

На примере сброса бита CTR_TX

Для начала вот дефайны:

```
#define USB_EPnR_CTR_RX          0x00008000
#define USB_EPnR_DTOG_RX        0x00004000
#define USB_EPnR_STAT_RX        0x00003000
#define USB_EPnR_STAT_RX_0      0x00001000
#define USB_EPnR_STAT_RX_1      0x00002000
#define USB_EPnR_SETUP          0x00000800
#define USB_EPnR_EP_TYPE        0x00000600
#define USB_EPnR_EP_TYPE_0      0x00000200
#define USB_EPnR_EP_TYPE_1      0x00000400
#define USB_EPnR_EP_KIND        0x00000100
#define USB_EPnR_CTR_TX         0x00000080
#define USB_EPnR_DTOG_TX        0x00000040
#define USB_EPnR_STAT_TX        0x00000030
#define USB_EPnR_STAT_TX_0      0x00000010
#define USB_EPnR_STAT_TX_1      0x00000020
#define USB_EPnR_EA             0x0000000F
```

Чтобы сбросить бит CTR_TX:

- 1) сохраняем значение EPnR во временную переменную
`endpoints[n].status = USB -> EPnR[n];`
- 2) используем макрос
`endpoints[n].status = CLEAR_CTR_TX(status);`

По шагам, допустим у нас было:

CTR_RX	DTOG_RX	STAT_RX [1:0]	SETUP	EP_TYPE [1:0]	EP_KIND	CTR_TX	DTOG_TX	STAT_TX [1:0]	EA[3:0]
rc_w0	t	t	r	rw	rw	rc_w0	t	t	rw
1	0	1 1	0	1 1	0	1	1	1 0	0 0 0 1

Макрос выглядит следующим образом (макросы находятся в файле `usb_lib.h`):

```
#define CLEAR_CTR_TX(R) (R & (~ (USB_EPnR_CTR_TX |
                                USB_EPnR_STAT_RX |
                                USB_EPnR_STAT_TX |
                                USB_EPnR_DTOG_RX |
                                USB_EPnR_DTOG_TX)))
```

Заменяем **R** содержимым `endpoints[n].status`, то есть **0b1011011011100001** или **0xB6E1**, и собираем второй аргумент из дефайнов (16 старших нулевых бит откину за ненужностью):
`0x0080 | 0x3000 | 0x0030 | 0x4000 | 0x0040 = 0x70F0`

Теперь выглядит так:

```
endpoints[n].status = 0xB6E1 & (~0x70F0);
endpoints[n].status = 0xB6E1 & 0x8F0F;
endpoints[n].status = 0x8601.
```

3) Производим запись в регистр

```
USB -> EPnR[n] = endpoints[n].status;
```

	CTR_RX	DTO_G_RX	STAT_RX[1:0]		SETUP	EP_TYPE[1:0]		EP_KIND	CTR_TX	DTOG_TX	STAT_TX[1:0]		EA[3:0]			
Доступ	rc_w0	t	t		r	rw		rw	rc_w0	t	t		rw			
Было	1	0	1	1	0	1	1	0	1	1	1	0	0	0	0	1
Записываем	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1
Стало	1	0	1	1	0	1	1	0	0	1	1	0	0	0	0	1

Как видно сбросился только бит CTR_TX. Если не вник, смотри на тип доступа:

rc_w0 - при записи 0, записывает 0. Запись 1 - игнорируется;

t - при записи 1 инвертирует бит. Запись 0 - игнорируется;

r - только чтение, запись игнорируется;

rw - обычный доступ, 1 запишет 1, 0 запишет 0.

В моем коде это строки (файл usb_lib.c) - 203, 209, 210. Только там я сбрасываю сразу 2 бита CTR_RX и CTR_TX (макрос другой, но все аналогично).