

AKUISISI DATA MENGGUNAKAN *UNIVERSAL SERIAL BUS* (USB)

oleh :
Muhamad Rahmady Setyanugroho
L2F096605

ABSTRAK

Kemampuan PC telah berkembang pesat, jumlah pekerjaan yang dapat ditangani semakin banyak, dan jumlah perangkat yang ingin disambungkan dengan PC juga semakin banyak. Tetapi terdapat batasan yaitu jumlah port yang dapat disambungkan ke perangkat keras yang terlalu sedikit.

Untuk mengatasi hal tersebut, dibuatlah sebuah port masukan/keluaran baru berupa bus serial berkecepatan tinggi yang dikenal dengan nama Universal Serial Bus (USB).

I. PENDAHULUAN

Piranti dengan menggunakan USB saat ini telah banyak beredar di pasaran dan digunakan untuk berbagai macam aplikasi. Pengembangan piranti, baik masukan/keluaran, kontrol, maupun yang lainnya dengan menggunakan USB terbuka sangat luas mengingat kelebihan-kelebihan yang dimiliki USB. Namun belum banyak yang membahas dan menggunakan protokol USB dalam membuat piranti masukan/keluaran.

Tugas Akhir ini membahas sistem komunikasi menggunakan *port* USB. Untuk itu dirancang sebuah piranti antar muka masukan/keluaran sederhana untuk keperluan akuisisi data dengan spesifikasi sebagai berikut :

1. Antar muka masukan adalah logika 8 bit yang diatur menggunakan saklar.
2. Antar muka keluaran adalah tampilan 8 bit dengan LED.
3. Antar muka masukan/keluaran dihubungkan ke komputer dengan bantuan mikrokontroler.
4. Hubungan antara PC dengan mikrokontroler dilakukan melalui *port* USB.
5. Mikrokontroler yang digunakan adalah mikrokontroler AT89C51 buatan Atmel yang termasuk keluarga MCS-51.
6. Kecepatan transfer data terbatas sebesar kecepatan maksimal pengolahan data mikrokontroler.
7. Sistem operasi yang digunakan pada PC adalah Windows 98.
8. Perangkat lunak pada PC disusun menggunakan bahasa pemrograman Visual Basic 6.

II. DASAR TEORI

1.1. *Universal Serial Bus* (USB)

USB merupakan *port* masukan/keluaran baru yang dibuat untuk mengatasi kekurangan-kekurangan *port* serial maupun paralel yang sudah ada. USB dibuat dengan kelebihan-kelebihan sebagai berikut ^[1] :

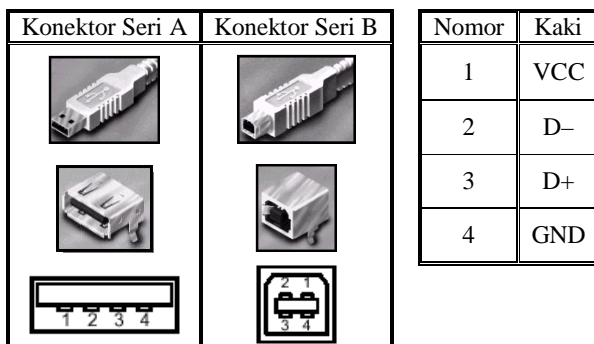
1. *Hot-plugable*, yang berarti piranti masukan/keluaran yang menggunakan USB dapat ditambahkan ketika PC menyala.

2. Mudah digunakan karena piranti masukan/keluaran yang terpasang dikenali oleh PC menggunakan *driver* yang sesuai kemudian konfigurasinya akan dikerjakan secara otomatis.
3. Semua piranti dipasang menggunakan satu tipe konektor.
4. Kecepatan USB sangat tinggi, dapat mencapai 12 Mbps (untuk USB Rev 1.1, yang digunakan pada Tugas Akhir ini) yang tentunya jauh lebih cepat dibanding *port* serial dan paralel yang ada saat ini.
5. Jumlah piranti yang dapat dipasang pada 1 PC mencapai 127 piranti (dengan bantuan hub yang dapat dipasang sampai 5 tingkat), suatu batasan yang sangat tinggi untuk ekspansi piranti masukan/keluaran.
6. Piranti dengan USB dapat menggunakan catu daya dari PC (untuk penggunaan arus tidak lebih dari 500mA) sehingga tidak membutuhkan tambahan catu daya luar.
7. Hemat listrik karena piranti dapat mati secara otomatis apabila tidak digunakan (PC dalam keadaan *suspend*).
8. Adanya deteksi dan pemulihan kesalahan yang handal. Kesalahan data dideteksi dan transaksi diulang lagi untuk memastikan data dikirim/diterima dengan benar.
9. Merupakan piranti eksternal PC sehingga tidak perlu membuka kotak PC atau merancang suatu kartu antarmuka dalam penggunaan piranti masukan/keluaran dengan USB.

Sinyal USB

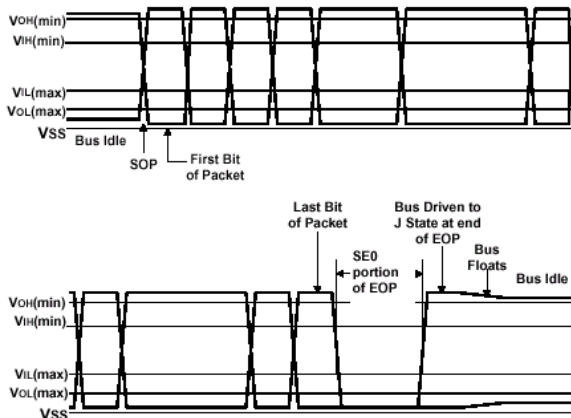
Komunikasi pada USB dilakukan secara serial. Serial lebih dipilih dibanding dengan paralel karena kebutuhan kabel yang lebih sedikit, sehingga lebih murah, dan lebih mudah diterapkan dalam konfigurasi dinamik. Yang dimaksud dengan konfigurasi dinamik adalah suatu sistem masukan/keluaran dapat dipasang atau dikonfigurasi ulang dengan memasang atau melepas kabel ketika PC bekerja. Pada konfigurasi dinamik tidak diperlukan booting ulang ^[1].

Hubungan piranti masukan/keluaran dengan PC melalui USB dilakukan secara asimetrik yang berarti konektor pada kedua ujung kabel tidak sama sehingga harus diketahui ujung mana yang dipasang pada master dan ujung mana yang dipasang pada slave. Terminologi yang diadopsi oleh spesifikasi USB adalah “*upstream*” (menuju PC) dan “*downstream*” (menuju piranti masukan/keluaran). Ujung “*upstream*” mengatur protokol dan menginstruksikan ujung “*downstream*” untuk membalas pada waktu yang ditentukan. Konektor tipe A dipasang pada PC sedangkan tipe B dipasang pada piranti masukan/keluaran. Konektor pada USB memiliki 4 kaki, yaitu VCC +5V (atau sering disebut VBUS), Data- (D-), Data+ (D+), dan GND, seperti ditunjukkan pada Gambar 1.



Gambar 1 Tipe Konektor USB dan Susunan Kakinya

Untuk mengurangi derau (*noise*), USB menggunakan sinyal diferensial [2]. Kabel data USB terhubung dari titik ke titik (*point-to-point*) dan pengiriman data dilakukan secara *half duplex*, yang berarti hanya salah satu ujung kabel yang dikemukakan pada satu waktu. Protokol mengatur kedua ujung kabel bergantian mengirim informasi. Gambar 2 menunjukkan sinyal data USB, sedangkan tabel 1 menunjukkan karakteristik tegangan sinyal USB.



Gambar 2 Sinyal Data USB

Tabel 1 Karakteristik Tegangan Sinyal USB

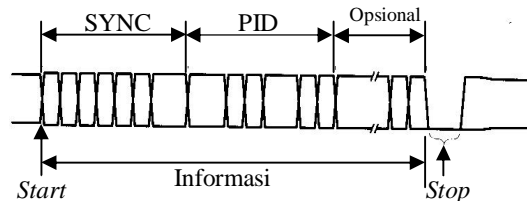
| Parameter | Simbol | Min | Max | Satuan |
|-----------|--------|-----|-----|--------|
| Input | | | | |
| Tinggi | VIH | 2,0 | - | V |
| Rendah | VIL | - | 0,8 | V |
| Output | | | | |
| Tinggi | VOH | 2,8 | 3,6 | V |
| Rendah | VOL | 0,0 | 0,3 | V |

Kabel data USB tidak memiliki sinyal CLOCK, sehingga komunikasi dengan USB dapat dikatakan sebagai asinkron.

Sebuah bus kecepatan penuh memiliki kondisi D+ tinggi dan D- rendah. Kondisi ini biasa dikatakan sebagai logika “1” dari bus, namun untuk mencegah kerancuan dan mempertahankan konsistensi dengan spesifikasi USB, kondisi ini kemudian disebut sebagai kondisi “J”. Kondisi yang lain adalah kondisi “K” yang biasa dikatakan logika “0” dari bus yaitu saat D+ rendah dan D- tinggi.

Paket Pada USB

Elemen komunikasi paling dasar pada sebuah bus data USB adalah paket. Sebuah paket memiliki tiga bagian : *start* (awal), informasi dan *stop* (akhir), seperti ditunjukkan pada Gambar 3.



Gambar 3 Paket Data USB

Terdapat 16 tipe paket USB, 10 diantaranya sudah didefinisikan dan 6 diantaranya digunakan sebagai cadangan, seperti ditunjukkan pada Tabel 2.

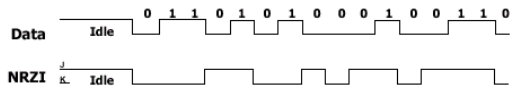
Tabel 2 Tipe Paket USB

| Nilai PID | Tipe Paket | Kategori Paket |
|-----------|------------|----------------|
| 0101 | SOF | Token |
| 1101 | SETUP | Token |
| 1001 | IN | Token |
| 0001 | OUT | Token |
| 0011 | DATA0 | Data |
| 1011 | DATA1 | Data |
| 0010 | ACK | Handshake |
| 1010 | NAK | Handshake |
| 1110 | STALL | Handshake |
| 1100 | PRE | Special |
| Lainnya | RESERVED | RESERVED |

Pada jalur data digunakan sebuah proses encode yang disebut Pembalikan Tidak Kembali Ke Nol (*Non Return to Zero Inverted – NRZI*), seperti ditunjukkan pada Gambar 4.

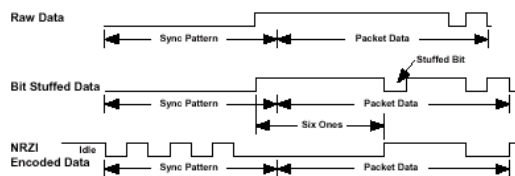
Protokol NRZI memiliki aturan sebagai berikut :

1. Ayun setiap bit untuk data 0.
2. Jangan ayun setiap bit untuk data 1.



Gambar 4 Enkode Data dengan Protokol NRZI

Apabila terdapat data 1 yang panjang, maka tidak terjadi perubahan bit pada bus sehingga penerima dapat kehilangan sinkronisasi. Oleh karena itu, pada spesifikasi USB, dikenal istilah *bit stuffing*, yaitu ditambahkan sebuah data 0 setelah datangnya 6 bit data 1 yang berurutan, seperti ditunjukkan pada Gambar 5.



Gambar 5 Algoritma Bit Stuffing

Transaksi Pada USB

Terdapat empat tipe transaksi yang berbeda untuk menangani tipe data yang berbeda yang dapat dikirimkan melalui bus. Piranti lunak pada PC host mempunyai dua parameter yang harus diperhatikan, yaitu waktu pengiriman, dan kualitas pengiriman. Tipe data yang berbeda membutuhkan parameter waktu dan kualitas yang berbeda. Tabel 3 menunjukkan transaksi USB.

Tabel 3 Tipe Transaksi USB

| Tipe | Parameter penting | Ukuran data maksimum | Contoh |
|--------------------|--------------------|------------------------------------|------------------|
| <i>Interrupt</i> | Kualitas | 64 byte (8 byte untuk kec. rendah) | Mouse, keyboard |
| <i>Bulk</i> | Kualitas | 64 byte | Printer, scanner |
| <i>Isochronous</i> | Waktu | 1024 byte | Speaker, video |
| Kontrol | Waktu dan Kualitas | 64 byte (8 byte untuk kec. rendah) | Kontrol sistem |

2.2. Mesin Antarmuka Serial (*Serial Interface Engine – SIE*)

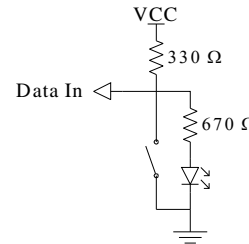
Bagian penting dari antarmuka USB adalah Mesin Antarmuka Serial (*Serial Interface Engine – SIE*). SIE menerima bit-bit dari penerima USB, melakukan validasi dan menyediakan byte yang valid ke antarmuka SIE. Byte yang diterima dari antarmuka SIE kemudian dikirimkan secara serial ke dalam bus USB. Pada Tugas Akhir ini digunakan SIE berupa IC PDIUSB12 dari Philips.

III. PERANCANGAN

3.1. Perancangan Perangkat Keras

Piranti Masukan

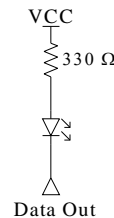
Piranti masukan adalah 8 bit data yang disusun menggunakan saklar yang dihubungkan dengan LED sebagai penanda. Rancangannya ditunjukkan pada Gambar 6.



Gambar 6 Rancangan Piranti Masukan

Piranti Keluaran

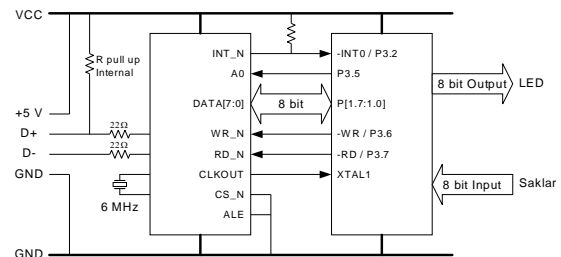
Piranti keluaran menggunakan 8 buah LED sebagai tampilan. Rancangannya ditunjukkan pada Gambar 7.



Gambar 7 Rancangan Piranti Keluaran

Piranti Komunikasi Antara PC

dengan Mikrokontroler menggunakan USB
Piranti komunikasi menggunakan USB dirancang antara PC dengan mikrokontroler melalui sebuah IC Philips PDIUSB12 sebagai Mesin Antarmuka Serial (*Serial Interface Engine – SIE*). Rancangannya ditunjukkan pada Gambar 8.



Gambar 8 Piranti Komunikasi PC dengan Mikrokontroler menggunakan USB

3.2. Perancangan Perangkat Lunak

Sebelum memulai transaksi pengiriman dan penerimaan data, PC perlu mengenali piranti terlebih dahulu, proses inisialisasi ini dikenal dengan nama proses enumerasi. Informasi mengenai piranti dan konfigurasinya disusun dalam bentuk *descriptor*.

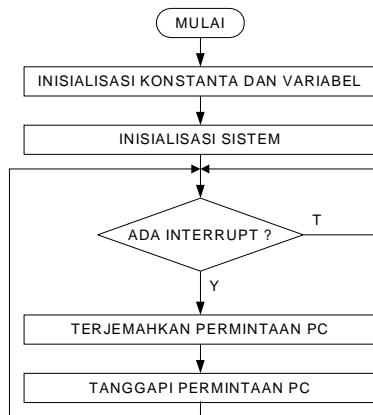
Oleh karena itu, sebelum menyusun perangkat lunak, *descriptor* yang menerangkan tentang piranti

dan konfigurasi harus ditentukan dan dilengkapi terlebih dahulu.

Setelah menyusun *descriptor*, tahap berikutnya adalah menentukan tanggapan yang harus diberikan untuk setiap permintaan yang dikirimkan PC. Dalam komunikasi USB, semua transaksi dilakukan berdasarkan permintaan PC. Piranti harus menanggapi permintaan tersebut dengan tindakan yang sesuai supaya dapat dikenali dan berkomunikasi dengan PC.

3.2.1. Perangkat Lunak Mikrokontroler

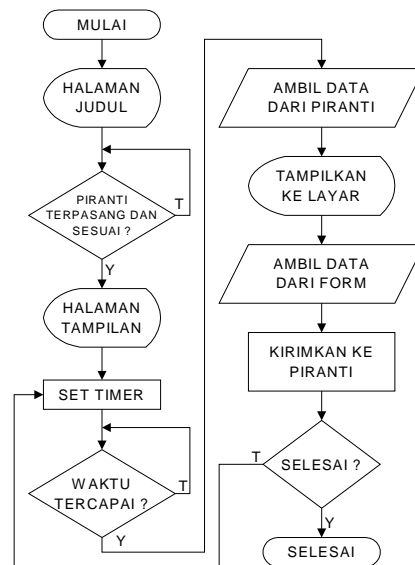
Perangkat lunak mikrokontroler disusun dengan bahasa assembly dan diisikan ke dalam Flash EPROM internal yang dimiliki mikrokontroler AT89C51. Perangkat lunak ini memuat *descriptor* piranti serta rutin-rutin yang harus dijalankan piranti dalam menanggapi permintaan PC. Diagram alirnya ditunjukkan pada Gambar 9.



Gambar 9 Diagram Alir Perangkat Lunak Mikrokontroler

3.2.2. Perangkat Lunak PC

Penyusunan perangkat lunak PC dalam Tugas Akhir ini dilakukan dengan menggunakan Visual Basic 6.0. yang dijalankan pada sistem operasi Windows 98. Perangkat lunak ini mencakup fungsi-fungsi pengenalan piranti, penulisan data ke piranti, pembacaan data piranti, dan penampilan data hasil pembacaan ke layar. Pembacaan dan penulisan data piranti diaktifkan dengan pewaktu (*timer*). Proses pengenalan piranti, pengiriman data ke piranti dan pembacaan data piranti dilakukan dengan menggunakan fungsi-fungsi Windows *Application Programming Interface* (API). Windows API merupakan fungsi-fungsi eksternal yang terdapat dalam pustaka (*library*) Windows maupun *library* lainnya. Diagram alirnya ditunjukkan pada Gambar 10.



Gambar 10 Diagram Alir Perangkat Lunak PC

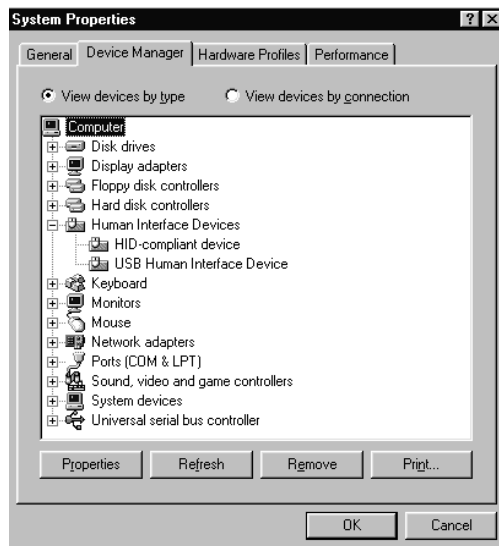
Program diawali dengan menampilkan Halaman Judul. Setelah itu dilakukan pengenalan piranti. Pengenalan piranti dilakukan dengan mencari apakah terdapat piranti yang terpasang. Hal ini dilakukan dengan memeriksa simpul-simpul *Plug 'n Play* (PnP) pada PC. Apabila terdapat piranti yang terpasang, program akan mengambil nama piranti tersebut kemudian membuka hubungan dengan menggunakan fungsi *CreateFile* dari *library* kernel32.dll. Apabila piranti terpasang dan memiliki nama yang sesuai, program akan menampilkan Halaman Tampilan, namun bila piranti tidak terpasang atau nama piranti tidak sesuai, program akan menampilkan pesan kesalahan.

Pada Halaman Tampilan, semua proses pengiriman dan penerimaan data yang diaktifkan oleh *timer* dilakukan. Pengiriman data dilakukan dengan menggunakan fungsi *WriteFile* dari *library* kernel32.dll.

IV. ANALISA DAN PEMBAHASAN

Pengujian dilakukan untuk memperlihatkan hasil dan fungsi kerja perangkat keras dan perangkat lunak dari sistem yang sudah dibuat. Pengujian meliputi keberhasilan pengenalan piranti, pengiriman data dari PC ke piranti masukan/keluaran dan penerimaan data dari piranti masukan/keluaran oleh PC.

Pengujian dilakukan dengan memasang piranti pada PC. Pada saat dipasang, piranti akan dikenali sebagai *HID Compliant Device* dan *USB Human Interface Device*. Gambar 11 menunjukkan tampilan jendela *System Properties > Device Manager* dari Windows 98 saat piranti dipasangkan ke PC.



Gambar 11 Tampilan Jendela System Properties > Device Manager

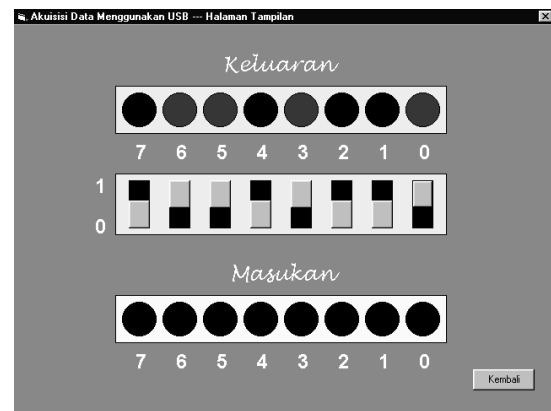
Langkah berikutnya adalah menguji keberhasilan pengiriman dan penerimaan data. Pengujian dilakukan dengan menjalankan perangkat lunak PC yang telah disusun. Perangkat lunak dijalankan dengan membuka *file* bernama TugasAkhir.exe.

Setelah program tereksekusi, akan muncul tampilan awal berupa Halaman Judul yang berisi judul Tugas Akhir, tombol Masuk untuk masuk ke prosedur berikutnya, dan tombol Selesai untuk mengakhiri dan keluar dari program. Tampilan Halaman Judul ditunjukkan pada Gambar 12.



Gambar 12 Tampilan Halaman Judul

Apabila tombol Masuk ditekan, program akan menampilkan Halaman Tampilan. Pada halaman ini seluruh operasi pengiriman dan penerimaan data dilakukan. Tampilan Halaman Tampilan ditunjukkan pada Gambar 13.



Gambar 13 Tampilan Halaman Tampilan

Pengujian pengiriman data dilakukan dengan mengubah-ubah posisi tombol saklar pada Halaman Tampilan ke posisi 0 atau 1. Perubahan ini dilakukan dengan menekan tombol saklar. Hasil pengiriman data dilihat dari tampilan LED pada piranti keluaran. Dari hasil pengujian didapatkan kondisi logika yang diatur pada Halaman Tampilan ditampilkan sesuai oleh LED pada piranti keluaran. Hal ini berarti proses pengiriman data berjalan dengan baik. Sedangkan pengujian penerimaan data dilakukan dengan mengubah-ubah posisi saklar pada piranti ke posisi ON atau OFF. Hasil penerimaan data dilihat dari tampilan pada Halaman Tampilan. Dari hasil pengujian didapatkan kondisi logika yang diatur pada saklar ditampilkan sesuai pada Halaman Tampilan. Hal ini berarti proses penerimaan data berjalan dengan baik.

V. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari Tugas Akhir yang telah dibuat, dapat ditarik beberapa kesimpulan sebagai berikut :

1. Perancangan perangkat keras piranti yang menggunakan USB, terutama untuk piranti masukan/keluaran sederhana, lebih mudah dibanding menggunakan *port* serial atau paralel karena dapat menggunakan satu dari *bus* sehingga tidak perlu merancang satu tersendiri.
2. Protokol USB yang rumit membuat perancangan perangkat lunak, terutama perangkat lunak pada mikrokontroler menjadi lebih sulit dibanding jika menggunakan protokol *port* serial dan paralel.
3. Perancangan perangkat lunak PC lebih mudah dibanding menggunakan *port* serial atau paralel karena apabila perangkat lunak mikrokontroler sudah benar, saat dipasangkan pada PC, piranti akan langsung dikenali oleh PC tanpa perlu menggunakan driver tersendiri.

5.2. Saran

1. Tugas Akhir ini hanya menggunakan piranti masukan/keluaran yang sederhana. Untuk berikutnya dapat dikembangkan piranti yang lebih kompleks, untuk keperluan akuisisi data maupun kendali.
2. Pada Tugas Akhir ini digunakan spesifikasi USB Revisi 1.1 yang hanya mendukung kecepatan transfer sampai dengan 12 Mbps. Untuk pengembangan lebih lanjut, dapat dipelajari penggunaan spesifikasi USB Revisi 2.0 yang mendukung kecepatan transfer sampai dengan 480 Mbps yang tentunya akan jauh lebih handal.

DAFTAR PUSTAKA

1. Hyde, John, *USB design by example : a practical guide to building I/O devices*, John Wiley & Sons, Inc., Toronto, 1999.
2. Compaq Computer Corporation, Hewlett-Packard Company, Intel Corporation, Lucent Technologies Inc, Microsoft Corporation, NEC Corporation, Koninklijke Philips Electronics N.V., , *Universal Serial Bus specification revision 2.0*, Compaq Computer Corporation, Hewlett-Packard Company, Intel Corporation, Lucent Technologies Inc, Microsoft Corporation, NEC Corporation, Koninklijke Philips Electronics N.V., 2000.
3. Peacock, Craig, *Using the PDIUSB11*, 2000.
4. Philips Semiconductors, *PDIUSB12 USB interface device with parallel bus datasheet*, Philips Semiconductors, 1999.
5. Hadi, Rahadian, Pemrograman Windows API dengan Microsoft Visual Basic, Elex Media Komputindo, Jakarta, 2001.
6. ATMEL Corporation, *AT89C51 in circuit programming*, ATMEL Corporation, 1997.
7. ATMEL Corporation, *AT89 series hardware description*, ATMEL Corporation, 1997.
8. ATMEL Corporation, *Designing boards with Atmel AT89C51, AT89C52, AT89C1051, and AT89C2051 for writing flash at in-circuit test*, ATMEL Corporation, 1997.
9. ATMEL Corporation, *Flash microcontroller architectural overview*, ATMEL Corporation, 1997.
10. ATMEL Corporation, *Flash microcontroller memory organization*, ATMEL Corporation, 1997.
11. ATMEL Corporation, *Microcontroller instruction test*, ATMEL Corporation, 1997.
12. ATMEL Corporation, *Using a personal computer to program the AT89C51 / C52 / LV51 / LV52 / C1051 / C2051*, ATMEL Corporation, 1997.
13. ATMEL Corporation, *AT89C51 datasheets*, ATMEL Corporation, 2000.
14. Elektuur, *Data sheet book 1 : data IC linier, TTL dan CMOS*, kumpulan data penting komponen elektronika (diterjemahkan oleh Wasito Suyono), Elex Media Komputindo, Jakarta, 1997.
15. Hayt, William H., Kemmerly, Jack E., Rangkaian listrik jilid 1 (diterjemahkan oleh Pantur Silaban), Edisi ke-4, Penerbit Erlangga, Jakarta, 1991.
16. Malvino, Albert P., Leach, Donald P., Prinsip-prinsip dan penerapan digital (diterjemahkan oleh Irwan Wijaya), Edisi ke-3, Penerbit Erlangga, Jakarta, 1987.
17. Smith, Eric A., Whisler, Valor, Marquis, Hank, Visual Basic 6 bible, IDG Books Worldwide, Inc., Foster City, 1998.
18. Tocci, Ronald J., *Digital system : principle and application, 5th ed.*, Prentice Hall Inc., New Jersey, 1997.
19. Tokheim, Roger L., Elektronika digital (diterjemahkan oleh Sutisno), Edisi ke-2, Jakarta : Penerbit Erlangga, 1995.



* Muhamad Rahmady Setyanugroho, cST, kelahiran Semarang, 2 Mei 1979, saat ini masih “aktif” sebagai mahasiswa Jurusan Teknik Elektro Fakultas Teknik Universitas Diponegoro Semarang.

Mengetahui,
Pembimbing II

Agung Budi Prasetijo, ST, MIT
NIP. 132 137 932