

Re: Capturing USB data?

Source: <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.mfc/2008-08/msg00959.html>

- *From:* Joseph M. Newcomer <newcomer@xxxxxxxxxxxxx>
 - *Date:* Sat, 30 Aug 2008 20:46:38 -0400
-

See below...

On Sat, 30 Aug 2008 11:07:31 -0500, "Peter Olcott" <NoSpam@xxxxxxxxxxxxx> wrote:

What is the best approach to capture all hardware (especially keyboard and mouse) inputs and also be able to send these same hardware inputs to every OS/hardware platform (especially Windows, Apple Mac OS X, and Linux/Unix) ???

There isn't one. There is not what that is platform-independent, there is not one that is device-independent. As far as I know, and as far as I can tell from the kind of behavior I'm seeing, mice and keyboards have unique interface specifications. In addition, the ways of intercepting these events differ with the operating system. Each is unique.

I was thinking about two basic approaches:
(1) USB hardware device that a USB Keyboard and Mouse plug into, along with USB and Keyboard and Mouse drivers for this device.

There are several layers of protocol here

application <--> HID class <--> device minidriver <--> USB class driver <--> USB minidriver

The only standard within Windows is at the HID-class level protocol. Below that, you have no idea what the bits mean. But the Mac uses a different software organization, and Unix/linux use a different software organization. Therefore, you will have to think of each situation as being unique as far as the OS. And anything below the device minidriver is essentially random bits. You would have to emulate the driver functionality on your own, and there is not world enough and time for that.

(2) Modified versions of existing Keyboard/Mouse device drivers that provide for capturing and instantiating events.

Re: Capturing USB data?

So how do you go to Microsoft, Logitech, etc. and say "I want a copy of your mouse driver code so I can modify it to support my product"? Observation: with no success. Actually, the MS mouse driver source might be in the DDK, but you are in for a massive learning curve to build drivers and modify them.

You could consider, in the MS world, building a mouse filter driver and a keyboard filter driver. You could probably succeed in doing one of these in a couple months (they are fairly straightforward, as drivers go).

But what would scare me, if I were providing oversight on this kind of project, is that it appears to have become an infinite time sink to make any progress. It is now capable of absorbing as many resources as you can throw at it into the indefinite future.

Why does a low-level mouse hook or low-level keyboard hook not do the job? Don't look for an OS-independent solution; build the right abstract interface and then worry about the implementation on an OS-by-OS basis. For Windows, this is a low-level hook. For the Mac or linux, it is some other mechanism.

joe

"Joseph M. Newcomer" <newcomer@xxxxxxxxxxxx> wrote in message news:h3pib41j6fqt9jp6bvji0shqh3ar8sqi4u@xxxxxxxxxxxx

What does capturing every keystroke and mouse click have to do with USB? Not all machines have USB keyboards or USB mice. If your question was "how do I capture every keystroke and mouse click?" that is quite a DIFFERENT question than "how do I capture USB input?"

For example, the format of the USB packets from different vendors can be different; their low-level drivers convert this information from their (possibly proprietary, most likely undocumented) format to mouse clicks and keystrokes, so capturing the USB data isn't going to tell you very much.

Adapting keyboard and mouse drivers is not at all the same as capturing USB data.

Observation: sometimes, when I reboot, my mouse is completely "reprogrammed"; the scroll wheel is interpreted as mouse clicks, mouse clicks are interpreted as mouse motion, etc. This is because on boot it did not properly communicate with the mouse because of the KVM switch. If I switch to another computer and switch back, all is fixed. This suggests

Re: Capturing USB data?

that different brands of mice use different communication packet encodings.

So I'm not sure that capturing USB packets is a particularly valuable, or even useful, concept here. A low-level keyboard and mouse hook should do the job (but note that on Vista, security may prevent getting at these unless you are running at high integrity)

It is important to state the question, not ask for how to achieve some implementation detail. I answered your question, but in the context of "how do I capture USB input to see mouse and keyboard events?" the answer is "this is the wrong question to be asking"

Note that you would also have to intercept all the proprietary packets for tablets, joysticks, game paddles, touch screens, and other devices which create events that can generate character data or request a program to activate. In fact, your program would have to understand the proprietary USB encoding of every HID device in the world.

When something as vanilla as a Microsoft mouse and a Logitech mouse send different packets, I think this is definitely the wrong approach.
joe

On Sat, 30 Aug 2008 09:51:38 -0500, "Peter Olcott"
<NoSpam@xxxxxxxxxxxxxxxx> wrote:

What I really need to do is to capture every keystroke and mouse click and also be able to feed these to the system.

I already know how to do this for MS Windows.

I need to generalize this capability across all other platforms, especially Mac OS X, and Linux. My current thinking is adapting a keyboard and mouse driver on these platforms. I would probably start with Intel/Windows because this is the platform that I am the most familiar with. I was thinking that restricting my research to the USB interface might simplify things since all three platforms provide this interface.

I would eventually want this same capability literally

Re: Capturing USB data?

across every platform, including hand-held devices. Initially this capability may be limited to devices that provide a USB interface for their hardware input device(s), eventually this restriction would also be removed.

"David Webber" <dave@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
wrote in
message
news:u5ATL2qCJHA.3576@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

"Peter Olcott" <NoSpam@xxxxxxxxxxxxxx>
wrote in message
[news:9kbuk.1863\\$Fm6.1271@xxxxxxxxxxxxxxxxxxxx](mailto:news:9kbuk.1863$Fm6.1271@xxxxxxxxxxxxxxxxxxxx)

How would I go about
capturing all of the data sent
to
a
particular USB port?

I have never done this in any general sense,
but I can
confirm that signals from a MIDI piano
keyboard plugged
into a USB port are automatically picked up
by the usual
MIDI-in routines of the Windows
Multimedia API.

Dave

--

David Webber
Author of 'Mozart the Music Processor'
<http://www.mozart.co.uk>
For discussion/support see
<http://www.mozart.co.uk/mozartists/maillinglist.htm>

Joseph M. Newcomer [MVP]
email: newcomer@xxxxxxxxxxxxxx
Web: <http://www.flounder.com>
MVP Tips: http://www.flounder.com/mvp_tips.htm

Re: Capturing USB data?

Joseph M. Newcomer [MVP]

email: newcomer@xxxxxxxxxxxxx

Web: <http://www.flounder.com>

MVP Tips: http://www.flounder.com/mvp_tips.htm

.