



## Sommaire

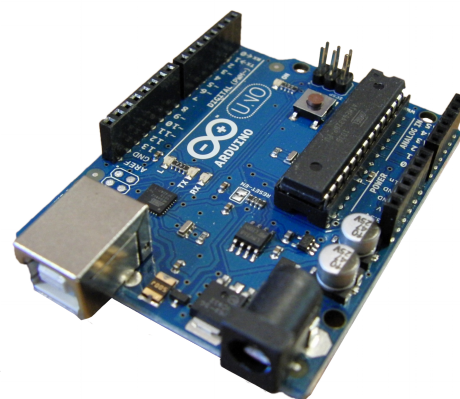
1. Introduction.....	1
2. Arduino / Genuino.....	2
2.1. Connexions.....	2
2.2. Programmation.....	2
3. Programmation Blockly@rduino.....	3
3.1. Lancement de l'EDI Arduino.....	3
3.2. Appel de l'éditeur Blockly@rduino.....	3
3.3. Configuration Blockly@rduino.....	4
3.4. Découverte.....	5
3.5. Description du programme en blocs.....	6
3.6. Exécution.....	7
3.7. Éléments de langage.....	7
4. Utilisation de l'IDE Arduino.....	8
4.1. Procédure d'utilisation.....	8
4.2. Résolution de problèmes : Erreur au téléversement.....	8
5. Installation par le professeur.....	9
5.1. Résolution de problèmes.....	9
6. Configuration Blockly@rduino.....	10

## 1. INTRODUCTION

Arduino est une carte électronique à **microprocesseur** permettant le prototypage de traitements du signal et de l'information. Ce matériel peut être **programmé** afin d'acquérir, **traiter**, analyser et produire des **signaux électriques**. Il permet ainsi d'effectuer des tâches très diverses comme :

- la mesure de grandeurs physiques pour réaliser des [capteurs](#),
- le pilotage d'une automobile, d'un [robot](#), ...
- la [domotique](#) (le contrôle des appareils domestiques - éclairage, chauffage...),
- etc.

En fait ce n'est qu'une sorte de calculatrice programmable qui va lire des informations externes provenant de capteurs, calculer, puis écrire les résultats sur des pré-actionneurs (distributeurs d'énergie) pour commander toutes sortes d'appareils.





## 2. ARDUINO / GENUINO

La carte Arduino/Genuino UNO basée sur un ATMEGA 328 coûte environ 20 €. Les cartes d'extension de type SHIELD coûtent de 30 à 70 €.

!! CE MATÉRIEL EST FRAGILE ET NE VOUS SERA CONFIE QUE LORSQUE CELA SERA NÉCESSAIRE !!

SI VOUS L'ENDOMMAGEZ CE SERA UNE FAUTE ET NON PAS UNE ERREUR, PARCE QUE VOUS N'AUREZ RESPECTÉ NI LES CONSIGNES NI LE MATÉRIEL ! VOUS DEVREZ DONC REMPLACER CE MATÉRIEL À VOS FRAIS !

### 2.1. Connexions

La carte Arduino UNO se connecte sur un ordinateur (PC Windows, PC Linux, Mac) par un cordon standard USB. La carte peut être auto-alimentée par le port USB. Le port USB de la carte ARDUINO est géré par l'ordinateur comme un port série virtuel (COMx).

#### A la connexion par un élève :

Si une boîte de dialogue d'installation de périphérique apparaît :

- NE PAS annuler l'installation de périphérique,
- NE PAS demander l'accès à internet pour cette fois, le professeur a probablement installé le pilote,
- sélectionner l'*installation automatique*.

#### A la première connexion USB (généralement par le professeur)

Si le pilote Arduino est requis (ou depuis le gestionnaire de périphériques), demander la recherche de pilote dans le dossier \drivers de l'application Arduino. Exemple :

<C:\Program files\Arduino\Drivers>

### 2.2. Programmation

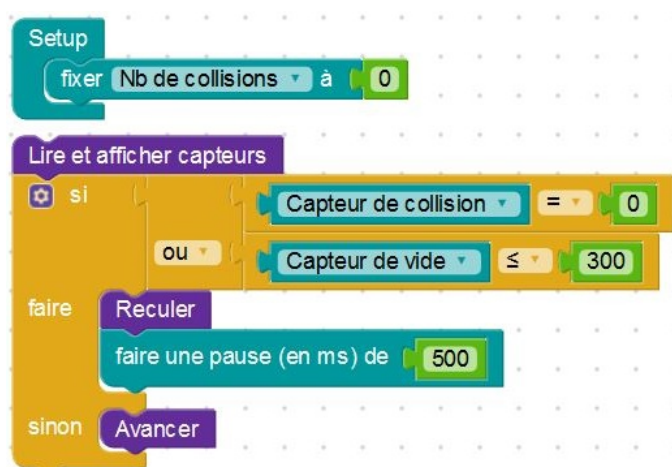
Il existe différents outils de développement d'un système à microprocesseur :

Éditeur de lignes de programmation :  
**IDE Arduino**



- Saisie principalement au clavier
- En général syntaxe anglophone

Éditeur graphique (blocs)  
**Ardublock**



- Saisie graphique à la souris
- L'éditeur peut être en français



## 3. PROGRAMMATION BLOCKLY@ARDUINO

### 3.1. Lancement de l'EDI Arduino

Un EDI est un **Environnement de Développement Intégré**, qui permet

- la saisie des lignes de programme,
- la compilation (conversion du texte du programme en codes microprocesseur),
- le téléversement du programme compilé dans la cible, la carte électronique,
- l'utilisation de divers outils d'écriture de programmes et de tests.

- Copier le raccourci suivant sur votre bureau :  
**Ma Classe\Documents en consultation\Exploration\_SI\Information\Arduino\Blockly@rduino**
- Lancer l'IDE Arduino, un sketch vierge s'ouvre
- *A la première utilisation seulement* :
  - Menu **Outils** | **Type de carte** | **Arduino Uno**
  - Menu **Outils** | **Port série** | **ComXX**  
(en général utiliser un port supérieur à 2, ex : **COM4**)
- 



### 3.2. Appel de l'éditeur Blockly@rduino

- Appeler le menu Outils | [Blockly@rduino](#)  
Une nouvelle fenêtre s'ouvre.



### 3.3. Configuration Blockly@arduino

#### La page d'accueil

... montre une petite vidéo de mise en route lorsque vous êtes en ligne, probablement inaccessible en fonctionnant sur Blockly hors-ligne.

- Cocher la case  
« Don't see this message again »
- Cliquer sur « Understood, thanks »



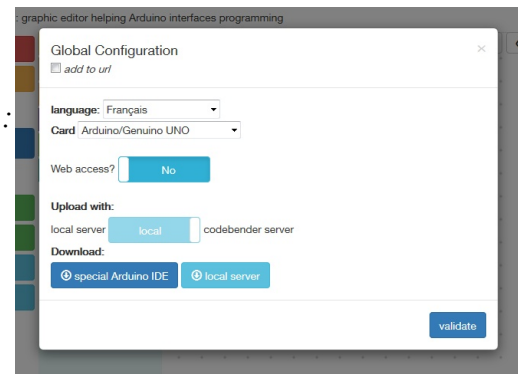
#### Configuration globale

- Cliquer sur le premier bouton (rouge) de la colonne gauche  
« Global configuration ».

*Une boîte de dialogue apparaît :*

- Sélectionner
  - Language = « Français »
  - Card = « Arduino/Genuino UNO »
- Valider

La page se recharge en français, et la carte Arduino sélectionnée apparaît sous le bouton de configuration.



#### Configurer les blocs

- Cliquer sur le bouton orange de la colonne gauche  
« configurer les blocs ».

*Une boîte de dialogue apparaît :*

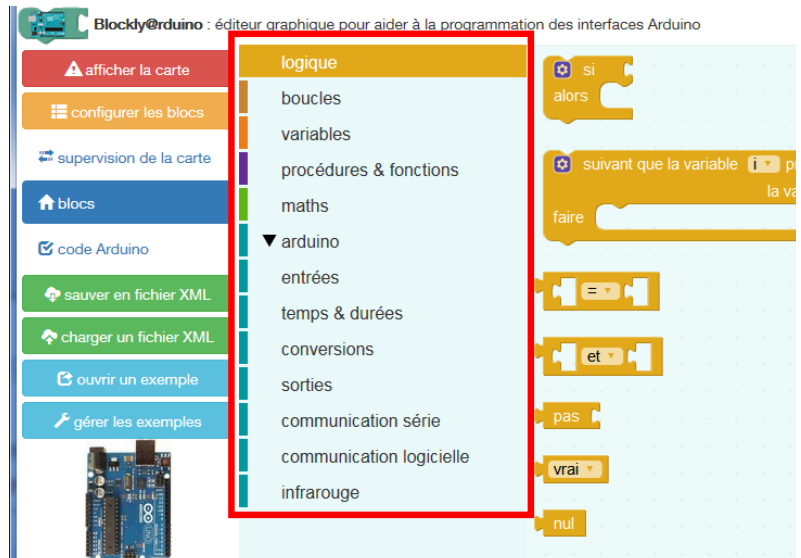
- Sélectionner la catégorie « Arduino pour débutants »
- Valider



### 3.4. Découverte

#### Blocs

- Cliquer à gauche sur la section « blocs » ;
- Cliquer au centre sur les différentes rubriques
  - logique,
  - boucles,
  - variables, ...
  - y compris en développant la rubrique « arduino »,
- Lire attentivement les contenus et formes des blocs fonctionnels proposés à droite ;
- Prêter une attention particulière aux blocs « sorties » qui contiennent les actions sur les « sorties numériques ».



#### Code Arduino

- Cliquer à gauche sur la section « code Arduino » ;
- Apparaît la structure minimale de programme nécessaire pour l'arduino :

Une fonction d'initialisation `setup()` qui n'est appelée et exécutée qu'une seule fois au lancement du programme :

```
void setup() {
}
```

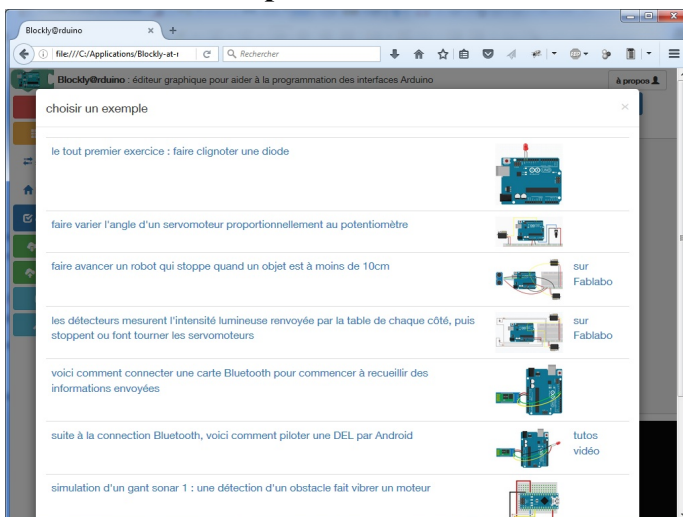
Une fonction d'exécution `loop()` qui est exécutée sans arrêt en boucle :

```
void loop() {
}
```

#### Fenêtre de compilation

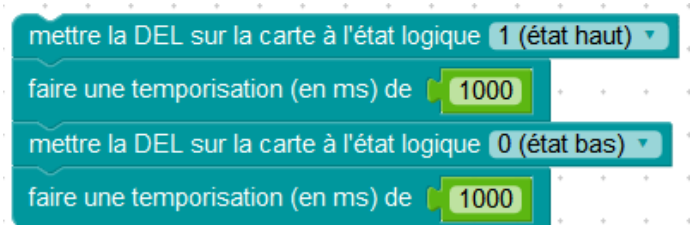
- Si la partie inférieure de la page affiche un message d'erreur du type « La connexion a échoué » :  
Clic-droit sur cette zone, menu contextuel **Ce cadre | Actualiser le cadre**

#### Ouvrir un exemple



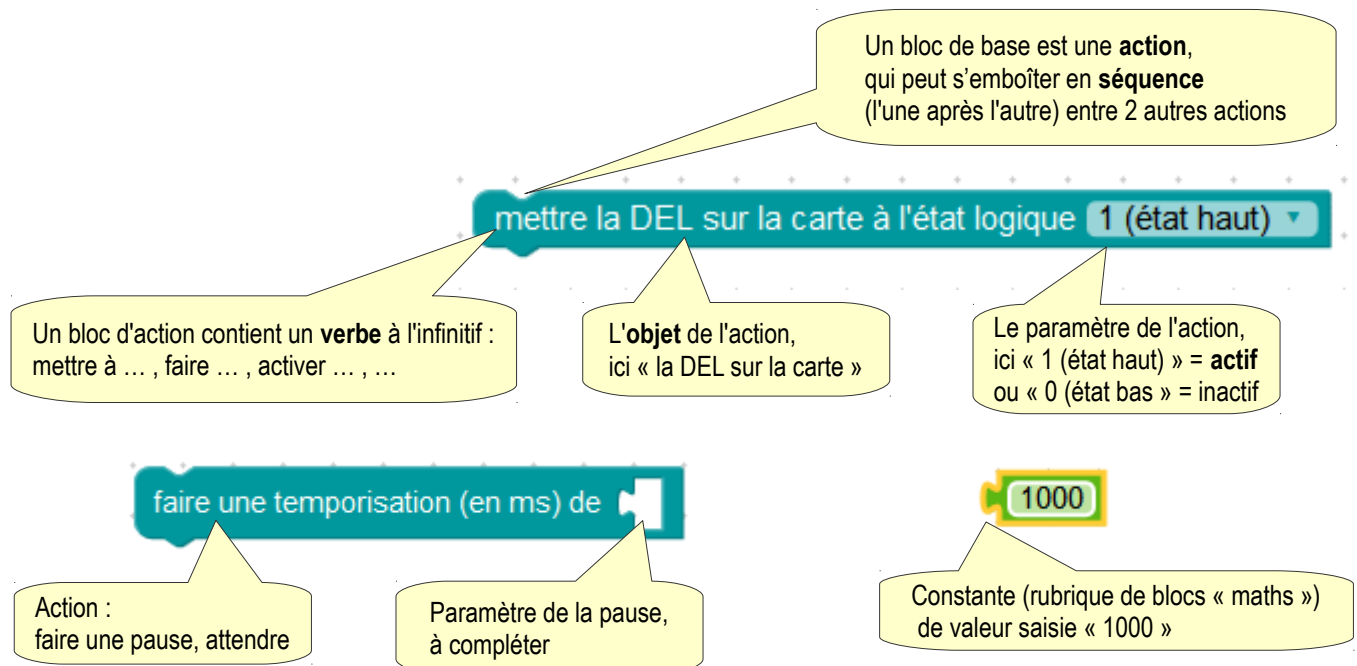
- Cliquer à gauche sur la section « ouvrir un exemple » ;
- Cliquer sur « **le tout premier exercice : faire clignoter une diode** »  
Apparaît une boîte de confirmation « Voulez-vous remplacer les blocs actuels ? »

- Confirmer par « OK »  
Les blocs de programme apparaissent :





### 3.5. Description du programme en blocs



#### Note :

L'ensemble des blocs d'instructions est **répété en boucle sans fin**. Donc ici :

```
On allume la DEL
On attend 1 seconde
On éteint la DEL
On attend 1 seconde
et on reboucle, ce qui crée un clignotement !
```

#### Code Arduino

- Cliquer à gauche sur la section « code Arduino ».
- Voici le code généré automatiquement par BlocklyArduino :

```
void setup() {
  pinMode(13, OUTPUT);
}

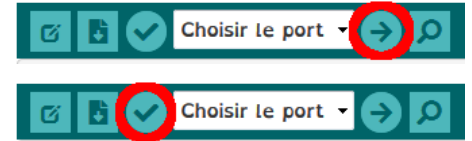
void loop() {
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```



### 3.6. Exécution

Le code est prêt à être transféré dans la carte Arduino.

- Sélectionner si nécessaire le numéro de **port** de communication noté sur « Blockly communication »
- Cliquer sur le bouton **Téléverser dans l'Arduino** pour un test rapide et aperçu du résultat ;
- Cliquer sur le bouton **Vérification par EDI Arduino** pour des vérifications plus approfondies par l'EDI original Arduino.



#### Si le téléversement Blockly ne fonctionne pas

Ce code est prêt à être transféré dans la carte Arduino.

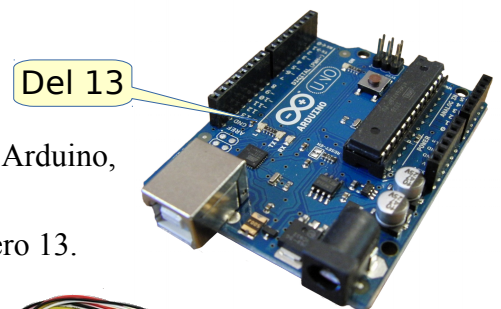
- Sélectionner tout le code, Clic-droit « **Copier** » ou au clavier **Ctrl-C**
- Basculer sur l'EDI Arduino (Raccourci sur le bureau), coller le programmer
- Compiler et téléverser depuis l'EDI Arduino : voir page suivante.

### 3.7. Éléments de langage

#### « DEL sur la carte »

C'est la Diode ElectroLuminescente orange présente sur la carte Arduino, donc visible sous la carte de connecteurs GROVE.

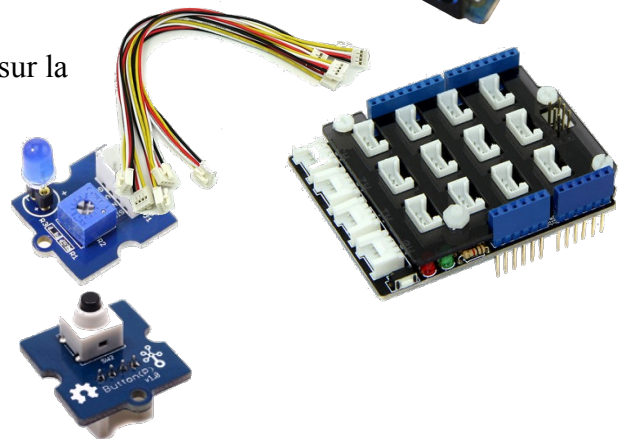
Elle est connectée sur la sortie logique (ou « numérique ») numéro 13.



#### Carte GROVE

C'est une carte de connexion d'entrées/sorties enfichée sur la carte Arduino. On peut y connecter :

- des DELs  
sur les sorties de **broches numériques** N° 2 à 8  
pour les allumer (mettre à « 1 »)  
ou les éteindre (mettre à « 0 ») ;
- des boutons poussoirs  
sur les **entrées numériques** N° 2 à 8  
pour lire leur état logique « 1 » (appuyé)  
ou « 0 » (relâché) ;



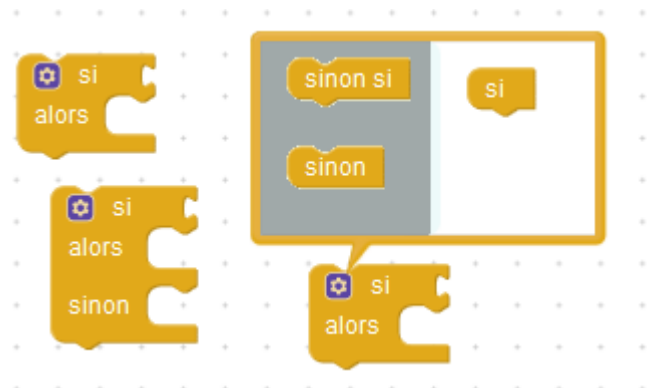
#### Logique

Différentes structures logiques sont disponibles :

- **Si** (la condition est vraie)  
**alors** ... (faire ...)
- **Si** (la condition est vraie)  
**alors** ... (faire ...)  
**sinon** ... (faire...)
- ...

Pour modifier la structure :




- Cliquer sur la petite roue dentée « Paramètres »,  
une bulle rectangulaire apparaît ;
- faire glisser les options voulues de la partie gauche de la bulle à la partie droite sous le « **si** ».





## 4. UTILISATION DE L'IDE ARDUINO

### 4.1. Procédure d'utilisation

- COLLER le programme copié depuis BlocklyArduino
-  Menu **Croquis | Vérifier / Compiler** (Ctrl-R)
-  Menu **Fichier | Téléverser** (Ctrl-U)
-  **Fichier | Enregistrer** (Ctrl-S) le programme lorsque vous êtes satisfait des essais.

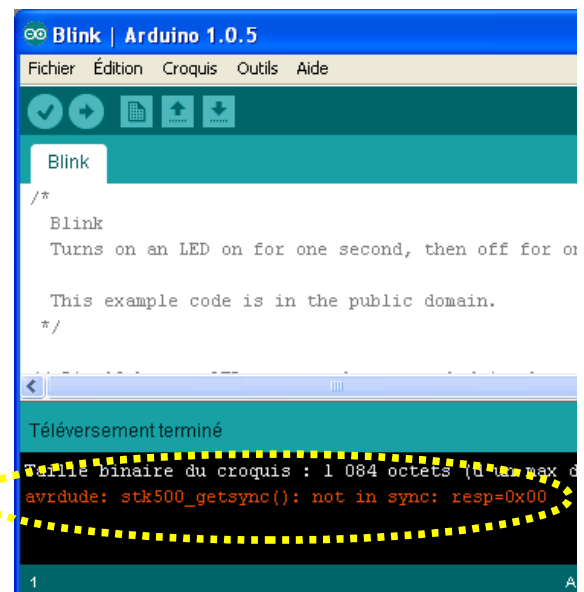
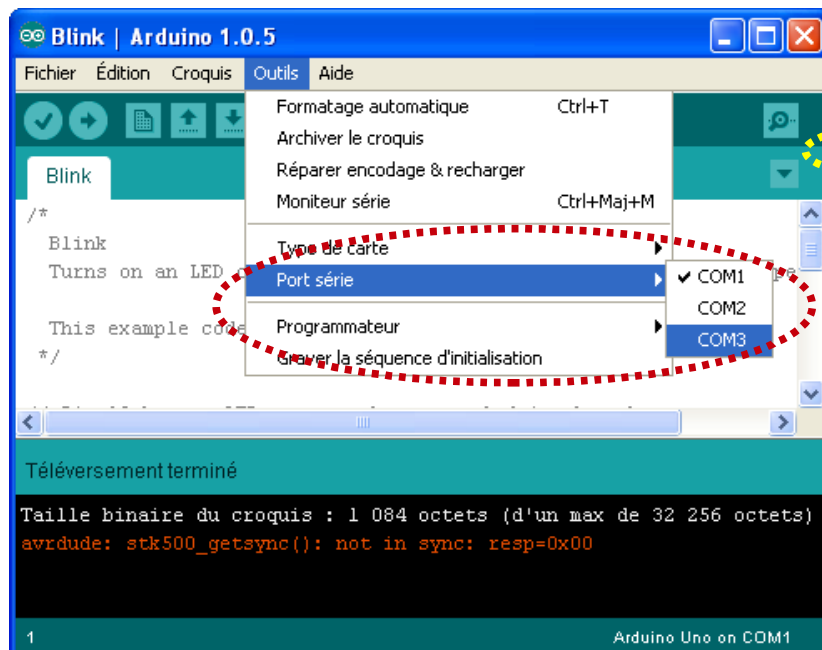
### 4.2. Résolution de problèmes : Erreur au téléversement

**avr\_dude: stk500\_getsync: not in sync: resp=0x00**

Ici la compilation s'est bien déroulée  
(Taille binaire du croquis: xxxx octets),  
mais un problème est apparu lors de l'envoi du programme dans  
la carte ARDUINO.

Ceci signifie que le logiciel de développement n'a pas été  
capable de communiquer avec la carte à travers le port USB.

Vérifier alors le numéro de port COM utilisé par le menu  
**Outils | Port série | COMx ...**



Sur un ordinateur de bureau, le port  
USB est probablement attribué à un  
numéro de **COM supérieur ou égal à 3**  
(les ports 1 et 2 étant les ports RS232 de la  
carte mère).