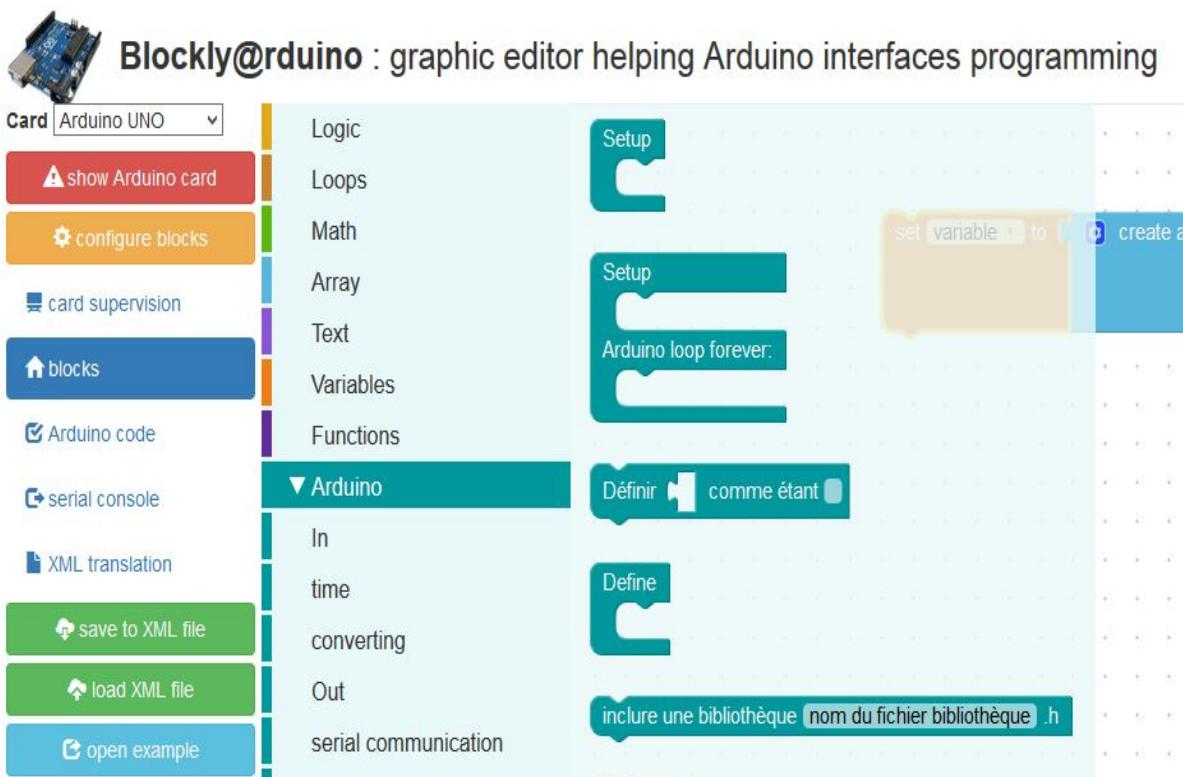


Table des matières

Fonctions	3
IMPORTANT Arduino	4
Démos	4
Utiliser en local dans son navigateur web	4
Utilisation en 5 temps	4
Auteurs et contributeurs	4

Blockly@rduino est un programme web, un **webware graphique permettant la programmation graphique pour Arduino et sa traduction en code**.



Blockly@rduino est basé sur [Blockly](#), un éditeur de programme graphique par une interface web. Il fournit un seul type de traduction de la programmation graphique (*contrairement à Blockly qui traduit en Python, en JavaScript, etc et permet de le simuler en ligne : [exemple](#)*), pour générer du code à copier coller dans l'IDE Arduino.

Blockly@rduino possède **de nombreuses** bibliothèque de blocs pour les modules Grove, TechnoZone51, DFRobot, etc particulièrement adaptées à la découverte des micro-contrôleurs et aux expérimentations de systèmes mécatroniques.

Cette version a été francisée et réorganisée pour la faire utiliser au mieux par des élèves débutants dans le code.

Toutes les remarques sont les bienvenues pour améliorer la syntaxe !

Lien de téléchargement : <https://github.com/technologiescollege/Blockly-at-rduino>

Fonctions

- programmation pour Arduino en drag'n'drop graphique,
- génère un code totalement compatible avec le code Arduino,
- interaction des cartes Arduino grâce aux modules Grove rajoutés : plus de 10 blocs spécifiques à des capteurs/actionneurs Grove,
- possibilité de charger automatiquement des fichiers sur le site grâce aux paramètres de l'adresse URL (*voir les démos ci-dessous*).

IMPORTANT Arduino

Pour le téléversement dans la carte Arduino directement depuis le navigateur, il faut installer le plugin Codebender : <https://codebender.cc/static/plugin> ou bien copier-coller le code produit dans l'EDI Arduino (<https://github.com/technologiescollege/arduino>).

Démos

Blockly@rduino est un outil web, un webware par référence aux software, hardware, etc. Vous pouvez le tester/utiliser à l'adresse suivante :

<http://technologiescollege.github.io/Blockly-at-rduino>

Vous pouvez donc aussi lier directement l'ouverture de la page à l'ouverture d'un fichier XML hébergé (ou tout en local) :

- [démonstration 1 : faire clignoter une diode](#)
- [démonstration 2 : faire varier un servo-moteur proportionnellement à une consigne de potentiomètre](#)

Le créateur de BlocklyDuino a aussi posté une [vidéo](#).

Utiliser en local dans son navigateur web

Si vous préférez l'utiliser en local, il vous suffit de télécharger le code depuis Github ([Download Zip](#)) et d'ouvrir le fichier `index.html` dans votre navigateur (**de préférence Firefox**).

Vous pouvez l'héberger sur un serveur, et ainsi y faire accéder de nombreux postes en pointant vers ce fichier 'index.html'.

Utilisation en 5 temps

1. utiliser votre navigateur pour ouvrir Blockly@rduino grâce au fichier 'index.html'
2. sélectionner la carte et la catégorie de blocs que l'on veut utiliser
3. créer son programme Arduino par drag'n'drop (*cliqué-déplacé*) des blocs
4. sélectionner l'onglet 'Arduino' pour en copier le contenu dans l'[IDE Arduino](#)
5. presser le bouton 'Téléverser' pour transférer le code dans la mémoire de la carte Arduino

Auteurs et contributeurs

Fred Lin (@gasolin). Merci à Neil Fraser, Q.Neutron de Blockly <http://code.google.com/p/blockly/>. Merci à l'équipe d'Arduino et Seeeduino pour les blocs spécifiques. Ce projet s'est aussi inspiré de [arduiblock](#) et [modkit](#). Améliorations par Julien Rat. Modifications, simplifications et francisation par Sébastien Canet. Fonctionnalités accrues par Olivier Métayer.

Au début des temps il y a eu [StarLogo](#), qui a influencé MIT Scratch1 (*dont la fameuse 1.4 qui a servi de base au fameux S4A, mais pour cela il leur a fallu taper dans le code*), puis une autre équipe MIT a créé le [MIT App Inventor](#) pour les applications Android, ce qui a influencé la création de [Blockly](#). L'idée est de fournir la base d'une programmation visuelle à une traduction vers du code (Python, Java, etc) ou autre chose....

Ils sont même allés jusqu'à proposer un créateur graphique de blocs :

<https://developers.google.com/blockly/custom-blocks/block-factory>

Du coup, un bidouilleur appelé Gasolin a créé le premier projet autour de Blockly en [BlocklyDuino](#). Mais comme il ne faisait plus de mises à jour, ne répondait pas aux mails, je suis reparti de la dernière version de Blockly (je ne suis pas le seul, vous trouverez plein de projets identiques sur le web), j'ai intégré du code de Gasolin, j'ai pris quelques bières et hop ! J'ai appelé un pote à la rescousse pour l'aspect graphique, et hop 2 ! C'est encore en chantier, mais fonctionnel.

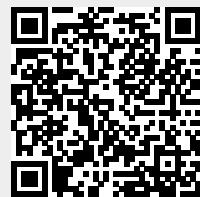
Pour du virtuel pur, il y aurait un gros travail d'interception de code et de conception graphique....c'est dans une future TODO-list....

Pour l'instant un fichier (le fameux compilerflasher.js) intercepte le code et l'envoie au site [CodeBender](#) grâce au plugin de ton navigateur :

Blockly → bloc graphique → compilerflasher.js → plugin codebender → site codebender → code Arduino → carte Arduino

Qu'un site extérieur accède à votre matériel représente un peu une faille de sécurité...donc le Graal actuel sur lequel tout le monde bosse est un pseudo-codebender local.

From:
<https://wiki.libreduc.cc/> - **LibrEduc**



Permanent link:
https://wiki.libreduc.cc/fr:arduino:blockly_rduino

Last update: **2025/01/16 20:24**