

Table des matières

1. Conversion Blockly@rduino → µBlockly	3
1.1. Usage	3
1.2. Comportement	3
1.3. Variable d'environnement	3
1.4. Fichiers concernés	3
1.5. Limites connues	4
1.6. Voir aussi	4

1. Conversion Blockly@rduino → µBlockly

Ce document décrit le script `npm run convert-rduino`, qui transforme une sauvegarde Blockly@rduino (XML / .B@) en workspace JSON chargeable dans µBlockly.

1.1. Usage

```
npm run convert-rduino -- <fichier-entree> [fichier-sortie.json]
```

Exemple :

```
npm run convert-rduino -- chemin/vers/projet.xml out/workspace.json
```

Exemple (si le dépôt voisin Blockly@rduino est présent, comme pour les tests) :

```
npm run convert-rduino -- ../Blockly-at-rduino/examples/blink/blink.xml  
out/workspace.json
```

- **Entrée** : .xml, .B@ ou tout export XML Blockly@rduino (« enregistrer projet »).
- **Sortie** : JSON compatible avec **Ouvrir** dans µBlockly (workspace.json par défaut si le second argument est omis).

1.2. Comportement

Cas	Traitement
Blocs Arduino de base	Mappés vers les types µBlockly via <code>scripts/lib/rduino-block-map.mjs</code>
Blocs « définitions » orange (base_define_bloc)	Dépliés en blocs racine
Piles sans setup / loop	Enveloppées dans <code>board_base_setup_loop</code> (corps loop) sauf si <code>UCB_NO_WRAP=1</code>
Extensions (Grove, mBot, TechnoZone, ...)	Conservées sous leur nom d'origine si non mappées — peuvent ne pas charger ; voir <code>unmappedBlockTypes</code> dans la sortie

1.3. Variable d'environnement

```
UCB_NO_WRAP=1 npm run convert-rduino -- projet.xml out.json
```

Ne pas envelopper les blocs orphelins dans setup/loop.

1.4. Fichiers concernés

Fichier	Rôle
scripts/convert-rduino-workspace.mjs	Point d'entrée CLI
scripts/lib/rduino-block-map.mjs	Table de correspondance des types de blocs
scripts/lib/rduino-xml-to-ucb-json.mjs	Parsing XML → JSON workspace
tests/rduino-convert.test.ts	Tests (blink, analogique, Starter_Kit_prj7)

1.5. Limites connues

- Couverture limitée au **cœur Arduino** ; les extensions matérielles nécessitent d'étendre la table de mapping.
- Les projets très anciens ou fortement personnalisés peuvent exiger une reprise manuelle de certains blocs.

1.6. Voir aussi

- [Architecture](#) — scripts npm ;
- [Intégration](#) — intégration du dépôt.

From:

<https://wiki.libreeduc.cc/> - **LibreEduc**

Permanent link:

https://wiki.libreeduc.cc/fr:arduino:ucblockly:convert_rduino

Last update: **2026/08/11 00:18**

