

Table des matières

<u>I) COMMENT ECRIRE PLUSIEURS BITS EN MEME TEMPS ?</u>	<u>2</u>
I-1) ODR	2
I-2) BSRR	2
I-3) BRR	3

I) Comment écrire plusieurs bits en même temps ?

Il y a 3 registres de contrôle qui affectent les sorties des broches des GPIO.

I-1) ODR

L'écriture dans ODR définit les 16 broches à la fois, par exemple, `GPIOB->ODR = 0xF00F` définit :

- à **1** les broches **B0 à B3** et **B12 à B15**
- à **0** les broches **B4 à B11**.

Et ceci quel que soit l'état précédent du GPIO.

On peut écrire `GPIOB->ODR |= (1 << 4)` pour mettre la broche GPIOB4 à **1**.

On peut écrire `GPIOB->ODR &= ~(1 << 4)` pour mettre la broche GPIOB4 à **0**.

I-2) BSRR

BSRR est un registre de 32 bits qui permet de spécifier en une seule fois les bits :

- à mettre à 1 ;
- à mettre à 0 ;
- à ne pas modifier.

Les 16 bits de poids faibles du registre BSRR **permettent d'indiquer les bits à mettre à 1**. Mettre à **1** l'un de ces bits a pour conséquence **de mettre à 1** le bit de même rang dans le registre ODR.

Les 16 bits de poids forts du registre BSRR **permettent d'indiquer les bits à mettre à 0**. Mettre à **1** l'un de ces bits a pour conséquence **de mettre à 0** le bit de même rang dans le registre ODR.

Lorsqu'un bit de rang x **est à zéro à la fois** dans les 16 bits de poids faibles **et** dans les bits de poids forts, c'est que bit correspondant dans le registre ODR doit rester inchangé !

Lorsqu'un bit de rang x **est à un à la fois** dans les 16 bits de poids faibles **et** dans les bits de poids forts, c'est une incohérence, mais dans ce cas la priorité est donnée aux 16 bits de poids faibles et donc le bit correspondant du registre ODR, sera mis à **1** !

Exemple, `GPIOC->BSRR = 0x000701E0`

<- POIDS FORTS >	<- POIDS FAIBLES >
0000 0000 0000 0111	0000 0001 1110 0000

Définit les broches C5 à C8 à 1,
Définit les broches C0 à C2 à 0,
et laisse tous les autres bits inchangés.

Les poids faibles sont prioritaires, par exemple si l'on modifie légèrement l'exemple précédent comme ceci : `GPIOC->BSRR = 0x000701E1`

<- POIDS FORTS >	<- POIDS FAIBLES >
0000 0000 0000 011 1	0000 0001 1110 000 1

Alors c'est le bit 0 des poids faibles qui aura priorité sur le bit 0 des poids forts, un bit ne pouvant pas être à la fois à 1 et à 0, il fallait bien choisir...

I-3) BRR

L'écriture dans BRR revient à écrire dans les poids forts de BSRR, c'est-à-dire que $\text{GPIOx} \rightarrow \text{BRR} = x$ est équivalent à $\text{GPIOx} \rightarrow \text{BSRR} = (x \ll 16)$.