lebih praktis dan efisien. Perkembangan teknologi itu dapat memberikan manfaat bila diterapkan secara tepat. Seiring dengan hal tersebut, maka kebutuhan teknologi juga sangat dibutuhkan dalam dunia medis, terkhususnya tenaga keperawatan dalam hal pemberian cairan infus dimana perawat harus mengecek setiap saat yang ditentukan dan menggantinya dengan yang baru, tetapi seringkali pasien tidak mengetahui saat cairan infus tersebut habis dan kerepotan untuk menekan tombol ke ruang penjaga untuk memberitahukan bahwa cairan infusnya habis ataupun tidak menetes. Dalam tugasnya memantau kondisi infus pasien biasanya perawat harus memeriksa kondisi infus pasien tiap waktu yang telah diperkirakan sebelumnya, sehingga perawat harus mondar-mandir memeriksa keadaan dari infus pasien. Padahal hal ini juga dapat menyebabkan timbulnya komplikasi lain antara lain darah dapat tersedot naik ke selang infus dan dapat membeku pada selang infus sehingga mengganggu kelancaran aliran cairan infus (Zainuri dkk., 2012). Selain itu, jika tekanan pada infus tidak stabil, darah yang membeku pada selang infus dapat tersedot kembali masuk ke dalam pembuluh darah. Darah yang membeku (*blood clot*) tersebut dapat beredar keseluruh tubuh dan dapat menyumbat kapiler darah di paru sehingga menyebabkan emboli di paru (Waitt et al., 2004)

Untuk memahami permasalahan yang timbul karena kelalaian petugas, diharapkan juga dapat memudahkan perawat dalam pemantauan penggantian cairan intravena serta mengurangi resiko keterlambatan penggantian cairan intravena bagi pasien. Pada tahun 2012, terdapat penelitian terdahulu oleh Syahrul dan Hidayat dengan judul “*Sistem Pemantauan Infus Secara Terpusat*” sedangkan pada tahun 2010 oleh Fathur Zaini dkk melakukan penelitian dengan mengangkat judul “*Pengembangan Prototipe Sistem Kontrol dan Monitoring Untuk Pasien Berbasis Jaringan Nirabel*”. Hal yang membedakan dengan penelitian ini adalah monitoring dan bisa mengontrol kecepatan aliran cairan infus serta sensor yang digunakan dimana pada penelitian terdahulu menggunakan sinar inframerah, laser, dan photodiode serta pengontrolan melalui komputer. Sedangkan pada penelitian ini sensor yang digunakan adalah *load cell* yang berfungsi sebagai sensor berat dari cairan intravena mulai dalam keadaan penuh hingga batas minimal yang ditentukan serta monitoring melalui android secara *real time*.

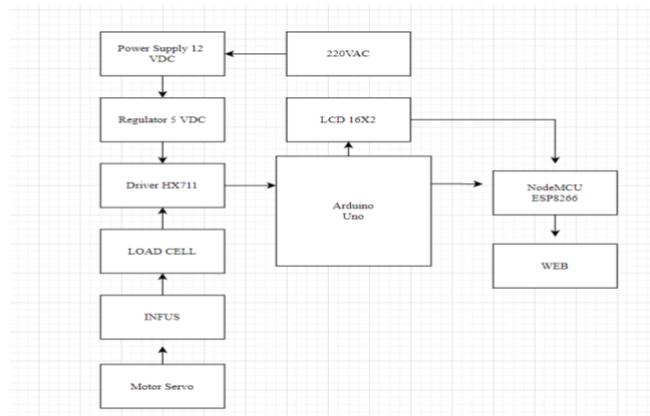
Pemilihan sensor didasari karena Sensor *Load Cell* adalah sensor berat yang tidak akan mengurangi serta memperangaruhi kandungan yang terdapat pada cairan intravena dan pemilihan android sebagai media monitoring didasari karena android memiliki ukuran yang kecil sehingga bisa dibawa kemana saja serta memudahkan perawat dalam memonitoring sisa cairan infus dari mana saja tanpa harus pergi ke ruangan kontrol untuk memonitoring sisa cairan infus pasien melalui komputer yang ada di ruangan kontrol. Data dari hasil sensor akan dikirim ke controller yang menggunakan Arduino, controller ini akan mengolah data yang diperoleh untuk dikirimkan melalui *transmitter* ke *receiver*. Setelah data diterima oleh *receiver* maka dapat menampilkan data yang diterima melalui Android.

METODE

Metode yang digunakan pada tugas akhir ini adalah perancangan *hardware* dan *software*. Perancangan *hardware* terdiri dari perancangan mekanik dan komponen kelistrikan yang digunakan dalam tugas akhir ini. Sedangkan perancangan *software* terdiri dari blok diagram, *flowchart* dan jenis program yang digunakan dalam pembuatan alat ini.

A. Blok Diagram

Adapun blok diagram dari alat seperti dibawah ini:



Gambar 1. Blok diagram sistem keseluruhan

B. Perancangan Alat

1. Perancangan *Hardware*

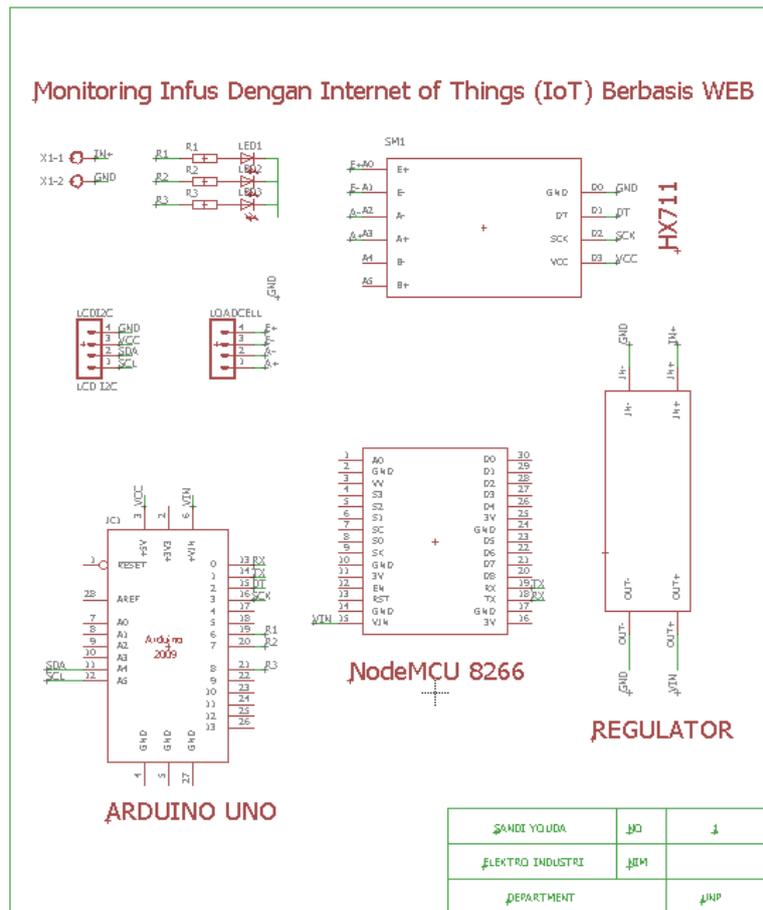
Secara garis besar perancangan *hardware* atau perangkat keras terdiri dari rancangan mekanik dan elektronik.



Gambar 2. Desain alat keseluruhan

2. Perancangan Elektronik

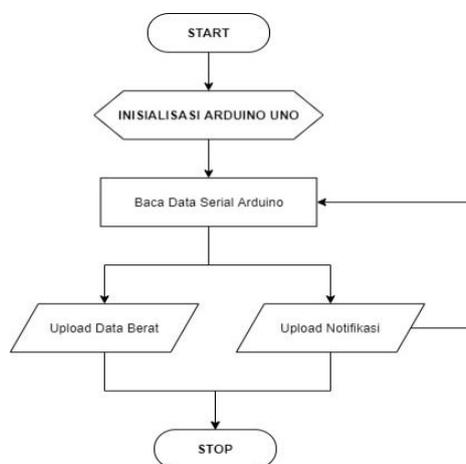
Rangkaian elektronik pada tugas akhir ini menampilkan pemasangan komponen elektronik secara keseluruhan.



Gambar 3. Schematic rangkaian keseluruhan sistem

3. Diagram Alir atau Flowchart

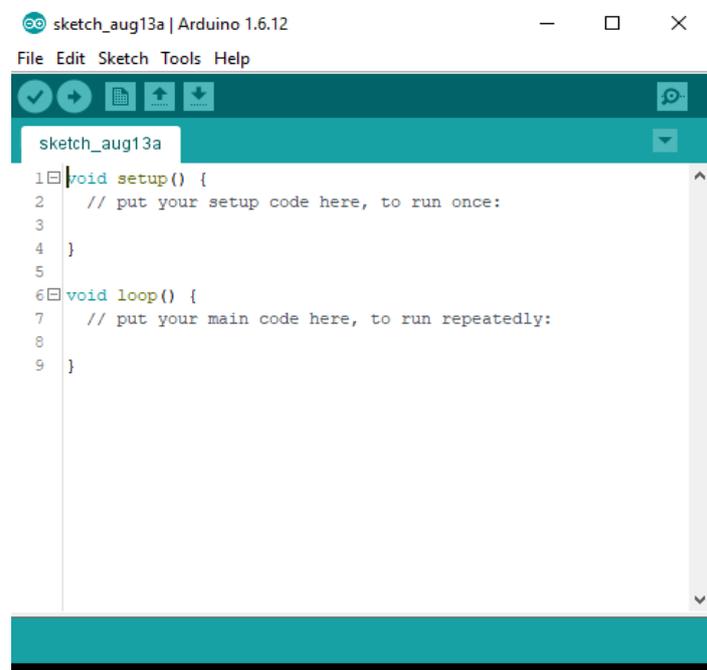
Diagram alir atau *flowchart* adalah suatu metode yang menggambarkan aliran proses dari suatu operasi, diagram ini sangat cocok untuk dipelembatkan dengan memanfaatkan algoritma yang dituliskan pada komputer. *Flowchart* dapat digunakan untuk menggambarkan cara kerja dari alat yang akan dibuat kedalam simbol-simbol agar lebih mudah untuk dipahami, bentuk diagram alir atau *flowchart* alat dapat dilihat pada gambar-gambar dibawah ini :



Gambar 4. Flowchart kerja alat pada NodeMCU 8266

C. Perancangan *Software*

Pada tugas akhir ini, pemograman alat yang digunakan adalah *software* Arduino IDE. *Software* ini menggunakan bahasa pemograman C. *Listign* program arduino ini dikenal dengan nama *sketch*, “*void setup*” dan “*void loop*” merupakan dua buah fungsi dari tiap-tiap *sketch*.



Gambar 5. Tampilan text editor arduino IDE

Void setup dijalankan pada saat *sketch* atau program Arduino mulai. Fungsi digunakan untuk menginisiasi variabel, mendeklarasikan pin yang akan digunakan, menggunakan *library* dan lain lain. Sedangkan *Void loop* dijalankan setelah fungsi setup sudah selesai dijalankan, *void loop* bertujuan untuk mengeksekusi dan menjalankan program yang sudah dibuat. Fungsi ini akan secara aktif mengontrol *board* Arduino baik membaca *input* atau merubah *output* berbeda dengan *void setup* yang hanya dijalankan sekali oleh Arduino, fungsi *void loop* akan dijalankan berulang kali oleh Arduino secara berkala. Oleh sebab itu diberi nama *loop* yang artinya berulang.

HASIL DAN PEMBAHASAN

Pengujian Sensor dan Kontrol Infus

1. Pengukuran

Pengukuran sensor infus ini bertujuan untuk mengetahui apakah data yang dihasilkan oleh sensor dan ditampilkan pada aplikasi sesuai dengan keadaan sebenarnya. Pada sensor infus penulis menggunakan sensor optik untuk mengetahui kapan ada tetesan lalu tetesan ini dihitung waktu dengan tetesan berikutnya, jika waktu didapatkan maka dapat diketahui jumlah tetesan dalam 1 menit.

$$\text{Kecepatan Infus (Dpm)} = \frac{6000}{(\text{Selisih waktu tetesan (ms)})} \quad (1)$$

Data yang diperoleh tersebut disajikan melalui tabel yang dapat dilihat seperti tabel 1 dibawah ini.

Tabel 1. Hasil pengujian sensor infus

No Percobaan	Persentase Bukaan (%)	Perhitungan Manual (Drop/Minute)	Data Sensor Infus Drop/Minute	Error (%)
1	0	0	0	0
2	75	64	68	6.25
3	50	125	130	4
4	25	156	162	3.84
5	0	190	201	7.3

Bagian ini memuat data (dalam bentuk ringkas), analisis data dan interpretasi terhadap hasil. Hasil dapat disajikan dengan tabel atau grafik untuk memperjelas hasil secara verbal, karena adakalanya tampilan sebuah ilustrasi lebih lengkap dan informative dibandingkan dengan tampilan dalam bentuk narasi. Hasil perhitungan nilai *Error* Sensor Infus :

$$\begin{aligned} \% \text{ Error Percobaan 1} &= \frac{(\text{Nilai Terukur} - \text{Nilai Sebenarnya})}{\text{Nilai Sebenarnya}} \times 100\% \\ &= \frac{0-0}{0} \times 100\% \\ &= 0\% \end{aligned}$$

$$\begin{aligned} \% \text{ Error Percobaan 2} &= \frac{(\text{Nilai Terukur} - \text{Nilai Sebenarnya})}{\text{Nilai Sebenarnya}} \times 100\% \\ &= \frac{68-64}{64} \times 100\% \\ &= 6.25\% \end{aligned}$$

$$\begin{aligned} \% \text{ Error Percobaan 3} &= \frac{(\text{Nilai Terukur} - \text{Nilai Sebenarnya})}{\text{Nilai Sebenarnya}} \times 100\% \\ &= \frac{130-125}{125} \times 100\% \\ &= 4\% \end{aligned}$$

$$\begin{aligned} \% \text{ Error Percobaan 4} &= \frac{(\text{Nilai Terukur} - \text{Nilai Sebenarnya})}{\text{Nilai Sebenarnya}} \times 100\% \\ &= \frac{162-156}{156} \times 100\% \\ &= 3.84\% \end{aligned}$$

$$\begin{aligned} \% \text{ Error Percobaan 5} &= \frac{(\text{Nilai Terukur} - \text{Nilai Sebenarnya})}{\text{Nilai Sebenarnya}} \times 100\% \\ &= \frac{201-190}{190} \times 100\% \\ &= 7.3\% \end{aligned}$$

Analisa

Dari data yang kita dapatkan berdasarkan tabel hasil pengukuran sensor infus dengan data *Error* yang beragam pada setiap kondisi. Jika nilai *error* dirata-ratakan maka didapatkan nilai *Error* total dari sensor infus yaitu sebesar 4.278%.

A. Pengujian Sensor *Load Cell*

1. Pengujian

Tujuan dilakukannya Pengujian sensor *load cell* pada penghitung sisa adalah untuk mengetahui akurasi dari pembacaan sensor *load cell* itu sendiri dan membandingkan berat benda yang terbaca oleh sensor dengan berat benda sebenarnya. Karena data berat yang didapatkan oleh sensor ini nantinya akan dikonversi ke persentase sisa cairan infus dengan membagi berat saat ini dengan berat penuh infus.

$$\% \text{ Sisa} = \frac{\text{Berat Terukur(gram)} - \text{Berat Kantong(gram)}}{\text{Berat Penuh(gram)} - \text{Berat Kantong(gram)}} \times 100\% \quad (2)$$

Adapun data yang didapatkan dari hasil pengujian dapat dilihat pada tabel 2 dibawah ini.

Tabel 2. Hasil pengukuran sensor *load cell*

No Pengujian	Berat Infus dengan Timbangan (Gram)	Berat Infus dengan Sensor (Gram)	Persentase Sisa Timbangan (%)	Persentase Sisa Sensor (%)	Error Sensor (%)
1	557	554	100	100	0
2	466	461	79.7	79.2	0.6
3	417	402	68.8	66.9	2.7
4	379	375	60.4	60	0.61
5	324	320	48.2	47.7	1
6	293	290	41.3	41	0.7
7	243	241	30.2	30	0.6
8	177	175	15.5	15.4	0,6
9	148	147	9.1	0.91	0
10.	107	106	0	0	0

Pada Pengujian kali ini terdapat beberapa perbedaan antara alat ukur manual dengan pembacaan sensor *load cell* dimana kejadian ini disebut dengan *Error*. Dibawah ini akan menunjukkan nilai *Error* pengukuran sensor *load cell*:

$$\begin{aligned} \% \text{ Error Pengujian 1} &= \frac{(\% \text{ Sebenarnya} - \% \text{ Terukur})}{\% \text{ Sebenarnya}} \times 100\% \\ &= \frac{100 - 100}{100} \times 100\% \\ &= 0\% \end{aligned}$$

$$\begin{aligned} \% \text{ Error Pengujian 2} &= \frac{(\% \text{ Sebenarnya} - \% \text{ Terukur})}{\% \text{ Sebenarnya}} \times 100\% \\ &= \frac{79.7 - 79.2}{79.7} \times 100\% \\ &= 0.6\% \end{aligned}$$

$$\begin{aligned} \% \text{ Error Pengujian 3} &= \frac{(\% \text{ Sebenarnya} - \% \text{ Terukur})}{\% \text{ Sebenarnya}} \times 100\% \\ &= \frac{68.8 - 66.9}{68.8} \times 100\% \\ &= 2.7\% \end{aligned}$$

$$\begin{aligned} \% \text{ Error Pengujian 4} &= \frac{(\% \text{ Sebenarnya} - \% \text{ Terukur})}{\% \text{ Sebenarnya}} \times 100\% \\ &= \frac{60.4 - 60}{60.4} \times 100\% \\ &= 0.6\% \end{aligned}$$

$$\begin{aligned} \% \text{ Error Pengujian 5} &= \frac{(\% \text{ Sebenarnya} - \% \text{ Terukur})}{\% \text{ Sebenarnya}} \times 100\% \\ &= \frac{48.2 - 47.7}{48.2} \times 100\% \\ &= 1\% \end{aligned}$$

$$\begin{aligned} \% \text{ Error Pengujian 6} &= \frac{(\% \text{ Sebenarnya} - \% \text{ Terukur})}{\% \text{ Sebenarnya}} \times 100\% \\ &= \frac{41.3 - 41}{41.3} \times 100\% \\ &= 0.7\% \end{aligned}$$

$$\begin{aligned}\% \text{ Error Pengujian 7} &= \frac{(\% \text{ Sebenarnya} - \% \text{ Terukur})}{\% \text{ Sebenarnya}} \times 100\% \\ &= \frac{30.2 - 30}{30.2} \times 100\% \\ &= 0.6\%\end{aligned}$$

$$\begin{aligned}\% \text{ Error Pengujian 8} &= \frac{(\% \text{ Sebenarnya} - \% \text{ Terukur})}{\% \text{ Sebenarnya}} \times 100\% \\ &= \frac{15.5 - 15.4}{15.5} \times 100\% \\ &= 0.6\%\end{aligned}$$

$$\begin{aligned}\% \text{ Error Pengujian 9} &= \frac{(\% \text{ Sebenarnya} - \% \text{ Terukur})}{\% \text{ Sebenarnya}} \times 100\% \\ &= \frac{9.1 - 9.1}{9.1} \times 100\% \\ &= 0\%\end{aligned}$$

$$\begin{aligned}\% \text{ Error Pengujian 10} &= \frac{(\% \text{ Sebenarnya} - \% \text{ Terukur})}{\% \text{ Sebenarnya}} \times 100\% \\ &= \frac{0 - 0}{0} \times 100\% \\ &= 0\%\end{aligned}$$

2. Analisa

Pada pengujian sensor *load cell* didapatkan data yang terlihat pada tabel yang membuktikan akurasi sensor *load cell* yang digunakan pada tugas akhir ini memiliki akurasi cukup bagus karena didapatkan *error* yang sangat kecil pada 10 kali pengujian dengan data berat yang berbeda-beda dengan berat dari berat penuh sampai kantong infus kosong. Didapatkan *error* minimal 0 % dan maksimal sebesar 2.7%.

B. Pengujian Keseluruhan

Pada Pengujian ini penulis melakukan pengujian sebanyak 10 kali dengan pengujian data secara keseluruhan maksudnya dengan menjalankan program secara keseluruhan baik program nodeMCU, arduino, dan aplikasi secara bersamaan. Hal ini diperlukan untuk mengetahui apakah integrasi pada aplikasi berjalan sesuai yang diinginkan atau tidak. Pengujian dilakukan dengan menguji pembacaan data secara keseluruhan dari awal hingga akhir dalam satu waktu. Dengan begitu bisa ditentukan apakah sistem menjalankan semua kondisi sesuai dengan yang ada pada program.

Alur pengujian dimulai alat akan diberi sumber tegangan 220VAC lalu alat akan menampilkan data nama dan nim penulis. Setelah itu alat akan berikan perintah naik sebanyak 5 kali lalu data akan dilihat apakah diterima dengan sempurna. Data tetesan dan data persen juga akan diuji apakah yang ditampilkan pada aplikasi sama dengan yang ditampilkan LCD. Hal ini dilakukan untuk memastikan data yang dibaca di plant dan data yang dibaca di aplikasi sama nilainya. Nantinya juga dilihat apakah sudut servo akan berubah atau tidak. Lalu dengan cara yang sama alat akan diberi perintah turun sebanyak 5 kali. Lalu dilihat data dan juga servo apakah sudutnya berubah atau tidak.

Data dari setiap percobaan penulis sajikan dalam bentuk tabel yang berisikan data hasil pengujian beserta. Adapun tabel yang penulis maksud dapat dilihat pada tabel 3 dibawah ini.

Tabel 3. Hasil pengujian keseluruhan

Percobaan Ke	Perintah	Data Kecepatan Diterima (Alat)	Data Tetesan Diterima (Aplikasi)	Servo Berubah	Status Percobaan
1	Tambah	Ya	Ya	Ya	Berhasil
2	Tambah	Ya	Ya	Ya	Berhasil
3	Tambah	Ya	Ya	Ya	Berhasil
4	Tambah	Ya	Ya	Ya	Berhasil
5	Tambah	Ya	Ya	Ya	Berhasil
6	Kurangi	Ya	Ya	Ya	Berhasil
7	Kurangi	Ya	Ya	Ya	Berhasil
8	Kurangi	Ya	Ya	Tidak	Tidak
9	Kurangi	Ya	Ya	Ya	Berhasil
10	Kurangi	Ya	Ya	Ya	Berhasil

Berdasarkan percobaan yang dilakukan dengan cara diatas dapat dilihat bahwa tingkat keberhasilan setiap parameter pengujian cukup bagus yakni 100% pada penerimaan data kecepatan, 100% pada penerimaan data di aplikasi, dan 90% pada gerakan servo, maka dapat di hitung:

$$\begin{aligned} \% \text{ Error Percobaan } 7 &= 100 - \frac{\text{Jumlah Persentase Keberhasilan}(\%)}{3} \quad (3) \\ &= 100 - \frac{100+100+90}{3} \\ &= 3.33\% \end{aligned}$$

Untuk persentsi error keseluruhan keseluruhan ini adalah 100 dikurangi rata-rata persentase 3 parameter pengujian dan didapat hasil sebesar 3.33%.

KESIMPULAN

Pada tugas akhir kali ini dapat disimpulkan bahwa alat berjalan dengan baik dan setiap sub sistem serta secara keseluruhan cukup sesuai dengan desain secara mekanik maupun secara *software*. Pada pelaksanaannya alat dibuat sesuai dengan target-target sesuai dengan yang tertuang pada perancangan alat. Bahkan pada pengujian secara keseluruhan alat menunjukkan efisiensi yang sampai 96.7 % atapun dengan total hanya sebesar 3.33%. Keberhasilan pada percobaan yang dilakukan sebanyak 10 kali dengan 2 kondisi yang berbeda.

Untuk itu kelebihan alat dapat kita lihat dalam fungsi kemudahan dalam melakukan monitoring yang didapatkan dapat dikatakan sudah cukup baik karena alat dapat memisahkan dengan tepat. Sehingga efisiensi fungsi dapat kita simpulkan cukup baik. Namun, masih ada masalah dalam melakukan kontrol karena kemungkinan servo tidak merespon perintah dari aplikasi masih ditemukan.

Serta kekurangan pada tugas akhir ini memang keterbatasan dalam akses data karena baru pada 1 alat saja. Mekanisme kontrol kecepatan tetesan infus menggunakan motor servo tampaknya masih kurang efisien. Sehingga dibutuhkan kesabaran untuk mendapatkan kecepatan tetesan infus yang sesuai dengan yang diharapkan.

REFERENSI

- Administrator. (2013). *Macam-Macam Website*. www.naevaweb.com, diakses 07 April 2021.
- Akbar, M. I. (2013). *UTS II pengantar elektronika (TM)*.
- Arduino. (2010). *Pengenalan Arduino*. Retrieved from <https://Arduino.cc/>
- Arief, M. R. (2011). *Pemrograman Web Dinamis Menggunakan PHP dan MySQL*. Yogyakarta: CV ANDI OFFSET.
- Arnata, A., Cahaya, N., Intannia, D. (2012). *Prevalensi Kejadian Berpotensi Interaksi Obat Pada Pasien Intensive Care Unit (ICU) Di Rsud Ulin Banjarmasin Tahun 2012*.

- Fajar, A. (2016). Prototype robot pemindah barang menggunakan arduino. *Other Thesis*. Yogyakarta: UPN Veteran
- Hidayat, S. (2009). *Sistem Pemantauan Infus Pasien Terpusat*. Bandung.
- Kusniadi. (2015). *Pengertian, Jenis, Fungsi dan Cara Kerja Router*.
- Lister, E. C. (1988). *Mesin dan Rangkaian Listrik*. Jakarta : Erlangga.
- Prasetyo, M. A. (2017). *Monitoring dengan ESP8266*.
- Rachman, dkk. (2010). Pengembangan prototipe sistem kontrol dan monitoring infus untuk pasien berbasis jaringan nirkabel (Zigbee). *11 th Seminar on Intelligent Technology and Its Applications*, SITIA 2010. ISSN 2087-331X.
- Rachmat, C. A. (2010). *Algoritma Dan Pemograman Dengan Bahasa C*. Yogyakarta: Andi Offset.
- Rangkuti, S. (2016). *Arduino & Proteus Simulasi dan Praktik*. Bandung: Informatika Bandung
- Ruslan, A. (2016). Monitoring cairan infus berdasarkan indikator kondisi dan laju cairan infus menggunakan jaringan wifi. *Jurnal Ilmiah ILKOM UNHAS*, 8(3).
- Syahwil, M. (2013). *Panduan Mudah Simulasi & Praktek Mikrokontroler Arduino*. Yogyakarta: CV Andi Offset.
- Wicaksono, M. F., dan Hidayat. (2017). *Mudah Belajar Mikrokontroler Arduino Diseratai 23 Proyek, termasuk Proyek Ethernet dan Wireless Client Server*. Bandung: Informatika Bandung.
- Zainuri, A. (2012). *Monitoring dan Identifikasi Gangguan Infus Menggunakan Mikrokontroler AVR*.