Table des matières

1. Contributing to µcBlockly 3
1.1. Before you start 3
1.2. Common tasks 3
1.3. Reporting issues and pull requests 4

1. Contributing to µcBlockly

Thank you for considering contributing. This short guide will help you get started.

1.1. Before you start

Compatibility: µcBlockly **v3** requires **Blockly 13** (`blockly ^13` and aligned `@blockly/*` plugins). See the root README and KERNEL.md.

1. **Read the architecture**

See ARCHITECTURE.md for:

- Source layout (`src/` and subfolders: `core/`, `host/`, `demo/`)
- Conventions (comments, TSDoc / TypeDoc, block definitions vs code generation)
- Where to change what (new blocks, types, toolbox, boards, languages, themes)

1. **Set up the project**

- `npm install`
- `npm run dev` for development with hot reload
- `npm run build` for production build (demo + core bundle)
- `npm run build:core` for the embeddable kernel only
- `npm run docs` to generate API documentation (TypeDoc) in `docs/api/` — see TYPEDOC.md

1. **Quality checks before a PR**

- `npm run lint` — ESLint (including `eslint-plugin-tsdoc` for comment syntax)
- `npm run test` — Arduino generator and Blockly@rduino converter tests
- `npm run check-translations` — after editing `src/languages/`
- `npm run build:lib` — ensure TypeScript compiles

1. **Code style**

- Documentation comments (**TSDoc**, consumed by TypeDoc) in **English**
- Use `@packageDocumentation`, `@description`, `@remarks`, `@param`, `@returns` for public APIs (see TYPEDOC.md)

1.2. Common tasks

Task	Where to look
Add or change a block (UI)	`src/categories/blockly/` + `index.ts`
Add or change code generation for a block	`src/generators/arduino/blocks/` + `src/generators/arduino.ts`
Add block types / compatibility	`src/generators/arduino/block_types/`, `block_types.ts`, `block_types_registry.ts` — TYPES_ET_COMPATIBILITES.md
Change toolbox or categories	`src/toolbox.ts`

Task	Where to look
Add a board or change pins	`src/boards.ts`
Add or edit a language	`src/languages/` — then `npm run check-translations`
Convert Blockly@rduino project	`npm run convert-rduino` — see CONVERT_RDUINO.md
Regenerate functions inventory	`npm run extract-functions` → `docs/functions-list.md`

1.3. Reporting issues and pull requests

- **Bugs and feature requests**: use the GitHub issue tracker.
- **Pull requests**: welcome. Please keep changes focused and reference any related issue.

Keeping ARCHITECTURE.md and this file up to date when you change the project layout or workflow is appreciated.

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

Permanent link:
https://wiki.libreduc.cc/fr:arduino:ucblockly:contributing

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