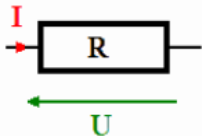
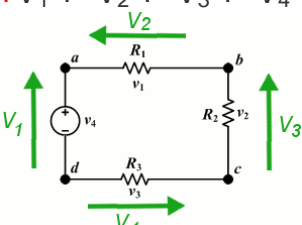
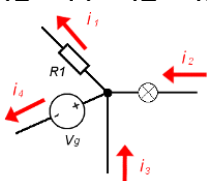
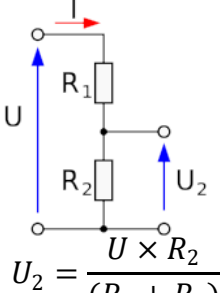
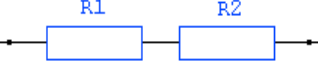
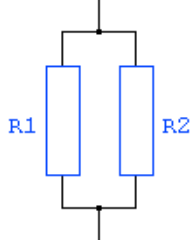

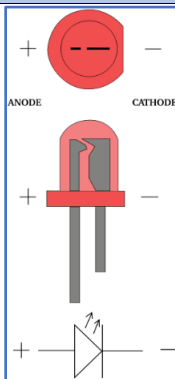
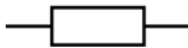

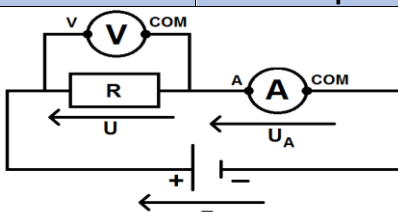
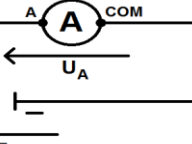
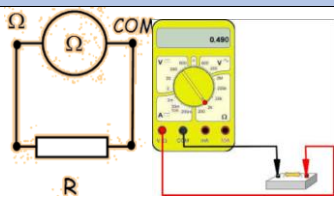
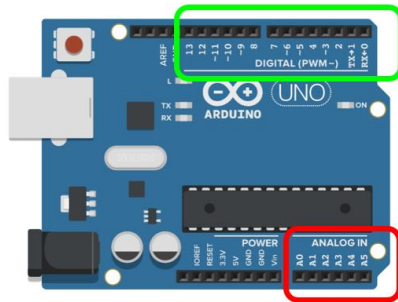


Lois de Kirchhoff		
Loi d'Ohm	Loi des mailles	Loi des nœuds
$U = R \times I$ <p>U : tension en volt (symbole : V) R : valeur de la résistance, en ohm (symbole : Ω) I : courant en Ampère (symbole : A)</p> 	<p>La somme des tensions le long d'une maille est toujours nulle</p> $+V_1 + -V_2 + -V_3 + -V_4 = 0$ 	<p>La somme des courants qui entrent dans un nœud est égale à la somme des courants qui sortent d'un nœud.</p> $I_1 + I_4 = I_2 + I_3$ 

Associations de résistances		
Pont diviseur de tension	Résistances en série	Résistances en parallèle
 $U_2 = \frac{U \times R_2}{(R_1 + R_2)}$	 $R_{\text{équivalente}} = R_1 + R_2$	 $\frac{1}{R_{\text{équivalente}}} = \frac{1}{R_1} + \frac{1}{R_2}$

Composants électriques		
Résistance		Diode
	<p>Une résistance est un composant électronique dont la principale caractéristique est d'opposer une plus ou moins grande résistance (mesurée en ohms) à la circulation du courant électrique.</p> <p>Les anneaux de couleur sur la résistance permettent, grâce à l'utilisation d'un code universel, de connaître sa valeur en Ohms.</p>	
Symbole européen	Symbole américain	 <p>Une diode est un composant électronique qui ne laisse passer le courant que dans un sens. Le courant passe de l'anode à la cathode mais ne peut pas passer de la cathode vers anode.</p>
		

Appareils de mesure		
Voltmètre	Ampèremètre	Ohmètre
		



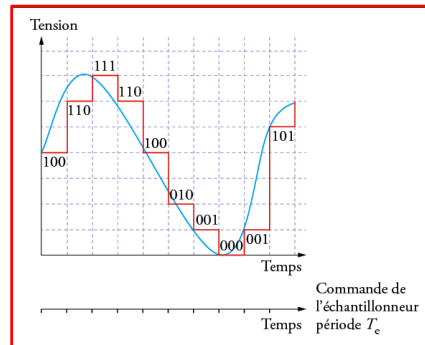
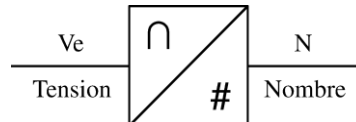
Les broches analogiques

Elles ne peuvent être utilisées qu'en entrée (INPUT)

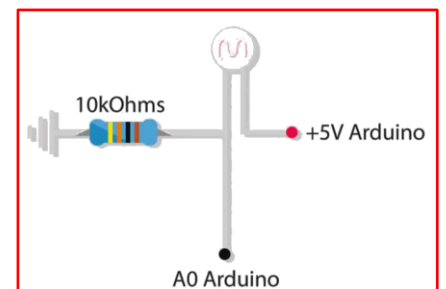
`infoCapteur = analogRead(A0);`

L'information reçue est une tension comprise entre 0 et 5V.

L'information est codée sur 10 bits ($2^{10} = 1023$). Les valeurs vont donc de 0 à 1023.



- Exemple : bouton tournant (potentiomètre)
- Exemple : résistance variant en fonction de la température ou de la luminosité



Les broches numériques

Elles peuvent être utilisées en entrée :

`infoCapteur = digitalRead(8);`

La valeur stockée dans 'infoCapteur' ne peut prendre que 2 valeurs : « 0 » ou « 1 » (« LOW » ou « HIGH »)

- Exemple : bouton poussoir

Elles peuvent être utilisées en sortie :

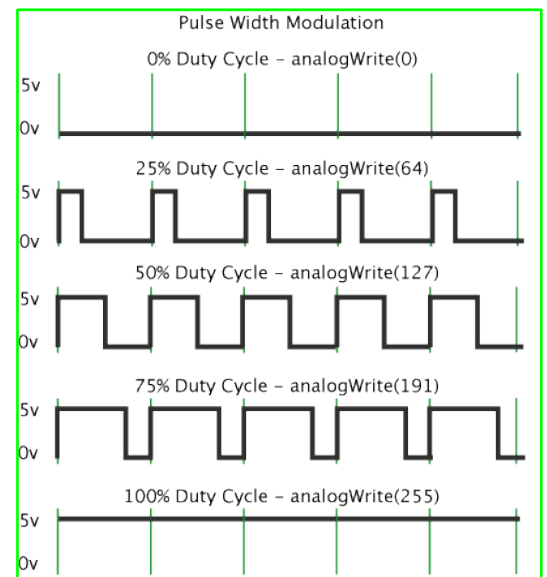
`pilotageLampe = digitalWrite(12, HIGH);`

- Exemple : allumer ou éteindre une lampe

Elles peuvent être utilisées en sortie analogique :

`analogWrite(13, valeur_rapport_cyclique);`

- Exemple : modifier l'intensité lumineuse d'une diode
- Exemple : positionner l'axe d'un servomoteur à un certain angle
- Exemple : faire varier la vitesse de rotation d'un moteur



La valeur du rapport cyclique est codée sur 1 octet (8 bits)

Les valeurs du rapport cyclique vont donc de 0 à 255 ($2^8 = 256$)

La structure d'un programme

1	Déclaration des constantes ; Déclaration des variables
2	Setup Exécuté 1 seule fois, au lancement du programme Configuration des entrées/sorties; Initialisation des variables
3	Loop Exécuté en continu; Les instructions sont exécutées en boucle jusqu'à l'appui sur « Reset »

```

1 int brocheCapteur = A0;    // selection de la broche sur laquelle est connectée le capteur
  int brocheLED = 13;        // selection de la broche sur laquelle est connectée la LED
  int valeurCapteur = 0;     // variable stockant la valeur du signal reçu du capteur

2 void setup() {
  // broche de la LED configurée en sortie
  pinMode(ledPin, OUTPUT);
}

3 void loop() {
  // lecture du signal du capteur
  valeurCapteur = analogRead(brocheCapteur);
  // allume la LED
  digitalWrite(brocheLED, HIGH);
  // delai de "valeurCapteur" millisecondes
  delay(valeurCapteur);
  // éteint la LED
  digitalWrite(brocheLED, LOW);
  // delai de "valeurCapteur" millisecondes
  delay(valeurCapteur);
}

```

La ponctuation

- Toute ligne de code se termine par un point-virgule « ; »
- Le contenu d'une fonction est délimité par des accolades « { } »
- Les paramètres d'une fonction sont contenus entre parenthèses « () »

Les variables

int	« integers » les entiers relatifs, codés sur 16 bits, compris entre « -32 768 ; ...-3 ; -2 ; -1 ; 0 ; 1 ; 2 ; ...32 767 »
unsignedint	« unsignedintegers » les entiers naturels, codés sur 16 bits, compris entre « 0 ; 1 ; 2 ; 3 ; ; 65 534 ; 65 535 »
boolean	Variable logique qui ne peut prendre que deux valeurs « true » et « false »
float	Nombres décimaux (nombres à virgule). Attention il faut utiliser un point (et non une virgule) avant les décimales
byte	Variable codée sur 8 bits (donc 1 octet) qui peut prendre les valeurs entières comprises entre 0 et 255

Les opérateurs

OPÉRATEURS ARITHMÉTIQUES

- = (égalité)
- + (addition)
- - (soustraction)
- * (multiplication)
- / (division)
- % (modulo)

OPÉRATEURS BOOLÉENS

- && (ET booléen)
- || (OU booléen)
- ! (NON booléen)

OPÉRATEURS DE COMPARAISON

- == (égal à)
- != (différent de)
- < (inférieur à)
- > (supérieur à)
- <= (inférieur ou égal à)
- >= (supérieur ou égal à)

OPÉRATEURS COMPOSÉS

- ++ (incréméntation)
- -- (décréméntation) (à re