

TUGAS AKHIR

PENENTUAN WAKTU SERVIS KENDARAAN BERMOTOR MENGUNAKAN MIKROKONTROLER

*Diajukan Untuk Memenuhi Syarat Memperoleh
Gelar Sarjana Teknik Elektro Pada Fakultas Teknik
Universitas Muhammadiyah Sumatera Utara*

Disusun Oleh:

MUHAMMAD FADLI RANGKUTI

1907220030



PROGRAM STUDI TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA

MEDAN

2023

HALAMAN PENGESAHAN

Tugas Akhir ini diajukan oleh:

Nama : Muhammad Fadli Rangkuti
NPM : 1907220030
Program Studi : Teknik Elektro
Judul Skripsi : Penentuan Waktu Servis Kendaraan Bermotor Menggunakan Mikrokontroler
Bidang ilmu : Kontrol

Telah berhasil dipertahankan di hadapan Tim Penguji dan diterima sebagai salah satu syarat yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Muhammadiyah Sumatera Utara.

Medan, 04 Maret 2024

Mengetahui dan menyetujui:

Dosen Pembimbing

Muhammad Adam, S.T., M.T

Dosen Pembanding I

Faisal Irsan Pasaribu, S.T., M.T

Dosen Pembanding II

Elvy Sahnur, S.T., M.Pd



LEMBAR PERNYATAAN DAN PERSETUJUAN

Kami yang bertanda tangan di bawah ini menerangkan bahwa skripsi yang berjudul di bawah ini:

Penentuan Waktu Servis Kendaraan Bermotor Menggunakan Mikrokontroler

Ditulis oleh Mahasiswa/i yang bernama:

Muhammad Fadli Rangkuti

(NPM: 1907220030)

untuk kemudian disebut sebagai Pihak ke-1,

adalah benar merupakan sebagian hasil dari penelitian Dosen yang melibatkan Mahasiswa/i (Pihak ke-1) di bawah ini:

Judul penelitian : Penentuan Waktu Servis Kendaraan Bermotor Menggunakan Mikrokontroler

Nama dosen : Muhammad Adam, S.T., M.T

Jenis penelitian : Mandiri

Nomor kontrak :

Untuk kemudian disebut sebagai Pihak ke-2.

Untuk itu Pihak ke-2 berhak mempublikasikan isi Skripsi seluruhnya tanpa harus meminta izin dari Pihak ke-1. Sedangkan Pihak ke-1 wajib meminta izin terlebih dahulu kepada Pihak ke-2 bila ingin mempublikasikan isi Skripsi ini.

Demikian Surat Pernyataan dan Persetujuan ini dibuat dengan sebenarnya tanpa ada paksaan dari pihak manapun.

Medan, 04 Maret 2024

Yang membuat pernyataan dan persetujuan:

Pihak ke-2 (Dosen)

Pihak ke-1 (Mahasiswa/i)

(Muhammad Adam, S.T., M.T)
NIDN: 1029057402

(Muhammad Fadli Rangkuti)
NPM: 1907220030

Diketahui oleh:
Ketua Program Studi Teknik Elektro



(Faisal Irsan Pasaribu, S.T., M.T)
NIDN: 1030118101

SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Saya yang bertanda tangan di bawah ini:

Nama Lengkap : Muhammad Fadli Rangkuti

Tempat /Tanggal Lahir : Medan/16 Januari 2001

NPM : 1907220030

Fakultas : Teknik

Program Studi : Teknik Elektro

Menyatakan dengan sesungguhnya dan sejujurnya, bahwa laporan Tugas Akhir saya yang berjudul:

“PENENTUAN WAKTU SERVIS KENDARAAN BERMOTOR MENGGUNAKAN MIKROKONTROLER”,

Bukan merupakan plagiarisme, pencurian hasil karya milik orang lain, hasil kerja orang lain untuk kepentingan saya karena hubungan material dan non-material, ataupun segala kemungkinan lain, yang pada hakekatnya bukan merupakan karya tulis Tugas Akhir saya secara orisinal dan otentik.

Bila kemudian hari diduga kuat ada ketidaksesuaian antara fakta dengan kenyataan ini, saya bersedia diproses oleh Tim Fakultas yang dibentuk untuk melakukan verifikasi, dengan sanksi terberat berupa pembatalan kelulusan/kesarjanaan saya.

Demikian Surat Pernyataan ini saya buat dengan kesadaran sendiri dan tidak atas tekanan ataupun paksaan dari pihak manapun demi menegakkan integritas akademik di Program Studi Teknik Elektro, Fakultas Teknik, Universitas Muhammadiyah Sumatera Utara.

Medan, Juni 2024

Saya yang menyatakan,



Mhd Fadli Rangkuti

ABSTRAK

Perawatan kendaraan bermotor sangatlah penting untuk menjamin supaya kendaraan yang kita miliki tetap awet. Hal ini membuat para pengguna kendaraan bermotor untuk rajin dan rutin menservis kendaraannya. Oleh karena itu pengguna harus memiliki suatu sistem pada kendaraannya supaya dapat mengingatkan kapan kendaraan harus mendapatkan perawatan. Sistem ini akan memperingatkan pengguna untuk segera menservis kendaraannya jika jarak tempuh yang telah ditentukan terlewati. Hal ini akan mengurangi kemungkinan kerusakan mesin yang parah pada kendaraan bermotor, khususnya sepeda motor. Sistem ini menggunakan sensor *proximity* sebagai sensor jarak. Sensor akan mendeteksi jumlah putaran roda kendaraan. Dengan menggunakan sebuah mikrokontroler, jumlah putaran roda nantinya akan diolah untuk menentukan jarak yang telah ditempuh kendaraan. Jarak tempuh dan teks peringatan akan ditampilkan melalui LCD. Led indikator digunakan sebagai tanda peringatan kepada pengguna. Penelitian ini telah berhasil membuat suatu alat penentu waktu servis kendaraan. Alat ini dapat bekerja dengan baik dan dapat digunakan untuk penggunaan sehari-hari. Dilakukan nya penelitian ini guna menghasilkan alat yang berfungsi sebagai penentu waktu servis kendaraan bermotor secara dini sehingga pengguna dapat melakukan perawatan pada kendaraannya tepat waktu. Dengan menggunakan Mikrokontroler AT328 mesin kendaraan menjadi sangat efisien dan memudahkan masyarakat untuk mengetahui jarak waktu servis. Jadi, dilakukan nya penelitian ini guna mencegah terjadinya kerusakan yang parah pada mesin kendaraan, khususnya sepeda motor akibat kurangnya perawatan mesin kendaraan bermotor dan menjaga oli agar kendaraan dalam kondisi prima.

Kata kunci : LCD, Sensor *Proximity*, Mikrokontroler AT328 Tombol

ABSTRACT

Motor vehicle maintenance is very important to ensure that our vehicles remain durable. This condition make the motor vehicle user to be deligent and routine for servicing their vehicles. There for the users should have a system in their vehicle in order to reminded when the vehicles should get a maintenance. This system will warn the users to service the vehicle immediately if the predetermined mileage has passed, so will decrease the possibility of severe engine failure on motor vehicles,especially motor cycles. This system uses proximity sensor as a distance sensor. Sensor will detect the rotation of vehicle wheels. Using a microcontroler, number of spinning wheels will be processed to determine the distance that has been taken. Mileage and warning text will be displayed through a LCD. Led indicator is use as a warning sign to user. This research has succeeded in producing a vehicle reparation time determination device. This device is suitable for daily usage. This research was carried out to produce a tool that functions as an early determinant of motor vehicle service times so that users can carry out maintenance on their vehicles on time. By using the AT328 Microcontroller the vehicle engine becomes very efficient and makes it easier for people to know the interval between service times. So, this research was carried out to prevent serious damage to vehicle engines, especially motorbikes, due to lack of maintenance of motor vehicle engines and maintaining oil so that vehicles are in top condition.

Keyword : LCD, Proximity Sensor, AT328 Button Microcontroller

KATA PENGANTAR

Assalamua'alaikum Warahmatullahi Wabarakatuh

Dengan nama Allah Yang Maha Pengasih lagi Maha Penyayang. Segala puji dan syukur penulis ucapkan kehadiran Allah SWT yang telah memberikan karunia dan nikmat yang tiada terkira. Salah satu dari nikmat tersebut adalah keberhasilan penulis dalam menyelesaikan laporan Tugas Akhir ini yang berjudul “Penentuan Waktu Servis Kendaraan Bermotor Menggunakan Mikrokontroler” sebagai syarat untuk meraih gelar akademik Sarjana Teknik pada Program Studi Teknik Elektro Fakultas Teknik Universitas Muhammadiyah Sumatera Utara (UMSU), Medan. Banyak pihak telah membantu dalam menyelesaikan laporan Tugas Akhir ini, untuk itu penulis menghaturkan rasa terimakasih yang tulus kepada:

1. Ayahanda tercinta Ahmad Fitri Rangkuti, Ibunda tercinta Supiani, serta seluruh keluarga yang telah memberikan bantuan moril maupun materil serta nasehat dan doanya untuk penulis demi selesainya Tugas Akhir ini.
2. Bapak Munawar Alfansury Siregar, S.T., M.T., selaku Dekan Fakultas Teknik Universitas Muhammadiyah Sumatera Utara yang telah memberikan perhatian sehingga tugas akhir ini dapat terselesaikan dengan baik.
3. Bapak Dr. Ade Faisal M. Sc., Ph.D., selaku Wakil Dekan Fakultas Teknik Universitas Muhammadiyah Sumatera Utara
4. Bapak Faisal Irsan Pasaribu, S.T., M.T., selaku Ketua Program Studi Teknik Elektro Universitas Muhammadiyah Sumatera Utara
5. Ibu Elvy Sahnur, S.T., M.Pd., selaku Sekretaris Program Studi Teknik Elektro Universitas Muhammadiyah Sumatera Utara
6. Bapak Muhammad Adam, S.T., M.T., selaku Pembimbing dalam tugas akhir ini yang telah memberikan bimbingannya, masukan dan bantuan sehingga tugas sarjana ini dapat terselesaikan dengan baik.
7. Seluruh Dosen dan Staff Pengajar di Program Studi Teknik Elektro Fakultas Teknik Universitas Muhammadiyah Sumatera Utara

8. Seluruh rekan-rekan seperjuangan mahasiswa Program Studi Teknik Mesin khususnya kelas A1 Pagi yang telah banyak membantu dan memberikan semangat kepada penulis dengan memberikan masukan-masukan yang bermanfaat selama proses perkuliahan maupun dalam penulisan Tugas Akhir ini.
9. Kepada parthner saya yang bernama Frenny Novembrianda Damanik yang sudah menemani saya dan memberi dukungan serta membantu saya selama mengerjakan laporan ini.
10. Seluruh staff Tata Usaha di biro Fakultas Teknik, Universitas Muhammadiyah Sumatera Utara.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari kata sempurna dan tidak luput dari kekurangan, karena itu dengan senang hati dan penuh lapang dada penulis menerima segala bentuk kritik dan saran dari pembaca yang sifatnya membangun demi kesempurnaan penulisan Tugas Akhir ini. Akhir kata penulis mengharapkan semoga Tugas Akhir ini dapat bermanfaat bagi kita semua dan semoga Allah SWT selalu merendahkan hati atas segala pengetahuan yang kita miliki. Amiin ya rabbal alamin.

Wassalamua'alaikum Warahmatullahi Wabarakatuh.

Medan, 2023

Muhammad Fadli Rangkuti

DAFTAR ISI

	Halaman
ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR GAMBAR	vii
DAFTAR TABEL	ix
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan Penelitian.....	2
1.4 Ruang Lingkup Penelitian.....	2
1.5 Manfaat Penelitian.....	3
1.6 Sistematika Penulisan.....	3
BAB II TINJAUAN PUSTAKA	5
2.1 Mikrokontroler Atmega328.....	5
2.2 Fitur ATmega328.....	7
2.3 Konfigurasi Pin ATmega328P.....	7
2.4 Liquid Crystal Display (LCD) 16x2.....	8
2.5 Sensor <i>Proximity</i>	11
2.6 <i>LIGHT EMITTING DIODA</i> (LED).....	15
2.7 Perawatan Mesin Sepeda Motor.....	17
2.8 Ukuran Ban Sepeda Motor.....	18
2.9 Pelumas.....	18
2.10 Aki (Accu).....	19
BAB III RANCANGAN PENELITIAN	21
3.1 Tempat dan Waktu Penelitian.....	21

3.1.1	Rangkaian Umum Alat.....	21
3.2	Perancangan Perangkat Keras.....	23
3.2.1	Rangkaian Sensor Jarak.....	23
3.2.2	Rangkaian Mikrokontroler AT328.....	23
3.2.3	Rangkaian Tombol.....	25
3.2.4	Rangkaian Indikator LED.....	26
3.2.5	Rangkaian LCD.....	28
3.2.6	Rangkaian Regulator Tegangan.....	29
3.3	Perancangan Perangkat Lunak.....	29
3.3.1	Program Utama.....	29
3.3.2	SubRutin Tombol Pilihan.....	31
3.3.3	Subrutin Indikator LED dan Teks Peringatan.....	32
BAB IV HASIL DAN PEMBAHASAN.....		34
4.1	Implementasi Penentu Waktu Servisan.....	34
4.1.1	Pengujian Perangkat Keras.....	35
4.1.2	Pengujian LED indikator dan Teks Peringatan	38
4.1.3	Pengujian Rangkaian Sensor Jarak.....	40
4.1.4	Pengujian Tombol.....	41
4.1.5	Pengujian Daya Alat.....	41
4.2	Analisis Hasil Pengukuran.....	42
4.2.1	Pengukuran Jarak.....	42
4.2.2	Pengujian Perangkat Lunak.....	46
BAB V PENUTUP.....		53
5.1	Kesimpulan.....	53
5.2	Saran.....	53
DAFTAR PUSTAKA.....		54
LAMPIRAN.....		57

DAFTAR GAMBAR

Gambar 2.1	Fitur ATmega328.....	7
Gambar 2.2	Konfigurasi Pin ATmega328P.....	8
Gambar 2.3	Liquid Crystal Display (LCD)16x2.....	9
Gambar 2.4	LCD 2 x 16.....	9
Gambar 2.5	Cara Kerja Sensor Proximity.....	12
Gambar 2.6	Sensor <i>Proximity</i>	15
Gambar 2.7	Konstruksi LED.....	15
Gambar 2.8	Rangkaian LED.....	16
Gambar 3.1	Blok Diagram Penentu Waktu Servis Kendaraan Bermotor.....	21
Gambar 3.2	Rangkaian Mikrokontroler AT328.....	23
Gambar 3.3	Rangkaian Tombol Logika Aktif Rendah.....	25
Gambar 3.4	Rangkaian Tombol Jarak Servis.....	26
Gambar 3.5	Rangkaian LED Sederhana.....	27
Gambar 3.6	Rangkaian Indikator LED.....	28
Gambar 3.7	Rangkaian LCD 4 Bit.....	29
Gambar 3.8	Rangkaian Regulator 5 Volt.....	29
Gambar 3.9	Diagram Alir Program Utama.....	31
Gambar 3.10	Diagram Alir Subrutin Tombol Pilihan.....	32
Gambar 3.11	Diagram Alir Subrutin Indikator LED dan Teks Peringatan....	33
Gambar 4.1	Alat Penentu Waktu Servis Kendaraan Bermotor.....	34
Gambar 4.2	Konstruksi Tampilan LCD serta LED dan Tombol.....	35
Gambar 4.3	Konstruksi Peletakan Sensor Proximity.....	35
Gambar 4.4	Kondisi Alat Saat Dinyalakan.....	36
Gambar 4.5	Mesin OK.....	36
Gambar 4.6	Kondisi “WARNING”.....	37
Gambar 4.7	Kondisi “SERVICE”.....	37
Gambar 4.8	Kondisi “ON”.....	38
Gambar 4.9	Kondisi “OFF”.....	38
Gambar 4.10	Grafik Jarak Tempuh dan Eror Jarak.....	44

Gambar 4.11 Grafik Perbandingan Jarak Tempuh dan Presentasi Eror Jarak 45

DAFTAR TABEL

Tabel 2.1	Simbol Pin Pin LCD.....	10
Tabel 2.2	Daftar Warna LED dan V_{LED}	17
Tabel 2.3	Jenis dan Fungsi Zat Aditifl.....	18
Tabel 4.1	Pengujian LED Indikator dan Teks Peringatan.....	39
Tabel 4.2	Hail Pengujian Rangkaian Sensor dan Komprator.....	40
Tabel 4.3	Pengujian Tombol.....	41
Tabel 4.4	Pengujian Daya.....	42
Tabel 4.5	Perbandingan Jarak Terukur Pada Alat dan Odometer.....	43
Tabel 4.6	Tabel Eror Jarak.....	46

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Sepeda motor adalah salah satu kendaraan bermotor yang terjangkau harganya oleh masyarakat dan sangat efisien untuk membawa barang maupun penumpang ke tempat yang cukup jauh. Oleh karena itu banyak masyarakat menjadikannya sebagai alat transportasi utama. Sepeda motor pun bisa lebih hemat dibandingkan transportasi massal yang belum sempurna terutama bila tinggal di tempat yang kekurangan transportasi massal. Secara umum, pengguna sepeda motor tentu berharap bisa mengendarai sepeda motornya untuk jangka waktu yang panjang. Untuk itu pemilik sepeda motor harus memperhatikan perawatan mesin karena kondisi mesin sepeda motor juga tergantung dari pemeliharaan dan kebiasaan pemiliknya dalam mengendarainya.

Bagian utama kendaraan bermotor yang perlu perhatian khusus dalam hal perawatan adalah bagian mesin. Mesin merupakan penggerak utama dari kendaraan bermotor. Mesin membutuhkan pelumas atau oli. Pelumas merupakan penopang utama dari kerja mesin dan juga menentukan performa serta daya tahan mesin. Fungsi pelumas yaitu melumasi (*lubricating*) seluruh komponen yang bergerak di dalam mesin untuk mencegah terjadinya kontak langsung antar komponen yang terbuat dari logam akibat perputaran mesin. Pelumas memiliki kadar kekentalan (*viskositas*). Kadar kekentalan pada pelumas dapat berkurang karena gesekan yang terjadi pada mesin kendaraan. Apabila kadar kekentalan pelumas sudah tidak baik lagi, maka mesin akan cepat panas. Hal ini disebabkan karena berkurangnya kemampuan pelumas untuk melumasi mesin sehingga gesekan yang ditimbulkan semakin besar. Apabila hal ini dibiarkan terus menerus maka akan terjadi kerusakan pada mesin kendaraan.

Untuk menghindari terjadinya kerusakan mesin, maka harus dilakukan perawatan pada kendaraan bermotor secara rutin. Pada umumnya waktu perawatan kendaraan bermotor berdasarkan dari jarak yang ditempuh oleh kendaraan itu dan jenis pelumas yang dipakai. Ada dua jenis pelumas, yakni

mineral dan sintetis. Pelumas mineral adalah campuran antara minyak bumi yang ditambah zat aditif, sedangkan pelumas sintetis adalah minyak bumi yang melalui proses kimiawi diubah menjadi bahan sintetis. Daya tahan bahan sintesis terhadap panas lebih tinggi sehingga pelumas tidak mudah rusak dan tahan lebih lama terhadap oksidasi.

Oleh karena itu, perawatan mesin sepeda motor khususnya dalam hal penggantian pelumas harus dilakukan sebelum kadar kekentalan pelumas berkurang dan mengakibatkan kerusakan mesin. Dengan kesibukan yang ada, seringkali pengguna kendaraan bermotor melupakan jadwal perawatan yang seharusnya dilakukan. Untuk itu maka diperlukan suatu sistem yang dapat mengingatkan pengguna kendaraan bermotor apabila kendaraan bermotor yang dipakai sudah pada waktunya untuk memperoleh perawatan guna menghindari kerusakan yang timbul pada mesin secara lebih dini.

1.2 Rumusan Masalah

1. Bagaimana Penentuan Waktu Servis Kendaraan Bermotor Menggunakan Mikrokontroler
2. Bagaimana rancang penggunaan sensor proximity pada alat penentuan waktu servis kendaraan bermotor menggunakan mikrokontroler

1.3 Tujuan Penelitian

Adapun tujuan penelitian dari skripsi, yaitu :

1. Bagaimana untuk menghasilkan alat yang dapat berfungsi sebagai penentu waktu servis kendaraan bermotor secara dini sehingga pengguna dapat melakukan perawatan pada kendaraannya tepat waktu
2. Bagaimana untuk mengetahui pemakaian kendaraan dengan perhitungan dari kilometer
3. Bagaimana untuk menjaga kendaraan tersebut supaya dapat digunakan dalam jangka panjang

1.4 Ruang Lingkup Penelitian

Agar perancangan sistem dan pembahasan laporan tentang alat penentu waktu servis kendaraan bermotor ini menjadi lebih spesifik maka diberikan beberapa batasan sebagai berikut:

1. Penelitian ini berfokus pada alat penentuan waktu servis kendaraan bermotor menggunakan mikrokontroler
2. Penelitian ini berfokus pada sensor proximity untuk mengetahui jumlah putaran pada alat penentuan waktu servis kendaraan bermotor menggunakan mikrokontroler
3. Penelitian ini berfokus pada perubahan lampu peringatan dan teks peringatan pada alat penentuan waktu servis kendaraan bermotor menggunakan mikrokontroler

1.5 Manfaat Penelitian

1. Manfaat dari penelitian ini adalah untuk mencegah terjadinya kerusakan yang parah pada mesin kendaraan, khususnya sepeda motor akibat kurangnya perawatan mesin kendaraan bermotor.
2. Manfaat dari penelitian ini adalah untuk menjaga oli agar kendaraan dalam kondisi prima
3. Manfaat dari penelitian ini adalah untuk memberi peringatan jarak tempuh dan service berkala

1.6 Sistematika Penulisan

Adapun sistematika dari penulisan Tugas Akhir ini adalah sebagai berikut :

BAB I PENDAHULUAN

Bab ini membahas mengenai latar belakang, rumusan masalah, tujuan dan Batasan, manfaat penelitian, serta sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini membahas tentang Mikrokontroler AT328, Fitur ATmega328, Konfigurasi Pin ATmega328P, Liquid Crystal Display (LCD) 16x2, Sensor *Proximity* dan jenis-jenis sensor proximity, LIGHT EMITTING DIODE (LED), Perawatan Mesin

Sepeda Motor, Ukuran Ban Sepeda Motor, Pelumas, Aki (Accu).

BAB III RANCANGAN PENELITIAN

Bab ini membahas dimana tempat dan waktu penelitian, Perancangan Perangkat Keras yang terdiri dari (Rangkaian sensor jarak, Rangkaian Mikrokontroler ATmega328, Rangkaian Tombol, Rangkaian Indikator LED, Rangkaian LCD, Rangkaian Regulator Tegangan), Perancangan Perangkat Lunak (Program Utama, Sub Rutin Tombol Pilihan, Subrutin Indikator LED dan Teks Peringatan).

BAB IV HASIL DAN PEMBAHASAN

Bab ini membahas Implementasi penentu waktu servis kendaraan bermotor, pengujian perangkat keras, pengujian LED indikator dan teks peringatan, pengujian rangkaian sensor jarak, pengujian tombol, pengujian daya alat, analisis hasil pengukuran, pengukuran jarak, pengujian perangkat lunak.

BAB V KESIMPULAN DAN SARAN

Bab ini membahas mengenai kesimpulan dan saran Penentuan Waktu Servis Kendaraan Bermotor Menggunakan Mikrokontroler untuk mempermudah menghasilkan alat yang dapat berfungsi sebagai penentu waktu servis kendaraan bermotor secara dini sehingga pengguna dapat melakukan perawatan pada kendaraannya tepat waktu.

BAB II

TINJAUAN PUSTAKA

2.1 Mikrokontroler AT328

Mikrokontroler AT328 adalah mikrokontroler keluaran dari atmel yang mempunyai arsitektur RISC (Reduce Instruction Set Computer) yang mana setiap proses eksekusi data lebih cepat dari pada arsitektur CISC (Completed Instruction Set Computer), ATmega328 merupakan mikrokontroler keluarga AVR 8 bit. Beberapa tipe mikrokontroler yang sama dengan ATmega328 ini antara lain ATmega8535, ATmega16, ATmega32, ATmega328, yang membedakan antara mikrokontroler antara lain adalah, ukuran memori, banyaknya GPIO (pin input/output), peripheral (USART, timer, counter, dll). Dari segi ukuran fisik, ATmega328 memiliki ukuran fisik lebih kecil dibandingkan dengan beberapa mikrokontroler diatas. Namun untuk segi memori dan periperial lainnya ATmega328 tidak kalah dengan yang lainnya karena ukuran memori dan periperialnya relatif sama dengan ATmega8535, ATmega32, hanya saja jumlah GPIO lebih sedikit dibandingkan mikrokontroler diatas.

ATmega328 memiliki 3 buah PORT utama yaitu PORTB, PORTC, dan PORTD dengan total pin input/output sebanyak 23 pin. PORT tersebut dapat difungsikan sebagai input/output digital atau difungsikan sebagai periperial lainnya.

1. Port B Port B merupakan jalur data 8 bit yang dapat difungsikan sebagai input/output. Selain itu PORTB juga dapat memiliki fungsi alternatif seperti di bawah ini.
 - a. ICP1 (PB0), berfungsi sebagai Timer Counter 1 input capture pin.
 - b. OC1A (PB1), OC1B (PB2) dan OC2 (PB3) dapat difungsikan sebagai keluaran PWM (Pulse Width Modulation).
 - c. MOSI (PB3), MISO (PB4), SCK (PB5), SS (PB2) merupakan jalur komunikasi SPI.
 - d. Selain itu pin ini juga berfungsi sebagai jalur pemograman serial (ISP).
 - e. TOSC1 (PB6) dan TOSC2 (PB7) dapat difungsikan sebagai sumber clock external untuk timer. f. XTAL1 (PB6) dan XTAL2 (PB7) merupakan

sumber clock utama mikrokontroler.

- 2 Port C Port C merupakan jalur data 7 bit yang dapat difungsikan sebagai input/output digital. Fungsi alternatif PORTC antara lain sebagai berikut.
 - a. ADC6 channel (PC0,PC1,PC2,PC3,PC4,PC5) dengan resolusi sebesar 10 bit. ADC dapat kita gunakan untuk mengubah input yang berupa tegangan analog menjadi data digital
 - b. I2C (SDA dan SDL) merupakan salah satu fitur yang terdapat pada PORTC. I2C digunakan untuk komunikasi dengan sensor atau device lain yang memiliki komunikasi data tipe I2C seperti sensor kompas, accelerometer nunchuck.
- 3 Port D Port D merupakan jalur data 8 bit yang masing-masing pin-nya juga dapat difungsikan sebagai input/output. Sama seperti Port B dan Port C, Port D juga memiliki fungsi alternatif dibawah ini.
 - a. USART (TXD dan RXD) merupakan jalur data komunikasi serial dengan level sinyal TTL. Pin TXD berfungsi untuk mengirimkan data serial, sedangkan RXD kebalikannya yaitu sebagai pin yang berfungsi untuk menerima data serial.
 - b. Interrupt (INT0 dan INT1) merupakan pin dengan fungsi khusus sebagai interupsi hardware. Interupsi biasanya digunakan sebagai selaan dari program, misalkan pada saat program berjalan kemudian terjadi interupsi hardware/software maka program utama akan berhenti dan akan menjalankan program interupsi.
 - c. XCK dapat difungsikan sebagai sumber clock external untuk USART, namun kita juga dapat memanfaatkan clock dari CPU, sehingga tidak perlu membutuhkan external clock.
 - d. T0 dan T1 berfungsi sebagai masukan counter external untuk timer 1 dan timer 0.
 - e. AIN0 dan AIN1 keduanya merupakan masukan input untuk analog comparator

2.2 Fitur ATmega328



Gambar 2.1 Mikrokontroler ATmega328

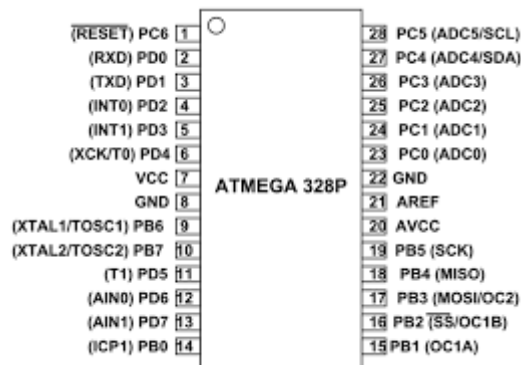
ATmega328 adalah mikrokontroler keluaran dari atmel yang mempunyai arsitektur RISC (Reduce Instruction Set Computer) yang mana setiap proses eksekusi data lebih cepat dari pada arsitektur CISC (Completed Instruction Set Computer).

Mikrokontroler ini memiliki beberapa fitur antara lain:

1. Memiliki EEPROM (Electrically Erasable Programmable Read Only Memory) sebesar 1KB sebagai tempat penyimpanan data semi permanen karena EEPROM tetap dapat menyimpan data meskipun catu daya dimatikan.
2. Memiliki SRAM (Static Random Access Memory) sebesar 2KB.
3. Memiliki pin I/O digital sebanyak 14 pin 6 diantaranya PWM (Pulse Width Modulation) output.
4. 32 x 8-bit register serba guna.
5. Dengan clock 16 MHz kecepatan mencapai 16 MIPS.
6. 32 KB Flash memory dan pada arduino memiliki bootloader yang menggunakan 2 KB dari flash memori sebagai bootloader.
7. 130 macam instruksi yang hampir semuanya dieksekusi dalam satu siklus clock.

2.3 Konfigurasi Pin ATmega328P

ATmega328P mempunyai kaki standar 28 pin yang mempunyai fungsi masing-masing. Untuk lebih jelasnya tentang konfigurasi pin ATmega328P dapat dilihat pada Gambar 2.2 seperti berikut:



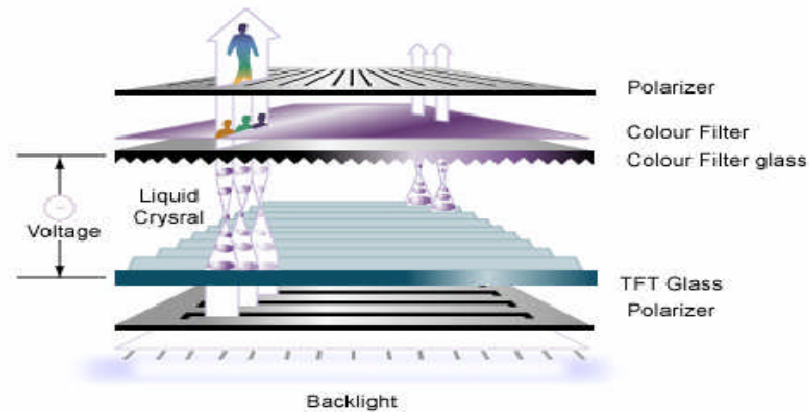
Gambar 2.2 Konfigurasi pin ATMEGA328

Adapun rincian dan fungsi dari susunan pin ATMEGA328P adalah sebagai berikut :

1. VCC merupakan pin yang berfungsi sebagai masukan catu daya.
2. GND merupakan pin Ground.
3. Port B (PB0 – PB7) merupakan pin masukan/keluaran dua arah (full duplex) dan dengan masing-masing port memiliki fungsi khusus.
4. Port C (PC0 – PC6) merupakan pin masukan/keluaran dua arah (full duplex) dan dengan masing-masing port memiliki fungsi khusus.
5. Port D (PD0 – PD7) merupakan pin masukan/keluaran dua arah (full duplex) dan dengan masing-masing port memiliki fungsi khusus.
6. RESET merupakan pin yang digunakan untuk mengatur ulang mikrokontroler.
7. XTAL1 dan XTAL2, merupakan pin masukan external clock.
8. AVCC merupakan pin masukan tegangan untuk ADC (Analog-Digital Converter).
9. AREF merupakan pin masukan tegangan referensi untuk ADC.

2.4 Liquid Crystal Display (LCD) 16x2

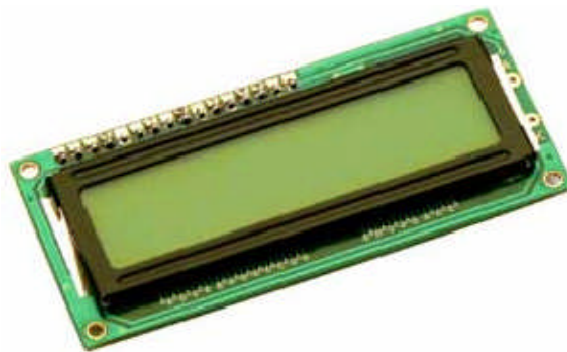
LCD adalah komponen yang berfungsi untuk menampilkan (display) suatu karakter. Bahan utama LCD berupa *Liquid Crystal*. Apabila diberi arus listrik sesuai dengan jalur yang telah dirancang pada konstruksi LCD, *Liquid Crystal* akan berpendar menghasilkan suatu cahaya dan cahaya tersebut akan membentuk suatu karakter tertentu. Gambar konstruksi LCD disajikan pada Gambar 2.3.



Gambar 2.3 Konstruksi LCD

LCD yang sering digunakan adalah jenis LCD M1632. M1632 merupakan modul LCD dengan tampilan 2 x 16 (2 baris, 16 kolom) dengan konsumsi daya rendah. Modul tersebut dilengkapi dengan mikrokontroler yang didesain khusus untuk mengendalikan LCD. Mikrokontroler HD44780 buatan Hitachi yang berfungsi sebagai pengendali LCD memiliki CGROM (*Character General Read OnlyMemory*), CGRAM (*Character General Random Access Memory*), dan DDRAM (*Display Data Random Access Memory*).

LCD bertipe ini memungkinkan pemrogram untuk mengoperasikan komunikasi data secara 8 bit atau 4 bit. Jika menggunakan jalur data 4 bit akan ada 7 jalur data (3 untuk jalur kontrol & 4 untuk jalur data). Jika menggunakan jalur data 8 bit maka akan ada 11 jalur data (3 untuk jalur kontrol & 8 untuk jalur data). Tiga jalur kontrol ke LCD ini adalah EN (*Enable*), RS (*RegisterSelect*) dan R/W (*Read/Write*). Gambar dari LCD 2 x 16 disajikan pada Gambar 2.4.



Gambar 2.4. LCD 2 x 16

LCD jenis M1623 memiliki jumlah pin sebanyak 16 yang memiliki fungsi berbeda-beda. Simbol pin-pin tersebut disajikan pada Tabel 2.1.

Tabel 2.1 Simbol Pin-pin LCD

Nomor Pin	Simbol	Nomor Pin	Simbol
9	DB2	9	DB2
10	DB3	10	DB3
11	DB4	11	DB4
12	DB5	12	DB5
13	DB6	13	DB6
14	DB7	14	DB7
15	A	15	A
16	K	16	K

Fungsi pin :

1. V_{lcd}, merupakan pin yang digunakan untuk mengatur tebal tipisnya karakter yang tertampil dengan cara mengatur tegangan masukan. Ditentukan sebesar 10K hingga 20K.
2. DB0 s/d DB7, merupakan jalur data yang dipakai untuk menyalurkan kode ASCII maupun perintah pengatur LCD.
3. RS (*Register Select*), merupakan pin yang dipakai untuk membedakan jenis data yang dikirim ke LCD. Jika RS berlogika '0', maka data yang dikirim adalah perintah untuk mengatur kerja LCD. Jika RS berlogika '1', maka data yang dikirimkan adalah kode ASCII yang ditampilkan.
4. R/W (*Read/Write*), merupakan pin yang digunakan untuk mengaktifkan pengiriman dan pengembalian data ke dan dari LCD. Jika R/W berlogika '1', maka akan diadakan pengambilan data dari LCD. Jika R/W berlogika '0', maka akan diadakan pengiriman data ke LCD.
5. E (*Enable*), merupakan sinyal sinkronisasi. Saat E berubah dari logika '1' ke '0', maka data di DB0 s/d DB7 akan diterima atau diambil dari port mikrokontroler.
6. A (*Anoda*) dan K (*Katoda*), merupakan pin yang digunakan untuk menyalakan *backlight* dari layar LCD.

2.5 Sensor Proximity

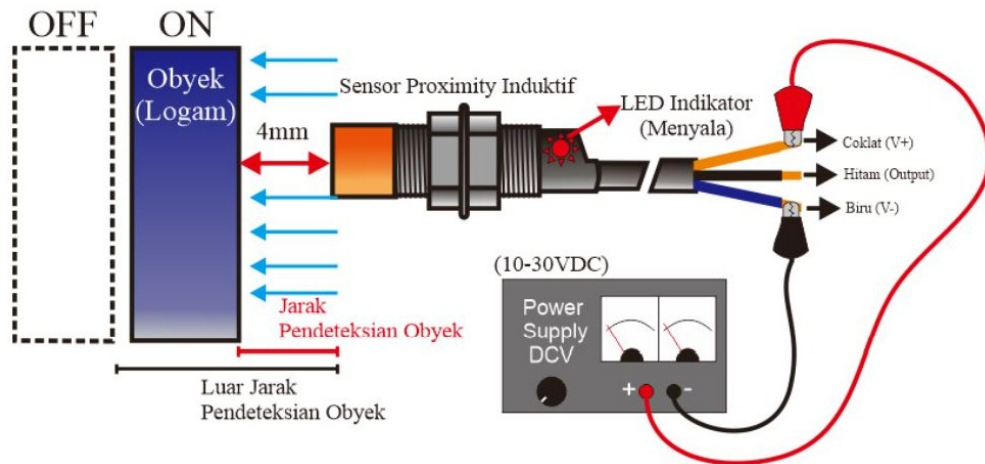
Sensor Proximity adalah alat atau perangkat yang dapat mendeteksi perubahan jarak pada suatu benda. Namun proses tersebut terjadi dengan tanpa adanya kontak fisik. Sensor proximity di Indonesia juga familiar dengan istilah sensor jarak. Dalam prosesnya, sensor proximity memakai pengantar radiasi elektromagnetik. Inilah yang membuat perangkat bisa mendeteksi keberadaan benda atau kondisinya meskipun tanpa ada kontak fisik. Contoh pemanfaatan sensor proximity sering digunakan untuk kepentingan yang sangat beragam. Diantaranya ada yang digunakan untuk mendeteksi bahan. Selain itu, ada pula yang digunakan untuk mendeteksi lingkungan yang berbeda. Pengaplikasiannya yaitu seperti digunakan pada smartphone ataupun berbagai perangkat elektronik yang lainnya. Sensor proximity juga sering digunakan untuk beragam mesin industri. Contohnya seperti mesin plastik, mesin cetak, mesin pengolah logam dan lain sebagainya.

Cara kerja Sensor proximity sering disingkat sebagai P-Sensor. Seperti yang kita bahas di atas, fungsi sensor proximity yakni sebagai detektor keberadaan sebuah benda atau objek. Lalu bagaimana cara kerja sensor proximity?

Adapun penjelasan tentang cara kerja proximity adalah sebagai berikut:

1. Untuk melakukan deteksi pergerakan objek di sekitarnya, ternyata proximity sensor memanfaatkan adanya radiasi elektromagnetik (medan elektromagnetik). Dimana sensor jarak tersebut juga mengatur interval nominal agar bisa melaporkan objek yang terdeteksi.
2. Jadi, saat terdapat benda atau objek mendekati sensor maka akan tercipta sebuah sinyal. Benda atau objek tersebut bisa bersifat logam maupun non logam. Lalu kemudian signal tersebut akan dihubungkan dengan berbagai sistem otomatisasi.
3. Sensor Proximity terdiri dari device elektronik solid state yang tampilannya dalam kondisi terbungkus. Dengan keadaan terbungkus, maka akan melindungi perangkat tersebut dari getaran, korosif, ataupun cairan dan imiawi yang berlebihan.

4. Dalam proses kerjanya, sensor gerak ini dapat diandalkan. Selain nilai akuratnya yang tinggi, sensor tersebut juga dapat digunakan untuk mendeteksi benda-benda yang sangat kecil sekalipun.



Gambar 2.5 Cara Kerja Sensor Proximity

Jenis-Jenis Sensor Proximity

Sensor Proximity ini dibagi menjadi empat jenis yang berbeda. Adapun penjelasan lengkap mengenai jenis-jenis Sensor Proximity adalah sebagai berikut :

1. Induktif Proximity Sensor (Sensor Proximity Induktif)

Sensor Proximity Induktif adalah sensor yang bisa melakukan deteksi jika terdapat benda logam besi maupun non-ferro di sekitarnya. Fungsinya adalah untuk mendeteksi peralatan logam, menghitung benda logam, hingga aplikasi posisi. Dalam penggunaannya, Sensor Proximity jenis ini sering digunakan untuk pengganti saklar mekanis. Selain kuat dan handal, alat tersebut juga dipercaya bisa membuat kinerja yang lebih cepat dan akurat. Terutama jika dibandingkan dengan saklar mekanis biasa. Untuk mendapatkan medan elektromagnetik dengan frekuensi tinggi, pada umumnya sensor kedekatan induktif diproduksi dari koil atau inti ifrit. Karena hal tersebut pula, sensor ini sering digunakan untuk mendeteksi logam dalam mesin. Contoh pengaplikasian lainnya yakni digunakan sebagai perangkat otomasi.

2. Capacitive Proximity Sensor (Sensor Jarak Kapasitif)

Capacitive proximity sensor adalah jenis sensor yang bisa mendeteksi beberapa objek dalam satu waktu. Diantaranya yaitu yang berbentuk gerakan, komposisi kimia, level tekanan, level fluida dan juga komposisi lainnya. Sensor proximity kapasitif ini bisa melakukan deteksi menggunakan dielektrik dengan kapasitas rendah. Misalnya saja untuk mendeteksi adanya objek berbahan plastik, kaca, atau bahkan bahan dielektrik yang berupa cairan sekalipun. Sensor kedekatan kapasitif menghasilkan medan elektrostatik. Dalam proses kerjanya, sensor tersebut juga memiliki substansial yang mirip dengan sensor induktif. Eleman aktif yang terdapat pada sensor kapasitif adalah terdiri dari dua buah elektroda logam. Yang mana elektroda ini dimasukan dalam rangkaian resonansi dengan frekuensi tinggi. Jadi ketika ada objek yang mendekati permukaan sensor. Maka bidang elektrostatik dalam alat tersebut akan mendeteksi adanya objek.

Prinsip kerja sensor proximity capacitive adalah sebagai berikut:

- a. Objek hanya dapat dideteksi dengan jarak tertentu, yakni maksimal 2 cm.
- b. Metode pemasangan dilakukan dengan cara dibenamkan pada metal (flush) dan juga didekatkan pada objek-objek disekitarnya (non flush).
- c. Medan elektrostatik akan mendeteksi beberapa jenis objek, baik logam maupun non logam

3. Ultrasonic Proximity Sensor (Sensor Jarak Ultrasonik)

Ultrasonic proximity sensor adalah sensor kedekatan ultrasonic yang memiliki sistem operasi yang menyerupai sonar atau radar, yaitu dengan menghasilkan gelombang yang berupa gema, lalu kemudian akan dipantulkan ketika ada objek yang mendekatinya. Jenis sensor ini sering digunakan untuk berbagai keperluan. Diantaranya digunakan untuk mengukur jarak benda, proses otomasi pabrik dan lain sebagainya. Untuk dapat menghitung kedekatan jarak objek, sensor ultrasonik terlebih dahulu harus menghitung penerimaan sinyal dan juga transmisi sinyal.

4. Photoelektrik Proximity Sensor (Sensor Jarak Fotolistrik)

Photoelektrik Proximity Sensor atau Sensor Jarak Fotolistrik merupakan jenis sensor yang menggunakan media elemen fotolistrik untuk dapat mendeteksi objek. Sensor dengan jenis fotolistrik memiliki beberapa keunggulan. Salah satu keunggulannya yakni dari segi jarak sensor yang mampu mendeteksi objek jauh sekalipun.

Sensor Proximity Berdasarkan Tegangan Kerjanya

Berdasarkan tegangan kerjanya, jenis sensor proximity dibedakan menjadi dua yaitu:

Sensor Proximity Berdasarkan Tegangan Kerjanya

Berdasarkan tegangan kerjanya, jenis sensor proximity dibedakan menjadi dua yaitu:

- a. Proximity sensor dengan tegangan kerja 10 sampai dengan 30 VDC.
- b. Proximity sensor dengan tegangan kerja 24 sampai dengan 240 VAC.

Sensor Proximity Berdasarkan Output (Nilai Keluaran)

Berdasarkan nilai outputnya, jenis sensor proximity adalah sebagai berikut:

- a. Output NPN NO
- b. Output NPN NC
- c. Output PNP NO
- d. Output PNP NC
- e. Output PNP NO+NC
- f. Output NPN NO+NC
- g. Output NO
- h. Output NC
- i. Output NO+NC
- j. Output Lights On/Off

Sensor proximity adalah alat yang memiliki peranan yang cukup penting. Itulah mengapa beberapa alat dan teknologi menggunakan jenis sensor yang satu ini. Pada sensor proximity, terdapat beberapa komponen-komponen yang menyusunnya. Komponen-komponen yang terdapat pada sensor proximity diantaranya adalah oscilator, coil hingga output. Sebagai alat pendeteksi, sensor

proximity memiliki beberapa kelebihan dan kekurangan. Apa saja kelebihan dan kekurangan sensor proximity?

1. Kelebihan Sensor Proximity

Kelebihan sensor proximity adalah seperti:

- a. Memiliki nilai keakuratan yang tinggi.
- b. Tingkat switch yang tinggi.
- c. Sensor dapat bekerja pada tipe lingkungan yang keras.
- d. Dapat digunakan sesuai kepentingan. Baik untuk mengenali objek ataupun mendeteksi keadaan

2. Kekurangan Sensor Proximity

Adapun kekurangan sensor proximity adalah sebagai berikut:

- a. Objek yang dideteksi terbatas.
- b. Jangkauan jarak sensor juga cukup terbatas. Karena posisinya harus dekat dengan alat.
- c. Ukuran, bentuk, dan jenis objek juga mempengaruhi kemampuan sensor proximity

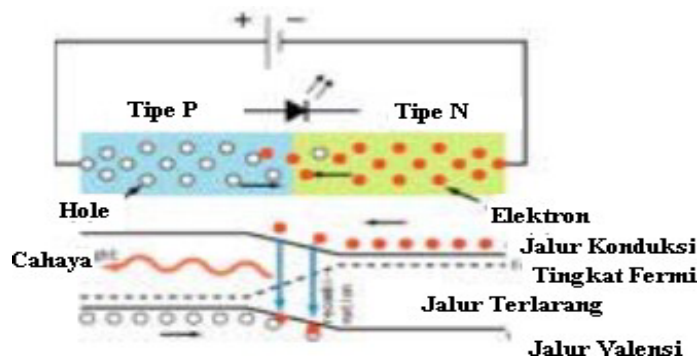


Gambar 2.6. Sensor *Proximity*

2.6 LIGHT EMITTING DIODA (LED)

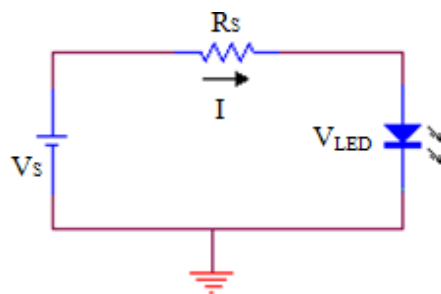
LED (*Light Emitting Diode*) secara konstruksi terbuat sebagai dioda PN *junction* bahan tipe P dan tipe N. Yang membedakan keduanya adalah bahan yang digunakan. Dioda PN *junction* atau yang biasa disebut dioda saja terbuat dari bahan Silikon (Si) atau Germanium (Ge), aliran arusnya dapat melalui *trapping level* yang biasa dinamakan tingkat *Fermi*. Sedangkan LED terbuat dari bahan GaAs, GaP atau GaAsP yang mempunyai sifat *direct gap*. Artinya untuk mengalirkan arus, elektron harus berpindah dari tingkat jalur konduksi langsung

ke jalur valensi. Kontruksi LED dibawah ini :



Gambar 2.7 Kontruksi LED

LED mempunyai penurunan tegangan lazimnya dari 1,5 V sampai 2,5 V untuk arus diantara 10 dan 50 mA. Penurunan tegangan yang tepat tergantung dari arus LED, warna, kelonggaran, dan sebagainya. Pancaran sinar LED tergantung dari arusnya. Idealnya, cara terbaik untuk mengendalikan pancaran sinar ialah dengan menjalankan LED dengan sumber arus. Gambar dari rangkaian LED disajikan pada Gambar 2.8.



Gambar 2.8 Rangkaian LED

Dari rangkaian Gambar 2.8, dapat ditentukan persamaan arus I yang melewati LED, yaitu :

$$I = \frac{(V_S - V_{LED})}{R_S}$$

Dengan :

V_{LED} = penurunan tegangan LED (Volt)

V_S = Tegangan sumber (Volt)

R_S = Resistor yang tersusun seri dengan LED (Ohm)

I = Arus (Ampere)

Makin besar tegangan sumber, makin kecil pengaruh V_{LED} . Dengan kata lain V_s yang besar menghilangkan pengaruh perubahan tegangan V_{LED} . Biasanya, arus LED ada diantara 10 mA sampai 50 mA. Untuk LED dengan warna yang berbeda, maka juga akan mempunyai V_{LED} yang berbeda pula. Tabel 2.2 merupakan tabel daftar V_{LED} sesuai dengan warna nyala LED.

Tabel 2.2 Daftar Warna LED dan V_{LED}

Warna LED	V_{LED}
Infrared	1,6 V
Merah	1,8 – 2,1 V
Jingga	2,2 V
Kuning	2,4 V
Hijau	2,6 V
Biru	3,0 – 3,5 V
Putih	3,0 – 3,5 V
Ultraviolet	3,5 V

2.7 Perawatan Mesin Sepeda Motor

Sepeda motor adalah sebuah mesin yang terbuat dari ribuan komponen. Secara umum, pengguna sepeda motor tentu berharap bisa mengendarai sepeda motornya untuk jangka panjang. Untuk itu pemilik sepeda motor harus memperhatikan perawatan mesin karena kondisi mesin sepeda motor juga tergantung dari pemeliharaan dan kebiasaan pemiliknya dalam mengendarainya. Pemeliharaan mesin dan pelumas sepeda motor amatlah vital. Hal ini besar pengaruhnya terhadap jangka waktu penggantian pelumas.

Banyak faktor yang mempengaruhi pemeliharaan mesin dan penggantian pelumas. Salah satu faktor tersebut adalah jarak yang telah tempuh oleh sepeda motor tersebut. Semakin jauh jarak yang ditempuh maka semakin perlu mesin sepeda motor mendapatkan perawatan dan penggantian pelumas. Dengan mendapatkan perawatan mesin serta penggantian pelumas secara tepat waktu dapat mencegah kerusakan yang parah pada sepeda motor. Periode penggantian

pelumas mesin sepeda motor sesuai rekomendasi pabrikan sepeda motor adalah sekitar 2000 km hingga 2500 km, namun juga bisa mencapai 3000 km. Hal ini tergantung dari jenis pelumas yang digunakan juga.

2.8 Ukuran Ban Sepeda Motor

Ban sepeda motor mempunyai ukuran yang bermacam-macam. Sebagai indikator dari spesifikasi teknisnya, produsen ban menuliskan uraian (kode) beberapa spesifikasi teknis ban tersebut di dinding ban (*side wall tyre*). Kode ban motor didesain untuk memberitahu apa yang perlu diketahui tentang ban tersebut. Ada dua macam sistem kode ban sepeda motor, yaitu sistem metrik (*metric system*) dan sistem inchi (*inch system*).

2.9 Pelumas

Pelumas adalah penopang utama dari kerja sebuah mesin. Bukan itu saja, bahkan pelumas juga menentukan performa dan daya tahan mesin. Semakin baik kualitas pelumas yang digunakan, semakin baik pula performa dan daya tahan mesin. Fungsi pelumas bukan hanya sebagai pelumas saja, melainkan juga sebagai pendingin dan pembersih mesin. Sebagai pelumas, pelumas melumasi (*lubricating*) seluruh komponen yang bergerak di dalam mesin untuk mencegah terjadinya kontak langsung antar komponen yang terbuat dari logam. Dalam hal ini, unsur kekentalan (*viskositas*) sangat penting. Dahulu, pelumas tidak memakai aditif. Kelemahannya, dipergunakan 1500 kilometer sudah perlu diganti. Maka untuk memperbaikinya ditambah zat aditif tersebut. Jenis dan fungsi bahan aditif dapat dilihat pada Tabel 2.3

Tabel 2.3 Jenis dan Fungsi Zat Aditif

Jenis	Fungsi
Peningkat angka viskositas	Mengurangi perubahan terhadap suhu, iritkan bahan bakar, hemat pemakaian pelumas, mudah start dalam keadaan dingin.
Deterjen, Dispersan	Menjaga agar kotoran karbon dan sebagainya tetap tersuspensi dalam pelumas, sampai pada saat penggantian tiba.
Senyawa Alkali	Menetralkan asam, mencegah korosi karena

	serangan asam.
Anti-Aus, Pengubah Gesekan	Membentuk film pelindung pada onderdil mesin. Mengurangi aus, menghemat bahan bakar.
Inhibitor Oksidasi	Mencegah/mengontrol oksidasi pelumas, pembentukan,pernis/sludge dan zat organik korosif. Membatasi kenaikan viskositas yang terjadi selama pemakaian.
Inhibitor Karet	Mencegah karat pada permukaan logam, membentuk film permukaan atau menetralkan asam.
Penekan Titik Tuang	Memperendah titik beku pelumas sehingga mudah mengalir pada suhu rendah.
Zat Anti Buih	Mengurangi buih dalam mesin

Pelumas modern merupakan cairan amat kompleks, sebagaimana fungsinya yaitu:

1. Menghalangi kontak logam ke logam yang mempercepat aus, menghemat hilangnya energi karena gesekan berlebihan.
2. Mencegah serangan kimia atas logam.
3. Mendispersi pengotor, agar tidak menjadi endapan yang berbahaya bagi mesin.
4. Mendinginkan onderdil mesin.
5. Menjaga agar dirinya sendiri tidak mengurai atau rusak.

2.10 Aki (Accu)

Accu adalah salah satu jenis baterai isi ulang, yang berfungsi memasok energi listrik ke sebuah kendaraan bermotor untuk keperluan *Starting* (menyalakan motor stater), *Lighting* (Lampu) dan *Ignition* (pengapian). Keberadaannya merupakan salah satu langkah untuk memberikan kenyamanan penggunaan kendaraan. Accu ini berasal dari kata accumulator yaitu alat untuk mengakumulasi sesuatu, dalam hal ini muatan listrik dengan melakukan penyimpanan saat kekurangan dan pengaliran pada saat tersedia, dipergunakan. Pada umumnya, accu menyediakan tegangan sebesar 12 volt. Tegangan ini didapat dengan cara menghubungkan enam *sel galvanik* secara “seri”. Setiap sel

menyediakan 2,1 volt, jadi apabila di *charge* penuh, akan menghasilkan 2,1 volt x 6 sel = 12,6 volt. Kapasitas baterai adalah jumlah ampere jam (Ah = kuat arus/Ampere x waktu/hour), berarti baterai dapat memberikan/mensuplai sejumlah isinya secara rata-rata sebelum tiap selnya menyentuh tegangan/voltase turun (*drop voltage*) yaitu sebesar 1,75 V atau tegangan aki telah mencapai 10,5 volt (1,75volt x 6). Daya yang mampu disediakan oleh aki dapat diperoleh dengan persamaan 2.6.

$$P = V \times I$$

Dengan :

P = Daya aki (Watt)

V = Tegangan aki (Volt)

I = Arus aki (Ampere/jam)

Dengan demikian aki dengan tegangan 'V' dan arus sebesar 'I' mampu mensuplai alat dengan daya 'P' selama satu jam penuh. Sedangkan daya tahan aki untuk dapat mensuplai suatu alat dengan daya tertentu hingga mencapai tegangan minimumnya dapat diperoleh dengan persamaan 2.7.

BAB III

RANCANGAN PENELITIAN

3.1 Tempat Dan Waktu Penelitian

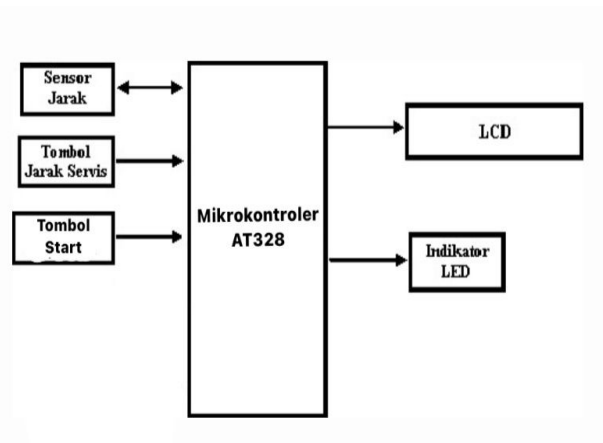
Penelitian tugas akhir ini dilakukan di Jalan Brigjen Zein Hamid Gang Johar, Kec. Medan Johor, Kel. Suka Maju, Titi Kuning Medan Sumatera Utara.

Perancangan dan pembuatan Alat serta riset data terhadap tugas akhir ini berlangsung dimulai dari April 2023 sampai dengan Desember 2023.

3.1.1 Rancangan Umum Alat

Rancangan umum alat Penentu Waktu Servis Kendaraan Bermotor dibagi dalam dua bagian besar, yaitu :

1. Perancangan perangkat keras ditunjukkan pada Gambar 3.1 yang terdiri dari: rangkaian mikrokontroler AT328 dan sistem pendukung, sensor jarak, rangkaian tombol servis, indikator LED dan rangkaian LCD sebagai penampil.
2. Perancangan perangkat lunak yang terdiri dari program utama, pemilihan jarak servis, penghitung jarak dan penampil LCD.



Gambar 3.1 Blok Diagram Penentu Waktu Servis Kendaraan Bermotor

Cara kerja dari setiap blok pada diagram perancangan Gambar 3.1 adalah sebagai berikut :

1. Tombol jarak servis digunakan sebagai masukan awal yang ditentukan oleh pemakai. Tombol tersebut masing-masing terdiri dari 2 buah *push button*, yaitu *up* dan *down*.

2. Tombol *start* digunakan waktu pertama kali untuk mulai mengaktifkan alat supaya dapat mulai menghitung jarak yang ditempuh. Tombol *start* ditekan setelah pengguna mengeset nilai jarak tempuh yang sesuai.
3. Sensor jarak akan menghitung putaran roda dengan mendeteksi putaran cakram pada sepeda motor. Apabila terjadi putaran, maka jarak yang ditempuh akan bertambah sesuai dengan perbandingan antara putaran dan jarak. Sensor jarak ini menggunakan sensor *proximity*.
4. Mikrokontroler akan membandingkan jarak yang telah ditempuh dengan jarak tempuh terbaik kendaraan bermotor untuk segera dilakukan perawatan.
5. Indikator LED dan teks peringatan LCD akan menyala dengan spesifikasi berikut :
 - a. LED 1 (berwarna hijau) menyala
Keadaan jika jarak tempuh belum mencapai 90% dari jarak servis yang ditentukan ($\text{jarak tempuh} < 90\% \times \text{jarak servis}$).
Teks peringatan LCD akan menampilkan “Mesin OK”.
 - b. LED 2 (berwarna kuning) menyala
Keadaan jika jarak tempuh telah mencapai 90% dari jarak servis hingga mencapai jarak servis yang telah ditentukan ($90\% \times \text{jarak servis} \leq \text{jarak tempuh}$).
Teks peringatan LCD akan menampilkan “Warning”.
 - c. LED 3 (berwarna merah) menyala
Keadaan jika jarak tempuh telah melampaui jarak servis yang telah ditentukan ($\text{jarak tempuh} > \text{jarak servis}$).
Teks peringatan LCD akan menampilkan “Service”.
6. Tombol riset digunakan pada saat kendaraan bermotor diberikan perawatan serta untuk mengembalikan jarak yang telah ditempuh menjadi keadaan awal atau menjadi 0 km kembali.

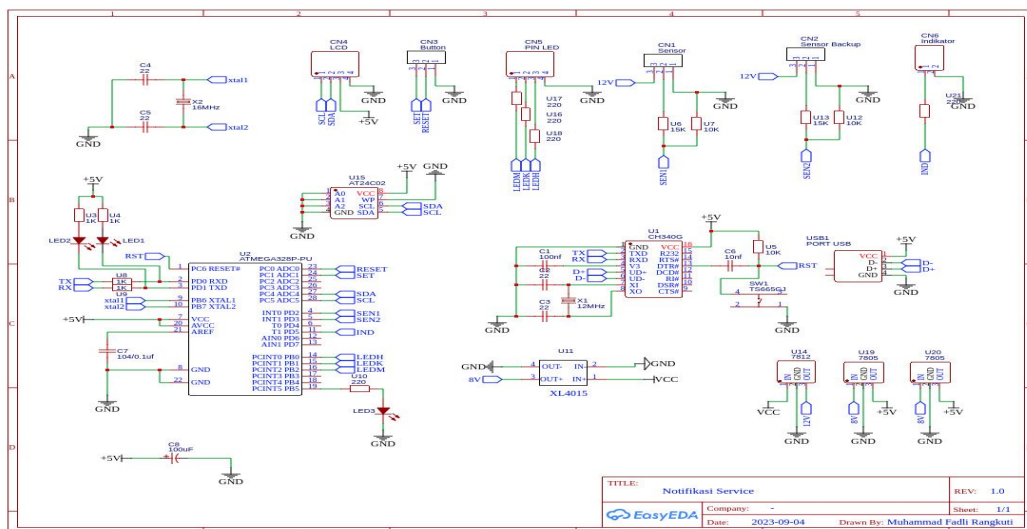
3.2 Perancangan Perangkat Keras

3.2.1 Rangkaian Sensor Jarak

Rangkaian sensor jarak yang digunakan terdiri dari sensor *proximity*. *Proximity* dipasang pada bagian ban depan sepeda motor untuk mendeteksi jumlah putaran dari cakram rem depan dan mengubahnya menjadi pulsa-pulsa. Tiap putaran akan menghasilkan satu pulsa yang kemudian akan dihitung oleh mikrokontroler.

3.2.2 Rangkaian Mikrokontroler AT328

Seluruh program untuk sistem kendali penentu waktu servis kendaraan bermotor ini dikendalikan oleh mikrokontroler AT328. Rangkaian mikrokontroler AT328 disajikan pada Gambar 3.2.



Gambar 3.2 Rangkaian Mikrokontroler AT328

Pada gambar 3.2 untuk rangkaian osilator menggunakan Kristal 12MHz dan dua buah kapasitor 33pF sehingga besarnya periode untuk menjalankan satu instruksi ke instruksi berikutnya dapat dihitung menggunakan persamaan 2.1. Untuk mengembalikan kondisi kerja mikrokontroler pada posisi awal dibutuhkan logika 1 pada pin RST. Bila besar $R_1 = 10K$ dan mikrokontroler AT328 mempunyai logika 1 antara rentang tegangan 3V sampai dengan 5V maka besarnya R_2 dapat dihitung dengan menggunakan persamaan 2.2, yaitu:

1. Pada saat $V_{rst} = 3V$

$$3 = \frac{10000}{10000 + R_2} \times 5$$

$$R_2 = 6,66 \text{ K}\Omega$$

2. Pada saat $V_{rst} = 5V$

$$5 = \frac{10000}{10000 + R_2} \times 5$$

$$R_2 = 0$$

Maka pada perancangan dipilih nilai resistor R_2 yang dekat dengan 0 dan dipilih sebesar 20Ω . Rangkaian reset dirancang agar reset terjadi pada saat sistem mendapatkan catu daya pada pertama kali. Untuk membuat rangkaian reset, ada hal-hal yang harus diperhatikan, yaitu :

1. Lebar pulsa untuk sinyal reset harus lebih besar dari 24 periode osilasi atau duasiklus mesin. Dimana satu siklus mesin (t) = $1/fosc$.
2. *Rise time* dari catu daya (V_{cc}) tidak lebih dari 1 milidetik.
3. Waktu *start up* dari osilator tidak lebih dari 10 milidetik
4. Tegangan reset yang diperlukan pada pin mikrokontroler adalah lebih dari 3,5 volt.

Dari ketentuan di atas, dengan menggunakan frekuensi osilator 12 MHz, maka lebar pulsa reset minimum harus lebih besar dari :

$$t = \left(10^{-2} + \frac{24}{f_{osc}}\right) s$$

$$t = \left(10^{-2} + \frac{24}{12 \times 10^6}\right) s$$

$$t = 10,002 \text{ ms}$$

Dengan tegangan reset minimum sebesar 3,5 volt, maka nilai resistor dan kapasitor pada rangkaian reset dapat diperoleh dengan :

$$\begin{aligned} VRST &= V_{cc}(1 - e^{-t/(R1.C3)}) \\ 3,5 &= 5(1 - e^{-0,010002/(R1.C3)}) \\ 3,5 &= 5 - 5 \cdot e^{-0,010002/(R1.C3)} \\ 5 \cdot e^{-0,010002/(R1.C3)} &= 5 - 3 \end{aligned}$$

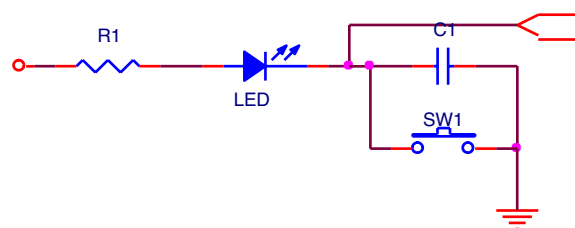
$$\begin{aligned}
e^{-0,010002/(R1.C3)} &= 0,3 \\
(e^{-0,010002/(R1.C3)}) &= \ln(0,3) \\
-0,010002/(R1.C3) &= -1,203972 \\
R1.C3 &= 0,010002/1,203972 \\
R1.C3 &= 8,307502.10^{-3}
\end{aligned}$$

Dengan menggunakan R1 sebesar 10kΩ, maka pada perancangan besar nilai kapasitor C3 sebesar 83 uF.

3.2.3 Rangkaian Tombol

Pengguna dapat memilih jarak servis dan tinggi ban kendaraan dengan cara menekan tombol pilihan. Masing-masing menggunakan 2 buah tombol pilihan yaitu *up* dan *down*. Tombol pilihan ini akan menghitung cacahan dari setiap penekanan yang selanjutnya akan diproses di dalam mikrokontroler.

Ada 2 jenis tombol yang dapat digunakan pada mikrokontroler yaitu tombol dengan logika aktif rendah dan logika aktif tinggi. Pada perancangan ini digunakan logika aktif rendah. Rangkaian tombol dengan logika rendah disajikan pada Gambar 3.3



Gambar 3.3 Rangkaian Tombol Logika Aktif Rendah

Kapasitor yang ada pada rangkaian digunakan untuk mengantisipasi terjadinya *bouncing* tegangan saat saklar *push button* ditekan atau dilepas. Sedangkan LED berfungsi sebagai indikator penanda cacahan setiap penekanan. Besar arus maksimum untuk LED adalah sebesar 50mA, maka besar nilai resistor untuk I_{max} dapat dihitung dengan persamaan 2.4 :

$$R = \frac{(VS - V_{LED})}{I_{max}}$$

$$R = \frac{I_{\max}}{(5-1.5)}$$

$$R = \frac{50mA}{50mA}$$

$$R = 70\Omega$$

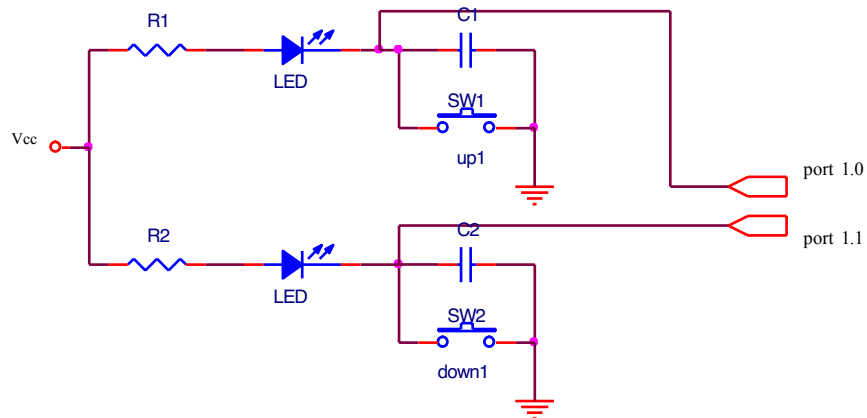
Dalam perancangan, besar arus yang akan digunakan adalah sebesar 10mA. Maka besar nilai resistor yang digunakan adalah :

$$R = \frac{(V_S - V_{LED})}{I}$$

$$R = \frac{(5-1.5)}{10mA}$$

$$R = 350\Omega$$

Maka pada perancangan dipilih nilai resistor R₂ sebesar 330Ω. Banyaknya tombol pemilih mode ini ada dua macam yaitu tombol yang berfungsi untuk memilih jarak servis dan memilih tinggi ban kendaraan. Gambar rangkaian tombol pilihan jarak servis disajikan pada Gambar 3.4.



Gambar 3.4. Rangkaian Tombol Jarak Servis

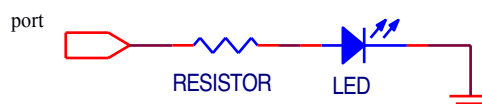
3.2.4 Rangkaian Indikator LED

Rangkaian ini sering sekali digunakan untuk indikator sekaligus informasi awal kepada pemakai. Hal ini memudahkan pemakai untuk mengetahui lebih dini apakah jarak yang telah ditempuh kendaraan bermotor telah melampaui jarak servis atau belum. Digunakan tiga buah LED dengan warna hijau, kuning dan

merah dengan spesifikasi penyalan LED sebagai berikut :

- LED berwarna hijau menyala jika jarak tempuh belum mencapai 0,9 dari jarak servis yang ditentukan.
- LED berwarna kuning menyala jika jarak tempuh telah mencapai 0,9 dari jarak servis hingga mencapai jarak servis yang telah ditentukan.
- LED berwarna merah menyala jika jarak tempuh telah melampaui jarak servis yang telah ditentukan.

Rangkaian LED sederhana dapat dilihat pada Gambar 3.5.



Gambar 3.5 Rangkaian LED

Dengan arus yang diinginkan adalah 10mA, maka untuk menentukan nilai resistor pada rangkaian dapat dihitung dengan persamaan 2.4 :

- LED Berwarna Hijau

$$R = \frac{V_{cc} - V_{LED}}{I_{LED}} = \frac{5 - 2,6}{10mA} = 240\Omega$$

- LED Berwarna Kuning

$$R = \frac{V_{cc} - V_{LED}}{I_{LED}} = \frac{5 - 2,4}{10mA} = 260\Omega$$

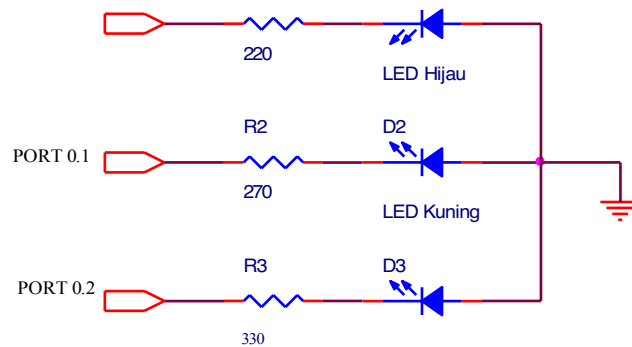
- LED Berwarna Merah

$$R = \frac{V_{cc} - V_{LED}}{I_{LED}} = \frac{5 - 1,95}{10mA} = 305\Omega$$

Dalam perancangan digunakan resistor sebesar 220Ω, 270Ω dan 330Ω.

Gambar rangkaian indikator LED disajikan pada Gambar 3.6.

R1 D1



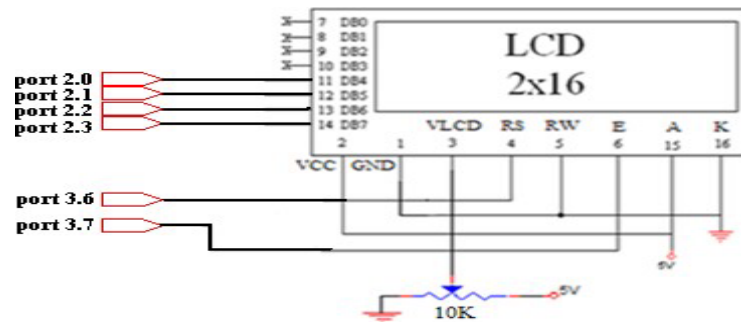
Gambar 3.6 Rangkaian Indikator LED

3.2.5 Rangkaian LCD

LCD yang digunakan adalah LCD 1602 dan menggunakan *i2c converter*. LCD bertipe ini memungkinkan pemrogram untuk mengoperasikan komunikasi data secara 8 bit atau 4 bit. Jika menggunakan jalur data 4 bit akan ada 7 jalur data (3 untuk jalur kontrol dan 4 untuk jalur data). Jika menggunakan jalur data 8 bit maka akan ada 11 jalur data (3 untuk jalur kontrol dan 8 untuk jalur data). Tiga jalur kontrol ke LCD ini adalah EN (*Enable*), RS (*Register Select*) dan R/W (*Read/Write*).

Interface LCD merupakan sebuah *parallel bus*, hal ini sangat memudahkan dan sangat cepat dalam pembacaan dan penulisan data dari atau ke LCD. Kode ASCII yang ditampilkan sepanjang 8 bit dikirim ke LCD secara 4 atau 8 bit pada satu waktu. Jika mode 4 bit yang digunakan, maka 2 *nibble* data dikirim untuk membuat sepenuhnya 8 bit (pertama dikirim 4 bit MSB lalu 4 bit LSB dengan pulsa *clock* EN setiap *nibblenya*). Mengirim data secara paralel baik 4 atau 8 bit merupakan 2 mode operasi primer untuk membuat sebuah aplikasi *interface* LCD. Penentuan mode operasi merupakan hal yang paling penting. Mode 8 bit sangat baik digunakan ketika kecepatan menjadi keutamaan dalam sebuah aplikasi dan setidaknya minimal tersedia 11 pin I/O (3 pin untuk kontrol, 8 pin untuk data). Sedangkan mode 4 bit minimal hanya membutuhkan 7 bit (3 pin untuk kontrol, 4 untuk data). Berdasarkan datasheet tegangan kontras (V_{lcd}) maksimum sebesar 5V dan resistor potensial R_{LCD} ditentukan sebesar 10K hingga 20K.

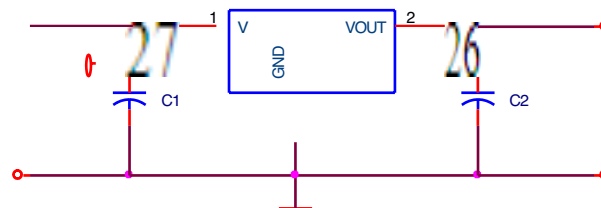
Sehingga dalam perancangan digunakan sebuah resistor variabel sebesar 10K yang berfungsi untuk membatasi tegangan yang masuk ke pin VLcd. Rangkaian LCD dengan mode 4 bit disajikan pada Gambar 3.7.



Gambar 3.7 Rangkaian LCD Mode 4 bit

3.2.6 Rangkaian Regulator Tegangan

Perangkat ini digunakan untuk mencatu seluruh komponen yang ada di alat ini. Perangkat ini harus terus beroperasi sehingga dibutuhkan sumber energi dari aki sepeda motor. Regulator yang dibutuhkan adalah untuk menyesuaikan tegangan accu sepeda motor 12 V dengan tegangan mikrokontroler, sensor dan LCD yang bekerja pada tegangan 5 volt. Rangkaian regulator dapat dilihat pada Gambar 3.8 di balik halaman ini:



Gambar 3.8 Rangkaian Regulator 5 Volt

3.3 Perancangan Perangkat Lunak

3.3.1 Program Utama

Program utama berisi gambaran umum tentang bagaimana cara kerja dari penentu waktu servis kendaraan bermotor ini bekerja. Program dimulai dengan penginisialisasian dari port-port, register dan variabel yang digunakan. Untuk jarak tempuh dan jarak servis dinyatakan dalam satuan kilometer dan tinggi ban

dalam centimeter. Setelah inisialisasi, dilanjutkan dengan pembacaan masukan yang diinterupsi dari tombol pilihan sehingga didapatkan jarak servis dan tinggi ban.

Kemudian mikrokontroler akan menunggu interupsi tombol *start* untuk mulai menjalankan proses selanjutnya. Proses selanjutnya yaitu mikrokontroler akan menghitung keliling dari roda sepeda motor yang digunakan. Dari informasi tinggi ban yang ditentukan oleh pengguna dan dengan diameter velg roda 18 inci (57cm), maka keliling roda dihitung menggunakan persamaan berikut :

$$k = 3,14 \times ((2 \times d) + 57\text{cm})$$

dengan :

d = tinggi ban (cm)

Setelah menghitung keliling roda, kemudian mikrokontroler akan mengirimkan sinyal ke sensor *proximity* supaya sensor mulai bekerja dan menghitung jumlah putaran yang dibaca untuk nantinya diolah menjadi jarak tempuh. Jumlah putaran roda akan bertambah tiap ada putaran dari cakram roda dan dideteksi oleh sensor *proximity*. Tiap putaran cakram akan menghasilkan satu buah sinyal, dan sinyal tersebut akan diakumulasikan dan nantinya akan dikonversikan menjadi jarak tempuh. Pada program, perhitungan jarak yang telah ditempuh dihitung dengan persamaan :

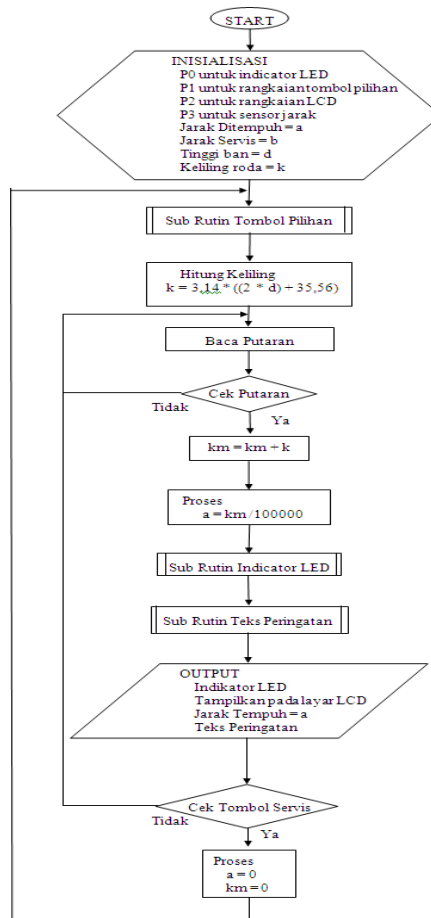
$$km = km + k$$

$a = km / 100000$ dengan :

a = jarak yang telah ditempuh atau jarak tempuh (km) km = jarak tempuh (cm)

k = keliling roda (cm)

Dalam display LCD, jarak tempuh ditampilkan dalam satuan kilometer. Karena keliling roda dalam satuan centimeter, maka harus dibagi 100000 supaya menjadi satuan kilometer. Secara umum diagram alir program utama dari penentu waktu servis kendaraan bermotor dapat dilihat pada Gambar 3.9.



Gambar 3.9 Diagram Alir Program Utama

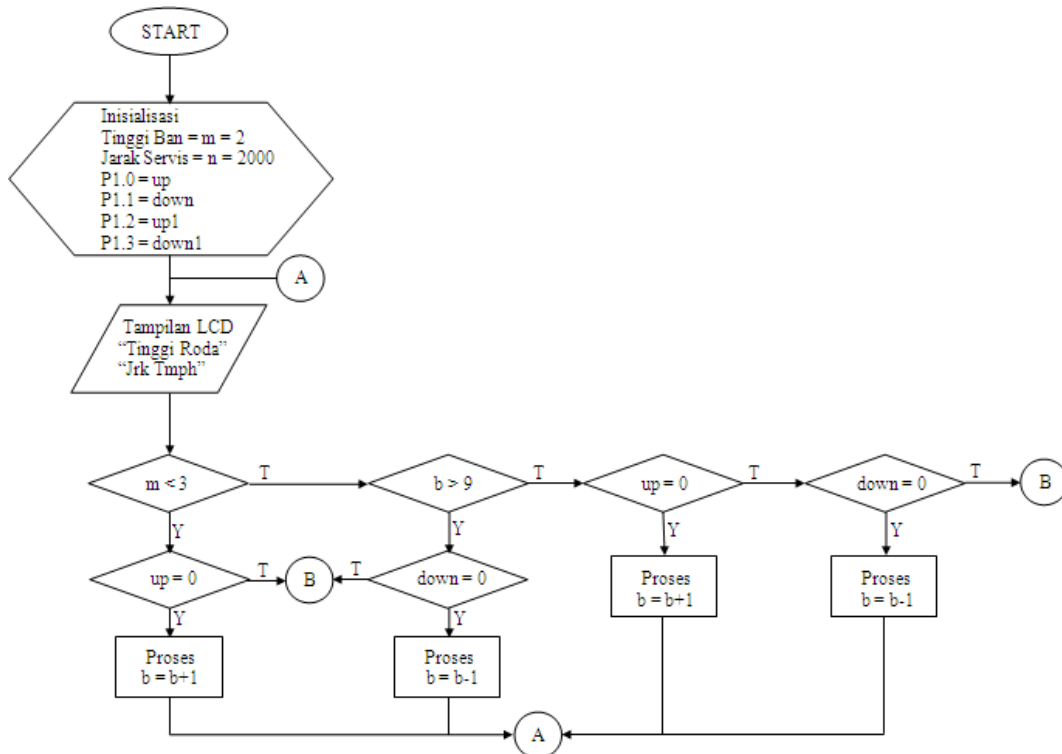
3.3.2 Sub Rutin Tombol Pilihan

Untuk diagram alir subrutin tombol pilihan dapat dilihat pada Gambar 3.10 yang terdiri dari dua pilihan untuk jarak servis dan tinggi ban. Masing-masing tombol pilihan terdiri buah tombol naik (*up*) dan turun (*down*) yang merupakan masukan dari mikrokontroler.

Kondisi awal untuk tombol pilihan jarak servis yaitu bernilai 1500 kilometer dan pada saat tombol di tekan lama yang berwarna hijau maka akan terjadi kelipatan 500 kilometer. Jarak servis maksimum ditentukan sebesar 4000 kilometer. Hal ini dilakukan agar alat ini masih dapat digunakan dalam beberapa tahun ke depan sejalan dengan perkembangan jenis pelumas yang semakin maju dan berkembang. Jarak servis minimum ditentukan sebesar 2000 karena jarak servis minimum yang disarankan untuk saat ini adalah sebesar 2000 kilometer.

Cacahan naik dan turun ditentukan sebesar 500 kilometer. Hal ini dikarenakan jarak servis yang umum digunakan dalam rentang kenaikan antara ratusan kilometer hingga ribuan kilometer.

3.3.3 Subrutin Indikator LED dan Teks Peringatan



Gambar 3.10. Diagram Alir Subrutin Tombol Pilihan

Diagram alir subrutin indikator LED digunakan untuk menentukan warna LED yang menyala pada saat alat ini mulai bekerja. Jumlah LED yang digunakan yaitu tiga buah yaitu hijau, kuning dan merah. Untuk teks peringatan LCD juga terdiri dari tiga macam. Masing masing teks peringatan akan berubah sesuai dengan warna LED yang menyala.

Spesifikasi penyalan ketiga LED dan teks peringatan LCD tersebut yaitu:

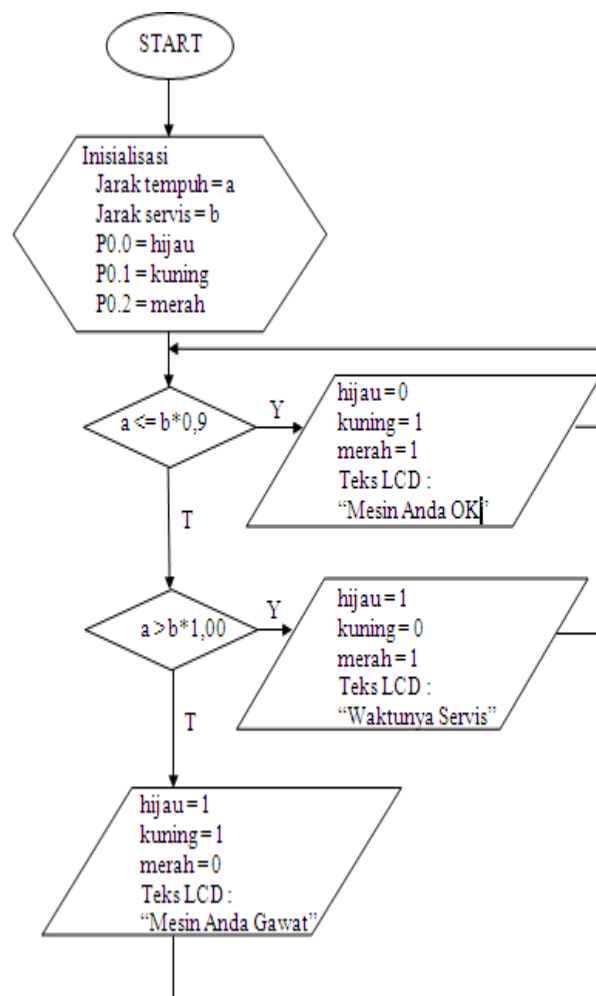
1. Warna hijau menyala, warna kuning dan merah mati

Keadaan ini terjadi jika jarak tempuh bernilai awal nol hingga 90% dari jarak servis. Diambil nilai 90% karena sebagai persiapan untuk mencapai jarak servis sehingga perawatan dapat dilakukan sebelum mencapai nilai jarak servis yang ditentukan. Pada keadaan ini teks peringatan pada layar LCD

bertuliskan “NORMAL”.

2. Warna kuning menyala, warna hijau dan merah mati
Keadaan jika jarak tempuh telah mencapai 90% hingga 100% dari jarak servis. Keadaan ini sebagai jeda yang dianjurkan supaya melakukan perawatan. Pada keadaan ini teks peringatan pada layar LCD bertuliskan “WARNING”.
3. Warna merah menyala, warna hijau dan kuning mati
Keadaan ini jika jarak tempuh telah melebihi 100% jarak servis. Jarak tempuh telah melampaui jarak servis. Pada keadaan ini teks peringatan pada layar LCD bertuliskan “SERVICE”.

Gambar diagram alir subrutin indikator LED dapat dilihat pada Gambar 3.11.

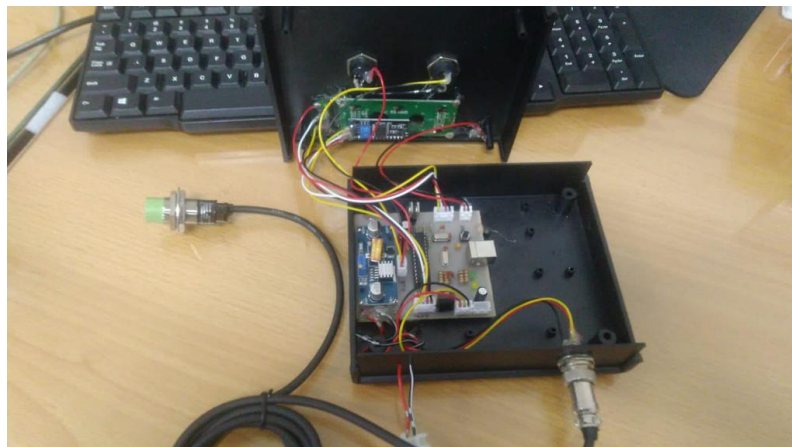


Gambar 3.11. Diagram Alir Subrutin Indikator LED dan Teks Peringatan

BAB IV HASIL DAN PEMBAHASAN

4.1 Implementasi Penentu Waktu Servis Kendaraan Bermotor

Untuk mengetahui suatu alat atau program dapat bekerja dengan baik atau tidak diperlukan pengujian terhadap kinerja alat atau program yang telah dirancang. Pengujian dilakukan terhadap sistem secara keseluruhan dan pada setiap blok. Pengujian berupa pengukuran terhadap perangkat dan respon yang dihasilkan. Pengujian yang dilakukan meliputi pengujian rangkaian sensor jarak, tampilan layar LCD, rangkaian tombol, LED indikator, pengujian daya alat, perbandingan pengukuran alat dengan odometer, dan analisis perangkat lunak. Hasil implementasi alat Penentu Waktu Servis Kendaraan Bermotor yang telah dirancang dapat dilihat pada Gambar 4.1.



Gambar 4.1 Alat Penentu Waktu Servis Kendaraan Bermotor

Mikrokontroler yang digunakan sebagai otak dari perangkat ini adalah ATmega328. Mikrokontroler akan mengatur tampilan-tampilan yang muncul pada layar LCD dan LED indikator berdasarkan masukan yang dikirim melalui penekanan tombol dan sensor. Konstruksi tampilan LCD dan LED indikator dapat dilihat pada Gambar 2.6. Sensor *proximity* diletakkan pada cakram rem depan yang terdapat pada kendaraan bermotor. Konstruksi sensor *proximity* dapat dilihat pada Gambar 2.7. Masukan tombol terdiri dari enam buah tombol yang masing-masing fungsi tombol dijabarkan sebagai berikut :

1. Tombol *Merah* : tombol untuk Riset dengan menahan selama 5 detik dan untuk menyalkan lampu display ditekan 1 kali.
2. Tombol *Hijau* : tombol untuk memilih kilometer dari kelipatan 1500 sampai dengan 4000 kilometer dengan menekan selama 5 detik ditahan



Gambar 4.2 Kontruksi Tampilan LCD Serta Led dan Tombol



Gambar 4.3 Kontruksi Peletakan Sensor *Proxiymity*

Berfungsi sebagai untuk perhitungan jumlah putaran roda pada alat penentuan waktu servis kendaraan bermotor menggunakan mikrokontroler

4.1.1 Pengujian Perangkat Keras

1. Hasil Pengujian Sistem Keseluruhan

2. Pengujian LED Indikator dan Teks Peringatan
3. Pengujian Rangkaian Sensor Jarak
4. Pengujian LCD
5. Pengujian Tombol
6. Pengujian Daya Alat

Menampilkan teks melalui layar LCD, yaitu "Langsung Start" dan "load config" pada baris pertama dan kedua.



Gambar 4.4 Kondisi Alat Saat Dinyalakan

2. Kondisi setelah start

Kondisi ini alat akan menampilkan teks melalui layar LCD yaitu "Odometer =0km" dan "status= Mesin ok" pada garis pertama dan kedua. Kondisi ini alat siap bekerja untuk menghitung jarak tempuh kendaraan.

a. "status=Mesin OK

Teks ini muncul jika jarak tempuh belum mencapai 90% dari jarak servis yang telah ditentukan. Kondisi ini dapat dilihat pada Gambar 4.5



Gambar 4.5 Mesin OK

b. "Status=Warning"

Teks ini muncul jika jarak tempuh antara 75% jarak servis hingga mencapai servis yang telah ditentukan. Kondisi ini dapat di lihat pada Gambar 4.6.



Gambar 4.6 Kondisi "WARNING"

c. "Status = Service"

Teks ini muncul jika jarak tempuh telah mencapai jarak servis yang telah ditentukan. Kondisi ini dapat dilihat pada Gambar 4.7.



Gambar 4.7 Kondisi "SERVICE"

3. Kondisi setelah kendaraan diservis

Jika kendaraan telah diservis maka kondisi alat harus dikembalikan pada kondisi awal dengan cara menekan tombol warna merah tahan selama 5 detik. Setelah ditekan maka tampilan pada LCD akan Kembali seperti semula sama seperti saat dihidupkan pertama kali atau sama dengan kondisi nomor satu

diatas.

4. Kondisi Layar LCD Sesuai Kontak Kendaraan

Kondisi ini yaitu pada saat kita memakai kendaraan atau tidak yang ditandai dengan mengatur kondisi kunci montak kendaraan. Dalam hal ini tampilan LCD akan tampil sesuai dengan kondisi kunci kontak terdapat dua kondisi kunci kontak, yaitu kondisi kunci kontak ON dan OFF. Kondisi layer LCD saat kunci ON ataupun OFF dapat dilihat pada Gambar 4.6 dan Gambar 4.7.



Gambar 4.8 “kondisi ON”



Gambar 4.9 “Kondisi OFF”

4.1.2 Pengujian LED Indikator dan Teks Peringatan

LED indicator dan teks penringatan digunakan sebagai penanda kondisi kendaraan bermotor pada saat tersebut. LED indicator dan teks peringatan mewakili tiga kondisi kendaraan bermotor yang berbeda, yaitu:

- a. Kondisi Aman : LED indikator berwarna hijau menyala. Kondisi terjadi jika jarak tempuh kurang atau sama dengan 75% dari jarak servis. Teks peringatan menampilkan ”Mesin OK”.
- b. Kondisi saatnya Servis : LED indikator berwarna kuning menyala. Kondisi terjadi jika jarak tempuh lebih dari 75% dan kurang dari jarak servis. Teks

peringatan menampilkan “Warning”.

- c. Kondisi Servis : LED indikator berwarna merah menyala. Kondisi terjadi jika jarak tempuh telah melebihi jarak servis. Teks peringatan menampilkan “Service”

Pengujian yang dilakukan ini sesuai dengan perancangan sesungguhnya yang menggunakan jarak tempuh sejauh 2000km. Pengujian ini hanya bertujuan untuk mengetahui serta mengecek kinerja LED indikator dan teks peringatan apakah telah sesuai dengan perancangan yang menggunakan toleransi 100%. Dengan jarak servis sejauh 2000km, maka jarak aman 75% dari jarak servis yaitu sebesar 1500km pengujian dapat dilihat pada tabel 4.1.

Pengujian yang dilakukan sesuai dengan perancangan sesungguhnya yang menggunakan jarak tempuh sejauh 2000 km. Jarak tempuh dipilih sebesar 2000 km dengan alasan bahwa pengujian ini hanya bertujuan untuk mengetahui serta mengecek kinerja LED indikator dan teks peringatan apakah telah sesuai dengan perancangan yang menggunakan toleransi jarak sebesar 100%.

Dengan jarak servis sejauh 2000 km, maka jarak aman 90% dari jarak servis yaitu sebesar 1.800 km. Pengujian dapat dilihat pada Tabel 4.1.

Tabel 4.1 Pengujian LED Indikator dan Teks Peringatan

No	Jarak Tempuh	LED Indikator			Teks Peringatan	Keterangan
		Hijau	Kuning	Merah		
1	0km	On	Off	Off	Mesin OK	
2	500 km	On	Off	Off	Mesin OK	
3	1000km	On	Off	Off	Mesin OK	-
4	1200km	On	Off	Off	Mesin OK	-
5	1300km	On	Off	Off	Mesin OK	-
6	1499km	On	Off	Off	Mesin OK	-
7	1500km	Off	On	Off	Warning	Nyala LED berubah
8	1979km	Off	On	Off	Warning	-
9	1980km	Off	Off	On	Service	Nyala LED berubah
10	2000km	Off	Off	On	Service	

Jarak tempuh yang digunakan merupakan jarak tempuh yang tercatat pada alat. Urutan penyalaan LED indikator yaitu mulai dari hijau, kuning kemudian

merah. Pada keadaan awal (jarak tempuh 0 km) LED berwarna hijau akan menyala (on). Saat jarak tempuh mencapai 1.500 km, LED berwarna hijau akan mati (off) dan LED berwarna kuning menyala (on). Dan pada jarak tempuh mencapai 1.980 km, LED berwarna kuning akan mati (off) dan LED berwarna merah akan menyala (on). Dari data pengujian Tabel 4.1 diketahui bahwa LED indikator menyala sesuai dengan kondisi yang telah dirancang dan teks peringatan mampu menampilkan teks dengan baik pada ketiga kondisi berbeda sesuai dengan perancangan. Maka dapat disimpulkan bahwa LED indikator dan teks peringatan dapat bekerja dengan baik.

4.1.3 Pengujian Rangkaian Sensor Jarak

Rangkaian sensor jarak terdiri dari sebuah *Proximity* yang didalamnya terdapat sebuah led *infra red* sebagai sumber cahaya dan *fototransistor* sebagai pendeteksi pantulan cahaya. Output sensor akan dilewatkan sebuah komparator *inverting* yang kemudian akan menjadi masukan bagi mikrokontroler. Rangkaian komparator menggunakan op-amp LM324 dengan tegangan referensi sebesar 2 Volt. Output sensor akan berubah setiap kali ada perubahan warna pada permukaan cakram rem depan kendaraan bermotor. Hasil pengujian yang dilakukan pada rangkaian sensor dan komparator dapat dilihat pada Tabel 4.2.

Tabel 4.2 Hasil Pengujian Rangkaian Sensor dan Komparator

No	Warna Permukaan Pemantul Cahaya	Indikator LED	Output Sensor	Output Komparator
1	Hitam	ON	4,84V	3,775 V
2	Putih	OFF	239 mV	0,751 V

Dari hasil pengujian tabel 4.2, dapat disimpulkan bahwa sensor dapat bekerja dengan baik karena keluaran 3,775 volt sebagai logika tinggi "1" dan 0,751 volt sebagai logika rendah "0" dapat digunakan pada aplikasi mikrokontroler.

4.1.4 Pengujian Tombol

Tombol digunakan sebagai masukan mikrokontroler tombol hijau berguna untuk mengatur jarak tempuh dari 1.000 sampai 4.000 km dan tombol merah berfungsi untuk meriset pada saat telah di service agar Kembali pada saat awal alat di gunakan.

Pengujian tombol ini dilakukan dengan mengukur besar tegangan keluaran yang dihasilkan pada saat tombol ditekan atau tidak ditekan. Data hasil pengujian tombol disajikan dalam bentuk tabel seperti pada Tabel 4.3.

Tabel 4.3 Pengujian Tombol

No	Tombol ke-	Keadaan Tombol	Tegangan Output
1	1 (Tombol Hijau)	Tidak Ditekan	4,04 V
		Ditekan	2,2mV
2	2 (Tombol Merah)	Tidak Ditekan	4,04 V
		Ditekan	2,3mV

Dari pengujian tombol Tabel 4.3, dapat disimpulkan bahwa tombol dapat bekerja dengan baik. Tombol dapat digunakan dengan baik untuk aplikasi mikrokontroler karena memiliki tegangan logika rendah “0” sebesar 2,2 - 2,4 milivolt dan tegangan logika tinggi “1” sebesar 4,04 volt.

4.1.5 Pengujian Daya Alat

Pengujian ini dilakukan untuk mengetahui seberapa besar daya yang dibutuhkan alat ini pada saat *standby* atau pada saat kendaraan tidak digunakan. Penentu waktu servis kendaraan ini menggunakan catu daya yang berasal dari aki kendaraan. Hal ini berhubungan dengan daya yang mampu disediakan oleh aki untuk mencatu alat penentu waktu servis. Pada saat kendaraan digunakan, daya pada aki kendaraan tidak akan berkurang karena secara bersamaan aki akan *discharge* oleh perputaran mesin kendaraan. Namun pada saat kendaraan tidak digunakan, aki digunakan terus-menerus tanpa adanya pengisian kembali. Lama kelamaan hal ini akan menyebabkan aki mengalami penurunan daya dan dapat menyebabkan aki drop. Pengujian dilakukan yaitu pengujian kemampuan aki untuk mengetahui seberapa lama aki mampu mencatu alat pada keadaan *standby* tanpa mendapatkan pengisian kembali (*charge*).

Aki yang digunakan yaitu aki sepeda motor dengan spesifikasi 12V-5Ah. Artinya aki ini mampu memberikan kuat arus sebesar 5A selama satu jam. Daya yang mampu disediakan aki dapat dihitung dengan persamaan 2.6. Ini berarti aki mampu memberikan daya rata-rata sebesar 60 watt dalam satu jam. Pengujian selanjutnya yaitu pengujian daya pada alat, yaitu daya yang dibutuhkan untuk mensuplai alat. Pengujian dilakukan dengan cara mengukur nilai tegangan dan arus alat pada saat alat digunakan ataupun pada saat *standby*. Hasil pengujian ini disajikan pada Tabel 4.4.

Tabel 4.4. Pengujian Daya

No	Pengujian	Keadaan Alat	
		Menyala	Standby
1	Tegangan	12,45 V	12,45 V
2	Arus	100,8 mA	45,7 mA
3	Daya	1.25496W	0.568965 W

Dari tabel 4.4, besar daya yang dibutuhkan alat pada saat *standby* yaitu sebesar 0.568965 watt. Diketahui bahwa aki mampu mensuplai sebuah alat dengan daya 60 watt per jam, maka dapat diperkirakan berapa lama daya tahan aki mampu mencatu alat ini hingga tegangan minimumnya yaitu sebesar 10,5 volt. Daya tahan aki untuk mencatu alat tanpa adanya pengisian kembali dan bahwa aki masih mampu untuk digunakan kembali dapat dihitung dengan persamaan 2.7. Dari hasil pengujian, maka aki dapat diperkirakan mampu mencatu alat selama 105 jam secara berturut-turut tanpa perlu adanya pengisian kembali dan tanpa merusak aki itu.

4.2 Analisis Hasil Pengukuran

4.2.1 Pengukuran Jarak

Pengujian ini dilakukan untuk mengetahui tingkat akurasi alat yang telah dirancang. Pengujian dilakukan dengan memasang alat pada kendaraan dan kendaraan dikendarai melalui jalan hingga tercapai jarak yang dikehendaki. Hasil jarak yang telah dicatat oleh alat kemudian dibandingkan dengan jarak yang dicatat oleh odometer mekanik yang ada pada kendaraan. Kedua hasil tersebut

kemudian dibandingkan untuk memperoleh tingkat *error* yang terjadi pada alat yang telah dirancang.

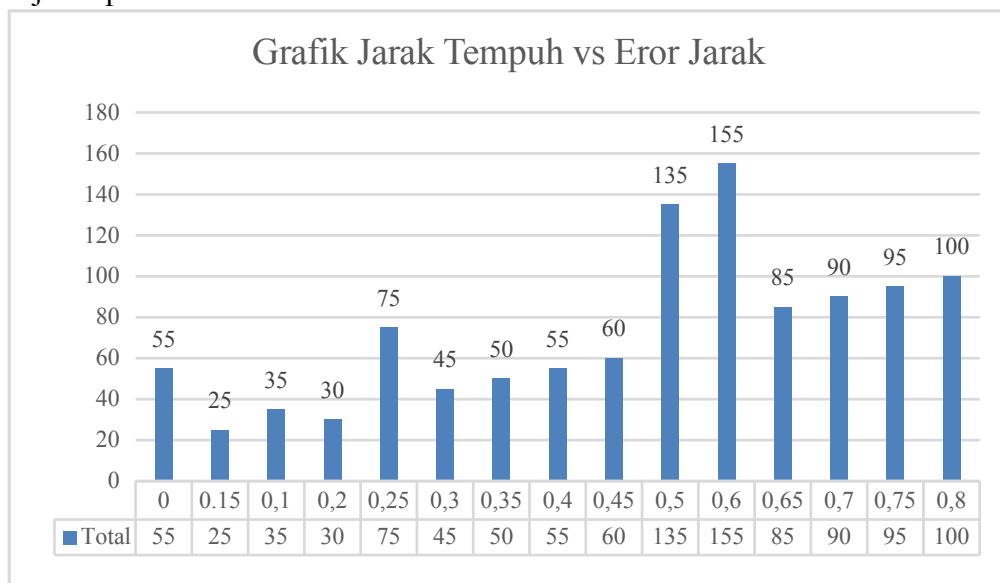
Pengujian yang dilakukan tidak sesuai dengan perancangan pada awalnya. Pada perancangan, pengujian akan dilakukan pada kendaraan Honda New CB150R, menggunakan *velg* dengan diameter 18 inchi. Pengujian yang dilakukan menggunakan *velg* dengan diameter 18 inchi dan tinggi ban sebesar 57 centimeter. Hasil pengujian dan perbandingan dapat dilihat pada Tabel 4.5

Tabel 4.5 Perbandingan Jarak Terukur Pada Alat dan Odometer

No	Jarak Terukur(km)		Error	
	Alat	Odometer	Km	%
1	0	0	0	0
2	1	1	0	0
3	2	2	0	0
4	3	3	0	0
5	4	4	0	0
6	5	5	0	0
7	6	6	0	0
8	7	7	0	0
9	8	8	0	0
10	9	9	0	0
11	10	10	0	0
12	15	15,1	0,1	0,67
13	20	20,1	0,1	0,67
14	25	25,15	0,15	0,6
15	30	30,2	0,2	0,67
16	35	35,25	0,25	0,71
17	40	40,25	0,25	0,625
18	45	45,3	0,3	0,67
19	50	50,35	0,35	0,7
20	55	55,4	0,4	0,72
21	60	60,45	0,45	0,75
22	65	65,5	0,5	0,75
23	70	70,5	0,5	0,7
24	75	75,6	0,6	0,8
25	80	80,6	0,6	0,75
26	85	85,65	0,65	0,76
27	90	90,7	0,7	0,78
28	95	95,75	0,75	0,79
29	100	100,8	0,8	0,8

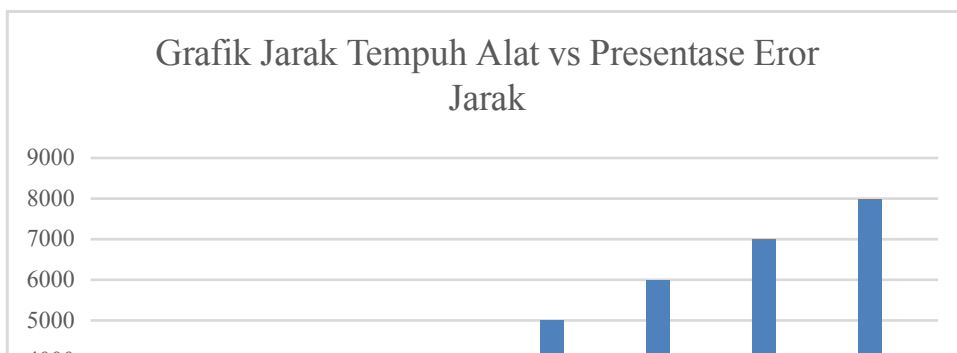
Dari tabel pengujian Tabel 4.5, diperoleh jarak yang terukur oleh alat dan odometer mekanik. *Error* yang diperoleh bisa dikatakan kecil karena *error* yang timbul yaitu antara 0,6% hingga 0,8%.

Dari data pengujian Tabel 4.5 dapat digambarkan grafik hasil perbandingan jarak tempuh dan *error* yang terjadi serta grafik antara jarak tempuh dan presentase error jaraknya. Grafik perbandingan jarak tempuh dan error dapat dilihat pada Gambar 4.10. Grafik perbandingan jarak tempuh dan presentase error disajikan pada Gambar 4.11.



Jarak Tempuh(Km)

Gambar 4.10 Grafik Perbandingan Jarak Tempuh dan Error Jarak



Gambar 4.11. Grafik Perbandingan Jarak Tempuh dan Presentase Error Jarak

Dari tabel pengujian dan grafik perbandingan yang dilakukan dengan jarak tempuh 100 kilometer, dapat diketahui bahwa *error* yang terjadi cenderung berupa garis lurus atau linier. Karena grafik error di atas adalah linier, maka jarak 100 kilometer ini dapat dipergunakan untuk mewakili jarak tempuh sebenarnya sejauh 10000 kilometer. Berdasarkan grafik perbandingan jarak tempuh dan presentase error jarak, dapat dilihat bahwa presentase error berubah-ubah antara 0,6% hingga 0,8% dari jarak tempuh. Perubahan presentase akan tetap stabil antara 0,6% hingga 0,8% dan akan cenderung mengecil menjadi antara 0,7% hingga 0,8% pada saat jarak tempuh telah melampaui 50 kilometer.

Dari data yang diperoleh, maka dengan metode hampiran kuadrat terkecil dapat diperoleh persamaan garis lurus nya. Dengan mengambil beberapa sampel data yang ada, persamaan garis linier untuk *error* pengukuran jarak tempuh dapat dihitung dengan persamaan 2.10. Persamaan garis lurus untuk Grafik 4.13. yaitu:

$$y = -0,04 + 0,00814x$$

Dari persamaan garis linear yang diperoleh, maka besar *error* untuk jarak tempuh yang sebenarnya yaitu antara 2000 hingga 10000 kilometer dapat diperoleh. Besar *error* untuk jarak 2000 hingga 10000 kilometer dapat dilihat pada Tabel 4.6.

Tabel 4.6. Tabel Error Jarak

No	Jarak Tempuh (Km)	Error Jarak (y=-0,04+0,00814x)	Persentase Error (%)
1	1000	8,1 Km	0,81

2	2000	16,24 Km	0,812
3	3000	24,38 Km	0,8126
4	4000	32,52 Km	0,813
5	5000	40,66 Km	0,8132
6	6000	48,8 Km	0,8133
7	7000	56,94 Km	0,8134
8	8000	65,08 Km	0,8135
9	9000	73,22 Km	0,8135
10	10000	81,36 Km	0,8136

Jarak tempuh yang terukur oleh alat lebih lambat dari jarak yang terukur oleh odometer. Namun karena perbedaan jarak yang terukur jauh lebih kecil dari jarak peringatan yang telah ditentukan, yaitu 10% dari jarak servis maka hal ini dapat diabaikan.

Dari analisis data dan pengujian maka dapat disimpulkan bahwa alat dapat bekerja dengan baik karena dapat mengukur jarak tempuh dengan baik dan hanya menghasilkan *error* yang kecil.

4.2.2 Pengujian Perangkat Lunak

Perangkat lunak yang digunakan untuk membuat program mikrokontroler yaitu Bascom-8051. Perangkat lunak (program) tersebut terdiri dari program utama dan beberapa program subrutin.

```
#include <Wire.h>
#include "SparkFun_External_EEPROM.h"
ExternalEEPROM mem;
#define led_ind 13
#define led_merah 10
#define led_hijau 8
#define led_kuning 9
#define led_putih 5
#define pin_reset A1
#define pin_set A0
#define sensor_pin 3
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd (0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE );
```

```
int ledState = LOW;
int buttonSet, buttonReset;
unsigned long previousMillis = 0;
```

```

const long interval = 1000;
String totalOdometer;
int maxKm, totalMaxKm,jM;
byte memOdo = 1;
byte memMaxKm = 10;
byte memJarakM = 14;
int counterSet = 0;
int counterReset = 0;
int persentase;
String statusServis = "";
bool statusSensor = false;
float totalMeter = 0;
float jarakCm, jarakM;
long jarakKm;
float diameterRoda = 57;//<<kalibrasi diameter roda satuan cm
int countDetik = 0;
int selectedScreen = 0;
void setup() {

  Serial.begin(115200);
  pinMode(led_ind, OUTPUT);
  pinMode(led_kuning, OUTPUT);
  pinMode(led_hijau, OUTPUT);
  pinMode(led_merah, OUTPUT);
  pinMode(pin_set, INPUT_PULLUP);
  pinMode(pin_reset, INPUT_PULLUP);
  pinMode(sensor_pin, INPUT);
  lcd.begin(16, 2);
  lcd.clear();
  lcd.setCursor(0,0);//KOLOM,BARIS
  lcd.print("LANGSUNG START!");
  delay(500);
  lcd.setCursor(0,1);
  lcd.print(" load config. ");
  Serial.println("bismillah");
  jarakCm = 22/7*diameterRoda;
  Wire.begin();
  if (mem.begin() == false)
  { Serial.println("No memory detected. Freezing.");
    while (1) ; }
  Serial.println("Memory detected!");
  int maxBytes = 300; //Ideally we write this many bytes
  if (maxBytes > mem.getMemorySizeBytes())
    maxBytes = mem.getMemorySizeBytes();
  for (int x = 0 ; x < maxBytes ; x++)
    mem.write(x, (uint8_t)('0' + x));
}

```

```

Serial.print("Detected number of address bytes: ");
Serial.println(mem.detectAddressBytes());
Serial.print("Detected pageSizeBytes: ");
Serial.println(mem.detectPageSizeBytes());
mem.getString(memOdo, totalOdometer);
//unsigned long nextEEPROMLocation = mem.putString(memOdo,
totalOdometer);
jarakKm = totalOdometer.toInt();
mem.get(memMaxKm, maxKm);
mem.get(memJarakM, jM);
if(jM>0){ jarakM = jM*100;}
else{ jarakM = 0; }
Serial.print("jarak Meter tersimpan:");
Serial.println(jarakM);
totalMaxKm = maxKm*100;
Serial.print("Max KM Tersimpan :");
Serial.println(totalMaxKm);
//Serial.print("Next available EEPROM location: ");
//Serial.println(nextEEPROMLocation);
digitalWrite(led_hijau, LOW);
digitalWrite(led_kuning, LOW);
digitalWrite(led_merah, LOW);
animasiLed(10);
lcd.clear();
//delay(4000);}

void loop() {if(Serial.available(>0){
String setNilaiOdo = Serial.readStringUntil('\n');
setNilaiOdo.trim();
jarakKm = setNilaiOdo.toInt();
mem.putString(memOdo, setNilaiOdo);
lcd.clear();}

buttonSet = digitalRead(pin_set);
buttonReset = digitalRead(pin_reset);
int readSensor = digitalRead(sensor_pin);

if(readSensor==LOW && statusSensor==false){ jarakM+=jarakCm/100;
int mod = int(jarakM) % 100;
if(mod == 0){int nilaiM = jarakM/100;
mem.put(memJarakM,nilaiM);}

if(jarakM>1000){jarakKm++;
jarakM=jarakM-1000;
totalOdometer = String(jarakKm);

```

```

    mem.put(memJarakM,0);
    mem.putString(memOdo,totalOdometer); }
statusSensor = true;}

if(readSensor==HIGH && statusSensor==true){statusSensor = false; }

unsigned long currentMillis = millis();
if (currentMillis - previousMillis >= interval) { countDetik++;
previousMillis = currentMillis;
if (ledState == LOW) {ledState = HIGH;}
else {ledState = LOW; }
digitalWrite(led_ind, ledState);
if(buttonSet == LOW){ counterSet++; }
else {counterSet = 0;}

if(buttonReset == LOW) {digitalWrite(led_putih, HIGH);
counterReset++;}
else {digitalWrite(led_putih,LOW);
counterReset = 0;}

if(counterSet>=3){ totalMaxKm+=500;
if(totalMaxKm>4000){totalMaxKm = 1500;}
if(totalMaxKm<0){totalMaxKm = 1500}

maxKm = totalMaxKm/100;
mem.put(memMaxKm, maxKm);

lcd.clear();
lcd.setCursor(0,0);
lcd.print("SET MAX ODOMETER");
lcd.setCursor(0,1);
lcd.print("MaxOdo: ");
lcd.print(totalMaxKm);
lcd.setCursor(14,1);
lcd.print("KM");
delay(2000);
if(counterReset>=5){
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" RESET ODOMETER ");
lcd.noBacklight();
delay(500);
lcd.backlight();
delay(500);
lcd.noBacklight();
delay(500);
}
}

```

```

    lcd.backlight();
    delay(500);
    lcd.noBacklight();
    delay(500);
    lcd.backlight();
    totalOdometer = "0";
    mem.putString(memOdo, totalOdometer);
    mem.put(memJarakM, 0);
    jarakM = 0;
    animasiLed(5);
    delay(1000);}

mem.getString(memOdo, totalOdometer);
int countTotal = totalOdometer.toInt();
persentase = map(countTotal,0, totalMaxKm,0,100);

if(persentase>=99){statusServis = " SERVICE";
    digitalWrite(led_merah, HIGH);
    digitalWrite(led_kuning, LOW);
    digitalWrite(led_hijau, LOW);}
else if(persentase>=75){statusServis = " WARNING";
    digitalWrite(led_kuning, HIGH);
    digitalWrite(led_hijau, LOW);
    digitalWrite(led_merah, LOW);}
else{ statusServis = " NORMAL";
    digitalWrite(led_merah, LOW);
    digitalWrite(led_kuning, LOW);
    digitalWrite(led_hijau, HIGH) }
lcd.clear();
screenLCD(selectedScreen)}

if(buttonSet==LOW){ lcd.backlight();
    countDetik=0;
    selectedScreen ++;
    if(selectedScreen>1){selectedScreen=0; }
    lcd.clear();
    screenLCD(selectedScreen);
    delay(200);}

if(buttonReset==LOW){lcd.backlight();
    countDetik=0;}

if(countDetik>=60){ selectedScreen = 0;
    lcd.noBacklight();
    countDetik=0;}

```

```

Serial.print("total Odometer:");
Serial.print(totalOdometer);
Serial.print(" | totalMaxKm:");
Serial.print(totalMaxKm);
Serial.print(" | persentase:");
Serial.print(persentase);
Serial.print(" | buttonReset:");
Serial.print(buttonReset);
Serial.print(" | buttonSet:");
Serial.print(buttonSet);
Serial.print(" | maxKm:");
Serial.print(maxKm);
Serial.print(" | jarak Meter:");
Serial.print(jarakM);
Serial.print(" | jarak KiloMeter:");
Serial.print(jarakKm);
Serial.print(" | detik:");
Serial.print(countDetik);
Serial.println();}

```

```

void animasiLed(int totalLoop){ int statusLed = LOW;
for(int i=0;i<=totalLoop; i++){if(statusLed==LOW){statusLed = HIGH; }
else{statusLed = LOW;}
digitalWrite(led_hijau, statusLed);
delay(50);
digitalWrite(led_kuning, statusLed);
delay(50);
digitalWrite(led_merah, statusLed);
delay(50);
digitalWrite(led_putih, statusLed);
delay(50);}}

```

```

void screenLCD(int selected){if(selected==0){ lcd.setCursor(0,0);
lcd.print("Odometer: ");
if(jarakKm<10000){lcd.setCursor(9,0); }
if(jarakKm<10000){lcd.setCursor(10,0); }
if(jarakKm<1000){ lcd.setCursor(11,0); }
if(jarakKm<100){lcd.setCursor(12,0); }
if(jarakKm<10){ lcd.setCursor(13,0); }
lcd.print(totalOdometer);
lcd.setCursor(14,0);
lcd.print("KM");
lcd.setCursor(0,1);
lcd.print("Status: ");
lcd.print(statusServis); }
if(selected==1){ lcd.clear();

```



```
lcd.setCursor(0,0);  
lcd.print("Meter : ");  
if(jarakM<1000){lcd.setCursor(9,0); }  
if(jarakM<100){ lcd.setCursor(10,0);}  
if(jarakM<10){lcd.setCursor(11,0); }
```

```
lcd.print(jarakM);  
lcd.setCursor(15,0);  
lcd.print("M");  
lcd.setCursor(0,1);  
lcd.print("MaxOdo: ");  
lcd.print(totalMaxKm);  
lcd.setCursor(14,1);  
lcd.print("KM");}}
```

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil perancangan dan pengamatan pada penentu waktu servis kendaraan dapat diambil kesimpulan sebagai berikut:

1. Sistem penentu waktu servis kendaraan telah bekerja dengan baik, yaitu sistem dapat memberikan peringatan servis pada saat sebelum, pada saat ataupun setelah melewati jarak servis dengan baik sesuai dengan pengaturan jarak servis.
2. Sistem memiliki akurasi pengukuran jarak yang baik.
3. Alat mampu bertahan hidup dalam keadaan *standby* selama empat hari tanpa harus adanya pengisian aki kembali.

5.2 Saran

Sehubungan dengan ketidaksempurnaan alat ini, maka untuk pengembangan lebih lanjut ada beberapa saran agar alat ini jauh lebih baik, yaitu sebagai berikut:

1. Karena pengukuran jarak tempuh pada alat ini menggunakan satuan per kilometer, maka untuk pengukuran lebih presisi lagi jarak tempuh dapat ditambah dengan satuan per seratus meter.
2. Konstruksi casing alat dapat diperbagus supaya dapat digabung dengan *dash board* kendaraan sehingga lebih menarik.

DAFTAR PUSTAKA

- A.Aprillisianti, G.D.Yara, dan H.Setiawan, Sistem Keamanan Sepeda Motor Menggunakan Password Berbasis Mikrokontroler ATmega238, Jurusan Matematika dan Jurusan Sistem Informasi, Universitas Sriwijaya. 2012.
- A.Pangestu, (2014) Sumardi, dan Sudjadi, *Perancangan Alat Pengaman Dan Tracking Kendaraan Sepeda Motor Dengan Menggunakan Mikrokontroler ATmega644PA*, Jurusan Teknik Elektro, Universitas Diponegoro.
- Admin, (2015), *Tips Mengetahui Kode, Memilih Dan Memelihara Ban Motor*, <http://ishak.unpad.ac.id/?p=1252>, diakses tanggal 30 November 2016.
- Admin,(2008), *Cara Mengganti Oli Yang Benar*, <http://roda-dua.com/pages/news/?act=show&id=9>,diakses tanggal 29 Oktober 2009
- Aris, (2017), *Buku Pintar Sepeda Motor*, Penerbit Media Pressindo,Jakarta.
- Cullen, Charles G.(2015), *Aljabar Linear Dengan Penerapannya*, PT. Gramedia Pustaka Utama, Jakarta.
- Danu, Marantika. (2017). *Rancang bangun sistem kunci kombinasi berbasis mikrokontroler pada kendaraan roda dua*. Universitas Andalas. Program Studi Sistem Komputer.
- E.M.Yanie, (2011) *Pembuatan Alat Pengamanan Sepeda Motor Menggunakan Password Berbasis Mikrokontroler AVR ATmega8535*, Jurusan Teknik Instrumentasi dan Elektronika, Universitas Diponegoro.
- E.M.Yanie, *Pembuatan Alat Pengamanan Sepeda Motor Menggunakan Password Berbasis Mikrokontroler AVR ATmega328*, Jurusan Teknik Instrumentasi dan Elektronika, Universitas Diponegoro. 2011.
- F.R.Doni, dan T.Widianto, (2015) *Rancangan Pengaman Sepeda Motor Berbasis Mikrokontroler ATmega16 Dengan Kontrol Arduino*, Jurusan Teknik Informatika, Universitas AMIK BSI Purwokerto. Carmudi, 2014, *Periode Ganti Oli*

- Gusti Rama Kumbo, Ario (2012) *Sepeda Motor dengan Menggunakan Mikrokontroller Arduino Nano*. Jurnal, Teknik Elektro. Pontianak : Universitas Tanjung Pura.
- J.R.Oroh, E.Kendekallo, S.R.U.ASompie, J.O.Wuwung, Rancang Bangun Sistem Keamanan Motor Dengan Pengenalan Sidik Jari, Jurusan Teknik Elektro, Universitas Sam Ratulangi. 2014.
- Jaenal,(2009), <http://jaenal91.wordpress.com/2009/04/03/optocoupler/>, diakses tanggal 6 "Optocoupler", November 2009
- K.D.Artika, Rancang Bangun Sistem Pengaman Pada Sepeda Motor Dengan Memanfaatkan Sensor Encoder Dan Sensor Ping, Jurusan Teknik Mesin Otomotif, Politeknik Tanah Laut. 2013.
- Muhammad Fitrah Zambak (2020), *Mikrokontroler*. Jurusan Teknik Elektro, Universitas Muhammadiyah Sumatera Utara.
- Muhammad Fitrah Zambak (2022), *monitoring pemakaian listrik berbasis Mikrokontroler*. Jurusan Teknik Elektro, Universitas Muhammadiyah Sumatera Utara.
- Putro, Eko Agfianto, (2012), *Belajar Mikrokontroler AT89C51/52/55 Teori & Aplikasi*, Penerbit Gava Media.
- R.H.Suki, N.Adah, dan A.Zainuri, Implementasi RFID Sebagai Pengaman Pada Sepeda Motor Untuk Mengurangi Tindak Pencurian, Jurusan Teknik Elektro, Universitas Brawijaya. 2014
- R.R.Rachmat, dan E.S.Julian, (2016) *Pengaman Sepeda Motor Berbasis Mikrokontroler*, Jurusan Teknik Elektro, Universitas Trisakti.
- R.Saleh, dkk. Sistem Keamanan Motor Menggunakan Telepon Selular Berbasiskan Komunikasi Dua Arah, Jurusan Sistem Komputer, Universitas Bina Nusantara. 2007.
- Rimbawati (2015), *Pemanfaatan Transisto*. Jurusan Teknik Elektro, Universitas Muhammadiyah Sumatera Utara.

Shato, (2008) *Fototransistor*, <http://shatomedia.com/2008/12/fototransistor/>, diakses tanggal 6 November 2009

Suratman, Drs. M., (2002), *Servis Dan Teknik Reparasi Sepeda Motor*, Pustaka Grafika,Bandung.

TonyVirawan,(2006), *Suku Cadang, Pelumas dan BBM*, diakses tanggal 8 Oktober 2009.

Vol.6,No.2 (2020). Jurnal Hasil Penelitian dan Karya Ilmiah dalam Bidang Teknik Industri. 100. *Sistem Antrian dan Penjadwalan Mekanik di Bengkel Sepeda Motor*.

LAMPIRAN LISTING PROGRAM

```
#include <W
#include "SparkFun_External_EEPROM.h"
ExternalEEPROM mem;
#define led_ind 13
#define led_merah 10
#define led_hijau 8
#define led_kuning 9
#define led_putih 5
#define pin_reset A1
#define pin_set A0
#define sensor_pin 3
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE );

int ledState = LOW;
int buttonSet, buttonReset;
unsigned long previousMillis = 0;
const long interval = 1000;
String totalOdometer;
int maxKm, totalMaxKm,jM;
byte memOdo = 1;
byte memMaxKm = 10;
byte memJarakM = 14;
int counterSet = 0;
int counterReset = 0;
int persentase;
String statusServis = "";
bool statusSensor = false;
float totalMeter = 0;
float jarakCm, jarakM;
long jarakKm;
float diameterRoda = 57;//<<kalibrasi diameter roda satuan cm
int countDetik = 0;
int selectedScreen = 0;
void setup() {

  Serial.begin(115200);
  pinMode(led_ind, OUTPUT);
  pinMode(led_kuning, OUTPUT);
  pinMode(led_hijau, OUTPUT);
  pinMode(led_merah, OUTPUT);
  pinMode(pin_set, INPUT_PULLUP);
```

```

pinMode(pin_reset, INPUT_PULLUP);
pinMode(sensor_pin, INPUT);
lcd.begin(16, 2);
lcd.clear();
lcd.setCursor(0,0);//KOLOM,BARIS
lcd.print("LANGSUNG START!");
delay(500);
lcd.setCursor(0,1);
lcd.print(" load config. ");
Serial.println("bismillah");
jarakCm = 22/7*diameterRoda;
Wire.begin();
if (mem.begin() == false)
{ Serial.println("No memory detected. Freezing.");
  while (1) ; }
Serial.println("Memory detected!");
int maxBytes = 300; //Ideally we write this many bytes
if (maxBytes > mem.getMemorySizeBytes())
  maxBytes = mem.getMemorySizeBytes();
for (int x = 0 ; x < maxBytes ; x++)
  mem.write(x, (uint8_t>('0' + x));

Serial.print("Detected number of address bytes: ");
Serial.println(mem.detectAddressBytes());
Serial.print("Detected pageSizeBytes: ");
Serial.println(mem.detectPageSizeBytes());
mem.getString(memOdo, totalOdometer);
//unsigned long nextEEPROMLocation = mem.putString(memOdo,
totalOdometer);
jarakKm = totalOdometer.toInt();
mem.get(memMaxKm, maxKm);
mem.get(memJarakM, jM);
if(jM>0){ jarakM = jM*100;}
else{ jarakM = 0; }
Serial.print("jarak Meter tersimpan:");
Serial.println(jarakM);
totalMaxKm = maxKm*100;
Serial.print("Max KM Tersimpan :");
Serial.println(totalMaxKm);
//Serial.print("Next available EEPROM location: ");
//Serial.println(nextEEPROMLocation);
digitalWrite(led_hijau, LOW);
digitalWrite(led_kuning, LOW);
digitalWrite(led_merah, LOW);
animasiLed(10);
lcd.clear();

```

```

//delay(4000);}

void loop() {if(Serial.available(>0){
  String setNilaiOdo = Serial.readStringUntil('\n');
  setNilaiOdo.trim();
  jarakKm = setNilaiOdo.toInt();
  mem.putString(memOdo, setNilaiOdo);
  lcd.clear();}

  buttonSet = digitalRead(pin_set);
  buttonReset = digitalRead(pin_reset);
  int readSensor = digitalRead(sensor_pin);

  if(readSensor==LOW && statusSensor==false){ jarakM+=jarakCm/100;
  int mod = int(jarakM) % 100;
  if(mod == 0){int nilaiM = jarakM/100;
  mem.put(memJarakM,nilaiM);}

  if(jarakM>1000){jarakKm++;
  jarakM=jarakM-1000;
  totalOdometer = String(jarakKm);
  mem.put(memJarakM,0);
  mem.putString(memOdo,totalOdometer); }
  statusSensor = true;}

  if(readSensor==HIGH && statusSensor==true){statusSensor = false; }

  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis >= interval) { countDetik++;
  previousMillis = currentMillis;
  if (ledState == LOW) {ledState = HIGH;}
  else {ledState = LOW; }
  digitalWrite(led_ind, ledState);
  if(buttonSet == LOW){ counterSet++; }
  else {counterSet = 0;}

  if(buttonReset == LOW){digitalWrite(led_putih, HIGH);
  counterReset++;}
  else {digitalWrite(led_putih,LOW);
  counterReset = 0;}

  if(counterSet>=3){ totalMaxKm+=500;
  if(totalMaxKm>4000){totalMaxKm = 1500;}
  if(totalMaxKm<0){totalMaxKm = 1500}

  maxKm = totalMaxKm/100;

```



```

mem.put(memMaxKm, maxKm);

lcd.clear();
lcd.setCursor(0,0);
lcd.print("SET MAX ODOMETER");
lcd.setCursor(0,1);
lcd.print("MaxOdo: ");
lcd.print(totalMaxKm);
lcd.setCursor(14,1);
lcd.print("KM");
delay(2000);
if(counterReset>=5){
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" RESET ODOMETER ");
  lcd.noBacklight();
  delay(500);
  lcd.backlight();
  delay(500);
  lcd.noBacklight();
  delay(500);
  lcd.backlight();
  delay(500);
  lcd.noBacklight();
  delay(500);
  lcd.backlight();
  totalOdometer = "0";
  mem.putString(memOdo, totalOdometer);
  mem.put(memJarakM, 0);
  jarakM = 0;
  animasiLed(5);
  delay(1000);}

mem.getString(memOdo, totalOdometer);
int countTotal = totalOdometer.toInt();
persentase = map(countTotal,0, totalMaxKm,0,100);

if(persentase>=99){statusServis = " SERVICE";
  digitalWrite(led_merah, HIGH);
  digitalWrite(led_kuning, LOW);
  digitalWrite(led_hijau, LOW);}
else if(persentase>=75){statusServis = " WARNING";
  digitalWrite(led_kuning, HIGH);
  digitalWrite(led_hijau, LOW);
  digitalWrite(led_merah, LOW);}
else{ statusServis = " NORMAL";

```

```

    digitalWrite(led_merah, LOW);
    digitalWrite(led_kuning, LOW);
    digitalWrite(led_hijau, HIGH) }
    lcd.clear();
    screenLCD(selectedScreen)}

if(buttonSet==LOW){ lcd.backlight();
    countDetik=0;
    selectedScreen ++;
    if(selectedScreen>1){selectedScreen=0; }
    lcd.clear();
    screenLCD(selectedScreen);
    delay(200);}

if(buttonReset==LOW){lcd.backlight();
    countDetik=0;}

if(countDetik>=60){ selectedScreen = 0;
    lcd.noBacklight();
    countDetik=0;}

Serial.print("total Odometer:");
Serial.print(totalOdometer);
Serial.print(" | totalMaxKm:");
Serial.print(totalMaxKm);
Serial.print(" | persentase:");
Serial.print(persentase);
Serial.print(" | buttonReset:");
Serial.print(buttonReset);
Serial.print(" | buttonSet:");
Serial.print(buttonSet);
Serial.print(" | maxKm:");
Serial.print(maxKm);
Serial.print("| jarak Meter:");
Serial.print(jarakM);
Serial.print("| jarak KiloMeter:");
Serial.print(jarakKm);
Serial.print("| detik:");
Serial.print(countDetik);
Serial.println();}

void animasiLed(int totalLoop){ int statusLed = LOW;
    for(int i=0;i<=totalLoop; i++){if(statusLed==LOW){statusLed = HIGH; }
    else {statusLed = LOW;}
    digitalWrite(led_hijau, statusLed);
    delay(50);
}

```

```

digitalWrite(led_kuning, statusLed);
delay(50);
digitalWrite(led_merah, statusLed);
delay(50);
digitalWrite(led_putih, statusLed);
delay(50);}}

void screenLCD(int selected){if(selected==0){ lcd.setCursor(0,0);
  lcd.print("Odometer: ");
  if(jarakKm<10000){lcd.setCursor(9,0); }
  if(jarakKm<10000){lcd.setCursor(10,0); }
  if(jarakKm<1000){ lcd.setCursor(11,0); }
  if(jarakKm<100){lcd.setCursor(12,0); }
  if(jarakKm<10){ lcd.setCursor(13,0); }
  lcd.print(totalOdometer);
  lcd.setCursor(14,0);
  lcd.print("KM");
  lcd.setCursor(0,1);
  lcd.print("Status: ");
  lcd.print(statusServis); }
if(selected==1){ lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Meter : ");
  if(jarakM<1000){lcd.setCursor(9,0); }
  if(jarakM<100){ lcd.setCursor(10,0);}
  if(jarakM<10){lcd.setCursor(11,0); }

  lcd.print(jarakM);
  lcd.setCursor(15,0);
  lcd.print("M");
  lcd.setCursor(0,1);
  lcd.print("MaxOdo: ");
  lcd.print(totalMaxKm);
  lcd.setCursor(14,1);
  lcd.print("KM");}}

```



UMSU
Unggul | Cerdas | Terpercaya
Bila mengunggah surat ini agar disebutkan nomor dan tanggalnya

MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
FAKULTAS TEKNIK

UMSU Terakreditasi A Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 69/SK/BAN-PT/Akred/PT/III/2019
Pusat Administrasi: Jalan Mukhter Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003
<https://fatek.umsu.ac.id> fatek@umsu.ac.id [umsumedan](#) [umsumedan](#) [umsumedan](#) [umsumedan](#)

**PENENTUAN TUGAS AKHIR DAN PENGHUJUKAN
DOSEN PEMBIMBING**

Nomor :119/IL3AU/UMSU-07/F/2023

Dekan Fakultas Teknik Universitas Muhammadiyah Sumatera Utara, berdasarkan rekomendasi Atas Nama Ketua Program Studi Teknik Elektro Pada Tanggal 03 Februari 2023 dengan ini Menetapkan :

Nama : MUHAMMAD FADLI RANGKUTI
Npm : 1907220030
Program Studi : TEKNIK ELEKTRO
Semester : 8 (Delapan)
Judul Tugas Akhir : PENENTUAN WAKTU SERVIS KENDERAAN BERMOTOR
MENGUNAKAN MIKROKONTROLLER .

Pembimbing : MUHAMMAD ADAM ST, MT

Dengan demikian diizinkan untuk menulis tugas akhir dengan ketentuan :

1. Bila judul Tugas Akhir kurang sesuai dapat diganti oleh Dosen Pembimbing setelah mendapat persetujuan dari Program Studi Teknik ELEKTRO
2. Menulis Tugas Akhir dinyatakan batal setelah 1 (satu) Tahun dan tanggal yang telah ditetapkan.

Demikian surat penunjukan dosen Pembimbing dan menetapkan Judul Tugas Akhir ini dibuat untuk dapat dilaksanakan sebagaimana mestinya.

Ditetapkan di Medan pada Tanggal.
Medan, 12 Rajab 1444 H
03 Februari 2023 M



Munawar Alfansury Siregar, ST.,MT
NIDN: 0101017202



DAFTAR RIWAYAT HIDUP



DATA PRIBADI

Nama Lengkap : MUHAMMAD FADLI RANGKUTI
Alamat : JL. SUKA TIRTA NO 32 LK XIII
Npm : 1907220030
Tempat/Tanggal Lahir : Medan, 16 JANUARI 2001
Jenis Kelamin : Laki-laki
Agama : Islam
Status : Belum Menikah
No Telepon/ Whatsapp : 082199535154
Email : mhdfadliir@gmail.com
Tinggi/Berat Badan : 165 cm/70 kg
Kewarganegaraan : Indonesia

ORANG TUA

Nama Ayah : ahmmad fitri rangkuti
Agama : Islam
Nama Ibu : supiani
Agama : Islam
Alamat : JL. Suka tirta no 32 LK XIII

RIWAYAT PENDIDIKAN

2006-2012 : SD NEGERI 067775 MEDAN
2012-2015 : SMP NEGERI 34 MEDAN
2015-2018 : SMA NEGERI 13 MEDAN
2019-2023 : S1 - Teknik Elektro Universitas Muhammadiyah Sumatera Utara (UMSU)



UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
(UMSU)

FAKULTAS TEKNIK-TEKNIK ELEKTRO

BERITA ACARA BIMBINGAN TUGAS AKHIR (SKRIPSI)

Nama : Muhammad Fadli Rangkuti
NPM : 1907220030
Fakultas/Jurusan : Teknik/ Teknik Elektro
Judul Tugas Akhir : "Penentuan Waktu Servis Kendaraan Bermotor Menggunakan Mikrokontroler"

No	Tanggal	Catatan Asistensi	Paraf Pembimbing
1.	4 Des 23	Perbaiki format judul sesuai Standart Penulisan	f
2.	5 Des 23	Perbaiki jumlah Abstrak dan Kata kunci yg digandakan	f
3	6 Des 23	Gambar harus berwarna	f
4.	Des 23	Daftar pustaka harus 5 thin terakhir yg jadi referensi	f
		Acc. bimbing tugas Akhir. Landings dgn Kopradi : ELEKTRO.	f

Mengetahui,
Pembimbing I


Muhammad Adam, S.T., M.T

Penentuan Waktu Servis Kendaraan Bermotor Menggunakan Mikrokotroler

Nama : Muhammad Fadli Rangkuti
 NPM : 1907220030

Dosen Pembimbing : Muhammad Adam, S.T., M.T

No	Hari/Tanggal	Kegiatan	Paraf
1.	Sabtu. 1/4 ²³	Pelajari tata cara penulisan Skripsi	☺
2.	Rabu 13/4 ²²	Apa yg menjadi tujuan	☺
3.	Kem. 27 28/4 ²²	Bab II Perbanyak teori Dasarnya yang mendukung	☺
4.	Selasa 3/5 ²³	Perhatikan tabel dan keterangan harus sesuai	☺
5.	Rabu 17/5 ²³	Pelajari fununarium saya serpro	☺
6.	Kem. 27 26/5 ²³	Dawat Plocat proses Pembuatan alat	☺
7.	Rabu 21/5 ²³	Aec. lanjut untuk Serpro — ijin ke Prodi	☺


 Muhammad Adam, S.T.



UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA (UMSU)

FAKULTAS TEKNIK-TEKNIK ELEKTRO

BERITA ACARA BIMBINGAN TUGAS AKHIR (SKRIPSI)

Nama : Muhammad Fadli Rangkuti
NPM : 1907220030
Fakultas/Jurusan : Teknik/ Teknik Elektro
Judul Tugas Akhir : "Penentuan Waktu Servis Kendaraan Bermotor
Menggunakan Mikrokontroler"

No	Tanggal	Catatan Asistensi	Paraf Pembimbing
1.	15/7/23	Perbaiki format Judulnya	f
2.	3/8/23	Perjelas kata kunci	f
3.	18/8/23	Perhatikan format. Spasi pada kata	f
4.	29/8/23	Perjelas persamaan Gambar	f
5.	5/9/23	Perjelas tabel jenis Wanita Led	f
6.	23/9/23	Mana Diagram Arusnya	f
7.	23/11/23	Acc. Lanjut Semhas. Koordinasi dengan Kaprodi	f

Mengetahui,
Pembimbing I


Muhammad Adam, S.T., M.T