2 December 1998

Philips USB Hub (PDIUSBH11A) Rev 1.1
Release Notes

Philips Semiconductors - Asia Product Innovation Centre
Visit http://www.flexiusb.com

Disclaimers:

Life Support – These products are not designed for use in life support appliances, devices or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips Semiconductors customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips Semiconductors for any damages resulting from such application.

Right to make change – Philips Semiconductors reserves the right to make changes, without notice, in the products, including circuits, standard cells, and/or software, described or contained herein in order to improve design and/or performance. Philips Semiconductors assumes no responsibility or liability for the use of any of these products, conveys no license or title under any patent, copyright, or mask work right to these products and makes no representations or warranties that these products are free from patent, copyright or mask work right infringement, unless otherwise specified.

Application information – Applications that are described herein for any of these products are for illustrative purposes only. Philips Semiconductors make no representation or warranty that such applications will be suitable for the specified use without further test or modification

LICENSES

Purchase of Philips I^{2}C components

Purchase of Philips I^{2}C components conveys a license under the Philips I^{2}C patent to use the components in the I^{2}C system provided the system conforms to the I^{2}C specification defined by Philips. This specification can be ordered using the code 9398 393 40011.

Philips USB Hub (PDIUSBH11A) Rev 1.1 – Release Notes

INTRODUCTION

This demonstration kit uses the PDIUSBH11A (H11A) to implement a bus-powered USB keyboard hub. There is a single DIP socket for an MCS51 family of micro-controllers. The minimum ROM size for implementing a keyboard Hub is 4K.

FEATURES

This release of the Keyboard Hub comes with the following components:

- Philips Keyboard Hub
- Diskette containing the firmware and **Release Notes** document
- USB Cable

Philips Keyboard Hub PCB

The keyboard PCB comes with footprints for using the PDIUSBD11 (D11) as well as the PDIUSBH12 (H12). However, due to space constraints, the H11A is hard-soldered onto the PCB. This does not hinder development for the D11 and H12, as the embedded port of the H11A is functionally equivalent to the D11 and H12 without their physical downstream ports.

1. CONNECTORS

Here is a table describing the connectors:

Name	Description
J1	USB Upstream connector
J2	USB Downstream connector, Port 2∗
J3	USB Downstream connector, Port 3∗
J4	USB Downstream connector, Port 4∗
J5	USB Downstream connector, Port 5∗
JP1	Expansion bus (micro-controller side)
JP2	Expansion bus (Hub board side)
JP4	PS/2 Connector (Female)
JP5	No implementation
JP6	No implementation

∗The embedded port of the H11A is implemented as Port 1. All the physical ports are numbered from Port 2 to Port 5.

2. EXPANSION BUS

The Keyboard Hub PCB is made to be breakable so that the separated part containing the H11A can be accessible for debugging. When breaking the PCB, the micro-controller needs to communicate with the H11A via the expansion bus. The following list describes the pins of the bus:

Pin number	Description
1 - GND	Ground
2 - Clkout	Clock output from the H11A
3 - TXD	Transmit (from the UART of the micro-controller)
4 - SCL	SCL - to the H11A/H12/D11
5 - SDA	SDA - to the H11A/H12/D11
6 - INT_N	Interrupt from the H11A, active low
7 - SUSPEND	The Suspend indicator from the H11A can be forced low to keep H11A from suspending
8 - Rcx	Receive (from the UART of the micro-controller)
9 - HWSCL	I/O from the micro-controller
10 - HWSDA	I/O from the micro-controller

3. INDICATORS

The following table describes the LED indicators:

Name	Description
LED1 (green)	This indicates that the downstream port 5 is enabled. Whenever the downstream port is attached and a proper port detection is done, the downstream port is enabled. If the H11A is in GoodLink™ mode, this LED blinks when traffic is detected at Port 5.
LED2 – 4 (green)	The same as LED1, except that these indicators correspond to downstream ports 2-5 in the following way: LED2 – Downstream Port 4 LED3 – Downstream Port 3 LED4 – Downstream Port 2
LED5 (red)	This LED indicates that the downstream ports are powered. When the Hub is configured, this LED should be lit when the downstream ports are powered.

Philips USB Hub (PDIUSBH11A) Rev 1.1 – Release Notes

H12-D11 H11A Converter

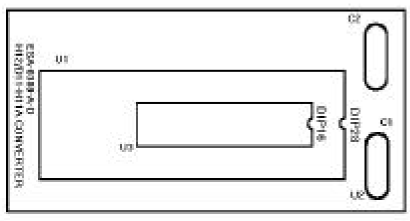

FIG 1: H12-D11 H11A CONVERTER

This converter is a piggy-back module on the H11A socket. It matches the DIP foot-print with PDIUSBD11 and PDIUSBH12. When inserting the converter, the capacitor C2 should rest directly on top of the H11A socket.

SYSTEM CONFIGURATION

The demonstration kit serves as an evaluation -and-development platform for using the PDIUSBH11A, PDIUSBH12 and the PDIUSBD11. Each of the configurations requires the designated firmware on the micro-controller before it can perform its specific function.

Keyboard Hub

1. H11A or H12 (KBHUB213.asm)

All four downstream ports are utilized when using H11A. However, to use H12, the descriptor for the Hub needs to be changed to accommodate only 2 downstream ports instead of 4.

The firmware to implement **this** is KBHUB213.asm. KBHUB213.hex, an Intel-hex version, is also contained on the disk. The micro-controller is programmed with this code before shipping.

The keyboard functionality rests on the embedded port (Port 1). As soon as the Hub is configured, the keyboard functionality will be enumerated.

Keyboard with PS2 mouse (Device Only)

2. D11 (KBMS1_7.asm)

This application uses a D11 in combination with a micro-controller to implement a simple USB Keyboard. The D11 acts as a front-end to buffer USB traffic until the micro-controller has time to service via I2C. This shows the use of multiple endpoints to handle traffic from different applications.

The Host recognizes a single device with multiple interfaces. Therefore, the keyboard enumerates itself as an HID composite device.

The firmware to implement this is KBMS1_7.asm. KBMS1_7.hex, an Intel-hex version, is also contained on the disk.

BUGS AND FIXES

1. Overcurrent Fix

To remove false overcurrent detentions due to in-rush current during plugging in of downstream port devices, it is recommended that a low-pass configuration as shown below be used.

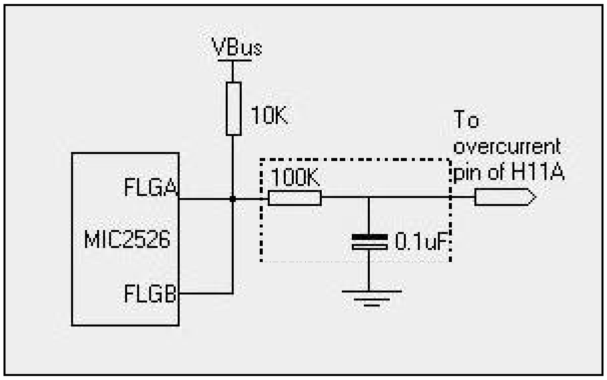

FIG. 2: OVERCURRENT FIX

2. Suspend pull-up Fix

The suspend pin from the H11A is an open-drain output . It should be pulled up to a +5V instead of +3.3V.

3. Internal H11A Reset

The Rev 1.1 no longer relies on an RC configuration for its power on reset. The Reset_N pin is connected directly to +3.3V.

4. Suspend Current

The Keyboard Hub is not designed to meet the suspend current designated at 500 uA. The current budgeting for the main components at suspend state is shown below:

Item	Current Consumed	Remarks
78L033	400 uA	Common Voltage regulator
D+ pull-up,15 K resistor	200 uA	Mandatory. This is absorbed into H11A. Measuring the suspend of the H11A would yield about 200 uA, the bulk of which is consumed by this pull-up resistor.
Micro-controller	18 mA (Active)	Power-down mode is not a standard

8051	4 mA (Idle) 50 uA (Power-down)	feature.
H11A	1.5 mA (PLL running) 5 uA (Lazy Clock, exclusive D+ pull-up)	Switching to to Lazy Clock on suspend will reduce power consumption

From the table, it is clear that to be within the current limit of 500 uA, the 3.3V voltage regulator and the micro-controller have to consume less than 300 uA. Micrel's LDO Voltage regulator in general has a low quiescent current. One example is MIC5207-3.3BZ which consumes 5 uA on shutdown.

[The keyboard hub was designed as an aid to develop products using the H11A; there was therefore no need to integrate costly components.]