



**ALAT *MONITORING* TETESAN INFUS MENGGUNAKAN WEB
SECARA *ONLINE* BERBASIS ESP8266 DENGAN PEMROGRAMAN
ARDUINO IDE**

PROYEK AKHIR

**Diajukan kepada Fakultas Teknik Universitas Negeri Yogyakarta untuk
Memenuhi Sebagian Persyaratan Guna Memperoleh
Gelar Ahli Madya Teknik**



Oleh :

Septian Prastyo Aji

NIM. 14507134015

PROGRAM STUDI TEKNIK ELEKTRONIKA

FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

2017

LEMBAR PERSETUJUAN

PROYEK AKHIR

ALAT *MONITORING* TETESAN INFUS MENGGUNAKAN WEB
SECARA ONLINE BERBASIS ESP8266 DENGAN PEMROGRAMAN
ARDUINO IDE



Oleh

SEPTIAN PRASTYO AJI

14507134015

Telah diperiksa dan disetujui oleh pembimbing:

Untuk diuji

Yogyakarta, 3 Oktober 2017

Mengetahui,

Kaprodi Teknik Elektronika

Menyetujui,

Pembimbing Proyek Akhir

Dr. Sri Waluyanti

NIP. 19581218 198603 2 001

Drs. Djoko Santoso, M.Pd

NIP. 19580422 198403 1 002

LEMBAR PENGESAHAN
PROYEK AKHIR
ALAT *MONITORING* TETESAN INFUS MENGGUNAKAN WEB
SECARA *ONLINE* BERBASIS ESP8266 DENGAN PEMROGRAMAN
ARDUINO IDE

Dipersiapkan dan Disusun Oleh:

SEPTIAN PRASTYO AJI

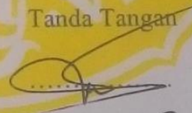
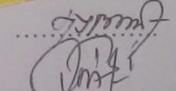
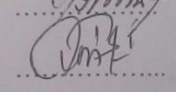
14507134015

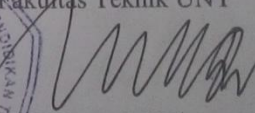
Telah dipertahankan didepan Dewan Penguji Proyek Akhir
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

Pada tanggal

Dan Dinyatakan Telah Memenuhi Syarat Guna Memperoleh Gelar
Ahli Madya Teknik

SUSUNAN DEWAN PENGUJI

Nama	Jabatan	Tanda Tangan	Tanggal
1. Drs. Djoko Santoso, M.Pd	Ketua Penguji		25/10 2017
2. Dr. Sri Waluyanti	Sekretaris Penguji		26/10 2017
3. Dessy Irmawati, M.T	Penguji		26/10 2017

Yogyakarta,
Dekan Fakultas Teknik UNY

Dr. Widarto, M.Pd.
NIR 19631230 198812 1 001

LEMBAR PERNYATAAN KEASLIAN

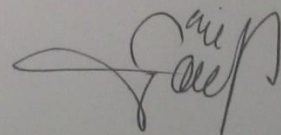
Yang bertanda tangan dibawah ini :

Nama : Septian Prastyo Aji
NIM : 14507134015
Program Studi : Teknik Elektronika D-III
Judul Proyek Akhir : Alat *Monitoring* Tetesan Infus Menggunakan Web
Secara *online* Berbasis ESP8266 dengan
Pemrograman Arduino IDE

Menyatakan bahwa Proyek Akhir ini adalah hasil pekerjaan saya sendiri, dan sepanjang pengetahuan saya, tidak berisi materi yang ditulis oleh orang lain sebagai persyaratan penyelesaian studi di Universitas Negeri Yogyakarta atau perguruan tinggi lain, kecuali bagian-bagian tertentu saya ambil sebagai acuan dengan mengikuti kaidah penulisan karya ilmiah yang benar. Jika ternyata terbukti pernyataan ini tidak benar, sepenuhnya menjadi tanggung jawab saya.

Yogyakarta, 05 Oktober 2017

Yang menyatakan,



Septian Prastyo Aji

NIM. 14507134015

MOTTO

Sesungguhnya bersama kesulitan ada kemudahan. Maka apabila engkau telah selesai dari suatu urusan, tetaplah bekerja keras (untuk yang lain). Dan hanya tuhanmulah engkau berharap. (QS. Al-Insyirah, 6-8)

*"Manusia tidak merancang untuk gagal, mereka gagal untuk merancang."
(William J. Siegel)*

*"Hiduplah seperti pohon kayu yang lebat buahnya; hidup di tepi jalan dan dilempari orang dengan batu, tetapi dibalas dengan buah."
(Abu Bakar Sibli)*

LEMBAR PERSEMBAHAN

Sujud dan syukur kepada Allah SWT. Cinta dan kasih sayang-Mu telah memberikan ku kekuatan, membekali ku dengan ilmu serta memperkenalkan ku dengan cinta. Atas karunia serta kemudahan yang Engkau berikan, akhirnya Laporan Proyek Akhir ini dapat terselesaikan. Sholawat dan salam selalu terlimpahkan kepada junjungan kita Nabi Besar Muhammad SAW.

Kupersembahkan karyaku kepada keluarga sebagai tanda bakti, hormat, dan rasa terima kasih yang tiada terhingga kepada keluarga yang telah memberikan kasih sayang, segala dukungan, dan cinta kasih yang tiada terhingga.

PROYEK AKHIR

ALAT *MONITORING* TETESAN INFUS MENGGUNAKAN WEB SECARA *ONLINE* BERBASIS ESP8266 DENGAN PEMROGRAMAN ARDUINO IDE

Oleh : Septian Prastyo Aji

NIM : 14507134015

ABSTRAK

Tujuan pembuatan alat ini adalah untuk mendapatkan rancang bangun perangkat keras, perangkat lunak, dan unjuk kerja dari alat monitoring tetesan infus menggunakan web secara *online* berbasis ESP8266 dengan pemrograman Arduino IDE, yang berfungsi untuk memantau keadaan infus pasien di rumah sakit sehingga tidak terjadi keterlambatan dalam penanganan infus.

Pembuatan alat *monitoring* tetesan infus ini terdiri dari beberapa tahapan yaitu identifikasi kebutuhan, analisis kebutuhan, blok diagram rangkaian, perencanaan sistem, langkah pembuatan alat, *flowchart* program, pengujian alat dan pengambilan data. Pembuatan alat *monitoring* tetesan infus ini menggunakan besi sebagai rangka. Alat ini menggunakan NodeMCU ESP8266-E12 sebagai kontroler, Modul Sensor IR *Obstacle Avoidance* sebagai pendeteksi tetesan infus, *software* Arduino IDE untuk memprogram alat, dan web sebagai tampilan untuk *monitoring* tetesan infus.

Hasil pengujian menunjukkan bahwa tahapan perancangan alat dapat di implementasikan dan digunakan di rumah sakit pada infus 500 ml menggunakan infus set makro dengan keberhasilan hingga 97,13 %. Program Arduino IDE dengan NodeMCU ESP8266 dapat terintegrasi, sehingga program yang dijalankan dapat berfungsi dengan baik. Modul Sensor IR *Obstacle Avoidance* dapat mendeteksi tetesan infus dengan kecepatan yang berbeda-beda walaupun sulit untuk mengatur jarak pantulan terhadap objek. Tampilan pada web berjalan sesuai dengan fungsinya, dapat memantau ketersediaan cairan infus, mendeteksi kehabisan infus serta mengetahui kemacetan pada infus.

Kata Kunci: *Monitoring* Infus, NodeMCU ESP8266, Modul Sensor IR *Obstacle Avoidance*

KATA PENGANTAR

Assalamu'alaikum wr. wb.

Segala Puji dan syukur penulis panjatkan kehadiran Allah SWT dengan rahmat dan hidayah-Nya Laporan Proyek Akhir ini dapat terselesaikan tanpa halangan yang berarti. Sholawat serta salam tercurah pada Nabi Agung kita Rasulullah Muhammad SAW dan keluarga, sahabat dan orang-orang yang istiqomah di jalan-Nya.

Dalam menyusun Laporan Proyek Akhir ini penulis merasa banyak kekurangan karena terbatasnya kemampuan dan pengetahuan penulis. Oleh karena itu pada kesempatan ini penulis menyampaikan ucapan terima kasih dan penghargaan yang sebesar-besarnya kepada:

1. Drs. Djoko Santoso, M.Pd selaku pembimbing penyusunan laporan proyek akhir dan dosen pendamping akademik.
2. Dr. Sri Waluyanti, M.Pd. selaku Ketua Program Studi Diploma III, Koordinator Proyek Akhir Jurusan Pendidikan Teknik Elektronika Universitas Negeri Yogyakarta.
3. Dr. Fatchul Arifin, M.T. selaku Ketua Jurusan Pendidikan Teknik Elektronika Universitas Negeri Yogyakarta.
4. Dr. Widarto, M.Pd. selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta.
5. Keluarga yang telah memberikan semangat, dukungan dan motivasi.
6. Seluruh Dosen Pengajar Teknik Elektronika Universitas Negeri Yogyakarta atas bekal ilmu yang diberikan kepada penulis

7. Semua pihak yang tidak dapat saya sebutkan satu persatu yang telah membantu hingga terselesainya laporan ini.

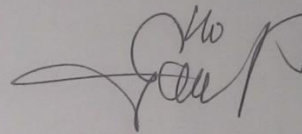
Penulis menyadari bahwa laporan ini masih banyak kekurangan dan jauh dari sempurna walaupun penulis telah berusaha untuk mendekati kesempurnaan, maka penulis berharap para pembaca memberikan saran dan kritik yang membangun.

Akhir kata penulis mohon maaf yang sebesar-besarnya apabila ada kekeliruan di dalam penulisan laporan ini.

Wassalamu'alaikum wr. wb.

Yogyakarta, 05 Oktober 2017

Penulis,



Septian Prastyo Aji

DAFTAR ISI

LEMBAR PERSETUJUAN.....	2
LEMBAR PENGESAHAN	iii
LEMBAR PERNYATAAN KEASLIAN	iv
MOTTO	v
LEMBAR PERSEMBAHAN	vii
ABSTRAK	viii
KATA PENGANTAR	viii
DAFTAR ISI.....	xix
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR	xiv
DAFTAR LAMPIRAN.....	xv
BAB I	1
Pendahuluan	1
A. Latar Belakang	1
B. Identifikasi Masalah	3
C. Batasan Masalah.....	4
D. Rumusan Masalah	4
E. Tujuan	5
F. Manfaat	5
G. Keaslian Gagasan.....	6
BAB II.....	7
Pendekatan Pemecahan Masalah.....	8
A. Terapi Intravena (Infus)	8
B. NodeMCU ESP8266-12E	13

C. Modul Sensor IR Obstacle Avoidance	17
D. Modul Step Down DC-DC LM2596.....	19
E. Perangkat Lunak Arduino IDE	21
F. Website.....	26
G. Cayenne.....	29
BAB III	32
Konsep Rancangan.....	32
A. Identifikasi Kebutuhan	32
B. Analisis Kebutuhan	32
C. Blok Diagram Rangkaian	33
D. Perancangan Sistem	34
E. Langkah Pembuatan	36
F. Perangkat Lunak.....	39
G. Spesifikasi Alat	41
H. Rencana Pengujian	41
I. Tabel Uji.....	43
J. Pengoperasian Alat.....	46
BAB IV	48
Pengujian dan Pembahasan	48
A. Pengujian.....	48
B. Pembahasan.....	55
BAB V.....	61
Kesimpulan dan Saran.....	61
A. Kesimpulan	61
B. Keterbatasan Alat	62

C. Saran.....	63
Daftar Pustaka	64
Lampiran	66

DAFTAR TABEL

Tabel 1. Rencana Pengujian NodeMCU ESP8266-12E	43
Tabel 2. Rencana Pengujian Sensor Inframerah	44
Tabel 3. Rencana Pengujian Aplikasi Cayenne	44
Tabel 4. Rencana Pengujian Catu Daya Tanpa Beban.....	44
Tabel 5. Rencana Pengujian Catu Daya Dengan Beban	45
Tabel 6. Rencana Pengujian Jumlah Tetesan	45
Tabel 7. Rencana Uji Kerja Batas Infus.....	46
Tabel 8. Hasil Pengujian NodeMCU ESP8266-12E.....	49
Tabel 10. Hasil Pengujian Sensor Inframerah.....	50
Tabel 11. Hasil Pengujian Aplikasi Cayenne.....	51
Tabel 12. Hasil Pengujian Catu Daya Tanpa Beban	53
Tabel 13. Hasil Pengujian Catu Daya Dengan Beban.....	53
Tabel 14. Hasil Pengujian Jumlah Tetesan	54
Tabel 15. Hasil Uji Kerja Batas Infus	54

DAFTAR GAMBAR

Gambar 1. Pin Diagram NodeMCU ESP8266.....	14
Gambar 2. Cara Kerja Sensor.....	17
Gambar 3. konfigurasi Modul Sensor IR Obstacle Avoidance.....	18
Gambar 4. Blok diagram Modul Step Down DC-DC LM2596.....	21
Gambar 5. perangkat lunak Arduino IDE	23
Gambar 6. Logo Platform Cayenne	30
Gambar 7. Contoh Tampilan Dashboard Platform Cayenne	31
Gambar 8. Diagram Blok Rangkaian	33
Gambar 9. Rangkaian Alat.....	35
Gambar 10. Blok Diagram Catu Daya	36
Gambar 11 Tiang Infus	37
Gambar 12. Controller box	39
Gambar 13. Sensor infus holder.....	39
Gambar 14. Flowchart program	40

DAFTAR LAMPIRAN

Lampiran 1. Gambar Alat	67
Lampiran 2. Skema Rangkaian	68
Lampiran 3. Program Alat	69
Lampiran 4. Data Sheet NodeMCU ESP8266-12E	72
Lampiran 5. Data Sheet Modul Sensor IR Obstacle Avoidance	80
Lampiran 6. Data Sheet Modul Step Down DC-DC LM2596	81

BAB I

Pendahuluan

A. Latar Belakang

Setiap pasien rawat inap yang ada di rumah sakit, poliklinik ataupun di puskesmas tidak sedikit yang memerlukan cairan infus. Cairan infus ini berada di dalam kantung plastik atau botol kaca yang khusus. Apabila cairan infus habis maka perawat harus 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 (Syahrul & Hidayat, 2009). Apabila terjadi masalah seperti penyumbatan atau kehabisan cairan jika tidak segera ditangani akan berbahaya bagi pasien, akibatnya dapat menyebabkan timbulnya komplikasi lain antara lain darah dari pasien dapat tersedot naik ke selang infus dan dapat membeku pada selang infus sehingga mengganggu kelancaran aliran cairan infus. 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 ke seluruh tubuh dan dapat menyumbat kapiler darah di paru sehingga menyebabkan emboli di paru (Waite, 2004). Infus yang ada saat ini masih banyak penggunaannya secara manual, dimana kesalahan-kesalahan masih sering terjadi.

Seringkali dalam suatu rumah sakit jumlah pasien tidak seimbang dengan jumlah petugas mediknya, khususnya pada bagian pelayanan keperawatan

yang bertugas 24 jam memantau kondisi pasien rawat inap satu per satu. Akibat keterbatasan itu kemungkinan kelalaian petugas jaga sangat bisa terjadi, terutama pada pemantauan kondisi cairan infus pasien. 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, oleh karena itu perlu solusi untuk mengatasi masalah yang terjadi dengan membuat alat yang dapat *memonitoring* pemakaian infus tersebut di rumah sakit, sehingga tugas perawat yang sedang berjaga dapat dimudahkan dengan hanya mengawasi monitor yang telah menghitung jumlah tetesan infus setiap pasien pada rumah sakit.

Menyikapi permasalahan tersebut, penulis mencoba membuat suatu alat yang diberi nama “Alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 dengan pemrograman Arduino IDE” Alat ini akan *memonitoring* 4 infus sekaligus dengan cara menghitung setiap tetesan infus yang ada pada botol infus dengan menggunakan modul sensor IR *obstacle avoidance*, kemudian hasil penghitungannya akan diproses menggunakan NodeMCU ESP8266-12E yang diprogram menggunakan arduino IDE lalu dikirimkan melalui jaringan internet menggunakan perangkat wifi sebagai pengirim data dari alat pemantau ke internet, sehingga pemantauan dapat dilakukan bebas di mana saja termasuk ke komputer yang berada pada ruangan perawat yang sedang berjaga. Peringatan kehabisan atau tersendatnya cairan infus dilihat melalui web menggunakan aplikasi Cayenne yang

memonitor ke 4 botol infus. Dalam hal ini diharapkan tugas perawat dapat dimudahkan karena sudah tidak perlu hilir mudik ke dalam ruang inap karena isi botol infus telah dipantau pada komputer dan sekaligus meningkatkan kualitas pelayanan di suatu rumah sakit.

B. Identifikasi Masalah

Dari uraian latar belakang masalah di atas, maka dapat dibuat identifikasi masalah sebagai berikut:

1. Pasien rawat inap sering tidak menyadari keadaan infus yang tersendat atau kehabisan cairan.
2. Masih banyak rumah sakit yang kekurangan tenaga kerja perawat serta cara kerja perawat yang kurang efisien dalam pengecekan cairan infus masih keluar masuk ke kamar pasien sehingga kelalaian perawat mungkin saja terjadi.
3. Bahayanya keterlambatan penanganan dalam penggantian infus dapat mengakibatkan berbagai macam komplikasi.
4. Belum adanya alat untuk mendeteksi tetesan infus menggunakan modul sensor IR *obstacle avoidance*, serta kontroler sekaligus pengiriman data untuk mendeteksi kehabisan dan kemacetan cairan infus menggunakan ESP8266.
5. Belum adanya program arduino IDE yang terintegrasi dengan web untuk memonitoring lebih dari 2 botol cairan infus.

C. Batasan Masalah

Dari banyaknya permasalahan yang diangkat dari identifikasi masalah, maka penulis membatasi ruang lingkup pembahasan tugas akhir ini pada masalah 4 dan 5. Pembuatan Proyek akhir ini akan menggunakan modul sensor IR *obstacle avoidance* untuk mendeteksi tetesan infus, NodeMCU ESP8266-12E untuk memproses data sekaligus mengirim ke internet. Hasil pembacaannya dapat dilihat pada web dengan aplikasi Cayenne yang akan memonitor 4 botol infus sekaligus serta memberi peringatan kehabisan dan tersendatnya cairan infus

D. Rumusan Masalah

Berdasarkan latar belakang masalah, identifikasi masalah, dan batasan masalah yang ada, maka dapat ditentukan rumusan masalah sebagai berikut:

1. Bagaimana merancang Alat *monitoring* tetesan infus menggunakan NodeMCU ESP8266-12E dengan modul sensor IR *obstacle avoidance*?
2. Bagaimana cara merancang program Alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 dengan pemrograman Arduino IDE?
3. Bagaimana unjuk kerja Alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 dengan pemrograman Arduino IDE?

E. Tujuan

Adapun tujuan dari pembuatan alat ini adalah:

1. Merealisasikan rancangan alat *monitoring* tetesan infus menggunakan NodeMCU ESP8266-12E dengan modul sensor IR *obstacle avoidance*.
2. Merealisasikan program Alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 dengan pemrograman Arduino IDE.
3. Mengetahui unjuk kerja Alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 dengan pemrograman Arduino IDE.

F. Manfaat

Pembuatan alat diharapkan dapat memberikan manfaat sebagai berikut:

1. Mahasiswa
 - a. Sebagai sarana implementasi pengetahuan yang didapatkan saat di bangku pendidikan.
 - b. Mampu merealisasikan teori yang didapatkan selama mengikuti perkuliahan.
 - c. Sebagai wujud kontribusi terhadap Universitas baik dalam citra maupun daya tawar terhadap masyarakat luas.
2. Dunia kesehatan
 - a. Dengan pemanfaatan alat *monitoring* infus nirkabel ini perawat pada rumah sakit tidak perlu bolak balik melakukan pengecekan infus pada kamar pasien.

- b. Berguna untuk meminimalisir pasien kehabisan atau tersendatnya cairan infus terutama pasien sedang tertidur.
3. Dunia usaha dan dunia industri:
- a. Dapat digunakan sebagai pengembangan produk elektronika yang dapat diaplikasikan sebagai *monitoring* tetesan infus pasien.
 - b. Berguna bagi rumah sakit yang ingin memakai alat *monitoring* infus ini sehingga dapat menambah efisien kerja para perawat.

G. Keaslian Gagasan

Berikut ini beberapa penelitian yang relevan, yang bisa dijadikan acuan untuk karya proyek akhir ini diantaranya sebagai berikut :

1. Perancangan Dan Pembuatan Alat *Monitoring* Cairan Infus Dengan Menggunakan Komunikasi Wireless Pada PC Berbasis Mikrokontroler ATmega16, Ardy Bernard Sinaga, Universitas Sumatera Utara, 2014. Karya ini kontrollernya menggunakan mikrokontroler ATmega16 yang hanya bisa mengontrol jalannya alat saja, sedangkan pengiriman data menggunakan radio frekuensi.
2. Sistem Monitoring infus Berbasis Mikrokontroler ATmega16, Raharjo Anton Dwi, 2014. Pada karya ini media penampil menggunakan LCD yang tampilannya menjadi kurang menarik serta membutuhkan kabel yang panjang untuk sampai ke ruang perawat.

Pembuatan Alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 dengan pemrograman Arduino IDE merupakan realisasi dan inovasi baru pengembangan dari kedua penelitian di atas. Penggunaan

NodeMCU ESP8266-12E ini dapat mengontrol sekaligus mengirim data sehingga biaya yang dikeluarkan lebih terjangkau, lalu Modul Sensor IR *Obstacle Avoidance* yang lebih sensitif dalam membaca setiap tetesan dan penggunaan web dengan tampilan yang lebih menarik serta dapat memantau 4 buah infus sekaligus dalam mengecek ketersediaan jumlah cairan infus tanpa menggunakan kabel.

BAB II

Pendekatan Pemecahan Masalah

A. Terapi Intravena (Infus)

Para ahli kesehatan menjabarkan pengertian atau definisi terapi intravena dalam berbagai bentuk dan macam rupa. Berikut ini merupakan pengertian terapi intravena menurut ahli sesuai dengan perspektifnya. Terapi Intravena adalah menempatkan cairan steril melalui jarum, langsung ke vena pasien. Biasanya cairan steril mengandung elektrolit (natrium, kalsium, kalium), nutrient (biasanya glukosa), vitamin atau obat (Smeltzer & Bare, 2002). Terapi intravena adalah pemberian sejumlah cairan ke dalam tubuh, melalui sebuah jarum, ke dalam pembuluh vena (pembuluh balik) untuk menggantikan kehilangan cairan atau zat-zat makanan dari tubuh (Darmadi, 2010). Terapi intravena digunakan untuk memberikan cairan ketika pasien tidak dapat menelan, tidak sadar, dehidrasi atau syok, untuk memberikan garam yang diperlukan untuk mempertahankan keseimbangan elektrolit, atau glukosa yang diperlukan untuk metabolisme dan memberikan medikasi (Perry & Potter, 2006).

Tujuan pemberian terapi intravena yaitu untuk memberikan atau menggantikan cairan tubuh yang mengandung air, elektrolit, vitamin, protein, lemak, dan kalori, yang tidak dapat dipertahankan secara adekuat melalui oral, memperbaiki keseimbangan asam basa, memperbaiki volume komponen-komponen darah, memberikan jalan masuk untuk pemberian obat-obatan ke dalam tubuh, memonitor tekanan vena sentral (CVP), memberikan

nutrisi pada saat sistem pencernaan mengalami gangguan (Perry & Potter, 2006). Peran perawat dalam memenuhi kebutuhan cairan dan elektrolit pada pasien amatlah penting, dan perawat harus memiliki pengetahuan terkait rumus kebutuhan cairan dan elektrolit dan rumus tetesan infus sehingga kebutuhan cairan diberikan sesuai. Pemenuhan kebutuhan cairan dan elektrolit diberikan oleh perawat harus sesuai dengan indikasi medis, dimana peran kolaboratif perawat sangat penting dalam penentuan jenis cairan dan jumlah kebutuhan cairan. Untuk itu perawat harus mengetahui jumlah kebutuhan cairan yang dibutuhkan oleh pasien yang didapat berdasarkan penilaian/pengkajian oleh perawat. Adapun hal-hal yang perlu diperhatikan dalam pemberian cairan dan elektrolit yaitu:

1. Kebutuhan air dan elektrolit pada bayi dan anak-anak menurut Mervyn

Maze (2004) adalah.

a. Kebutuhan air

0-10 kg = 4 ml/kg/jam (100 ml/kg/hari)

10-20 kg = 40 ml + 2 ml/kg/jam setiap kg di atas 10 kg (1000 ml + 50 ml/kg/hari setiap kg di atas 10 kg)

>20 kg = 60 ml + 1 ml/kg/jam setiap kg di atas 20 kg (1500 ml + 20 ml/kg/hari setiap kg di atas 20 kg)

b. Kebutuhan kalium 2,5 mEq/kg/hari

c. Kebutuhan natrium 2-4 mEq/kg/hari

2. Kebutuhan Air Dan Elektrolit Pada Orang Dewasa menurut Latief (2002) adalah.
 - a. Kebutuhan air sebanyak 30-35 ml/kg/hari.
 - b. Kebutuhan kalium 1-2 mEq/kg/hari.
 - c. Kebutuhan natrum 2-3 mEq/kg/hari.
3. Faktor yang mempengaruhi peningkatan kebutuhan cairan
 - a. Demam (kebutuhan meningkat 12% setiap 1°C , jika suhu $> 37^{\circ}\text{C}$).
 - b. Hiperventilasi.
 - c. Suhu lingkungan yang tinggi.
 - d. Aktivitas yang ekstrim/berlebihan.
 - e. Setiap kehilangan yang abnormal seperti diare atau poliuria.
4. Faktor yang mempengaruhi penurunan terhadap kebutuhan cairan
 - a. Hipotermi (kebutuhan menurun 12% setiap 1°C , jika suhu $< 37^{\circ}\text{C}$).
 - b. Kelembaban lingkungan yang sangat tinggi.
 - c. Oliguria atau anuria.
 - d. Hampir tidak ada aktivitas.
 - e. Retensi cairan misal gagal jantung.
5. Gangguan / masalah pemenuhan kebutuhan cairan
Dehidrasi adalah kekurangan cairan eksternal dapat terjadi karena penurunan asupan cairan dan kelebihan pengeluaran cairan. Dehidrasi dapat menyebabkan pengeluaran cairan 4-6 L (Dehidrasi Berat) atau kehilangan 2-4 L (dehidrasi sedang), mata cekung, turgor kulit buruk,

serum natrium 159-166 mEq/L (dehidrasi berat) dan serum natrium 152-158 mEq/L (untuk dehidrasi sedang).

Kehilangan cairan tubuh biasanya dapat diperbaiki dalam waktu 2 hari. Pada hari pertama diberikan setengah dari kebutuhan cairan melalui mulut, anus, dan per infus. Agar kebutuhan yang sudah dijadwalkan dapat tercapai. Bila seseorang kehilangan cairan tubuh cukup berat, sedangkan infus yang diberikan terlalu cepat maka hal ini hanya memicu terjadinya keracunan dan kejang. Sebab, sel-sel otak memiliki perubahan pada konsentrasi cairan tubuh yang lebih tinggi dari pada sel-sel dalam tubuh lainnya. Selain itu, pemberian cairan tubuh dengan cepat akan menyebabkan sembab pada sel-sel otak. Seorang tenaga kesehatan selain mengetahui kebutuhan cairan dan elektrolit juga wajib mengetahui cara menghitung tetesan infus dengan cepat dan tepat. Cara menghitung tetesan infus tidak boleh ditentukan asal-asalan, ada 2 macam infus set yang digunakan yaitu infus set makro yang biasa digunakan untuk memberikan terapi intravena pada pasien dewasa tetapi ada juga yang digunakan untuk anak-anak jika digunakan untuk terapi rehidrasi, lalu infus set mikro yang biasa digunakan untuk memberikan terapi intravena pada pasien anak-anak dan bayi, akan tetapi ada juga pasien dewasa yang menggunakan infus set mikro seperti pada pasien gagal ginjal kronis. Berikut cara mudah untuk menghitung tetesan infus per menit (TPM) secara sederhana yang dirumuskan oleh Puruhito (1995) adalah:

$$\text{Tetesan per menit (makro)} = \frac{\text{Jumlah cairan yang dimasukkan (ml)}}{\text{lamanya infus (jam)} \times 3}$$

$$\text{Tetesan per menit (mikro)} = \frac{\text{Jumlah cairan yang dimasukkan (ml)}}{\text{lamanya infus (jam)}}$$

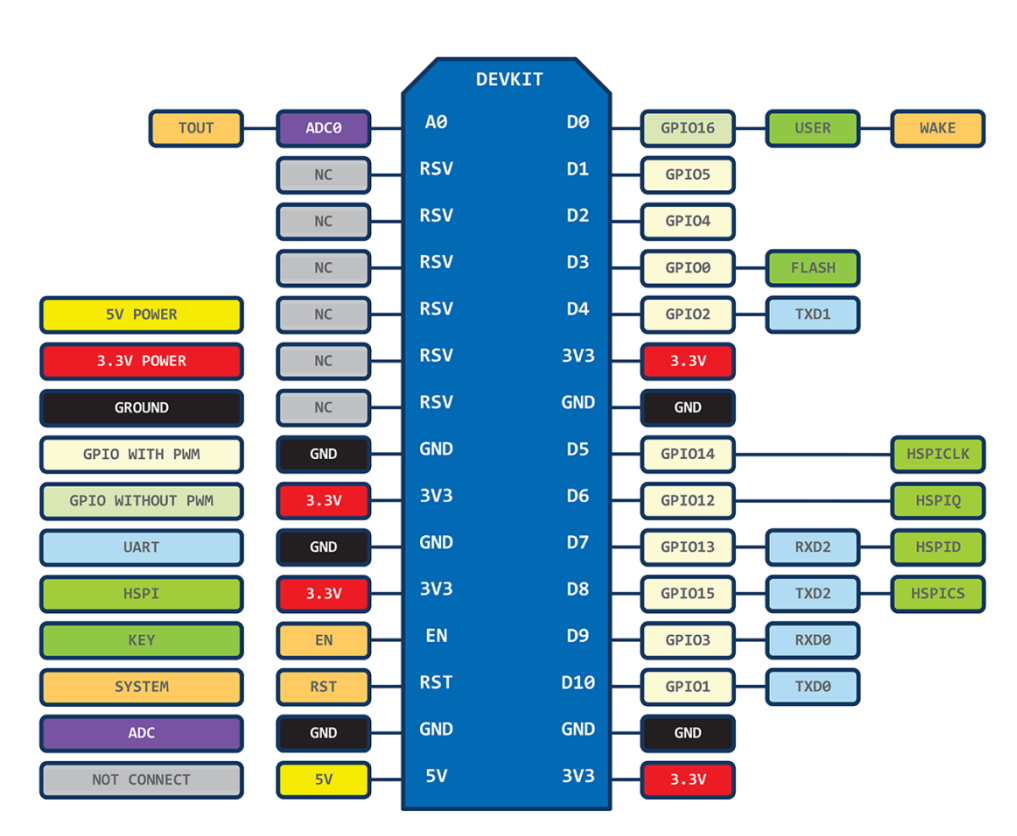
Menurut Darmadi (2010) beberapa komplikasi yang dapat terjadi dalam pemasangan infus: hematoma, yakni darah mengumpul dalam jaringan 21 tubuh akibat pecahnya pembuluh darah arteri vena, atau kapiler, terjadi akibat penekanan yang kurang tepat saat memasukkan jarum, atau “tusukan” berulang pada pembuluh darah. Infiltrasi, yakni masuknya cairan infus ke dalam jaringan sekitar (bukan pembuluh darah), terjadi akibat ujung jarum infus melewati pembuluh darah. Plebitis, atau bengkak (inflamasi) pada pembuluh vena, terjadi akibat infus yang dipasang tidak dipantau secara ketat dan benar Emboli udara, yakni masuknya udara ke dalam sirkulasi darah, terjadi akibat masuknya udara yang ada dalam cairan infus ke dalam pembuluh darah, rasa perih/sakit dan reaksi alergi. Keterlambatan perawat dalam menangani penggantian dan kemacetan cairan infus dapat memberikan dampak negatif terhadap pasien dengan terjadinya komplikasi seperti darah pasien tersedot naik ke selang infus dan dapat membeku pada selang infus, sehingga mengganggu kelancaran aliran infus. 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 ke seluruh tubuh dan dapat menyumbat kapiler darah di paru-paru sehingga menyebabkan emboli di paru-paru. Jika berbagai hal tersebut terjadi maka tempat pemasangan infus harus dipindahkan dan

dipasang ke pembuluh darah vena lain, yang tidak menutup kemungkinan dapat menyebabkan timbulnya berbagai komplikasi yang jauh lebih berbahaya akibat pemasangan yang tidak dilakukan dengan benar. Menganti botol infus dilakukan apabila cairan sudah berada di leher botol dan tetesan masih berjalan. Sebaiknya, prosedur ini dilakukan dalam 24 jam untuk mencegah flebitis dan pembentukan trombus.

B. NodeMCU ESP8266-12E

Nodemcu merupakan sebuah open source platform IoT dan pengembangan kit yang menggunakan bahasa pemrograman Lua untuk membantu pembuat dalam membuat produk IoT atau bisa dengan memakai sketch dengan arduino IDE. Nodemcu juga memiliki board yang berukuran sangat kecil yaitu panjang 4.83cm, lebar 2.54cm, dan dengan berat 7 gram, selain itu NodeMCU juga memiliki harga yang relatif terjangkau, tapi walaupun ukurannya yang kecil dan harganya yang terjangkau board ini sudah dilengkapi dengan fitur wifi dan firmwarena yang bersifat opensource. WiFi atau *Wireless Fidelity* merupakan seperangkat standar yang digunakan untuk komunikasi jaringan lokal tanpa kabel (WLAN) yang didasari pada spesifikasi IEEE 802.11 (Yuhefizar, 2008). WiFi menggunakan sinyal radio yang bekerja pada frekuensi tertentu sehingga dengan adanya WiFi ini semua perangkat yang terhubung bisa terkoneksi dengan internet. *Hotspot* merupakan sarana terkoneksi jaringan internet tanpa kabel dengan menggunakan standar *wireless* LAN, namun demikian dalam menjalankan *hotspot* diperlukan sarana lain, seperti Notebook/Laptop/PDA yang memiliki

fasilitas *wireless* LAN (Ida, 2010). Pengembangan Kit Nodemcu ini didasarkan pada modul ESP8266, yang mengintegrasikan GPIO, PWM (Pulse Width Modulation), IIC , 1-Wire dan ADC (Analog to Digital Converter) semua dalam satu board, pin diagram NodeMCU ESP8266 ditunjukkan pada gambar 1.



Gambar 1. Pin Diagram NodeMCU ESP8266

(github.com)

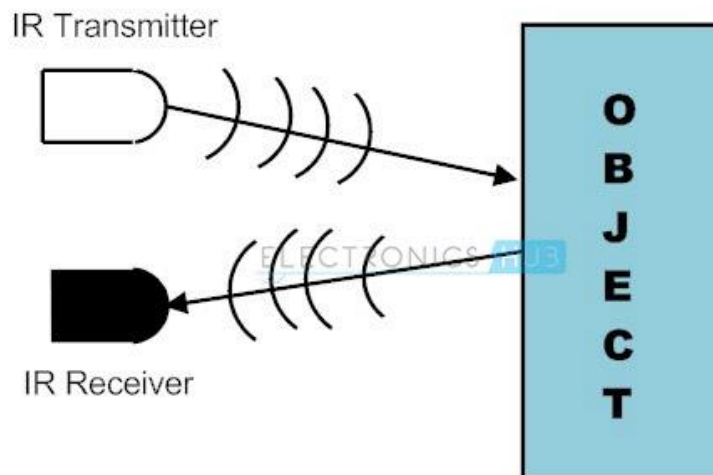
Adapun spesifikasi yang terdapat pada board ini yaitu:

1. *Board* ini berbasis ESP8266 Serial WiFi SoC (*single on Chip*) dengan *onboard* USB to TTL. Untuk *Wireless* standar yang digunakan adalah IEEE 802.11b/g/n.
2. *Tantalum capasitor* 100 *micro farad* dan yg kecil 10 *micro farad*.
3. 3.3 v LDO regulator.
4. *Cp2102 usb to UART bridge*.
5. Kemudian tombol *reset*, lalu port usb, dan terdapat tombol *flash*.
6. Terdapat 9 GPIO yang di dalamnya ada 3 pin PWM, 1 x ADC Channel, dan pin RX TX.
7. Pin seberangnya terdapat AD0 sebagai *analog sample*.
8. Pin *ground*.
9. S3 & s2 sebagai pin gpio.
10. S1 mosi(*Master Output Slave Input*) yaitu jalur data dari master dan masuk ke dalam *slave*, sc cmd/cs.
11. S0 miso(*Master Input Slave Input*) yaitu jalur data keluar dari *slave* dan masuk ke dalam master.
12. Sk yg merupakan SCLK dari master ke *slave* yang berfungsi sebagai *cloc*.
13. Pin Vin sebagai masukan tegangan.
14. GPIO dapat full kontrol lewat jaringan wifi.
15. GPIO dengan arus keluaran masing2 15mA dengan tegangan 3V.
16. Built in 32-bit MCU.

17. Board ini dapat di program langsung lewat USB, tanpa menggunakan rangkaian tambahan.
18. Pengembangan *Board* dengan *Open-Source Firmware* ini dapat dipergunakan untuk mengembangkan aplikasi IoT hanya dengan beberapa baris *script* Lua .

C. Modul Sensor IR *Obstacle Avoidance*

Modul Sensor IR *Obstacle Avoidance* merupakan sebuah modul yang terdiri dari chip comparator LM393 yang stabil serta inframerah dan photodiode yang berfungsi sebagai pendeteksi halangan atau objek di depannya. Sensor inframerah ini menggunakan prinsip pantulan cahaya infrared sebagai penentu nilai nya. Ketika modul sensor mendeteksi sebuah halangan atau object di depan sensor maka akan diperoleh pantulan cahaya dengan intensitas yang diatur sensitivitasnya. Nilai yang dihasilkan adalah high atau low, yang kemudian bisa digunakan oleh MCU untuk melakukan kontrol terhadap device lain. Sensor ini dapat mendeteksi objek berjarak 2 cm sampai 30 cm dengan sudut 35 °. Berikut cara kerja Modul Sensor IR *Obstacle Avoidance* pada gambar 2.

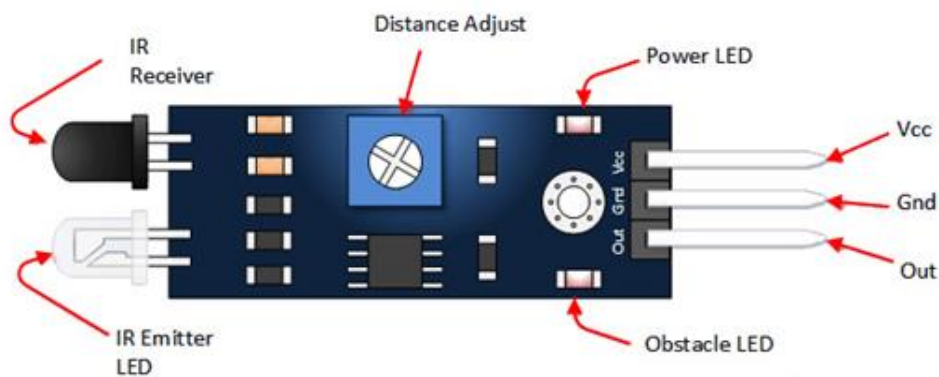


Gambar 2. Cara Kerja Sensor

(<http://www.electronicshub.org>)

berikut adalah cara kerja Modul Sensor IR *Obstacle Avoidance*:

1. Ketika *power-up*, IR *emitter* akan memancarkan cahaya *infrared* yang kasat mata.
2. Cahaya tersebut kemudian dipantulkan oleh objek yang ada di depannya, Cahaya terpantul ini kemudian diterima oleh IR *receiver*.
3. Terdapat Op-Amp LM363 yang berfungsi sebagai komparator antara resistansi IR *receiver* dan resistansi trimpot pengatur sensitivitas.
4. Saat terkena cahaya *infrared* pantulan objek tadi, resistansi IR *receiver* akan mengecil sehingga *output* Op-Amp menjadi *high/5V* dan menghidupkan LED sensor.
5. *Output* Op-Amp ini juga terhubung dengan pin “OUT” yang dihubungkan ke Arduino.



Gambar 3. konfigurasi Modul Sensor IR Obstacle Avoidance

<http://henrysbench.capnfatz.com>

Deskripsi

1. Vcc Tegangan kerja 3,3V sampai 5V

- | | |
|--------------------|--|
| 2. Gnd | <i>Ground Input</i> |
| 3. Out | <i>Output</i> |
| 4. Power LED | Menyala saat daya diterapkan |
| 5. Obstacle LED | Menyala saat hambatan terdeteksi |
| 6. Distance adjust | Menyesuaikan jarak deteksi. |
| 7. IR emitter | LED pemancar inframerah |
| 8. IR Receiver | Menerima sinyal yang ditransmisikan oleh pemancar inframerah |

D. Modul Step Down DC-DC LM2596

Modul stepdown lm2596 adalah modul yang memiliki IC LM2596 sebagai komponen utamanya. IC LM2596 adalah sirkuit terpadu / integrated circuit yang berfungsi sebagai Step-Down DC *converter* dengan *current* rating 3A. Terdapat beberapa varian dari IC seri ini yang dapat dikelompokkan dalam dua kelompok yaitu versi *adjustable* yang tegangan keluarannya dapat diatur, dan versi *fixed voltage output* yang tegangan keluarannya sudah tetap / *fixed*.

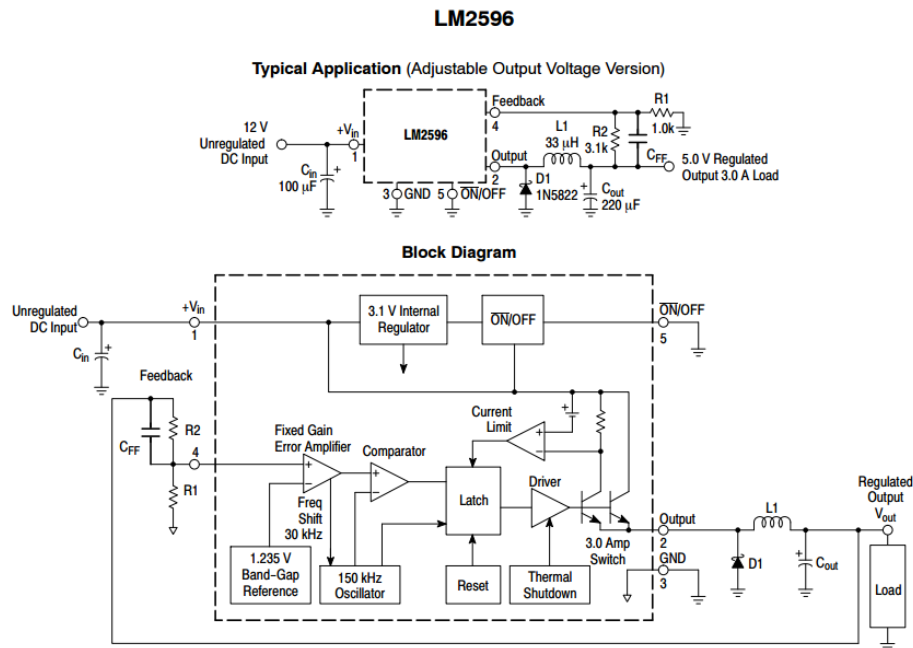
Modul ini memiliki spesifikasi :

1. Module Properties: non-isolated step-down module (buck)
2. Rectification: non-synchronous rectification
3. *Input voltage* :4.5-35V
4. *Output Voltage* :1.25-30V (adjustable)
5. *Output current*: rated current 2A, *Recommended less than* 2A,13W

6. *Efficiency*: Up to 92% (The higher the *output* voltage, the higher the *Efficiency*)
7. *Switching frequency*: 150KHz
8. *Minimum pressure*: 2V
9. *Operating Temperature: Industrial* (-40°C to +85°C) (*output power* dari 10W atau kurang)
10. *Full load temperature rise*: 40°C
11. *Load regulation*: $\pm 0.5\%$
12. *Voltage regulation*: $\pm 0.5\%$

Output Efficiency :

1. *Input* 12.00V 0.497A *Output* 4.953V 0.998A 4.941W *Efficiency* 82.85%
2. *Input* 12.00V 1.032A *Output* 4.943V 1.998A 9.878W *Efficiency* 79.76%
3. *Input* 12.00V 1.632A *Output* 4.921V 2.999A 14.76W *Efficiency* 75.37%
4. *Input* 23.99V 1.738A *Output* 11.96V 2.999A 35.88W *Efficiency* 86.05%



Gambar 4. Blok diagram Modul Step Down DC-DC LM2596

(www.onsemi.com)

E. Perangkat Lunak Arduino IDE

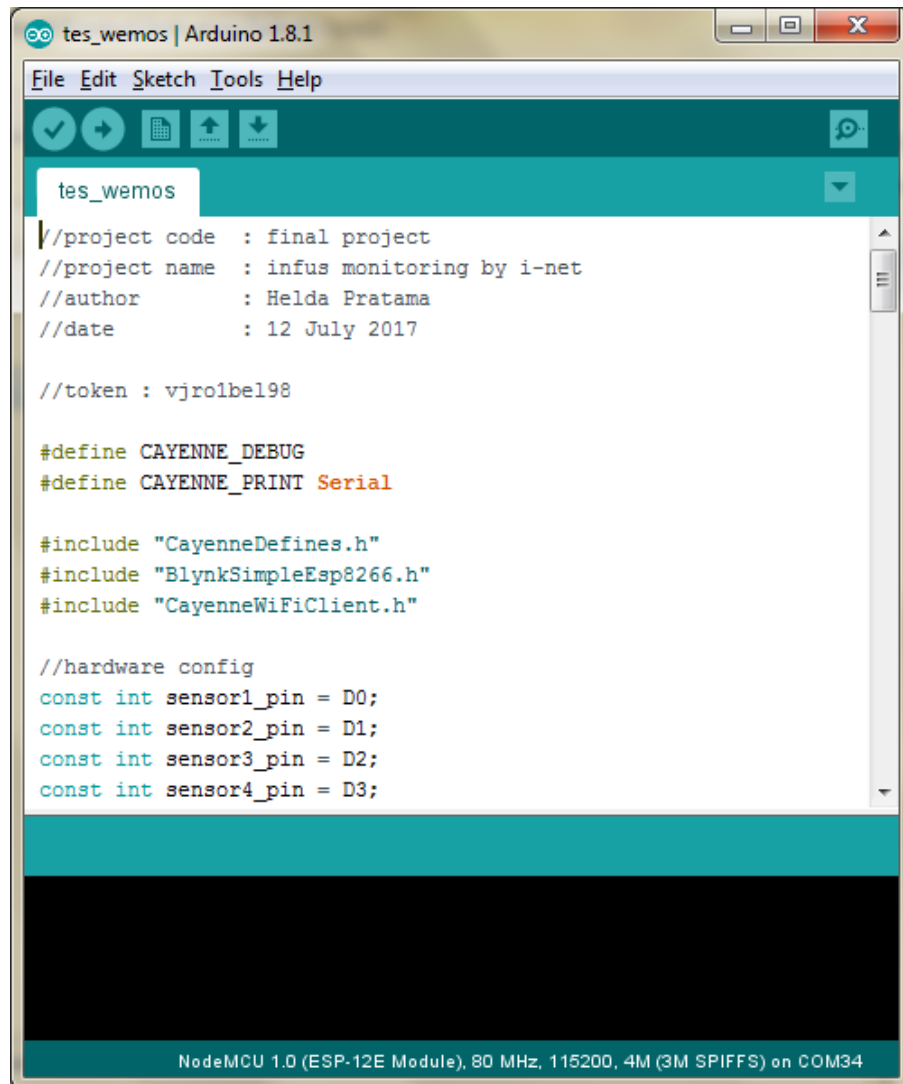
IDE itu merupakan kependekan dari *Integrated Development Environment*, atau secara bahasa mudahnya merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Disebut sebagai lingkungan karena melalui software inilah Arduino dilakukan pemrograman untuk melakukan fungsi-fungsi yang dibenamkan melalui sintaks pemrograman. Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa pemrograman Arduino (*Sketch*) sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. Sebelum dijual ke pasaran, IC mikrokontroler Arduino telah ditanamkan suatu program bernama

Bootlader yang berfungsi sebagai penengah antara *compiler* Arduino dengan mikrokontroler.

Arduino IDE dibuat dari bahasa pemrograman JAVA. Arduino IDE juga dilengkapi dengan library C/C++ yang biasa disebut *Wiring* yang membuat operasi input dan *output* menjadi lebih mudah. Arduino IDE ini dikembangkan dari software Processing yang dirombak menjadi Arduino IDE khusus untuk pemrograman dengan Arduino.

1. Menulis Sketch Program

Program yang ditulis dengan menggunakan Arduino Software (IDE) disebut sebagai sketch. Sketch ditulis dalam suatu editor teks dan disimpan dalam file dengan ekstensi .ino. Teks editor pada Arduino *Software* memiliki fitur-fitur seperti *cutting/paste* dan *searching/replacing* sehingga memudahkan kamu dalam menulis kode program. Software Arduino IDE terdapat semacam message box berwarna hitam yang berfungsi menampilkan status, seperti pesan eror, *compile*, dan *upload* program. Di bagian bawah paling kanan *software* Arduino IDE, menunjukan board yang terkonfigurasi beserta COM Ports yang digunakan.



Gambar 5. perangkat lunak Arduino IDE

(Sumber : Dokumentasi pribadi)

2. Menu menu yang ada pada sketch Arduino IDE:

a. Verify

berfungsi untuk melakukan checking kode yang kamu buat apakah sudah sesuai dengan kaidah pemrograman yang ada atau belum

b. Upload

Berfungsi untuk melakukan kompilasi program atau kode yang kamu buat menjadi bahasa yang dapat dipahami oleh mesin alias si Arduino.

c. New

berfungsi untuk membuat Sketch baru

d. Open

Berfungsi untuk membuka sketch yang pernah kamu buat dan membuka kembali untuk dilakukan editing atau sekedar upload ulang ke Arduino.

e. Save

Berfungsi untuk menyimpan Sketch yang telah kamu buat.

f. Serial Monitor

Berfungsi untuk membuka serial monitor. Serial monitor disini merupakan jendela yang menampilkan data apa saja yang dikirimkan atau dipertukarkan antara arduino dengan sketch pada port serialnya. Serial Monitor ini sangat berguna sekali ketika kamu ingin membuat program atau melakukan debugging tanpa menggunakan LCD pada Arduino. Serial monitor ini dapat digunakan untuk menampilkan nilai proses, nilai pembacaan, bahkan pesan error.

g. File

1) *New*, berfungsi untuk membuat membuat sketch baru dengan bare minimum yang terdiri void setup() dan void loop().

- 2) *Open*, berfungsi membuka sketch yang pernah dibuat di dalam drive.
- 3) *Open Recent*, merupakan menu yang berfungsi mempersingkat waktu pembukaan file atau sketch yang baru-baru ini sudah dibuat.
- 4) *Sketchbook*, berfungsi menunjukkan hirarki *sketch* yang kamu buat termasuk struktur foldernya.
- 5) *Example*, berisi contoh-contoh pemrograman yang disediakan pengembang Arduino, sehingga kamu dapat mempelajari program-program dari contoh yang diberikan.
- 6) *Close*, berfungsi menutup jendela Arduino IDE dan menghentikan aplikasi.
- 7) *Save*, berfungsi menyimpan *sketch* yang dibuat atau perubahan yang dilakukan pada *sketch*
- 8) *Save as...*, berfungsi menyimpan *sketch* yang sedang dikerjakan atau *sketch* yang sudah disimpan dengan nama yang berbeda.
- 9) *Page Setup*, berfungsi mengatur tampilan page pada proses pencetakan.
- 10) *Print*, berfungsi mengirimkan file sketch ke mesin cetak untuk dicetak.
- 11) *Preferences*, disini kam dapat merubah tampilan *interface* IDE Arduino.

- 12) *Quit*, berfungsi menutup semua jendela Arduino IDE. *Sketch* yang masih terbuka pada saat tombol *Quit* ditekan, secara otomatis akan terbuka pada saat Arduino IDE dijalankan.

F. Website

Website merupakan kumpulan halaman web yang saling terhubung dan filefilenya saling terkait. Web terdiri dari *page* atau halaman, dan kumpulan halaman yang dinamakan *homepage*. Homepage berada pada posisi teratas, dengan halamanhalaman terkait berada di bawahnya. Biasanya setiap halaman di bawah homepage disebut *child page*, yang berisi hyperlink ke halaman lain dalam web. (Gregorius, 2000, h:30). Penggunaan website memungkinkan untuk mengawasi cairan infus secara *Real-Time* sehingga langsung dapat mengetahui aktivitas yang sedang terjadi di situs atau aplikasi. Laporan diperbarui terus menerus sehingga jumlah tetesan infus dapat selalu terlihat dalam monitor serta kecepatan tetesan juga dapat diketahui.

Website awalnya merupakan suatu layanan sajian informasi yang menggunakan konsep hyperlink, yang memudahkan surfer atau pengguna internet melakukan penelusuran informasi di internet. Informasi yang disajikan dengan web menggunakan konsep multimedia, informasi dapat disajikan dengan menggunakan banyak media, seperti teks, gambar, animasi, suara, atau film. Berikut adalah jenis-jenis script pada website:

1. HTML

HTML (Hypertext Markup Language) merupakan suatu script dimana kita bisa menampilkan informasi dan daya kreasi kita melalui internet. HTML sendiri adalah suatu dokumen teks biasa yang mudah untuk dimengerti dibandingkan bahasa pemrograman lainnya, dan karena bentuknya itu maka HTML dapat dibaca oleh platform yang berlainan seperti windows, unix dan lainnya. (Sampurna, 1996, h:6). HTML merupakan bahasa pemrograman fleksibel dimana kita bisa meletakkan script dari bahasa pemrograman lainnya, seperti JAVA, VB, C, dan lainnya. Hypertext dalam HTML berarti bahwa kita dapat menuju ke suatu tempat, misal website atau halaman homepage lain, dengan cara memilih suatu link yang biasanya digaris bawahi atau diwakili oleh suatu gambar. Selain link ke website atau homepage halaman lain, hypertext ini juga mengizinkan kita untuk menuju ke salah satu bagian dalam satu teks itu sendiri. HTML tidak berdiri sendiri, agar ia dapat bertugas dalam membangun halaman web, ia harus ditulis dalam software atau aplikasi tertentu, yang dikenal sebagai HTML Editor. HTML Editor inilah yang bertugas untuk “menerjemahkan” bahasa HTML menjadi halaman web yang siap dilihat oleh para surfer di seluruh dunia.

2. Cascading Style Sheets (CSS)

CSS (*Cascading Style Sheets*) banyak digunakan untuk memperluas kemampuan HTML dalam memformat dokumen atau untuk mepercantik tampilan *web*, bahkan untuk pemosisian dan *layouting* halaman *web*.

Dengan mendefinisikan suatu *style* sekali saja itu akan dapat digunakan berulang kali. CSS telah didukung kebanyakan *browser*, terutama versi baru sehingga penempatan *layout* menjadi lebih fleksibel. Membuat HTML menggunakan *tag* minimal berpengaruh terhadap ukuran *file*, dapat menampilkan isi utama terlebih dahulu sementara gambar dan yang lain dapat ditampilkan sesudahnya. Selain itu penggunaan CSS pada *file* terpisah dapat mempermudah perubahan tampilan situs secara keseluruhan.

3. JavaScript

JavaScript mulanya bernama *LiveScript*, dikembangkan pertama kali tahun 1995 di *Netscape Communication*. Pada akhir tahun 1995 *Netscape Communication* dan *Sun Microsystem* berkolaborasi dan mengganti nama *LiveScript* menjadi *JavaScript*. *JavaScript* adalah bahan skrip yang ditempatkan pada kode HTML dan diproses pada sisi klien. Dengan adanya bahasa ini maka kemampuan dokumen HTML menjadi lebih luas. Sebagai contoh, digunakan untuk validasi masukan pada formulir sebelum diproses ke tahap selanjutnya. Bisa untuk membuat permainan interaktif dan juga bisa untuk menambah desain *web*.

4. PHP

Pada awalnya PHP merupakan singkatan dari *Personal Home Page tools*, yang berguna untuk memonitor pengunjung suatu *web*. PHP mula-mula dikembangkan oleh Rasmus Lerdorf. Istilah PHP kemudian mengacu pada *Hypertext Preprocessor*. PHP kemudian lebih dikembangkan untuk

membangun aplikasi *web* yang mendukung *database*, biasanya dipasangkan dengan MySQL. PHP merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di *server*. Hasilnya akan dikirimkan ke *client*, tempat pemakai menggunakan *browser*. Secara khusus, PHP dirancang untuk membentuk *web* dinamis, dimana dapat membentuk suatu tampilan berdasarkan keinginan dari si perancang *web*.

5. PHP Untuk MySQL

PHP telah menyediakan fasilitas koneksi untuk hampir semua program database populer baik yang komersial maupun yang gratis. MySQL adalah salah satu program database gratis yang cukup handal. MySQL merupakan *software* database yang termasuk paling populer di lingkungan linux, kepopuleran ini karena ditunjang performansi *query* dari databasenya yang bisa dikatakan paling cepat dan jarang bermasalah. Namun MySQL telah tersedia juga di lingkungan Windows. PHP untuk Windows secara default telah mendukung MySQL.

G. Cayenne

Cayenne merupakan platform pengembangan dengan system *drag-and-drop* milik myDevices yang menyediakan akses untuk fitur-fitur Arduino saat akan digunakan menjadi *board Internet of Things (IoT)*. Saat ini para penggiat IoT di seluruh dunia dapat memanfaatkan berbagai macam *shield* untuk di gunakan sebagai platform IoT. Platform ini menyediakan pengaturan cukup mudah, termasuk di dalamnya Wi-fi, BLE, IR, NFC, dan lain sebagainya.

Platform yang dibangun dengan tujuan mempermudah pembangunan ekosistem IoT ini, menyediakan set alat yang dapat digunakan untuk memvisualisasikan data yang di peroleh dari sensor, maupun mengendalikan aktuator yang terhubung dengan layanan, melalui *web dashboard* ataupun aplikasi *mobile* berbasis internet. Logo platform Cayenne ditunjukkan seperti pada gambar 6.

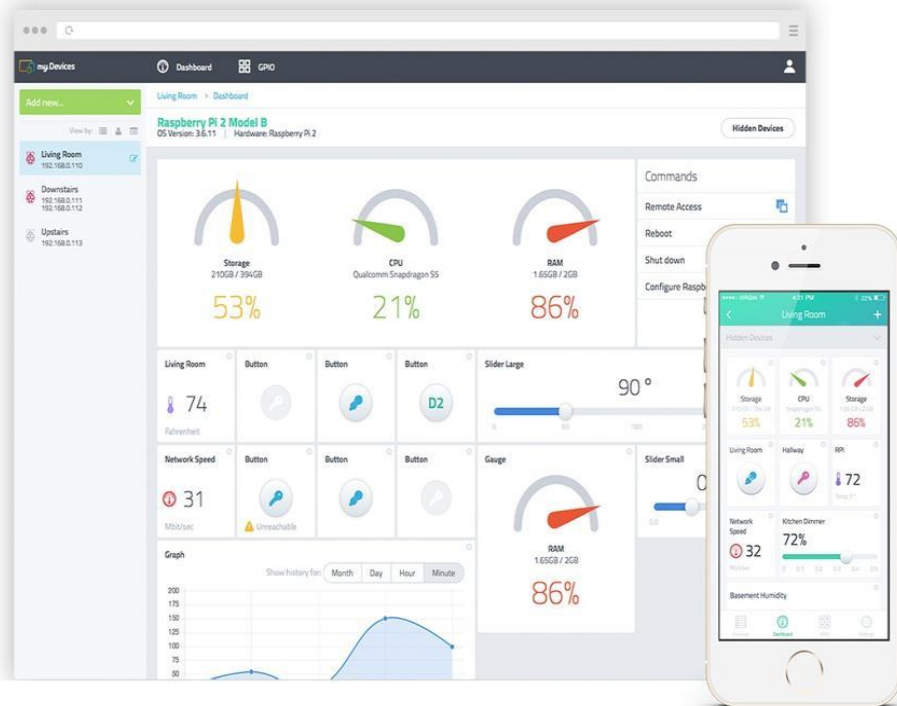


Gambar 6. Logo Platform Cayenne

Cayenne menawarkan layanan *cloud* yang dapat terkoneksi dengan berbagai macam jenis Arduino dengan variasi *shield*-nya yang sangat banyak. Selain itu, sebagai fitur unggulan, Cayenne menawarkan kemudahan dalam mengatur, melakukan konfigurasi, dan integrasi dengan hanya menggunakan metode *drag-and-drop* untuk papan pengelolanya.

Platform Iot *project builder* ini juga benar-benar konsen dalam membuat integrasi sistemnya dengan IoT. Hal ini dapat kita lihat dari penyediaan berbagai macam *sketch* untuk Arduino IDE yang dapat digunakan dengan mudah dan aman dalam menghubungkan platform tersebut dengan perangkat Arduino. Platform ini digunakan untuk membangun peringatan dan pemicu antara dua platform *board* yang akan digunakan. Platform pengembangan ekosistem IoT ini juga memberikan kemudahan untuk membuat *widget* baru untuk sensor ataupun aktuator yang terkoneksi dengan

layanan. Contoh tampilan *dashboard* platform Cayenne ditunjukkan seperti pada Gambar 7.



Gambar 7. Contoh Tampilan Dashboard Platform Cayenne

BAB III

Konsep Rancangan

A. Identifikasi Kebutuhan

Untuk merancang suatu system Alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 ini harus diperhatikan berbagai macam kebutuhan komponen yaitu:

1. Dibutuhkannya sensor pendeteksi tetesan infus.
2. Dibutuhkannya mikrokontroler untuk mengendalikan rangkaian dari sistem.
3. Dibutuhkannya komunikasi untuk mengirimkan data ke komputer.
4. Dibutuhkannya *software* untuk memprogram mikrokontroler.
5. Dibutuhkannya *power supply* untuk mendukung kerja sistem tersebut.
6. Dibutuhkan rangkaian penurun tegangan dari DC 12V ke DC 3,3V.
7. Dibutuhkannya website untuk menampilkan jumlah tetesan infus.

B. Analisis Kebutuhan

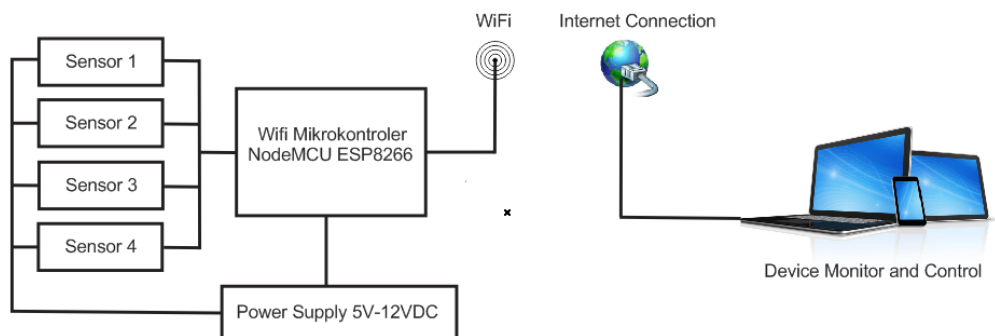
Berdasarkan identifikasi kebutuhan di atas, maka diperoleh beberapa analisis kebutuhan terhadap sistem yang akan dirancang adalah sebagai berikut:

1. Modul Sensor IR *Obstacle Avoidance* sebagai pendeteksi infus.
2. NodeMCU ESP8266-12E untuk mengendalikan rangkaian serta mengirimkan data ke komputer.
3. *Software* Arduino IDE sebagai aplikasi pemrogram mikrokontroler.

4. Menggunakan *Regulator Power Supply* dengan *output* DC +5V dan +12V dari *input* 220V.
5. Menggunakan Modul *Step Down* DC-DC LM2596 sebagai rangkaian penurun tegangan dari DC 12V ke DC 3,3V.
6. Menggunakan website dengan aplikasi Cayenne untuk menampilkan jumlah tetesan infus.

C. Blok Diagram Rangkaian

Diagram blok merupakan gambaran dasar dari rangkaian system yang akan dirancang. Setiap diagram blok memiliki fungsi masing masing. Adapun diagram blok rangkaian yang dirancang adalah seperti pada gambar 8 :



Gambar 8. Diagram Blok Rangkaian

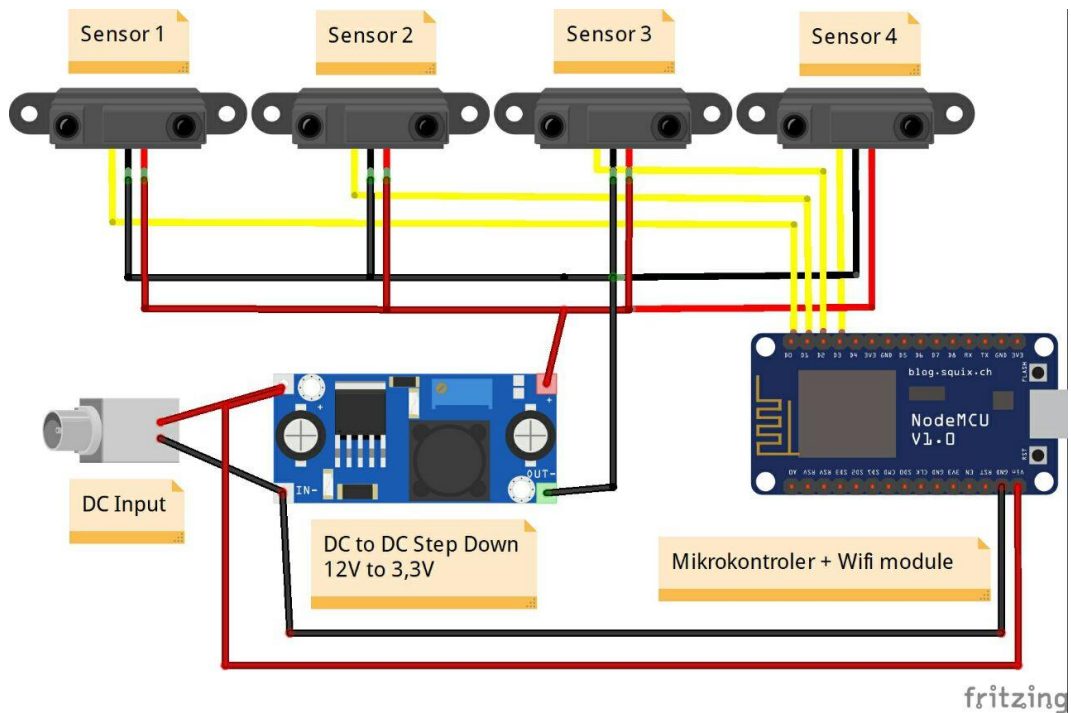
Gambar diagram alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 ini memperlihatkan bagian-bagian *hardware* yang digunakan dalam rangkaian monitoring penggunaan infus, juga menjelaskan aliran proses system kerja rangkaian dari input hingga *output*. Terdapat beberapa bagian dari diagram blok di atas antara lain. Sensor, mikrokontroler

Modul Sensor IR *Obstacle Avoidance*, NodeMCU ESP8266-12E, dan web. Input yang dideteksi oleh sensor yaitu untuk mendeteksi setiap tetes dari cairan infus. Prinsip kerjanya dengan menggunakan pasangan pemancar dan penerima inframerah (IR / *Infrared*), dimana penerima akan mendeteksi objek di depannya yang memantulkan cahaya yang dipancarkan oleh pemancar inframerah, keluaran dari modul ini berupa sinyal digital yang akan bernilai *low (active low)* pada saat mendeteksi halangan / ada objek di depannya. Unit pengolah data terdiri atas mikrokontroler NodeMCU ESP8266 yang dilengkapi wifi. Unit pengolah data berfungsi sebagai pemroses data, memberikan intruksi menghitung jumlah tetesan infus dan mendeteksi kemacetan cairan infus lalu mengirimkan data ke server. Unit *output* menggunakan aplikasi Cayenne pada desktop diuji dengan cara membuka browser yang sudah terkoneksi dengan internet.

D. Perancangan Sistem

1. Rangkaian NodeMCU ESP8266-12E

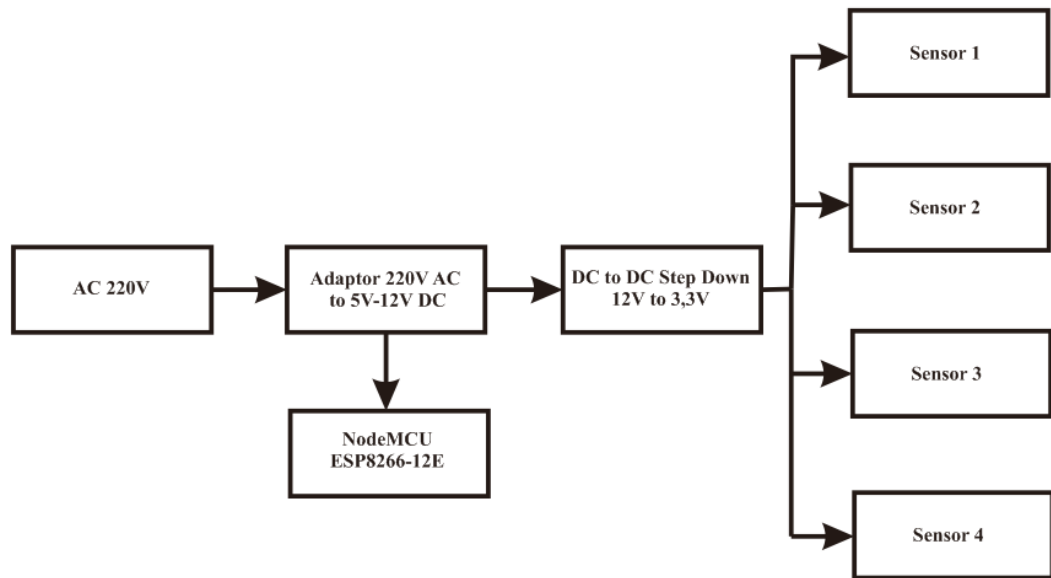
Rangkaian NodeMCU ESP8266-12E ini adalah sebuah otak dan sistem kendali rangkaian alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 ini menggunakan 4 port I/O yang semuanya digunakan sebagai masukan data ke sensor inframerah. Port yang digunakan untuk masukan data ke sensor yaitu D0, D1, D2, dan D3, sedangkan suplai daya didapat langsung dari *step down*. Berikut ini rangkaian NodeMCU ESP8266-12E pada gambar 9.



Gambar 9. Rangkaian Alat

2. Blok Rangkaian Catu Daya

Supply utama yang digunakan pada rangkaian alat ini menggunakan Catu daya 12V dan 5V, tegangan 12V dialirkan ke *Step Down DC to DC* menjadi 3,3V yang kemudian dialirkan kembali ke 4 sensor inframerah. NodeMCU ESP8266-12E mendapatkan *supply* tegangan sebesar 5V. Blok diagram catu daya dapat dilihat pada gambar 10.

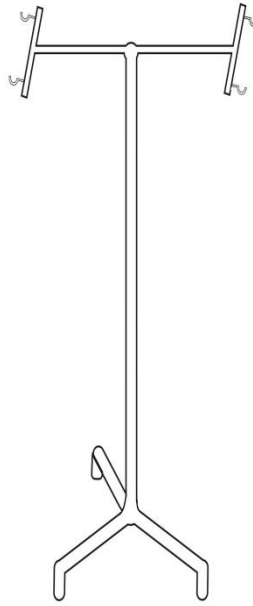


Gambar 10. Blok Diagram Catu Daya

E. Langkah Pembuatan

1. Pembuatan tiang infus untuk Alat Monitoring Tetesan Infus.

Pembuatan tiang infus ini menggunakan besi agar kuat untuk menyangga 4 infus, dibagian atas terdapat gantungan untuk meletakkan kantung infus dan dibagian tengah atas digunakan sebagai tempat komponen, agar tiang infus dapat berdiri maka dibagian bawah menggunakan 3 kaki besi yang telah dilas. Berikut desain tiang infus pada gambar 11.



Gambar 11. Tiang Infus

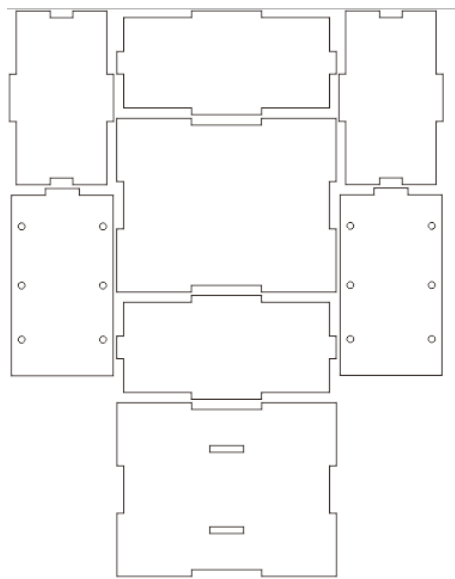
Langkah-langkah yang harus dilakukan dalam pembuatan tiang infus.

- a. Menyiapkan bahan dan alat yang dibutuhkan, bahan pada tiang infus ini menggunakan besi dengan diameter 3cm dan 2cm serta menggunakan cat semprot berwarna putih.
- b. Memotong besi sesuai dengan ukuran yang dibutuhkan.
- c. Mengelas besi yang akan disambung.
 - 1) Membuat coakan pada bagian yang akan dilas. Boleh salah satu dan bisa juga dua-duanya, sehingga setelah nanti di satukan akan ada tempat untuk nanti cairan elektroda. Sehingga proses pengelasan benar-benar matang.

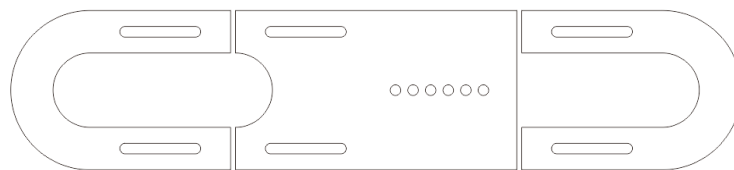
- 2) Meletakkan massa mesin las pada salah satu bagian bahan yang akan dilas. Memasukkan elektroda pada panel penjepit elektroda di mesin las.
 - 3) Setelah bahan siap untuk di las, mendekatkan ujung elektroda pada bahan yang akan dilas secara perlahan.
 - 4) Memutar perlahan tang elektroda jika area yang dilas cukup luas hingga cairan elektroda menutup rapat permukaan bagian yang akan dilas.
 - 5) Hasil yang baik saat proses pengelasan dapat dilihat saat permukaan yang dilas berbentuk seperti gelombang rapat dan teratur menutup sempurna bagian yang dilas.
 - 6) Membersihkan kerak yang menutupi bagian yang dilas dengan menggunakan palu setelah selesai lalu periksa kembali apakah terdapat bagian yang belum sempurna.
- d. Mengecat tiang infus secara merata dengan cat semprot setelah tahap pengelasan selesai.
2. Membuat box Alat Monitoring Tetesan Infus dan sensor *holder*
- Langkah pembuatannya sama saja yaitu:
- a. Hal yang pertama dilakukan yaitu mengukur kebutuhan ukuran dari kedua buah box.
 - b. Langkah selanjutnya adalah mendesain *box* dengan *software* CorelDraw seperti pada gambar 12 dan gambar 13, pada gambar *controller box* akan digunakan untuk meletakkan NodeMCU ESP8266, *step down*, dan

catu daya. Gambar 13 digunakan untuk meletakkan modul sensor IR *obstacle avoidance* pada bagian *drip chamber*.

- c. Setelah selesai mendesain *box*, *cutting akrilik* menggunakan mesin laser.
- d. Setelah itu, melakukan penyambungan kabel-kabel instalasi kemudian merakit komponen yang telah siap untuk dipasang ke dalam box.
- e. Melakukan uji coba pengetesan Alat Monitoring Tetesan Infus.



Gambar 12. *Controller box*

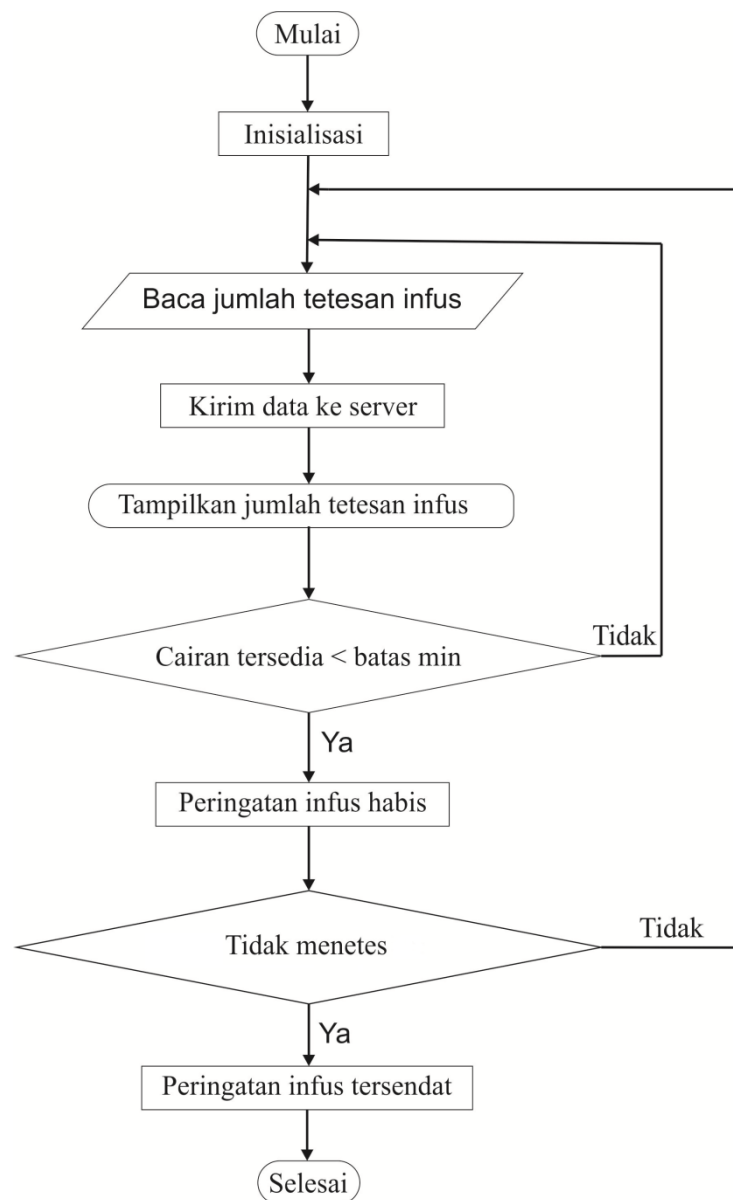


Gambar 13. *Sensor infus holder*

F. Perangkat Lunak

Software pada alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 dibangun menggunakan dua jenis *software*.

Software yang pertama adalah *software* untuk memprogram mikrokontroler dan melakukan *compile* ke perangkat NodeMCU ESP8266 menggunakan Arduino IDE. *Software* yang kedua adalah *software* untuk membuat *server* pada internet menggunakan Cayenne myDevices. Algoritma pemrograman yang akan ditanamkan pada perangkat mikrokontroler ditunjukkan seperti pada Gambar 14.



Gambar 14. *Flowchart* program

G. Spesifikasi Alat

Alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 memiliki spesifikasi sebagai berikut.

1. Dapat menghitung tetesan infus sebanyak 500ml.
2. Dapat mendeteksi jika infus hampir habis.
3. Dapat mendeteksi jika ada kemacetan infus.
4. Menggunakan mikrokontroler NodeMCU ESP8266-12E.
5. Menggunakan listrik 220V sebagai *inputan* daya.
6. Hasil tampilan dilihat secara *online* dengan membuka aplikasi Cayenne.

H. Rencana Pengujian

Pengujian alat dilakukan untuk mendapatkan data penelitian dengan menggunakan dua pengujian, yaitu:

1. Uji fungsional

Pengujian ini dilakukan dengan cara menguji setiap bagian alat berdasarkan karakteristik dan fungsi masing-masing bagian perangkat sesuai diagram blok. Pengujian ini dilakukan untuk mengetahui apakah setiap bagian dari perangkat telah bekerja sesuai dengan fungsi dan keinginan. Bagian-bagian yang diuji antara lain bagian NodeMCU ESP8266-12E, rangkaian sensor inframerah, tampilan aplikasi cayenne, dan rangkaian catu daya. Dari pengujian ini akan diketahui kinerja dari alat yang dibuat.

a. NodeMCU ESP8266-12E

Pengujian mikrokontroler dilakukan untuk mengetahui kinerja mikrokontroler sebagai pengolah data dapat bekerja dengan baik.

Pengujian dilakukan dengan menggunakan komputer atau laptop dengan cara menghubungkan port USB pada mikrokontroler ke *port* USB komputer atau laptop. Pengujian ini untuk mengetahui apakah *chip* mikrokontroler dapat berfungsi dengan baik saat dimasuki (di-*download*) file .hex hasil *compile* dari program Arduino IDE

b. Sensor inframerah

Pengujian sensor dilakukan untuk mengetahui kinerja sensor Modul Sensor IR *Obstacle Avoidance* dalam mendeteksi tetesan infus. Pengujian dilakukan dengan cara mengamati hasil pembacaan setiap tetesan infus oleh sensor Modul Sensor IR *Obstacle Avoidance*. Hasil pembacaan sensor ditampilkan pada layar komputer yang telah tersambung ke jaringan internet serta dapat membuka aplikasi Cayenne. Hasil pembacaan ini kemudian dibandingkan dengan jumlah tetesan sebenarnya. Nilai ini didapatkan dari hasil perhitungan dengan cara manual.

c. Aplikasi Cayenne

Aplikasi Cayenne pada desktop diuji dengan cara membuka browser Google Chrome yang sudah terkoneksi dengan internet. Sebelum membuka browser pada desktop maupun *smartphone*, kita pastikan bahwa perangkat sudah diaktifkan dan terkoneksi dengan internet melalui sambungan wifi. Setelah browser terbuka, kemudian kita masukkan alamat www.cayenne-mydevices.com pada browser.

Jika sudah terhubung, kita melakukan *login* dengan akun yang sudah didaftarkan.

d. Catu Daya

Pengujian catu daya dilakukan untuk memastikan tegangan keluaranya tidak lebih dari ambang batas tegangan yang diminta komponen. Pengujian catu daya dilakukan dengan cara mengukur tegangan *output* yang dihasilkan oleh catu daya. Pengujian catu daya dilakukan dengan dua metode, yaitu pengujian tanpa beban dan pengujian ada beban.

2. Uji unjuk kerja

Pengujian ini dilakukan untuk mengetahui kinerja sistem secara keseluruhan. Pada pengujian ini akan mengamati jumlah tetesan infus 500 ml hingga habis dan menguji kerja batas infus untuk memberi peringatan kehabisan infus

I. Tabel Uji

1. Tabel Uji Fungsional

a. Pengujian NodeMCU ESP8266-12E

Tabel 1. Rencana Pengujian NodeMCU ESP8266-12E

No	Pengujian	Gambar	Keterangan
1	Serial COM		
2	<i>Chip</i>		

b. Pengujian Sensor Inframerah

Tabel 2. Rencana Pengujian Sensor Inframerah

No	Pembacaan sensor	Jumlah cairan infus (tetes)				Keterangan
		1 menit	10 menit	30 menit	60 menit	
1	Sensor 1					
2	Sensor 2					
3	Sensor 3					
4	Sensor 4					

c. Pengujian Aplikasi Cayenne

Tabel 3. Rencana Pengujian Aplikasi Cayenne

No	Pengujian	Hasil	Keterangan
1	Tampilan Utama		
2	Peringatan Kehabisan Infus		
3	Peringatan Kemacetan Infus		

d. Tabel Pengujian Catu Daya

1) Pengujian Tanpa Beban

Tabel 4. Rencana Pengujian Catu Daya Tanpa Beban

No	Pengukuran Pada	Jumlah	V Out	V Out Terbaca	Kesalahan	Keterangan
1	Nodemcu ESP8266-12E	1				
2	Sensor Inframerah	1				
		2				
		3				
		4				

2) Pengujian Dengan Beban

Tabel 5. Rencana Pengujian Catu Daya Dengan Beban

No	Pengukuran Pada	Jumlah	V Out	V Out Terbaca	Kesalahan	Keterangan
1	Nodemcu ESP8266-12E	1				
2	Sensor Inframerah	1				
		2				
		3				
		4				

2. Tabel Uji Unjuk Kerja

a. Tabel Pengujian Jumlah Tetesan

Tabel 6. Rencana Pengujian Jumlah Tetesan

No	Jumlah tetes permenit (tpm)	Kantong Infus (Tetes)				Waktu
		1	2	3	4	
1	20 tpm					
2	30 tpm					
3	40 tpm					
4	50 tpm					
5	60 tpm					
6	Rata-rata					

b. Uji Kerja Batas Infus

Tabel 7. Rencana Uji Kerja Batas Infus

NO	Kapasitas Infus (%)				Peringatan Kehabisan Infus			
	30 tpm	40 tpm	50 tpm	60 tpm	1	2	3	4
1	100%	100%	100%	100%				
2	90 %	90 %	90 %	90 %				
3	80 %	80 %	80 %	80 %				
4	70 %	70 %	70 %	70 %				
5	60 %	60 %	60 %	60 %				
6	50 %	50 %	50 %	50 %				
7	40 %	40 %	40 %	40 %				
8	30 %	30 %	30 %	30 %				
9	20 %	20 %	20 %	20 %				
10	10 %	10 %	10 %	10 %				
11	0 %	0 %	0 %	0 %				

J. Pengoperasian Alat

Pengoprasian alat ini dapat dilakukan dengan cara sebagai berikut.

1. Pastikan alat terhubung dengan tegangan listrik 220 V.
2. Pasang botol infus pada tiang yang telah tersedia.
3. Pasang infus set dengan cara melubangi botol infus ditempat yang sudah tersedia untuk jalannya aliran tetesan infus.
4. Pasang sensor infus pada tabung infus set, pastikan jarak sensor sesuai dengan keluarnya tetesan infus
5. Buka aplikasi Cayenne dengan cara memasukkan alamat *www.cayenne-mydevices.com* pada browser, lalu login dengan akun yang telah didaftarkan.
6. Pasang jarum infus pada vena pasien yang telah dipilih, pemasangan ini harus sesuai dengan SOP pemasangan infus.

7. Atur kecepatan tetesan infus dengan pengatur tetesan yang ada di infus set.
8. Perawat dapat langsung memantau keadaan infus pasien pada monitor, pastikan kondisi internet baik dan stabil.

BAB IV

Pengujian dan Pembahasan

A. Pengujian

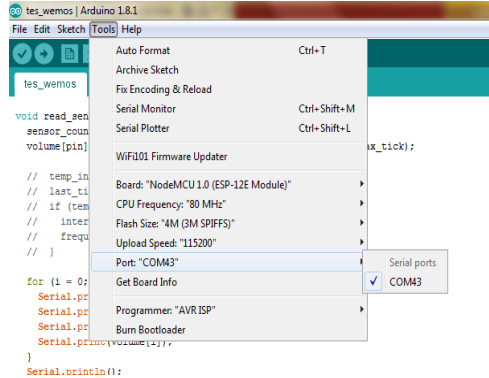

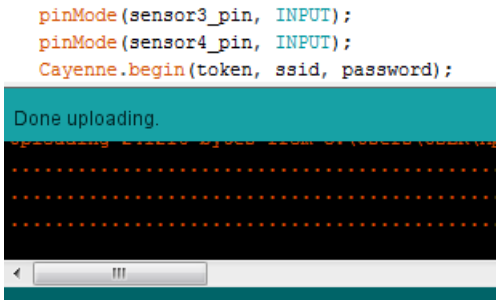
Pengujian terhadap Alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 dengan pemrograman Arduino IDE dilakukan untuk mengetahui kinerja masing masing komponen dan keseluruhan. Hasil dari pengujian alat dan pengambilan data tersebut diharapkan mampu mendapatkan data yang *valid* dan alat bekerja sesuai dengan fungsi dan tujuannya.

1. Tabel Uji Fungsional

a. Pengujian NodeMCU ESP8266-12E

Pengujian NodeMCU ESP8266-12E bertujuan untuk mengetahui program yang dibuat sudah dapat berjalan dengan baik. Pengujian ini dilakukan dengan cara memasukkan program pada aplikasi Arduino IDE ke NodeMCU ESP8266-12E.

Tabel 8. Hasil Pengujian NodeMCU ESP8266-12E

No	Pengujian	Gambar	Keterangan
1	Serial COM		<p>Arduino IDE terkoneksi dengan perangkat atau board NodeMCU pada port COM43</p>
	Serial COM		<p>board NodeMCU terhubung dengan komputer pada port COM43</p>
2	Chip		<p>Chip mikrokontroler dalam kondisi baik karena program dapat diupload ke mikrokontroler. Ditandai dengan “done uploading” pada arduino IDE.</p>

b. Pengujian Sensor Inframerah

Pengujian inframerah dilakukan dengan meneteskan cairan infus dengan kecepatan yang berbeda pada masing-masing sensor dari yang berkecepatan rendah, sedang, dan kecepatan tinggi. Kecepatan infus akan disesuaikan dengan kebutuhan cairan pasien dirumah sakit. Sensor dapat mendeteksi tetesan infus dapat diketahui dengan melihat led indikator di sensor serta menyesuaikan di aplikasi Cayenne.

Tabel 9. Hasil Pengujian Sensor Inframerah

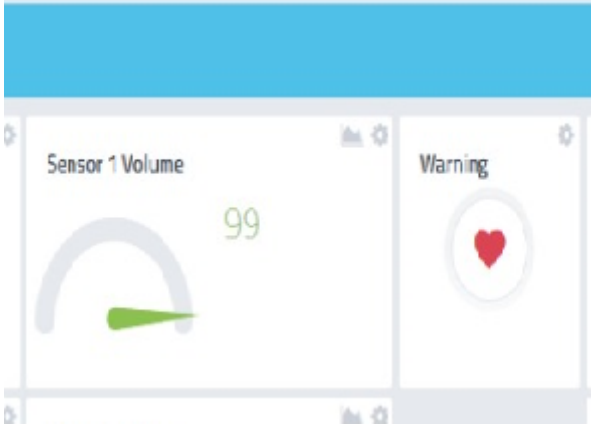
No	Pembacaan sensor	Jumlah cairan infus (tetes)				Sesuai dengan kebutuhan pasien
		1 menit	10 menit	30 menit	60 menit	
1	Sensor 1	14	141	420	834	Anak dengan berat badan 10 kg
2	Sensor 2	21	211	632	1250	Anak dengan berat badan 20 kg
3	Sensor 3	17	172	514	1000	Dewasa dengan berat badan 40 kg
4	Sensor 4	30	305	515	1750	Dewasa dengan berat badan 70 kg

c. Pengujian Aplikasi Cayenne

Pengujian aplikasi Cayenne ini bertujuan untuk mengetahui apa yang akan ditampilkan pada saat infus berjalan, infus habis, dan infus tersendat.

Tabel 10. Hasil Pengujian Aplikasi Cayenne

No	Pengujian	Hasil	Keterangan
1	Tampilan Utama		Terdapat 4 indikator dari masing-masing infus yang menunjukkan ketersediaan infus
2	Peringatan Kehabisan Infus		Infus dengan jarum berwarna merah menunjukkan bahwa cairan infus akan segera habis

3	Peringatan Kemacetan Infus		Hati berwarna merah menandakan bahwa infus tersendat.
---	----------------------------------	--	--

d. Tabel Pengujian Catu Daya

Pengujian catu daya dilakukan untuk memastikan tegangan yang keluar tidak melebihi tegangan yang dibutuhkan komponen. Pengujian catu daya dilakukan dengan cara mengukur tegangan *output* yang dihasilkan oleh catu daya. Pengukuran dilakukan pada mikrokontroler NodeMCU dan 4 sensor inframerah dengan tujuan untuk memastikan hasil keluaran tetap stabil pada angka yang dibutuhkan oleh masing masing komponen.

1) Pengujian Tanpa Beban

Tabel 11. Hasil Pengujian Catu Daya Tanpa Beban

No	Pengukuran Pada	Perangkat ke-	V Out	V Out Terbaca	Kesalahan	Keterangan
1	NodeMCU ESP8266-12E	1	12	11,70	2,5 %	Selisih 0,30 v
2	Sensor Inframerah	1	5	4,9	2 %	Selisih 0,1 v
		2	5	4,9	2 %	Selisih 0,1 v
		3	5	4,9	2 %	Selisih 0,1 v
		4	5	4,9	2 %	Selisih 0,1 v

2) Pengujian Dengan Beban

Tabel 12. Hasil Pengujian Catu Daya Dengan Beban

No	Pengukuran Pada	Perangkat ke-	V Out	V Out Terbaca	Kesalahan	Keterangan
1	NodeMCU ESP8266-12E	1	12	11,9	0,83 %	Selisih 0,1 v
2	Sensor Inframerah	1	5	4,99	0,2 %	Selisih 0,01 v
		2	5	4,99	0,2 %	Selisih 0,01 v
		3	5	4,99	0,2 %	Selisih 0,01 v
		4	5	4,99	0,2 %	Selisih 0,01 v

2. Tabel Uji Unjuk Kerja

a. Tabel Pengujian Jumlah Tetesan

Pengujian jumlah tetesan ini berfungsi untuk mengetahui jumlah maksimal tetesan pada botol infus 500ml. Pada pengujian ini menggunakan kecepatan infus yang berbeda-beda sehingga dapat diketahui waktu yang ditempuh untuk menghabiskan cairan infus. Setelah mengetahui batas maksimal maka hasilnya akan dimasukkan kedalam program Arduino IDE.

Tabel 13. Hasil Pengujian Jumlah Tetesan

No	Kecepatan infus (tpm)	Jumlah cairan infus (Tetes)				Waktu
		Sensor 1	Sensor 2	Sensor 3	Sensor 4	
1	20 tpm	6487	6415	6437	6358	5 jam 30 menit
2	40 tpm	6496	6386	6442	6233	2 jam 45 menit
3	60 tpm	6472	6393	6459	6345	1 jam 50 menit
4	Rata-rata	6485	6398	6446	6312	

b. Uji Kerja Batas Infus

Pengujian kerja batas infus dimaksudkan untuk mengetahui indikasi yang terjadi pada saat infus berjalan dengan perubahan warna pada jarum yang menunjukkan ketersediaan infus. Pengujian ini dilakukan dengan cara menjalankan program pada alat serta mengamati jarum penunjuk kapasitas infus pada aplikasi Cayenne.

Tabel 14. Hasil Uji Kerja Batas Infus

No	Kapasitas Infus (%)				Peringatan kehabisan infus			
	30 tpm	40 tpm	50 tpm	60 tpm	1	2	3	4
1	100%	100%	100%	100%	Hijau	Hijau	Hijau	Hijau
2	90 %	90 %	90 %	90 %	Hijau	Hijau	Hijau	Hijau
3	80 %	80 %	80 %	80 %	Hijau	Hijau	Hijau	Hijau
4	70 %	70 %	70 %	70 %	Hijau	Hijau	Hijau	Hijau
5	60 %	60 %	60 %	60 %	Hijau	Hijau	Hijau	Hijau
6	50 %	50 %	50 %	50 %	Kuning	Kuning	Kuning	Kuning
7	40 %	40 %	40 %	40 %	Kuning	Kuning	Kuning	Kuning
8	30 %	30 %	30 %	30 %	Kuning	Kuning	Kuning	Kuning
9	20 %	20 %	20 %	20 %	Kuning	Kuning	Kuning	Kuning
10	10 %	10 %	10 %	10 %	Merah	Merah	Merah	Merah
11	0 %	0 %	0 %	0 %	Merah	Merah	Merah	Merah

B. Pembahasan

Berdasarkan hasil pengukuran dan percobaan dari beberapa rangkaian dan komponen pada proyek akhir ini, dapat disimpulkan bahwa rangkaian ini dapat bekerja dengan baik dan sesuai dengan fungsinya. Pada pengukuran beberapa rangkaian sistem terdapat perbedaan sedikit dari hasil pengukuran dengan apa yang diperoleh dari teori atau dari *datasheet* komponen. Perbedaan hasil pengukuran terjadi karena ada beberapa faktor seperti nilai komponen yang tidak sesuai dengan labelnya, kondisi alat ukur yang tidak bagus, toleransi nilai komponen dari pabrik dan kurang telitinya dalam pengukuran.

1. Pembahasan Uji Fungsional

a. Analisis Pengujian NodeMCU ESP8266-12E

Berdasarkan percobaan memasukkan program dari *software* Arduino IDE pada NodeMCUESP8266-12E, program berjalan dengan baik. Arduino IDE terkoneksi dengan perangkat atau board NodeMCU pada port COM43 serta chip mikrokontroler dalam kondisi baik karena program dapat *diupload* ke mikrokontroler. Ditandai dengan “*done uploading*” pada Arduino IDE. Dengan masuknya program dari Arduino IDE maka NodeMCU dapat mengontrol perintah-perintah yang ada diprogram.

b. Analisis Pengujian Sensor Inframerah

Berdasarkan analisis kebutuhan, sensor inframerah digunakan sebagai pembaca tetesan infus, sensor ini bekerja mengirimkan data

digital ke NodeMCU ESP8266 setiap cairan infus menetes. Kebutuhan tegangan komponen ini adalah 5V yang diambil dari *stepdown*. Hasil dari percobaan dengan masing-masing sensor inframerah dapat bekerja dengan sesuai tujuan dan fungsinya yaitu sensor dapat mendeteksi setiap tetesan infus dengan kecepatan yang berbeda-beda sehingga kebutuhan pasien dengan kecepatan infus tinggi juga dapat terdeteksi dengan baik. Hal-hal yang perlu dihindarkan agar kinerja sensor inframerah terjamin dengan baik yaitu:

1) Catu daya tidak stabil atau kurang dari 5V

Pemakaian catu daya harus 5V dengan jumlah resistansi maksimal +20%. Apabila kurang atau lebih dari 5V sensor inframerah tidak bisa digunakan dengan normal dan bisa terjadi hangus terbakar karena kelebihan arus.

2) Sensitifitas sensor inframerah

Pada bagaian sensor terdapat trimpot (potensio) yang digunakan sebagai pengatur jarak untuk mendeteksi objek. Untuk mendeteksi tetesan infus maka trimpot pada sensor harus diatur terlebih dahulu agar tetesan infus dapat terbaca, namun saat infus tersenggol maka akan mempengaruhi pantulan dari sensor inframerah ke photodiode maka harus melakukan pengaturan ulang dengan trimpot yang berada pada sensor.

3) Sensor *holder*

Memasang sensor inframerah pada sensor *holder* harus sesuai dengan arah yang akan dilalui tetesan infus agar sensor inframerah dapat selalu mendeteksi tetesan infus. Bahan sensor *holder* terbuat dari *acrilic* yang kaku dan diletakkan di *drip chamber* yang terbuat dari plastik yang kaku maka sangat mudah bergeser, maka dari itu pemasangan sensor *holder* harus sekuat mungkin.

e. Analisis Pengujian Aplikasi Cayenne

Karakter yang dimunculkan oleh *software* Cayenne sudah sesuai dengan dengan program yang diperintahkan dalam Arduino IDE yang dimasukan dalam mikrokontroler NodeMCU ESP8266 namun masih ada batasan untuk menentukan jumlah satuan tetes infus dan volume infus yang digunakan hanya bisa 500 ml saja, tampilan yang dapat dilihat dalam *software* Cayenne diantaranya dapat menunjukkan menu utama berupa tampilan *gauge* jumlah infus yang tersedia serta angka jumlah tetesan infus, selain itu ada juga tampilan untuk memperingati bahwa infus akan habis berupa jarum yang menunjukkan ketersediaan infus akan condong ke arah kiri dan berwarna merah. Tampilan pada *software* Cayenne sangat bergantung sekali pada koneksi internet yang cepat dan stabil agar data yang ditampilkan dapat berjalan secara *real time*, *software* Cayenne sering *offline* jika koneksi internet berjalan lambat.

f. Analisis Pengujian Catu Daya

1) Tanpa beban

Setelah dilakukan pengukuran, keluaran pengukuran NodeMCU ESP8266 tanpa beban adalah 11,70, tidak murni 12V dan memiliki rata-rata eror 2,5% dengan selisih 0,30V . Untuk pengukuran 4 sensor infra merah mempunyai rata-rata keluaran 4,9V dan memiliki rata-rata eror 2% dengan selisih 0,1% dengan tegangan yang normal.

2) Dengan beban

Pengukuran catu daya dengan beban, keluaran pengukuran NodeMCU ESP8266 tanpa beban adalah 11,9, dan memiliki rata-rata eror 0,83% dengan selisih 0,1V . Untuk pengukuran 4 sensor infra merah dengan beban mempunyai rata-rata keluaran 4,99V dan memiliki rata-rata eror 0.2% dengan selisih 0,01 dengan tegangan yang normal. Berdasarkan hasil pengujian di atas dapat disimpulkan bahwa pembagian tegangan untuk menyuplai masing masing komponen sudah cukup baik dan stabil untuk memenuhi sistem alat.

2. Pembahasan Uji Unjuk Kerja

a. Analisis Pengujian Jumlah Tetesan

Pengujian jumlah tetesan infus sampai habis memiliki hasil yang berbeda-beda, pada percobaan pertama tetesan infus menetes dengan kecepatan 20 tpm selama 5 jam 30 menit, maka seharusnya jumlah tetesan akan menetes sebanyak 6600 tetes. Percobaan ke dua tetesan

infus menetes dengan kecepatan 40 tpm selama 2 jam 45 menit, maka seharusnya jumlah tetesan akan menetes sebanyak 6600 tetes. percobaan ke tiga tetesan infus menetes dengan kecepatan 60 tpm selama 1 jam 50 menit, maka seharusnya jumlah tetesan akan menetes sebanyak 6600 tetes. Jumlah tetesan rata-rata dari setiap percobaan adalah 6600 tetes, maka sensor 1 memiliki rata-rata jumlah tetes habis mencapai 6485 memiliki eror sebesar 1,74%, sensor 2 memiliki rata-rata jumlah tetes habis mencapai 6398 memiliki eror sebesar 3,06%, sensor 3 rata-rata jumlah tetes habis 6446 memiliki eror sebesar 2,33%, dan sensor 4 rata-rata jumlah tetes habis mencapai 6312 memiliki eror sebesar 4,36%.

$$\text{Rata-rata eror adalah } \frac{1,74+3,06+2,33+4,36}{4} = 2,87 \%$$

Faktor yang mempengaruhi perbedaan jumlah tetes habis dipengaruhi karena beberapa faktor seperti pengisian cairan infus yang tidak sama serta sensor inframerah yang tidak mendeteksi jatuhnya tetesan infus, atau sensor *holder* yang bergeser sehingga jarak pantulan sensor menjadi berubah. Hasil dari pengujian ini akan dimasukkan ke program Arduino IDE sebagai batas maksimal tetesan infus.

b. Uji Kerja Batas Infus

Setelah melakukan pengujian batas infus, tampilan yang ada dalam web dengan menggunakan aplikasi Cayenne sesuai dengan yang diperintahkan pada program Arduino IDE. Batas infus dapat dilihat dengan jarum serta warna jarum yang memberi peringatan ketersediaan

cairan infus dengan warna hijau yang berarti jumlah cairan infus masih aman untuk dilanjutkan, warna oranye yang menunjukkan bahwa jumlah cairan infus akan segera diganti, warna merah yang berarti jumlah cairan infus yang harus cepat diganti.

BAB V

Kesimpulan dan Saran

A. Kesimpulan

Berdasarkan hasil riset alat monitoring tetesan infus selama pengerjaan tugas akhir, maka dapat disimpulkan:

1. Perangkat keras alat monitoring tetesan infus yang dirancang menggunakan mikrokontroler NodeMCU ESP8266-12E sebagai kendali utama serta pengirim data melalui jaringan WiFi dan modul sensor IR *Obstacle Avoidance* sebagai pendeteksi tetesan infus dapat dapat dirancang menggunakan catu daya dengan keluaran DC +12V dan +5V lalu untuk menurunkan tegangan menggunakan Modul *Step Down* DC-DC LM2596 sebagai rangkaian penurun tegangan dari DC 12V ke DC 3,3V.
2. Perangkat lunak alat monitoring tetesan infus berbasis ESP8266 dapat dirancang dengan pemrograman menggunakan *software* Arduino IDE, setelah mengunduh *library* ESP8266 dan Cayenne terlebih dahulu di Arduino IDE agar dapat *dicompile*. Tampilan untuk memonitor tetesan infus dapat dilihat dengan web menggunakan *cloud* Cayenne yang praktis dan juga bebas biaya, namun tampilan yang diinginkan tetap dapat berjalan dengan baik selama koneksi internet berjalan lancar dan stabil.
3. Hasil pengujian dengan menggunakan 4 buah infus dengan isi cairan 500 ml dengan kecepatan tetesan berbeda-beda, sensor inframerah dapat mendeteksi tetesan infus dan mengirimkan data berupa sinyal digital.

NodeMCU ESP8266 dapat mengolah data dan mengirim data menggunakan jaringan WiFi. Tampilan yang terdapat pada web berjalan dengan *real time* tergantung kecepatan koneksi internet, kondisi tetesan infus dapat dimonitor sampai tetesan infus benar-benar sudah habis dengan tingkat keberhasilan 97,13%. Hasil pengukuran tegangan menunjukkan keluaran pengukuran NodeMCU ESP8266 tanpa beban adalah 11,70, memiliki rata-rata keberhasilan 97,5%, Untuk pengukuran 4 sensor infra merah mempunyai rata-rata keluaran 4,9V dan memiliki rata rata keberhasilan 98%. Keluaran pengukuran NodeMCU ESP8266 tanpa beban adalah 11,9, dan memiliki rata-rata keberhasilan 99,17%, Untuk pengukuran 4 sensor infra merah dengan beban mempunyai rata-rata keluaran 4,99V dan memiliki rata-rata keberhasilan 99.8%.

B. Keterbatasan Alat

Alat *monitoring* tetesan infus menggunakan web secara *online* berbasis ESP8266 dengan pemrograman Arduino IDE, memiliki keterbatasan dalam sistem kerjanya, antara lain:

1. Sensor infus *holder* yang kurang baik sehingga sensor sering tergeser.
2. Pengaturan jarak pantulan sesnsor ke objek sulit untuk diatur.
3. Aplikasi Cayenne membutuhkan koneksi internet yang cepat dan stabil.

Karakter dan fitur yang digunakan masih terbatas, sehingga volume infus tidak dapat diatur secara manual.

4. Botol infus belum menggunakan botol infus yang pabrikan, sehingga volume cairan tidak selalu tepat.

C. Saran

Berdasarkan keterbatasan waktu, kemampuan dan dana, masih banyak kekurangan dalam pengerjaan alat yang dibuat ini, maka dari itu penulis menyarankan sebagai berikut:

1. Penambahan sensor *holder* dengan desain dan materi bahan yang lebih baik lagi agar sensor tidak mudah bergeser.
2. Sensor inframerah sebaiknya diletakkan berlawanan arah dengan phottodioda pada *drip chamber* sehingga jarak pantulan terhadap tetesan infus lebih stabil.
3. Menggunakan *transmitter* dan *receiver* yang tidak ketergantungan terhadap koneksi internet. Mengembangkan web atau *software* dengan karakter dan fitur yang lebih banyak sehingga volume infus yang akan digunakan dapat diatur secara manual.
4. Alat monitoring tetesan infus di rumah sakit sebaiknya perlu kalibrasi awal dahulu untuk mengurangi resiko yang terjadi pada pasien.

Daftar Pustaka

- A, Potter, & Perry, A. G. (2006). *Buku Ajar Fundamental Keperawatan: Konsep, Proses, Dan Praktik, edisi 4, Volume.2*. Jakarta: EGC.
- Arcelia, Monalisa, & Islami, D.N. (2016). *Pengenalan NodeMCU esp8266 versi 12e*. Diakses pada tanggal 25 Juni 2017 dari <http://dirakit.com/project/66>.
- Arduino. (2015). *Arduino Software (IDE)*. Diakses pada tanggal 25 Juni 2017 dari <https://www.arduino.cc/en/Guide/Environment>
- Arjunaldi. (2017). *Preview LM2596 Step Down Module*. Diakses pada tanggal 30 Juni 2017 dari <http://arjunaldi.staff.telkomuniversity.ac.id/previewlm2596-step-module/>
- Darmadi. (2010). *Infeksi Nosokomial*. Jakarta : salemba
- Evers, AS, and Mervyn Maze. (2004). *Anesthetic Pharmacology: Physiologic Principles and Clinical Practice*. United Kingdom : Churchill Livingstone.
- Gregorius, Agung. (2001). *Desain Web Interaktif Dengan Frontpage 2000 dan Dreamweaver 4*. Jakarta: PT Elex Media Komputindo.
- Ida, W. (2010). *Pengaruh Area Hotspot(Wi-Fi) Bagi Pemenuhan Kebutuhan Informasi Pemustaka Di Kantor Perpustakaan Daerah Kabupaten Jepara*. Tesis, tidak dipublikasikan. Universitas Diponegoro, Semarang.
- Kadir, Abdul. (2002). *Pengenalan Sistem Informasi*. Yogyakarta: Andi
- Latief, AS, dkk. (2002). *Petunjuk Praktis Anestesiologi : Terapi Cairan Pada Pembedahan*. Edisi Kedua. Bagian Anestesiologi dan Terapi Intensif, FKUI.
- Loehoeri, Soebagjo, & Wirjoatmodjo, Moefrodi. (2007). *Rehidrasi*. Jakarta. BAIPD. J I.E IV. FKUI.
- Puruhito. (1995). *Dasar-Dasar Pemberian Cairan dan Elektrolit Pada Kasus-Kasus Bedah*. Surabaya. Airlangga Univercity Press.
- QQTrading . (2016). *IR Infrared Obstacle Detaction Sensor Module 2 - 30cm FC-51*. Diakses pada tanggal 30 Juni 2017 dari <http://qqtrading.com.my/ir-infrared-obstacle-detaction-sensor-module-fc-5>
- Sampurna. (1996). *World Wide Web*. Jakarta: Andi.

Semiconductor Components Industries. (2008). *LM2596*. Diakses pada tanggal 30 Juni 2017 dari <https://www.onsemi.com/pub/Collateral/LM2596-D.PDF>.

Smeltzer, & Bare (2002), *Keperawatan Medikal-Bedah*. Jakarta:EGC.

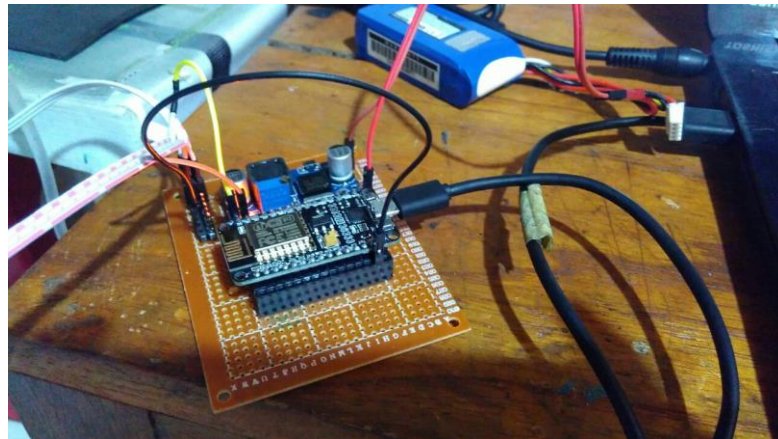
Syahrul, & Hidayat. (2009). Sistem Pemantauan Infus Pasien Terpusat . *Jurnal Teknik Komputer* , Vol. 17 No.1 (1- 12).

Waite C., Waite P., & M Pirmohamed. (2004). Intravenous therapy. *Postgrad Med J*, 80:1–6. doi: 10.1136/pgmj.2003.010421.Smeltzer, S.C, & Bare. (2002). *Buku Ajar Keperawatan Medikal Bedah, edisi 8 volume 2*. Jakarta : EGC.

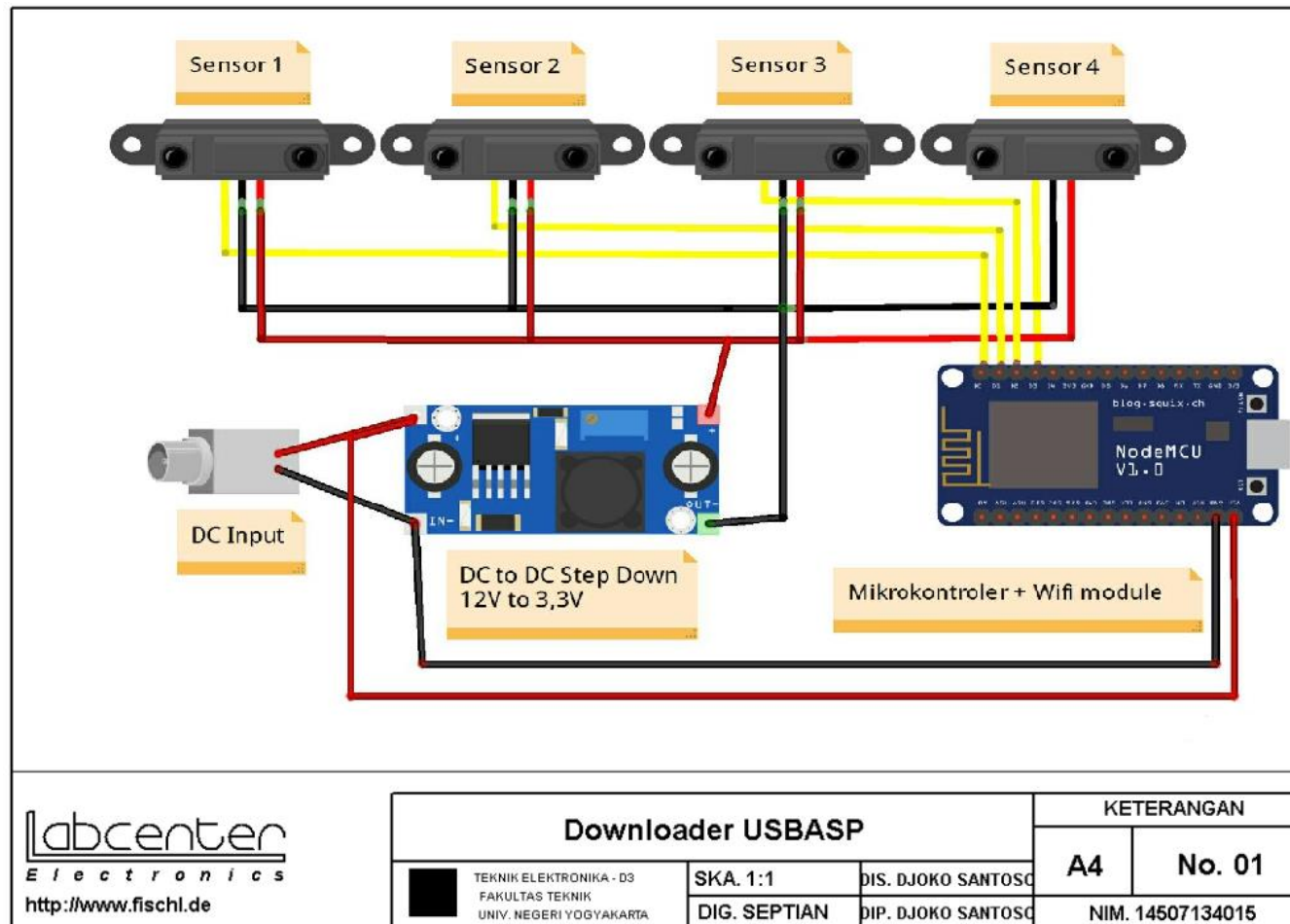
Yuhefizar. (2008). *10 Jam Menguasai Internet Teknologi dan API*. Bandung: Alexmedia.

Lampiran

Lampiran 1. Gambar Alat



Lampiran 2. Skema Rangkaian Keseluruhan



Lampiran 3. Program Alat

```
//project code : final project
//project name : infus monitoring by i-net
//author : septian
//date : 12 July 2017

// *****

#define CAYENNE_DEBUG
#define CAYENNE_PRINT Serial
#define max_tick 6500

//hardware config
#include "CayenneDefines.h"
#include "BlynkSimpleEsp8266.h"
#include "CayenneWiFiClient.h"

//global variable
const int sensor1_pin = D0;
const int sensor2_pin = D1;
const int sensor3_pin = D2;
const int sensor4_pin = D3;

//global variable
int i, detik, tick, timer = 0;
int sensor_count[4], sensor_state[4];
float volume[4];
int warning_time=60;
long interval=1000;

//server data
char token[] = "rqlglx7ivl";
```



```

char ssid[] = "AERO DEPARTMENT";

char password[] = "";

void read_sensor(int pin) { //nama sub program

    sensor_count[pin]++;

    volume[pin] = float(100 * (max_tick - sensor_count[pin]) / max_tick);

    // temp_interval[i] = current_time - last_time[i];
    // last_time[i] = current_time;
    // if (temp_interval[i] >= 0) {
    //     interval[i] = temp_interval[i];
    //     frequency[i] = 60000 / interval[i];
    // }

    for (i = 0; i < 4; i++) {
        Serial.print("c:");

        Serial.print(sensor_count[i]);

        Serial.print(" V:");

        Serial.print(volume[i]);

    }

    Serial.println();
}

void setup() {

    Serial.begin(115200);

    pinMode(sensor1_pin, INPUT);

    pinMode(sensor2_pin, INPUT);

    pinMode(sensor3_pin, INPUT);

    pinMode(sensor4_pin, INPUT);

    Cayenne.begin(token, ssid, password);

    Serial.println("Hello World...");

}

```

```

char state = 0;

void loop() {
    Cayenne.run();

    // #####
    // START MILIS
    // #####
    unsigned long currentMillis = millis();
    if(currentMillis - previousMillis > interval) {
        detik++;
        Serial.println(detik);
    }
    // #####

    // WARNING
    // #####
    if (detik>=warning_time) { //
        Serial.println("WARNING !!!!!!!");
    }
    // #####

    tick++;
    if (tick > 1000) {

        timer++;
        tick = 0;
        Serial.println(timer);
        read_sensor(0);
    }

    if (timer > 30) state = 1;
}

```

```

//sensor 1
if (sensor_state[0] == 0 && !digitalRead(sensor1_pin)) {
    read_sensor(0);
    sensor_state[0] = 1;
    detik=0;
}

if (sensor_state[0] == 1 && digitalRead(sensor1_pin)) {
    sensor_state[0] = 0;
}


//sensor 2
if (sensor_state[1] == 0 && !digitalRead(sensor2_pin)) {
    read_sensor(1);
    sensor_state[1] = 1;
    detik=0;
}

if (sensor_state[1] == 1 && digitalRead(sensor2_pin)) {
    sensor_state[1] = 0;
}


//sensor 3
if (sensor_state[2] == 0 && !digitalRead(sensor3_pin)) {
    read_sensor(2);
    sensor_state[2] = 1;
    detik=0;
}

if (sensor_state[2] == 1 && digitalRead(sensor3_pin)) {
    sensor_state[2] = 0;
}


//sensor 4

```

```

    if (sensor_state[3] == 0 && !digitalRead(sensor4_pin)) {
        read_sensor(3);
        sensor_state[3] = 1;
        detik=0;
    }
    if (sensor_state[3] == 1 && digitalRead(sensor4_pin)) {
        sensor_state[3] = 0;
    }

    delay(1);
}

//Sensor 1
CAYENNE_OUT(V0) {
    Cayenne.virtualWrite(V0, sensor_count[0]); }

CAYENNE_OUT(V1) {
    Cayenne.virtualWrite(V1, volume[0]);
}

//Sensor 2
CAYENNE_OUT(V2) {
    Cayenne.virtualWrite(V2, sensor_count[1]);
}

CAYENNE_OUT(V3) {
    Cayenne.virtualWrite(V3, volume[1]);
}

//Sensor 3
CAYENNE_OUT(V4) {
    Cayenne.virtualWrite(V4, sensor_count[2]);
}

```

```
}

CAYENNE_OUT(V5) {
    Cayenne.virtualWrite(V5, volume[2]);
}

//Sensor 4
CAYENNE_OUT(V6) {
    Cayenne.virtualWrite(V6, sensor_count[3]);
}

CAYENNE_OUT(V7) {
    Cayenne.virtualWrite(V7, volume[3]);
}
```

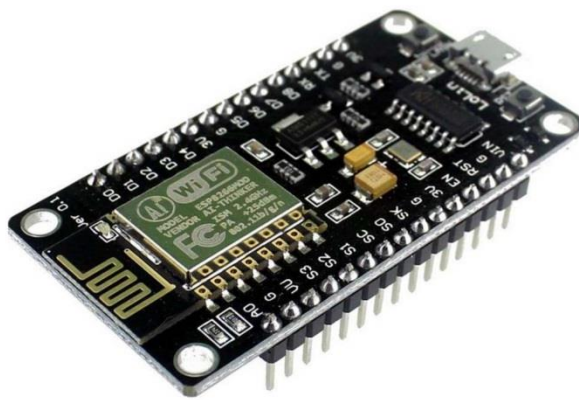
Lampiran 4. Data Sheet NodeMCU ESP8266-12E



Handson Technology

User Manual V1.2

ESP8266 NodeMCU WiFi Devkit



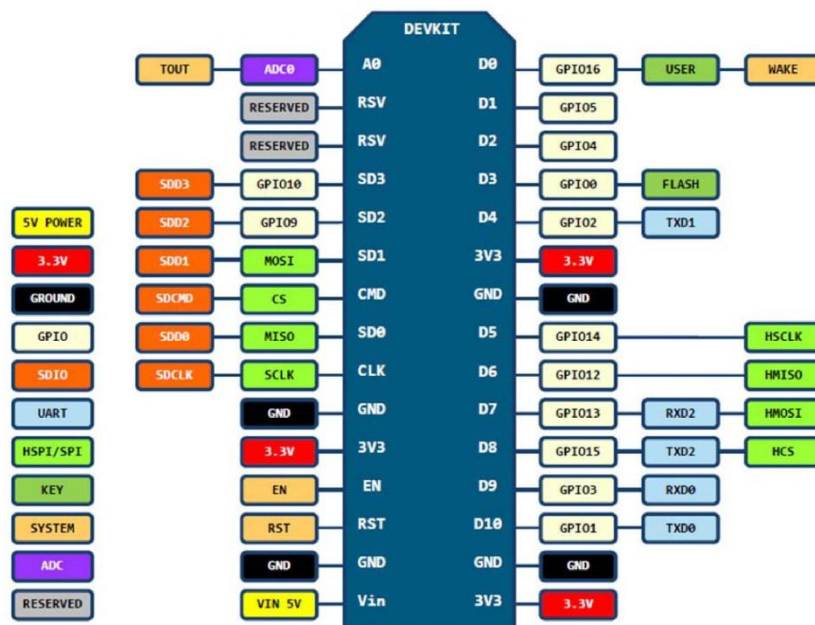
The ESP8266 is the name of a micro controller designed by Espressif Systems. The ESP8266 itself is a self-contained WiFi networking solution offering as a bridge from existing micro controller to WiFi and is also capable of running self-contained applications.

This module comes with a built in USB connector and a rich assortment of pin-outs. With a micro USB cable, you can connect NodeMCU devkit to your laptop and flash it without any trouble, just like Arduino. It is also immediately breadboard friendly.

1. Specification:

- Voltage:3.3V.
- Wi-Fi Direct (P2P), soft-AP.
- Current consumption: 10uA~170mA.
- Flash memory attachable: 16MB max (512K normal).
- Integrated TCP/IP protocol stack.
- Processor: Tensilica L106 32-bit.
- Processor speed: 80~160MHz.
- RAM: 32K + 80K.
- GPIOs: 17 (multiplexed with other functions).
- Analog to Digital: 1 input with 1024 step resolution.
- +19.5dBm output power in 802.11b mode
- 802.11 support: b/g/n.
- Maximum concurrent TCP connections: 5.

2. Pin Definition:



D0(GPIO16) can only be used as gpio read/write, no interrupt supported, no pwm/i2c/ow supported.

3. Using Arduino IDE

The most basic way to use the ESP8266 module is to use serial commands, as the chip is basically a WiFi/Serial transceiver. However, this is not convenient. What we recommend is using the very cool Arduino ESP8266 project, which is a modified version of the Arduino IDE that you need to install on your computer. This makes it very convenient to use the ESP8266 chip as we will be using the well-known Arduino IDE. Following the below step to install ESP8266 library to work in Arduino IDE environment.

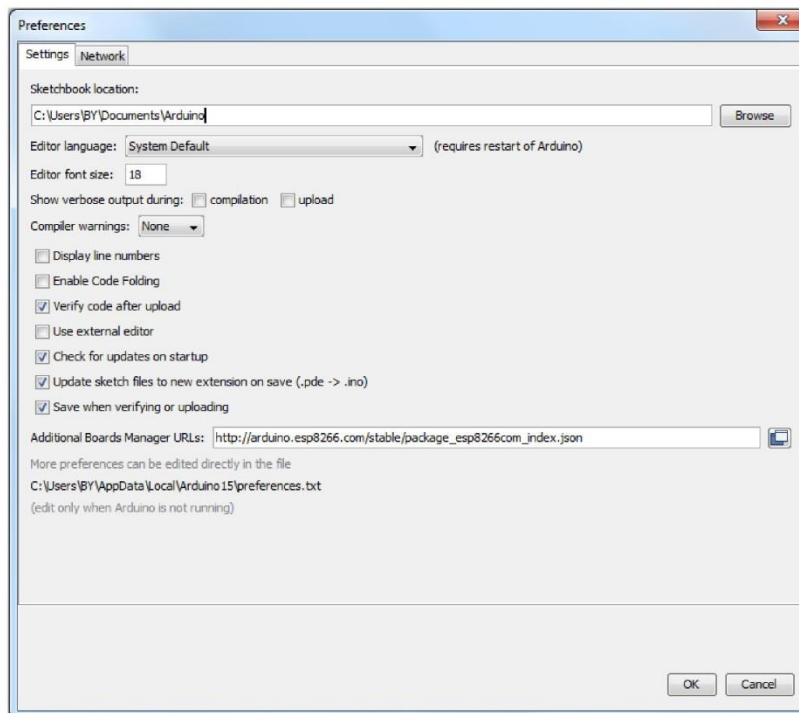
3.1 Install the Arduino IDE 1.6.4 or greater

[Download Arduino IDE from Arduino.cc \(1.6.4 or greater\) - don't use 1.6.2 or lower version! You can use your existing IDE if you have already installed it.](#)

[You can also try downloading the ready-to-go package from the ESP8266-Arduino project, if the proxy is giving you problems.](#)

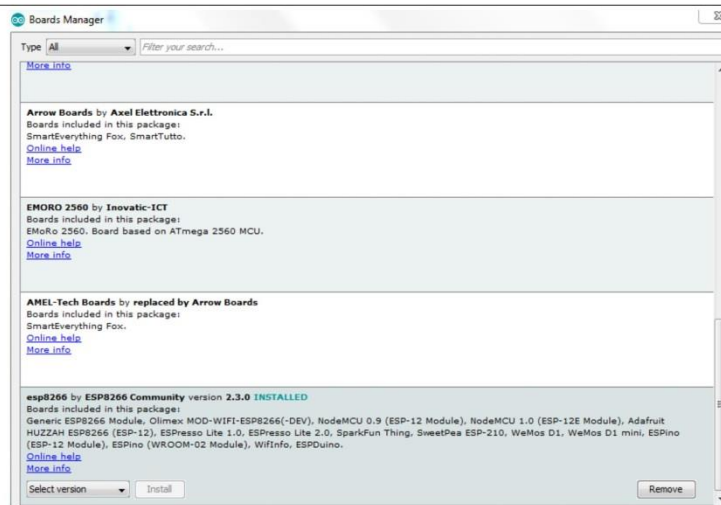
3.2 Install the ESP8266 Board Package

Enter http://arduino.esp8266.com/stable/package_esp8266com_index.json into *Additional Board Manager URLs* field in the Arduino v1.6.4+ preferences.



Click 'File' -> 'Preferences' to access this panel.

Next, use the Board manager to install the ESP8266 package.

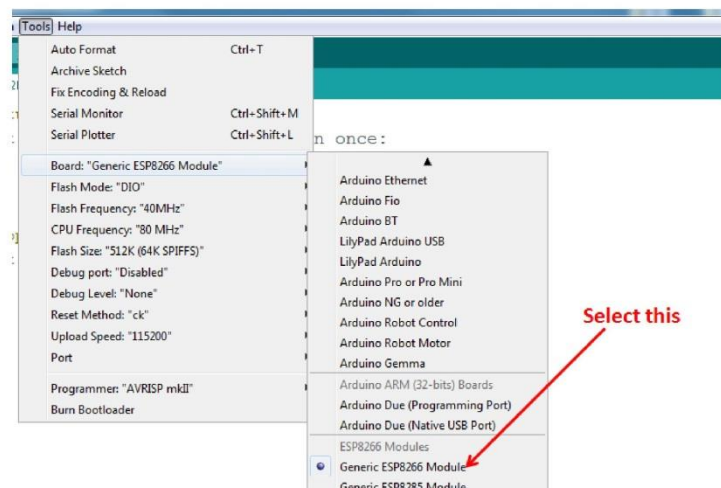


Click 'Tools' -> 'Board:' -> 'Board Manager...' to access this panel.

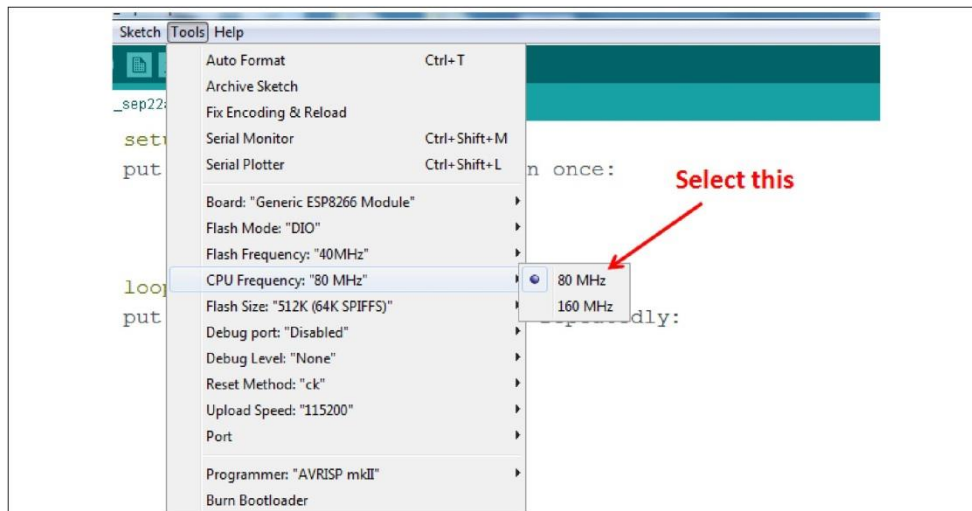
Scroll down to 'esp8266 by ESP8266 Community' and click "Install" button to install the ESP8266 library package. Once installation completed, close and re-open Arduino IDE for ESP8266 library to take effect.

3.3 Setup ESP8266 Support

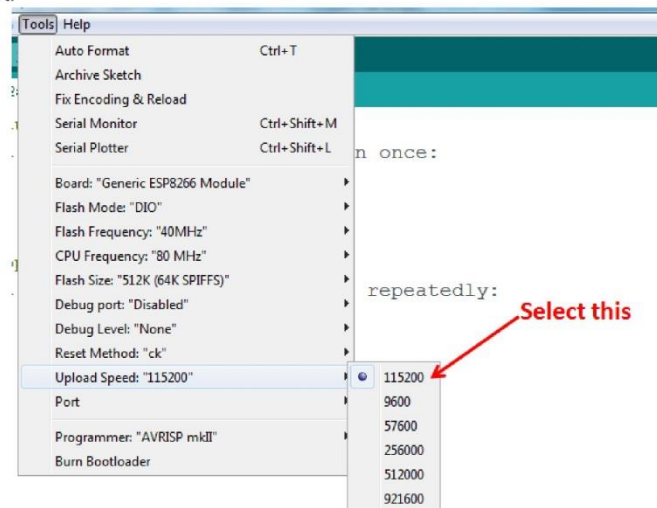
When you've restarted Arduino IDE, select 'Generic ESP8266 Module' from the 'Tools' -> 'Board:' dropdown menu.



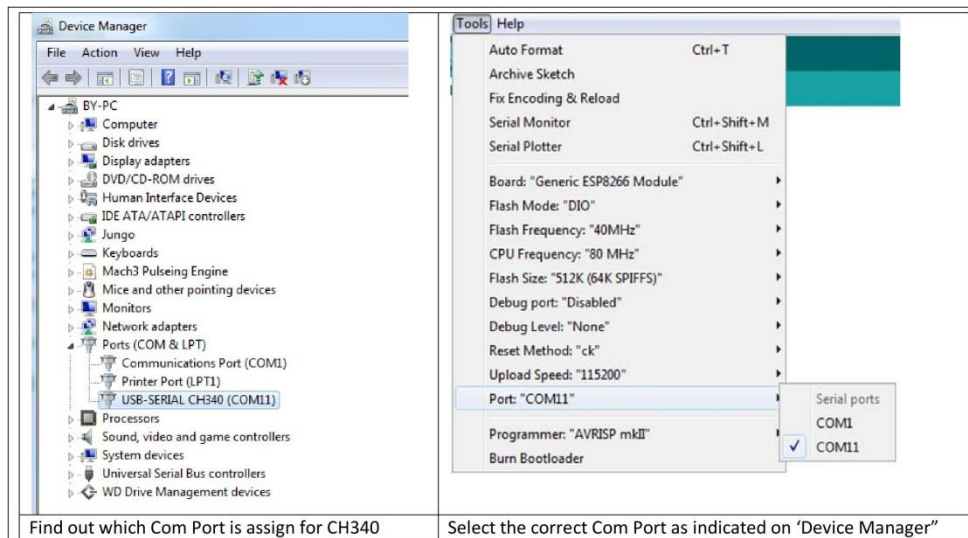
Select 80 MHz as the CPU frequency (you can try 160 MHz overclock later)



Select '115200' baud upload speed is a good place to start - later on you can try higher speeds but 115200 is a good safe place to start.



Go to your Windows 'Device Manager' to find out which Com Port 'USB-Serial CH340' is assigned to. Select the matching COM/serial port for your CH340 USB-Serial interface.



Find out which Com Port is assign for CH340

Select the correct Com Port as indicated on 'Device Manager'

Note: if this is your first time using CH340 "USB-to-Serial" interface, please install the driver first before proceed the above Com Port setting. The CH340 driver can be download from the below site:

<https://github.com/nodemcu/nodemcu-devkit/tree/master/Drivers>

3.4 Blink Test

We'll begin with the simple blink test.

Enter this into the sketch window (and save since you'll have to). Connect a LED as shown in Figure3-1.

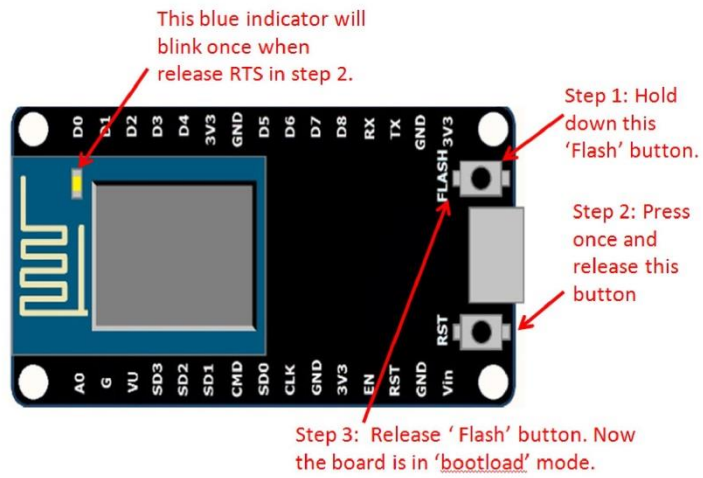
```
void setup() {
  pinMode(5, OUTPUT); // GPIO05, Digital Pin D1
}

void loop() {
  digitalWrite(5, HIGH);
  delay(900);
  digitalWrite(5, LOW);
  delay(500);
}
```

Now you'll need to put the board into bootload mode. You'll have to do this before each upload. There is no timeout for bootload mode, so you don't have to rush!

- Hold down the 'Flash' button.
- While holding down 'Flash', press the 'RST' button.
- Release 'RST', then release 'Flash'

- When you release the 'RST' button, the blue indicator will blink once, this means its ready to bootload.



Once the ESP board is in bootload mode, upload the sketch via the IDE, Figure 3-2.

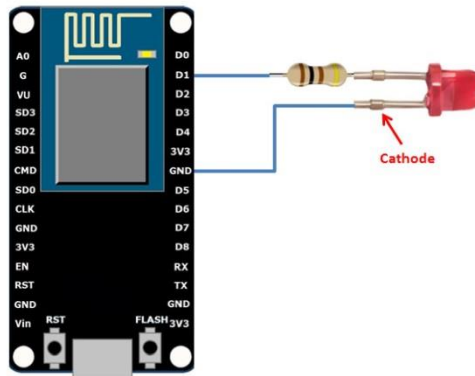


Figure3-1: Connection diagram for the blinking test

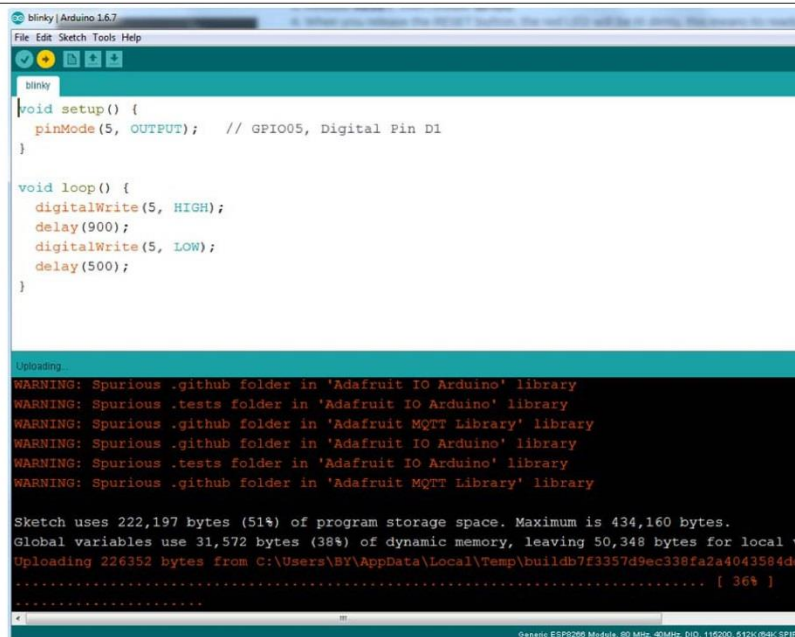


Figure 3.2: Uploading the sketch to ESP8266 NodeMCU module.

The sketch will start immediately - you'll see the LED blinking. Hooray!

3.5 Connecting via WiFi

OK once you've got the LED blinking, let's go straight to the fun part, connecting to a webserver. Create a new sketch with this code:

Don't forget to update:

```

const char* ssid = "yourssid";
const char* password = "yourpassword";

```

to your WiFi access point and password, then upload the same way: get into bootload mode, then upload code via IDE.

```

/*
 * Simple HTTP get webclient test
 */

#include <ESP8266WiFi.h>

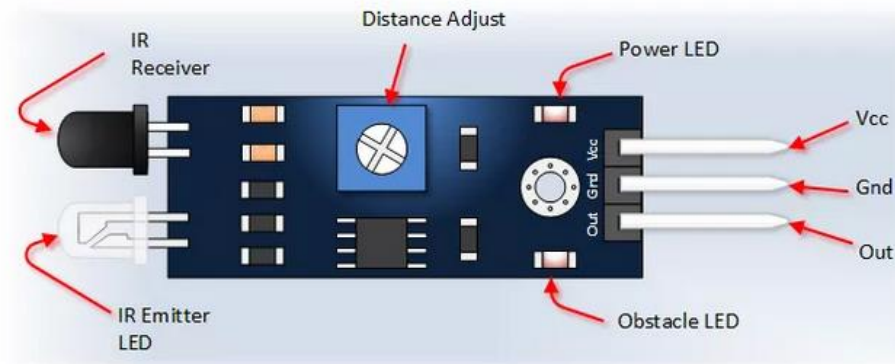
const char* ssid = "handson"; // key in your own SSID
const char* password = "abc1234"; // key in your own WiFi access point
password

```

Lampiran 5. Data Sheet Modul Sensor IR *Obstacle Avoidance*

IR Obstacle Detection Module Pin Outs

The drawing and table below identify the function of module pin outs, controls and indicators.



Pin, Control Indicator	Description
Vcc	3.3 to 5 Vdc Supply Input
Gnd	Ground Input
Out	Output that goes low when obstacle is in range
Power LED	Illuminates when power is applied
Obstacle LED	Illuminates when obstacle is detected
Distance Adjust	Adjust detection distance. CCW decreases distance. CW increases distance.
IR Emitter	Infrared emitter LED
IR Receiver	Infrared receiver that receives signal transmitted by Infrared emitter.

Lampiran 6. Data Sheet Modul Step Down DC-DC LM2596

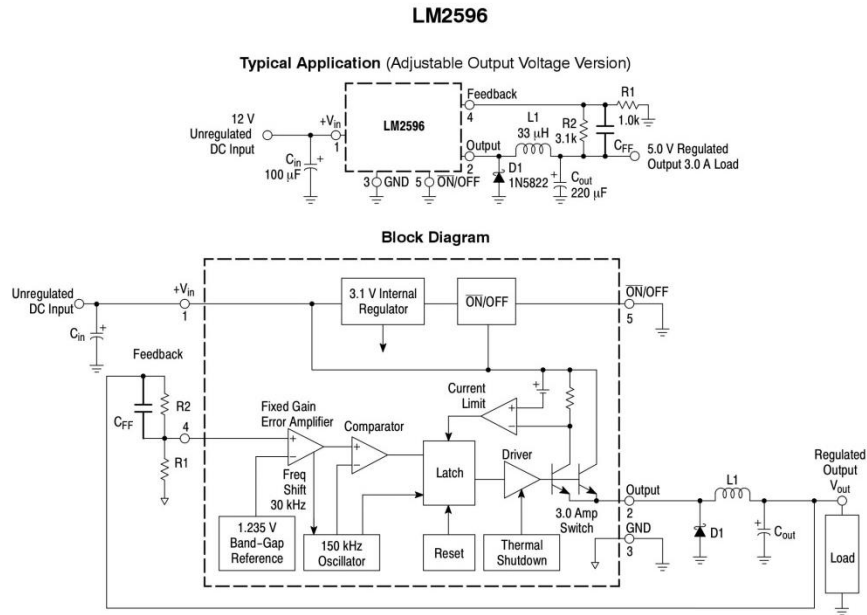


Figure 1. Typical Application and Internal Block Diagram

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Maximum Supply Voltage	V_{in}	45	V
ON/OFF Pin Input Voltage	—	$-0.3 \text{ V} \leq V \leq +V_{in}$	V
Output Voltage to Ground (Steady-State)	—	-1.0	V
Power Dissipation			
Case 314B and 314D (TO-220, 5-Lead)	P_D	Internally Limited	W
Thermal Resistance, Junction-to-Ambient	$R_{\theta JA}$	65	°C/W
Thermal Resistance, Junction-to-Case	$R_{\theta JC}$	5.0	°C/W
Case 936A (D ² PAK)	P_D	Internally Limited	W
Thermal Resistance, Junction-to-Ambient	$R_{\theta JA}$	70	°C/W
Thermal Resistance, Junction-to-Case	$R_{\theta JC}$	5.0	°C/W
Storage Temperature Range	T_{stg}	-65 to +150	°C
Minimum ESD Rating (Human Body Model: C = 100 pF, R = 1.5 kΩ)	—	2.0	kV
Lead Temperature (Soldering, 10 seconds)	—	260	°C
Maximum Junction Temperature	T_J	150	°C

Stresses exceeding Maximum Ratings may damage the device. Maximum Ratings are stress ratings only. Functional operation above the Recommended Operating Conditions is not implied. Extended exposure to stresses above the Recommended Operating Conditions may affect device reliability.

LM2596

PIN FUNCTION DESCRIPTION

Pin	Symbol	Description (Refer to Figure 1)
1	V_{in}	This pin is the positive input supply for the LM2596 step-down switching regulator. In order to minimize voltage transients and to supply the switching currents needed by the regulator, a suitable input bypass capacitor must be present (C_{in} in Figure 1).
2	Output	This is the emitter of the internal switch. The saturation voltage V_{sat} of this output switch is typically 1.5 V. It should be kept in mind that the PCB area connected to this pin should be kept to a minimum in order to minimize coupling to sensitive circuitry.
3	GND	Circuit ground pin. See the information about the printed circuit board layout.
4	Feedback	This pin is the direct input of the error amplifier and the resistor network R2. R1 is connected externally to allow programming of the output voltage.
5	ON/OFF	It allows the switching regulator circuit to be shut down using logic level signals, thus dropping the total input supply current to approximately 80 μ A. The threshold voltage is typically 1.6 V. Applying a voltage above this value (up to $+V_{in}$) shuts the regulator off. If the voltage applied to this pin is lower than 1.6 V or if this pin is left open, the regulator will be in the "on" condition.

OPERATING RATINGS (Operating Ratings indicate conditions for which the device is intended to be functional, but do not guarantee specific performance limits. For guaranteed specifications and test conditions, see the Electrical Characteristics.)

Rating	Symbol	Value	Unit
Operating Junction Temperature Range	T_J	-40 to +125	$^{\circ}$ C
Supply Voltage	V_{in}	4.5 to 40	V

LM2596

SYSTEM PARAMETERS

ELECTRICAL CHARACTERISTICS Specifications with standard type face are for $T_J = 25^\circ\text{C}$, and those with boldface type apply over full Operating Temperature Range -40°C to $+125^\circ\text{C}$

Characteristics	Symbol	Min	Typ	Max	Unit
LM2596 (Note 1, Test Circuit Figure 15)					
Feedback Voltage ($V_{in} = 12\text{ V}$, $I_{Load} = 0.5\text{ A}$, $V_{out} = 5.0\text{ V}$)	$V_{FB, nom}$		1.23		V
Feedback Voltage ($8.5\text{ V} \leq V_{in} \leq 40\text{ V}$, $0.5\text{ A} \leq I_{Load} \leq 3.0\text{ A}$, $V_{out} = 5.0\text{ V}$)	V_{FB}	1.193 1.18		1.267 1.28	V
Efficiency ($V_{in} = 12\text{ V}$, $I_{Load} = 3.0\text{ A}$, $V_{out} = 5.0\text{ V}$)	η	–	73	–	%
Characteristics	Symbol	Min	Typ	Max	Unit
Feedback Bias Current ($V_{out} = 5.0\text{ V}$)	I_b		25	100 200	nA
Oscillator Frequency (Note 2)	f_{osc}	135 120	150	165 180	kHz
Saturation Voltage ($I_{out} = 3.0\text{ A}$, Notes 3 and 4)	V_{sat}		1.5	1.8 2.0	V
Max Duty Cycle "ON" (Note 4)	DC		95		%
Current Limit (Peak Current, Notes 2 and 3)	I_{CL}	4.2 3.5	5.6	6.9 7.5	A
Output Leakage Current (Notes 5 and 6) Output = 0 V Output = –1.0 V	I_L		0.5 6.0	2.0 20	mA
Quiescent Current (Note 5)	I_Q		5.0	10	mA
Standby Quiescent Current (ON/OFF Pin = 5.0 V ("OFF")) (Note 6)	I_{stby}		80	200 250	μA
ON/OFF PIN LOGIC INPUT					
Threshold Voltage			1.6		V
$V_{out} = 0\text{ V}$ (Regulator OFF)	V_{IH}	2.2 2.4			V
V_{out} = Nominal Output Voltage (Regulator ON)	V_{IL}			1.0 0.8	V
ON/OFF Pin Input Current					
ON/OFF Pin = 5.0 V (Regulator OFF)	I_{IH}	–	15	30	μA
ON/OFF Pin = 0 V (regulator ON)	I_{IL}	–	0.01	5.0	μA

- External components such as the catch diode, inductor, input and output capacitors can affect switching regulator system performance. When the LM2596 is used as shown in the Figure 15 test circuit, system performance will be as shown in system parameters section.
- The oscillator frequency reduces to approximately 30 kHz in the event of an output short or an overload which causes the regulated output voltage to drop approximately 40% from the nominal output voltage. This self protection feature lowers the average dissipation of the IC by lowering the minimum duty cycle from 5% down to approximately 2%.
- No diode, inductor or capacitor connected to output (Pin 2) sourcing the current.
- Feedback (Pin 4) removed from output and connected to 0 V.
- Feedback (Pin 4) removed from output and connected to +12 V to force the output transistor "off".
- $V_{in} = 40\text{ V}$.