

# Corrigé des exercices du 14 mars 2016

## Entrées-sorties des microcontrôleurs

1) Ecrivez les instructions qui mettent les pattes dans l'état suivant :

- P1.0 en sortie, à l'état 1
- P1.6 en sortie, à l'état 0
- P1.3 en entrée (sans résistance de tirage).

```
P1DIR |= (1<<0); P1OUT |= (1<<0);  
P1DIR |= (1<<6); P1OUT &=~(1<<6);  
P1DIR &=~(1<<3);
```

2) Réécrivez le programme qui fait clignoter la LED rouge du LaunchPad, sans utiliser `pinMode` et `digitalWrite`.

Mettez l'instruction d'initialisation de la sortie dans `setup()` et programmez le clignotement dans `loop()`, en utilisant la procédure `delay` pour l'attente.

```
void setup () {  
    P1DIR |= (1<<0);  
}  
void loop () {  
    P1OUT |= (1<<0);  
    delay (500);  
    P1OUT &=~(1<<0);  
    delay (500);  
}
```

3) Simplifiez votre programme en utilisant la fonction logique OU-exclusif.

L'opérateur en C est le circonflexe : ^

En vous aidant de la table de vérité du OU-exclusif, regardez ce que produit l'instruction :

```
P1OUT ^= (1<<0);
```

```
void loop () {  
    P1OUT ^= (1<<0);  
    delay (500);  
}
```

4) Ecrivez les instructions pour mettre la patte P1.3 en entrée avec une résistance de tirage vers le haut (pull-up).

```
P1DIR &=~(1<<3);  
P1REN |= (1<<3);  
P1OUT |= (1<<3);
```

5) (*optionnel*) Réécrivez (encore une fois...) le programme qui fait clignoter la LED rouge du

LaunchPad, mais cette fois sans utiliser aucune procédure Arduino (pas de delay, ni setup, ni loop).

Le programme aura la structure suivante :

```
int main () {
    WDTCTL = WDTPW + WTDHOLD; // désactivation du watchdog
    //... placez ici les initialisations
    while (1) { // boucle infinie
        //... clignotement
    }
}
```

Vous devez écrire vous-même les instruction (ou une procédure) pour l'attente, telle que :

```
volatile int i;
for (i=0; i<10000; i++) {}
```

```
int main () {
    volatile int i;
    WDTCTL = WDTPW + WTDHOLD ; // désactivation du watchdog
    P1DIR |= (1<<0);
    while (1) { // boucle infinie
        P1OUT ^= (1<<0);
        for (i=0; i<10000; i++) {}
    }
}
```