

Odometer Digital untuk Kendaraan dengan Mikrokontroler MCS51

Handry Khoswanto, Thiang

Dosen Fakultas Teknologi Industri, Jurusan Teknik Elektro – Universitas Kristen Petra
e-mail: handry@petra.ac.id, thiang@petra.ac.id

Jaja Wijaya Guntoro

Alumni Fakultas Teknologi Industri, Jurusan Teknik Elektro – Universitas Kristen Petra

Abstrak

Meski kendaraan bermotor bukan merupakan kebutuhan pokok, hampir semua orang menggunakannya. Kendaraan bermotor perlu dirawat agar tidak mudah rusak. Akan tetapi tidak semua orang ingat waktu perawatan kendaraan bermotor. Untuk mencegah kerusakan maka diperlukan suatu sistem yang dapat mengingatkan pengguna kendaraan. Pada umumnya waktu perawatan kendaraan dilihat dari jarak yang ditempuh. Untuk itu digunakan mikrokontroler AT89C52 dan optocoupler. Dengan mengkonversi putaran menjadi jarak, maka waktu perawatan dapat diketahui. Apabila sudah mendekati waktu perawatan, maka akan diberikan suatu peringatan. Sistem ini juga dilengkapi dengan jam dan *memory* yang digunakan untuk menyimpan data-data perawatan yang dilakukan. Odometer ini memiliki dimensi 16,6 cm x 11,2 cm x 5,1 cm dengan berat 273,5 gram dan menggunakan baterai 12V. Odometer ini mampu mengukur jarak hingga 999.999.999,9 km dan kecepatan (0 km/jam sampai 250 km/jam).

Kata kunci : Pengukur Jarak, Perawatan, Kendaraan.

Abstract

Although cars are not so necessity and luxurious, most people around the world use it as their first equipment. It needs to take care of, so there won't be any damages. Unfortunately not all people always aware of this. To avoid this destruction we need a tool to remaind the owner so he/she can be more aware of this. Mostly the age of using the vehicle depends on how far the distance which its' reached. This process could be done by using an AT89C52 microcontroller and optocoupler. By converting its rotation into a value of distance, we can predicted when the car will need to be treat. When it's the time for it, so there will be a warning about that in the display. This system is also equipped with clock and memory that can be used to save the data of the treatment that have been done. This Odometer has a 16,6 cm x 11,2 cm x 5,1 cm in dimension, with 273,5 gram of weight and used 12V battery. This equipment can measure distance until 999.999.999,9 km and speed (0 km/h to 250 km/h).

Keyword : Measuring Distance, Treatment, Vehicle.

1. Pendahuluan

Pada saat ini banyak kendaraan bermotor yang mengalami kerusakan, pada umumnya kerusakan yang terjadi akibat kurangnya perawatan. Bagian kendaraan yang dirawat pada umumnya adalah mesin. Mesin merupakan penggerak utama dari kendaraan bermotor. Mesin membutuhkan pelumas. Fungsi dari pelumas yaitu mengurangi gesekan yang terjadi pada mesin akibat dari putaran mesin. Pelumas memiliki kadar kekentalan. Kadar kekentalan pada pelumas dapat berkurang akibat gesekan

yang terjadi pada mesin. Apabila kadar kekentalan pelumas sudah tidak baik, maka mesin akan cepat panas, hal ini disebabkan karena gesekan yang ditimbulkan semakin besar. Apabila hal ini dibiarkan terus menerus maka mesin bisa pecah, sehingga sering terjadi kebocoran pada mesin. Untuk menghindari terjadinya kerusakan mesin maka kendaraan harus dirawat. Pada umumnya waktu perawatan kendaraan dilihat berdasarkan jarak tempuh dan waktu pemakaian. Dengan kesibukan yang ada, seringkali pengguna kendaraan melupakan jadwal perawatan yang harus dilakukan. Untuk itu diperlukan suatu sistem yang dapat mengingatkan pengguna kendaraan, apabila kendaraan yang dipakai sudah waktunya untuk dirawat.

Catatan : Diskusi untuk makalah ini diterima sebelum tanggal 1 Februari 2004. Diskusi yang layak muat akan diterbitkan pada Jurnal Teknik Mesin Volume 6 Nomor 1 April 2004.

Pembahasan makalah ini akan dimulai dengan batasan masalah, sistem odometer, perencanaan dan desain, kemudian diikuti oleh pengujian sistem dan kesimpulan.

2. Sistem Odometer Mekanis

Dibawah ini adalah gambar odometer mekanis dengan satuan jarak mil. Bila diketahui odometer mekanis memiliki reduksi roda N:1. Itu berarti poros *input* pada odometer ini harus berputar N kali untuk mencatat 1 mil.



Gambar 1. Odometer Mekanik

Odometer mekanis diputar oleh sebuah kabel fleksibel yang terbuat dari sebuah per yang dililit secara ketat. Kabel itu biasanya berputar didalam sebuah tabung logam protektif dengan sebuah blok karet. Pada sebuah sepeda, sebuah roda kecil yang menggulirkan roda sepeda memutar kabel, dan rasio roda gigi terhadap odometer harus diukur ke ukuran roda kecil ini. Pada sebuah mobil, sebuah roda gigi mengaktifkan poros *output* transmisi, memutar kabel.



Gambar 2. Reduksi Worm Gear

Kabel ini melingkar pada panel instrumen, dimana ini dihubungkan pada poros *input* odometer. Odometer ini menggunakan serangkaian dari tiga *worm gear* untuk mencapai reduksi roda giginya. Poros *input* menggerakkan ulir pertama, yang menggerakkan sebuah roda gigi. Setiap putaran penuh roda gigi hanya memutar roda gigi dengan satu gigi. Roda gigi itu memutar ulir lainnya, yang memutar roda gigi lainnya, yang memutar ulir terakhir dan akhirnya roda gigi terakhir, yang dikaitkan pada indikator sepersepuluh dari satu km. Tiap

indikator memiliki sebuah deret pasak yang menonjol keluar dari satu sisi, dan sebuah *range* yang tunggal 2 pasak pada sisi lainnya. Ketika rangkaian dua pasak sampai ke roda gigi plastik putih, salah satu dari roda gigi berada diantara pasak-pasak dan berputar dengan indikator sampai pasak-pasak melewati. Roda gigi ini juga melibatkan salah satu dari pasak-pasak pada indikator berikutnya yang lebih besar memutarnya sepersepuluh dari sebuah putaran.



Gambar 3. Output Worm Gear Menggerakkan Poros dan Memutar Indikator

Apabila kendaraan berjalan mundur, maka odometer mekanis juga akan berputar terbalik. Hal ini terjadi karena odometer ini benar-benar merupakan sebuah rangkaian roda gigi. Untuk memutar kembali odometer mekanik dapat juga dilakukan dengan cara mengaitkan kabel odometer pada sebuah *drill* dan menjalankannya ke arah belakang untuk memutar kembali odometer.

3. Sistem Odometer Digital

Odometer dengan sistem komputer memiliki sebuah magnet yang dilekatkan pada salah satu roda dan sebuah *pickup* yang dilekatkan pada kerangka. Ketika roda berputar, magnet melewati *pickup*, menghasilkan sebuah tegangan dalam *pickup*. Komputer menghitung *spike* tegangan, atau pulsa, dan menggunakannya untuk mengkalkulasikan jarak yang ditempuh.

Pada saat instalasi odometer digital, maka *perimeter* roda harus diprogram terlebih dahulu. *Perimeter* adalah jarak yang ditempuh ketika roda membuat satu putaran penuh. Tiap waktu komputer mendeteksi sebuah pulsa, kemudian pulsa dikalkulasi menjadi jarak dan memperbaharui *display* digital.

Banyak mobil modern menggunakan sebuah sistem seperti ini juga. Bukannya sebuah *pickup* magnetik pada sebuah roda, mereka menggunakan sebuah roda gigi yang dipasang

pada sebuah transmisi dan sebuah *sensor* magnetik yang menghitung jumlah pulsa ketika tiap gigi roda berlalu. Beberapa mobil menggunakan sebuah roda bercelah dan sebuah *pickup* optikal, seperti yang dilakukan oleh sebuah *mouse* komputer. Komputer dalam mobil telah mengetahui berapa banyak jarak yang ditempuh oleh mobil dengan tiap pulsa, dan menggunakan ini untuk memperbaharui pembacaan skala odometer.

Hal yang paling menarik tentang odometer mobil adalah bagaimana informasi diteruskan ke *dashboard*. Bukan lagi sebuah kabel berputar yang meneruskan sinyal jarak, tetapi jarak bersama dengan banyak data lainnya diteruskan melalui *bus* dari unit kontrol mesin ke *dashboard*. Mobil adalah seperti sebuah jaringan area lokal dengan banyak perlengkapan berbeda yang dihubungkan padanya. Berikut ini adalah beberapa perlengkapan yang mungkin dihubungkan pada jaringan komputer dalam sebuah mobil :

- Unit kontrol mesin.
- *Dashboard*.
- Kontrol *power window*.
- *Anti-lock breaking system*.
- Modul pengendalian *air bag*.
- Modul kontrol bodi (menjalankan lampu-lampu dalam, dll).
- Modul kontrol transmisi.

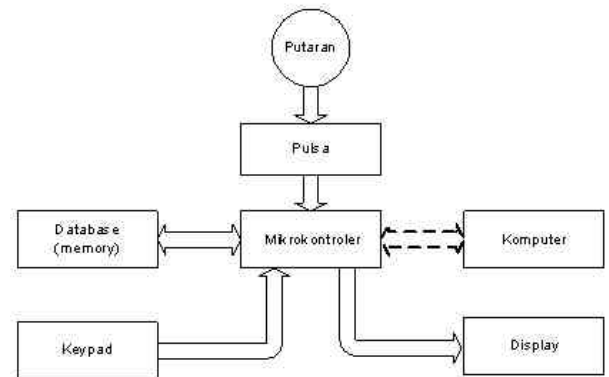
Banyak kendaraan sudah menggunakan sebuah protokol komunikasi yang distandarisasi, yang disebut dengan SAE J1850, memungkinkan seluruh dari modul elektronik berbeda berhubungan satu sama lain.

Unit kontrol mesin menghitung seluruh dari pulsa dan menelusuri seluruh jarak yang ditempuh oleh mobil. Ini berarti bahwa jika seseorang berusaha untuk “memundurkan” odometer, nilai yang disimpan dalam unit kontrol mesin akan tidak bersesuaian. Nilai ini akan dibaca menggunakan sebuah komputer diagnostik, yang dimiliki oleh seluruh departemen servis dealer mobil.

Unit kontrol mesin mengirimkan sebuah paket informasi yang terdiri atas sebuah *header* dan data. *Header* hanyalah sebuah bilangan yang mengidentifikasi paket sebagai sebuah pembacaan skala jarak, dan data merupakan sebuah bilangan yang sama dengan jarak yang ditempuh. *Panel* instrumen mengandung komputer lain yang mengetahui bagaimana mencari paket utama ini, dan bilamana melihatnya, ini memperbaharui odometer dengan nilai baru. Dalam mobil-mobil dengan odometer digital, *dashboard* benar-benar menampilkan nilai baru.

4. Perencanaan dan Desain

Untuk perencanaan hardware secara keseluruhan dapat dilihat pada blok diagram dibawah ini.



Gambar 4. Blok Diagram Sistem

Dari Blok Diagram, input berupa *sensor* yang dapat mendeteksi putaran roda. Apabila terjadi putaran, maka jarak yang ditempuh akan bertambah sesuai dengan perbandingan antara jarak dan putaran. Apabila jarak yang ditempuh sama dengan jarak servis dikurangi jarak peringatan, maka akan diberikan suatu peringatan pada LCD tiap 15 detik. Komputer pada blok diagram tersebut berfungsi untuk *transfer* data-data perawatan. Untuk transfer data antara komputer dengan mikrokontroler digunakan komunikasi serial dengan *baudrate* 19200 bps. Proses perhitungan jarak dan kecepatan serta pemberian peringatan servis dilakukan oleh mikrokontroler AT89C52. Adapun fitur-fitur yang akan diberikan pada odometer ini, yaitu:

- Pengukuran jarak.
- Pengukuran kecepatan.
- Pengukuran kecepatan tertinggi.
- Peringatan servis.
- Tanggal dan jam.
- Pemberian menu pada sistem yang digunakan untuk pengaturan sistem agar lebih mudah (*user friendly*) pada saat tidak terkoneksi dengan komputer.
- Pengaturan, penyimpanan, dan pengambilan data-data melalui komputer

5. Desain Hardware

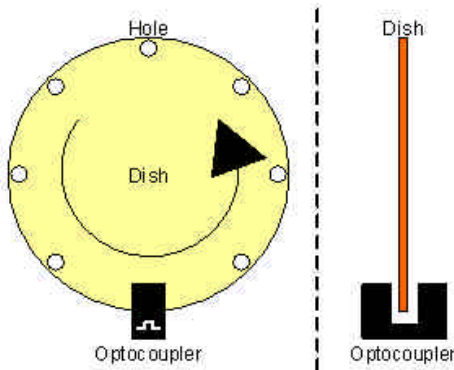
Sensor jarak dibuat dari potongan mekanik odometer sepeda motor Yamaha, sebuah piringan dan sebuah optocoupler. *Sensor* jarak dibuat dengan cara memodifikasi potongan mekanik dan menambahkan sebuah piringan yang dipasang pada poros mekanik odometer

dan sebuah optocoupler yang dipasang pada tepi piringan. Odometer hasil modifikasi dapat dilihat pada gambar berikut.



Gambar 5. Sensor Jarak

Pada gambar dibawah ini, terdapat 8 lubang dan sebuah optocoupler yang digunakan untuk membangkitkan pulsa. Satu kali putaran akan menghasilkan 8 pulsa

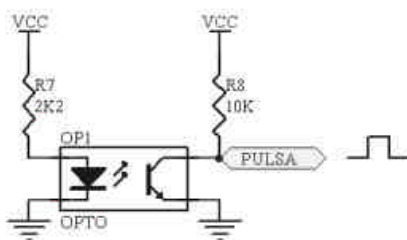


Gambar 6. Piringan Odometer

Dengan memanfaatkan cara kerja dari optocoupler, yaitu bila terhalang *output* akan *open* dan bila optocoupler tidak terhalang *output* akan *short*. Maka putaran dapat dikonversi menjadi pulsa yang dapat ditangkap oleh mikrokontroler.

Spesifikasi dari optocoupler yang digunakan yaitu tegangan forward pada diode sebesar 1,1 volt dengan arus forward sebesar 1,7 mA. Sehingga didapatkan nilai nilai resistor yang tercantum pada gambar rangkaian schematic berikut.

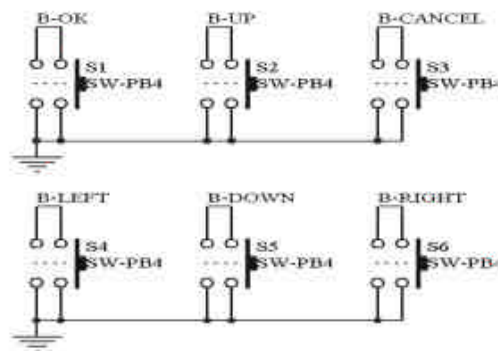
Berikut merupakan gambar rangkaian interface piringan odometer yang nantinya dibaca oleh mikrokontroler dalam bentuk pulsa.



Gambar 7. Schematic Optocoupler

Sistem ini juga dilengkapi dengan keypad dengan maksud agar bisa di *setting* tanpa menggunakan komputer. Karena akan sangat menyulitkan jika pengaturan data hanya bisa dilakukan melalui komputer. *User* harus memasang kabel *serial* pada sistem yang terpasang di mobil dengan komputer untuk dapat melakukan pengaturan pada sistem.

Pengaturan yang ada pada sistem harus dibuat *user friendly* sehingga mudah digunakan tanpa harus membaca buku panduan terlebih dahulu. Untuk itu sistem harus dilengkapi dengan menu. Untuk membuat suatu menu sesuai dengan kriteria tersebut, maka diputuskan untuk menggunakan enam buah tombol yaitu tombol 'ok', 'cancel', 'up', 'down', 'left', dan 'right'.



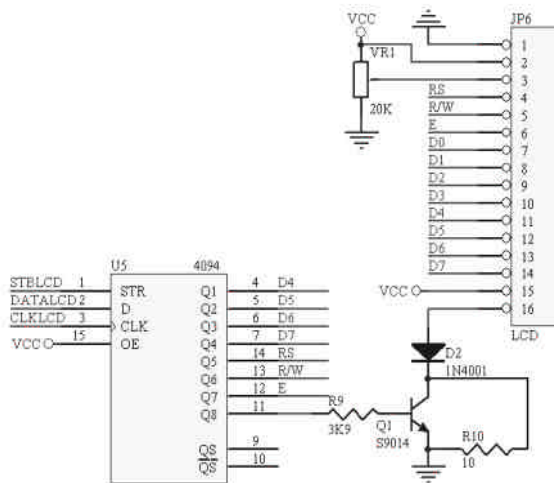
Gambar 8. Schematic Keypad

Pada rangkaian keypad diatas, tombol dihubungkan ke P2.0 sampai P2.5 dan ground. Karena port 2 pada mikrokontroler AT89C52 sudah memiliki *internal pull up*, maka pada port 2 tidak diberi rangkaian *pull up*.

Apabila tombol ditekan maka akan menghasilkan *logic 0* dan apabila tidak ditekan maka akan menghasilkan *logic 1*.

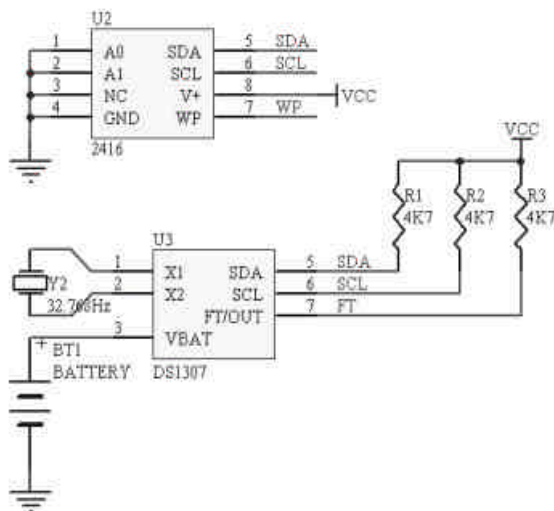
Perencanaan selanjutnya adalah perencanaan display LCD (Liquid Crystal Display) Total jumlah pin yang ada pada LCD tipe M1632 adalah 16 pin. Supaya sambungan ke mikrokontroler menjadi lebih mudah, maka pada LCD diberi IC buffer dari serial ke parallel yaitu IC 4094. Sehingga sambungan dari LCD ke mikrokontroler hanya berjumlah 5 pin, yaitu data, clock, strobe, vcc, dan ground. Buffer pada IC 4094 dapat menampung 8 bit data, sedangkan kontrol LCD untuk mode 8 bit berjumlah 12 bit, supaya IC 4094 yang digunakan hanya 1 buah. Maka pengiriman data pada LCD di-set ke mode 4-bit.

Karena pengiriman data yang dipakai adalah mode 4 bit, maka yang tersambung adalah D4, D5, D6, dan D7. Sisa pin dari IC 4094 dimanfaatkan untuk keperluan kontrol dan LED backlight.



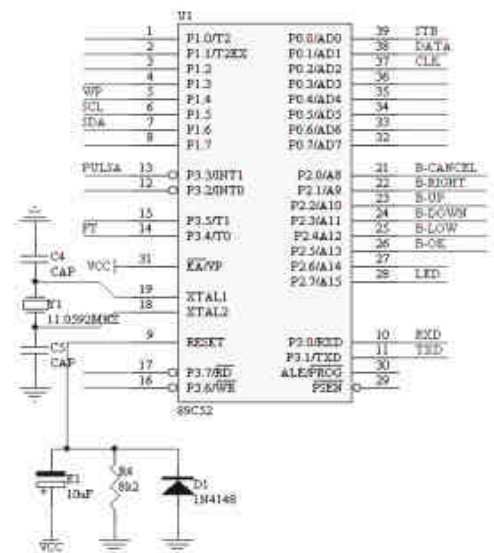
Gambar 9. Schematic Driver LCD

Pada sistem ini *memory* yang digunakan adalah AT24C16, dengan pertimbangan pembuatan alat menjadi lebih mudah dan murah, mengingat jumlah *pin* yang sedikit. AT24C16 memiliki kapasitas penyimpanan data sebesar 2K *byte*. AT24C16 ini menggunakan sistem I2C *bus*, dimana proses pengiriman data hanya dilakukan dengan menggunakan 2 kabel. Untuk jam digunakan DS1307 dengan pertimbangan proses pengiriman data menggunakan sistem I2C *bus*, sama seperti AT24C16 sehingga jalur data, *clock* dapat digabung. Sehingga dapat mempersingkat pembuatan *software* karena sama-sama menggunakan rutin untuk I2C *bus*.



Gambar 10. Schematic AT24C16 dan DS1307

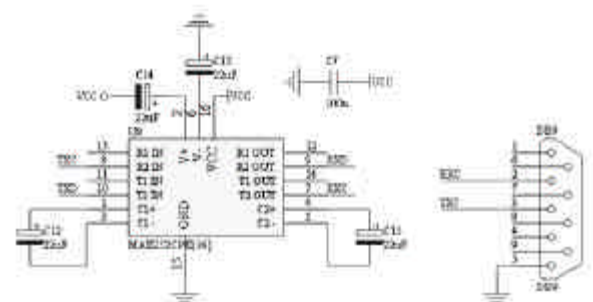
DS1307 memerlukan baterai eksternal yang digunakan untuk menjalankan jam apabila power suplay dimatikan. *Pin* FT/OUT dihubungkan ke T0 pada mikrokontroler, dengan tujuan perubahan waktu yang terjadi pada DS1307 dapat diketahui oleh mikrokontroler.



Gambar 11. Rangkaian Mikrokontroler AT89C52

Pada sistem ini menggunakan mikrokontroler 89C52 dengan spesifikasi *internal program memory* sebesar 8K *byte* dan *internal RAM* sebesar 256 *byte*. Secara umum mikrokontroler ini mampu dijalankan secara mandiri tanpa bergantung pada IC lainnya (*single chip*). Berikut disertakan gambar rangkaian mikrokontroler lengkap beserta dengan rangkaian power on reset dan rangkaian oscillatornya.

Sedangkan untuk mendownload data data yang didapatkan dari sistem odometer ini diperlukan unit interface dari mikrokontroler ke PC atau sebaliknya. PC menggunakan level 232 untuk keperluan serial interfacenya sedangkan mikrokontroler menggunakan level TTL. Unit interfacing dari sistem odometer ini menggunakan IC MAX232 seperti gambar di bawah ini.

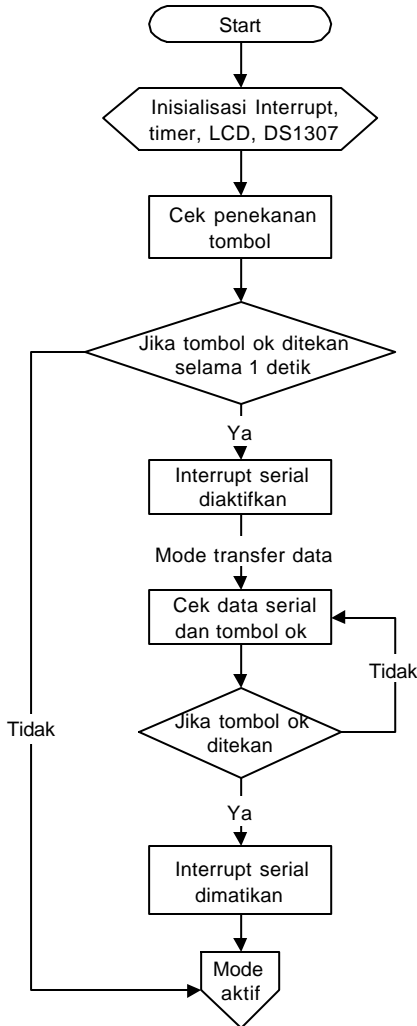


Gambar 12. Interface RS232

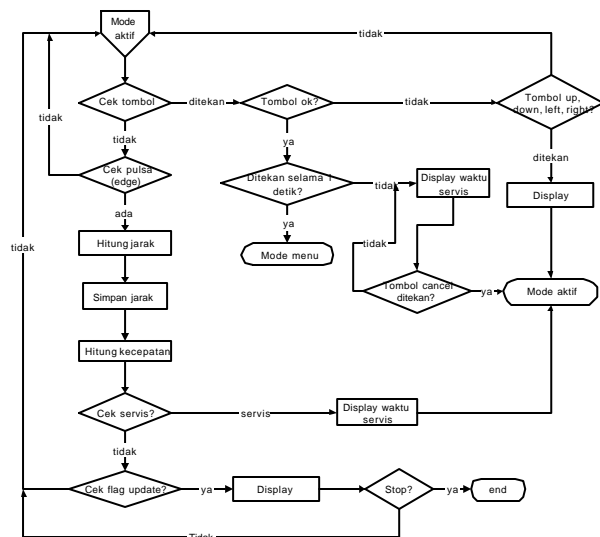
6. Perencanaan Software

Dalam perencanaan software dibagi menjadi dua bagian yaitu *high level language* dan *low level language*. *Low level language* digunakan

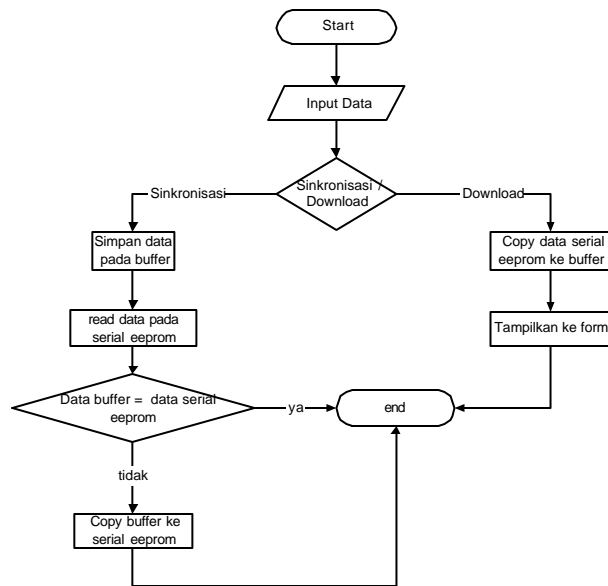
untuk memprogram mikrokontroler sedangkan *high level language* sebagai program download pada PC. Berikut *flow chart* dari kedua software tersebut.



Gambar 13. Flow Chart Sistem pada Waktu Start



Gambar 14. Flow Chart Sistem pada Saat Running



Gambar 15. Flow Chart pada PC



Gambar 16. Tampilan Program Komputer

7. Pengujian Sistem

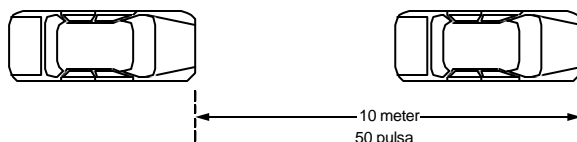
Pengujian sistem meliputi pengujian device penunjang keberhasilan sistem dan pengujian sistem itu sendiri. Pengujian device terdiri atas pengujian sensor putaran, pengujian penyimpanan dan pembacaan data, kalibrasi jarak tempuh. Pengujian sensor putaran dilakukan dengan cara memutar kabel odometer. Pada *sensor* putaran terdapat sebuah piringan dengan lubang ditepinya, percobaan dilakukan dengan cara mengukur tegangan pada optocoupler pada saat mengenai lubang dan pada saat tidak mengenai lubang.

Tabel 1. Pengukuran Tegangan Optocoupler

Tegangan	Terhalang	Tidak terhalang
V _{AK}	1,1 V	1,1 V
V _{CE(output)}	4,96 V	0,43 V

Pada sistem terdapat dua media untuk menyimpan data, yaitu *serial EEPROM AT24C16* dan *serial RTC DS1307*, kedua media ini memiliki karakteristik yang berbeda dalam hal penyimpanan data. *Serial EEPROM 24C16* memiliki keterbatasan penulisan ulang, tetapi dapat menyimpan data pada saat power dimatikan. *Serial RTC DS1307* tidak memiliki keterbatasan dalam penulisan ulang, tetapi *DS1307* membutuhkan baterai untuk menyimpan data dan menjalankan jam. Apabila baterai tidak dipasang, semua data yang ada pada *DS1307* akan hilang. Untuk mengetahui proses penyimpanan data yang dilakukan sudah benar, maka dilakukan pengujian dari data-data yang disimpan. Cara menguji penyimpanan data dilakukan dengan cara mengisi data pada *serial EEPROM* dan *serial real time clock* dengan menggunakan menu yang telah dibuat, dan untuk mengecek kebenarannya dilakukan dengan cara menampilkan data-data tersebut dengan menu yang ada. Selain menggunakan menu yang ada pada sistem, data pengisian juga dapat dicek melalui fasilitas download yang ada pada program komputer.

Pengujian ketiga adalah kalibrasi jarak tempuh sistem. Hal yang terpenting dalam sistem ini adalah pengukuran jarak. Sistem harus dapat mengukur jarak tempuh dengan baik. Ketepatan pengukuran jarak akan membuat perhitungan perkiraan waktu servis makin baik. Pengujian dilakukan dengan menggunakan mobil isuzu panther. Sebelum dilakukukan pengujian perlu dilakukan kalibrasi jarak per pulsa terlebih dahulu. Kalibrasi jarak per pulsa dilakukan dengan cara menghitung jumlah pulsa yang dihasilkan pada saat kendaraan menempuh jarak 10 meter.



Gambar 17. Sketsa Penentuan Kalibrasi Jarak

Hasil Kalibrasi menunjukkan jumlah pulsa yang dihasilkan pada saat menempuh jarak 10 meter sebanyak 50 pulsa. Sehingga jarak per pulsa dapat dihitung dengan dengan rumus berikut ini.

$$X = S / P$$

$$X = 10 \text{ meter} / 50 \text{ pulsa}$$

$$X = 20 \text{ cm} / \text{pulsa}$$

Untuk membuktikan sistem dapat mengukur jarak dengan baik, dilakukan pengujian. Pengujian dilakukan dengan cara membaca hasil pengukuran sistem dan membandingkannya dengan hasil pengukuran odometer mekanis. Kemudian kedua hasil pengukuran tersebut dibandingkan dengan papan jarak yang ada di tepi jalan. Berikut ini adalah hasil pengukuran jarak.

Tabel 2. Perbandingan Pembacaan Jarak

X = 20 CM/PULSA

Jarak (Km)	Pembacaan Odometer Mekanis (Km)	Pembacaan Sistem (Km)
0,5	0,5	0,5
1	1	1
2	2	2
3	3	3,1
4	4,1	4,1
5	5,1	5,2
6	6,2	6,2
7	7,2	7,3
8	8,2	8,3
9	9,3	9,4
10	10,3	10,4

Pengujian sistem akan dilakukan apabila seluruh pengujian setiap device telah peroleh hasil dengan baik. Pengujian pengukuran kecepatan dapat dilakukan dengan cara menghitung jumlah putaran per menit pada mekanik. Untuk mengukur jumlah putaran per menit digunakan *tachometer*.

Pada saat pengujian variable jarak per pulsa pada sistem di-set 20 cm/pulsa. Karena terdapat 8 lubang pada piringan odometer, maka satu putaran akan menghasilkan 8 pulsa.

Untuk data pengukuran kecepatan dengan menggunakan *tachometer*., diambil nilai rata-rata dari 10 kali percobaan, dan untuk mendapatkan nilai dalam satuan KM/JAM digunakan rumus dibawah ini.

$$V = \frac{R \times X \times C \times 60}{100000}$$

R : Putaran per menit.

X : Jarak per pulsa.

C : Jumlah lubang.

V : Kecepatan (KM/JAM)

Tabel 3. Pengukuran Kecepatan

X = 20 CM/PULSA C = 8

Hasil Pengukuran Sistem	Hasil Pengukuran Tachometer	
	(RPM)	(KM/JAM)
10	103,54 Putaran/Menit	9,939
20	207,85 Putaran/Menit	19,95
30	310,69 Putaran/Menit	29,826
40	412,92 Putaran/Menit	39,64
50	509,69 Putaran/Menit	48,93
70	712,81 Putaran/Menit	68,43
90	920,52 Putaran/Menit	88,37
110	1131,77 Putaran/Menit	108,65
130	1338,43 Putaran/Menit	128,49
150	1548,13 Putaran/Menit	148,62
170	1749,27 Putaran/Menit	167,93
190	1954,06 Putaran/Menit	187,59
210	2160,10 Putaran/Menit	207,37
230	2359,27 Putaran/Menit	226,49
250	2566,25 Putaran/Menit	246,36

Bila dibandingkan antara hasil pengukuran sistem dengan hasil pengukuran tachometer, maka hasil pengukuran sistem mendekati hasil pengukuran dengan menggunakan tachometer. Peringatan servis diberikan oleh sistem dengan cara menghitung waktu tempuh dan waktu servis kemudian membandingkannya dengan data remainder. Untuk menguji peringatan servis, dilakukan dengan cara mengatur kilometer servis dan remainder mendekati odometer, kemudian kendaraan dijalankan.

Tabel 4. Peringatan Servis

KM Servis = 000000040,0

Remainder = 0030,0 Km

KM Kendaraan	Estimasi Servis (Km)	Peringatan Servis
00000000,0	40,0	Tidak Ada
00000000,5	39,5	Tidak Ada
000000010,0	30,0	Ada
000000020,0	20,0	Ada
000000030,0	10,0	Ada
000000040,0	0	Ada
000000040,5	-0,5	Ada
000000041,0	-1,0	Ada
000000050,0	-10,0	Ada

Pengujian komunikasi serial dapat dilakukan dengan cara menekan tombol 'test serial' pada program Komputer. Pada saat user menekan tombol 'test serial' maka data yang dikirimkan oleh komputer akan diterima oleh sistem dan dikembalikan lagi ke komputer. Untuk melihat keberhasilan komunikasi serial dapat dilihat pada jendela 'Log' yang ada pada program komputer.

8. Kesimpulan

- Odometer telah dibuat sesuai dengan rencana, dengan kemampuan yaitu pengukuran jarak hingga 999.999.999,9 km dan pengukuran kecepatan dari 0 km/jam sampai 250 km/jam.
- Sistem dapat memberikan peringatan servis sebelum waktunya atau tepat pada waktunya sesuai dengan pengaturan servis. Sehingga dapat mencegah kerusakan kendaraan yang disebabkan oleh kelalaian pengguna.
- Dari hasil pengujian, sistem memiliki akurasi pengukuran jarak yang cukup baik. Meskipun terdapat ketidaksesuaian antara hasil pengukuran dengan jarak sesungguhnya, hal ini disebabkan karena faktor-faktor dari luar yaitu tekanan ban dan kondisi jalan.
- Pengukuran kecepatan diambil dengan cara menghitung jumlah putaran dalam satu detik, oleh karena itu display hasil pengukuran kecepatan di-update tiap satu detik.

Daftar Pustaka

1. Pranta, Anthony, *Pemrograman Borland Delphi*, Yogyakarta: Andi, 2001.
2. *Atmel AT89C52 Microcontroller Data Book*, San Jose, California: Atmel Corporation, 1995.
3. *Atmel AT24C16 2-Wire Serial EEPROM Data Book*, San Jose, California: Atmel Corporation, 2001.
4. *Dallas DS1307 64x8 Serial Real Time Clock Data Book*, San Gabriel Dr., Sunnyvale: Dallas Corporation, 2001.
5. Cantu, Marco, *Mastering Delphi 5*, San Francisco: SYBEX, 1999.