mpideXcode

Embedded Computing on Xcode

Installation Guide

Documentation http://embedXcode.weebly.com/

GitHub repository http://github.com/rei-vilo/embedXcode

Contact http://embedxcode.weebly.com/contact.html

How to Help

Contribute and help me buy books on Xcode through my Amazon Wish List.

Donate and help me buy embedded computing material via PayPal.

Improve the template on GitHub.

After having played with embedded computing platforms for a while, I was looking for one single IDE and a better one.

*Code-sense, colour syntaxing, check-as-you-type, click-to-error, self-documentation and tool-tip texts are some niceties **Xcode** brings.*

*The **embedXcode** project allows to use Xcode to develop for Arduino, chipKIT, LaunchPad MSP430, Wiring and LeafLabs Maple platforms.*

Because embedXcode relies on a modular design and on the boards IDEs for easier installation, virtually any board with a Processing-based Wiring-derived IDE can be implemented.

*As I'm not a professional, **please contribute** and help me buy books and material, and feel free to improve the template. Thanks!*

Summary

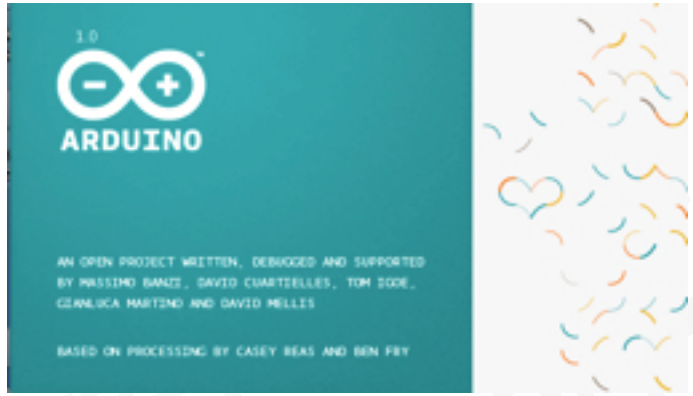
1. Install the Template

Before installing the template, you need to install Xcode and at least one IDE.

1.1. Install Xcode

Install Xcode from the DVD or download it from the Mac App Store.

1.2. Install the IDEs for the Boards



If you plan to use Arduino boards:

- Download and install Arduino 0023 or Arduino 1.0.
- Launch it.
- Define the path of the sketchbook in the menu `Aduino > Preferences > Sketchbook location`.

embedXcode identifies the version of Arduino automatically.

As a matter of fact, Wiring is considered as the framework of reference for embedded computing.

Arduino 0023 should be preferred over Arduino 1.0 because Arduino 0023 is Wiring compliant. So is chipKIT MPIDE 0023.

Arduino 1.0 has introduced many small changes in the syntax which are not compatible with previous release. Energia 1.0 is derived from Arduino 1.0.

So I strongly recommend to pick the release of Arduino which is compatible with the other platforms you plan to use:

- either Arduino 0023 with chipKIT MPIDE, Wiring and Maple IDES,
- or Arduino 1.0 with LaunchPad Energia IDE.

If you plan to use chipKIT boards:

- Download and install Mpide 0023.
- Launch it.
- Define the path of the sketchbook in the menu `Mpide > Preferences > Sketchbook location`.

If you plan to use Wiring boards:

- Download and install Wiring 1.0.
- Launch it.
- Define the path of the sketchbook in the menu `Wiring > Preferences > Sketchbook location`.

The two following files require to be deleted.

`/Applications/Wiring.app/Contents/Resources/Java/cores/AVR8Bit/program.cpp`

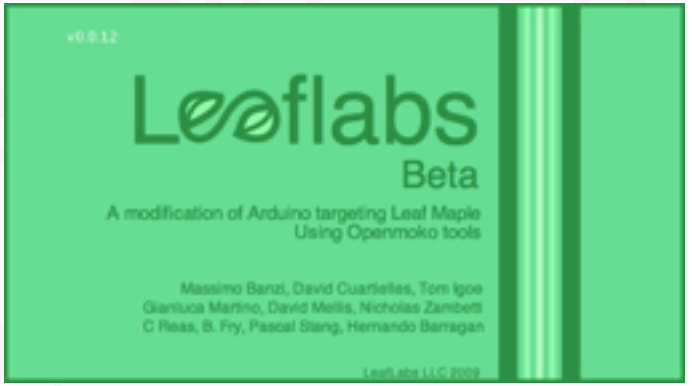
`/Applications/Wiring.app/Contents/Resources/Java/cores/AVR8Bit/makefile`

If you plan to use LaunchPad boards:

- Download and install Energia 1.0.
- Launch it.
- Define the path of the sketchbook in the menu `Energia > Preferences > Sketchbook location`.

If you plan to use Maple boards:

- Download and install MapleIDE.
- Launch it.
- Define the path of the sketchbook in the menu `MapleIDE > Preferences > Sketchbook location`.



The Maple reset script —which sends control signals over the USB-serial connection to restart and enter the bootloader— is written in Python and requires the PySerial library:

- Read the instructions at http://leaflabs.com/docs/unix-toolchain.html#os-x and
- Download PySerial library from http://pypi.python.org/pypi/pyserial.

For other boards with a Processing-based Wiring-derived IDE, the procedure is the same:

- Download and install the corresponding Processing-based Wiring-derived IDE.
- Launch it.
- Define the path of the sketchbook.
- Additionally, develop a specific makefile and adapt the `Step1` makefile.

1.3. Install Optional Tools

If you want to use the self-documentation, please install also:

Install Doxygen to parse the code, looks for comments and generate the HTML pages

I strongly recommend to install DoxyWizard included in the package for an easy tweaking of the parameters.

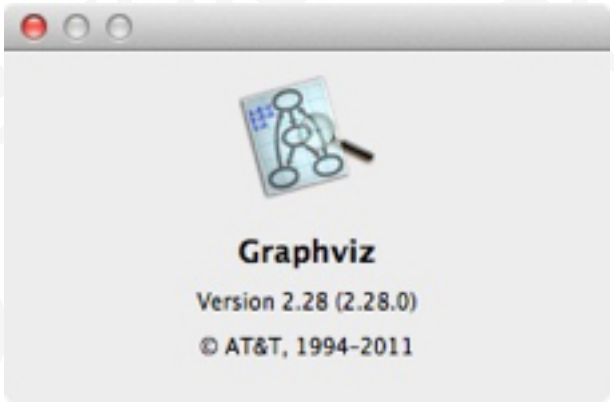
To ease and speed up the writing of the comments, I use the Automator Service Doxygen Helper developed by Fred McCann / Duck Rowing.

By just selecting a function and pressing ⌘⇧**D** cmd-shift-D, the helper generates a template for the comment lines.

Download and install it following the instructions provided on the Doxygen helper page.

```
80
81  ///
82  /// @brief      Description
83  /// @param      a a description
84  /// @param      b b description
85  /// @return     return value description
86  ///
87  uint16_t function(uint16_t a, uint16_t b) {
88    return a + b;
89  }
```

Install DataWiz to add dependency trees



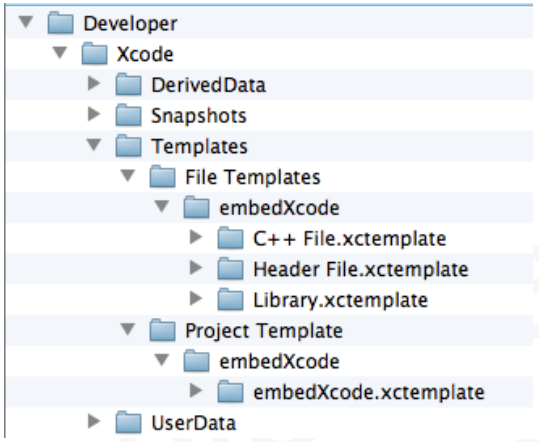
And optionally install TeXShop to build a PDF document from the files Doxygen has generated.

1.4. Install the Template

Now, the template can be installed:

Check and create `~ /Library/Developer/Xcode`.

Copy the folder Templates into `~/Library/Developer/Xcode`

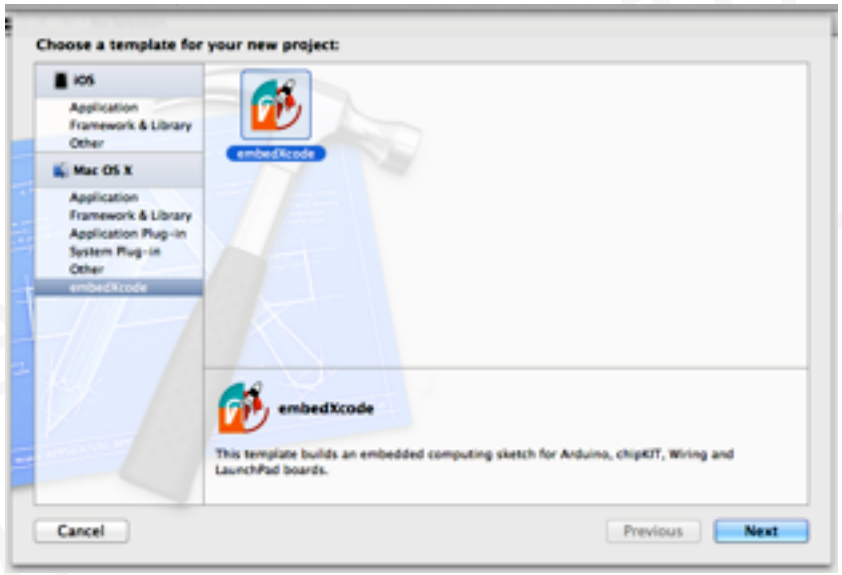


2. Create a New Project

Call the menu `File > New > New Project...` or press ⇧⌘N.

Select `embedXcode > embedXcode`.

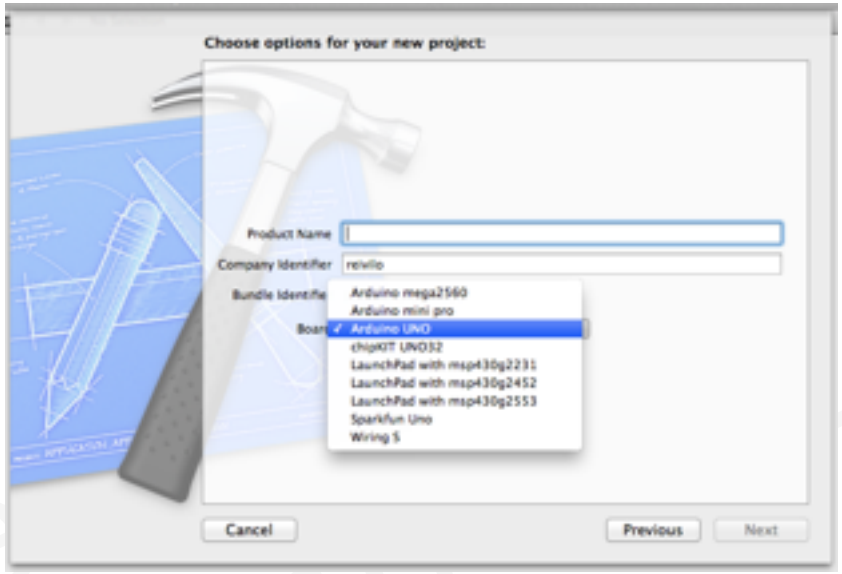
Click on `Next` to proceed to the next step.



Type in the name of the project.

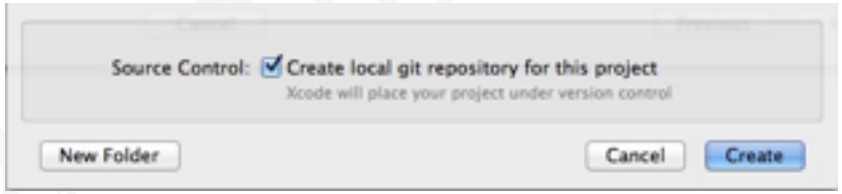
Select the board on the drop-down list.

Click on `Next` to proceed to the last step.



Select the folder where the project is going to be saved and check `Create local git repository for this project` if you want so.

Then click `Create` to confirm and create the project.



3. Configure the Project

The template doesn't define all the parameters, so some of them need to be set manually.

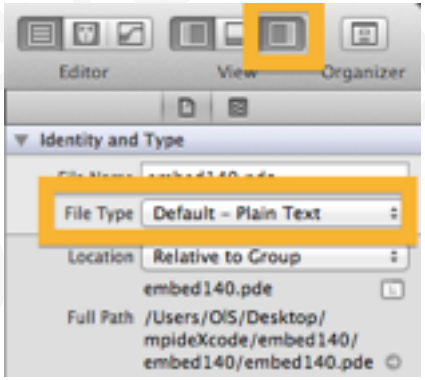
The goal is to have the following work done by the template, provided the right keywords are known.

3.1. Declare Sketch .pde File as C++ File

The sketch `.pde` file is considered as plan text. For code-sense, it should be declared as C++ file.

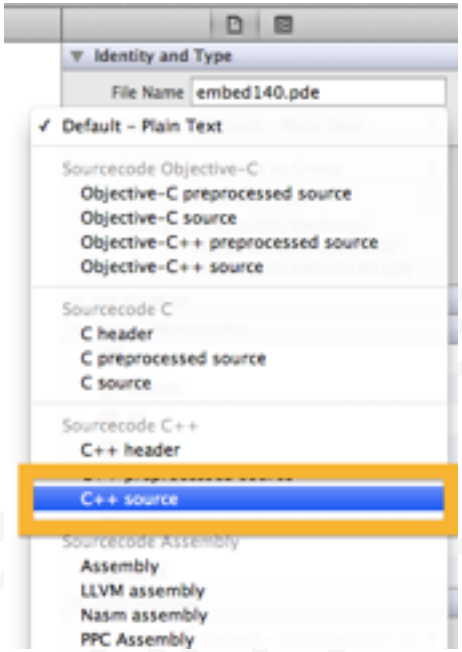
Select the sketch `.pde` file.

In the right-most column,

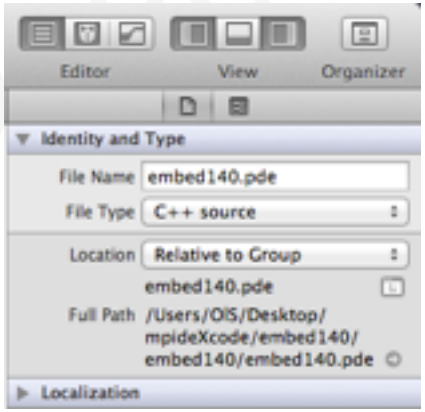


Click on the drop-down list of File Type.

Select `C++ source`.



Now, the sketch is considered as C++ code for code-sense.

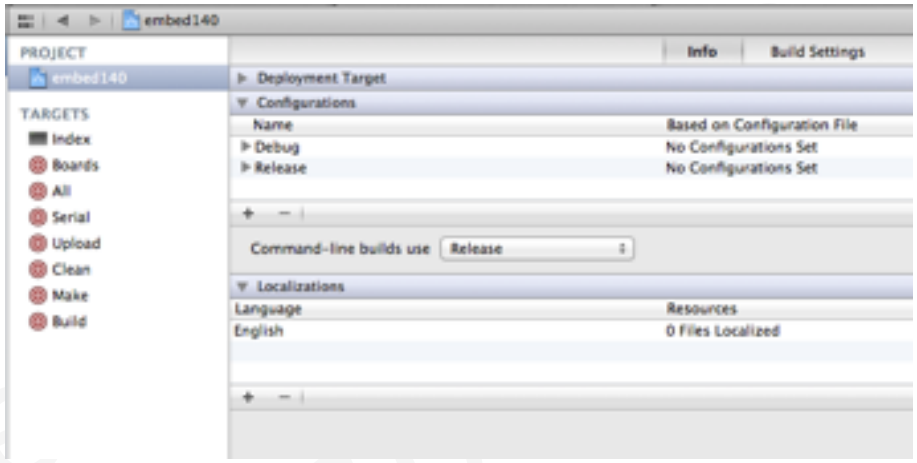


This manual procedure is considered as an issue and reported under #3 Declare PDE File as C++ Source.

3.2. Declare User's Sketchbook

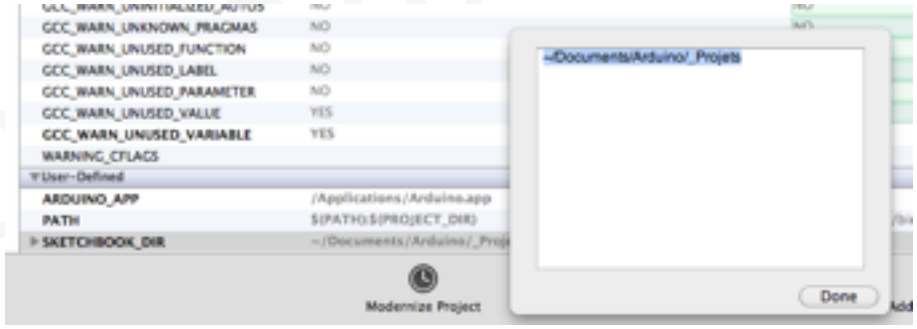
The user's sketchbook is a folder where the user's sketches are saved, among them the libraries in a dedicated sub-folder Libraries.

Select the project and the `Build Settings` pane.



At the very bottom, double-click on `SKETCHBOOK_DIR` and either type in the name of the folder or drag-and-drop it from a Finder window.

The ~ character is accepted.



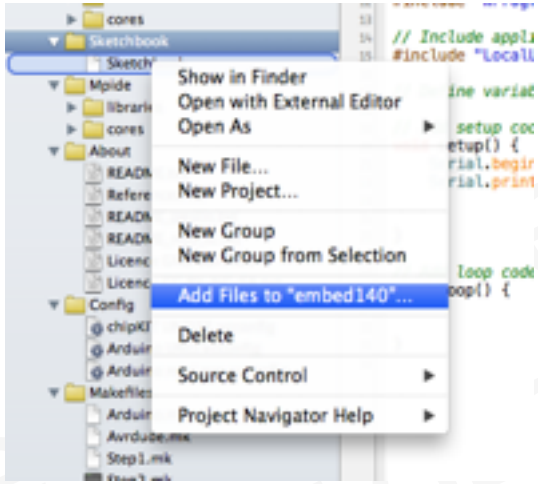
If no sketchbook is defined, `SKETCHBOOK_DIR` takes the value defined for Arduino or chipKIT during the installation process.

3.3. Add User's Libraries

Open the Sketchbook group on the project hierarchy.

Right-click to obtain the contextual menu.

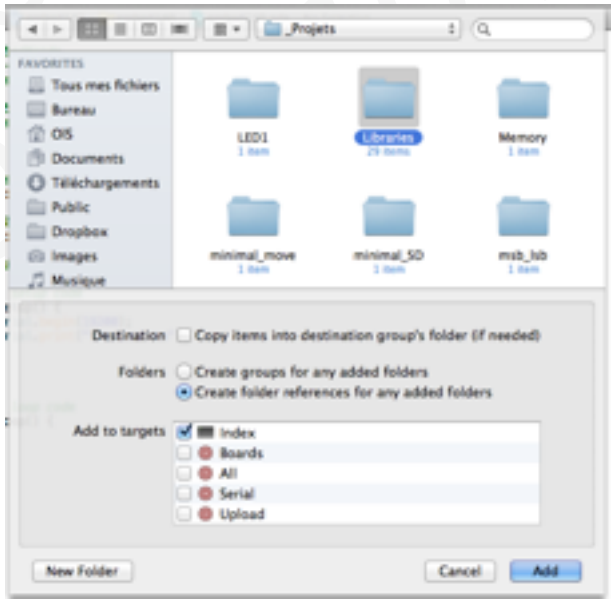
Choose `Add file to...`



Select the Library sub-folder on the sketchbook folder, tick `Add to target > Index` and validate with Add.

Both `Create group for any added folders` and `Create folder references for any added folders` are relevant.

Don't tick `Copy items into destination group's folder (in needed)` to avoid duplicating files.



The project hierarchy shows all your libraries.

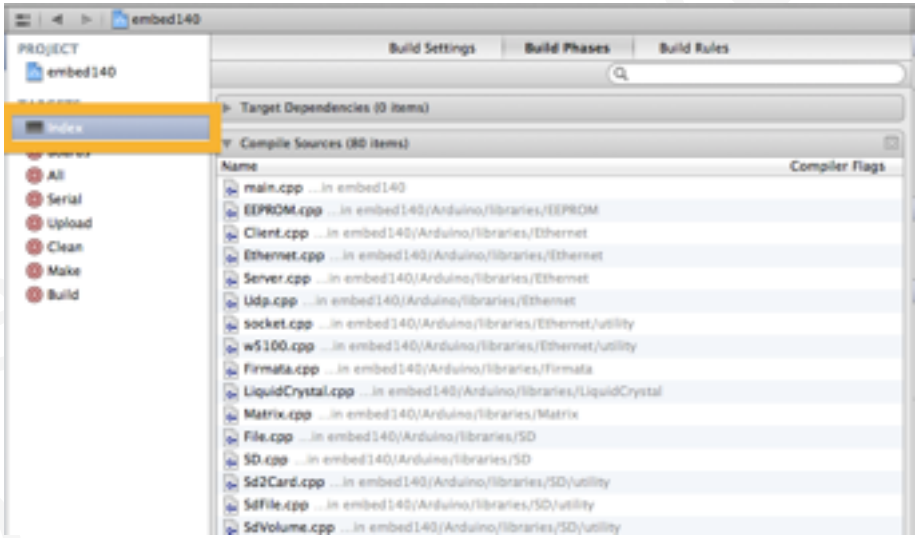
3.4. Declare Sources for Code-Sense

Standard C++ keywords are already known, but not some Arduino and user's library keywords.

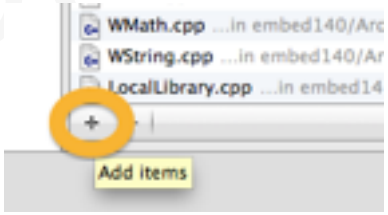
The selection of a board defines the headers for code-sense.

So Xcode needs to be taught where to find them.

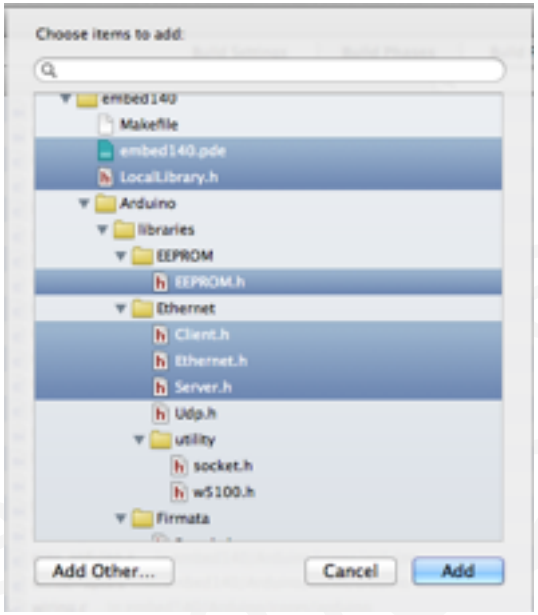
Select the target `Index` and the `Build Phases` pane.



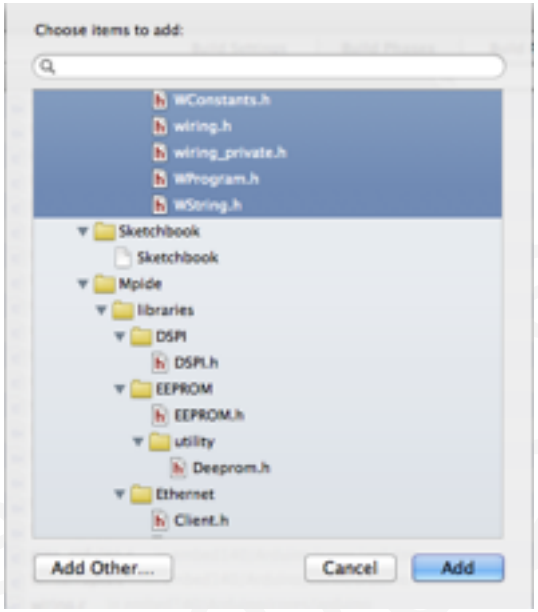
Go a the bottom of the list and click on the + button.



A list shows up.



Select all the `.h` and `.cpp` files and click on Add.



This manual procedure is considered as an issue and reported under #2 Populated Sources List for Code-Sense.

3.5. Define the Directories for the Target

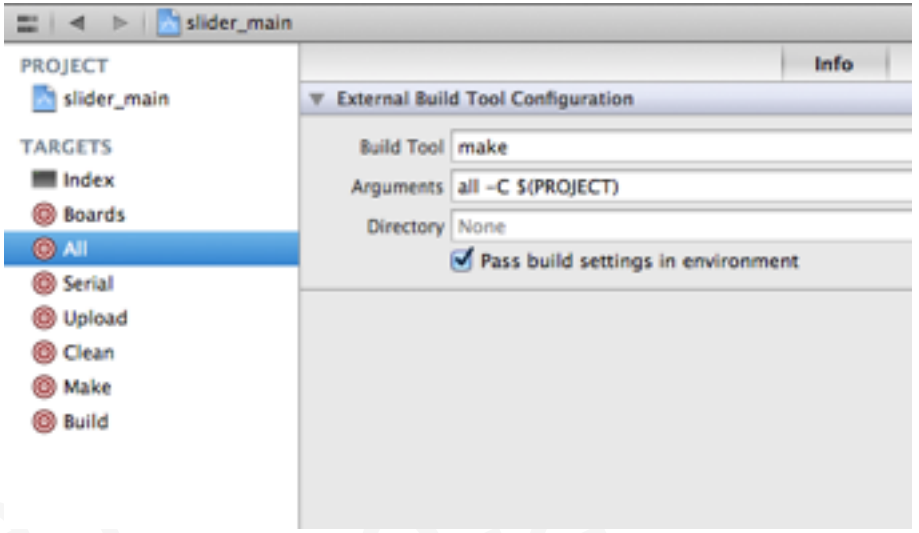
The template doesn't allow to specify the exact directory for the target. In case of an building error, the click-to-error feature may not work properly for the main sketch and the local libraries.

So Xcode needs to be taught where to find them.

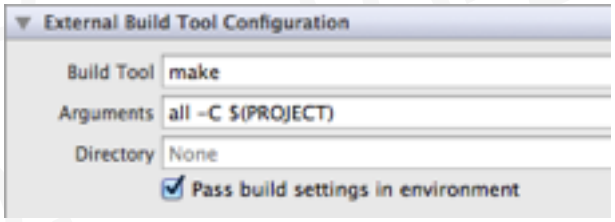
This is an optional procedure.

Select the target `All`.

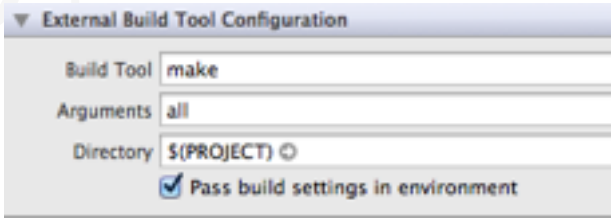
Feel free to update other targets you use often, as `Build` and `Make`.



By default, the template mentions `all -C $(PROJECT)` as arguments.



Remove `-C $(PROJECT)` from the arguments and add `$(PROJECT)` into directory.



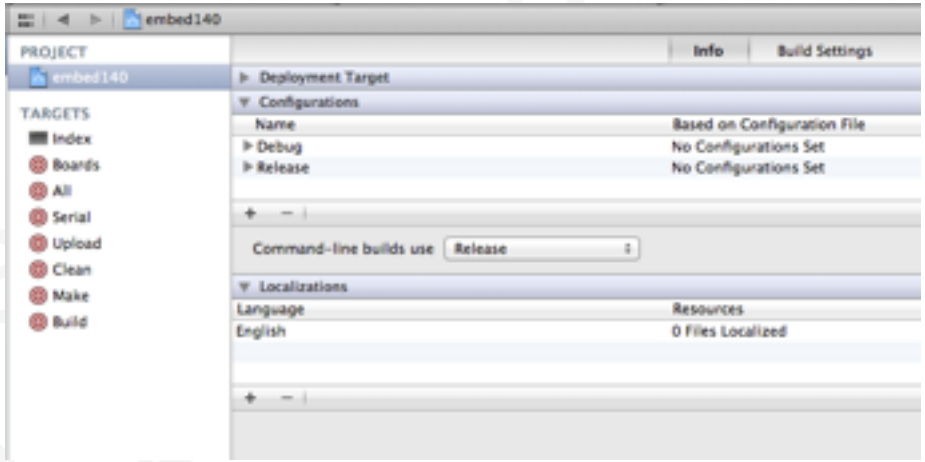
This manual procedure is considered as an issue and reported under #12 Define Directory for a Target.

You're ready now!

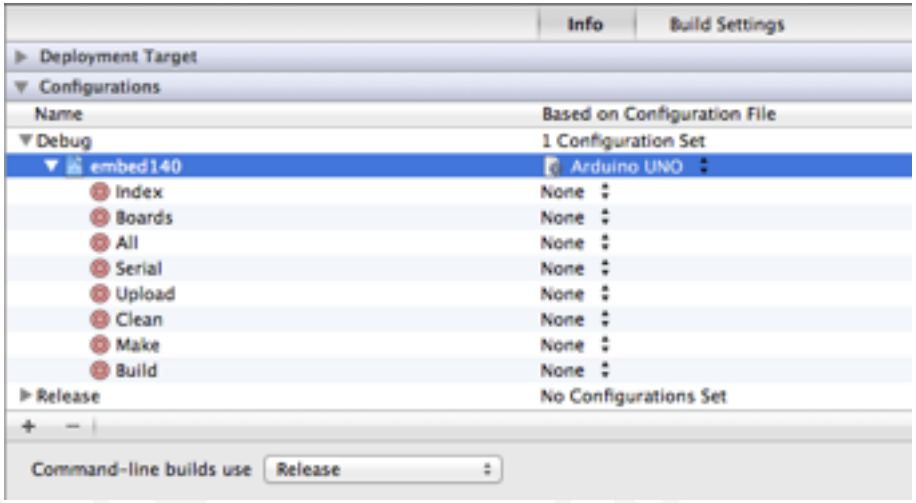
4. Use the Project

4.1. Change the Board

To change the board, select the project and the `Info` pane.

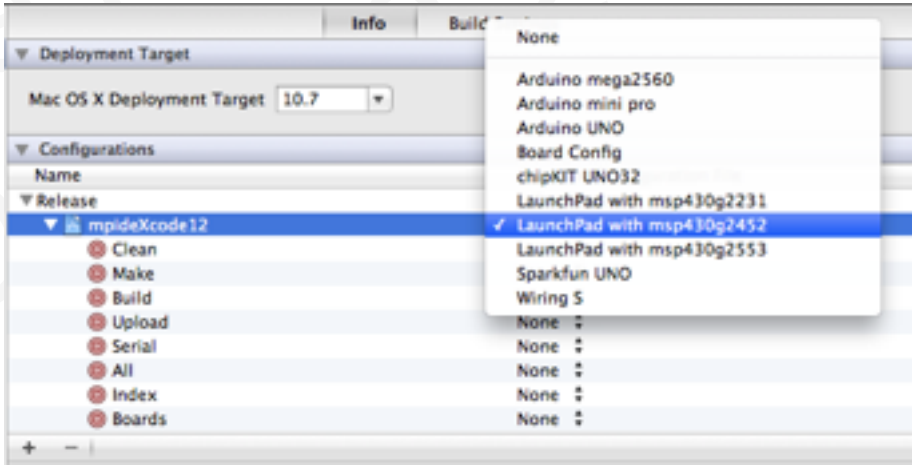


A drop-down list shows the boards available.



Just select one.

If your board isn't listed, you can create a configuration file.
Please refer to §6 Add a file.



The sketch contains conditional `#include` for the supported IDEs.

They are based on the micro-controller reference or on the IDE version. For more information, please refer to Manage Code for Multiple Platforms.

The version of Arduino, either 0023 or 1.0, is detected automatically and the corresponding library `Arduino.h` or `WProgram.h`. selected accordingly.

Those `#include` statements are included on a code snippet for easy use.

Please refer to Insert Code Snippet with #include Statements.

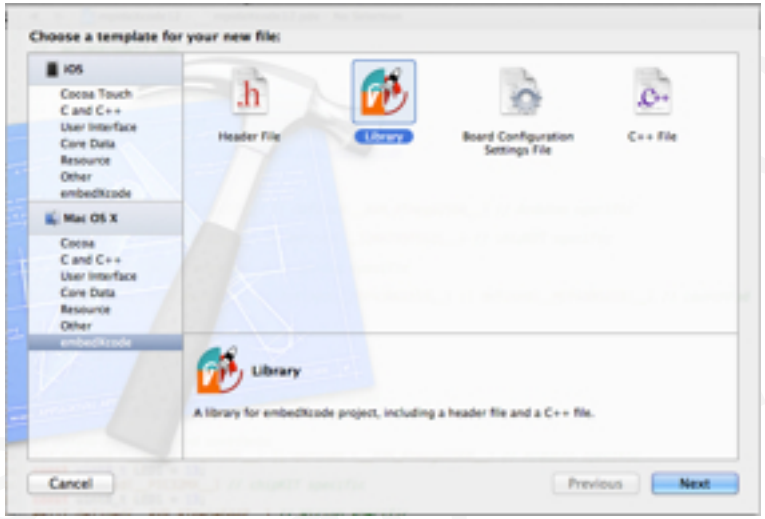
```
// Core library – MCU-based
#if defined (__AVR_ATmega328P__) || defined
(__AVR_ATmega2560__) // Arduino specific
  #if defined(ARDUINO) && (ARDUINO >= 100)
  #include "arduino.h" // – for Arduino 1.0
  #else
  #include "WProgram.h" // – for Arduino 23
  #endif
#elif defined(__32MX320F128H__) ||
defined(__32MX795F512L__) // chipKIT specific
#include "WProgram.h"
#elif defined(__AVR_ATmega644P__) // Wiring specific
#include "Wiring.h"
#elif defined(__MSP430G2452__) ||
defined(__MSP430G2553__) || defined(__MSP430G2231__) //
LaunchPad specific
#include "Energia.h"
#elif defined(MCU_STM32F103RB) ||
defined(MCU_STM32F103ZE) || defined(MCU_STM32F103CB) ||
defined(MCU_STM32F103RE) // Maple specific
#include "WProgram.h"
#endif
```

When a board is selected, the configuration file defines parameters for code-sense.

4.2. Add a File

Call the menu `File > New > New File...` or press ⌘N

Select embedXcode and then `Header File`, `C++ file`, `Library` or `Board Configuration Settings File`.



`Library` creates a header file and a C++ code file with the `#include` statement ready!

```
#include "LocalLibrary.h"
```

`Board Configuration Settings File` allows to define the settings for a new board.

Specify:

- `BOARD_TAG` is the unique identifier of the board, found in the `Boards.txt` file.
- `BOARD_PORT` defines the USB port to be used. This parameter is optional.
- `GCC_PREPROCESSOR_DEFINITIONS` is the name of the micro-controller of the board, found in the `Boards.txt` file.
- `HEADER_SEARCH_PATHS` needs to be updated with the reference of the IDE, `ARDUINO_APP` for Arduino, `MPIDE_APP` for Mpide, `WIRING_APP` for Wiring, or `ENERGIA_APP` for Energia.

The last two parameters improves the selection of the headers for code-sense.

4.3. Insert #include Statements From Code Snippet

A code snippet includes all the `#include` statements for selecting the core libraries.

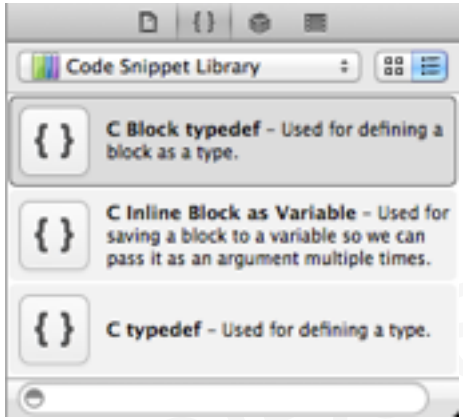
There are two versions: one MCU-based and another IDE-based.

For more information, please refer to Manage Code for Multiple Platforms.

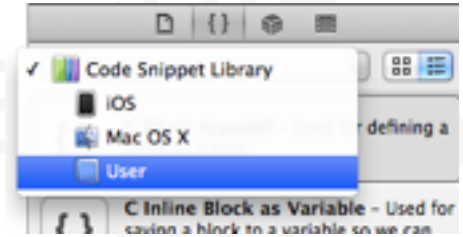
```
// Core library - MCU-based
#if defined (__AVR_ATmega328P__) || defined
(__AVR_ATmega2560__) // Arduino specific
  #if defined(ARDUINO) && (ARDUINO >= 100)
  #include "arduino.h" // - for Arduino 1.0
  #else
  #include "WProgram.h" // - for Arduino 23
  #endif
#elif defined(__32MX320F128H__) ||
defined(__32MX795F512L__) // chipKIT specific
#include "WProgram.h"
#elif defined(__AVR_ATmega644P__) // Wiring specific
#include "Wiring.h"
#elif defined(__MSP430G2452__) || defined(__MSP430G2553__)
|| defined(__MSP430G2231__) // LaunchPad specific
#include "Energia.h"
#elif defined(MCU_STM32F103RB) || defined(MCU_STM32F103ZE)
|| defined(MCU_STM32F103CB) || defined(MCU_STM32F103RE) //
Maple specific
#include "WProgram.h"
#endif
```

To display the code snippets, click on the right button of the `View` selector.

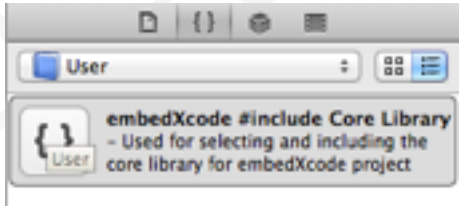
The library of code snippets is at the bottom of the right-most pane.

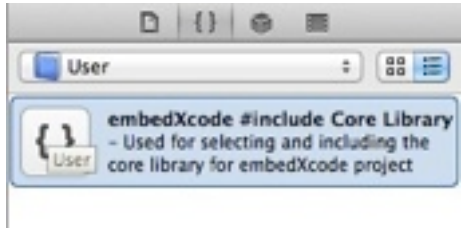


Select `User` on the drop-down list.



Select the `embedXcode #include Core Library` snippet.

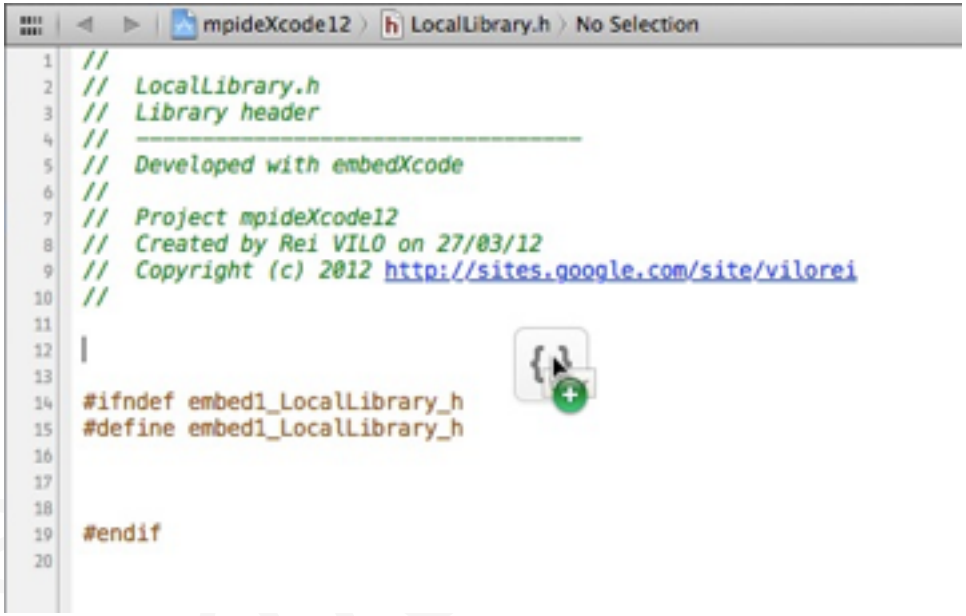




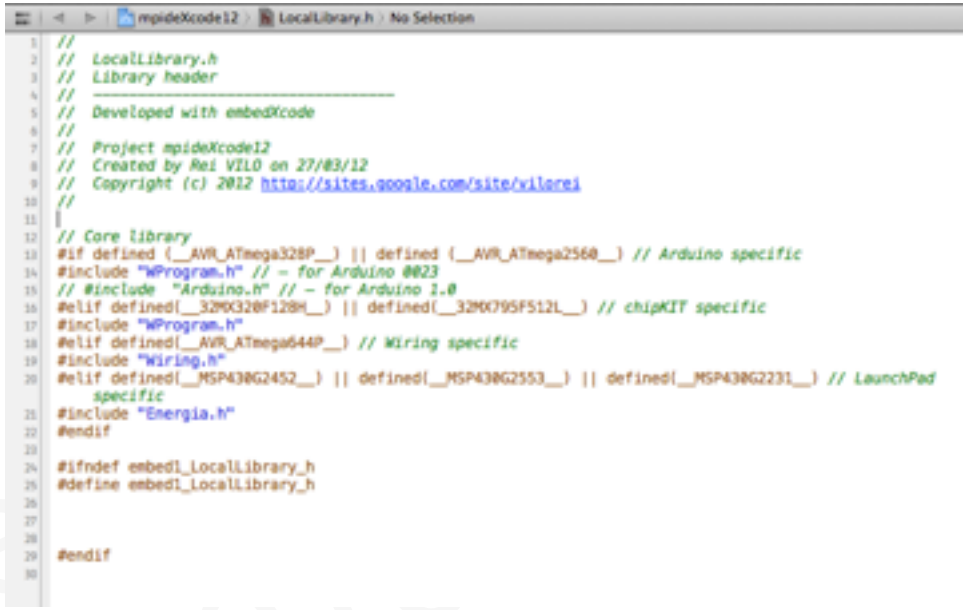
Click and drop to the destination.

The pointer changes for .

The cursor appears on the code.



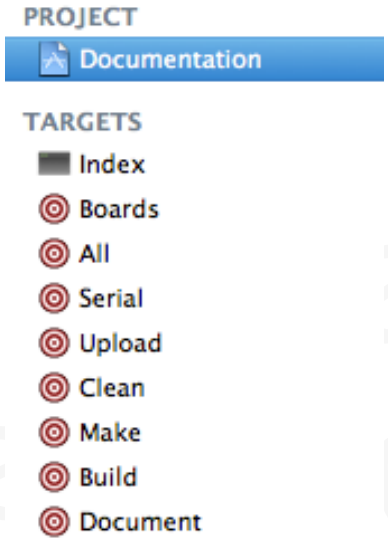
The code is inserted.



4.4. Select a target

Nine targets are offered:

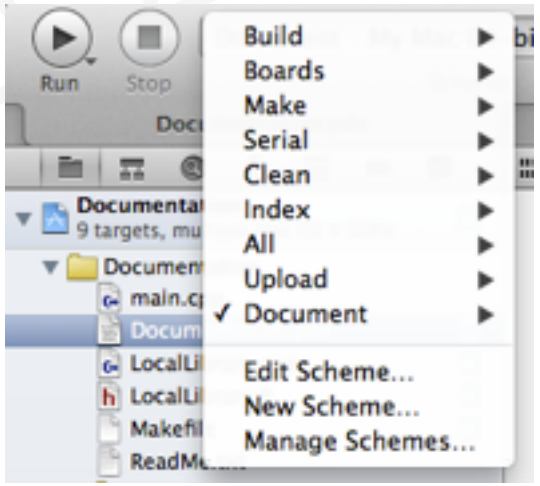
- `Clean` cleans the files from a previous compilation.
- `Make` compiles only the files changed since last compilation and links them.
- `Build` compiles all the files, changed and unchanged, and links them.
- `Upload` uploads the resulting HEX or BIN executable file to the board.
- `Serial` open the serial console in a Terminal window.
- `All` cleans the files from a previous compilation, compiles and links, uploads and open a serial window in Terminal.
- `Boards` lists all the boards with their tags and their names.
- `Document` builds the documentation.
- `Index` is a proxy target solely used for code-sense.



Just select the target you want from the drop-down list.

As a matter of fact, I mainly use `Build` and `All` for developing the code, and `Document` to write the documentation.

Other targets may be useful for specific needs.



4.5. Manage Code for Multiple Platforms

Managing code for multiples platforms is a real issue, and needs to take into account two dimensions:

- the **boards**, as Arduino Uno or Wiring S,
- and the **frameworks**, some of them with incompatible releases, as Arduino 0023, Arduino 1.0 or Wiring.

This can be done in two ways, MCU-based or IDE-based. Both are valid from an embedXcode point of view.

The first approach is **MCU-based** and relies on the micro-controller type.

This approach is compatible with the respective IDEs, as no new environment variable is created or required.

In the Arduino case, two frameworks exist so the IDE variable is required for disambiguation.

```
// Core library – MCU-based
#if defined (__AVR_ATmega328P__) || defined
(__AVR_ATmega2560__) // Arduino specific
  #if defined(ARDUINO) && (ARDUINO >= 100)
  #include "arduino.h" // – for Arduino 1.0
  #else
  #include "WProgram.h" // – for Arduino 23
#endif
#elif defined(__32MX320F128H__) ||
defined(__32MX795F512L__) // chipKIT specific
#include "WProgram.h"
#elif defined(__AVR_ATmega644P__) // Wiring specific
#include "Wiring.h"
#elif defined(__MSP430G2452__) || defined(__MSP430G2553__)
|| defined(__MSP430G2231__) // LaunchPad specific
#include "Energia.h"
#else // error
#error Platform not defined
#endif
```

The second approach is **IDE-based**. The IDEs defines a specific environment variable combine boards type and framework version.

For example, the Arduino IDE defines `ARDUINO=101` and passes it on to the tool-chain with `-D`, as `-DARDUINO=101`.

As at today, this approach is compatible with all IDEs.

The Arduino, Wiring and Maple IDEs set one single environment variables: `ARDUINO=23`, `ARDUINO=101`, `WIRING=100` and `MAPLE_IDE`.

The remaining two IDEs, MPIDE and Energia defines two environment variables, their own on top of the default one: `MPIDE=23` and `ARDUINO=23`, `ENERGIA=6` and `ARDUINO=101`.

So embedXcode tests `ARDUINO` after the specific variables.

The second approach allows more compact and easier to read code, and doesn't require maintenance at code level when a new MCU appears.

```
// Core library - IDE-based
#if defined(WIRING) // Wiring specific
#include "Wiring.h"
#elif defined(MAPLE_IDE) // Maple specific
#include "WProgram.h"
#elif defined(MPIDE) // chipKIT specific
#include "WProgram.h"
#elif defined(ENERGIA) // LaunchPad specific
#include "Energia.h"
#elif defined(ARDUINO) && (ARDUINO >= 100) // Arduino 1.0
specific
#include "Arduino.h"
#elif defined(ARDUINO) && (ARDUINO < 100) // Arduino 23
specific
#include "WProgram.h"
#else // error
#error Platform not defined
#endif
```

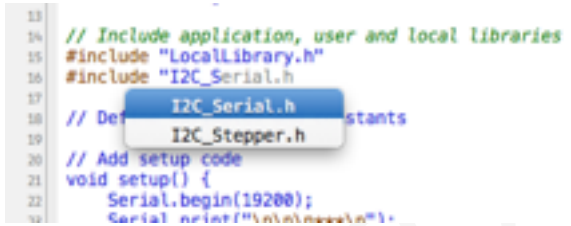
```
#if defined(WIRING) // Wiring specific
        text = in.trim();
#elif defined(ARDUINO) && (ARDUINO>=100) // for Arduino 1.0
        text.trim();
#else // all other cases
        text = text.trim();
#endif
```

4.6. Re-Index the Keywords for Code-Sense

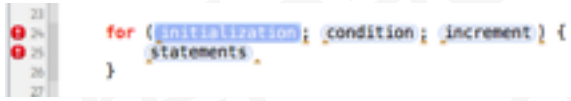
Code-sense is a major feature of Xcode.

Apart from pretty colours on the code and enhanced visibility, code-sense brings:

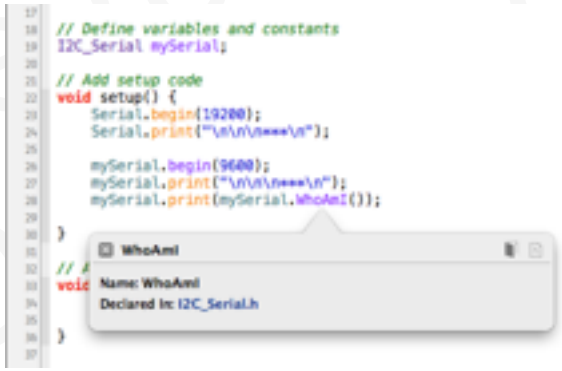
- auto-completion,



- code-snippets and check-as-you-type code monitoring,



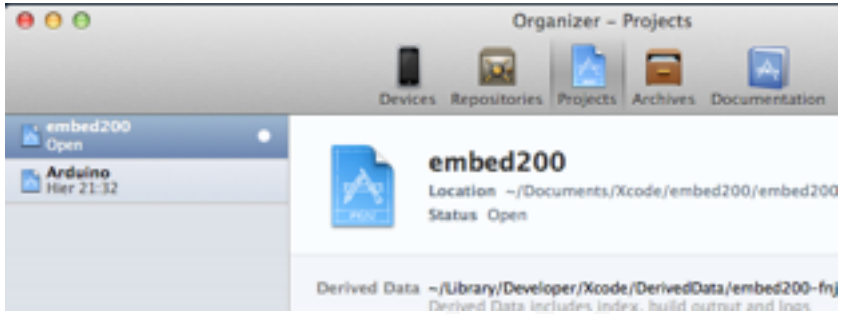
- click-to-definition



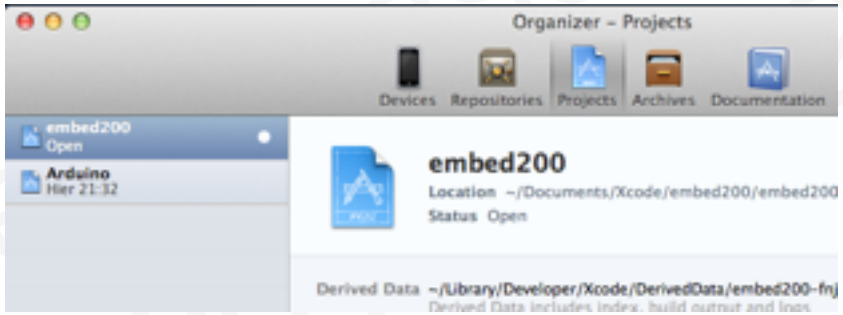
If code-sense doesn't work, we need to force a re-indexing of the key words.

To do so, first close the project.

Call the menu `Window > Organiser` and select the `Projects` pane.



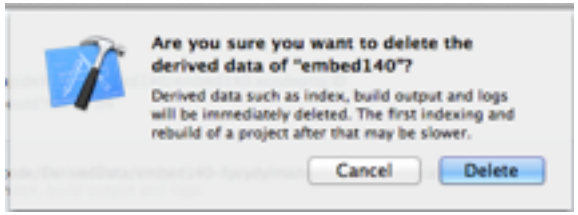
Select then the project.



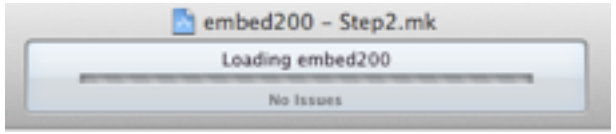
The index is saved within the `Derived Data` folder.

Click on the `Delete` button to delete them.

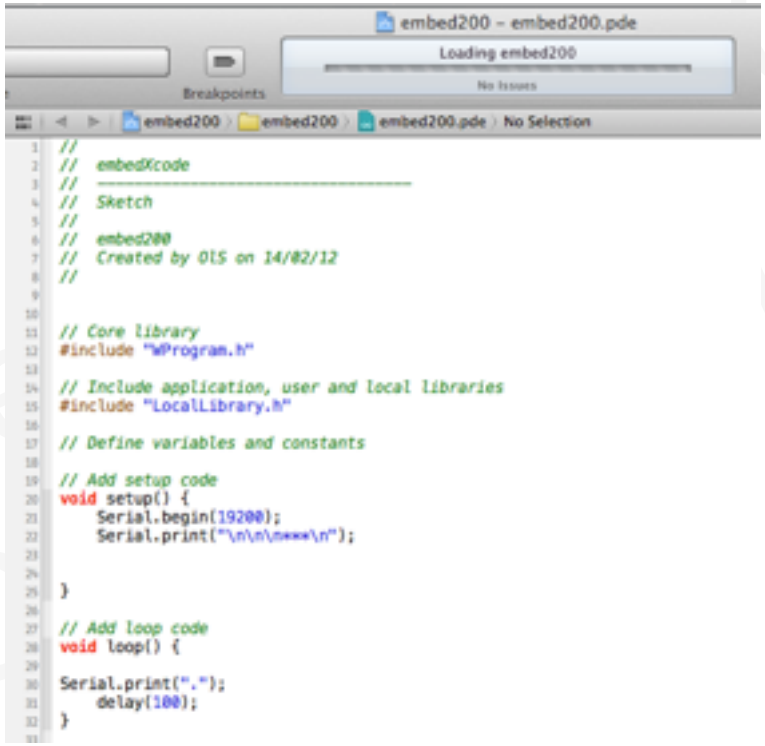
Confirm the deletion.



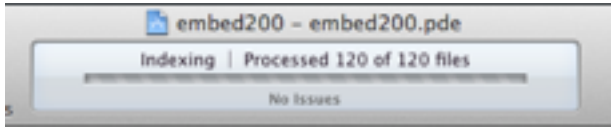
Load the project.



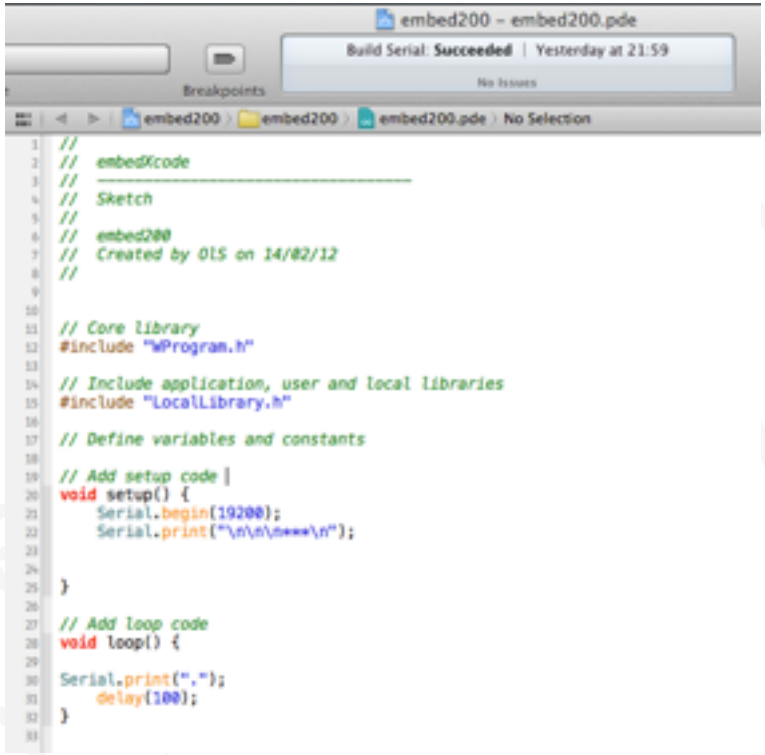
There's no code-sense yet: everything is in black-and-white, except standard C++ keywords.



The index is being built.



When the index is built, code-sense shows pretty colours.



5. Self-Document the Project

In order to obtain documentation,

- Add specific comments with defined keywords to the code,
- Build the documentation with the target **Document**,
- Use Quick Help to access the documentation.

5.1. Comment the Code

First step consists on adding specific comments with defined keywords right to the code.

Comments for self-documentation start with `///` instead of the standard `//` and include keywords with a `@` prefix.

This means that standard comments starting with the standard `//` aren't included in the documentation.

```
///
/// @file       Documentation.pde
/// @brief      Main sketch
```

Use the Doxygen Helper to speed up and ease the writing of comments for the functions.

Just select a function and press ⌘⇧**D** cmd-shift-D, the helper generates a template for the comment lines.

```
80
81  uint16_t function(uint16_t a, uint16_t b) {
82    return a + b;
83  }
84
85
```

Use the tab key to replace the light-blue fields with the comments.

In this example, the comment includes the `@brief` description of the function, list all the `@parameters` as well as the `@returned` value.

```
80
81  ///
82  /// @brief      Description
83  /// @param      a a description
84  /// @param      b b description
85  /// @return     return value description
86  ///
87  uint16_t function(uint16_t a, uint16_t b) {
88    return a + b;
89  }
```

Here's the sub-lists of the keywords I use:

- For the main page with details about the author, copyright, references, ...

Note the `@mainpage` keyword.

```
///
/// @mainpage   Documentation
/// @details    Test for doxygen integeration
/// @n
/// @n
/// @n @a       Developed with [embedXcode](http://embedXcode.weebly.com)
///
/// @author     Rei VILO
/// @author     http://embeddedcomputing.weebly.com
/// @date       09/06/12 14:02
/// @version    1
///
/// @copyright  © Rei VILO, 2012
/// @copyright  CC = BY NC SA
/// @n
///
/// @see        ReadMe.txt for references
/// @n
///
```

- For a file with details about the author, copyright, references, ...

The templates include self-documenting headers.

```
///
/// @file       filename
/// @brief      brief
/// @details    details
/// @n
/// @n
/// @n @a       Developed with [embedXcode](http://embedXcode.weebly.com)
/// @n
/// @author     author
/// @author     website
/// @date       date time
/// @version    version
/// @n
/// @copyright  © author, year
/// @copyright  CC = BY NC SA
///
/// @see        ReadMe.txt for references
/// @n
///
```

- For a function with details for parameters.

A result is documented with the keyword `@result`.

```
///
/// @brief      Blink a LED
///             LED attached to pin is light on for ms and then light off for ms
/// @param      pin pin to which the LED is attached
/// @param      times number of times
/// @param      ms duration in ms
///
void blink(uint8_t pin, uint8_t times, uint16_t ms);
```

The snippets for the documentation are under the `User` list.

- In the snippet for different details as note, warning, bug, to-do, test, ...

```
///
/// @note        note
/// @warning     warning
/// @since       since
/// @deprecated  deprecated
/// @bug         bug
/// @todo        todo
/// @test        test
///
```

- In the snippet for code

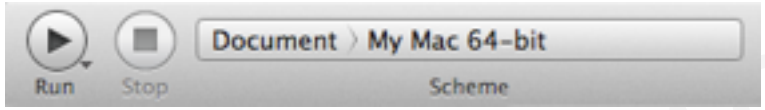
```
///
/// @code (.cpp)
///             code
/// @endcode
///
```

Doxygen includes many more options.

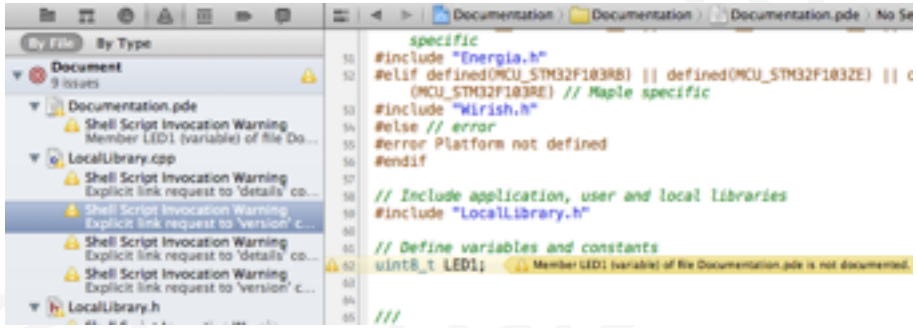
Please refer to its documentation.

5.2. *Build the Documentation*

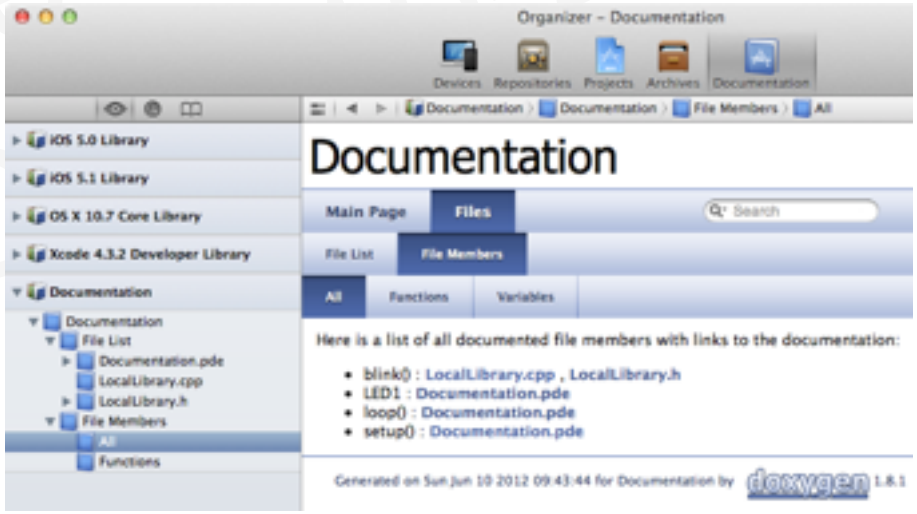
To build the documentation, just select the **Document** target and press **Run**.



Doxygen builds the documentation and issues warnings for undocumented portion of your code.



The documentation is packed in a specific file called `docset` and added to Xcode documentation.

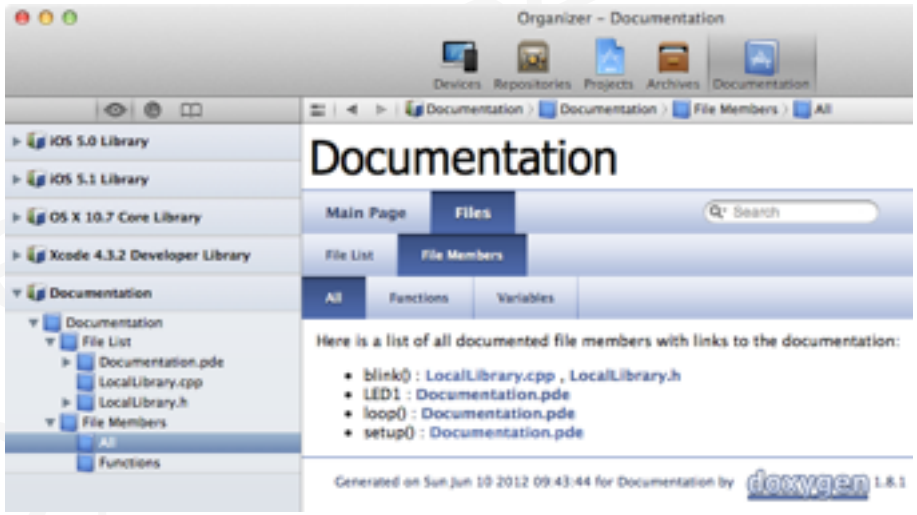


To prepare a PDF file, you need a Latex to PDF converter.
I've chosen TeXShop because it's easy to use.

Double-clicking on a `.tex` file launches TexShop and prepares the `.pdf` file.

5.3. Use the Documentation

The documentation is available on the Organiser.



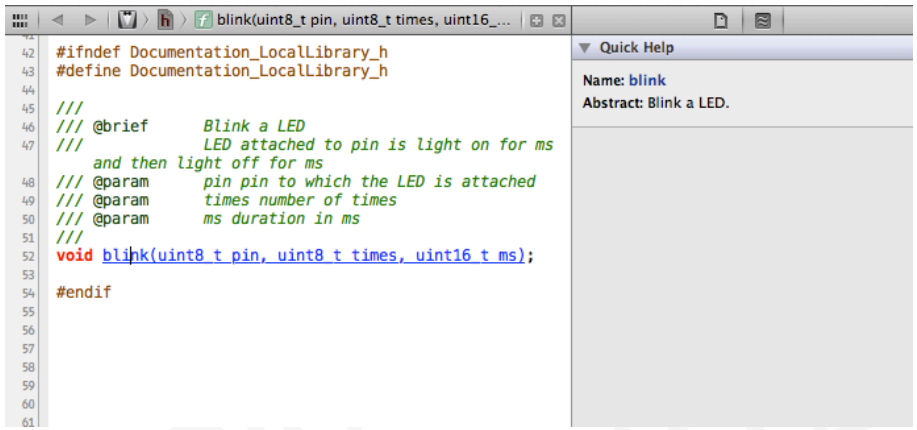
Quick help is also available.

Here, I press ⌥ alt while clicking on the name of the function `blink`: a contextual help pops-up.

Clicking on the name shown in blue **blink** in the help balloon launches the organiser and open the documentation to page related to that function!

When the cursor select a function, the Quick Help pane provides the brief description.

Clicking on the name shown in blue **blink** in the Quick Help pane launches the organiser and open the documentation to page related to that function!



The page contains all the information about the function.

6. Appendixes

6.1. HEX and BIN Files Size

Version 14 brings a significant improvement on the size of the HEX and BIN files over the previous versions.

Sizes are now close to and even better than those obtained with the corresponding IDEs.

All measures are in bytes.

Framework	Board	Maximum	IDE	Version 13	Version 14	Difference
Arduino 23	Arduino Uno	32 256	1 094	2 178	1 100	-6
	Arduino Mega 2560	258 048	1 664	4 022	1 670	-6
Arduino 1.0	Arduino Uno	32 256	1 152	3 092	1 142	10
	Arduino Mega 2560	258 048	1 696	5 458	1 718	-22
chipKIT MPIDE	chipKIT Uno32	126 976	5 928	8 892	5 928	0
LaunchPad Energia	LaunchPad with msp430g2452	8 192	630	1 098	634	-4
	LaunchPad with msp430g2553	16 384	638	1 530	642	-4
LeafLabs Maple IDE	LeafLabs Maple rev 5	108 000	12 896	13 316	12 016	880
Wiring	Wiring S	63 488	2 232	3 904	2 238	-6

6.2. What Has Been Tested

Platform	What Has Been Tested	What Has Not Been Tested	Reference Boards
Arduino	Sketch compilation and upload		Arduino Uno, Arduino mega2560 and Arduino mini pro Sparkfun Uno
chipKIT	Sketch compilation and upload		chipKIT UNO32
Wiring	Sketch compilation and upload		Wiring S
LaunchPad	Sketch compilation and upload		LaunchPad with MSP430G2542, MSP430G2331 and MSP430G2553
Maple	Sketch compilation and upload		Maple revision 5

6.3. Known Issues

Most of the issues are related to unknown keywords for the Xcode 4.3 template.

The most critical issues are:

- #2 Populated Sources List for Code-Sense
- #3 Declare PDE File as C++ Source
- #12 Define Directory for a Target

6.4. Version History

Date	mpideXcode	embedXcode	Installation Guide	Comment
Jan 22, 2012	a			Initial release
Feb 02, 2012	b			chipKIT operational
Feb 04, 2012	c			Code-sense operational
Feb 06, 2012	d			User libraries
Feb 06, 2012	e			Code checking while typing with Index as target
Feb 08, 2012	f			Code-sense, click-to-error with standard targets
Feb 14, 2012	g	1	1	Multi-application, check-as-you-type, template
Feb 18, 2012	6	2	2	Improvements
Feb 26, 2012		3		Modular makefiles
Mar 06, 2012	7	4	3	Arduino 1.0 implementation
Mar 12, 2012	8	5		Bugs fixed
Mar 15, 2012	9	6	4	All Arduino/chipKIT/Wiring/Energia and user libraries included by default
Apr 05, 2012	10	7	5	Wiring 1.0 and Energia 1.0 implementations
Apr 08, 2012	11	8	5	Bugs fixed
Apr 16, 2012	12	9	6	Code-sense reference defined by selected board
May 23, 2012	13	10	7	LeafLabs Maple implementation
Jun 03, 2012	14	11	8	MCU- or IDE-based platform identification for #include library
Jun 05, 2012	15	12	8	IDE-based identification for all platforms
Jun 14, 2012		13	9	Built-in self-documentation
Jun 25, 2012		14	10	HEX and BIN files size optimisation

6.4.1. Contributions and References

I've compiled a list of additional contributions and detailed references I've consulted to develop the emvedXcode template.

In case a contribution or reference is missing, please let me know so I could update the list.

Date	Title	Author	Links
Feb 07 2009	Bien documenter son code avec Doxygen et Xcode	mouviciel	http://mouviciel.free.fr/blog/index.php?2009/02/07/46-bien-documenter-son-code-avec-doxygen-et-xcode
Feb 28, 2009	Using Arduino in Xcode	Robert Atkins	http://robertcarlsen.net/2009/02/28/using-arduino-in-xcode-532
Mar 18, 2010	Documenting Objective-C with Doxygen Part I	Fred McCann	http://www.duckrowing.com/2010/03/18/documenting-objective-c-with-doxygen-part-i/
Mar 18, 2010	Documenting Objective-C with Doxygen Part II	Fred McCann	http://www.duckrowing.com/2010/03/18/documenting-objective-c-with-doxygen-part-ii/
Jun 04, 2010	A Makefile for Arduino Sketches	Martin Oldfield	http://bleaklow.com/2010/06/04/a_makefile_for_arduino_sketches.html
Sep 01, 2010	Using Doxygen to Create Xcode Documentation Sets	Apple	http://developer.apple.com/library/mac/#featuredarticles/DoxygenXcode/_index.html
Oct 18, 2010	How to compile AVR-Code with Mac OSX	Administrator	http://www.definefalsetrue.com/index.php/en/AVR/how-to-compile-avr-code-with-mac-osx.html
Jan 06, 2011	Thread Update: Linker Problem, Arduino Uno and Xcode	Rei Vilo	http://arduino.cc/forum/index.php/topic,49956.0.html
Mar 11, 2011	A minimal project template for Xcode 4	borealkiss	http://blog.boreal-kiss.net/2011/03/11/a-minimal-project-template-for-xcode-4/ https://github.com/borealkiss/Minimal-Template
May 14, 2011	Using the Doxygen Helper in Xcode 4	Fred McCann	http://www.duckrowing.com/2011/05/14/using-the-doxygen-helper-in-xcode-4/ http://www.duckrowing.com/wp-content/uploads/2011/05/xcode_doxygen_helper.tgz
Mar 21, 2011	Making custom templates for Xcode 4 – March 2011	Adam (red-glasses)	http://blog.red-glasses.com/index.php/tutorials/making-custom-templates-for-xcode-4-march-2011/

Date	Title	Author	Links
Apr 01, 2011	Advanced Arduino Hacking	Maik Schmidt	http://pragprog.com/magazines/2011-04/advanced-arduino-hacking https://github.com/maik/pragpub
May 30, 2011	Command-line Arduino development	Akkana Peck	http://shallowsky.com/software/arduino/arduino-cmdline.html
Jun 08, 2011	Fixing Xcode 4's Broken Code Completion	Ben Scheirman	http://benscheirman.com/2011/06/fixing-xcode-4s-broken-code-completion
Jun 23, 2011	Arduino from the command line	Martin Oldfield	http://mjo.tc/atelier/2009/02/arduino-cli.html http://mjo.tc/atelier/2009/02/acli/arduino-mk_0.6.tar.gz
Feb 11, 2011	pyserial 2.6, Python Serial Port Extension	Chris Liechti	http://pypi.python.org/pypi/pyserial
Jul 01, 2011	Arduino makefile for Xcode	Rei Vilo	http://embedXcode.weebly.com/arduino/20--arduino-makefile-for-xcode
Jul 21, 2011	MPLAB X project configurations for stk500v2 bootloader	svofski	http://www.chipkit.cc/forum/viewtopic.php?p=1285#p1285 http://pastebin.com/31XXwmUV
Jul 30, 2011	Programming Arduino with Xcode	Nick	http://makesomecode.com/2010/07/30/programming-arduino-with-xcode/
Aug 25, 2011	Minimal AVR project template for Xcode	Jens Willy Johannsen	http://stackoverflow.com/questions/6976500/avr-for-xcode-4
Oct 04, 2011	Visual Micro, Free Arduino Programming IDE for Microsoft Visual Studio	Visual Micro	http://www.visualmicro.com/
Oct 11, 2011	Arduino makefile	Álvaro Justen (Turicas)	https://github.com/turicas/arduinoMakefile/blob/master/resources.markdown
Nov 01, 2011	A command line toolkit for working with Arduino hardware	Амперка (amperka)	http://arduino.cc/forum/index.php/topic,77458.0.html https://github.com/amperka/ino
Nov 02, 2011	pyserial 2.6: Python Serial Port Extension	Chris Liechti	http://pypi.python.org/pypi/pyserial
Nov 04, 2011	Query on -ffunction-section and -fdata-sections options of gcc	fwhacking	http://stackoverflow.com/questions/4274804/query-on-ffunction-section-fdata-sections-options-of-gcc/11223700
Nov 06, 2011	Arduino Eclipse Plug-In	Jantje	http://www.baeyens.it/eclipse/ https://github.com/jantje/arduino-eclipse-plugin

Date	Title	Author	Links
Nov 12, 2011	Arduino with Xcode	Tim Knapen	https://github.com/timknapen/Arduino-With-XCode
Nov 15, 2011	Make documentation	gnu.org, collective	http://www.gnu.org/software/make/manual/html_node/index.html
Nov 15, 2011	Using the GNU Compiler Collection	gnu.org, collective	http://gcc.gnu.org/onlinedocs/gcc-4.6.2/gcc/
Dec 04, 2011	Initial repository closed	Rei Vilo	http://github.com/rei-vilo/Xcode-for-MPIDE-Arduino
Dec 04, 2011	Trunk continued by Tim Knapen	Tim Knapen	https://github.com/timknapen/Arduino-With-XCode
Dec 04, 2011	Master cloned by gnimmel	Gnimmel	https://github.com/gnimmel/Xcode-for-MPIDE-Arduino
Dec 09, 2011	chipKIT Compatible Arduino-based Makefile	Christopher Peplin	http://christopherpeplin.com/2011/12/chipkit-arduino-makefile https://github.com/peplin/arduino.mk
Dec 21, 2011	MPIDE 0023 mpide-0023-macosx-20111221	Ricklon	https://github.com/chipKIT32/chipKIT32-MAX/downloads
Jan 11, 2012	Xcode 4 external build system code completion	Mattias Wadman	http://stackoverflow.com/questions/8726869/xcode-4-external-build-system-code-completion
Jan 12, 2012	New IDE for Mac OS X written in Cocoa	fabiankr	http://arduino.cc/forum/index.php/topic,86028.0.html https://github.com/fabiankr/Cocoduino
Jan 16, 2012	Arduino with Xcode	Rei Vilo	https://github.com/rei-vilo/Arduino-With-XCode
Jan 22, 2012	mpideXcode — release a : initial release	Rei Vilo	https://github.com/rei-vilo/mpideXcode
Mar 18, 2012	Energia = Arduino IDE ported to LaunchPad MSP430	Robert Wessel	https://github.com/energia/Energia
Apr 25, 2012	Graphviz 2.28, Graph Visualization Software	ATT	http://www.graphviz.org/
May 19, 2012	Doxygen 1.8.1	Dimitri van Heesch	http://doxygen.org/
Jun 06, 2012	TeXShop 3.11	Richard Koch	http://www.texshop.org

6.5. Referenced Boards

Not all boards have been tested. Please refer to What Has Been Tested for more details.

According to the `Boards.txt` files:

Arduino	chipKIT	Wiring	LaunchPad	Maple
Arduino Uno	chipKIT UNO32	Wiring S @ 16 MHz	LaunchPad with mps430g2452	LeafLabs Maple Rev 3+ to Flash
Arduino Duemilanove or Nano w/ ATmega328	chipKIT MAX32	Wiring S with Play Shield @ 16 MHz	LaunchPad with mps430g2231	LeafLabs Maple Rev 3+ to RAM
Arduino Diecimila, Duemilanove, or Nano w/ ATmega168	chipKIT MAX32-USB for Serial	Wiring V1.0/Wiring Mini V1.0 @ 16 MHz	LaunchPad with mps430g2553	LeafLabs Maple Mini Rev 2 to Flash
Arduino Mega 2560	Cerebot MX3cK	Wiring V1.1 ATmega1281 @ 16 MHz		LeafLabs Maple Mini Rev 2 to RAM
Arduino Mega (ATmega1280)	Cerebot MX4cK	Wiring V1.1 ATmega2561 @ 16 MHz		LeafLabs Maple RET6 Edition to Flash
Arduino Mini	Cerebot MX7cK			LeafLabs Maple RET6 Edition to RAM
Arduino Fio	Cerebot 32MX4			LeafLabs Maple Native (Beta) to Flash
Arduino BT w/ ATmega328	Cerebot 32MX7			LeafLabs Maple Native (Beta) to RAM
Arduino BT w/ ATmega168	Microchip PIC32 Starter kit			
LilyPad Arduino w/ ATmega328	Microchip PIC32 Ethernet Starter kit			
LilyPad Arduino w/ ATmega168	Microchip PIC32 USB Starter kit II			

Arduino	chipKIT	Wiring	LaunchPad	Maple
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328	Microchip PIC32 Explorer 16			
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168	MirkoElektronika PIC32 Multimedia Board			
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328	MirkoElektronika PIC32 mikroMedia Board			
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168	Pic32 UBW32-MX460			
Arduino NG or older w/ ATmega168	Pic32 UBW32-MX795			
Arduino NG or older w/ ATmega8	Pic32 CUI32-Development Stick			

7. Copyright and Licence

Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0). This is a human-readable summary of the Legal Code.

7.1. Summary

You are free:

- **to Share** — to copy, distribute and transmit the work
- **to Remix** — to adapt the work

Under the following conditions:

- **Attribution** — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
- **Noncommercial** — You may not use this work for commercial purposes.
- **Share Alike** — If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.

With the understanding that:

- **Waiver** — Any of the above conditions can be **waived** if you get permission from the copyright holder.
- **Public Domain** — Where the work or any of its elements is in the **public domain** under applicable law, that status is in no way affected by the license.
- **Other Rights** — In no way are any of the following rights affected by the license:
  - Your fair dealing or **fair use** rights, or other applicable copyright exceptions and limitations;
  - The author's **moral** rights;
  - Rights other persons may have either in the work itself or in how the work is used, such as **publicity** or privacy rights.

Notice — For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.

Disclaimer — The Commons Deed is not a license. It is simply a handy reference for understanding the Legal Code (the full license) — it is a human-readable expression of some of its key terms. Think of it as the user-friendly interface to the Legal Code beneath. This Deed itself has no legal value, and its contents do not appear in the actual license.

Creative Commons is not a law firm and does not provide legal services. Distributing of, displaying of, or linking to this Commons Deed does not create an attorney-client relationship.

7.2. Legal Code

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. **"Adaptation"** means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. **"Collection"** means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(g) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.

c. **"Distribute"** means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

d. **"License Elements"** means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, Noncommercial, ShareAlike.

e. **"Licensor"** means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

f. **"Original Author"** means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

g. **"Work"** means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works

expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

h. **"You"** means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

i. **"Publicly Perform"** means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

j. **"Reproduce"** means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved, including but not limited to the rights described in Section 4(e).

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This

Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(d), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(d), as requested.

b. You may Distribute or Publicly Perform an Adaptation only under: (i) the terms of this License; (ii) a later version of this License with the same License Elements as this License; (iii) a Creative Commons jurisdiction license (either this or a later license version) that contains the same License Elements as this License (e.g., Attribution-NonCommercial-ShareAlike 3.0 US) ("Applicable License"). You must include a copy of, or the URI, for Applicable License with every copy of each Adaptation You Distribute or Publicly Perform. You may not offer or impose any terms on the Adaptation that restrict the terms of the Applicable License or the ability of the recipient of the Adaptation to exercise the rights granted to that recipient under the terms of the Applicable License. You must keep intact all notices that refer to the Applicable License and to the disclaimer of warranties with every copy of the Work as included in the Adaptation You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Adaptation, You may not impose any effective technological measures on the Adaptation that restrict the ability of a recipient of the Adaptation from You to exercise the rights granted to that recipient under the terms of the Applicable License. This Section 4(b) applies to the Adaptation as incorporated in a Collection, but this does not require the Collection apart from the Adaptation itself to be made subject to the terms of the Applicable License.

c. You may not exercise any of the rights granted to You in Section 3 above in any manner that is primarily intended for or directed toward commercial advantage or private monetary compensation. The exchange of the Work for other copyrighted works by means of digital file-sharing or otherwise shall not be considered to be intended for or directed toward commercial advantage or private monetary compensation, provided there is no payment of any monetary compensation in connection with the exchange of copyrighted works.

d. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and, (iv) consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(d) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

e. For the avoidance of doubt:

    i. **Non-waivable Compulsory License Schemes**. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

    ii. **Waivable Compulsory License Schemes**. In those jurisdictions in which the right to collect royalties through any statutory or

compulsory licensing scheme can be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License if Your exercise of such rights is for a purpose or use which is otherwise than noncommercial as permitted under Section 4(c) and otherwise waives the right to collect royalties through any statutory or compulsory licensing scheme; and,

iii. **Voluntary License Schemes**. The Licensor reserves the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License that is for a purpose or use which is otherwise than noncommercial as permitted under Section 4(c).

f. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING AND TO THE FULLEST EXTENT PERMITTED BY APPLICABLE LAW, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THIS EXCLUSION MAY NOT APPLY TO YOU.

6. **Limitation on Liability.**

EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at **http://creativecommons.org/**.