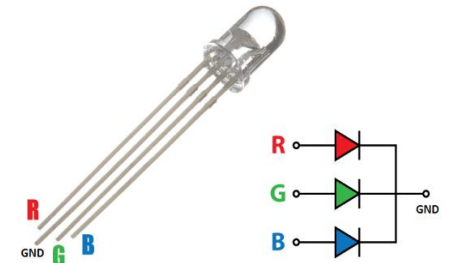


MLI (Modulation par Largeur d'Impulsion) / PWM (Pulse Width Modulation)

Lorsque l'on souhaite :

- Faire varier l'intensité lumineuse d'une diode ou d'un écran
- Choisir la nuance de couleur d'une diode RGB
- Faire tourner plus ou moins vite un moteur à courant continu
- Faire tourner d'un angle plus ou moins grand un servomoteur



On écrit sur une broche numérique (`analogWrite(numero_broche, valeur)`) de la carte Arduino une valeur appelée rapport cyclique.

Le rapport cyclique étant codé sur 8 bits, les valeurs vont de 0 à $2^8 - 1$ ($256 - 1 = 255$)

$$\alpha = \frac{\tau}{T}$$

rapport cyclique (sans unité)

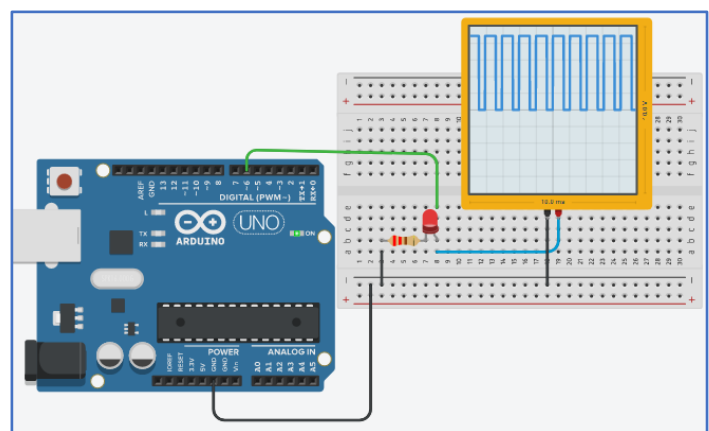
temps à l'état haut

τ

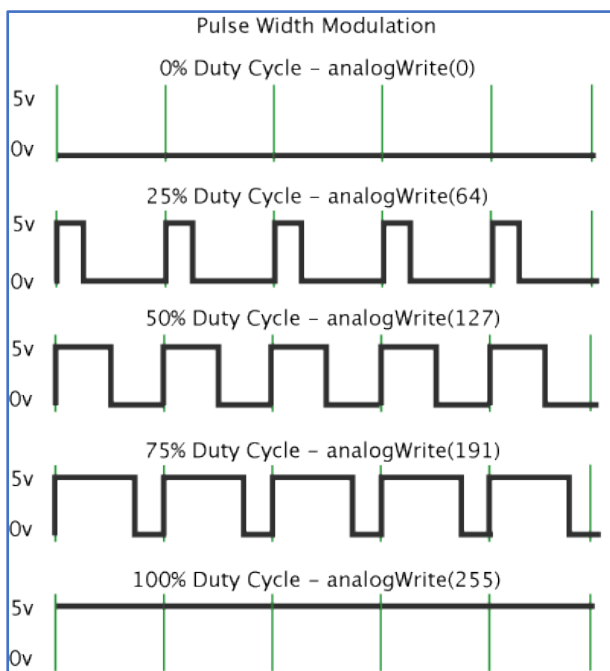
période du signal

T

même unité de temps (s, min, etc.)



Les broches PWM sont celles dont le nom est précédé d'une vague.



Q1 : Pour générer une tension moyenne de 4V, quelle valeur faut-il envoyer (valeur comprise entre 0 et 255), que vaut le rapport cyclique exprimé en pourcent ?

Q2 : Si l'on souhaite allumer à 60% de sa valeur maximale un DEL, quelle valeur doit-on écrire sur la broche de sortie ?

Q3 : Identifier les éléments du programme

| N° de lignes | Elements |
|--------------|-------------------------------------------------------------------|
| | Attendre 10millisecondes |
| | Définir la variable « led_pin » (type et valeur) |
| | Initialiser la broche utilisée comme une sortie |
| | Allumer la diode avec la valeur « i » |
| | Initialisation la vitesse de communication avec le moniteur série |
| | Définir un compteur pour incrémenter une variable « i » |
| | Définir un compteur pour décrémenter une variable « i » |
| | Faire en boucle |

```

1  int led_pin=6;
2  void setup()
3  {
4      pinMode(led_pin, OUTPUT);
5      Serial.begin(9600);
6  }
7
8  void loop()
9  {
10     for(int i=0; i<255; i++){
11         Serial.println(i);
12         analogWrite(led_pin, i);
13         delay(10);
14     }
15     for(int i=255; i>0; i--){
16         Serial.println(i);
17         analogWrite(led_pin, i);
18         delay(10);
19     }
20 }

```

Convertisseur analogique numérique(CAN)

Lorsque l'on souhaite :

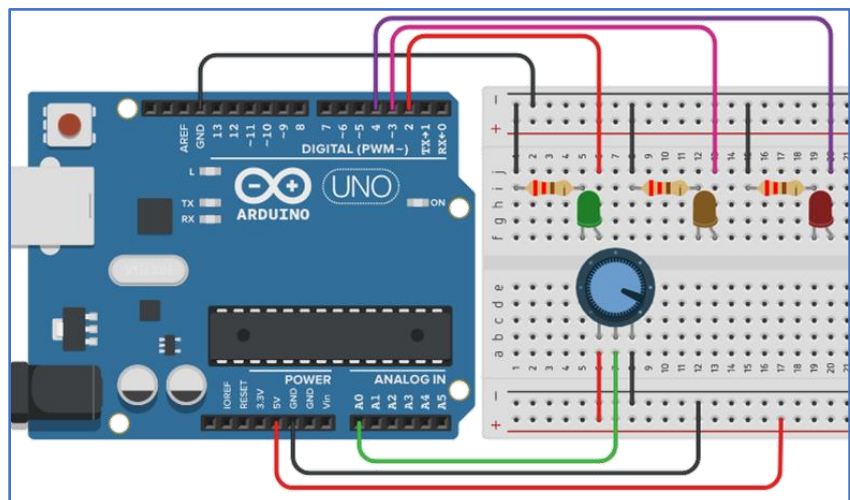
- Connaître de combien a été tourné un potentiomètre (bouton tournant)
- Le niveau de luminosité (à partir d'une résistance variable)
- La température de la pièce (à partir d'une résistance variable)

On demande à la carte Arduino de lire une valeur sur une broche analogique

(`analogRead(numero_broche)`).

Le convertisseur analogique numérique de la carte code sur 10 bits (donc de 0 à 1023) la tension comprise entre 0V et 5V.

Q4 : Calculer le quantum du convertisseur



Q5 : Identifier les éléments du programme

```

1  void setup()
2  {
3      pinMode(A0, INPUT);
4      Serial.begin(9600);
5
6      pinMode(2, OUTPUT);
7      pinMode(3, OUTPUT);
8      pinMode(4, OUTPUT);
9  }
10
11 void loop()
12 {
13     if (analogRead(A0) < 300) {
14         Serial.println(analogRead(A0));
15         digitalWrite(2, HIGH);
16         digitalWrite(3, LOW);
17         digitalWrite(4, LOW);
18         delay(1000); // Wait for 1000 millisecond(s)
19     } else {
20         if (analogRead(A0) >= 300 && analogRead(A0) < 600)
21             Serial.println(analogRead(A0));
22             digitalWrite(3, HIGH);
23             digitalWrite(2, LOW);
24             digitalWrite(4, LOW);
25             delay(1000); // Wait for 1000 millisecond(s)
26         } else {
27             Serial.println(analogRead(A0));
28             digitalWrite(4, HIGH);
29             digitalWrite(2, LOW);
30             digitalWrite(3, LOW);
31             delay(1000); // Wait for 1000 millisecond(s)
32         }
33     }
34 }

```

| N° delignes | Éléments |
|-------------|-----------------------------------------------------------------------------------|
| | Initialisation de la vitesse de communication entre la carte et le moniteur série |
| | Initialisation des broches comme des entrée ou des sorties |
| | Attendre 1 seconde |
| | Allumer la diode orange et éteindre les deux autres |
| | Allumer la diode rouge et éteindre les deux autres |
| | Ecrire la valeur lue sur la broche analogique, sur le moniteur série |
| | Actions réalisées si la valeur sur la broche analogique est supérieure ou égale à |



TP : Acquisition de l'information

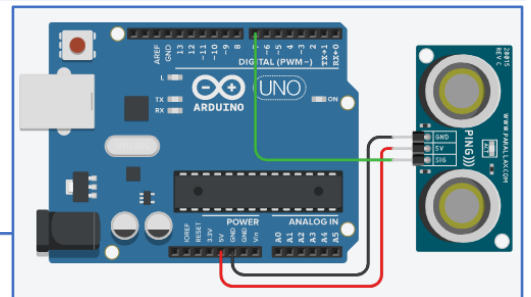
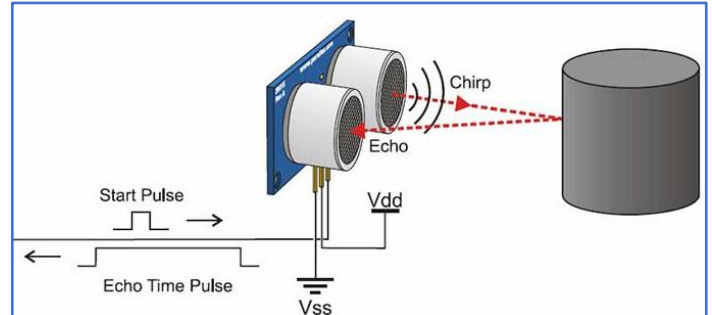
Spé SI
2021
Synthèse

300 tout en étant inférieure à 600.

Acquisition d'une distance avec un capteur à ultrasons

La vitesse du son dans l'air est d'environ 344m/s.

Q6 : Expliquer la valeur « 0,01723 » utilisée ligne 27



```

1
2 int cm = 0;
3
4 long readUltrasonicDistance(int triggerPin, int echoPin)
5 {
6   pinMode(triggerPin, OUTPUT); // Clear the trigger
7   digitalWrite(triggerPin, LOW);
8   delayMicroseconds(2);
9   // Sets the trigger pin to HIGH state for 10 microseconds
10  digitalWrite(triggerPin, HIGH);
11  delayMicroseconds(10);
12  digitalWrite(triggerPin, LOW);
13  pinMode(echoPin, INPUT);
14  // Reads the echo pin, and returns the sound wave travel time
15  // in microseconds
16  return pulseIn(echoPin, HIGH);
17 }
18
19 void setup()
20 {
21   Serial.begin(9600);
22 }
23
24 void loop()
25 {
26   // measure the ping time in cm
27   cm = 0.01723 * readUltrasonicDistance(7, 7);
28
29   Serial.print(cm);
30   Serial.println("cm");
31   delay(100); // Wait for 100 millisecond(s)
32 }
33

```

Q7 : Trouver sur le web la signification de « int » et de « long »

Variables & Data Types

byte, int, long, unsigned

| data type | bytes | min | max |
|---------------|-------|-------------|------------|
| char | 1 | -128 | 127 |
| byte | 1 | 0 | 255 |
| int | 2 | -32768 | 32767 |
| unsigned int | 2 | 0 | 65535 |
| long | 4 | -2147483648 | 2147483647 |
| unsigned long | 4 | 0 | 4294967295 |



Q8 : sur le programme, entourer en rouge la définition de la fonction et entourer en bleu l'appel de la fonction.

Très souvent, lorsque les composants utilisés sont complexes, on regroupe une partie des lignes de codes dans une fonction et on appelle cette fonction au moment où on en a besoin.

Cela permet de rendre plus lisible le programme principal.

```
void setup() {
  Serial.begin(9600);

  DashedLine(); ← Function is called here
  Serial.println("| Program Menu |");
  DashedLine(); ← Function is called again
}

void loop() {
}

void DashedLine()
{
  Serial.println("-----"); } Function is created here
```

