Objective Development Forums

Skip to content

| Search… | Search | Advanced search |

Packet Size > 8 bytes on Interrupt In Endpoint

Post Reply
- Email topic
- Print view

| Search this topic… | Search | Advanced search |

9 posts • Page **1** of **1**

Packet Size > 8 bytes on Interrupt In Endpoint (#p15668)

- Report this post (./report.php?f=8&p=15668)
- Quote (./posting.php?mode=quote&f=8&p=15668)
- (javascript:void(0);)

Postby **IvIePhisto** » Tue Aug 24, 2010 4:46 am

Hi!

I want to modify the V-USB code to allow my device to send more than 8 bytes in one packet. I know it contradicts the standard for low-speed USB devices but other things also work which shouldn't. I know that the buffers are to small, but that's changeable. It's also perfectly possible to supply packet lengths greater than 8 to usbSetInterrupt[3], but it doesn't send all of the data. As I understand it after the label "handleIn3" in "asmcommon.inc" the supplied packet length is sent by "rjmp sendCntAndReti", then the data is sent in "usbSendAndReti". Both labels reside in the clock-specific include files, I'm using 12Mhz so had a look in "usbdrvasm12.inc". I think there packet sizes greater than 8 should work, as the bytes are sent until "cnt" is 0 by "brne txLoop". The only problem I see for my case is "sbrc cnt, 4" after "handleIn3" as bit 4 is used for some NAK signaling, a packet size of 32 is "0b100000". Did I get everything right? Did I miss something? Ideas?

Top

Re: Packet Size > 8 bytes on Interrupt In Endpoint (#p15726)

- Report this post (./report.php?f=8&p=15726)
- Quote (./posting.php?mode=quote&f=8&p=15726)
- (javascript:void(0);)

Postby **ulao** » Mon Aug 30, 2010 3:48 pm

I took the liberty in contacting Christian myself. He admits his busy lifestyle does not easily lend to free time, hence he finds it hard to comment on the forums. However this is his reply.

> Did you investigate whether any host side driver accepts bigger packets?
> I doubt it...
>
> I see more problems, though: Since synchronization takes place only at
> the beginning of the communication, clock deviations sum up and the

> phase of the signals may drift away. Not so much of an issue with
> modules that have a built-in PLL to re-synchronize, but may be bad for
> the standard crystal based modules.
>
> And, as you have noticed, we encode other info in the length. So the
> maximum is probably 31 bytes unless you introduce bigger changes.
>
>
>
> Regards, Christian.

In regards to "Did you investigate whether any host side driver accepts bigger packets? " I'm 99% sure it does. In this case its an Xbox, and we know its packet size is 32. Also I'm pretty confident we dont need 32, as sending 8 seems to satisfy the Xbox. Sending 31 may be ok, I need to see what the last byte is for.

Top

Re: Packet Size > 8 bytes on Interrupt In Endpoint (#p15733)

- Report this post (./report.php?f=8&p=15733)
- Quote (./posting.php?mode=quote&f=8&p=15733)
- (javascript:void(0);)

Postby **IvIePhisto** » Tue Aug 31, 2010 2:24 pm

> And, as you have noticed, we encode other info in the length. So the
> maximum is probably 31 bytes unless you introduce bigger changes.

This is hard thing, to find all instructions that use the length bits.

> **ulao wrote:**
> In regards to "Did you investigate whether any host side driver accepts bigger
> packets? " I'm 99% sure it does. In this case its an Xbox, and we know its packet size
> is 32. Also I'm pretty confident we dont need 32, as sending 8 seems to satisfy the
> Xbox. Sending 31 may be ok, I need to see what the last byte is for.

I think we only need to send 20 bytes, as this is the report size. 20 in binary is "10100", but bit 4 is used for NAK information, so every part of the code that uses this bit has to be changed. This needs some serious Assembler tracing and I will try it. But as I am on vacation for some days, this will take some time...

Top

I got it to work on XBox (#p15904)

- Report this post (./report.php?f=8&p=15904)
- Quote (./posting.php?mode=quote&f=8&p=15904)
- (javascript:void(0);)

Postby **IvIePhisto** » Sat Sep 18, 2010 5:42 pm

OK, I got it and it works on XBox1. I modified interrupt in endpoint 3 (acting as endpoint 2) to send up to 27 bytes, with 4 bytes CRC and sync that makes 31 bytes transferred. Now I'm using

bit 6 for ready signalization on usbTxLen3, before 4 was used. As USBPID_NAK is used for this purposes which has bit 6 set to 1 as well as bit 4 it works on the XBox - Windows correctly spots an error (low-speed device with maximum packet size > 8) and signals the device to malfunction.

The code is scattered across some files, see "xbox.c", "usbdrv/usbdrv.h", "usbdrv/usbrv.c", "usbdrv/asmcommon.inc" and "usbconfig.h" in the folder Firmware/Trunk/usb in the Dual Strike SVN repository (http://svn.combo-convention.de/listing.php? repname=dualstrike&path=%2FFirmware%2FTrunk%2Fusb%2F&rev=320&sc=1) .

Top

Re: Packet Size > 8 bytes on Interrupt In Endpoint (#p15905)

- Report this post (./report.php?f=8&p=15905)
- Quote (./posting.php?mode=quote&f=8&p=15905)
- (javascript:void(0);)

Postby **ulao** » Sat Sep 18, 2010 6:46 pm

Congratulations! can't wait to try it out.

Top

Re: Packet Size > 8 bytes on Interrupt In Endpoint (#p16005)

- Report this post (./report.php?f=8&p=16005)
- Quote (./posting.php?mode=quote&f=8&p=16005)
- (javascript:void(0);)

Postby **ulao** » Mon Sep 27, 2010 12:03 am

I tried with no avail. I get
usbdrv/usbdrv.h:446: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'char'
but have copy and pasted a couple of times. Is there any way to zip up the usbdrv folder?

Top

Re: Packet Size > 8 bytes on Interrupt In Endpoint (#p16050)

- Report this post (./report.php?f=8&p=16050)
- Quote (./posting.php?mode=quote&f=8&p=16050)
- (javascript:void(0);)

Postby **IvIePhisto** » Fri Oct 01, 2010 1:08 pm

Sorry to answer so late, too much to do 🙁

> **ulao wrote:**
> Is there any way to zip up the usbdrv folder?

You could have checked out the SVN repository, but I made a package of the customized V-USB (http://svn.combo-convention.de/filedetails.php? repname=dualstrike&path=%2FFirmware%2Fusbdrv_with_iie3_big_packet_hack.zip) for you. I recommend diff to see the changes to your files 🙂

Top

Re: Packet Size > 8 bytes on Interrupt In Endpoint (#p16052)

- Report this post (./report.php?f=8&p=16052)
- Quote (./posting.php?mode=quote&f=8&p=16052)
- (javascript:void(0);)

Postby **ulao** » Fri Oct 01, 2010 4:53 pm

I could not figure out the svn url. Tried many ways. thx for this!

Top

Re: Packet Size > 8 bytes on Interrupt In Endpoint (#p19229)

- Report this post (./report.php?f=8&p=19229)
- Quote (./posting.php?mode=quote&f=8&p=19229)
- (javascript:void(0);)

Postby **ulao** » Wed Aug 03, 2011 5:11 am

Not sure you come by here or not anymore (could not reach you via email) hopping you could just spare a few moments. I was able to apply the patch but it seems the packets stop after 5 tries?

I did a test with a blink LED, and I get 5 blinks and then it stops. If I take out the sender

while(!usbInterruptIsReady3()) usbPoll();usbSetInterrupt3(XreportBuffer, 20);

then the blink is continuous. If I remove the patched usbdrv, then all is well (blinks away) but of course I only get 8 bytes. I also noticed your not using the new driver, I think the one you have is a bit old and possible parts of different drivers? plus you load all 27 packets but only send 20 (in the comment you state 21-27 have some needed info)? It also looks like you use disconnected/connect yet dont configure USB_CFG_PULLUP_IOPORTNAME , didnt think that was possible? The thing that gets me the most is the desc_device(see code below). Whether or not the svn is correct this is all very confusing. Would it be possible to try out the latest driver or perhaps lend some help my way? Would be so grateful!

How is this in any way an xbox? The packet size, I would assume is to be 0x40 but that depends on how you did the hack. Next the VIP/PID for XBOX is

0x5E, 0x04, //vendor ID
0x02, 0x02, //product ID

in some cases the product ID is something else. The Vendor in your code is 0x03, 0x04(0403 Future Technology Devices International, Ltd)? and the last 4 bits should be 1,0,0,0.

```
PROGMEM char usbDescriptorDeviceXBox[] = {     /* USB device descriptor */
    18,          /* sizeof(usbDescriptorDevice): length of descriptor in bytes
*/
    USBDESCR_DEVICE,       /* descriptor type */
    0x10, 0x01,            /* USB version supported */
    USB_CFG_DEVICE_CLASS,
    USB_CFG_DEVICE_SUBCLASS,
    0,                     /* protocol */
    8,                     /* max packet size */
    /* the following two casts affect the first byte of the constant only, but
     * that's sufficient to avoid a warning with the default values.
```

```
    */
    USB_CFG_FTDI_VENDOR_ID,   /* 2 bytes */
    0x29, 0x8E,               /* product ID (2 bytes) */
    USB_CFG_DEVICE_VERSION, /* 2 bytes */
    1,            /* manufacturer string index */
    2,            /* product string index */
    0,            /* serial number string index */
    1,            /* number of configurations */
};
```

I believe the root of my issues with the hacked firmware is that usbFunctionDescriptor nor usbFunctionSetup are ever called? Again, this works with the un hacked version.

Top

Display posts from previous: | All posts ▾ | Sort by | Post time ▾ | | Ascending ▾ | | Go |

Post Reply
- Email topic
- Print view

9 posts • Page **1** of **1**

Return to "V-USB"

Jump to
- LaunchBar
- •   LaunchBar General
- •   LaunchBar Actions
- •   LaunchBar Search Templates
- Little Snitch
- •   Little Snitch General
- Sharity
- •   Sharity General
- WebYep
- •   WebYep (english)
- •   WebYep (deutsch)
- Other Products
- •   V-USB
- •   V-USB Project Ratings

Who is online

Users browsing this forum: Yahoo [Bot] and 1 guest

Powered by phpBB® Forum Software © phpBB Limited