

1.5 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir dengan judul “Perancangan Touchpad PS 2 Berbasis Mikrokontroler AT89C51” terdiri dari 5 bab, dengan masing-masing bab adalah sebagai berikut :

BAB I PENDAHULUAN

Bab ini membahas tentang latar belakang masalah yang akan diteliti, rumusan masalah, batasan masalah, tujuan penelitian dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini memuat teori-teori yang berhubungan dengan penelitian yang akan dirancang dan semua teori yang merupakan acuan dari perancangan.

BAB III PERANCANGAN SISTEM

Bab ini menjelaskan perancangan sistem yang digunakan, cara mengimplementasikan rancangan.

BAB IV ANALISIS DAN PEMBAHASAN

Bab ini membahas perihal hasil-hasil pengujian yang diperoleh dari uji coba perakitan.

BAB V PENUTUP

Bab ini memuat kesimpulan dan saran-saran dari proses perancangan yang telah dilakukan.

BAB II

DASAR TEORI

2.1 *Touchpad*

Touchpad adalah sebuah *input device* yang sering digunakan pada komputer *laptop*. *Touchpad* digunakan sebagai penggerak *cursor* dengan memanfaatkan pergerakan jari pemakai, atau dengan kata lain sebagai pengganti fungsi *mouse* pada komputer *desktop* biasa. *Touchpad* memiliki ukuran luas yang bervariasi tetapi jarang ada yang ukurannya lebih luas dari 20cm² (8in²).

Touchpad umumnya bekerja dengan cara mendeteksi sifat *capacitance* dari jari manusia. *Sensor capacitance* ditempatkan sepanjang sumbu vertikal dan horizontal dari permukaan *touchpad*. Posisi/lokasi dari jari ditentukan dari kerja kombinasi antara *sensor capacitance* sumbu vertikal dan horizontal. Itulah sebabnya mengapa *touchpad* tidak dapat mendeteksi sentuhan dari sebuah pensil atau benda lain yang sejenis. Penggunaan sarung tangan juga dapat mempengaruhi kerja *sensor capacitance* walaupun kadang-kadang juga dapat berhasil. Keringat atau jari yang basah dapat juga menimbulkan masalah pada *touchpad* yang sangat tergantung dari pengukuran *capacitance* oleh *capacitance sensor*.

Touchpad biasanya memiliki beberapa buah tombol. Tombol-tombol itu bisa terletak di atas atau di bawahnya. Fungsi dari tombol-tombol pada *touchpad* sama seperti fungsi tombol dari *mouse*. Tergantung dari model *touchpad* dan

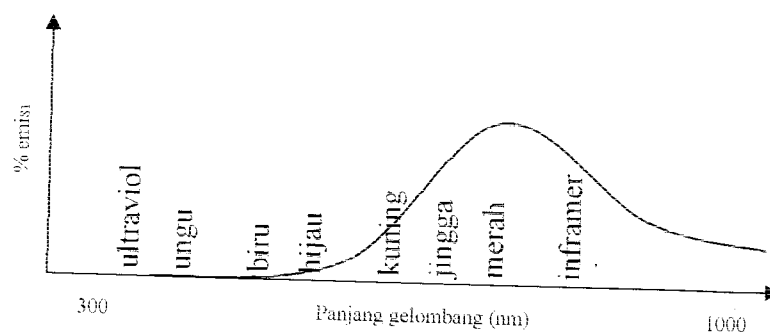
2.2 Sensor

Sensor/tranduser berfungsi untuk mengubah besaran fisis menjadi besaran listrik, sehingga akan lebih mudah diproses oleh rangkaian elektronik. *Sensor* yang digunakan ada dua buah yaitu *sensor infrared* dan *push button*.

2.2.1 Infrared

Cahaya *infrared* merupakan cahaya yang tidak tampak. Jika dilihat dengan spektroskop cahaya, maka radiasi inframerah akan tampak pada spektrum elektromagnetik dengan panjang gelombang diatas panjang gelombang cahaya merah. Dengan panjang gelombang ini maka cahaya inframerah tidak akan tampak oleh mata, namun radiasi panas yang dihasilkan masih terasa/dideteksi.

Benda yang menghasilkan panas, pada dasarnya juga menghasilkan radiasi inframerah termasuk tubuh manusia maupun tubuh binatang. Cahaya inframerah walaupun memiliki panjang gelombang yang sangat panjang tetapi tidak dapat menembus bahan-bahan yang tidak dapat melewatkan cahaya yang tampak, sehingga cahaya inframerah tetap mempunyai karakteristik seperti halnya cahaya yang tampak oleh mata.

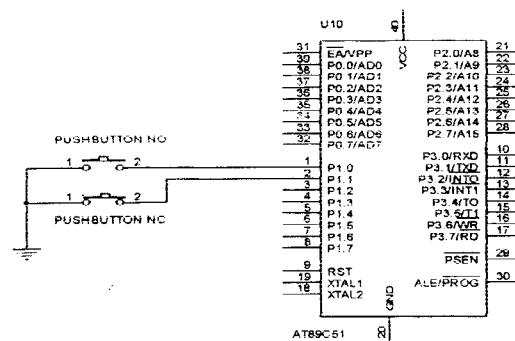


Gambar 2.1 Emisi Cahaya

mendefinisikan lama agar photodetektor inframerah merespons cahaya yang datang pada area penerimaan. Jika *respond time* terlalu besar, maka photodetektor tidak dapat merespons sinyal cahaya yang dimodulasi dengan carrier berfrekuensi tinggi. Hal ini dapat mengakibatkan *data loss*.

2.2.2 Push Button

Push button merupakan sebuah saklar/*switch* yang berfungsi menyambung dan memutuskan arus listrik. Pada *push button*, keadaan *contact* saat ditekan tidak sama dengan pada saat dilepas. Ada dua buah tipe *push button* yaitu NO (*Normaly Open*) dan NC (*Normaly Close*). Pada *push button* tipe NO, posisi saklar saat tidak ditekan dalam keadaan membuka (*open*) sehingga arus tidak akan mengalir, akan tetapi pada saat *push button* ditekan, posisi saklar akan menutup (*close*) sehingga arus akan mengalir. Pada tipe NC merupakan kebalikan dari tipe NO, posisi saklar akan menutup (*close*) pada saat *push button* tidak ditekan sehingga arus dapat mengalir, akan tetapi pada saat *push button* ditekan arus tidak akan dapat mengalir karena posisi saklar membuka (*open*).



Gambar 2.3 *Push Button* type NO dan type NC

tertinggi. Bit ini jarang digunakan dalam program, namun mikrokontroler secara implisit menggunakannya dalam operasi aritmatika bilangan BCD.

3. *Flag 0*

Bit ini menunjukkan apakah hasil operasinya nol atau tidak. Apabila hasil operasinya adalah nol, maka bit ini akan *diset 0*. Bit ini juga digunakan untuk membandingkan dua buah data. Bila kedua data sama maka akan *diset 1*, sedangkan jika kedua data berbeda akan *diset 0*.

4. Bit Pemilih *Register Bank*

Register Bank Select Bit (RS0 dan RS1) atau Bit Pemilih *Register Bank* digunakan untuk menentukan lokasi dari *Register Bank* (R0 hingga R7) pada memori, yaitu pada *Bank 0*, *Bank 1*, *Bank 2* atau *Bank 3*. RS0 dan RS1 akan selalu bernilai nol setiap kali sistem *direset* sehingga lokasi dari R0 hingga R7 akan berada dialamat 00H hingga 07H.

5. *Flag Overflow*

Flag Overflow akan *diset* jika pada operasi aritmatik menghasilkan bilangan yang lebih besar daripada 128 atau lebih kecil dari -128.

6. *Bit Parity*

Bit Paritas akan *diset* jika jumlah bit 1 dalam akumulator adalah ganjil dan akan *clear* jika jumlah bit 1 dalam akumulator

adalah genap. *Bit parity* ini juga digunakan untuk proses yang berhubungan dengan *serial port* yaitu sebagai *check sum*

d. *Register B*

Register B digunakan bersama akumulator untuk proses aritmatik selain dapat juga difungsikan sebagai *register* biasa. *Register* ini juga bersifat *bit addressable*.

e. *Stack Pointer*

Stack Pointer merupakan sebuah *register* 8 bit yang terletak di alamat 81H. Isi dari *Stack Pointer* ini merupakan alamat dari data yang disimpan di *stack*. *Stack Pointer* dapat *di edit* atau dibiarkan saja mengikuti standar sesudah terjadi *reset*. Jika *Stack Pointer* diisi data 5FH, area untuk proses penyimpanan dan pengambilan data dari dan ke *stack* adalah sebesar 32 *byte*, yaitu antara 60H hingga 7FH karena AT89C51 mempunyai *Internal RAM* sebesar 128 *byte*. Proses yang berhubungan dengan *stack* ini biasa dilakukan oleh instruksi-instruksi *Push*, *Pop*, *Acall* dan *Lcall*.

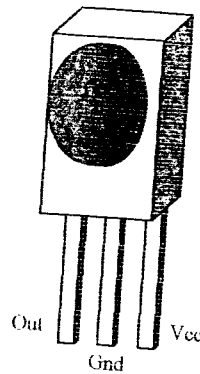
f. *Data Pointer*

Data Pointer atau DPTR merupakan *register* 16 bit dan terletak pada alamat 82H untuk DPL dan 83H untuk DPH. DPTR biasa digunakan untuk mengakses *source code* ataupun data yang terletak di memori eksternal.

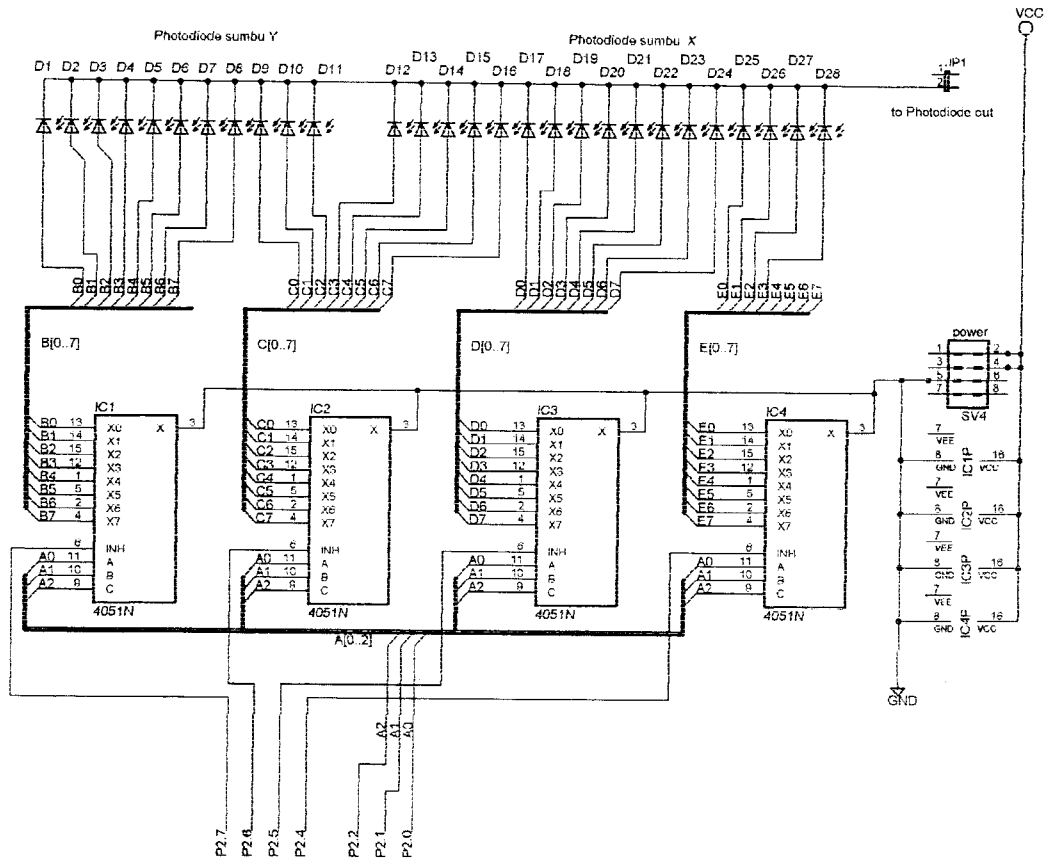
2.6 Modulator Dan Demodulator Infrared

Modulasi dilakukan dengan tujuan memperjauh jangkauan *infrared* dan membuat sinyal lebih tahan dari gangguan cahaya sekitar. *Modulator* dapat berupa pewaktu/*clock* seperti IC NE 555 ataupun dengan menggunakan mikrokontroler sebagai pewaktunya untuk menghasilkan modulasi sinyal.

Demodulator berfungsi untuk memisahkan kembali sinyal data yang telah dimodulasi dengan sinyal *carrier*, agar diperoleh sinyal data kembali. Modulasi dibutuhkan agar sinyal inframerah lebih kebal terhadap cahaya dari lingkungan sekitarnya. *Demodulator* yang digunakan yaitu IC IRM-8601, yang bekerja pada frekuensi 38 khz. IC ini hanya akan menerima masukan dari sinyal yang mempunyai frekuensi 38 khz dan membuang sinyal lain yang merupakan sinyal pengganggu.



Gambar 2.13 Konfigurasi Pin IRM-8601



Gambar 3.7 Rangkaian *Multiplexer* untuk *Photodiode*

3.2.4 Rangkaian *Pushbutton*

Rangkaian *pushbutton* berfungsi sebagai masukan untuk menggantikan fungsi tombol pada *mouse*. Rangkaian ini menggunakan 3 buah *pushbuttons* untuk menggantikan *mouse* yang memiliki 3 buah tombol yaitu *left button*, *middle button* dan *right button*.

siap melakukan komunikasi dengan *device*. *Device* akan mengirimkan dua *byte* data yaitu AAh dan 00h ke PS/2 *mouse port*. AAh mengindikasikan *self-test-passed* dan 00h merupakan ID *device*. Ketika kedua data ini diterima oleh komputer dan dicek, kemudian komputer menyuruh *device* untuk *reset* dan komputer akan menunggu sampai *device* mengirimkan sinyal *acknowledge* (FAh). Setelah itu komputer akan menentukan variabel *scaling factor* (E6h), *resolution* (E8h), *set sample rate* (F3h) dan lain-lain dengan cara mengirimkan data tersebut ke *device*. Pada setiap proses pengiriman data ke *device*, komputer akan mengecek sinyal *acknowledge* yang dikirimkan oleh *device* ke komputer, menandakan data telah diterima oleh *device* dan komputer tidak akan mengulang pengiriman data itu kembali. Setelah semua pengiriman variabel telah dilakukan, komputer mengirimkan sinyal *enable* (F4h) ke *device* menandakan *device* siap digunakan.

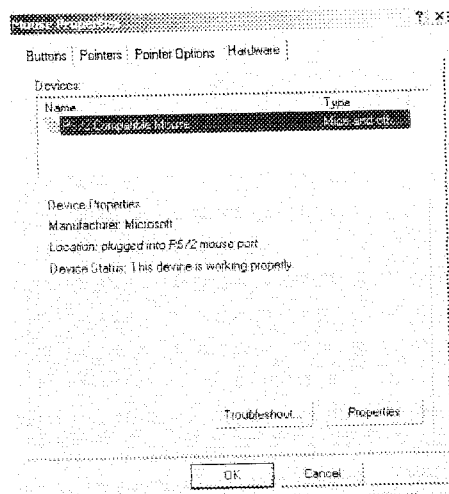
3.3.3 Program *Scanning Tombol*

Listing program scanning tombol berisi proses pembacaan tombol dari tiga buah tombol yang ada. *Program scanning pushbuttons* memiliki tiga prioritas, prioritas utama yaitu tombol *left button*, prioritas kedua yaitu tombol *right button* dan prioritas terakhir tombol *middle button*. Mikrokontroler akan mengambil prioritas yang paling tinggi dari tombol-tombol yang ditekan. Jika *left button* dan *right button* ditekan secara bersamaan, mikrokontroler hanya akan mengambil data dari *left button* karena tombol *left button* memiliki prioritas yang lebih tinggi daripada tombol *right button*.

4.2 Pengujian Sistem

4.2.1 Pengujian Proses Inisialisasi dengan Komputer

Pengujian dilakukan dengan cara menghubungkan rangkaian ke komputer melalui PS/2 *port*, setelah itu komputer dihidupkan agar *touchpad* dapat dikenali sebagai PS/2 *Compatible Mouse*. Keberhasilan dari inisialisasi *touchpad* dapat dilihat pada *mouse properties* di **Start**→**Control Panel**→**Mouse**→**Hardware**. Jika pada kolom *name* telah terisi dengan PS/2 *Compatible Mouse*, maka *touchpad* telah berhasil dikenali sebagai PS/2 *Compatible Mouse*.



Gambar 4.2 *Mouse Properties*

Komputer juga secara otomatis mencocokkan *driver* yang akan digunakan oleh PS/2 *Compatible Mouse*. *Driver* yang digunakan dapat dilihat di **Start**→**ControlPanel**→**Mouse**→**Hardware**→**Properties**→**Driver**→**DriverDetail**. Gambar 4.3 menunjukkan *driver* yang digunakan oleh *touchpad*.