

# Perancangan dan Pembuatan Adapter Penghubung Keyboard IBM PS/2 dengan Port USB Personal Computer

Resmana Lim<sup>1</sup>, Danny Christanto<sup>1</sup> & Anies Hannawati<sup>1,2</sup>

<sup>1</sup>Fakultas Teknologi Industri, Jurusan Teknik Elektro, Universitas Kristen Petra  
e-mail: resmana@petra.ac.id ; dctan1980@yahoo.com

<sup>2</sup>Electrical Engineering Department, Monash University, Melbourne – Australia  
e-mail: Anies.Purnamadajaja@eng.monash.edu.au

## Abstrak

*Universal Serial Bus (USB) sebagai suatu interface pada Personal Computer (PC) saat ini banyak digunakan untuk menghubungkan berbagai peralatan dengan PC seperti printer, scanner, mouse, webcam dan lain-lain. Dalam tulisan ini dibuat sebuah adapter untuk menghubungkan keyboard IBM PS/2 agar dapat dikenali dan dijalankan dalam lingkungan USB yang terdapat pada PC. Alat ini menggunakan sebuah IC interface USB yaitu PDIUSB12 yang akan menangani protokol USB dan sebuah microcontroller yang akan menangani semua data yang keluar ataupun masuk dari dan ke PDIUSB12 ataupun keyboard. Alat ini telah diuji dan terbukti mampu menangani proses enumerasi dan menyampaikan data yang diterima dari keyboard menuju PC via port USB. Meskipun alat ini tidak mendukung semua fungsi tombol pada keyboard, tetapi fungsi tombol-tombol utama bisa berjalan dengan normal pada penekanan satu tombol ataupun kombinasi beberapa tombol.*

**Kata kunci :** *Universal Serial Bus, Adapter Keyboard PS/2 ke USB, PDIUSB12.*

## Abstract

*Universal Serial Bus (USB) as a new interface especially on Personal Computer (PC) has become a standard for interfacing many devices to PC such as printer, scanner, mouse, webcam, etc. In this paper, a device that can be recognized by USB host and communicate through USB port correctly was developed. It has been implemented by making an adapter therefore IBM PS/2 keyboard could be recognized and operated in a USB environment. The device used a PDIUSB12 USB interface IC which handle USB protocol and an AT89C2051 microcontroller which handle all incoming or outgoing data from and to PDIUSB12 or keyboard. The device has been tested and proven capable of handling enumeration process and transfer data from keyboard to PC via a USB port. Although the device didn't support all functions of the keys, but the main keys worked normally on pressing one key or even combination keys.*

**Keywords :** *Universal Serial Bus, PS/2 Keyboard Adapter to USB port, PDIUSB12.*

## 1. Pendahuluan

*Universal Serial Bus (USB) sebagai suatu interface yang relatif baru terutama pada Personal Computer (PC) memiliki banyak keunggulan dibandingkan interface pendahulunya. Keunggulan USB antara lain dalam hal kecepatan dan kemudahan penggunaannya [9,11]. USB bersifat host-centric dimana semua transaksi dimulai oleh host. Dengan kemampuan hot-plug, yang memperbolehkan sebuah alat untuk dihubungkan atau dilepas dari PC kapan saja walaupun PC dalam keadaan menyala, membuat protokol komunikasi USB menjadi lebih rumit daripada protokol interface lain.*

Komunikasi antara *host* (komputer) dan *function* (alat yang dihubungkan) meliputi proses enumerasi yang mengandung serangkaian *request* dan *descriptor*. *Request* dikirim oleh *host* kepada *function* sebagai perintah atau permintaan data balasan. *Descriptor* dikirim oleh *function* kepada *host* sebagai data balasan yang diminta, yang berisi identitas, kemampuan, dan kebutuhan *function* [10].

Dalam paper ini, dibangun sebuah adapter untuk keyboard IBM PS/2 agar keyboard tersebut bisa dikenali oleh komputer (sebagai keyboard USB) dan beroperasi dengan interface USB (dikenali sebagai *Human Interface Device*). Sebuah microcontroller Atmel AT89C2051 [1] digunakan untuk mengartikan *request* yang diterima, menentukan *descriptor*, dan mengubah *scancode* AT keyboard IBM PS/2 menjadi *Human*

---

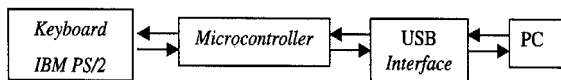
**Catatan:** Diskusi untuk makalah ini diterima sebelum tanggal 1 Juni 2004. Diskusi yang layak muat akan diterbitkan pada Jurnal Teknik Elektro volume 4, nomor 2, September 2004.

Interface Device (HID) Usage ID. *Microcontroller* tersebut berpasangan dengan USB *Interface Device* PDIUSB12 [5,8] yang menangani protokol pengiriman dan penerimaan data meliputi perubahan dalam kode *Non-Return to Zero Invert (NRZI)*, *bit stuffing*, proses *Cyclic Redundancy Check (CRC)*, *Packet Identifier (PID)*, dan *handshake*.

Selanjutnya paper ini membahas tentang desain sistem secara keseluruhan baik dari sisi perangkat keras maupun perangkat lunaknya, lantas dilanjutkan dengan pengujian sistem serta diakhir dengan diskusi.

## 2. Deskripsi Sistem

Sistem yang dibuat terdiri dari sebuah mikrokontroler yang menghubungkan keyboard IBM PS/2 dengan port USB pada PC. Gambar 1 menunjukkan blok diagram *adapter* yang telah dibuat.



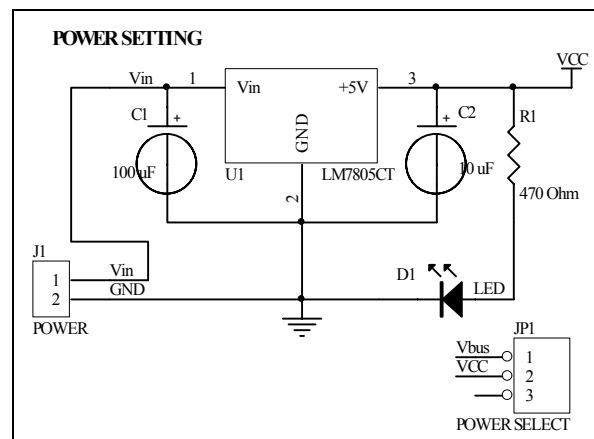
Gambar 1. Blok Diagram *Adapter*

Pada proses enumerasi, *request* dari komputer (PC) diterima USB *Interface* PDIUSB12 yang diteruskan ke *microcontroller* AT89C2051 untuk diartikan. Setelah mengartikan dan menentukan data balasnya, data tersebut dikirim ke PC melalui USB *Interface*.

*Keyboard* baru bisa digunakan setelah proses enumerasi berjalan lengkap dan tepat (sistem dikenali PC sebagai *keyboard* USB). Data *scancode* dari *keyboard* dibaca oleh *microcontroller* dan diubah ke *HID Usage ID* [10,11,12]. Data *HID* tersebut dikirimkan ke PC melalui USB *Interface*. Data dari PC (status LED) akan dikirimkan ke *microcontroller* melalui USB *Interface*. *Microcontroller* akan bertindak sebagai *host* bagi *keyboard* dan mengirimkan perintah untuk menyalakan atau mematikan LED.

## 3. Perangkat Keras

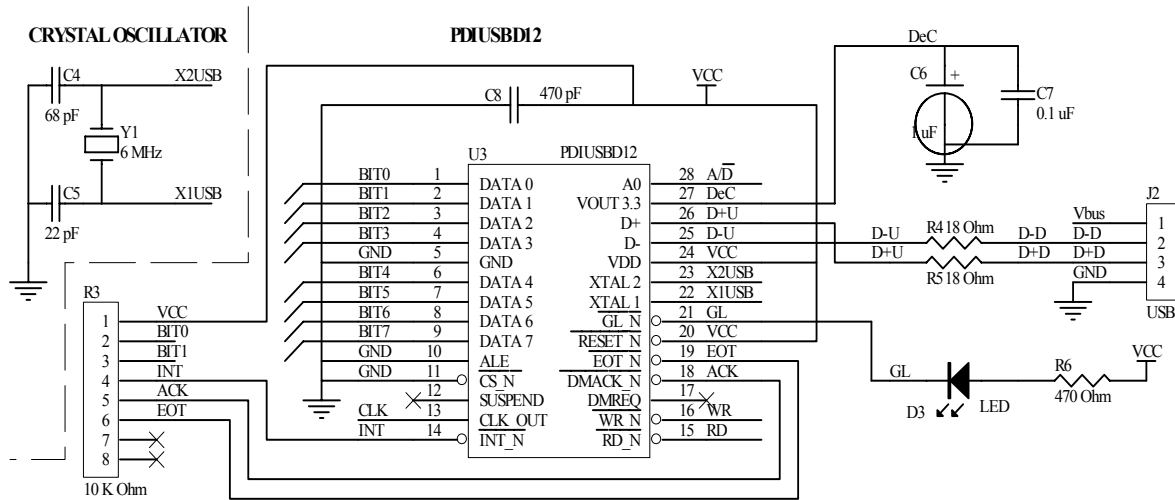
Meskipun USB memberikan kemudahan untuk mengambil arus dari *host* hingga 500 mA, tetapi *adapter* yang dibuat harus masih bisa mendapat tegangan dari sumber luar sehingga dirancanglah rangkaian seperti pada gambar 2 yang menggunakan LM7805 untuk mendapatkan tegangan 5 V.



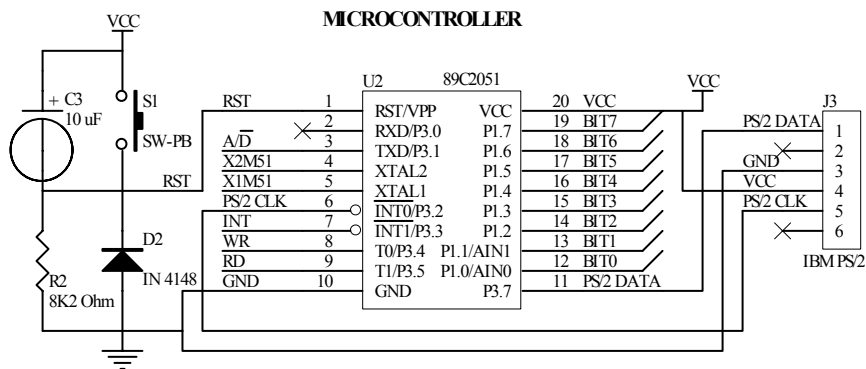
Gambar 2. Rangkaian Power Supply

Gambar 3 menunjukkan rangkaian *clock* dan PDIUSB12. Nilai komponen didapat dari rangkaian D12SMART *evaluation board* Philips [4]. PDIUSB12 dan AT89C2051 dihubungkan oleh 8 jalur data (DATA0 – DATA7) sebagai jalur transaksi data; 3 jalur *strobe* (A0, WR\_N, dan RD\_N) sebagai penanda proses pengiriman/pembacaan data/perintah; 1 jalur *interrupt* (INT\_N) sebagai penanda adanya *interrupt* pada PDIUSB12; dan 1 jalur *clock* (CLK\_OUT) sebagai masukan *clock* bagi AT89C2051. PDIUSB12 tidak memerlukan resistor *pull-up* eksternal sebagai penanda kecepatan karena telah memiliki *SoftConnect*, resistor terintegrasi yang bisa dinyalakan melalui program.

Gambar 4 menunjukkan rangkaian sistem mikrokontroler menggunakan AT89C2051. Antara AT89C2051 dan *keyboard* dihubungkan oleh 1 jalur data (sebagai jalur transaksi data) dan 1 jalur *clock* (sebagai penanda data *valid* pada saat pengiriman atau penerimaan data).



Gambar 3. Rangkaian PDIUSB12



Gambar 4. Rangkaian AT89C2051

#### 4. Perangkat Lunak

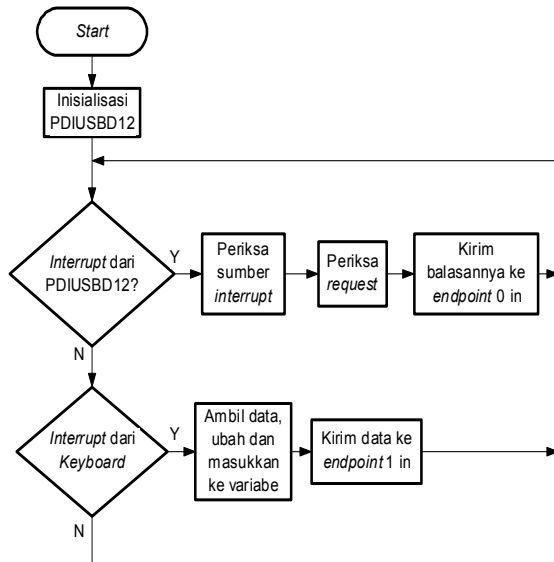
Program yang dibuat ditulis dengan menggunakan bahasa assembly untuk keluarga MCS-51. Program dibuat agar *microcontroller* bisa menangani proses inialisasi PDIUSB12 dan *interrupt* dari PDIUSB12 maupun *keyboard*. Gambar 5 menunjukkan *flowchart* dari program yang dibuat.

Proses pertama yang dilakukan adalah inialisasi semua variabel termasuk inialisasi PDIUSB12. Inialisasi PDIUSB12 digunakan antara lain untuk menentukan frekuensi keluaran CLKOUT dan menyalakan Soft-Connect.

Pada langkah berikutnya, *microcontroller* akan menunggu datangnya *interrupt*. Pada proses enumerasi, *interrupt* datang dari PDIUSB12. Lalu *microcontroller* memeriksa sumber *interrupt* (apakah dari *suspend change*, *bus reset*,

atau *endpoint*). Jika berasal dari *endpoint 0 out* (*host* mengirim data), maka *request* diperiksa untuk menentukan balasan yang sesuai. Data tersebut dikirim melalui *endpoint 0 in* agar bisa diambil *host* pada saat *polling* berikutnya (semua transaksi dimulai oleh *host*).

Pada penekanan *keyboard*, *interrupt* yang muncul digunakan oleh *microcontroller* untuk menerima data. Jika data yang diterima lengkap (*scancode* untuk penekanan atau pelepasan satu tombol), data tersebut diubah ke HID Usage ID dan dimasukkan ke variabel. Variabel tersebut dipindah ke *buffer endpoint 1 in* agar bisa diambil *host* pada saat *polling* berikutnya. Protokol keyboard PS/2 secara lengkap dapat diakses pada referensi [2,3,7], sedangkan protokol USB secara lengkap disajikan pada referensi [9,10,11].



Gambar 5. Flowchart Program

#### 4. Pengujian Sistem

Sistem yang telah dibuat diuji dengan menggunakan keyboard PS/2 New Touch Logitech. Pengujian yang dilakukan terdiri dari: pengujian inisialisasi chip PDIUSB12, pengujian transaksi enumerasi, pengujian fungsi keyboard dan pengukuran arus dan tegangan adapter.

##### 4.1 Pengujian Inisialisasi PDIUSB12

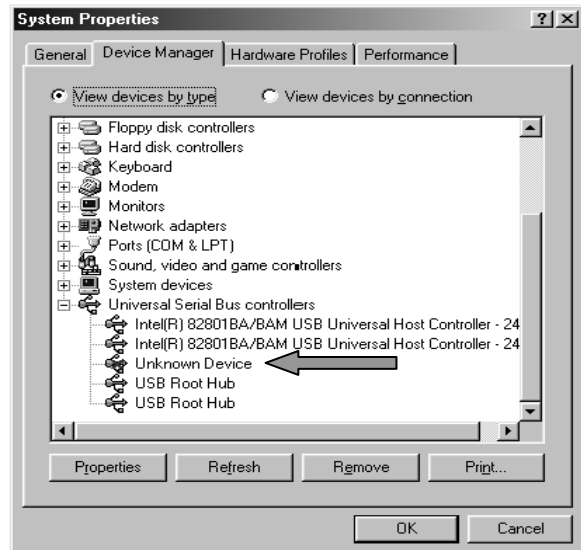
Setelah adapter dihubungkan ke PC dan inisialisasi PDIUSB12 selesai dengan lengkap dan tepat (untuk menyalakan SoftConnect), PC mendeteksi bahwa ada alat yang dihubungkan ke port USB. PC segera mengirimkan request untuk memulai proses enumerasi yang muncul pada adapter sebagai interrupt disertai nyala LED GoodLink berkedip. Karena pada tahap ini tidak ada request yang didukung maka proses enumerasi tidak bisa dilaksanakan dan System Properties Microsoft Windows 98 Second Edition tampak seperti pada gambar 6.

##### 4.2 Pengujian Transaksi Enumerasi

Pada tahap ini semua request untuk enumerasi bisa dilayani dan balasanya sudah ditentukan. PC akan melakukan proses enumerasi yang muncul pada adapter sebagai serangkaian interrupt yang disertai nyala LED GoodLink berkedip-kedip.

Di tengah proses enumerasi terdapat proses dimana sistem operasi meminta dan memasang

file driver yang dibutuhkan. Pada layar monitor muncul tampilan seperti pada gambar 7.

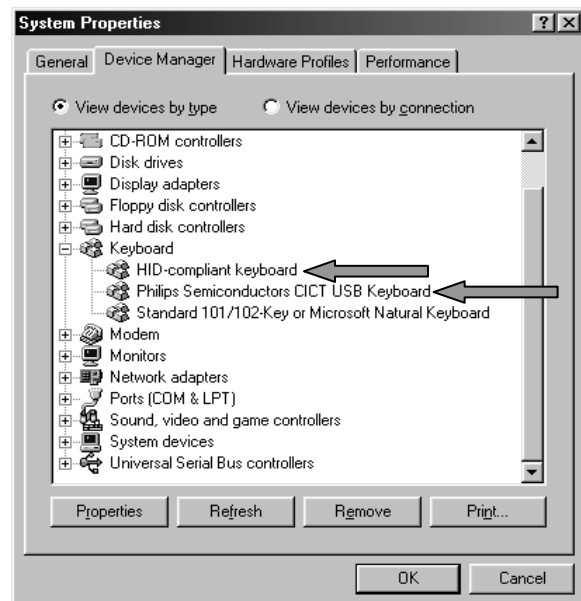


Gambar 6. Tampilan System Properties untuk Inisialisasi yang Berhasil



Gambar 7. Sistem Operasi Meminta dan Memasang File Driver

Setelah semua request dibalas, LED GoodLink menyala terus. System Properties akan tampak seperti pada gambar 8.



Gambar 8. Tampilan System Properties Setelah Enumerasi Berhasil

Urutan *request* yang diminta *host* dapat dilihat seperti terdapat pada tabel 1.

Tabel 1. Daftar *Request*

Urutan	bm Request Type	b Request	w Value low	w Value high	w Index low	w Index high	w Length low	w Length high	Arti
1	80H	6H	0H	1H	0H	0H	40H	0H	Device Desc.
2	0H	5H	2H	0H	0H	0H	0H	0H	Set Address
3	80H	6H	0H	1H	0H	0H	12H	0H	Device Desc.
4	80H	6H	0H	2H	0H	0H	9H	0H	Config. Desc.
5	80H	6H	0H	2H	0H	0H	FFH	0H	Config. Desc.
6	80H	6H	0H	1H	0H	0H	12H	0H	Device Desc.
7	80H	6H	0H	2H	0H	0H	F1H	3H	Config. Desc.
8	80H	6H	0H	2H	0H	0H	22H	0H	Config. Desc.
9	0H	9H	1H	0H	0H	0H	0H	0H	Set Config
10	21H	AH	0H	0H	0H	0H	0H	0H	Set Idle
11	81H	6H	0H	22H	0H	0H	7FH	0H	Get Class Descriptor
12	21H	9H	0H	2H	0H	0H	1H	0H	Set Report
13	21H	9H	0H	2H	0H	0H	1H	0H	Set Report
14	21H	9H	0H	2H	0H	0H	1H	0H	Set Report
15	21H	9H	0H	2H	0H	0H	1H	0H	Set Report

### 4.3 Pengujian Fungsi Keyboard

Setelah proses enumerasi selesai dan *adapter* dikenali oleh sistem operasi sebagai *keyboard* USB, selanjutnya *keyboard* sudah bisa digunakan untuk mengetik. Tabel 2 menunjukkan hasil pengujian dengan cara menekan satu per satu tombol dan melihat reaksi *keyboard* pada program aplikasi. Dalam pengujian ini digunakan beberapa program aplikasi yaitu *Microsoft Word*, *Microsoft Excel*, *Notepad*, dan *MCS51 Assembler*. Di sini terlihat bahwa *keyboard* berfungsi dengan baik yaitu memberikan reaksi sesuai dengan yang diharapkan.

Tabel 2. Hasil Uji Penekanan Satu Tombol

Nama Tombol	Reaksi pada Program Aplikasi
F1	Help pada banyak aplikasi
F2	Mengganti sebuah sel pada <i>Excel</i>
F3	Search pada <i>Notepad</i>
F4	Mengulangi tindakan terakhir pada <i>Word</i>
F5	Memunculkan jam dan tanggal pada <i>Notepad</i>
F6	Berpindah antara <i>split window</i> pada <i>Word</i>
F7	Pemeriksaan kata pada <i>Word</i>
F8	Mengaktifkan <i>extending selection</i> pada <i>Excel</i>
F9	Compile pada <i>MCS51 Assembler</i>
F10	Mengaktifkan menu pada <i>Excel</i> dan <i>Word</i>
F11	Menciptakan <i>chart</i> pada <i>Excel</i>
F12	Menampilkan <i>Save As</i> pada <i>Word</i>
Print Screen	Menyimpan gambar yang muncul di layar pada <i>Clipboard</i>
Scroll Lock	Menyalakan/mematikan LED <i>Scroll Lock</i>
Tab	Memberi jarak tabulasi pada <i>Word</i>
Caps Lock	Menyalakan/mematikan LED <i>Caps Lock</i> dan mencetak huruf kapital
Num Lock	Menyalakan/mematikan LED <i>Num Lock</i> dan mengunci tombol <i>keypad</i> agar memunculkan angka atau fungsi lain pada <i>keypad</i>
GUI (berlogo Windows)	Menampilkan menu <i>Start</i> pada <i>Windows</i>
Esc	Menghilangkan (non-aktif) menu
Alt	Mengaktifkan menu pada banyak aplikasi
Insert	Menyalakan/mematikan fasilitas <i>insert/overwrite</i> pada <i>Word</i>

Nama Tombol	Reaksi pada Program Aplikasi
Delete	Menghapus karakter pada aplikasi pengolah kata
Home	Memindah kursor pada posisi awal baris pada <i>Word</i>
End	Memindah kursor pada posisi akhir baris pada <i>Word</i>
Page Up	Memindah kursor beberapa baris ke atas pada <i>Word</i>
Page Down	Memindah kursor beberapa baris ke bawah pada <i>Word</i>
Tombol Panah	Memindah kursor satu baris ke atas/bawah atau satu karakter ke kiri/kanan pada <i>Word</i>
Backspace	Menghapus satu karakter di depan kursor pada <i>Word</i>
Enter	Memindah kursor ke baris baru di bawah posisi sebelumnya
Tombol huruf (qwerty)	Memunculkan huruf yang bersangkutan pada program pengolah kata
Tombol angka (12345)	Memunculkan angka yang bersangkutan pada program pengolah kata
Tombol karakter (space,./)	Memunculkan karakter yang bersangkutan pada program pengolah kata
Tombol aplikasi (sebelah tombol ctrl kanan)	Menampilkan menu pada program

Uji kombinasi tombol dilakukan dengan menekan beberapa kombinasi tombol yang umum digunakan pada sistem operasi *Microsoft Windows 98 Second Edition* dan beberapa program seperti pada pengujian satu tombol. Hasil uji terdapat pada tabel 3.

Tabel 3. Hasil Uji Penekanan Kombinasi Tombol

Kombinasi Tombol	Reaksi pada Program Aplikasi
Ctrl + Huruf	Melakukan perintah ( <i>Save, Copy, Paste, Cut, dll</i> ) pada aplikasi yang mendukung
Ctrl + Panah	Memindah kursor satu kata ke depan/belakang atau satu paragraf ke atas/bawah
Shift + Huruf	Menampilkan huruf kapital
Shift + Karakter (;'./)	Menampilkan karakter lain pada tombol
Shift + Panah	Memilih (blok) satu huruf ke depan/belakang atau satu baris ke atas/bawah pada <i>Word</i>
Alt + Tab	Berpindah <i>window</i> pada <i>Windows</i>
Alt + Huruf	Mengaktifkan menu atau tombol dengan huruf yang bergaris bawah
Alt + Space	Mengaktifkan menu <i>window</i>
Alt + F4	Mematikan program pada <i>Windows</i>
GUI + Huruf	Mengaktifkan aplikasi <i>Windows (Windows Explorer, Find, Run)</i>
Ctrl + Shift + Esc	Menampilkan menu <i>Start</i> pada <i>Windows</i>
Ctrl + Shift + Panah	Memilih (blok) satu kata ke depan/belakang atau satu paragraf ke atas/bawah pada <i>Word</i>
Ctrl + Alt + Del	Menampilkan menu <i>Close Program</i> pada <i>Windows</i>

Hasil uji kecepatan pada tabel 4 menunjukkan kemampuan *adapter* untuk menangani transaksi data yang relatif cepat untuk penggunaan sehari-hari tanpa mengacaukan sistem meskipun harus menangani dua *interrupt*. Pengujian ini dilakukan dengan cara pemberian input keyboard

dengan variasi kecepatan pengetikan. Di sini terlihat bahwa sistem mampu menangani kecepatan yg bervariasi sampai dengan maksimal 27 karakter/detik.

Tabel 4. Hasil Uji Variasi Kecepatan Pengetikan

Pengujian	Kecepatan Pengetikan (Karakter Per Detik)	Keterangan
1	12	Berhasil
2	21	Berhasil
3	27	Berhasil
4	30	gagal

#### 4.4 Pengukuran Arus dan Tegangan

Tabel 5, 6, dan 7 menunjukkan hasil pengukuran arus dan tegangan pada *adapter*. Hal ini dilakukan utamanya untuk memeriksa kebutuhan arus *adapter* yang diminta dari *host* dan memeriksa agar tegangan tetap berada dalam kisaran 5 V.

Tabel 5. Hasil Pengukuran Arus pada Vbus

Kondisi	Percobaan 1 (mA)	Percobaan 2 (mA)
Belum Terenumerasi (Booting)	35,9	36
Sesaat Setelah Terenumerasi dan sesaat setelah mendapat output report	41,9	42
Mengirimkan data (Tombol ditekan)	38	38,1
Tidak mengirimkan data (tombol dilepas)	41,4	41,5
LED Caps Lock menyala	53,3/52,9	53,4/53
LED Num Lock menyala	53,4/53	53,5/53,1
LED Scroll Lock menyala	53,8/53,4	53,9/53,5
LED Num Lock dan Scroll Lock menyala	64,6/64,2	64,8/64,4
LED Num Lock dan Caps Lock menyala	64,2/63,8	64,4/64
LED Caps Lock dan Scroll Lock menyala	64,6/64,2	64,8/64,4
Semua LED menyala	74,9/74,5	75/74,8

Tabel 6. Hasil Pengukuran Arus pada VCC Keyboard

Kondisi	Arus (mA)
Normal	1
LED Caps Lock menyala	13,4
LED Num Lock menyala	13,5
LED Scroll Lock menyala	14
LED Num Lock dan Scroll Lock menyala	25,7
LED Num Lock dan Caps Lock menyala	25,3
LED Caps Lock dan Scroll Lock menyala	25,7
Semua LED menyala	36,8

Tabel 7. Hasil Pengukuran Tegangan pada Vbus

Kondisi	VCC (volt)
Normal	5,02
1 LED menyala	5,01
2 LED menyala	5
Semua LED menyala	4,98

## 5. Kesimpulan

Dari hasil pengujian yang telah dilakukan, maka dapat diambil kesimpulan bahwa:

- *Adapter* mampu berkomunikasi dengan PC secara lancar melalui *port* USB termasuk pada saat proses enumerasi dari *host*. *Keyboard* IBM PS/2 yang dihubungkan ke *adapter* mampu beroperasi layaknya *keyboard* biasa baik dalam penekanan satu tombol maupun penekanan kombinasi tombol.
- Alat mampu dikenali oleh sistem operasi *Microsoft Windows 98 Second Edition* sebagai Philips CICT USB *Keyboard*, dengan demikian fungsi utama *keyboard* bisa berjalan.
- Dari semua tombol yang ada pada *keyboard* PS/2 *New Touch* Logitech, hanya tiga tombol *power* (*Sleep*, *Wake Up*, dan *Power*) yang terletak di sisi kanan atas yang tidak didukung oleh *adapter* ini.

## Daftar Pustaka

- [1]. "Atmel AT89C2051 8-bit Microcontroller with 2K Bytes Flash". Rev. 0368E. Februari 2000. 15 hal. [Atmel Corporation](http://www.atmel.com/atmel/acrobat/doc0368.pdf). 10 Desember 2001. <<http://www.atmel.com/atmel/acrobat/doc0368.pdf>>
- [2]. Chapweske, Adam. "PS/2 Mouse/Keyboard Protocol". 25 Juli 2002. <<http://panda.cs.ndsu.nodak.edu/~achapwes/PICmicro/PS2/ps2.htm>>
- [3]. Chapweske, Adam. "The AT-PS/2 Keyboard Interfacing". 25 Juli 2002. <<http://panda.cs.ndsu.nodak.edu/~achapwes/PICmicro/keyboard/atkeyboard.html>>
- [4]. "D12SMART Evaluation Board". Philips Semiconductors. 10 Agustus 2002. <[http://www.semiconductors.philips.com/files/bus/usb/eval\\_kits/d12.smart/schematics.zip](http://www.semiconductors.philips.com/files/bus/usb/eval_kits/d12.smart/schematics.zip)>
- [5]. "FAQ – PDIUSB12". 1 Oktober 1998. 9 hal. Philips Semiconductors. 12 Juni 2002. <[http://www.semiconductors.philips.com/acrobat/various/FAQ\\_PDIUSB12.pdf](http://www.semiconductors.philips.com/acrobat/various/FAQ_PDIUSB12.pdf)>

- [6]. "Firmware Programming Guide for PDIUSB12" Version 1.0. 23 September 1998. 22 hal. Philips Semiconductors. 12 Juni 2002. <[http://www.semiconductors.philips.com/acrobat/various/PDIUSB12\\_PROGRAMMING\\_GUIDE.pdf](http://www.semiconductors.philips.com/acrobat/various/PDIUSB12_PROGRAMMING_GUIDE.pdf)>
- [7]. "Keyboard Scan Code Specification". Windows Platform Design Notes. Revision 1.3a. 16 Maret 2000. 35 hal. Microsoft Corporation. 19 Juni 2002. <<http://download.microsoft.com/download/whistler/hwdev3/1.0/WXP/EN-US/scancode.exe>>
- [8]. "PDIUSB12, USB interface device with parallel bus" Rev. 08. 20 Desember 2001. 35 hal. Philips Semiconductors. 12 Juni 2002. <<http://www.semiconductors.philips.com/acrobat/datasheets/PDIUSB12-08.pdf>>
- [9]. Peacock, Craig. "USB in a Nutshell, Making Sense of the USB Standard". Second Release. 9 Mei 2002. 30 hal. Beyond Logic. 12 Juni 2002. <<http://www.beyondlogic.org/usbnutshell/usb-in-a-nutshell.pdf>>
- [10]. "Universal Serial Bus (USB) Device Class Definition for Human Interface Device (HID)". Version 1.11. 27 Juni 2001. 98 hal. USB Implementers Forum. 19 Juni 2002. <[http://www.usb.org/developers/data/devclass/hid1\\_11.pdf](http://www.usb.org/developers/data/devclass/hid1_11.pdf)>
- [11]. "Universal Serial Bus Specification" Revision 1.1. 23 September 1998. 327 hal. USB Implementers Forum. 5 Juni 2002. <<http://www.usb.org/developers/data/usbpec.zip>>
- [12]. "USB HID to PS/2 Scan Code Translation Table". Revision 1. 17 Juni 1999. 5 hal. Microsoft Corporation. 19 Juni 2002. <<http://www.microsoft.com/hwdev/download/tech/input/translate.pdf>>