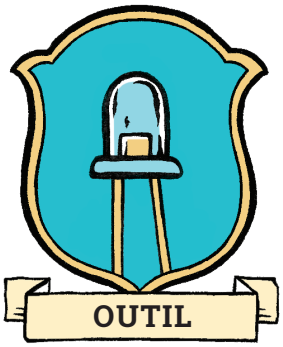
ARDUINO EN 6 DÉFIS

Un ensemble de fiches pour apprendre à utiliser Arduino
et pouvoir mener des projets originaux :
machines incroyables, mesures de physique,
votre imagination sera votre limite.

DÉFI

Six fiches « défi » pour découvrir
les fondamentaux des cartes Arduino



OUTIL

Des fiches « outil »
pour connaître le matériel

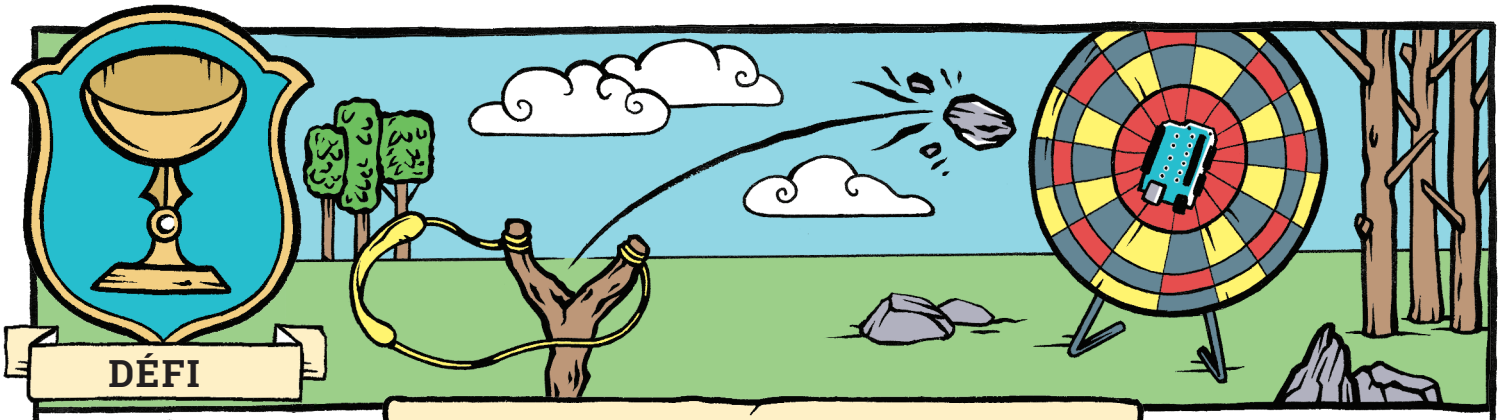
SAVOIR

Des fiches « savoir »
pour approfondir les notions

réalisation : **Marine Joumard**
en collaboration avec **Frédéric Bouquet** et **Julien Bobroff**,
équipe **La Physique Autrement**, **Université Paris-Sud**

— à retrouver en ligne sur : **www.opentp.fr** —

les images de circuits ont été créées avec le logiciel libre Fritzing



DÉFI

TESTEZ VOTRE CARTE

Réussir à faire clignoter la carte pour vérifier que tout marche !

FACILE

MATÉRIEL NÉCESSAIRE

– une carte Arduino
– un câble USB
– un ordinateur
(nécessaire pour tous les défis)

FICHES ASSOCIÉES

carte Arduino
logiciel Arduino IDE

INSTALLEZ LE LOGICIEL ARDUINO

Il faut un logiciel qui permette le dialogue entre votre ordinateur et votre carte Arduino. Pour cela, allez sur le site de référence **www.arduino.cc**, et cliquez sur l'onglet « Software ». Le programme Arduino IDE est libre et gratuit, installez-le en choisissant l'installateur qui correspond à votre ordinateur.

BRANCHEZ VOTRE CARTE ARDUINO

Une fois l'installation terminée, utilisez le câble USB pour brancher votre carte Arduino à votre ordinateur. La carte doit être reconnue par l'ordinateur. Si c'est le cas, le logiciel Arduino IDE doit détecter votre carte. Pour le vérifier, ouvrez ce logiciel et déroulez le menu Outils. Il y a deux paramètres qu'il faut vérifier dans ce menu : le type de carte et le port. Le paramètre « type de carte » doit correspondre à la carte que vous utilisez (Uno). Le paramètre « port » doit correspondre au port du votre ordinateur sur lequel est connectée la carte (par exemple « COM 11 : Arduino/Genuino Uno » si vous utilisez Windows, « \dev\tty. usbmodem … (Arduino Uno) » sur un ordinateur Apple). Si ces deux paramètres ne sont pas correctement configurés, la communication ne pourra pas s'établir.

ENVOYEZ UN PROGRAMME SUR LA CARTE

Les programmes sont écrits sur l'ordinateur, puis envoyés sur la carte Arduino. Pour vérifier la connexion avec votre carte, vous allez envoyer un programme de test. Ouvrez le menu Fichier, choisissez Exemples, Basics, et le programme « Blink ».

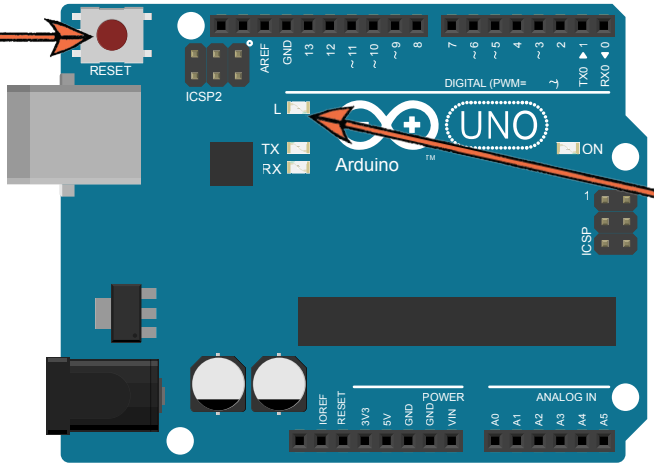
DÉFI – TESTEZ VOTRE CARTE

```
// the setup function runs once when you press reset or power the board
void setup() {

  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

TÉLÉVERSEZ

Envoyez ce programme sur la carte Arduino en cliquant sur l'icône « Téléverser » du logiciel. Votre ordinateur va transformer le programme en instructions compréhensibles par votre carte et lui envoyer par le câble USB.
Une fois installé sur votre carte (cela peut prendre quelques secondes), ce programme va s'exécuter en boucle.

Bouton reset de la carte. Il réinitialise la carte et relance le programme qui est installé dessus.

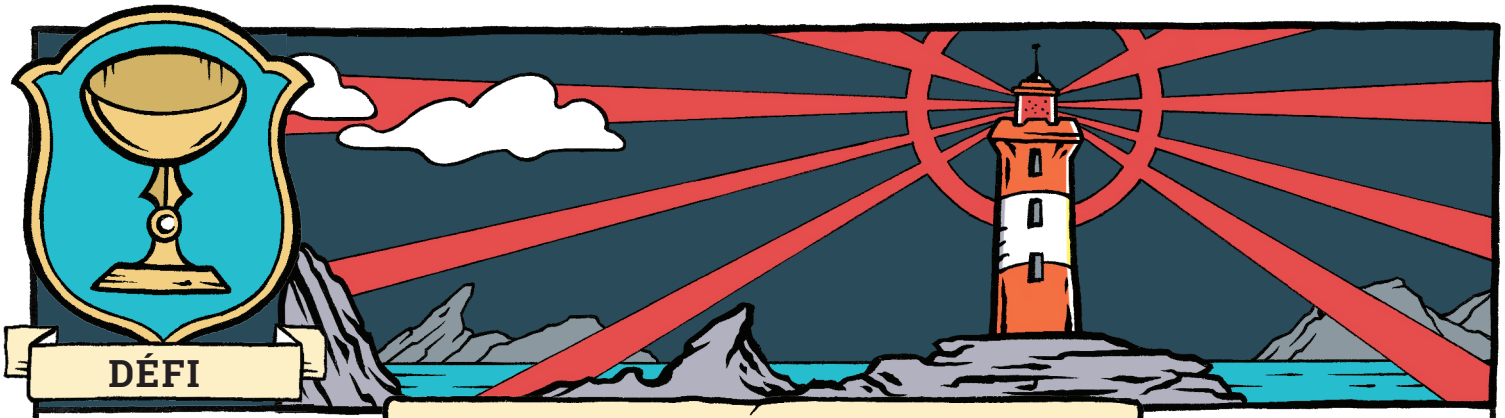


Petite diode lumineuse (LED) de test.
Cette LED est reliée au port numérique 13 sur la plupart des cartes Arduino.

Ce programme fait clignoter la LED de test. Si la LED de votre carte clignote : bravo !
Tout va bien, vous avez réussi à communiquer avec votre carte.

EN CAS DE PROBLÈME

– Vérifiez que vous avez bien configuré le type de carte Arduino et le port COM utilisé par le logiciel (menu Outils) ;
– Débranchez votre carte, arrêtez le logiciel Arduino, puis rebrancher et relancer le programme. Si ça ne fonctionne toujours pas, éteignez votre ordinateur puis relancez tout.



DÉFI

ALLUMEZ UNE LED

Vous allez contrôler la lumière !

FACILE

MATÉRIEL NÉCESSAIRE

- une LED rouge
- une résistance de 220 ohms
- un breadboard
- des petits fils électriques

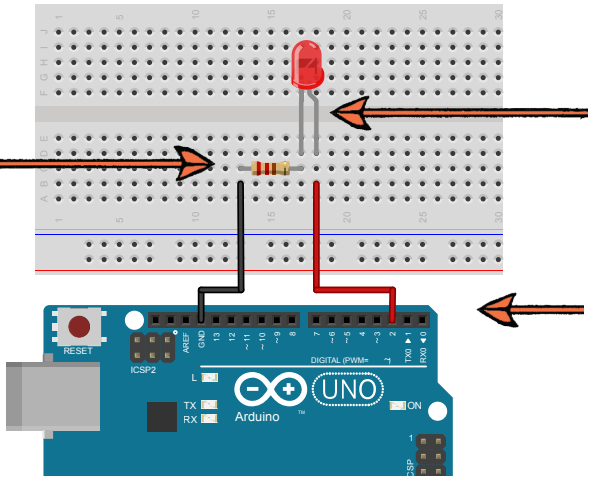
FICHES ASSOCIÉES

testez votre carte sorties numériques

RÉALISEZ CE MONTAGE ÉLECTRIQUE

Enfoncez bien les différents composants et fils dans le breadboard pour avoir un bon contact électrique.

Cette résistance évite d'envoyer trop de courant dans la LED et de la cramer.



Attention au sens pour brancher la LED : le fil le plus long est connecté avec le fil rouge (vers la tension +5 V).

Les connecteurs numérotés de 0 à 13 sont des ports numériques : vous pouvez imposer une tension de 5 V ou l'annuler sur chacun des ports.

RECOPIEZ CE PROGRAMME

HIGH : ça veut dire qu'on envoie 5 V dans le fil branché sur le port 2.

LOW : ça veut dire qu'on envoie zéro volt.

```
// setup pour initialiser la carte

void setup() {                    // début du setup
  pinMode(2, OUTPUT) ;            // initialise le port numérique 2
                                  //                     comme sortie
}                                 // fin du setup

// cette boucle va se répéter sans arrêt
void loop() {                     // début de la boucle
  digitalWrite(2, HIGH) ;         // allume la LED
  delay(1000) ;                   // attend 1 seconde (soit 1000 msec)
  digitalWrite(2, LOW) ;          // éteint la LED
  delay(1000) ;                   // attend 1 seconde (soit 1000 msec)
}                                 // fin de la boucle
```

Le texte en gris est un texte de commentaire, ignoré par l'ordinateur. Ce n'est pas nécessaire de le recopier mais il permet de mieux comprendre le programme.

Un programme similaire est facilement accessible à travers le menu du logiciel (menu Fichier, Exemples, Basics, programme Blink).

TÉLÉVERSEZ

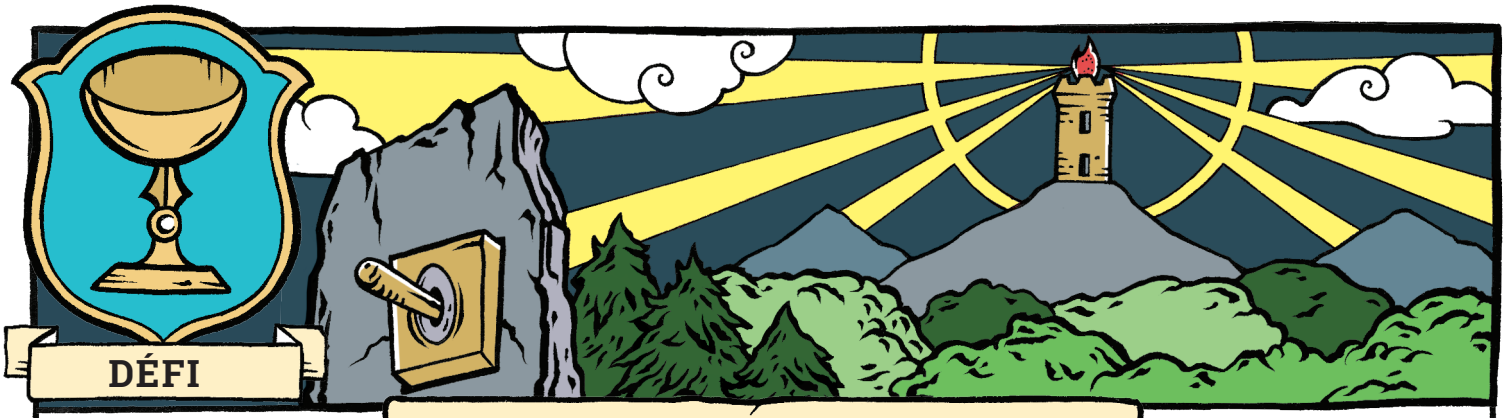
S'il ne se passe rien, c'est qu'il y a un problème ! Vérifiez votre montage électrique et la connexion entre votre ordinateur et la carte Arduino.

POUR ALLER PLUS LOIN

Essayez de changer la vitesse du clignotement. Pour çela, modifiez dans votre programme les valeurs des instructions delay( ), et observez ce qui se passe quand vous téléversez votre nouveau programme. Essayez d'allumer une autre LED branchée cette fois sur le port 3.

L'ULTIME DÉFI !

Modifiez votre programme pour que votre lumière lance un SOS en morse !



DÉFI

UTILISEZ UN INTERRUPTEUR

Vous allez contrôler votre carte en actionnant un interrupteur !

FACILE

MATÉRIEL NÉCESSAIRE

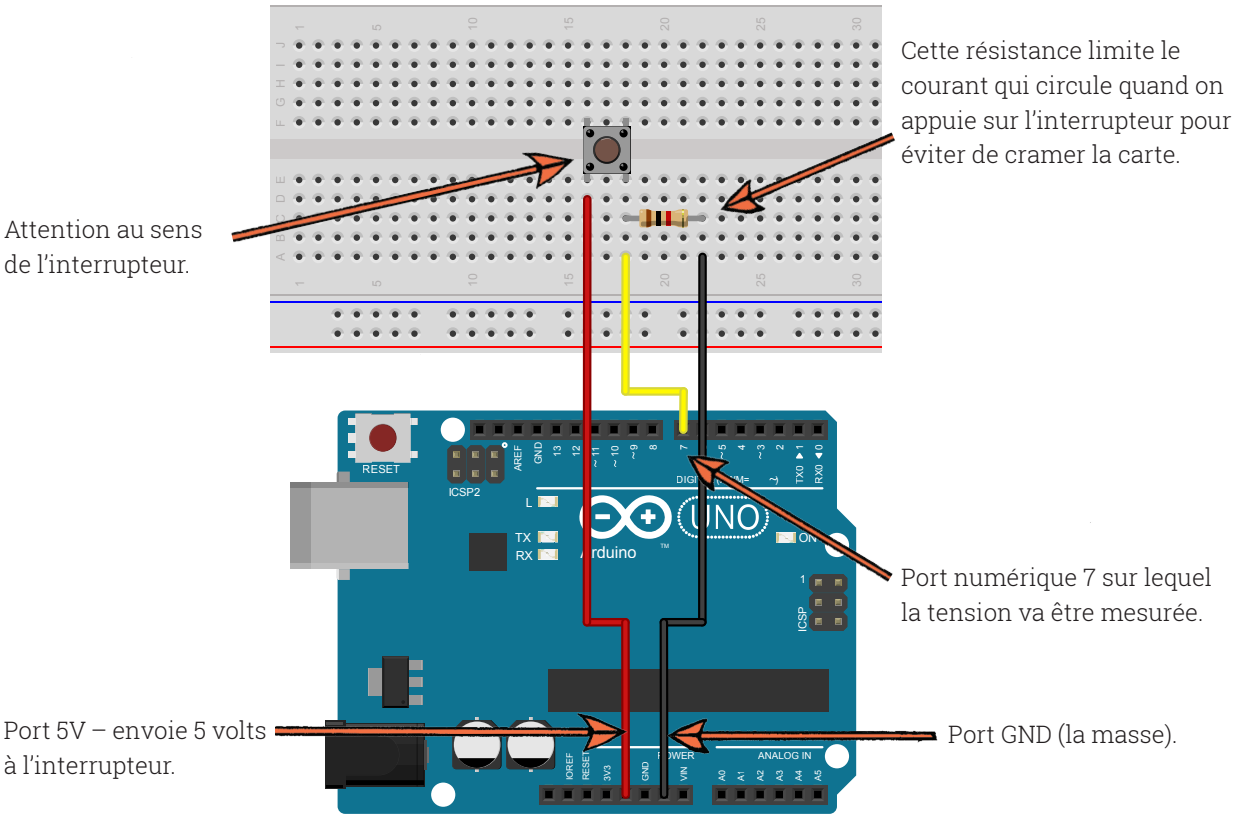
- un interrupteur
- une résistance de 1000 ohms
- un breadboard
- des petits fils électriques
- éventuellement un multimètre

FICHES ASSOCIÉES

testez votre carte
entrées numériques

RÉALISEZ CE MONTAGE ÉLECTRIQUE

L'interrupteur du kit de base Arduino possède quatre contacts. Les deux contacts utilisés dans le circuit doivent avoir une résistance nulle quand on appuie sur l'interrupteur, une résistance infinie sinon. Si vous avez un multimètre, utilisez-le pour vérifier cela. Sinon suivez le montage ci-dessous et faites pivoter l'interrupteur si ça ne fonctionne pas.



Quand on appuie sur l'interrupteur, la tension sur le fil jaune est de 5 volts. Quand on n'appuie pas, la tension est nulle grâce à la résistance qui relie le fil jaune à la masse. Sans cette résistance, la tension serait indéterminée.

DÉFI – UTILISEZ UN INTERRUPTEUR

RECOPIEZ CE PROGRAMME

L'instruction digitalRead(7) ordonne à la carte de lire la tension sur le port 7. Il y a deux résultats possibles :
HIGH (5 volts) ou LOW (0 volt).

Cette instruction demande à la carte d'envoyer un message à l'ordinateur par le câble USB.

```
// setup pour initialiser la carte
void setup() {          // début du setup
  pinMode(7, INPUT) ;  // initialise le port numérique 7 comme entrée
  Serial.begin(9600) ; // initialise la communication série avec l'ordinateur
}                       // fin du setup

// cette boucle va se répéter sans arrêt
void loop() {                    // début de la boucle
  if (digitalRead(7) == HIGH) {  // si on mesure 5 V sur l'entrée numérique 7
    Serial.println(" ON ! ") ;   // on envoie " ON ! " à l'ordinateur
  }
  else {                         // si on ne mesure pas 5 V
    Serial.println(" Off ! ") ; // on envoie " Off ! " à l'ordinateur
  }
  delay(100) ;                   // attend 0.1 seconde (100 millisecondes)
}                                // fin de la boucle
```

Un programme similaire est facilement accessible à travers le menu du logiciel (menu Fichier, Exemples, Basics, programme DigitalReadSerial).

TÉLÉVERSEZ

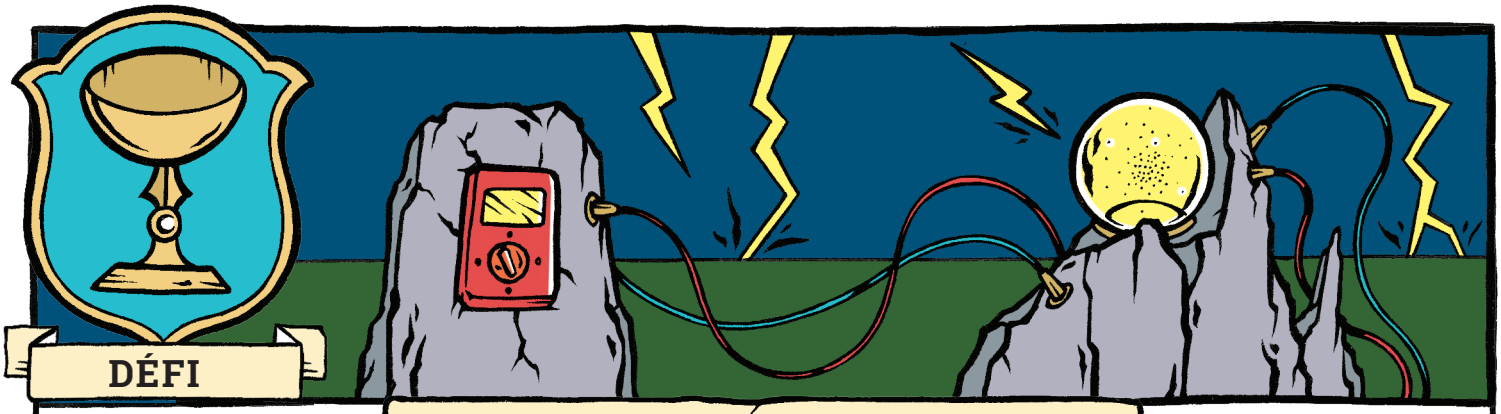
Il faut lire les messages que la carte envoie à l'ordinateur. Pour cela, utilisez le Moniteur série (dans le menu Outils du logiciel Arduino). Si des caractères bizarres apparaissent, vérifier la vitesse de connexion du Moniteur série, qui doit être la même que celle utilisée pour initialiser la communication dans le programme (9600 bauds). Appuyez maintenant sur l'interrupteur, et regardez ce qui se passe sur le Moniteur série !

POUR ALLER PLUS LOIN

Inversez les fils noirs et rouges, et observez ce qui se passe.

L'ULTIME DÉFI !

Modifiez votre programme pour que la LED de test sur la carte Arduino clignote quand on appuie sur l'interrupteur.



DÉFI

MESUREZ UNE TENSION

Vous allez mesurer une tension avec votre carte !

FACILE

MATÉRIEL NÉCESSAIRE

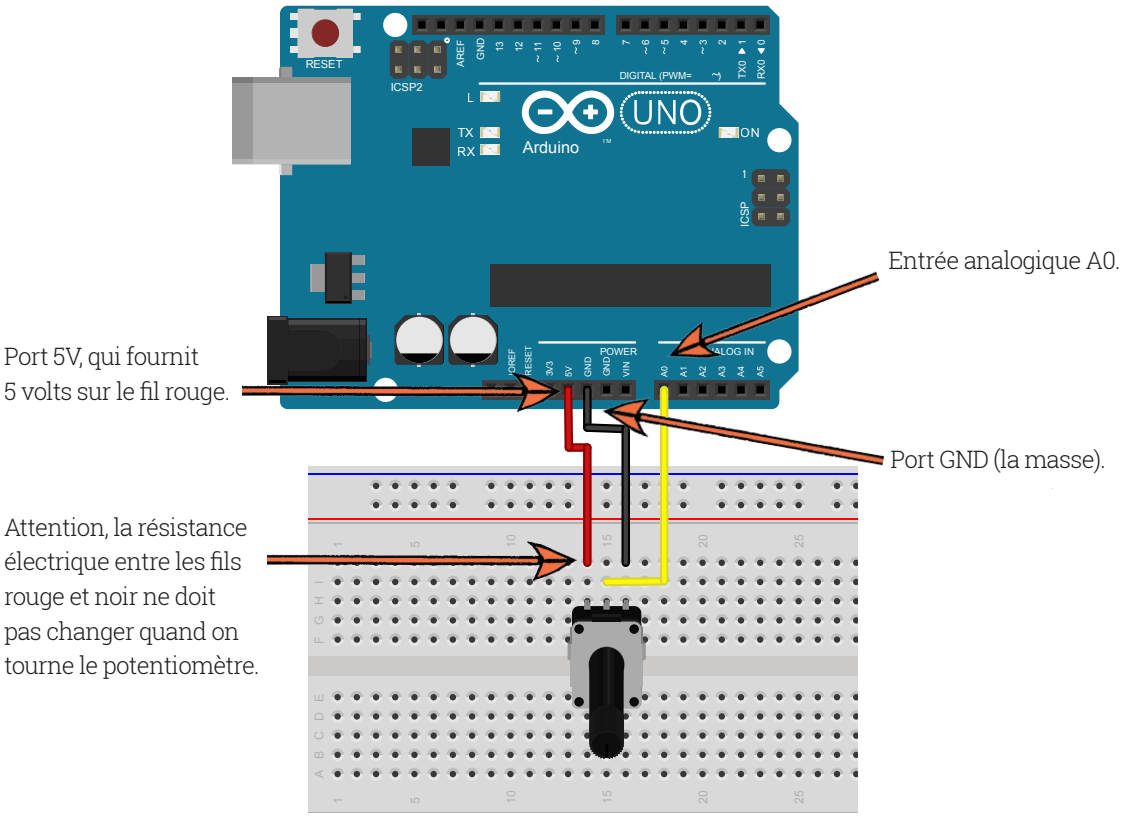
- un potentiomètre de 10 000 ohms
- un breadboard
- des petits fils électriques
- éventuellement un multimètre

FICHES ASSOCIÉES

testez votre carte
entrées analogiques

RÉALISEZ CE MONTAGE ÉLECTRIQUE

Un potentiomètre possède trois contacts. La résistance électrique entre deux de ces contacts ne varie pas quand on tourne le potentiométre : ce sont ces contacts qui doivent être reliés au port 5V et au port GND. (Si vous avez un multimètre, vous pouvez vérifier où se trouvent ces contacts, sinon faites confiance au montage ci-dessous.)



DÉFI – MESUREZ UNE TENSION

RECOPIEZ CE PROGRAMME

```
int MesureTension ;                // on définit la variable MesureTension
                                   // cette variable peut stocker des valeurs

// setup pour initialiser la carte
void setup() {                     // début du setup
  Serial.begin(9600) ;             // initialise la communication série avec l'ordinateur
}                                  // fin du setup

// cette boucle va se répéter sans arrêt
void loop() {                                  // début de la boucle
  MesureTension = analogRead(A0) ;             // on mesure la tension sur le port A0
                                               // et on l'attribue à la variable MesureTension
  Serial.println(MesureTension)                // on envoie la valeur mesurée à l'ordinateur
  delay(100) ;                                 // attend 0.1 seconde (100 millisecondes)
}                                              // fin de la boucle
```

Une variable permet de stocker une valeur dans un programme. Il faut les déclarer avant de les utiliser ; int est le type de la variable (un entier), MesureTension est le nom de la variable.

Petit temps d'attente entre deux mesures consécutives.

C'est cette instruction qui déclenche la mesure de la tension sur le port A0.

Un programme similaire est facilement accessible à travers le menu du logiciel (menu Fichier, Exemples, Basics, programme AnalogReadSerial).

TÉLÉVERSEZ

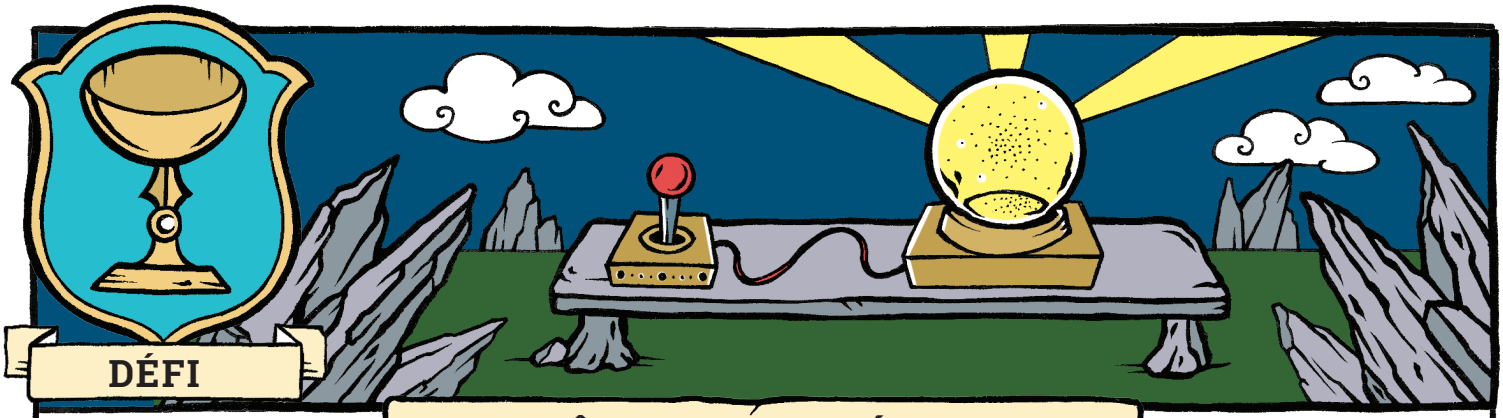
La carte Arduino va mesurer la tension sur le port A0 puis envoyer le résultat vers l'ordinateur. Pour les lire en direct depuis l'ordinateur, utilisez le Moniteur série (dans le menu Outils du logiciel Arduino). Si des caractères bizarres apparaissent, vérifiez la vitesse de connexion du Moniteur série, qui doit être la même que celle utilisée pour initialiser la communication dans le programme (ici 9600 bauds).
Tournez le potentiomètre, et regardez ce qui se passe sur le Moniteur série !

POUR ALLER PLUS LOIN

Branchez le fil rouge sur le port 3V3, et observez ce qui se passe (ce port délivre 3,3 volts). Trouvez la relation qui permet de traduire en volts ce qui est affiché dans le Moniteur série.

L'ULTIME DÉFI !

Modifiez votre programme pour que la LED de test sur la carte Arduino s'allume quand la tension mesurée dépasse 3,3 V (il faudra utiliser l'instruction « if… else… »).



DÉFI

CONTRÔLEZ VOTRE ÉCLAIRAGE

Vous allez contrôler l'intensité de l'éclairage d'une LED !

MOYEN

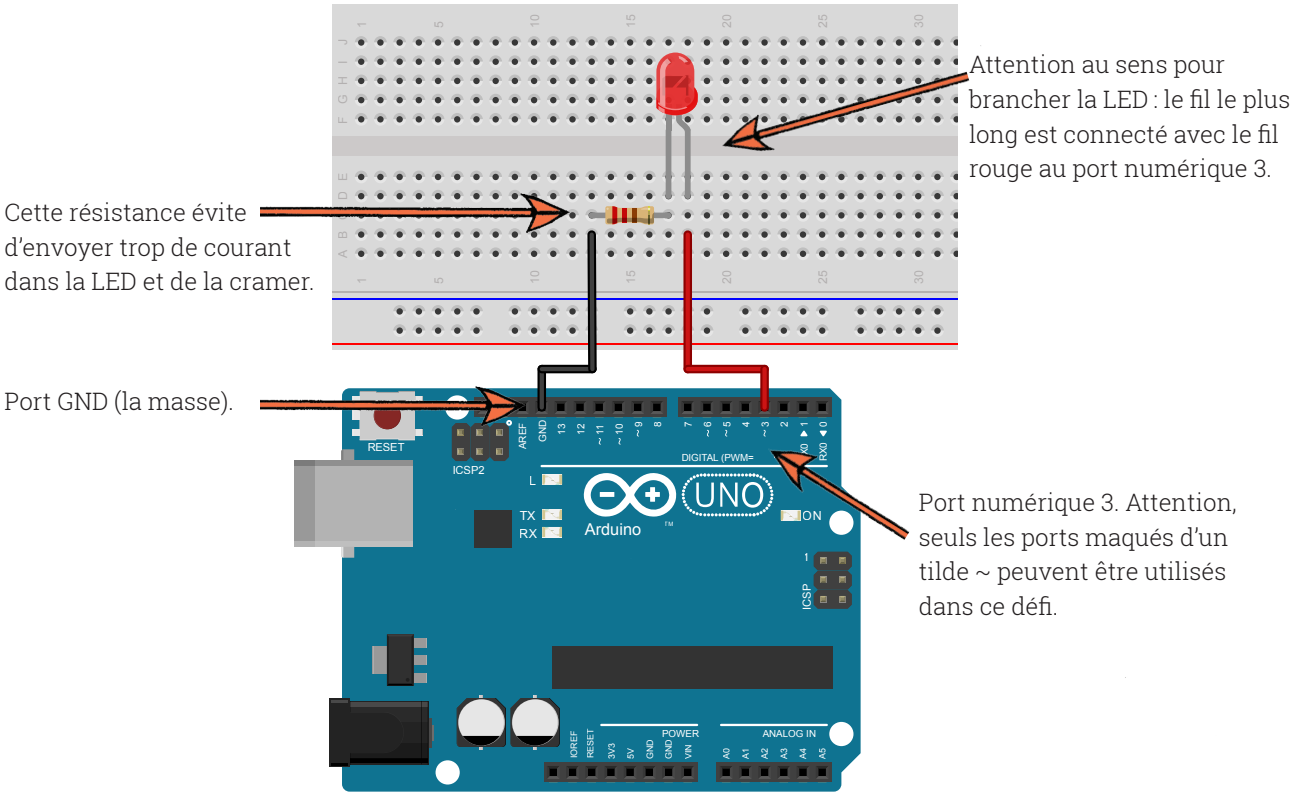
MATÉRIEL NÉCESSAIRE

- une LED rouge
- une résistance de 220 ohms
- un breadboard
- des petits fils électriques

FICHES ASSOCIÉES

testez votre carte sortie PWM

RÉALISEZ CE MONTAGE ÉLECTRIQUE



On peut contrôler la valeur moyenne de la tension sur les ports numériques marqués d'un tilde (voir la fiche « sortie PWM »).

RECOPIEZ CE PROGRAMME

Eclairage est une variable de type integer (un entier) qui va stocker la valeur de l'éclairage qu'on veut appliquer à la LED. On lui donne une valeur initiale de 0.

On impose une tension moyenne sur le port 3 qui vaut Eclairage et peut varier de 0 (ce qui correspond à 0 V) à 255 (ce qui correspond à 5 V).

```
int Eclairage = 0 ;   // on définit la variable Eclairage

// setup pour initialiser la carte
void setup() {                    // début du setup
  pinMode(3, OUTPUT);             // on initialise le port 3 comme une sortie
}                                 // fin du setup

// cette boucle va se répéter sans arrêt
void loop() {                                 // début de la boucle
  analogWrite(3, Eclairage) ;                  // on définit la tension de sortie du port 3
  delay(100) ;                                // attend 0.1 seconde (100 millisecondes)
  Eclairage = Eclairage + 10 ;               // on augmente de 10 la valeur de la variable
  if (Eclairage > 255) {                      // si on dépasse la valeur maximale autorisée
    Eclairage = 0 ;                           // alors on revient à 0
  }                                           // fin de la condition
}                                             // fin de la boucle
```

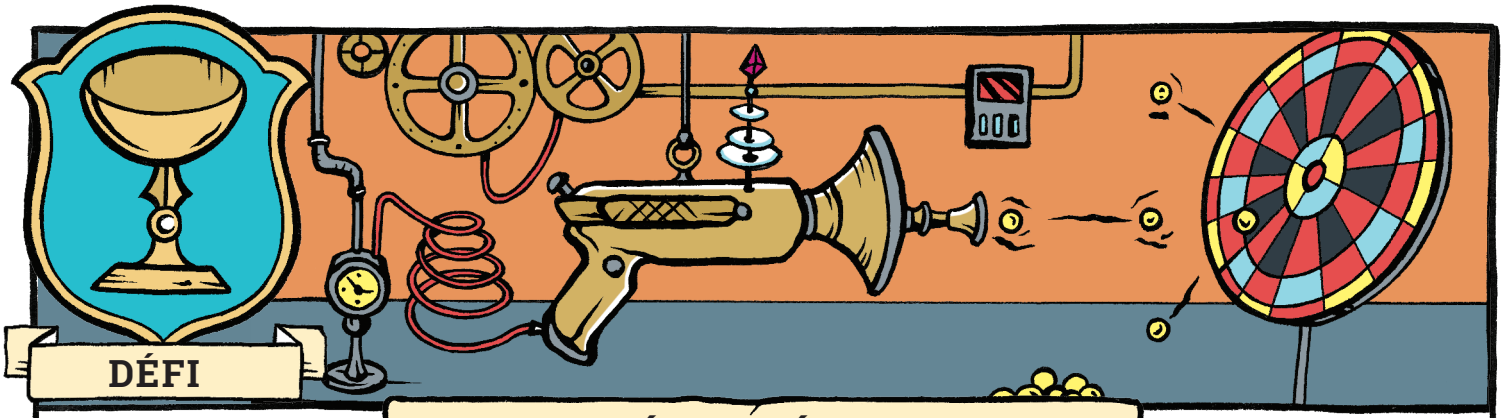
Un programme similaire est facilement accessible à travers le menu du logiciel (menu Fichier, Exemples, Basics, programme Fade).

TÉLÉVERSEZ

L'éclairage de la LED augmente progressivement !

L'ULTIME DÉFI !

Si vous avez déjà fait le défi « mesurez une tension », essayez de modifier le programme pour lire la tension réelle sur le port 3 en le reliant au port A0, et en envoyant le résultat sur le Moniteur série. Vérifiez si vous voyez bien la tension évoluer comme vous l'attendez (en volts).
Sinon, modifiez votre programme pour que votre éclairage augmente puis diminue progressivement. (et revenez à ce défi après avoir fait la fiche « mesurez une tension »).



DÉFI

DÉFI CRÉATIF

Soyez créatif et original… Et amusez-vous !

FACILE

MATÉRIEL NÉCESSAIRE

Tout le matériel dont vous disposez : capteurs et actuateurs, mais aussi légo, aimants, billes, papier aluminium, ballons, gobelets plastiques, élastiques, pâte à modeler, carton, etc.

FICHES ASSOCIÉES

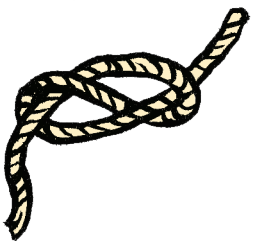
toutes

PRÉPARATION

Répartissez-vous en équipe de deux à quatre personnes. Rassemblez votre matériel sur une table facilement accessible en séparant les outils (capteurs et actuateurs : sonde de Hall, capteur de luminosité, accéléromètre, joystick, servomoteur, buzzer, etc.) et le matériel de bricolage, qui doit être le plus varié possible (aimant, pâte à modeler, pic à brochette, ballon, etc.).

LE DÉFI

Chaque équipe tire au hasard un outil et un accessoire de bricolage. Les équipes disposent ensuite d'un temps limité (typiquement 1h30 à 2h) pour concevoir et fabriquer un jeu avec comme contrainte d'utiliser cet outil et cet accessoire. À la fin du défi, les équipes se présentent leur jeu les unes aux autres et les font tester par les autres équipes. Pensez à prendre des photos et des petits films !



VARIANTES

Variante plus facile : Chaque groupe tire au hasard deux capteurs et deux accessoires, mais en choisit un de chaque.
Variante spéciale design : Le but du défi n'est plus de fabriquer un jeu, mais un objet utile, ou un objet inutile, ou un objet artistique, etc.

L'ULTIME DÉFI !

Pour ajouter encore des contraintes au jeu, faites tirer au sort une contrainte supplémentaire parmi celles-ci : drôle / qui surprend / pour l'apéro / challenge pour l'utilisateur / bruyant et lumineux / joue sur l'adresse / pour un enfant / sans les mains / artistique / sans les yeux…