

### ENTREES NUMERIQUES

Configurer la direction de la broche en entrée

```
pinMode(int [broche], INPUT);
```

Lire la valeur sur l'entrée correspondante

```
int a;
a = digitalRead(int [broche]);
```

[broche] = numéro de la broche à configurer

```
pinMode(3, INPUT);
int a = digitalRead(3);
```

Ex : récupère dans la variable a la valeur de la broche 3

### SORTIES NUMERIQUES

Configurer la direction de la broche en sortie

```
pinMode(int [broche], OUTPUT);
```

Mettre la sortie à '0' (logique)

```
digitalWrite(int [broche], LOW);
```

Mettre la sortie à '1' (logique)

```
digitalWrite(int [broche], HIGH);
```

[broche] = numéro de la broche à configurer

### SORTIES « ANALOGIQUES » / PWM

Les sorties numériques notées par le symbole ~ sur la carte, permettent de générer un **signal rectangulaire** de fréquence environ **1 kHz** et dont le **rapport cyclique** est **pilotable**

```
analogWrite(int [broche], int [valeur]);
```

[broche] = numéro de la broche à modifier

[valeur] = valeur comprise entre 0 et 255

0 : rapport cyclique de 0 %

255 : rapport cyclique de 100 %

```
analogWrite(9, 100);
```

Ex : applique un signal de rapport cyclique 100/255 sur la broche 9

### ALIMENTATION

L'alimentation se fait :

- soit par le port USB (ainsi que le téléversement des programmes)
- soit par un bloc externe ( $5V < V_{\text{alim}} < 12V$ )

**ATTENTION** : les broches n'acceptent que des tensions comprises entre **0 et 5V** / **Pas de tensions négatives**

### COMMUNICATION SERIE

Les broches **0** et **1**, notées **RX** et **TX** (ainsi que la liaison USB) permettent de transmettre des données selon la norme **RS232**

Configurer la communication

```
Serial.begin(int [baud]);
```

[baud] = vitesse de transmission

Envoyer un texte à afficher

```
Serial.print([texte à écrire]);
```

```
Serial.begin(9600);
Serial.print("a = ");
Serial.println(a);
```

Ex : démarre une communication à 9600 bauds, affiche : a = 3 (si a vaut 3) puis saute à la ligne suivante



### ENTREES ANALOGIQUES

La carte Arduino Uno possède 6 entrées analogiques reliées à un convertisseur analogique-numérique de 10 bits

(valeur entre 0 et 1023)

Récupérer la valeur d'une entrée analogique

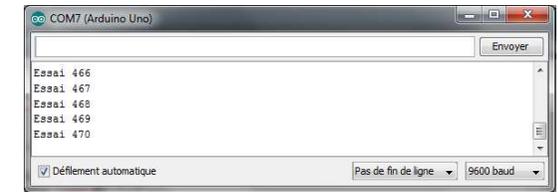
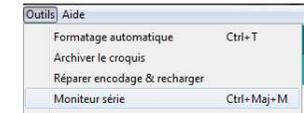
```
int a = analogRead(int [broche]);
```

```
int a = analogRead(A4);
Serial.print("a = ");
Serial.println(a);
```

Ex : récupère la valeur de l'entrée analogique A4 dans la variable a, puis l'envoie sur la liaison série pour l'afficher

### MONITEUR SERIE

Permet d'afficher les valeurs envoyées par la carte Arduino sur l'ordinateur



### STRUCTURE D'UN PROGRAMME

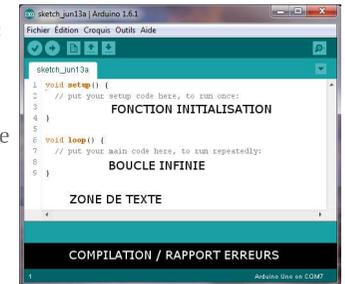
Un programme sous Arduino est composé de **deux parties** :

- une fonction **setup()** comportant l'initialisation de la carte (entrées/sorties...) exécutée qu'une seule fois
- une fonction **loop()** exécutée à l'infini

### ECRITURE

Le logiciel Arduino permet :

- d'éditer le programme
- de compiler
- de téléverser sur la carte



### COMPILATION



Ce bouton permet de compiler le code écrit en langage Arduino vers le langage compréhensible par la carte

### TELEVERSEMENT

La dernière étape est de téléverser le programme vers la carte pour qu'il soit exécuté. Avant cela, il faut avoir choisi le bon type de carte (Outils / Type de carte) et le port (Outils / Port Série)

Ce bouton permet alors de téléverser le programme vers la carte Arduino

**A VOUS DE JOUER...**