

DIGITAL BRIDGE

Leçon 5

But : l'instruction "**FOR**" LOOP permet d'exécuter des opérations qui se répètent sans devoir écrire plusieurs fois la même ligne de code. Contrairement à la fonction void loop () qui elle opère en continu (non-stop), l'instruction **For** loop exécute une loop un nombre limité de fois.

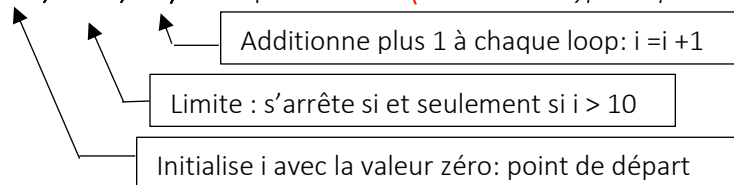
Exemple de code utilisant l'instruction **FOR** loop

```
void setup () {  
  int i;  
  
  Serial.begin(9600);  
  
  for (i = 0; i < 10; i++) {  
    Serial.print("i = ");  
    Serial.println(i);  
  }  
}  
  
void loop() {  
}
```

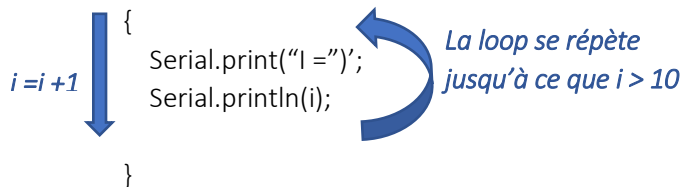
Notez que nous créons ici une loop dans la fonction void setup ()

Explication ligne par ligne:

- **int i;** déclaration de la variable i en tant que nombre entier (int, Integer en anglais)
- **Serial.begin(9600);** Commence la communication avec la console (moniteur de série) à un débit de 9600 bauds
- **for (i = 0; i < 10; i++)** entre parenthèses (*condition et type d'opération*)



- Debut de la loop avec { **code à exécuter jusqu'à ce que i soit plus grand que 10** }. Notez les lignes du code à exécuter par l'instruction for loop sont comprises entre les accolades { }



i	i=i+1	
0	i=0 + 1	1
1	i=1 + 1	2
2	i=2 + 1	3
...
10	i=10 + 1	11

STOP i > 10

```
{ Serial.print("i =");  
  Serial.println(i); }
```

L'instruction **Serial.print()** imprime à l'écran sur le moniteur de série (console) *i* = juste à côté de la valeur de la variable *i*. **Println** fait le renvoi à la ligne suivante (équivalent à "entrée", "enter", "return", ou "retour" sur le clavier de votre ordinateur).

Exercice n° 1 :

Code Arduino: Ecrivez le code de l'exemple de cette leçon dans un croquis Arduino et observez le résultat dans le moniteur de série. Ensuite modifiez-le en écrivant la ligne suivante : `for (i = 10 ; i > 0 ; i --)`. Que remarquez-vous ?

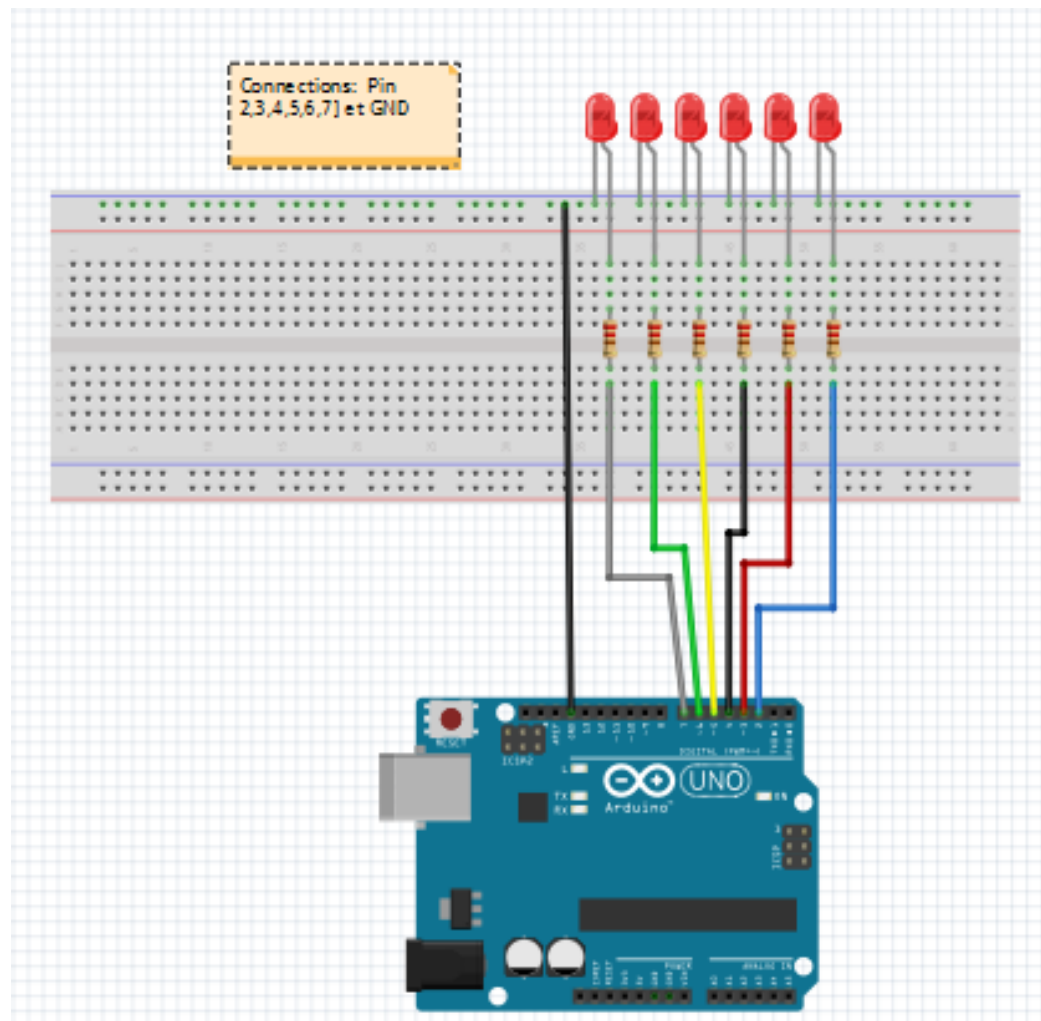
Utilisez les instructions apprises ici pour réécrire le code de la leçon 2

Exercice n° 2 :

Diagramme de raccordement :

Materiel:

- (6) resistance 220 ohms
- (6) LED (bleu, jaune, et rouge)
- (7) câbles
- (1) planche prototype



Code Arduino:

```
int timer = 100; // Delai ou temps de pause

int pins[] = { 2, 3, 4, 5, 6, 7 }; // array des points de raccordement (2 a l'index 0, 3 a l'index 1, 7 a l'index 5)

int num_pins = 6; // nombre total de points de raccordement (6)

void setup()
{
    int i;
    for (i = 0; i < num_pins; i++)
    {
        pinMode(pins[i], OUTPUT); // definit chaque point de raccordement comme etant un OUTPUT (sortie)
    }
}

void loop()
{
    int i;
    for (i = 0; i < num_pins; i++) { // loop through each pin...
        digitalWrite(pins[i], HIGH); // allume chaque LED l'une apres l'autre
        delay(timer); // pause
        digitalWrite(pins[i], LOW); // eteint
    }
    for (i = num_pins - 1; i >= 0; i--) {
        digitalWrite(pins[i], HIGH); //Allume et eteint en ordre inverse (decroissant index 6 a 0)
        delay(timer);
        digitalWrite(pins[i], LOW);
    }
}
```