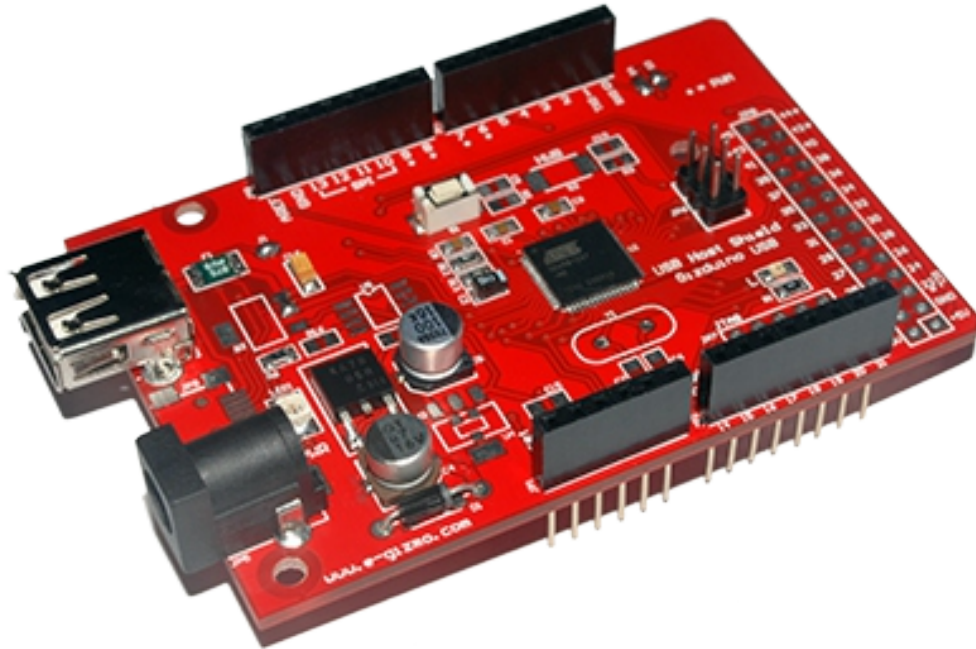


USB Host Shield 2



Features and Specifications

The e-Gizmo USB Host Shield 2 is used for interfacing with USB devices such as an optical mouse or a flash drive. The shield can also be used as an input/output expander with its extension pins. USB data are sent/received via Serial communication.

General Specifications:

Power input:

+5V

USB host controller:

AT90USB64/128



MAJOR COMPONENTS PRESENTATION

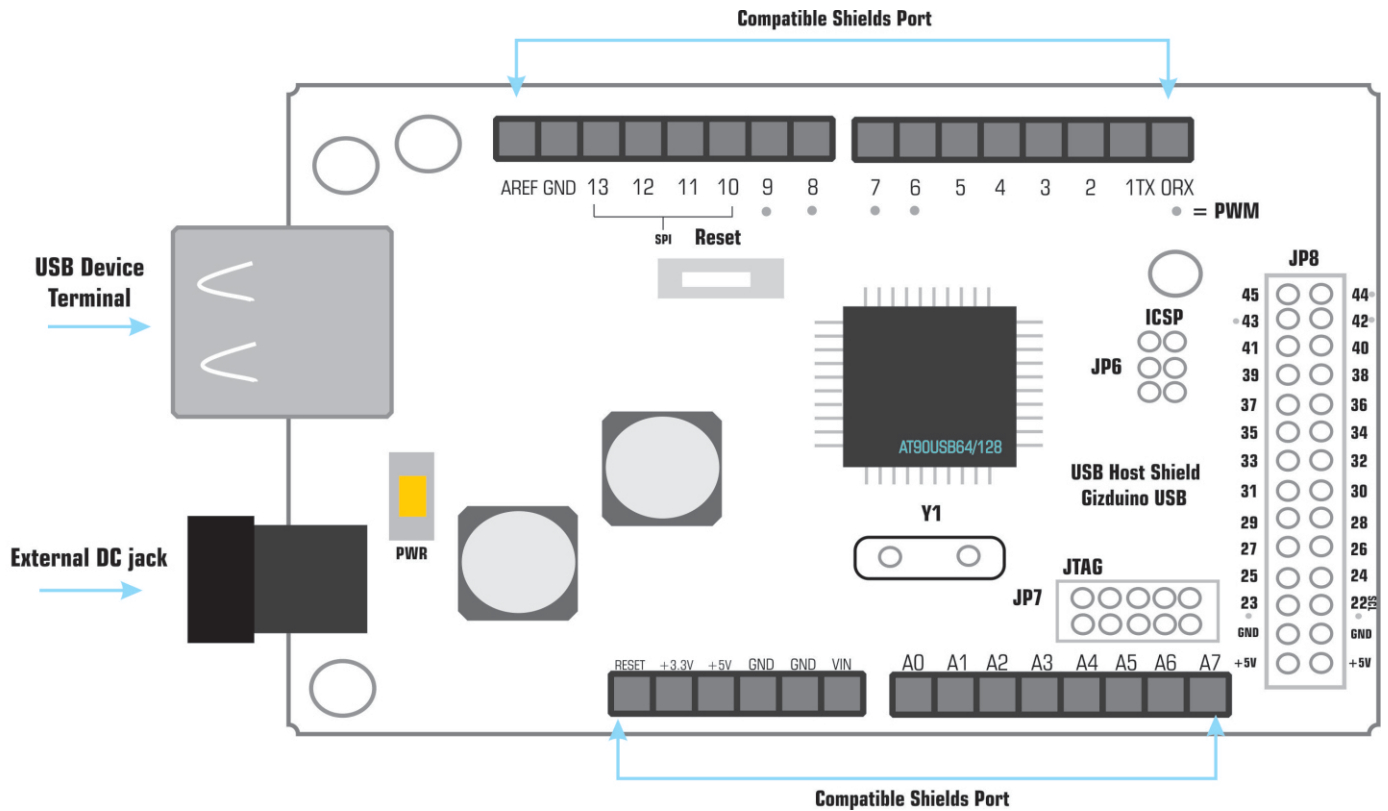
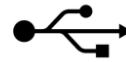


Figure 1. USB Host Shield 2 Components Presentation

USB Host Compatibility:

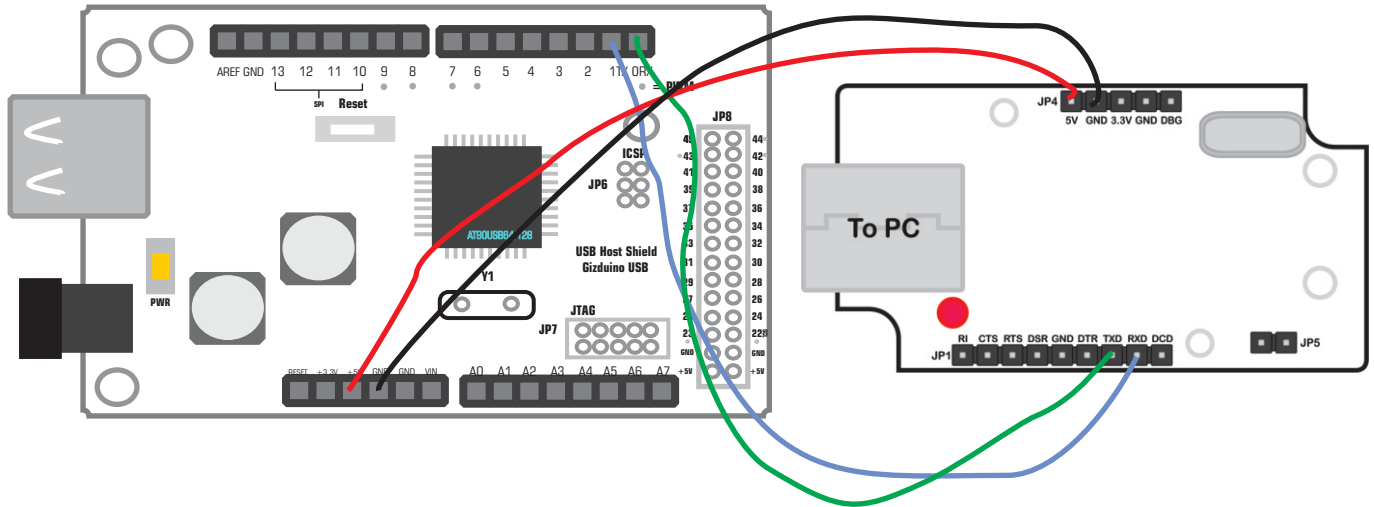
- Common Optical USB Mouse.
- Mass Storage Devices such as Flash Drives (FAT32) 2.0 or above.



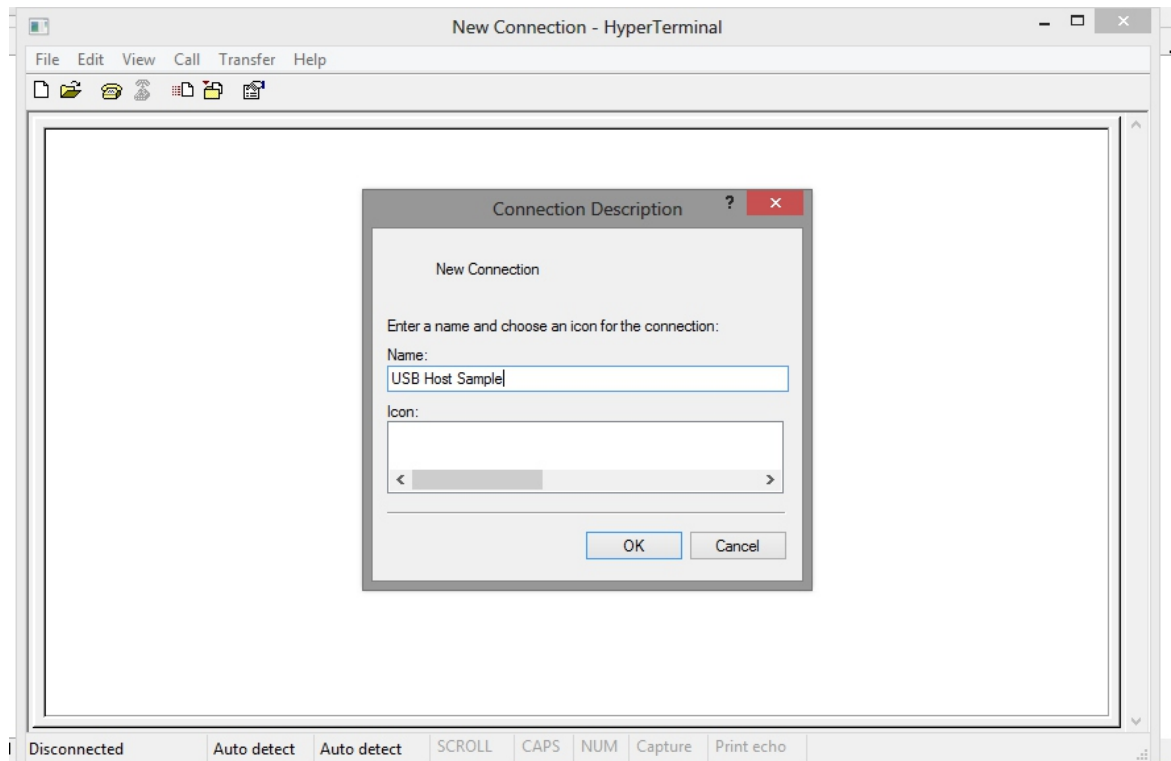
USER GUIDE



One of the many ways the user can use the USB Host Shield 2 is by connecting it to a USB-UART Converter via Serial Communication. This user guide contains the connections and default settings when used on a terminal emulator. Simply connect the USB Host Shield 2 to your USB-UART converter as shown on the image below,

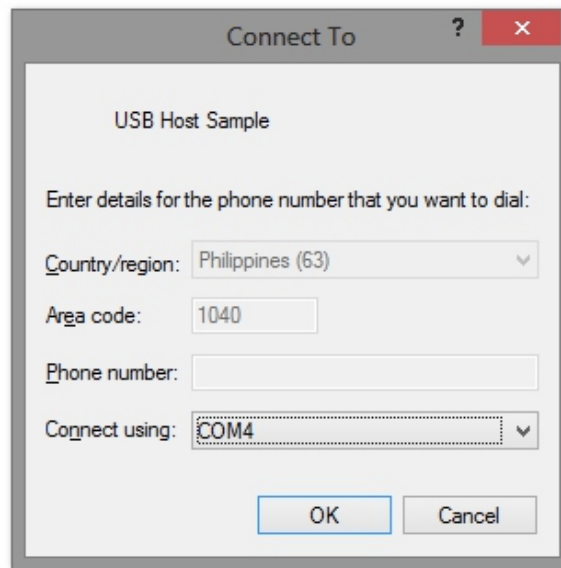


STEP1: Open Hyperterminal and enter any name for your connection

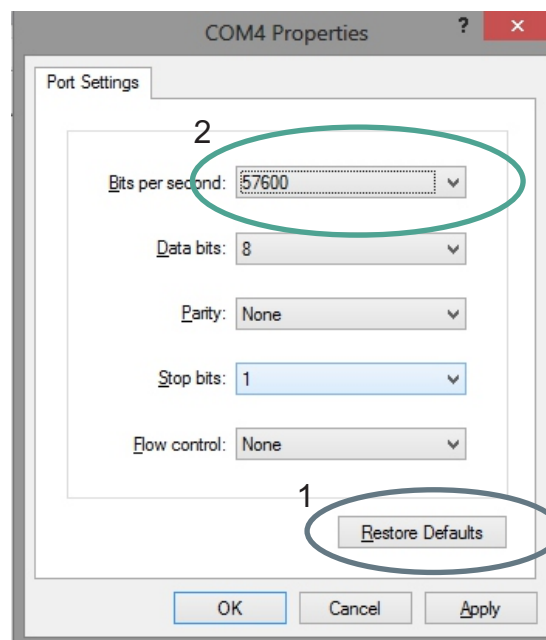




STEP2: The **Connect To** window shall show up. Select the COM port you used for the USB-UART converter.



STEP3: On the next window, click **Restore Defaults**, then change bits per second to 57,600 (this is the DEFAULT baud rate of the USB host)





Interfacing with a USB Mouse

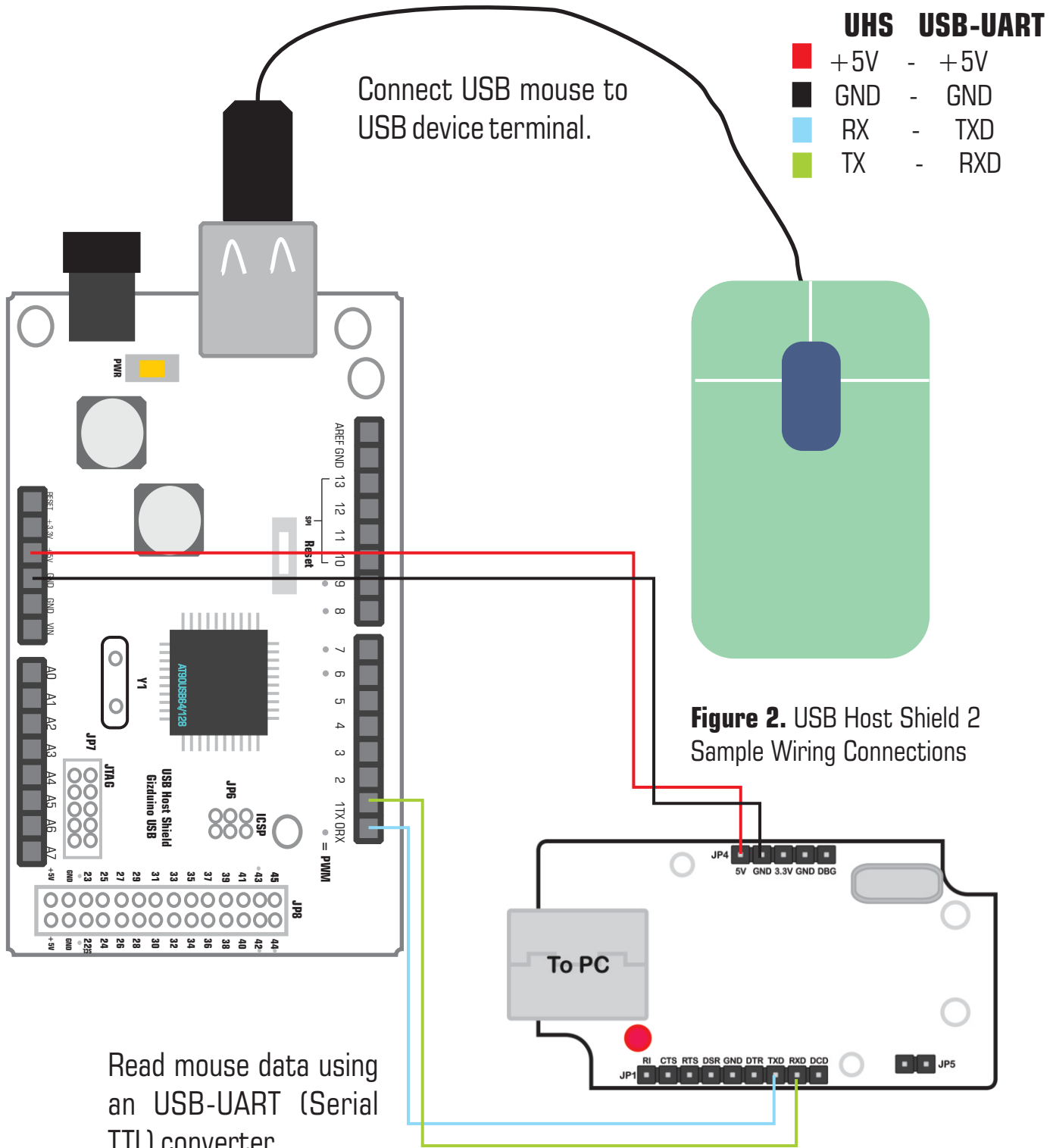


Figure 2. USB Host Shield 2 Sample Wiring Connections





Interfacing with a USB Mouse

The data received from the shield when using a mouse are divided into four set of bits and are shown as 0,0,0,0. Examples of data received upon mouse movement are as follows:

Mouse button, X axis, Y axis, Scroll

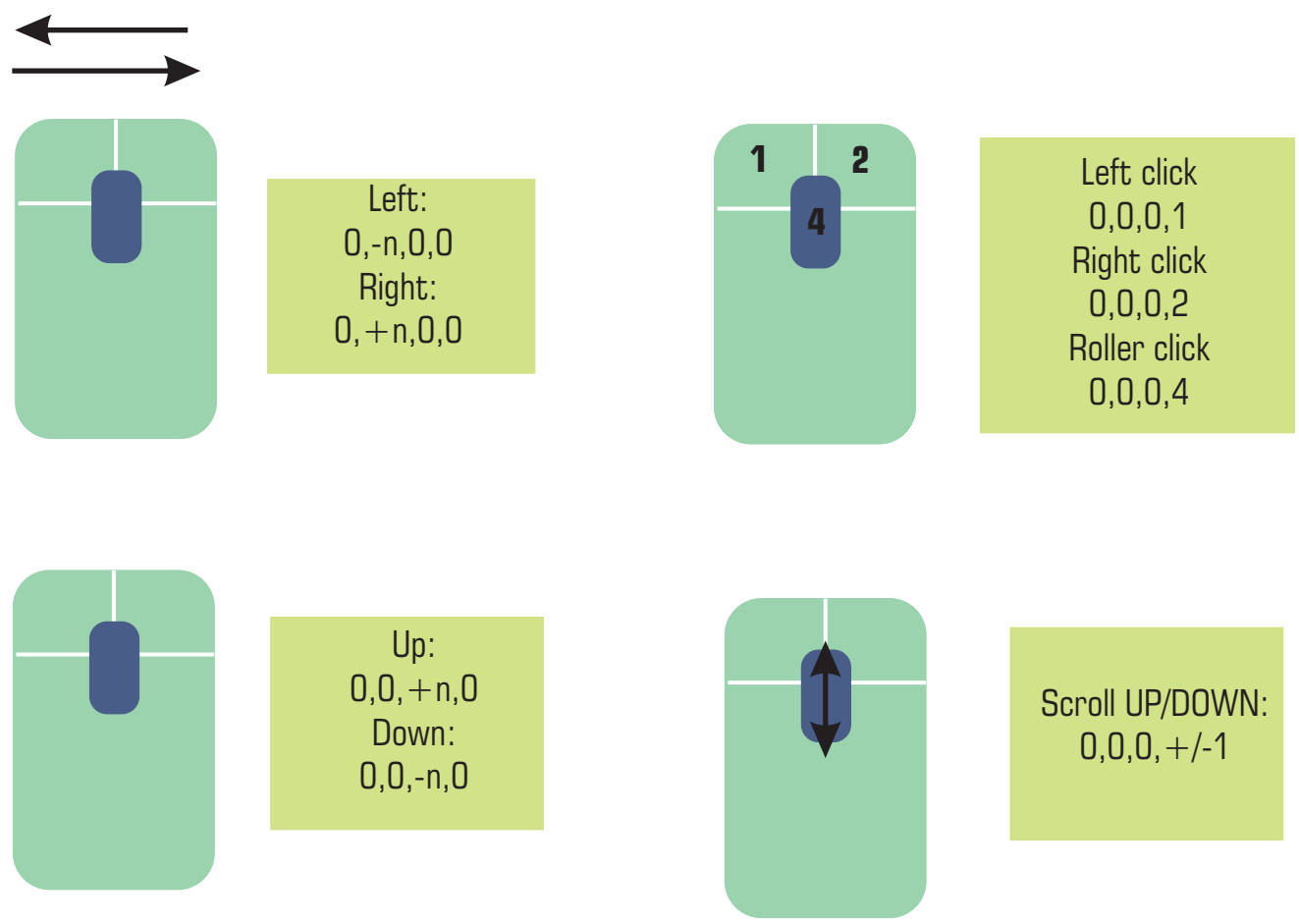


Figure 3. USB Host Shield 2 sample application illustrations



Interfacing with mass storage devices

Besides using a mouse, the USB host shield 2 can also access a flash drive through a serial terminal emulator such as hyperterminal. **The flash drive must be in FAT32 format for it to be accessible.** The manual provided by ATMEL is also provided in the same folder as this manual with the filename "Commands List.html".

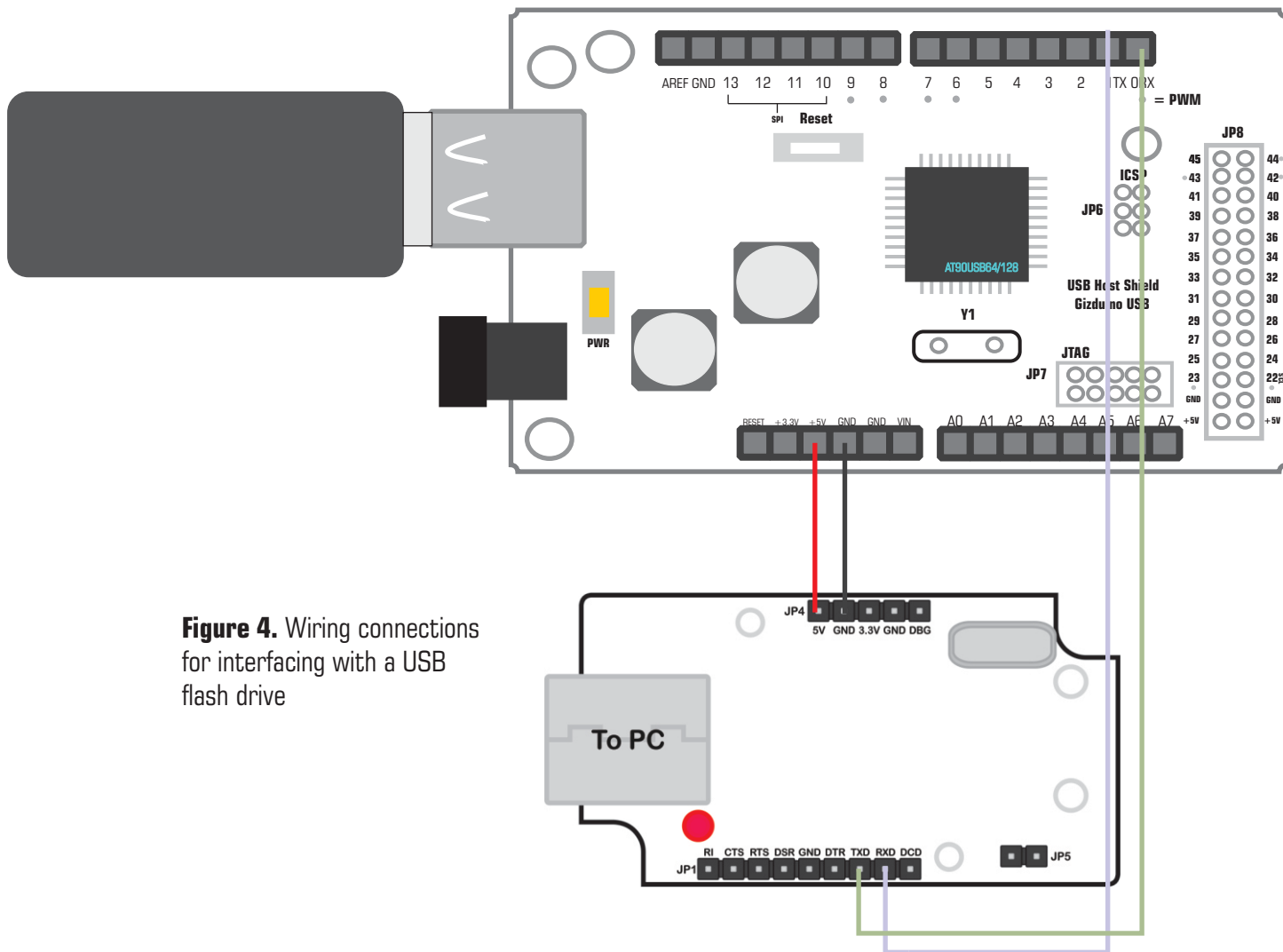


Figure 4. Wiring connections for interfacing with a USB flash drive



COMMANDS LIST



Start-up screen:

```
-----  
ATMEL AVR uShell  
-----
```

Memory interface available:
a: "On board data flash"

Start-up commands:

1. reboot - resets terminal
2. lsusb - shows flash drive information

Input:

```
$> lsusb
```

Output:

```
Device 1 @:0x05  
VID:058F, PID:6387, MaxPower is  
100mA, Device is bus-powered  
Control Endpoint is 64 bytes, Device is  
full-speed  
Device does not support remote wake-up  
Supported interface(s):01  
Interface nb:00, AltS nb:00, Class:08,  
SubClass:06, Protocol:80
```

```
Endpoint(s) Addr: 01 82  
Physical pipe(s): 01 02
```

3. resume - allows USB activity
4. suspend - suspends USB activity

Commands:

1. mount (disk) - mounts the selected disk.
Removable storage is usually named as b:

Input:

```
$> mount b
```

Output:

```
(none)
```

2. disk - checks all available disk and shows their functions

Input:

```
$> disk
```

Output:

```
Memory interface available:  
a: "On board data flash"  
b: "USB Remote memory"
```

3. a: or b: - selects selected drive

Input:

```
>$ a:
```

Output:

```
(none)
```

4. df - checks drive free space.

Input:

```
$> df
```

Output:

```
b: "USB Remote memory"  
Free space: 99 %
```

5. format drivename, with drivename - removes all files and reformats drive with



COMMANDS LIST



the drivename specified

Input:

```
$> format b
```

Output:

(none)

6. deltree - deletes files and directories permanently

Input:

```
$> deltree b:
```

Output:

(none)

7. mkdir dirname - makes a directory

Input:

```
$> mkdir b
```

Output:

(none)

8. cd - change directories

Input:

```
$> cd
```

Output:

(none)

9. cd dirname - change directories with specific directory name

Input:

```
$> cd b
```

Output:

(none)

10. ls - lists all files

Input:

```
$> ls
```

Output:

b: volume is "USB Remote memory"

Drive uses FAT32

Dir name is B:

Size	Name
------	------

Dir	0	DeleteMe
-----	---	----------

	3	Files
--	---	-------

	1	Dir
--	---	-----

10. mark - bookmarks current directory

Input:

```
$> mark b:
```

Output:

(none)

11. touch filename - makes a text file with the specified filename

Input:

```
$> touch DeleteMe
```

```
$> ls
```

Output:

b: volume is "USB Remote memory"





```
Drive uses FAT32
Dir name is B:
      Size Name
Dir    0  Deleteme
      1 file
```

12. cp filename - copy filename to bookmark

Input:

```
$> cp filename
```

Output:

```
(none)
```

13. cat filename1 filename2 - concatenates texts of specified filenames

Input:

```
$> cat DeleteMe Sample
```

Output:

```
(none)
```

14. append filename - adds text on specified filename

Input:

```
$> append DeleteMe
```

Output:

```
Simple text editor, enter char to
append, ^q to exit and save
DeleteMe
```

15. goto - goes to bookmark specified

Input:

```
$> goto
```

Output:

```
(none)
```



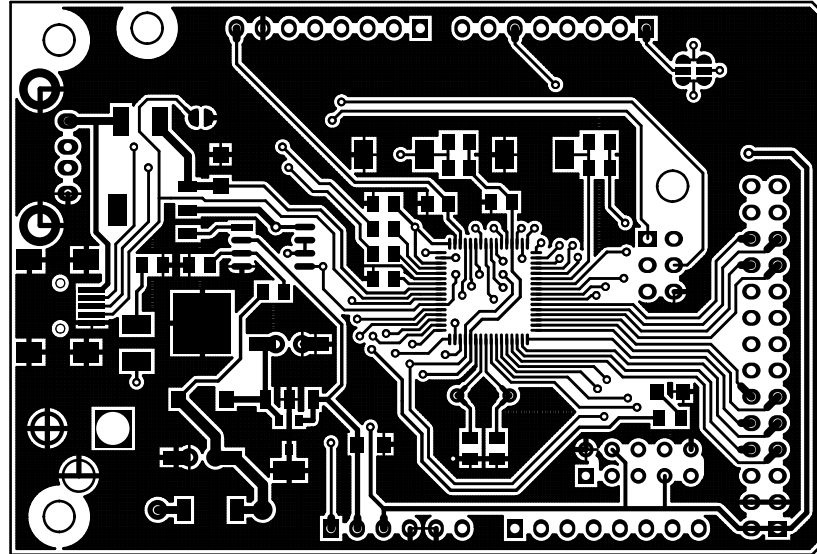
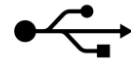


Figure 5. USB Host Shield 2 Front PCB Layout

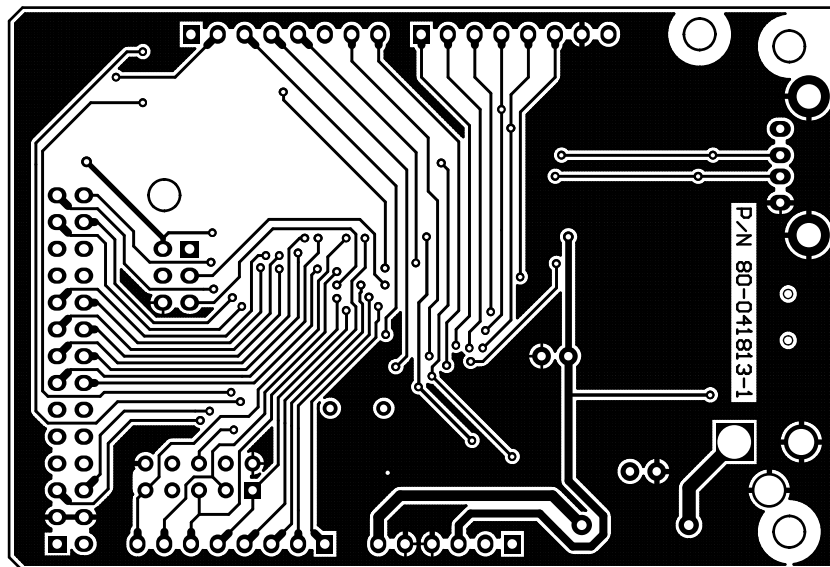


Figure 6. USB Host Shield 2 Back PCB Layout

PARTS PLACEMENT

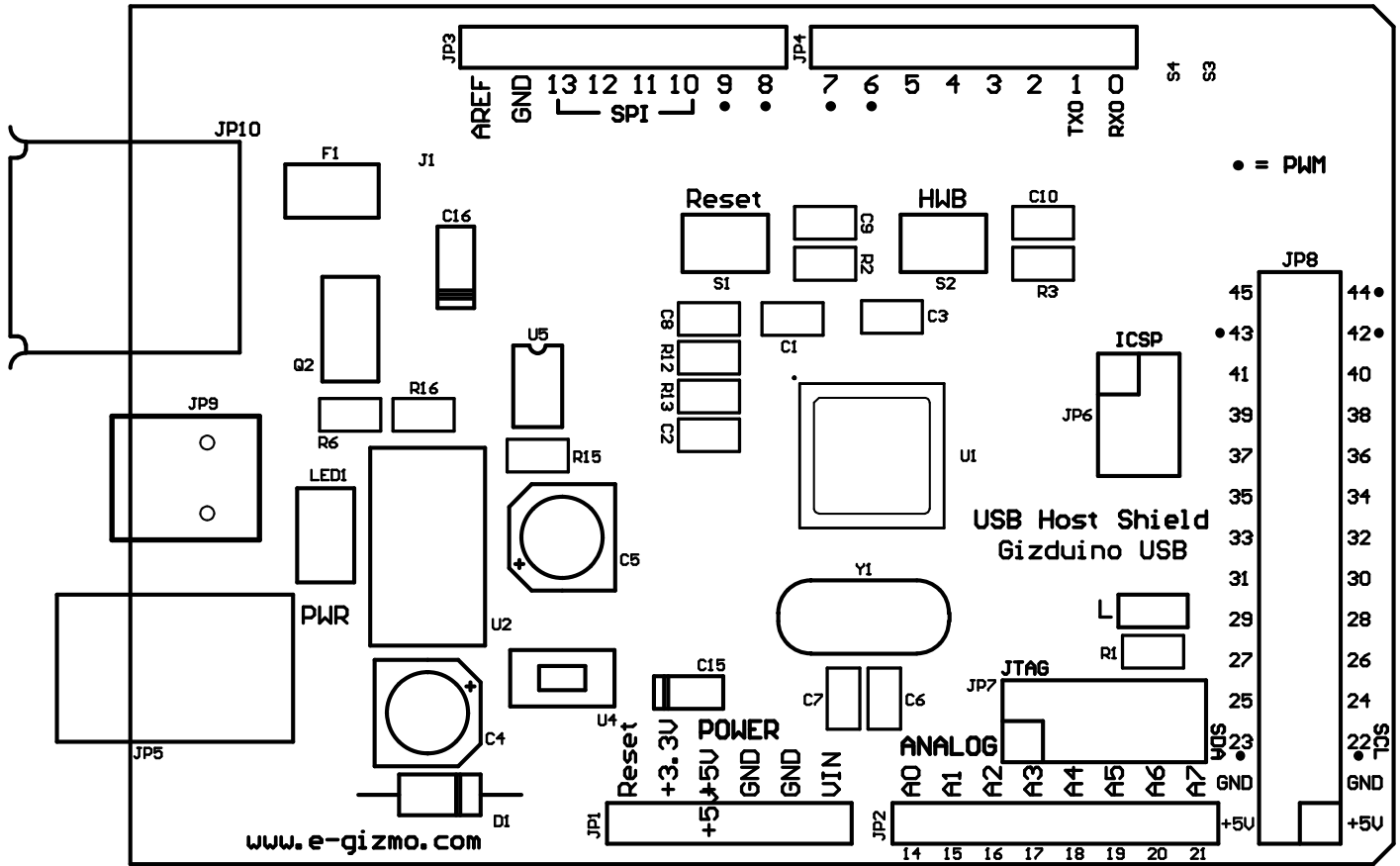


Figure 7. USB Host Shield 2 Parts Placement



SCHEMATIC DIAGRAM



USB HOST SHIELD 2

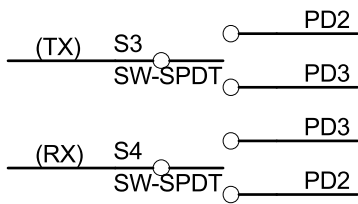
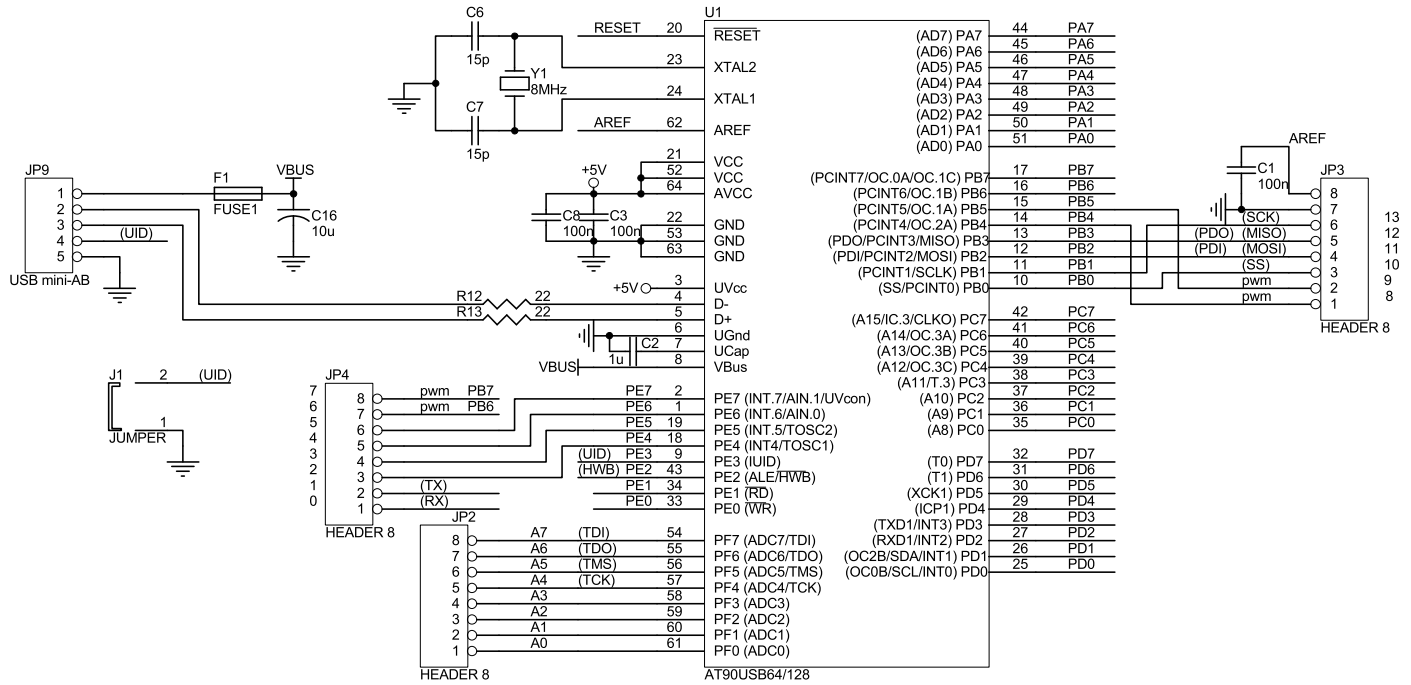


Figure 8. USB Host Shield 2 Schematic diagram part 1



SCHEMATIC DIAGRAM

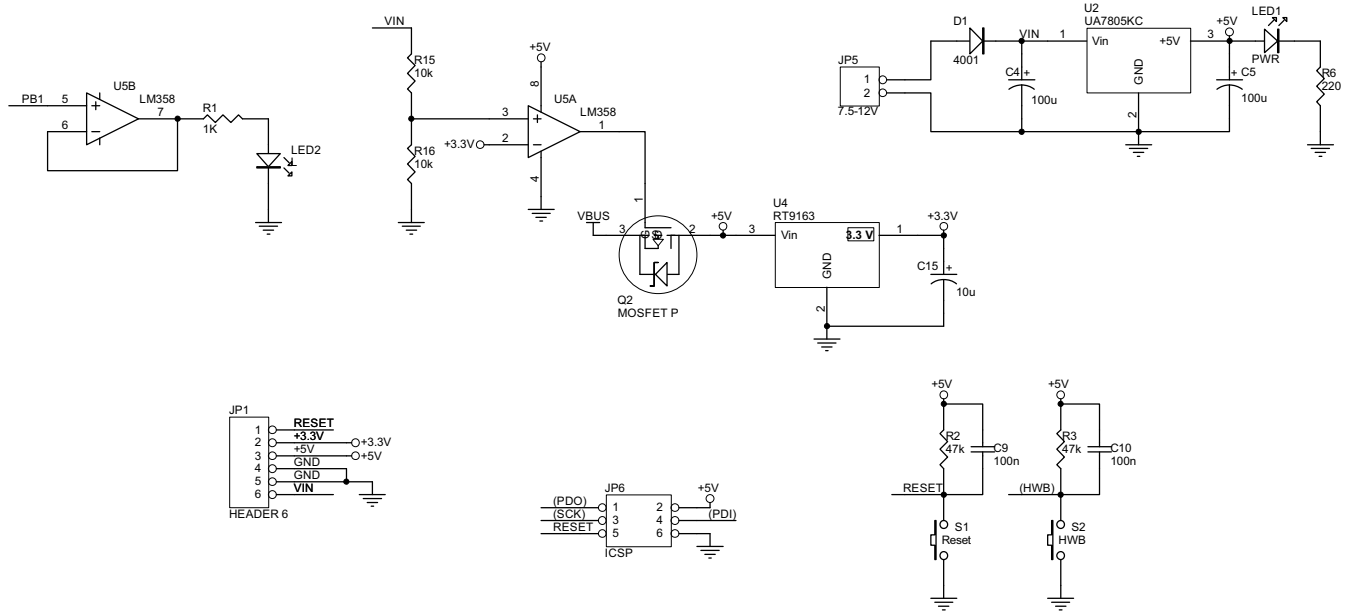
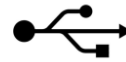


Figure 9. USB Host Shield 2 Schematic diagram part 2

