

Re: Supporting a PS/2 Keyboard

Source:

<http://www.tech-archive.net/Archive/WindowsCE/microsoft.public.windowsce.platbuilder/2006-12/msg00353.html>

- *From:* "Paul G. Tobey [eMVP]" <p space tobey no spam AT no instrument no spam DOT com>
 - *Date:* Wed, 6 Dec 2006 10:27:14 -0700
-

No, nothing that I've ever seen. It would not be portable, anyway, as the GPIO functions for each processor version would be different. Even if you found one that worked fine on Samsung processor, say, it wouldn't necessarily work on a PXA processor.

Yes, I guess so. That seems rather more complicated than just putting an 8042 on the board, but it might work. I don't have any USB to PS2 converters around here or I'd try it on the Mainstone board. I guess that you could do the same with whatever reference design hardware you have.

I don't know what the internals of the PS/2 to USB converters are, but it's not just translating signal levels or something. USB devices have to identify themselves among other behaviors.

Maybe it's just because I've done it this way, but adding an 8042-type device, seems much simpler than hard-wiring a USB to PS/2 converter onto your board...

Paul T.

"tvle" <tvle@xxxxxxxxxxxx> wrote in message
<news:1165423933.232177.276280@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>

Thanks. So there's not anything in Platform Builder that does a direct interface to the PS/2 connector then, right? E.g. The NOP keyboard driver doesn't help here then?

One alternative then is to get the hardware guy to put a PS/2 to USB plug in the hardware and have that plug connected to the USB port of the hardware board. I assume that CE's USB HID driver supports a USB keyboard out of the box right? So there's not an additional driver that needs to be included/loaded.

On the USB keyboard or the PS/2 to USB plug then there must be some active component that does the function of the 8042 and does it through the USB connection (?). I assume there can't be just a connection to convert PS/2 signals to USB signals but obviously I'm not that familiar with how that works. I want to make sure there's not some way of

Re: Supporting a PS/2 Keyboard

interfacing directly to the PS2 connector through the supplied PB code/drivers. I'd rather not try and implement the 8042's firmware in my code if there's already something available...

Thanks again for your time in replying to my messages.

Paul G. Tobey [eMVP] wrote:

As I said, anything is possible, but you're going to have to figure out the timing and interpretation of all of the keyboard signals. You get to be the first, so you'd better get started ;-)

The USB keyboard driver is an HID driver. The keyboard part is just the interpreter that converts the scan codes returned by the USB part of the driver into characters (that is, it knows what scan code on the keyboard corresponds to the "U" key and does the right thing to translate the scan code into "U", when asked by the keyboard driver).

There are various keypad drivers in BSPs for development boards and, as I said, there's the 8042 driver in source form in Platform Builder. You won't run out of examples. It's the hardware interfacing that's the hard part.

You can use an 8042 controller. Driver is written for you. Hardware cost goes up.

You can implement some sort of 8042 IP in an FPGA or CPLD. Driver is written for you (although you have to be sure that the interface to the "8042" really looks like an 8042). Hardware cost might go up, if you have to use a bigger FPGA/CPLD than you have on your board now, or, if you have free space in your programmable device, maybe not.

You can try to do everything from software with GPIO. You have to do everything in software. Hardware cost is not affected.

Paul T.

"tvle" <tvle@xxxxxxxxxxxx> wrote in message
news:1165361407.083282.288950@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Thanks for the reply.

We have a legacy UI device that we are trying to replace with a new Intel PXA270 based board and Windows CE 5.0. The legacy device allowed

Re: Supporting a PS/2 Keyboard

the user to plug in a PS/2 keyboard if they wanted to use it on some screens that required a lot of input. The UI device had no real intelligence to it and basically sent info back to a host computer via an RS232 serial link using a proprietary serial protocol.

The host computer received the message containing the key codes and reacted accordingly (by sending subsequent data back to the UI device).

The replacement device with CE does not have an 8042 in the hardware and the thought was that the PS/2 connector would be wired to the CPU I/O pins and software would handle interfacing to the keyboard.

Is this possible? How does the USB driver for a USB keyboard work in CE? Does it layer itself so that it eventually reuses the keyboard driver or associated dll's?

Or will the hardware need to be redesigned to have the 8042 controller or have a PS2 to USB converter plug?

Thanks for any and all comments and suggestions here.
Paul G. Tobey [eMVP] wrote:

Anything is possible, but it seems like there would be a lot of little gotchas that you'd have to implement in software running on the main processor. You could just put an 8042 in there; there's a driver for it in Platform Builder.

I don't understand what passing the codes from your device to the serial port has to do with how the keyboard codes are arriving. Am I missing something? Are you saying that you need something at a different level than

Re: Supporting a PS/2 Keyboard

what an application running on the device to which the keyboard was connected would get?

Paul T.

"tvle" <tvle@xxxxxxxxxxxx> wrote in message

news:1165357189.374073.223080@xx

Hi everyone,

Is it possible to have a PS/2 keyboard connector interfaced directly to the Intel PXA270 (via GPIO pins) and leverage the existing KeyBD driver to get scan codes from the keyboard? Obviously there's not an 8042 keyboard controller in the chain so I don't know if its possible to do this. We have an application which basically is supposed to grab the raw codes from the keyboard and passes them over a serial connection to another application.

I'd appreciate any comments or suggestions any of you may have on this.

Thanks.

Re: Supporting a PS/2 Keyboard