

## 2 La loi de Mariotte

### Matériel nécessaire

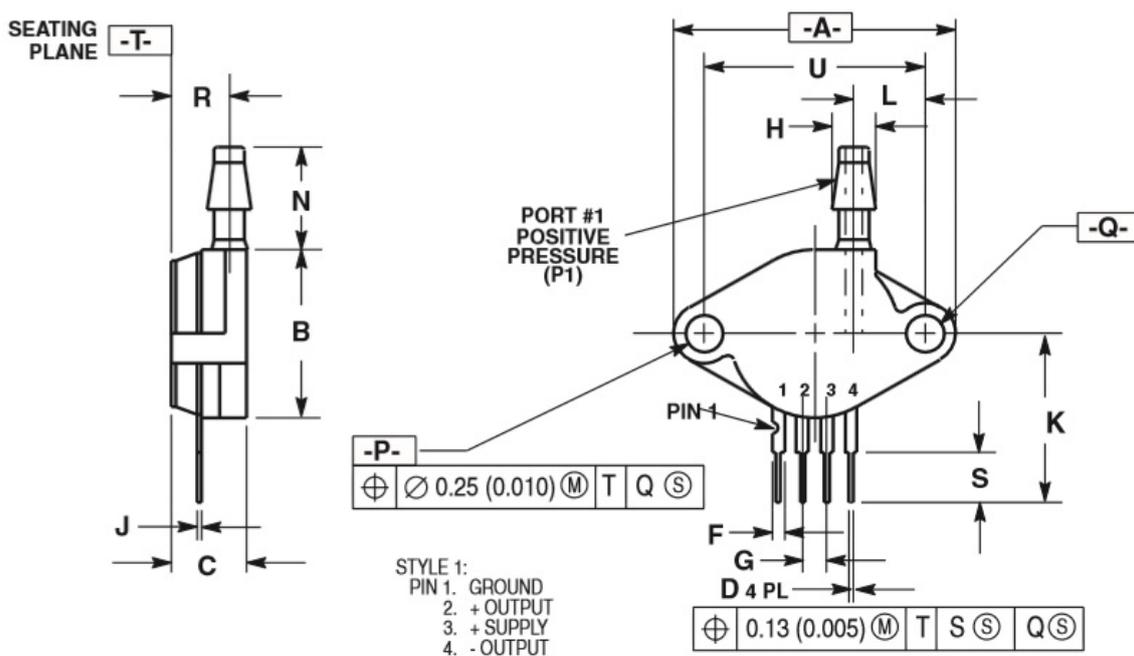
- Seringue
- Tube souple de raccordement
- Carte Arduino
- Ordinateur avec logiciel Arduino installé
- Capteur de pression absolue MPX2200AP
- Plaquette de connexion, fils de connexion
- Alimentation stabilisée 10 V
- Un ALI (AOP)
- Deux résistors de résistance  $R_1 = 1 \text{ k}\Omega$ , deux résistors de résistance  $R_2 = 10 \text{ k}\Omega$ .

### Fichiers Arduino

Programme à télécharger  
Fiche d'accompagnement  
[hatier-clic.fr/pc1221](http://hatier-clic.fr/pc1221)

### A. Description sommaire du capteur, nécessité d'un montage d'amplification

Le capteur MPX2200AP est un transducteur transformant le signal de pression  $P$  appliquée au niveau de son bec de raccordement en une tension électrique proportionnelle. Il possède quatre pattes comme l'indique le document technique d'accompagnement. Il s'agit du fichier « 10\_MPX2200.pdf » disponible à l'adresse [hatier-clic.fr/pc1221](http://hatier-clic.fr/pc1221).



- La patte 1, masse (GROUND) est raccordée à la masse (GND) d'Arduino.
- La patte 3, alimentation (SUPPLY) reçoit l'alimentation électrique.
- La tension électrique entre la patte 2, sortie + (+OUTPUT) et la patte 1, sortie - (-OUTPUT) est **proportionnelle** à la pression  $P$ .

Malheureusement, le capteur MPX2200AP :

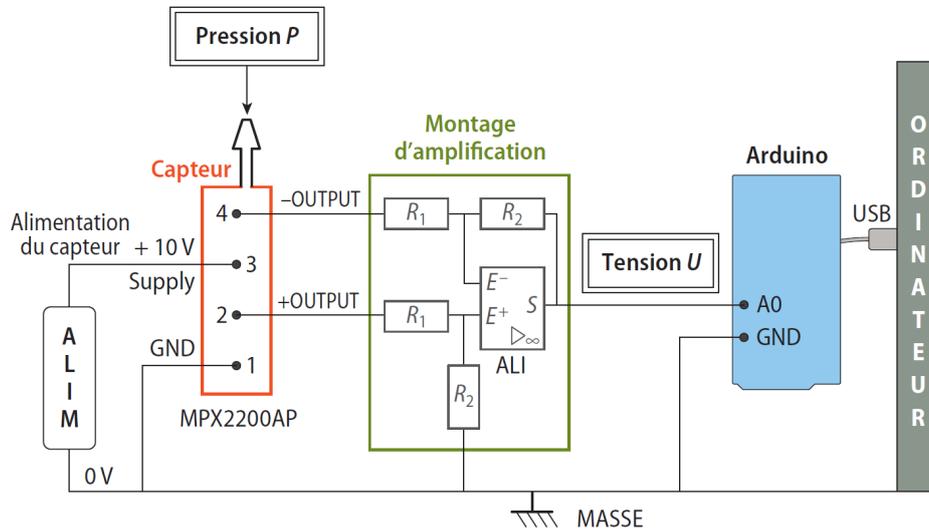
- nécessite une tension de 10 V ; les 5 V fournis par Arduino ne sont donc pas suffisants ;
- possède une sensibilité de 0,2 mV par kilopascal, et supporte une pression maximale de l'ordre de 200 kPa ; sa tension de sortie ne dépassera donc pas 40 mV, ce qui est insuffisant pour Arduino.

Il faut donc :

- utiliser une alimentation externe de 10 V ;
- réaliser un circuit d'amplification du signal, d'un facteur 10.

## B. Circuit de montage

Voici le montage permettant d'assurer l'alimentation et l'amplification d'un facteur 10 du signal délivré par le capteur :  $U = 10 \times (V_2 - V_4)$



## C. Programme d'acquisition et de conversion

Le programme « 10\_mpx2200ap.ino », disponible à l'adresse [hatier-clic.fr/pc1221](http://hatier-clic.fr/pc1221), assure :

1. l'acquisition du signal électrique  $U$  sur l'entrée analogique A0 d'Arduino ;
2. la boucle de mesure ;
3. la conversion de la tension électrique  $U$ , en volts, en pression  $P$ , en kilopascals.

Voici le principe de cette conversion :

Le capteur délivre une tension  $u = V_2 - V_4$  proportionnelle à la pression  $P$  avec une sensibilité égale à 0,2 mV par kilopascal. On en déduit :

$$P(\text{kPa}) \leftrightarrow u = 0,2 \times P(\text{mV}) \leftrightarrow U = 10 \times u = 2 \times P(\text{mV}) = 0,002 \times P(\text{V})$$

La tension électrique  $U$  appliquée sur l'entrée analogique A0 est convertie en une valeur numérique  $S$  envoyée à l'ordinateur selon la loi  $1\ 024 \leftrightarrow 5\ \text{V}$  donc  $S = 1\ 024 \times U/5$ .

On en déduit que  $S = 1\ 024 \times \frac{0,002 \times P}{5}$  donc  $P = \frac{5 \times S}{1\ 024 \times 0,002}$  en kilopascals.

```

mpx2200ap | Arduino 1.8.9
Fichier Édition Croquis Outils Aide
mpx2200ap
void setup() {
  Serial.begin(9600); // initialisation
}
void loop() { // début de la boucle de mesure répétitive
  float tension = analogRead(A0); // acquisition de la tension U = 10*u
  float pression = (5.0 * tension)/(1024.0 * 0,002); // calcul de la pression P par conversion
  Serial.print(pression); // affichage de la valeur de P
  Serial.print("kPa"); // affichage de l'unité
  delay(5000); // délai séparant deux mesures consécutives : 5000 ms = 5 secondes
} // fin de la boucle répétitive

Enregistrement terminé.
Le croquis utilise 3148 octets (9%) de l'espace de stockage de programmes. Le maximum est de 32256 octets.
Les variables globales utilisent 212 octets (10%) de mémoire dynamique, ce qui laisse 1836 octets disponibles.

8
Arduino/Genuino Uno sur COM4
    
```

## D. Réalisation complète de l'activité

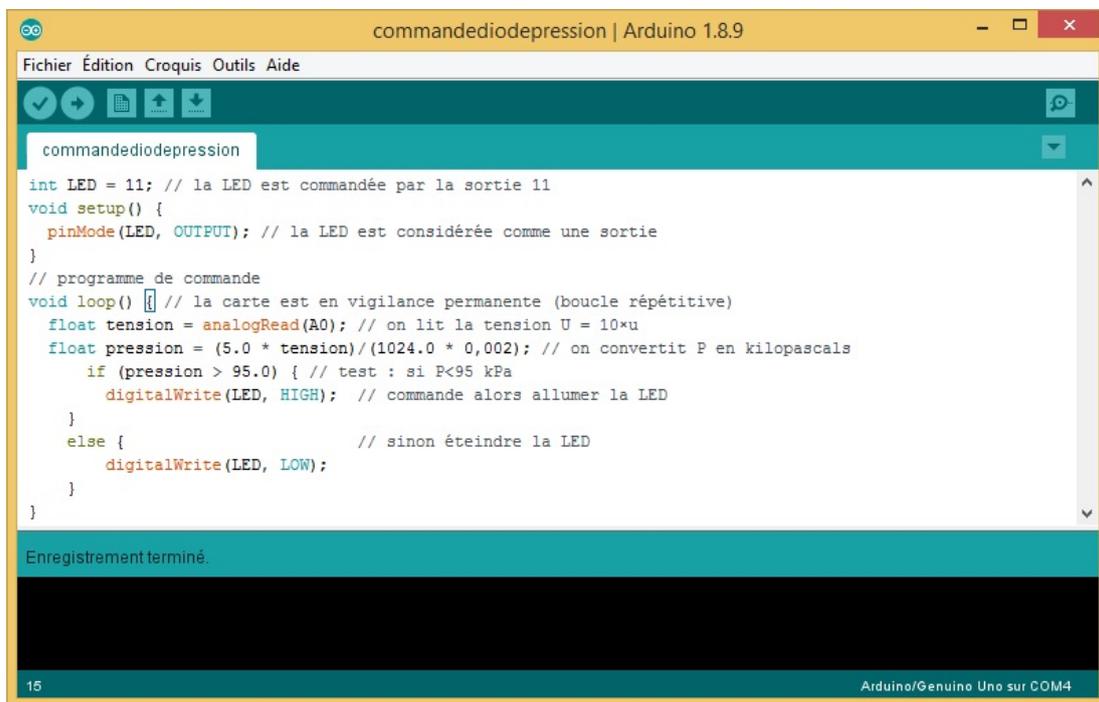
1. Faire le montage complet.
2. Relier le tuyau de sortie de la seringue à l'embout du capteur MPX2200AP.
3. Relier Arduino à l'ordinateur.
4. Démarrer le programme Arduino.
5. Saisir ou copier le programme.
6. Vérifier le programme en cliquant sur l'icône .
7. Téléverser le programme en cliquant sur l'icône .
8. Lancer le programme en cliquant sur l'icône . Une fenêtre « terminal série » s'ouvre et la valeur de la pression, mise à jour toutes les 5 secondes, s'affiche.

## E. Initiation à la robotique

Il est possible de commander le déclenchement d'un signal selon la valeur de la pression. Concrètement, le signal peut être :

- un message d'alerte (allumage d'une diode),
- ou plus complexe (mise en route d'un moteur, de fermeture d'un volet pour protéger une baie vitrée en cas de chute de la pression atmosphérique dans une situation cyclonique).

1. En plus du montage du paragraphe B, réaliser le montage suivant
  - Placer en série une LED rouge et un résistor (de protection)  $R = 330 \Omega$ .
  - Relier l'anode de l'association série LED et  $R$  à la sortie 11 de la carte Arduino (DIGITAL).
  - Relier la cathode de l'association série LED et  $R$  à la masse (GND) de la carte Arduino.
2. Taper ou copier le programme suivant (« 10\_commandediodepression.ino » disponible à l'adresse [hatier-clic.fr/pc1221](http://hatier-clic.fr/pc1221)) :



```

commandediodepression | Arduino 1.8.9
Fichier Édition Croquis Outils Aide
commandediodepression
int LED = 11; // la LED est commandée par la sortie 11
void setup() {
  pinMode(LED, OUTPUT); // la LED est considérée comme une sortie
}
// programme de commande
void loop() { // la carte est en vigilance permanente (boucle répétitive)
  float tension = analogRead(A0); // on lit la tension U = 10*u
  float pression = (5.0 * tension)/(1024.0 * 0,002); // on convertit P en kilopascals
  if (pression > 95.0) { // test : si P<95 kPa
    digitalWrite(LED, HIGH); // commande alors allumer la LED
  }
  else { // sinon éteindre la LED
    digitalWrite(LED, LOW);
  }
}
Enregistrement terminé.
15 Arduino/Genuino Uno sur COM4
    
```

3. Vérifier le programme en cliquant sur l'icône .
4. Téléverser le programme en cliquant sur l'icône .
5. Lancer le programme en cliquant sur l'icône .
6. Diminuer la pression en augmentant le volume dans la seringue (d'après la loi de Mariotte, lorsqu'à température constante le volume d'une quantité de matière de gaz augmente, sa pression diminue). Vérifier que la diode s'allume lorsque la pression est assez petite, donc que le volume est assez grand.