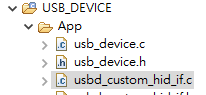
USB HID Programming

Usb HID report descriptor



CUSTOM\_HID\_ReportDesc\_FS

\_\_ALIGN\_BEGIN static uint8\_t CUSTOM\_HID\_ReportDesc\_FS[USBD\_CUSTOM\_HID\_REPORT\_DESC\_SIZE] \_\_ALIGN\_END =

{

  /\* USER CODE BEGIN 0 \*/

    0x05, 0x01,                    // USAGE\_PAGE (Generic Desktop)

    0x09, 0x00,                    // USAGE (Undefined)

    0xa1, 0x01,                    // COLLECTION (Application)

    0x09, 0x00,                    //   USAGE (Undefined)

    0x15, 0x00,                    //   LOGICAL\_MINIMUM (0)

    0x26, 0xff, 0x00,              //   LOGICAL\_MAXIMUM (255)

    0x95, 0x40,                    //   REPORT\_COUNT (64)

    0x75, 0x08,                    //   REPORT\_SIZE (8)

    0x81, 0x02,                    //   INPUT (Data,Var,Abs)

    0x09, 0x00,                    //   USAGE (Undefined)

    0x15, 0x00,                    //   LOGICAL\_MINIMUM (0)

    0x26, 0xff, 0x00,              //   LOGICAL\_MAXIMUM (255)

    0x95, 0x40,                    //   REPORT\_COUNT (64)

    0x75, 0x08,                    //   REPORT\_SIZE (8)

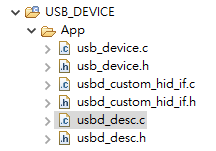
    0x91, 0x02,                    //   OUTPUT (Data,Var,Abs)

  /\* USER CODE END 0 \*/

    0xC0    /\*     END\_COLLECTION                \*/

};

Modify VID/PID



#define USBD\_VID     1155

#define USBD\_LANGID\_STRING     1033

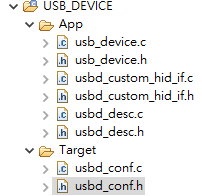
#define USBD\_MANUFACTURER\_STRING     "STMicroelectronics"

#define USBD\_PID\_FS     22352

#define USBD\_PRODUCT\_STRING\_FS     "STM32 Custom Human interface"

#define USBD\_CONFIGURATION\_STRING\_FS     "Custom HID Config"

#define USBD\_INTERFACE\_STRING\_FS     "Custom HID Interface"

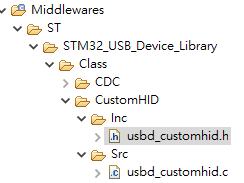


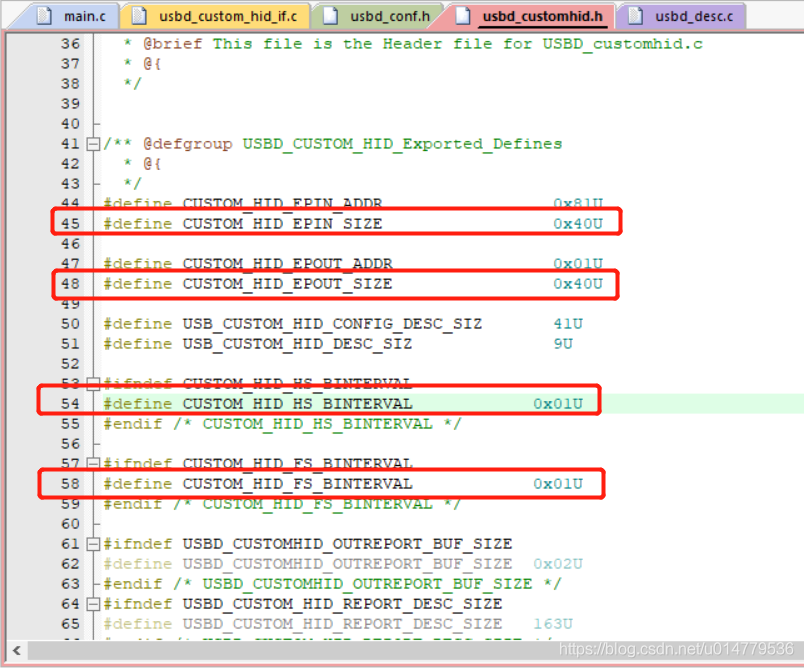
/\*---------- -----------\*/

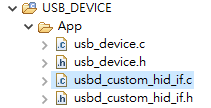
#define USBD\_CUSTOMHID\_OUTREPORT\_BUF\_SIZE     64U

/\*---------- -----------\*/

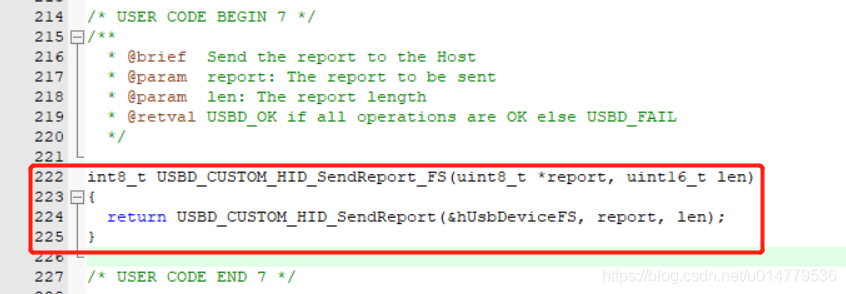
#define USBD\_CUSTOM\_HID\_REPORT\_DESC\_SIZE     33U

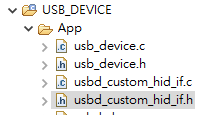




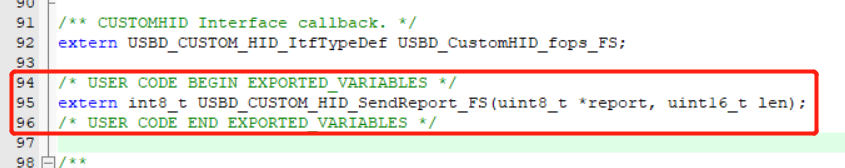


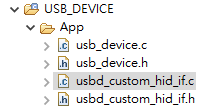
Modify **USBD\_CUSTOM\_HID\_SendReport\_FS to non-static**





Declare **USBD\_CUSTOM\_HID\_SendReport\_FS** as extern





*/\* USER CODE BEGIN INCLUDE \*/*

*#include <stdbool.h>*

*/\* USER CODE END INCLUDE \*/*

*/\* USER CODE BEGIN PV \*/*

*/\* Private variables ---------------------------------------------------------\*/*

*uint8\_t hid\_out\_report[64];*

*uint8\_t hid\_in\_report[64];*

*bool is\_hid\_out\_empty = true;*

*/\* USER CODE END PV \*/*

*static int8\_t CUSTOM\_HID\_OutEvent\_FS(uint8\_t event\_idx, uint8\_t state)*

*{*

*/\* USER CODE BEGIN 6 \*/*

*UNUSED(event\_idx);*

*UNUSED(state);*

*USBD\_CUSTOM\_HID\_HandleTypeDef \*hhid =*

*(USBD\_CUSTOM\_HID\_HandleTypeDef\*) (hUsbDeviceFS.pClassData);*

*memcpy(hid\_out\_report, hhid->Report\_buf, 64);*

*is\_hid\_out\_empty = false;*

*/\* Start next USB packet transfer once data processing is completed \*/*

*USBD\_CUSTOM\_HID\_ReceivePacket(&hUsbDeviceFS);*

*return (USBD\_OK);*

*/\* USER CODE END 6 \*/*

*}*

Use the following to other program

#include <stdbool.h>

#include "string.h"

#include "usbd\_customhid.h"

extern uint8\_t hid\_in\_report[64];

extern uint8\_t hid\_out\_report[64];

extern bool is\_hid\_out\_empty;

extern USBD\_HandleTypeDef hUsbDeviceFS;

extern int8\_t USBD\_CUSTOM\_HID\_SendReport\_FS(uint8\_t \*report, uint16\_t len);

in the main()

if (is\_hid\_out\_empty == false)

{

  memcpy(hid\_in\_report, hid\_out\_report, 64); // copy from out => in

  for (int i = 0; i < 64; i++)

  {

    hid\_in\_report[i] = hid\_in\_report[i];

  }

  USBD\_CUSTOM\_HID\_SendReport\_FS(hid\_in\_report, 64);

  is\_hid\_out\_empty = true;

}

Reference

HID

1. <http://213style.blogspot.com/2013/09/usb-human-interface-device.html>

PC

1. <https://www.codeproject.com/Articles/1244702/How-to-Communicate-with-its-USB-Devices-using-HID>

STM32

1. <https://blog.csdn.net/u014779536/article/details/104512992?utm_medium=distribute.pc_relevant_download.none-task-blog-baidujs-6.nonecase&depth_1-utm_source=distribute.pc_relevant_download.none-task-blog-baidujs-6.nonecase>
2. <https://blog.csdn.net/killf_123/article/details/107453942>
3. <https://www.twblogs.net/a/5c7124d8bd9eee68dc3f228c>
4. [HID API](https://blog.csdn.net/u010875635/article/details/73321066)
5. [USB HID and C Sharp](http://ahidlib.com/pages/programming_csharp.php?lang=en)
6. [HIDSharp](https://www.zer7.com/software/hidsharp)

aHID.dll

1. <http://ahidlib.com/pages/index.php?lang=en>
2. <http://ahidlib.com/pages/programming_csharp.php?lang=en>
3. <http://embedded24.net/>