



Retard à l'allumage !

Projet – Retard à l'allumage !

Capacités exigibles :

- | | 😊 | 😐 | 😞 |
|--|---|---|---|
| <input type="checkbox"/> Étudier la réponse d'un dispositif modélisé par un dipôle RC. | | | |
| <input type="checkbox"/> Déterminer le temps caractéristique d'un dipôle RC à l'aide d'un microcontrôleur, d'une carte d'acquisition ou d'un oscilloscope. | | | |

On retrouve les dipôles RC dans de nombreux circuits. Ils permettent par exemple de retarder l'extinction d'une lampe et servent par exemple de « temporisateur ».

http://www.doritique.fr/Articles/View_Article.php?num_article=97

Les retardateurs se vendent dans le commerce. Sur le site d'un des leaders mondiaux des produits et systèmes pour installations électriques et réseaux d'information, on trouve le composant électronique suivant :

Caractéristiques techniques

Caractéristiques produit

Relais - Retardé à la fermeture

- Permet de retarder la mise sous tension d'une charge (alarme, éclairage, contacteur)
- La temporisation démarre à la mise sous tension du relais
- A la fin de la temporisation (T), la charge est mise sous tension
- Nombre de modules : 1

Caractéristiques générales

Relais temporisés

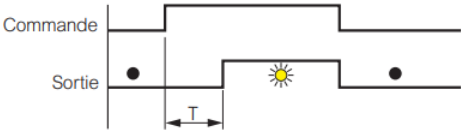
- Permettent de commander la mise sous tension et/ou la mise hors tension d'un circuit (éclairage, ventilation, automatisme, signalisation) en fonction d'une durée prédéterminée de 0,1 s à 100 h
- Tension d'alimentation : 12 à 230 V AC/DC (50/60 Hz)

<https://www.legrand.fr/pro/catalogue/31971-retarde-a-la-fermeture/relais-temporise-modulaire-retarde-a-la-fermeture-1-module>



Document 2 :

Sur la page du catalogue produits, on trouve le descriptif suivant :

Réf.	Relais
	Retardé à la fermeture
0 047 40	Permet de retarder la mise sous tension d'une charge (alarme, éclairage, contacteur)
	
	La temporisation démarre à la mise sous tension du relais A la fin de la temporisation (T), la charge est mise sous tension

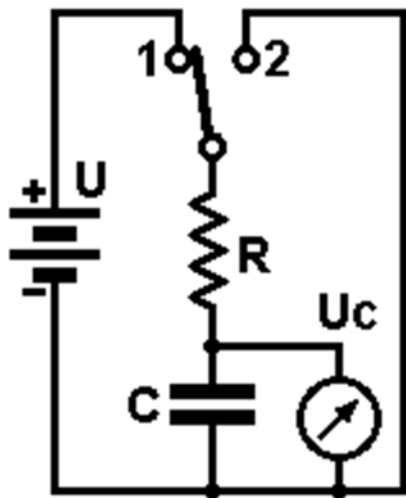
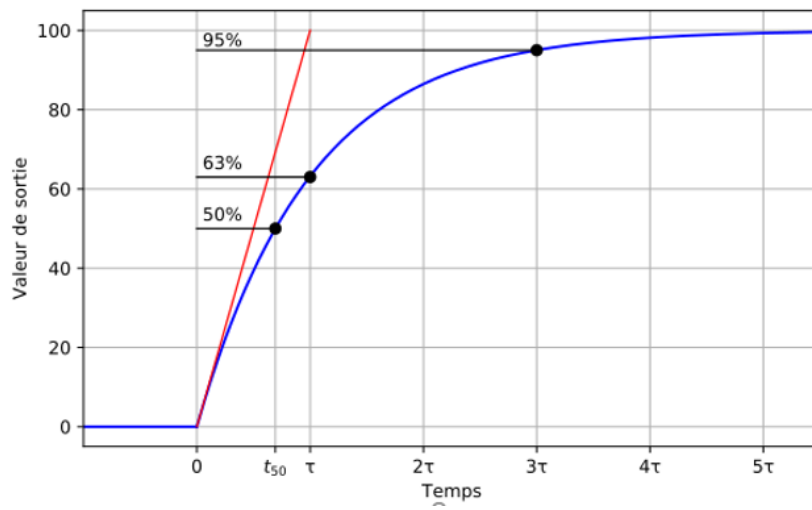
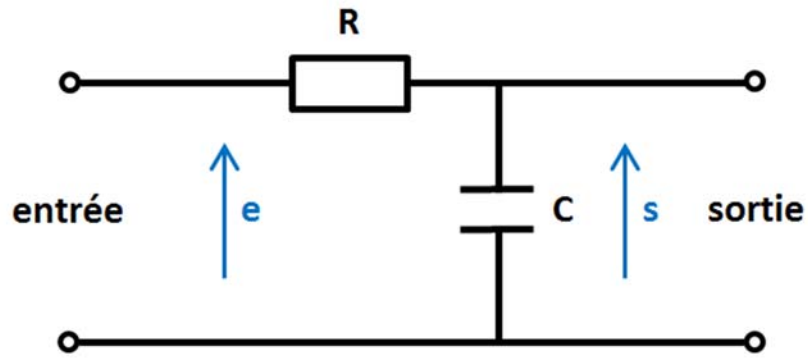
Objectif



Un bricoleur souhaite construire un tel dispositif à l'aide d'un circuit série RC et d'un microcontrôleur qui lui servira de relais.
Pour tester sa maquette, il cherche à retarder simplement l'allumage d'une DEL de 5s environ.

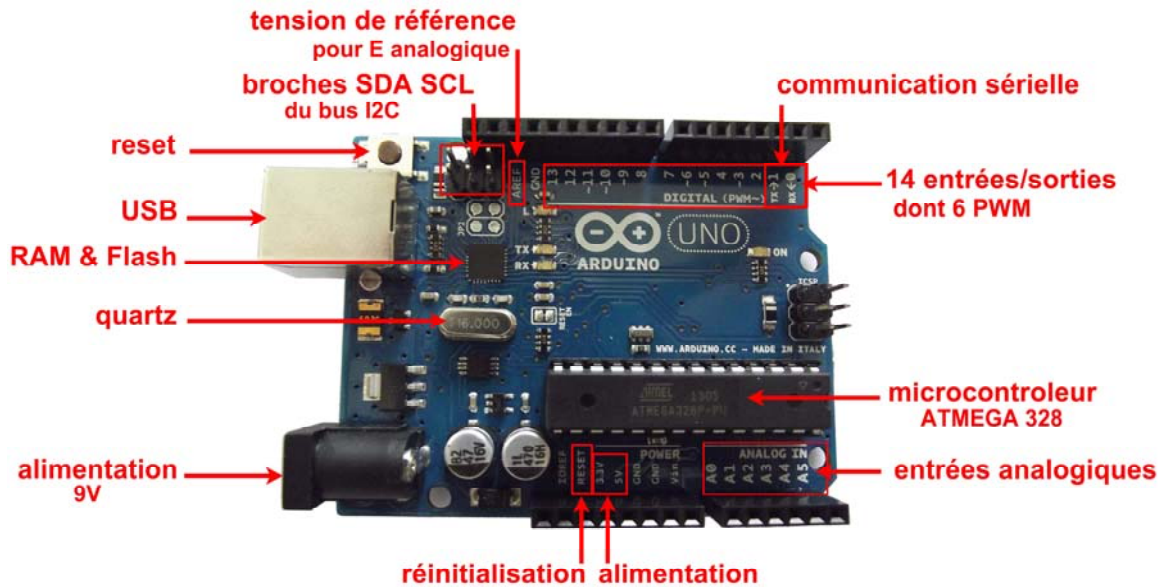


1. AIDE COMPETENCES PHYSIQUE-CHIMIE



$$\tau = R \times C$$

La Carte à microcontrôleur Arduino®



pinMode()

Configures the specified pin to behave either as an input or an output. **Used in a setup function.**

Syntax

```
pinMode(pin, mode)
```

pin: the Arduino pin number to set the mode of.

mode: INPUT, OUTPUT



setup()

The setup() function is called when a sketch starts. Use it to initialize variables, pin modes, start using libraries, etc.

Syntax
void setup{
...
}

The setup() function will only run once, after each powerup or reset of the Arduino board.



loop()

Use it to actively control the Arduino board.

Syntax
void loop{
...
}

After creating a setup() function, which initializes and sets the initial values, the loop() function does precisely what its name suggests, and loops consecutively, allowing your program to change and respond.



digitalWrite()

Write a HIGH or a LOW value to a digital pin.

Syntax
digitalWrite(pin, value)

If the pin has been configured as an OUTPUT with pinMode(), its voltage will be set to the corresponding value: 5V (or 3.3V on 3.3V boards) for HIGH, 0V (ground) for LOW.



AnalogRead()

Reads the value from the specified analog pin.

Syntax

`analogRead(pin)`

Arduino boards contain a multichannel, 10-bit analog to digital converter. This means that it will map input voltages between 0 and the operating voltage (5V or 3.3V) into integer values between 0 and 1023. On an Arduino UNO, for example, this yields a resolution between readings of: 5 volts / 1024 units or, 0.0049 volts (4.9 mV) per unit.



millis()

Returns the number of milliseconds since the Arduino board began running the current program.

Syntax

`time = millis()`

Returns the number of milliseconds since the Arduino board began running the current program. Data type: unsigned long.



Serial.print()

Prints data to the serial port as human-readable ASCII text.

Syntax

`Serial.print(val)`

Attention, paramétrage
nécessaire

```
void setup() {  
  Serial.begin(9600); // open the serial port at 9600 bps:  
}
```

3. AIDE COMPETENCE PROGRAMMER

```
int tension;  
unsigned long horloge;
```

```
void setup() { // put your setup code here, to run once:
```

```
  pinMode(A0,INPUT);  
  pinMode(2,OUTPUT);  
  Serial.begin(9600);  
}
```

```
void loop() { // put your main code here, to run repeatedly:
```

```
  tension=analogRead(A0);
```

```
  horloge=millis();
```

```
  Serial.print(horloge);  
  Serial.print(';');  
  Serial.println(tension);
```

```
  if (tension>640){digitalWrite(2,HIGH);}else{digitalWrite(2,LOW);};
```

```
  delay(250);  
}
```