

BAB II

LANDASAN TEORI

2.1 Perangkat Lunak (*Software*)

2.1.1 HTTP

HTTP (*Hypertext Transfer Protocol*) merupakan sebuah protokol dalam sebuah sistem jaringan pada web yang digunakan untuk mengirim semua *file* atau data (*resources*) secara virtual pada *world wide web*, apakah itu *file* HTML, *file* gambar, hasil dari sebuah proses atau apapun. Biasanya, HTTP berjalan melalui soket TCP/IP.

Browser merupakan sebuah HTTP *client*, hal ini disebabkan karena *browser* tersebut mengirimkan perintah ke HTTP *server* (*Web server*), yang kemudian akan mengirimkan respon balik ke *client*. *Port* standar untuk HTTP *server* adalah 80, tetapi juga bisa menggunakan *port-port* yang lain.

2.1.2 Apache *Web server*

Apache merupakan turunan dari *web server* yang dikeluarkan oleh NSCA (<http://hoohoo.nsga.uiuc.edu>), yaitu NSCA HTTPd sekitar tahun 1995-an. Pada dasarnya apache adalah "A PatCHy" (*patch*) dan pengganti dari NCSA HTTPd. Apache *web server* merupakan tulang punggung dari *World Wide Web* (www). *Web server* menunggu permintaan dari *client* yang menggunakan *browser*, seperti netscape navigator, internet explorer, mozilla, lynx dan lain-lain. *Web server* dalam berkomunikasi dengan *client*-nya menggunakan protokol HTTP (*Hyper Text Transfer Protocol*). Apache berada dibawah GNU, *General Public Licensi*

yang bersifat *free* sehingga apache dapat di *download* gratis pada alamat <http://apache.org>. Saat ini apache banyak digunakan sebagai *web server* untuk portal-portal besar. Adapun pertimbangan dalam memilih apache adalah :

1. Apache termasuk kedalam kategori *free software* (*software* gratis).
2. Instalasi apache sangat mudah.
3. Mampu beroperasi pada banyak *platform* sistem operasi, seperti AIX 3.1, BSDI 2.0, free BSD 2.1, HP-UX 9.07, IRIX 5.3, Linux, SolarisX86, windows dan lain-lain.
4. Mudah dalam pengkonfigurasian.
5. Apache mudah dalam penambahan *peripheral* lainnya ke dalam *platform webserver*, misalnya menambahkan modul.

Ada beberapa ciri khas dari Apache, yaitu:

1. Apache sangat cepat dalam merespon *client* melebihi *server* NCSA.
2. Apache menyediakan *fitur* untuk *multihome* dan *virtual server*.
3. Apache mempunyai level-level pengamanan.
4. Apache mempunyai komponen dasar terbanyak di antara *web server* lainnya.
Apache termasuk dalam *web server* yang lengkap.
5. Performansi dan konsumsi sumberdaya dari *webserver* apache tidak terlalu banyak sekitar 20 MB untuk file-file dasar dan setiap daemonya hanya memerlukan sekitar 950 KB memori.
6. Mendukung transaksi yang aman menggunakan SSL (*Secure Socket Layer*).
7. Banyak dukungan melalui *web*.
8. Kompatibilitas yang tinggi.

2.1.3 MySQL

MySQL adalah sebuah *database server* buatan T.c.X. Data konsulat AB, swedia. *Database* MySQL banyak di gunakan di internet karena kehandalannya. MySQL tidak banyak membutuhkan ruang hardisk yang besar untuk aplikasinya, dan mudah digunakan pada *database server*, sangat ideal untuk aplikasi yang kecil dan menengah. Keistimewaan MySQL adalah sebagai berikut:

1. *Language support*

Database MySQL dapat menampilkan pesan *error* dalam bahasa *czech*, belanda, inggris, estonia, perancis, jerman, hongaria, italia, norwegia, dan lain-lain. MySQL menggunakan ISO-8859-1 karakter tertentu untuk data dan pengurutan. Karakter tertentu untuk data dan pengurutan dapat dipilih ketika mengkompilasi *source*.

2. Bahasa pemrograman API untuk mengakses *database*

Aplikasi *database* MySQL dapat ditulis dengan bahasa tertentu seperti C, Pearl, PHP dan lain-lain.

3. *Large Table*

MySQL menyimpan masing-masing tabel dalam *database* seperti file, terpisah dalam direktori *database*. Ukuran maksimum *database* berkisar antara 4GB dan sistem operasi mendekati ukuran file maksimum.

4. Kecepatan, kekuatan dan kemudahan yang digunakan

MySQL lebih cepat tiga atau empat kali dari *database* komersial yang lainnya. MySQL sangat mudah untuk dikendalikan dan tidak membutuhkan *database* administrator terlatih untuk menginstal MySQL.

5. *Cost Advantage*

MySQL adalah *database* relasional yang *open source*. Didistribusikan secara gratis untuk Unix/Linux dan OS/2 dan untuk *platform* microsoft kita membutuhkan lisensi setelah mencobanya selama 30 hari.

2.1.4 PHP

PHP adalah kependekan dari *PHP Hypertext Preprocessor*, dalam bahasa interpreter yang mempunyai kemiripan dengan bahasa C dan Pearl yang mempunyai kesederhanaan dalam perintah. PHP dapat digunakan bersama dengan HTML sehingga memudahkan dalam membangun aplikasi *web* dengan cepat yang dieksekusi dan bekerja di komputer *server* kemudian *script* PHP akan mengirimkan hasil eksekusi ke komputer *client* melalui jalur http. PHP dapat digunakan untuk mengupdate *database*, menciptakan *database* dan mengerjakan perhitungan matematika. Seperti halnya dengan program *open source* lainnya, PHP dibuat dibawah GNU, *General Public Lisence* yang dapat di-*download* gratis di <http://www.php.net>.

2.1.5 Perl

Perl merupakan bahasa interpreter yang digunakan untuk optimasi pengolahan teks, mengelola informasi dari dari file-file teks tersebut, dan menampilkan laporan yang dibasiskan dalam informasi yang dibuat. Perl merupakan bahasa yang baik untuk memanipulasi berbagai teks dan manajemen task tersebut. Perl dikembangkan untuk menjadi sebuah bahasa yang praktis untuk digunakan, efisien dan lengkap serta menggunakan sumber daya yang terbilang

minimum. Bahasa perl pertama kali dibuat oleh Larry Wall dan tentunya dengan bantuan berbagai kontributor pengembang Perl diseluruh dunia.

2.2 Mikrokontroler PIC16F84A

2.2.1 Sekilas Mikrokontroler PIC16F84A

Mikrokontroler PIC16F84A merupakan salah satu mikrokontroler dari keluarga *PICmicro* yang populer digunakan sekarang ini, mulai dari pemula hingga para profesional. Hal tersebut karena PIC16F84A sangat praktis dan menggunakan teknologi *FLASH memory* sehingga dapat deprogram-hapus hingga sepuluh ribu kali. Keunggulan mikrokontroler jenis RISC ini dibanding dengan mikrokontroler 8-bit lain dikelasnya terutama terletak pada kecepatan dan kompresi kodenya. Selain itu PIC16F84A juga tergolong praktis dan ringkas karena memiliki kemasan 18 pin dengan 13 jalur I/O. Meskipun demikian, mikrokontroler PIC16F84A bukan termasuk mikrokontroler yang memiliki fitur yang lengkap. Dalam keluarga PICmicro, PIC16F84A tergolong mikrokontroler skala sedang (*Mid-Range*).

Anggota keluarga PICmicro buatan *microchip Inc* cukup banyak. Ada yang menggunakan *FLASH memory* dan ada pula yang jenis OTP (*One Time Programmable*). Mikrokontroler dari keluarga PICmicro yang populer, antara lain PIC12C508, PIC16C54, PIC16F84, PIC16F84A, dan PIC16F877.

2.2.2 Fitur PIC16F84A

Sebenarnya PIC16F84A bukanlah mikrokontroler yang istimewa dalam keluarga *PICmicro*. Namun demikian, PIC16F84A cukup mudah dipelajari dan

dapat dibidang memiliki kemampuan yang handal sebagai mikrokontroler yang hanya memiliki kemasan 18 pin.

Fitur-fitur pada PIC16F84A antara lain:

1. Hanya menggunakan 35 set instruksi.
2. Semua instruksi berukuran 14 bit.
3. Data berukuran 8 bit.
4. Memori program berukuran 1024 x 14 pada *flash memory* (10.000 kali baca-tulis).
5. 68 byte data RAM.
6. 64 byte data EEPROM
7. 15 register kegunaan khusus.
8. Delapan tingkat *stack* perangkat keras.
9. Empat sumber sela interupsi
 - a. Pena sela luar (INT).
 - b. Limpahan (*overflow*) pada *timer* (TMR0)
 - c. Port B (bit 4 ~ bit 7) bila berubah kondisi logikanya.
 - d. Penulisan data pada EEPROM telah selesai.
10. Memiliki 13 pin I/O (5 *port* A dan 8 *port* B).
11. Dapat langsung *men-drive* LED.
12. Terdapat *Timer* 8 bit dengan pembagi 8 bit.
13. *Watchdog Timer* (WDT) dengan internal osilator.
14. *Fuse* untuk kode pengamanan.
15. Mode pengurangan daya (*sleep*).

16. Dapat memakai beberapa jenis osilator.

- a. Osilator RC (RC).
- b. Kristal / Resonator (XT).
- c. Kristal kecepatan tinggi (HS).
- d. Kristal frekuensi rendah (LP).

17. Pemrograman didalam sistem.

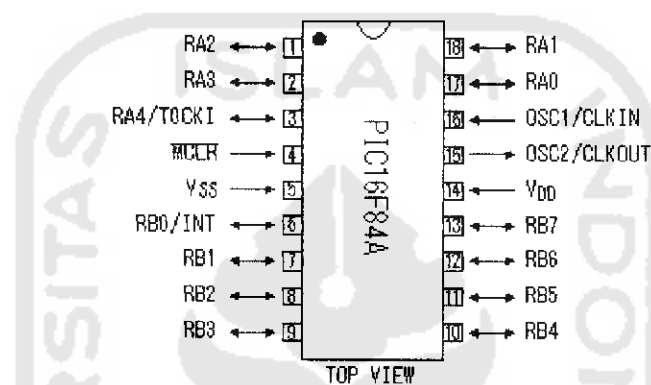
18. Berukuran fisik hanya 18 pin.

Mikrokontroler PIC16F84A terbuat dengan teknologi CMOS (*Complementary Metal Oxide Semiconductor*) sehingga dapat didayai dengan baterai. Mikrokontroler ini dapat didayai dengan menggunakan catu daya sebesar 2V sampai dengan 6V sehingga cukup fleksibel dalam pemakaiannya. Untuk pemrograman mikrokontroler ini dapat dipakai berbagai macam *programmer* yang tersedia di pasar terutama adalah *universal programmer*. Namun karena harganya sangat mahal penulis merakit *programmer*-nya sendiri yang jauh lebih murah (untuk mendapatkan rangkaiannya silahkan *browsing* di google dengan kata kunci *PIC programmer*.)

2.2.3 Deskripsi pin-pin

Mikrokontroler PIC16F84A diproduksi dalam kemasan 18 pin PDIP (*Plastic Dual In Line*) maupun 18 pin SO (*Small Outline*). Namun yang banyak terdapat dipasaran adalah dalam kemasan PDIP. Pin-pin untuk I/O sebanyak 13 pin, yang terdiri atas 5 pin untuk port A dan 8 pin untuk port B. Ada beberapa pin pada mikrokontroler ini yang memiliki fungsi ganda, yaitu pin3, pin4, pin12 dan

pin13. Pin3 dapat berfungsi sebagai pin I/O untuk port A atau sebagai jalur input untuk *clock* eksternal. Pin4 berfungsi sebagai pin reset dan juga dapat digunakan sebagai pin V_{pp} saat mode pemrograman. Sedangkan pin12 dan pin13 masing-masing berfungsi sebagai pin I/O dari port B namun sekaligus juga dapat berfungsi sebagai pin *clock* dan data masukan pada pemrograman mikrokontroler.



Gambar 2.1 Pin-pin mikrokontroler PIC16F84A

Tabel 2.1 Daftar pin-pin pada PIC16F84A beserta fungsinya

No.Pin	Nama Pin	Deskripsi
1	RA2	Pin kedua I/O port A
2	RA3	Pin ketiga I/O port A
3	RA4/TOCKI	I/O port A, input clock eksternal
4	MCLR/Vpp	Input reset, tegangan V_{pp} untuk pemrograman
5	Vss	Ground, 0V
6	RB0	Pin ke-nol I/O port B
7	RB1	Pin pertama I/O port B
8	RB2	Pin kedua I/O port B
9	RB3	Pin ketiga I/O port B
10	RB4	Pin keempat I/O port B
11	RB5	Pin kelima I/O port B
12	RB6	Pin keenam I/O port B, jalur clock pada mode pemrograman
13	RB7	Pin ketujuh I/O port B, jalur data pada mode pemrograman

14	Vdd	Pin untuk sumber tegangan positif (+5V)
15	OSC2	Pin osilator
16	OSC1	Pin osilator
17	RA0	Pin ke-nol I/O port A
18	RA1	Pin pertama I/O port A

2.2.4 Organisasi Memori

Memori pada PIC16F84A dapat dipisahkan menjadi 2 blok memori, satu untuk memori program dan satu untuk memori data. Memori EEPROM dan register GPR didalam RAM merupakan memori data sedangkan memori FLASH merupakan memori program.

2.2.4.1 Memori Program

Memori program direalisasikan dalam teknologi FLASH memori yang memungkinkan pemrogram melakukan program-hapus hingga sepuluh ribu kali. Pemrograman PIC16F84A dilakukan sebelum dipasang pada rangkaian aplikasi, atau ketika sistem sudah terpasang namun dikehendaki adanya *up-dating* pada program di dalamnya. Pemrograman berulang biasanya dilakukan pada saat pengembangan dan penyempurnaan sistem. Ukuran memori program untuk PIC16F84A adalah 1024 lokasi dengan lebar kata (*word*) 14 bit. Lokasi 00h dan 04h berturut-turut digunakan untuk vektor reset dan vektor interupsi.

2.2.4.2 Memori Data

Memori data PIC16F84A terdiri atas memori EEPROM dan memori RAM. EEPROM berukuran 64 byte dengan isi memori bersifat *non-volatile* (tidak akan hilang meskipun catu daya dimatikan). Memori ini tidak dialamati secara langsung, sehingga pengaksesannya dilakukan melalui register EEADR dan

EEDATA. Memori EEPROM biasa digunakan untuk menyimpan parameter-parameter penting, misalnya parameter untuk keperluan kalibrasi.

Memori RAM untuk data menempati 68 lokasi, mulai 0x0C hingga 0x4F. Lokasi RAM juga dikenal sebagai GPR (*General Purpose Register*). Register ini dapat diakses tanpa memperhatikan di bank mana kita saat itu.

SFR (*Special Function Register*) merupakan register-register yang digunakan untuk keperluan khusus bagi mikrokontroler. Register-register tersebut terletak pada 12 lokasi pertama di bank 0 maupun bank 1.

Address	Bank 0	Bank 1	Address
00h	TRISA	←	80h
01h	TRIO	OPTION_REG	81h
02h	PCL	←	82h
03h	STATUS	←	83h
04h	FSR	←	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	Unimplemented	←	87h
08h	EEDATA	EEDCON1	88h
09h	EEADR	←	89h
0Ah	PCLATH	←	8Ah
0Bh	INTCON	←	8Bh
0Ch - 4Fh	GPR	←	8Ch - CFh

Gambar 2.2 Peta memori pada PIC16F84A

2.2.5 Bank Memori

Selain dibagi atas SFR dan GPR, peta memori pada PIC16F84A juga dibagi atas bentuk *bank* (semacam halaman). Peta memori PIC16F84A dibagi atas dua bank, yaitu bank 0 dan bank 1. Bank 1 digunakan untuk kendali operasi yang aktual, misalnya untuk menetapkan bit-bit mana pada port A yang bertindak sebagai input dan mana yang output. Sedangkan bank 0 digunakan untuk

keperluan manipulasi data. Pemilihan *bank* dilakukan melalui bit RP0 dan RP1 pada register STATUS. Meski demikian, hanya bit RP0 yang efektif digunakan.

Untuk membuat salah satu bit pada port A bernilai tinggi (“1”), yang pertama harus dilakukan adalah menetapkan bit itu sebagai output. Penetapan ini harus dilakukan pada *bank* 1. Dengan demikian kita harus pindah ke *bank* 1, kemudian menetapkan salah satu bit/pin sebagai output. Setelah itu kita kembali lagi ke *bank* 0, dan mengirimkan logika “1” pada bit/pin yang bersangkutan.

2.2.6 Pemrograman PIC16F84A

Penulisan program *PICmicro* akan lebih mudah apabila kita menggunakan paket program yang sesuai. MPLAB merupakan salah satu paket program pada lingkungan windows yang memudahkan kita untuk mengembangkan suatu kode program. MPLAB mencakup berbagai *software* pendukung untuk pengembangan sistem *PICmicro*, seperti untuk keperluan *edit* program, *assembly* maupun simulasi.

MPLAB terdiri atas beberapa bagian :

1. *Project Manager*, yang bertugas menyatukan berbagai *file* dalam satu *project*.
2. *Text Editor* untuk keperluan pembangkitan dan pemrosesan program.
3. Simulator program untuk mensimulasikan fungsi program pada mikrokontroler.

Sebelum menginstal MPLAB, perlu diperhatikan sistem minimal untuk menjalankan MPLAB, yaitu :

1. Kompatibel pada PC 486 atau yang lebih tinggi.
2. Microsoft windows 3.1x, 95 atau versi yang lebih tinggi.

3. *VGA graphic card.*
4. 8 MB memori RAM (dianjurkan 32 MB).
5. 20 MB *space* memori pada *hard disk*
6. *Mouse.*

2.2.6.1 Programmer.

Setelah berhasil mengkompilasi suatu program tanpa *error*, langkah selanjutnya adalah men-*download* file .hex hasil kompilasi ke chip PIC16F84A. Proses ini dilakukan menggunakan suatu perangkat yang dinamakan *programmer* PIC. Ada beberapa macam dan tipe *programmer* yang dapat digunakan untuk keperluan ini. Beberapa programmer yang mudah digunakan antara lain JDM *programmer*, Universal *programmer*, WpicProg16, NOPPP dan lain-lain. *Programmer-programmer* tersebut ada yang menggunakan *port* serial dan ada pula yang menggunakan *port* paralel sebagai terminalnya. Dalam hal ini penulis menggunakan *programmer* NOPPP dengan pertimbangan, *programmer* ini sangat mudah untuk dibuat, komponen banyak di pasaran, dan tidak membutuhkan biaya yang besar dalam pembuatannya. NOPPP menggunakan *port* paralel sebagai terminalnya. *Programmer* ini menggunakan *power supply* dengan dua *output* yaitu +5V dan +12V sampai +14V. Hal ini disebabkan karena mikrokontroler PIC16F84A itu sendiri menggunakan 2 jenis input masukkan untuk *power supply*, +5V untuk catu daya mikro itu sendiri sedangkan +12V-14V adalah catu daya untuk membuat MCLR (pin4) berubah ke mode *write* sehingga PIC dapat menerima kode hex yang *transfer* dari PC.

2.3 Perangkat Keras (*Hardware*)

2.3.1 Motor Servo

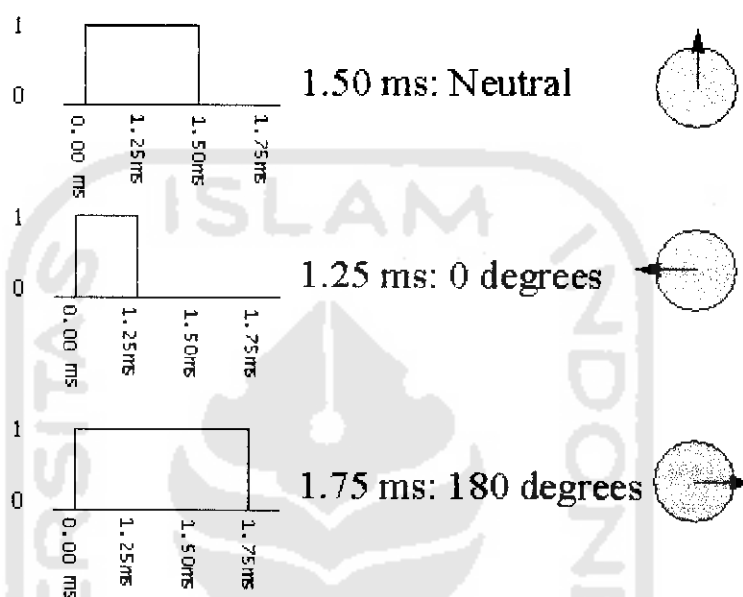
Kalangan *hobbies* biasanya menggunakan motor servo untuk keperluan mereka. Alat ini sangat populer dalam mengontrol sistem kendali pada mobil *radio control*, perahu, pesawat terbang model, mobil dan *robotic*. Harganya pun relatif murah untuk menyalurkan hobi di bidang elektronika.

Servo merupakan motor khusus yang disertai dengan rangkaian tambahan atau *feedback* posisi untuk menentukan posisi sumbu motor servo yang diinginkan. Rangkaian *feedback* posisi tersebut berupa potensiometer (*resistor variable*) yang dapat berputar sesuai dengan putaran *gear* utama motor servo. Rangkaian *feedback* ini senantiasa membandingkan nilai yang diberikan oleh perintah program terhadap nilai aktual potensiometer itu sendiri. Selisih atau *error* dari perbandingan kedua nilai tersebut akan dijadikan rujukan untuk menentukan arah posisinya, sehingga motor berputar untuk mengeliminasi selisih atau *error* menjadi nol, dengan demikian akan tercapailah posisi yang diinginkan.

2.3.1.1 Motor Servo Belum Dimodifikasi

Servo terdiri atas jenis yang belum dimodifikasi dan jenis yang sudah dimodifikasi. Bedanya, jenis servo yang belum dimodifikasi hanya dapat memutar sumbu servo ke posisi yang dikehendaki dengan jarak maksimal berkisar 90 sampai 180 derajat oleh sinyal yang dihubungkan ke pengendali servo. Servo ini mampu bergerak ke arah posisi yang diinginkan dengan cukup akurat. Servo ini juga memiliki torsi (hasil perkalian antara gaya rotasi servo dengan panjang lengan) yang cukup besar yang terhubung pada sumbu.

Servo yang belum dimodifikasi memiliki posisi tengah atau nol derajat yang merupakan titik tengah antara awal dan akhir jangkauan motor servo. Posisi titik tengah ini dicapai dengan memberikan lebar pulsa tertentu pada servo yang belum dimodifikasi.



Gambar 2.3 Contoh deretan pulsa

Servo didesain untuk menterjemahkan lebar pulsa positif (+5V) antara 1 sampai 2 mili detik (ms). Lebar pulsa ini merupakan pengontrol gerakan servo. Gerakan ini diatur oleh dua keadaan, yaitu pada tegangan sebesar 5 volt dengan lebar pulsa antara 1 sampai 2 milidetik (ms) dan pada keadaan ketika tanpa adanya tegangan dengan lebar pulsa yang dapat bervariasi antara 10 sampai 40 milidetik (ms) sesuai dengan rekomendasi. Pada lebar pulsa rendah ini tidak akan banyak mengubah atau mengurangi gerakan servo kerana informasi arah posisi gerakan servo dibawa oleh lebar pulsa positif 5 volt itu sendiri. Lebar pulsa rendah hanya menentukan kehalusan gerakan putaran servo. Jika lebar pulsa

rendah terlalu besar, akan mengakibatkan gerakan putaran servo kurang halus. Lebar pulsa rendah antara 10 sampai 40 ms merupakan rekomendasi agar gerakan servo bisa halus.

Pada servo yang belum dimodifikasi, jika lebar pulsa 5 volt diberikan sebesar 1,5 milidetik, sumbu servo akan berputar menuju titik tengah (posisi netral). Jika pulsa yang diberikan lebih kecil dari 1,5 milidetik sumbu motor servo akan bergerak menuju arah 0 derajat. Jika pulsa yang diberikan lebih besar dari 1,5 milidetik, sumbu motor servo akan bergerak menuju arah 180 derajat.

Motor servo memiliki 3 jenis kabel yaitu merah untuk *power* (5 volt), hitam untuk *ground*, dan putih untuk sinyal pengendali servo yang dihubungkan dengan I/O dari mikrokontroler.

Ketika sebuah servo dihubungkan dengan sinyal pengendali dengan sumber sinyalnya berupa pulsa, alat pengontrol yang mengirimkan deretan pulsa dapat diprogram untuk menghasilkan bentuk gelombang pada pin-pin I/O sehingga sinyal pengontrol servo dapat terkirim.

2.3.1.2 Motor Servo Sudah Dimodifikasi

Motor servo dapat dimodifikasi agar dapat berputar secara kontinyu (360 derajat), yaitu dengan cara memanipulasi *gear-gear* yang terdapat pada motor servo dan rangkaian *feedback* yang berupa potensiometer.

Servo yang sudah dimodifikasi, putaran akan bergerak kontinyu. Dengan memberikan lebar pulsa sebesar 1,3 ms akan menyebabkan motor servo berputar searah jarum jam. Sedangkan jika lebar pulsa diberikan sepanjang 1,7 ms akan menyebabkan sumbu servo berputar berlawanan dengan arah jarum jam. Namun

pulsa sepanjang 1,5 ms akan mengakibatkan sumbu servo tetap pada posisinya (tidak berputar). Hal ini sama dengan yang terjadi pada servo yang belum dimodifikasi, yaitu dikarenakan servo berada pada posisi netral ketika lebar pulsa yang diterimanya 1,5 ms.

2.3.2 Web Camera

Untuk memberikan *feedback* kepada *user* yang berada pada *remote location*, gambar yang ditangkap oleh sebuah *web camera* akan dikirimkan ke *client browser*. Prinsip inilah yang dipakai oleh perangkat lunak dari *web camera*. Gambar tersebut dapat dengan baik dikirimkan melalui *netscape* atau melalui *internet explorer* (untuk *internet explorer* diperlukan *script* khusus untuk dapat menangkap gambar yang dikirimkan oleh *web camera*) yang berada pada *client*. Kamera ini digunakan untuk mengetahui perubahan dari posisi motor servo sesuai dengan perintah yang dikirimkan oleh *user*.

2.4 Komunikasi Serial

Saat ini, hampir seluruh komputer dilengkapi dengan 2 *serial port* dan 1 *parallel port*. Walaupun keduanya sama-sama digunakan untuk melakukan komunikasi dengan perangkat luar, namun cara kerja keduanya sangatlah berbeda.

Pada *parallel port*, proses pengiriman dan penerimaan data 8 bit dilakukan secara bersamaan dengan menggunakan 8 kabel berbeda. Hal ini akan membuat proses transfer data akan menjadi sangat cepat, namun akan menggunakan jumlah kabel yang sangat banyak. *Parallel port* ini sangat banyak digunakan untuk komunikasi antara PC dan printer dan jarang sekali digunakan untuk jenis

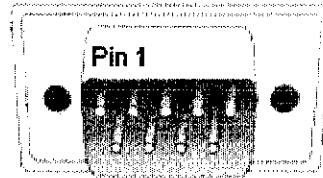
komunikasi yang lain. Pada *serial port*, pengiriman dan penerimaan data dilakukan per-bit melalui 1 kabel. Walaupun waktu pengirimannya akan menjadi 8 kali lebih lambat bila dibandingkan dengan *parallel port*, namun hanya memerlukan sedikit kabel untuk proses pengiriman datanya. Pada kenyataannya, untuk melakukan komunikasi 2 arah sangat dimungkinkan dengan menggunakan 3 kabel saja, yaitu 1 untuk proses pengiriman data, 1 untuk proses penerimaan data dan 1 lagi untuk *ground*.

2.4.1 Komunikasi melalui bits

Pengiriman data bit dilakukan setelah *start bit* dikirimkan terlebih dahulu. Data yang dikirimkan tersebut dapat berjumlah 5, 6, 7 atau 8 bit, tergantung dari angka yang ingin kita gunakan. Namun sebelumnya *transmitter* dan *receiver* harus di *set* agar menggunakan data bit yang sama. Namun kebanyakan perangkat komunikasi menggunakan 7 atau 8 data bit.

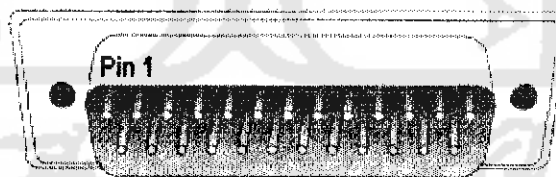
Yang perlu diperhatikan adalah ketika kita menggunakan 7 data bit maka kode ASCII yang bisa dikirimkan maksimal adalah 127, begitu juga jika 5 data bit, maka angka maksimal yang bisa dikirimkan tidak lebih dari 31. *Stop bit* akan dikirim setelah semua data bit selesai dikirimkan. *Stop bit* bernilai 1 atau logika tinggi, dan ia dapat dibaca dengan benar oleh sistem walaupun data sebelumnya juga bernilai 1 atau logika tinggi.

Berikut adalah gambar dari 2 jenis *serial port*, yaitu 9 pin *serial connector* dan 25 pin *serial connector*.



<u>Pin</u>	<u>Signal</u>	<u>In/Out</u>
1	CD - Carrier Detect	In
2	RD - Receive Data	In
3	TD - Transmit Data	Out
4	DTR - Data Terminal Ready	Out
5	SG - Signal Ground	---
6	DSR - Data Set Ready	In
7	RTS - Request to Send	Out
8	CTS - Clear to Send	In
9	RI - Ring Indicator	In

Gambar 2.4 9 pin serial connector



<u>Pin</u>	<u>Signal</u>	<u>In/Out</u>
1	Chassis Ground	---
2	TD - Transmit Data	Out
3	RD - Receive Data	In
4	RTS - Request to Send	Out
5	CTS - Clear to Send	In
6	DSR - Data Set Ready	In

7	SG - Signal Ground	---
8	CD - Carrier Detect	In
9	+ Transmit Current Loop Return	Out
11	- Transmit Current Loop Data	Out
18	+ Receive Current Loop Data	In
20	DTR - Data Terminal Ready	Out
22	RI - Ring Indicator	In
25	- Receive Current Loop Return	In

Gambar 2.5 25 pin serial connector

