



Manuel d'utilisation d'Adar

Adar est une application développée pour faire de l'acquisition de données avec une carte de type Arduino. L'application communique avec la carte par le biais du port Série.

Deux données peuvent être envoyées par la carte à l'application. Pour l'instant, il n'est pas possible d'en envoyer davantage. Par exemple, on peut envoyer le temps et la distance mesurée par un capteur ultrasons, ou encore la tension électrique et le courant.

Lorsque l'application reçoit les données, ces dernières sont inscrites dans le tableau et elles apparaissent aussi dans le graphique qui se trace au fur et à mesure. Le tout est pratiquement instantané.

Les données peuvent ensuite être copiées pour être analysées davantage.

1. Préparation de la carte de type Arduino

Le principe est assez simple : il faut d'abord choisir quelles valeurs doivent être envoyées à l'application.

On envoie d'abord la première valeur avec « print », on envoie ensuite deux espaces blancs « » et ensuite la 2^e valeurs avec « println ». Les deux espaces blancs sont utilisés par l'application pour séparer les deux valeurs numériques. Ces espaces sont attendus par l'application et doivent absolument s'y trouver.

La communication par le port série doit être faite absolument à une vitesse de 9600 bds.

On peut programmer la carte de type Arduino avec n'importe quel programme prévu à cet effet (Arduino IDE, mBlock,...).

Des exemples de code se trouvent dans le répertoire GitHub de Adar (<https://github.com/EmilieGuay/Adar>).

Exemple de code pour Arduino IDE

```
float temps =0;

void setup() {

  Serial.begin(9600);
  //À changer selon les besoins
  pinMode(A4, INPUT);
}

void loop() {

  //le temps est la première valeur envoyée
  temps=millis();
  Serial.print(temps/1000);

  //Séparateur
  Serial.print(" ");

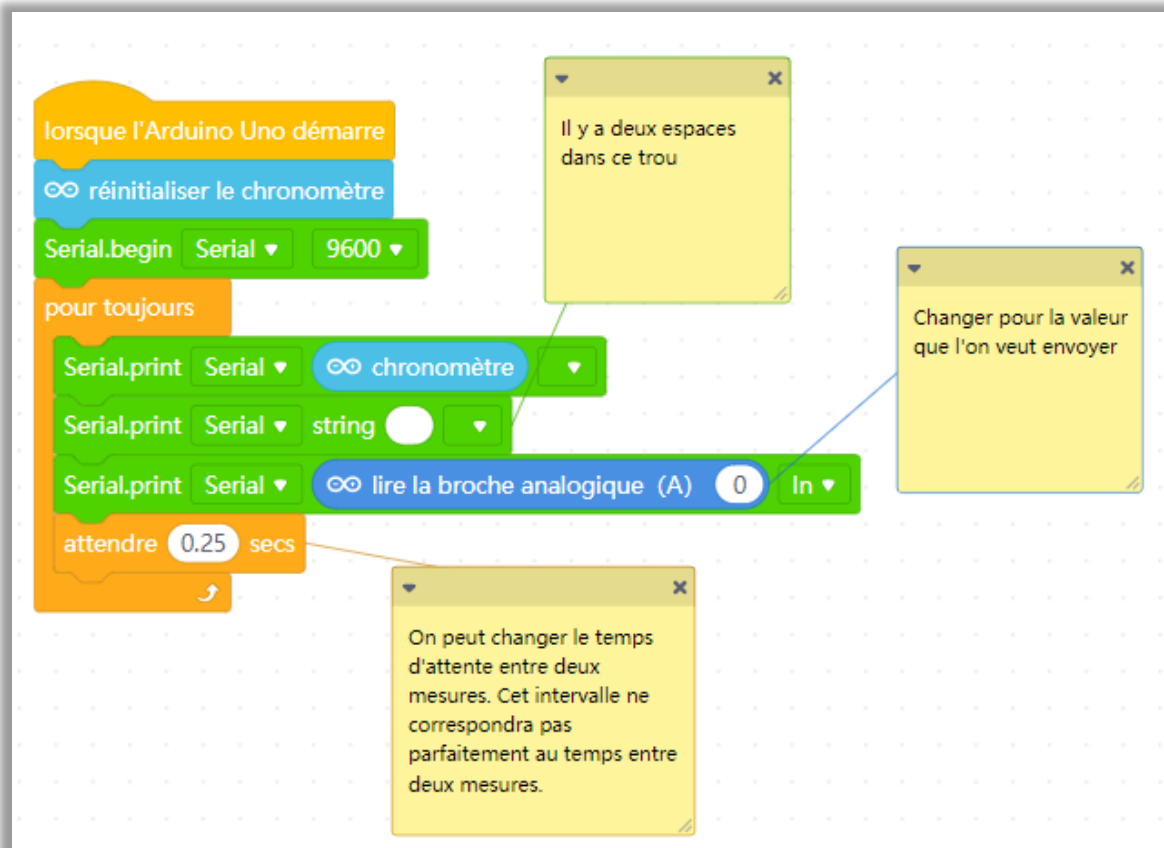
  //la 2e valeur que l'on veut envoyer :
  Serial.println(analogRead(A4));

  //on peut changer le délai (ici en millisecondes).
  //Attention, cela ne correspondra pas exactement à l'intervalle
  //de temps entre deux mesures.
  delay(10);
}
```

Si on augmente le temps dans « delay(10) », moins de valeurs seront prises.

La première valeur envoyée ici est le temps, mais il est possible de mettre n'importe quelle autre valeur, selon les besoins.

Exemple de code pour mBlock

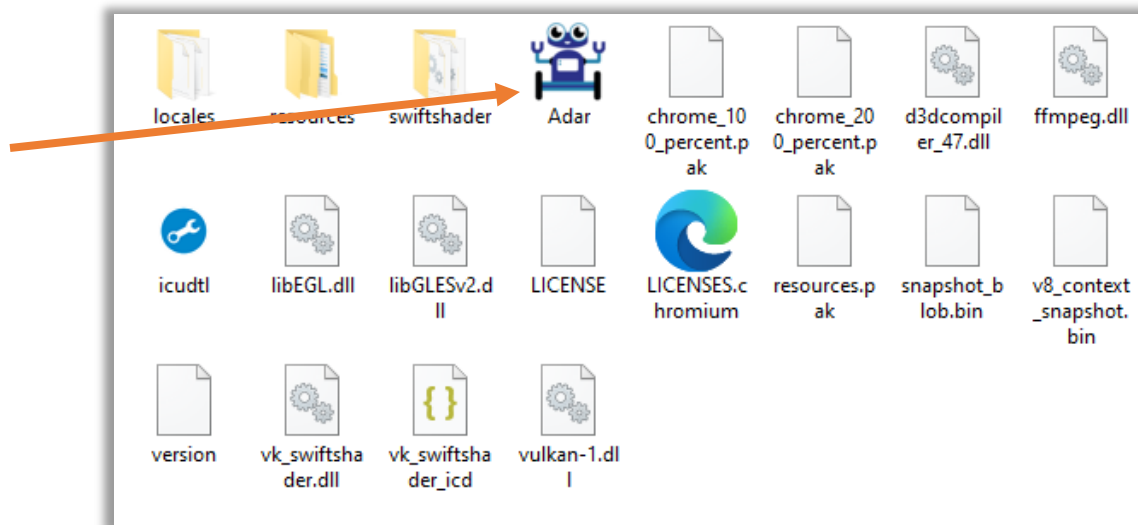


Le fichier Cinématique_Adar.mblock montre comment programmer la carte pour son utilisation avec un capteur ultrasons. Cela permet de faire des expériences de cinématique, puisque Adar aura des valeurs de distance en fonction du temps.

2. Utilisation d'Adar

Une fois la carte Arduino prête, il est très important de la déconnecter de l'application qui a servi à la programmer.

On ouvre ensuite l'application Adar en cliquant sur le fichier Adar dont l'icône est en forme de robot :



Une fois l'application démarrée, la page suivante apparaît :

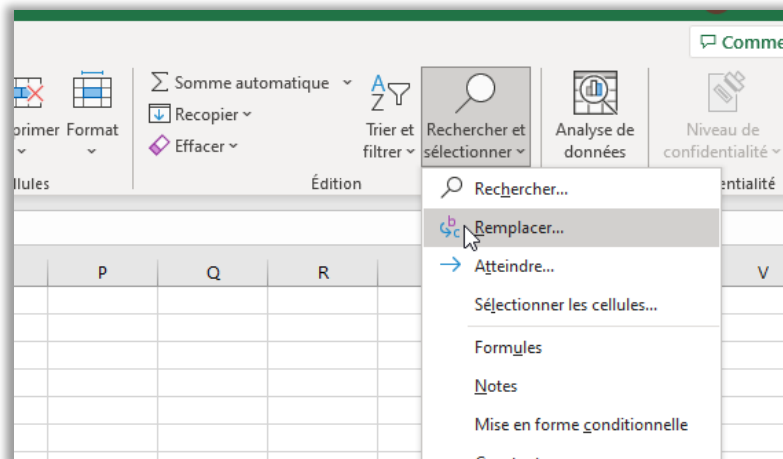


Le cheminement est alors assez simple. Les boutons changent de couleur au fur et à mesure qu'on peut les utiliser.

1. Cliquer sur *Afficher la liste des ports*.
2. Dans la liste déroulante, choisir le port de la carte de type Arduino. Si aucun port n'apparaît, s'assurer que la carte Arduino est bien branchée dans l'ordinateur.
3. Cliquer sur *Connexion*.
4. Préparer le montage pour faire l'expérience.
5. Lorsque tout est prêt, appuyer sur le bouton *Débuter*.
6. La collecte des données commence.
7. Pour arrêter la collecte des données, cliquer sur le bouton *Arrêter*.
8. Pour copier les valeurs dans le presse-papiers, cliquer sur le bouton *Copier*. On peut alors les coller dans un logiciel de calcul, Excel, par exemple, ou un autre.
9. Le bouton *Effacer* efface les données du tableau et du graphique. Il n'est pas possible d'annuler cette étape une fois qu'elle est faite.
10. On peut recommencer la collecte des données en cliquant sur *Débuter* à nouveau et ainsi de suite.

Notes :

1. Il est à noter que si l'expérience utilise une valeur de temps donnée par la carte Arduino, le temps ne reviendra pas à 0s lorsque l'on clique sur *Débuter* une 2^e fois. En effet, le chronomètre utilisé par Arduino revient à 0 uniquement lorsque la carte est mise hors tension. Pour remettre le chronomètre à 0 ou presque, cliquer sur *Déconnexion* et ensuite sur *Connexion* avant de cliquer sur *Débuter* à nouveau.
2. Les données qui sont copiées dans le presse-papiers utilisent le point et non la virgule comme séparateur décimal. Cependant, Excel utilise généralement la virgule. Pour faire le changement rapidement, aller dans *Rechercher et sélectionner* et choisir *Remplacer*.



Dans la fenêtre qui s'ouvre, mettre un point « . » dans la partie *Rechercher* et une virgule « , » dans la partie *Remplacer* et cliquer ensuite par *Remplacer tout*. Tous les points du document seront remplacés par des virgules.

