Acquisition de données analogiques et entrées clavier

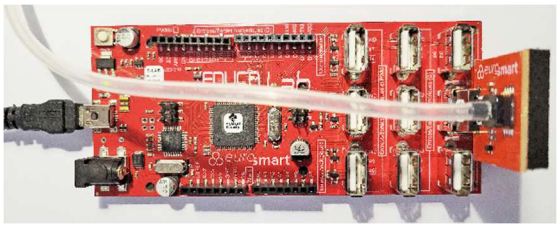
Principe : enregistrer un signal analogique point par point avec entrées de données au clavier.

Instructions

`void``([paramètres])`	Fonction ne renvoyant rien (avec ou sans paramètres)
type nom_fonction `([paramètres])`	Fonction renvoyant en résultat (**`return`** *valeur*)
`Serial.parseFloat()`	Renvoie un flottant lu dans le port série

Application : mesure de pression (capteur Elab-PA) et de volume (clavier).

Carte Educaduino (Arduino Mega) avec capteur de pression absolue :

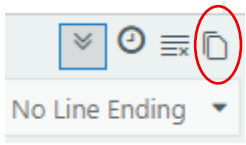


Le volume est lu sur la seringue et la valeur sera entrée au clavier.

Le capteur de pression absolue Elab-PA renvoie sur la broche **A9** une valeur de type float codée sur 10 bits donc comprise dans l'intervalle [0, 1023] correspondant à une pression (en Pa) dans l'intervalle [20 000, 400 000] (i.e. entre 200 et 4000 hPa).
Il faut donc définir une fonction permettant de calculer la pression réelle P à partir de la valeur mesurée V :

$P = aV + b$ où $a = \dfrac{P_{max} - P_{min}}{V_{max} - V_{min}}$ et $b = P_{max} - aV_{max}$ avec $P_{max} = 400000$, $P_{min} = 20000$, $V_{max} = 1023.0$, $V_{min} = 0.0$.

Ecrire le croquis (ou esquisse ou sketch) permettant d'entrer les valeurs de volume au clavier et de mesurer la pression avec le capteur et d'afficher les données dans le moniteur série.



Exploitation rapide : copier les données directement dans le moniteur série à l'aide du bouton :
Coller dans un fichier texte (Notepad++, Bloc-notes) et enregistrer au format texte (.txt).
Exploiter dans Régressi : tracer le produit PV en fonction de V. La loi de Mariotte est-elle vérifiée ?

Sauvegarde des données dans un fichier : cf. « Lire des données dans le port série avec Python ».

Aide

Le sketch page suivante permet de réaliser l'acquisition de grandeurs (analogiques ou numériques) en contrôlant l'acquisition au clavier :

- soit pour entrer des données au clavier pour une mesure difficile à effectuer via un capteur ;
- soit pour interrompre l'acquisition et effectuer un réglage entre deux mesures.

Par ailleurs, les acquisitions et les sorties sont regroupées dans deux fonctions (non indispensable mais données à titre d'exemple afin d'illustrer la structure de programmes complexes).

Sketch (principe → code à adapter)

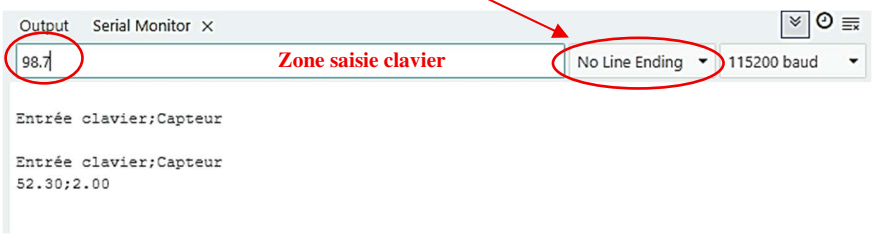
```
1  // Un seul capteur dans cet exemple : entrée analogique A0
2  #define capteurPin A0    // Broche Arduino utilisée
3
4  float capteurValeur;     // Variable de stockage de la valeur capteur
5  float entreeClavier;     // Données clavier issues du port série
6
7  void setup() {
8    Serial.begin(115200);
9    while (!Serial) {;}
10    Serial.println("\nEntrée clavier;Capteur"); // Colonnes (en-tête fichier)
11  }
12
13 void loop() {
14   if (Serial.available() > 0) {              // Si données dans le port série
15     entreeClavier = Serial.parseFloat();     // Conversion données port série
16     capteurValeur = lecture(capteurPin);     // Lecture capteur
17     ecriture(entreeClavier, capteurValeur);  // Ecriture dans le port série
18   }
19 }
20
21 float lecture(int broche) {
22   float valeurMesuree = analogRead(broche) / 1023.0 * 5.0; // Conversion
23   delay(2); // delai (ms) pour laisser le CAN réagir
24   return valeurMesuree;
25 }
26
27 void ecriture(float c, float v) {
28   Serial.print(c);
29   Serial.print(";");
30   Serial.println(v);
31 }
```

A la ligne 2, **`#define`** est une ***directive de compilation*** (cf. « Arduino – Mémento »).

A la ligne 21, on définit une fonction *lecture* admettant deux paramètres et renvoyant une valeur (syntaxe avec type sans *void*).
A la ligne 27, on définit une fonction *ecriture* admettant deux paramètres et ne renvoyant aucune valeur (*void* sans type).
L'ordre d'écriture des fonctions est sans importance.

A la ligne 15, la boucle est interrompue tant que rien n'est entré au clavier dans la zone de saisie (copie d'écran ci-dessous).
Après validation, les lignes 16 et 17 sont exécutées et la boucle recommence.
Lignes 15 et 28 des conversions sont effectuées car les données transitant par le port série sont des chaînes (lecture et écriture).

⚠ Bien sélectionner 'No Line Ending' dans le moniteur série.



Sans cette précaution, deux lignes apparaissent à chaque saisie.