

---

# **usbinfo Documentation**

***Release 1.1.0***

**Toshiro Yamada, Jeff Herman**

**Nov 18, 2018**



---

## Contents

---

<b>1</b>	<b>Installation</b>	<b>3</b>
<b>2</b>	<b>API documentation</b>	<b>5</b>
2.1	usbinfo module . . . . .	5
<b>3</b>	<b>Invocation of <code>usbinfo</code> command line tool</b>	<b>9</b>
<b>4</b>	<b>History</b>	<b>11</b>
4.1	Releases . . . . .	11
<b>5</b>	<b>Indices and tables</b>	<b>13</b>
	<b>Python Module Index</b>	<b>15</b>



USBInfo provides Python developers a way to uniformly access information about USB endpoints on a system without the need to understand the fine details of any one particular platform's implementation of USB. This is useful in robotics and device automation and allows developers to write more portable code.



# CHAPTER 1

---

## Installation

---

This version of USBInfo requires Python 2.6 or above running on a POSIX-compliant system.

USBInfo is on PyPI and can be installed using:

```
pip install usbinfo
```





## 2.1 usbinfomodule

The *usbinfomodule* provides methods for gathering information from the USB subsystem. The *usbinfomodule()* function, for example, returns a list of all USB endpoints in the system and information pertaining to each device.

For example:

```
import usbinfomodule
usbinfomodule.usbinfomodule()
```

might return something like the following:

```
[{'bInterfaceNumber': '',
  'bNumEndpoints': '1',
  'devname': '',
  'iManufacturer': 'Apple Inc.',
  'iProduct': 'XHCI Root Hub SS Simulation',
  'iSerialNumber': '',
  'idProduct': '8007',
  'idVendor': '05ac'},
 {'bInterfaceNumber': '0',
  'bNumEndpoints': '1',
  'devname': '',
  'iManufacturer': 'Apple Inc.',
  'iProduct': 'XHCI Root Hub SS Simulation',
  'iSerialNumber': '',
  'idProduct': '8007',
  'idVendor': '05ac'},
 {'bInterfaceNumber': '',
  'bNumEndpoints': '1',
  'devname': '',
  'iManufacturer': 'Apple Inc.',
  'iProduct': 'XHCI Root Hub USB 2.0 Simulation',
```

(continues on next page)

(continued from previous page)

```

    'iSerialNumber': '',
    'idProduct': '8007',
    'idVendor': '05ac'},
  {'bInterfaceNumber': '0',
   'bNumEndpoints': '1',
   'devname': '',
   'iManufacturer': 'Apple Inc.',
   'iProduct': 'XHCI Root Hub USB 2.0 Simulation',
   'iSerialNumber': '',
   'idProduct': '8007',
   'idVendor': '05ac'},
  {'bInterfaceNumber': '0',
   'bNumEndpoints': '2',
   'devname': '',
   'iManufacturer': 'Apple',
   'iProduct': 'Internal Memory Card Reader',
   'iSerialNumber': '000000000820',
   'idProduct': '8406',
   'idVendor': '05ac'},
  {'bInterfaceNumber': '',
   'bNumEndpoints': '1',
   'devname': '',
   'iManufacturer': 'Apple Inc.',
   'iProduct': 'Apple Internal Keyboard / Trackpad',
   'iSerialNumber': '',
   'idProduct': '0262',
   'idVendor': '05ac'},
  {'bInterfaceNumber': '0',
   'bNumEndpoints': '1',
   'devname': '',
   'iManufacturer': 'Apple Inc.',
   'iProduct': 'Apple Internal Keyboard / Trackpad',
   'iSerialNumber': '',
   'idProduct': '0262',
   'idVendor': '05ac'},
  {'bInterfaceNumber': '1',
   'bNumEndpoints': '1',
   'devname': '',
   'iManufacturer': 'Apple Inc.',
   'iProduct': 'Apple Internal Keyboard / Trackpad',
   'iSerialNumber': '',
   'idProduct': '0262',
   'idVendor': '05ac'},
  {'bInterfaceNumber': '2',
   'bNumEndpoints': '1',
   'devname': '',
   'iManufacturer': 'Apple Inc.',
   'iProduct': 'Apple Internal Keyboard / Trackpad',
   'iSerialNumber': '',
   'idProduct': '0262',
   'idVendor': '05ac'},
  {'bInterfaceNumber': '',
   'bNumEndpoints': '1',
   'devname': '',
   'iManufacturer': 'Apple Inc.',
   'iProduct': 'BRCM20702 Hub',
   'iSerialNumber': ''

```

(continues on next page)

(continued from previous page)

```

'idProduct': '4500',
'idVendor': '0a5c'},
{'bInterfaceNumber': '0',
'bNumEndpoints': '1',
'devname': '',
'iManufacturer': 'Apple Inc.',
'iProduct': 'BRCM20702 Hub',
'iSerialNumber': '',
'idProduct': '4500',
'idVendor': '0a5c'},
{'bInterfaceNumber': '',
'bNumEndpoints': '1',
'devname': '',
'iManufacturer': 'Kingston',
'iProduct': 'DataTraveler 2.0',
'iSerialNumber': 'AC221C280D9FFEABC85A1812',
'idProduct': '6545',
'idVendor': '0930'},
{'bInterfaceNumber': '0',
'bNumEndpoints': '2',
'devname': '/dev/disk2s1',
'iManufacturer': 'Kingston',
'iProduct': 'DataTraveler 2.0',
'iSerialNumber': 'AC221C280D9FFEABC85A1812',
'idProduct': '6545',
'idVendor': '0930',
'mount': '/Volumes/KINGSTON'}}]

```

`usbinfo.usbinfo()`

Return a list of dictionary describing attached USB endpoints.

This returns a list of USB endpoints attached to the system. Each entry in this list contains a dictionary containing information pertaining to that endpoint.

**Returns:** A list of dictionaries representing each USB endpoint containing the following keys:

- `idVendor` – USB vendor ID of device.
- `idProduct` – USB product ID of device.
- `iManufacturer` – Name of manufacturer of device.
- `iProduct` – Common name of of device.
- `bInterfaceNumber` – On a multi-endpoint device, this is the index of that endpoint.
- `devname` – On a serial communications device, this is the path to the character device file. On a mass storage device, this is the path to the block device file. On all other devices, this field does not exist.
- `mount` – On a mass storage device, this is the path to the mount point.



## CHAPTER 3

---

### Invocation of `usbinfo` command line tool

---

The **`usbinfo`** allows for gathering of information of endpoints on the USB subsystem from the command line. When invoked without any arguments, **`usbinfo`** prints a tabular representation of attached USB endpoints:

vid:pid	Manufacturer	Product	Serial Number	IF
↪ # ( Device Path => Mount Path				
05ac:8007	Apple Inc.	XHCI Root Hub SS Simulation		
05ac:8007	Apple Inc.	XHCI Root Hub SS Simulation		0
05ac:8007	Apple Inc.	XHCI Root Hub USB 2.0 Simulation		
05ac:8007	Apple Inc.	XHCI Root Hub USB 2.0 Simulation		0
05ac:8406	Apple	Internal Memory Card Reader	000000000820	
05ac:8406	Apple	Internal Memory Card Reader	000000000820	0
05ac:0262	Apple Inc.	Apple Internal Keyboard / Trackpad		
05ac:0262	Apple Inc.	Apple Internal Keyboard / Trackpad		0
05ac:0262	Apple Inc.	Apple Internal Keyboard / Trackpad		1
05ac:0262	Apple Inc.	Apple Internal Keyboard / Trackpad		2
0a5c:4500	Apple Inc.	BRCM20702 Hub		
0a5c:4500	Apple Inc.	BRCM20702 Hub		0
05ac:8289	Apple Inc.	Bluetooth USB Host Controller		
05ac:8289	Apple Inc.	Bluetooth USB Host Controller		0
05ac:8289	Apple Inc.	Bluetooth USB Host Controller		1
05ac:8289	Apple Inc.	Bluetooth USB Host Controller		2
05ac:8289	Apple Inc.	Bluetooth USB Host Controller		3
0930:6545	Kingston	DataTraveler 2.0	AC221C280D9FFEABC85A1812	
0930:6545	Kingston	DataTraveler 2.0	AC221C280D9FFEABC85A1812	0
↪ /dev/disk2s1 => /Volumes/KINGSTON				

The **`usbinfo`** script has several options:

**`--csv`**

Format output in CSV

**`-e, --endpoints`**

Display endpoint counts for each device

**`--endpoint-total`**

Print the total number of endpoints



## 4.1 Releases

### 4.1.1 Version 1.0

- Added *usbinfo* allowing scripts to obtain information from USB subsystem.
- Added `usbinfo` script to allow command line usage of *usbinfo*
- Added documentation

Pexpect is developed on [Github](#). Please report [issues](#) there as well.

### 4.1.2 Version 1.0.1

- Fix to include `devname` when running El Capitan

### 4.1.3 Version 1.0.2

- Added Python 3 support
- Convert [readthedocs.org](#) links to use [readthedocs.io](#)





## CHAPTER 5

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



**u**

usbinfo, 5



## Symbols

- csv
  - usbinfo command line option, 9
- endpoint-total
  - usbinfo command line option, 9
- e, -endpoints
  - usbinfo command line option, 9

## U

- usbinfo (module), 5
- usbinfo command line option
  - csv, 9
  - endpoint-total, 9
  - e, -endpoints, 9
- usbinfo() (in module usbinfo), 7