User Name Password Log in Help Register

□ Remember Me?

What's New? Forum

New Posts FAQ Calendar Community Forum Actions Quick Links Advanced Search

Forum Main Category Technical Support & Questions Adding a new USB type

Forum Rule: Always post complete source code & details to reproduce any issue!

If this is your first visit, be sure to check out the **FAQ** by clicking the link above. You may have to **register** before you can post: click the register link above to proceed. To start viewing messages, select the forum that you want to visit from the selection below.

Results 1 to 8 of 8

Thread: Adding a new USB type

Thread Tools Search Thread Display

06-22-2014, 07:53 PM #1

lhagan

Junior Member

Join Date: Nov 2013

Posts: 5

Adding a new USB type

I'm trying to add a new USB type that combines HID & Raw HID. I think I've worked most of it out from the instructions in "usb_desc.h", but all I get are "XYZ was not declared in this scope" errors when I try to use my new type.

In an attempt to narrow down the problem, I also tried just making an exact duplicate of Raw HID with a different name, and that doesn't work either.

In "boards.txt", I've added:

Code:

```
teensy3.menu.usb.rawhid2.name=Raw HID 2
teensy3.menu.usb.rawhid2.build.define0=-DUSB_R
teensy3.menu.usb.rawhid2.fake_serial=teensy_ga
```

..in "usb_desc.h":

Code:

```
#elif defined(USB_RAWHID2)
  #define VENDOR_ID             0x16C0
  #define PRODUCT_ID            0x0486
  #define RAWHID_USAGE_PAGE     0xFFAB   // rec
  #define RAWHID_USAGE          0x0200   // rec
  #define MANUFACTURER_NAME     {'T','e','e','
  #define MANUFACTURER_NAME_LEN 11
  #define PRODUCT_NAME          {'T','e','e','
  #define PRODUCT_NAME_LEN      18
  #define EP0_SIZE              64
  #define NUM_ENDPOINTS         6
  #define NUM_USB_BUFFERS       12
  #define NUM_INTERFACE         2
  #define RAWHID_INTERFACE      0        // Rawh
  #define RAWHID_TX_ENDPOINT    3
  #define RAWHID_TX_SIZE        64
  #define RAWHID_TX_INTERVAL    1
  #define RAWHID_RX_ENDPOINT    4
  #define RAWHID_RX_SIZE        64
  #define RAWHID_RX_INTERVAL    1
  #define SEREMU_INTERFACE      1        // Ser:
```

```
#define SEREMU_TX_ENDPOINT     1
#define SEREMU_TX_SIZE         64
#define SEREMU_TX_INTERVAL     1
#define SEREMU_RX_ENDPOINT     2
#define SEREMU_RX_SIZE         32
#define SEREMU_RX_INTERVAL     2
#define RAWHID_DESC_OFFSET     (9 + 9)
#define SEREMU_DESC_OFFSET     (9 + 9+9+7+7 +
```

...and in "usb_inst.cpp":

Code:

```
#ifdef USB_RAWHID2
usb_rawhid_class RawHID;
usb_seremu_class Serial;
#endif
```

Here's the full error message when trying to build the Raw HID Basic example code:

Code:

```
/Users/lhagan/Applications/Arduino.app/Content
Basic.pde: In function 'void setup()':
Basic:15: error: 'Serial' was not declared in
Basic.pde: In function 'void loop()':
Basic:29: error: 'RawHID' was not declared in
Basic:34: error: 'Serial' was not declared in
Basic:63: error: 'Serial' was not declared in
Basic:67: error: 'Serial' was not declared in
```

Of course, the example code builds just fine if I select "Raw HID" instead of "Raw HID 2". I feel like I must be missing something really simple, but I just can't seem to find it. Any suggestions would be much appreciated!

Setup: Arduino 1.0.5 & Teensyduino 1.19 on a Mac, Teensy 3

Reply With Quote

07-06-2014, 11:52 PM #2

lhagan

Junior Member

Join Date:	Nov 2013
Posts:	5

OK, so no excuse for missing a previous post on the exact same issue: http://forum.pjrc.com/threads/24431-...ll=1#post36841

However, that wasn't enough to get everything working, so I thought it'd help to provide a complete step-by-step example to hopefully save others some time in the future.

In "usb_desc.h", copy an existing type that contains one or more of the USB types you want. Then, start modifying it as needed:

1. Customize PRODUCT_NAME to whatever you want.
2. Change PRODUCT_NAME_LEN to match the number of characters in your new PRODUCT_NAME.
3. Copy the relevant lines from another USB type. For example, to add Keyboard, I copied the following from the USB_HID type:

   Code:

```
  #define KEYBOARD_INTERFACE    0
  #define KEYBOARD_ENDPOINT     3
  #define KEYBOARD_SIZE         8
  #define KEYBOARD_INTERVAL     1
```

4. Edit the XYZ_INTERFACE number. There's one for each interface, it starts at 0, and needs to be unique for each interface. In the Raw HID USB type, RAWHID_INTERFACE is 0 and SEREMU_INTERFACE is 1, so I made KEYBOARD_INTERFACE 2. It's simplest to just add your new interface(s) to the end.
5. Copy in the XYZ_DESC_OFFSET line from the other USB type and edit to the correct value. XYZ_DESC_OFFSET represents a size, always starts and ends with 9, and has the sum of the bLengths for all interfaces that precede it the middle. For example, the offset of interface2 would be (9 + interface0_length + interface1_length + 9). You can find the bLengths in usb_desc.c, but I've compiled them here for reference:

   Code:

   ```
   keyboard    9+9+7
   mouse       9+9+7
   seremu      9+9+7+7
   joystick    9+9+7
   flightsim   9+9+7+7
   rawhid      9+9+7+7
   midi        9+7+6+6+9+9+9+5+9+5
   serial      9+5+5+4+5+7+9+7+7
   ```

   In this example, RAWHID is the first interface, so RAWHID_DESC_OFFSET is just (9 + 9). SEREMU is the next interface, so its OFFSET adds the bLengths from the previous interface (RAWHID), thus SEREMU_DESC_OFFSET is (9 + 9+9+7+7 + 9). Keyboard is the last interface, so KEYBOARD_DESC_OFFSET has the bLengths from both RAWHID and SEREMU: (9 + 9+9+7+7 + 9+9+7+7 + 9)
6. Edit the CONFIG_DESC_SIZE. The approach here is similar to the XYZ_OFFSETs above, it does not have an extra 9 at the end. It starts with a 9 and includes the bLengths for all interfaces. In this example, it's (9 + rawhid_length + seremu_length + keyboard_length), or (9 + 9+9+7+7 + 9+9+7+7 + 9+9+7)
7. Add endpoints if needed. The original Raw USB type has 4 endpoints, one transmit (TX) and one receive (RX) each for RAWHID and SEREMU. Since I've added KEYBOARD, a 5th endpoint is needed (just a one TX as there is no RX for keyboard).

   Code:

   ```
   #define ENDPOINT5_CONFIG        END
   ```

8. Update the NUM_ENDPOINTS to match the number of endpoints you have (i.e. 5 in this

case).

9. Modify the endpoint number(s) for your new interface(s) to match the endpoint(s) added above, i.e.

   Code:

   ```
   #define KEYBOARD_ENDPOINT     5
   ```

10. Increase the NUM_USB_BUFFERS. I'm not actually sure if/when this is necessary, but this value varies from type to type. I made mine equal to the highest value (30). If anyone can provide some better guidance here, I'd love to update these instructions to be more accurate.

Here's the full example from my usb_desc.h:

Code:

```
  #define NUM_ENDPOINTS         5
  #define NUM_USB_BUFFERS           30
  #define NUM_INTERFACE             3
  #define RAWHID_INTERFACE      0       // Raw
  #define RAWHID_TX_ENDPOINT    3
  #define RAWHID_TX_SIZE        64
  #define RAWHID_TX_INTERVAL    1
  #define RAWHID_RX_ENDPOINT    4
  #define RAWHID_RX_SIZE        64
  #define RAWHID_RX_INTERVAL    1
  #define SEREMU_INTERFACE      1       // Ser
  #define SEREMU_TX_ENDPOINT    1
  #define SEREMU_TX_SIZE        64
  #define SEREMU_TX_INTERVAL    1
  #define SEREMU_RX_ENDPOINT    2
  #define SEREMU_RX_SIZE        32
  #define SEREMU_RX_INTERVAL    2
  #define KEYBOARD_INTERFACE    2       // Key
  #define KEYBOARD_ENDPOINT     5
  #define KEYBOARD_SIZE         8
  #define KEYBOARD_INTERVAL     1
  #define RAWHID_DESC_OFFSET    (9 + 9)
  #define SEREMU_DESC_OFFSET    (9 + 9+9+7+7 +
  #define KEYBOARD_DESC_OFFSET  (9 + 9+9+7+7 +
  #define CONFIG_DESC_SIZE      (9 + 9+9+7+7 +
  #define ENDPOINT1_CONFIG      ENDPOINT_TRANS
  #define ENDPOINT2_CONFIG      ENDPOINT_RECEI
  #define ENDPOINT3_CONFIG      ENDPOINT_TRANS
  #define ENDPOINT4_CONFIG      ENDPOINT_RECEI
  #define ENDPOINT5_CONFIG      ENDPOINT_TRANS
```

In "usb_inst.cpp", add a conditional statement to create instances of the C++ objects for your new type:

Code:

```
#ifdef USB_KEYBOARD_RAWHID
usb_keyboard_class Keyboard;
usb_rawhid_class RawHID;
usb_seremu_class Serial;
#endif
```

So you can select your new USB type from the Arduino Tools menu, add it to "boards.txt":

Code:

```
teensy3.menu.usb.keyboardrawhid.name=Keyboard
teensy3.menu.usb.keyboardrawhid.build.define0=
teensy3.menu.usb.keyboardrawhid.fake_serial=te
```

The teensy3..define0 value needs to match the #ifdef you put in "usb_inst.cpp".

At this point, you're almost ready to build, but you'll get a bunch of "XYZ was not declared in this scope" errors. To fix this, you need to add your new USB type to the "#if defined"s of the classes you're using (in "usb_xyz.h"). In this example, we need rawhid, seremu, and keyboard:

In "usb_rawhid.h":

Code:

```
#if defined(USB_RAWHID) || defined(USB_KEYBOAR
```

In "usb_seremu.h":

Code:

```
#if defined(USB_HID) || defined(USB_MIDI) || d
```

In "usb_keyboard.h":

Code:

```
#if defined(USB_HID) || defined(USB_SERIAL_HID
```

That should do it!

Reply With Quote

09-28-2014, 01:06 PM #3

Experimentalist

Senior Member

Join Date: Nov 2012
Location: Chipping Norton, UK
Posts: 278

Hi, thanks for this. I just got back to my project and had completely forgotten how I achieved this. I had deferred the RAW HID interface and lost all my updates later when I upgraded Arduino and Teensyduino. It was an uphill struggle the first time, this time you made it easy for me :0)

thanks
Ex

Reply With Quote

08-28-2015, 02:42 PM #4

SteelWolf

Junior Member

Join Date: Feb 2013
Posts: 18

I'll book mark this thread as i'll probably be renaming my Teensy 3.0 in windows setting to MechPit.

Reply With Quote

04-11-2017, 01:14 PM #5

dirkvl

Junior Member

Join Date: Apr 2017
Posts: 2

I tried to do the same, but adding Serial to 'keyboard mouse and touchscreen'

in boards.txt:

Code:

```
teensy31.menu.usb.hidtouch=Keyboard + Mouse +
teensy31.menu.usb.serialhid.build.usbtype=USB_
```

in usb_desc.h:

Code:

```
  #elif defined(USB_HID_TOUCHSCREEN_SERIAL)
  #define VENDOR_ID             0x16C0
  #define PRODUCT_ID            0x0481
  #define MANUFACTURER_NAME     {'T','e','e','
  #define MANUFACTURER_NAME_LEN 11
  #define PRODUCT_NAME          {'S','e','r','
  #define PRODUCT_NAME_LEN              33
  #define EP0_SIZE
  #define NUM_ENDPOINTS         6
  #define NUM_USB_BUFFERS               20
  #define NUM_INTERFACE                 5
  #define CDC_DATA_INTERFACE    2       // Ser
  #define CDC_ACM_ENDPOINT              2
  #define CDC_RX_ENDPOINT       3
  #define CDC_TX_ENDPOINT       4
  #define CDC_ACM_SIZE          16
  #define CDC_RX_SIZE           64
  #define CDC_TX_SIZE           64
  #define KEYBOARD_INTERFACE    0       // Key
  #define KEYBOARD_ENDPOINT     3
  #define KEYBOARD_SIZE         8
  #define KEYBOARD_INTERVAL     1
  #define KEYMEDIA_INTERFACE    3       // Key
  #define KEYMEDIA_ENDPOINT     4
  #define KEYMEDIA_SIZE         8
  #define KEYMEDIA_INTERVAL     4
  #define MOUSE_INTERFACE       1       // Mou
  #define MOUSE_ENDPOINT        6
  #define MOUSE_SIZE            8
  #define MOUSE_INTERVAL        2
```

in usb_inst.cpp:

Code:

```
#ifdef USB_HID_TOUCHSCREEN_SERIAL
usb_keyboard_class Keyboard;
usb_mouse_class Mouse;
usb_serial_class Serial;
usb_touch_class Touch;
#endif
```

It shows in the menu, compiles and uploads, but Serial is not added, it keeps showing 'emulated serial'...

In my application I send coordinates to a Teensy3.2 over Serial and then let it emulate touch events. This works when I select 'All of the above' as usb type, but then my sound is gone 🙂

Last edited by dirkvl; 04-11-2017 at 01:19 PM.

Reply With Quote

04-11-2017, 01:41 PM #6

tni

Senior Member

Join Date: Jan 2013

Posts: 843

> Originally Posted by **dirkvl**
>
> I tried to do the same, but adding Serial to 'keyboard mouse and touchscreen'

...

Your endpoint config is completely wrong.

#define NUM_ENDPOINTS 6
but later you have 7.

#define CDC_RX_ENDPOINT 3
...
#define KEYBOARD_ENDPOINT 3

They must be unique.

#define CDC_RX_ENDPOINT 3
#define ENDPOINT3_CONFIG ENDPOINT_TRANSIMIT_ONLY
You can't receive on a transmit endpoint.

Reply With Quote

04-11-2017, 01:56 PM #7

dirkvl

Junior Member
Join Date: Apr 2017
Posts: 2

this should be better:

Code:

```
#elif defined(USB_HID_TOUCHSCREEN_SERIAL)
#define VENDOR_ID                 0x16C0
#define PRODUCT_ID                0x0481
#define MANUFACTURER_NAME         {'T','e','e','
#define MANUFACTURER_NAME_LEN 11
#define PRODUCT_NAME              {'S','e','r','
#define PRODUCT_NAME_LEN                  33
#define EP0_SIZE
#define NUM_ENDPOINTS                     7
#define NUM_USB_BUFFERS                   30
#define NUM_INTERFACE                     6
#define CDC_IAD_DESCRIPTOR        1
#define CDC_STATUS_INTERFACE      0
#define CDC_DATA_INTERFACE        1       // Ser
#define CDC_ACM_ENDPOINT                  2
#define CDC_RX_ENDPOINT           3
#define CDC_TX_ENDPOINT           4
#define CDC_ACM_SIZE              16
#define CDC_RX_SIZE               64
#define CDC_TX_SIZE               64
#define KEYBOARD_INTERFACE        2       // Key
#define KEYBOARD_ENDPOINT         1
#define KEYBOARD_SIZE             8
#define KEYBOARD_INTERVAL         1
#define KEYMEDIA_INTERFACE        5       // Key
#define KEYMEDIA_ENDPOINT         7
#define KEYMEDIA_SIZE             8
#define KEYMEDIA_INTERVAL         4
#define MOUSE_INTERFACE           3       // Mou
#define MOUSE_ENDPOINT            5
```

but this still gives me emulated serial, where do i add this?

Last edited by dirkvl; 04-11-2017 at 04:25 PM.

Reply With Quote

04-11-2017, 05:53 PM #8

PaulStoffregen

Senior Member

Originally Posted by **dirkvl**

Join Date: Nov 2012
Posts: 23,258

> It shows in the menu, compiles and uploads, but Serial is not added, it keeps showing 'emulated serial'...

The "fake_serial=teensy_gateway" key in the menu lines of boards.txt controls this. Find the group of menu lines for your new USB type and delete the fake_serial line. This line also controls whether the serial monitor tries to access Teensy as a real serial device, or through a localhost connection to the HID gateway program.

Reply With Quote

Quick Navigation **Technical Support & Questions** **Top**

« Previous Thread | Next Thread »

Posting Permissions

You may not post new threads	**BB code** is On
You may not post replies	**Smilies** are On
You may not post attachments	**[IMG]** code is On
You may not edit your posts	**[VIDEO]** code is On
	HTML code is Off
	Forum Rules

-- Default Style PJRC Electronic Projects Archive Top

All times are GMT. The time now is 07:29 PM.