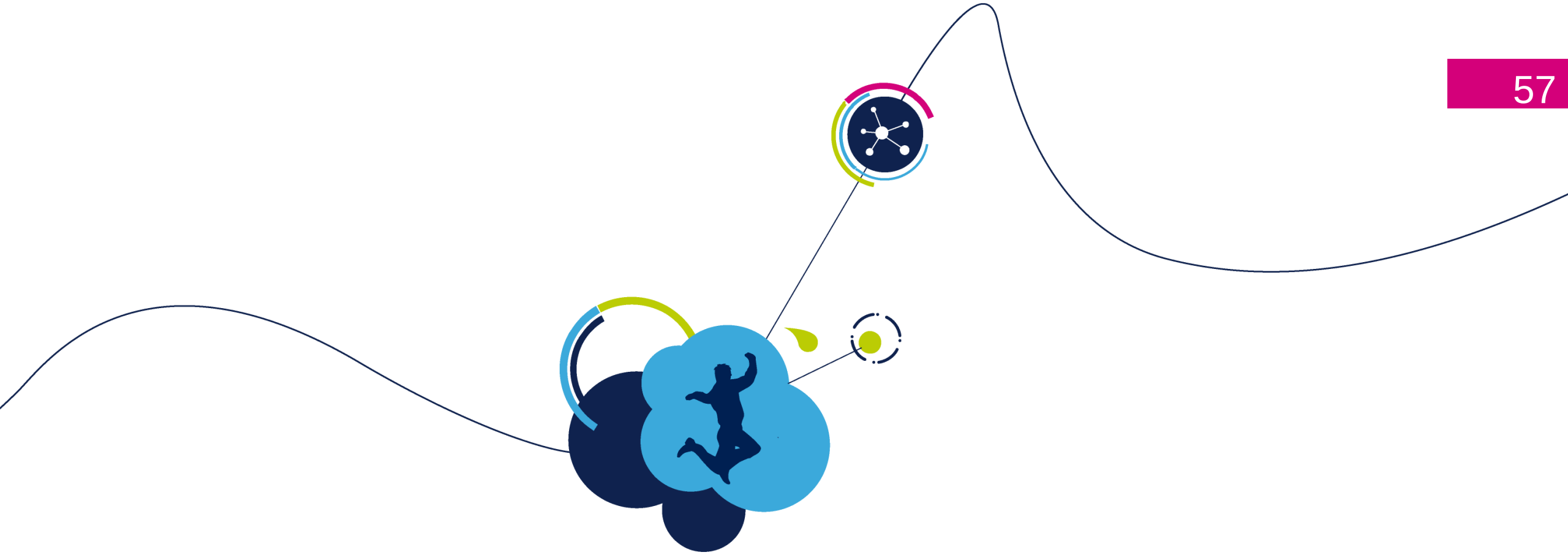
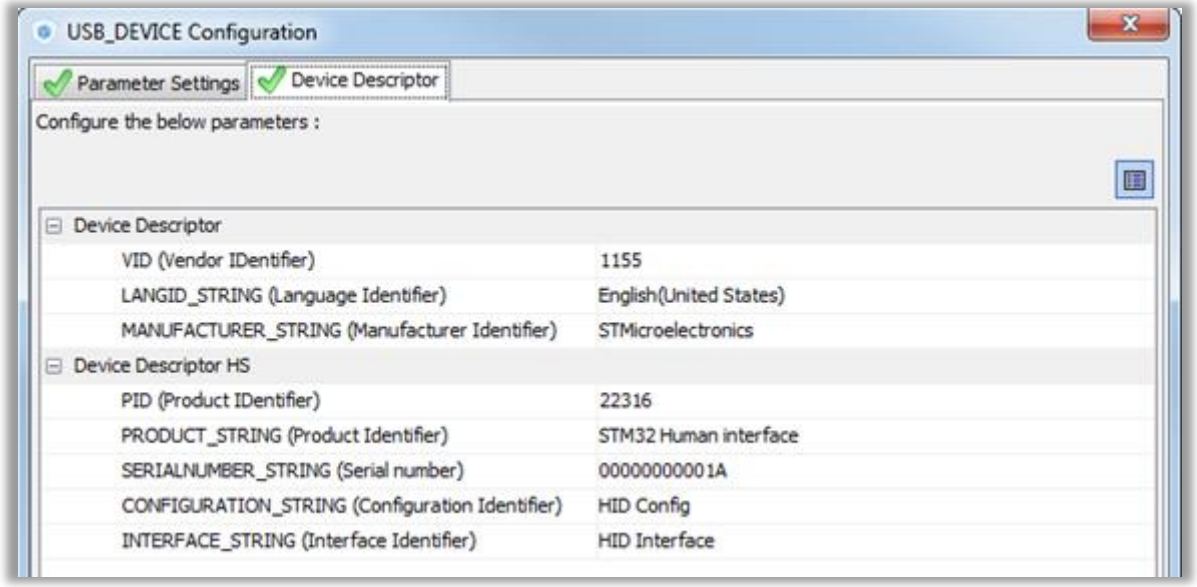


USB HID keyboard lab

USB HID keyboard lab

- Use previous HID_device CubeMX project
- In CubeMX change PID to 22316 and save the project with different name
- And regenerate code



USB HID keyboard lab

- In usbd_hid.h change size of report descriptor to 187 and EP size

```
#define HID_EPIN_SIZE                0x08
```

```
#define HID_MOUSE_REPORT_DESC_SIZE    187
```

- In usbd_hid.c change the protocol interface to keyboard

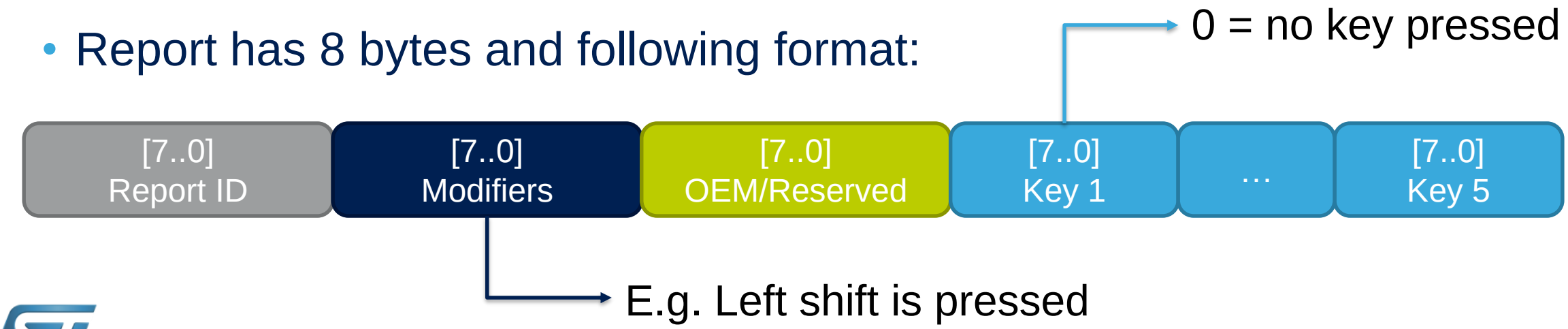
```
/*************** Descriptor of Joystick Mouse interface ***************/
 /* 09 */
 0x09,         /*bLength: Interface Descriptor size*/
 USB_DESC_TYPE_INTERFACE,/*bDescriptorType: Interface descriptor type*/
 0x00,         /*bInterfaceNumber: Number of Interface*/
 0x00,         /*bAlternateSetting: Alternate setting*/
 0x01,         /*bNumEndpoints*/
 0x03,         /*bInterfaceClass: HID*/
 0x01,         /*bInterfaceSubClass : 1=BOOT, 0=no boot*/
 0x01,         /*nInterfaceProtocol : 0=none, 1=keyboard, 2=mouse*/
 0,            /*iInterface: Index of string descriptor*/
 /******************** Descriptor of Joystick Mouse HID ********************/
```

USB HID keyboard lab

- Change HID report descriptor
  - Number of changes is quite big, please use attached file

HID_keyboard_report_descriptor.txt

- Report has 8 bytes and following format:



USB HID keyboard lab

- Changes in main.c:
  - Add USB handler extern and includes
  - Fill buffer with keyboard report

```
/* USER CODE BEGIN PV */

extern USBD_HandleTypeDef hUsbDeviceFS;

/* USER CODE END PV */

/* USER CODE BEGIN Includes */

#include "usbd_hid.h"
/* USER CODE END Includes */
/* USER CODE BEGIN PFP */
uint8_t buffer[8];
/* USER CODE END PFP */
 /* USER CODE BEGIN 2 */
  buffer[0]=1;//reportID
  buffer[1]=0;//modifier
  buffer[2]=0;//OEM
  buffer[3]=0x04;//keycode data - a
  buffer[4]=0;//keycode data
  buffer[5]=0;//keycode data
  buffer[6]=0;//keycode data
  buffer[7]=0;//keycode data
  /* USER CODE END 2 */
```

USB HID keyboard lab

- On button press, send HID report with keycode – demo is ready

```
/* USER CODE END 2 */
/* Infinite loop */
/* USER CODE BEGIN WHILE */

 while (1)
  {
  /* USER CODE END WHILE */
    if(HAL_GPIO_ReadPin(GPIOC,GPIO_PIN_13)==GPIO_PIN_SET)
    {

      buffer[3]=0x4E;//keycode data - PgDwn press
      USBD_HID_SendReport(&hUsbDeviceFS,buffer,8);
      HAL_Delay(100);
      buffer[3]=0x0;//keycode data - PgDwn release
      USBD_HID_SendReport(&hUsbDeviceFS,buffer,8);
      HAL_Delay(100);
     }
  }
```

- Some links about USB keyboard scan codes:
  - http://files.microscan.com/helpfiles/ms4_help_file/ms-4_help-02-46.html
  - https://www.win.tue.nl/~aeb/linux/kbd/scancodes-14.html