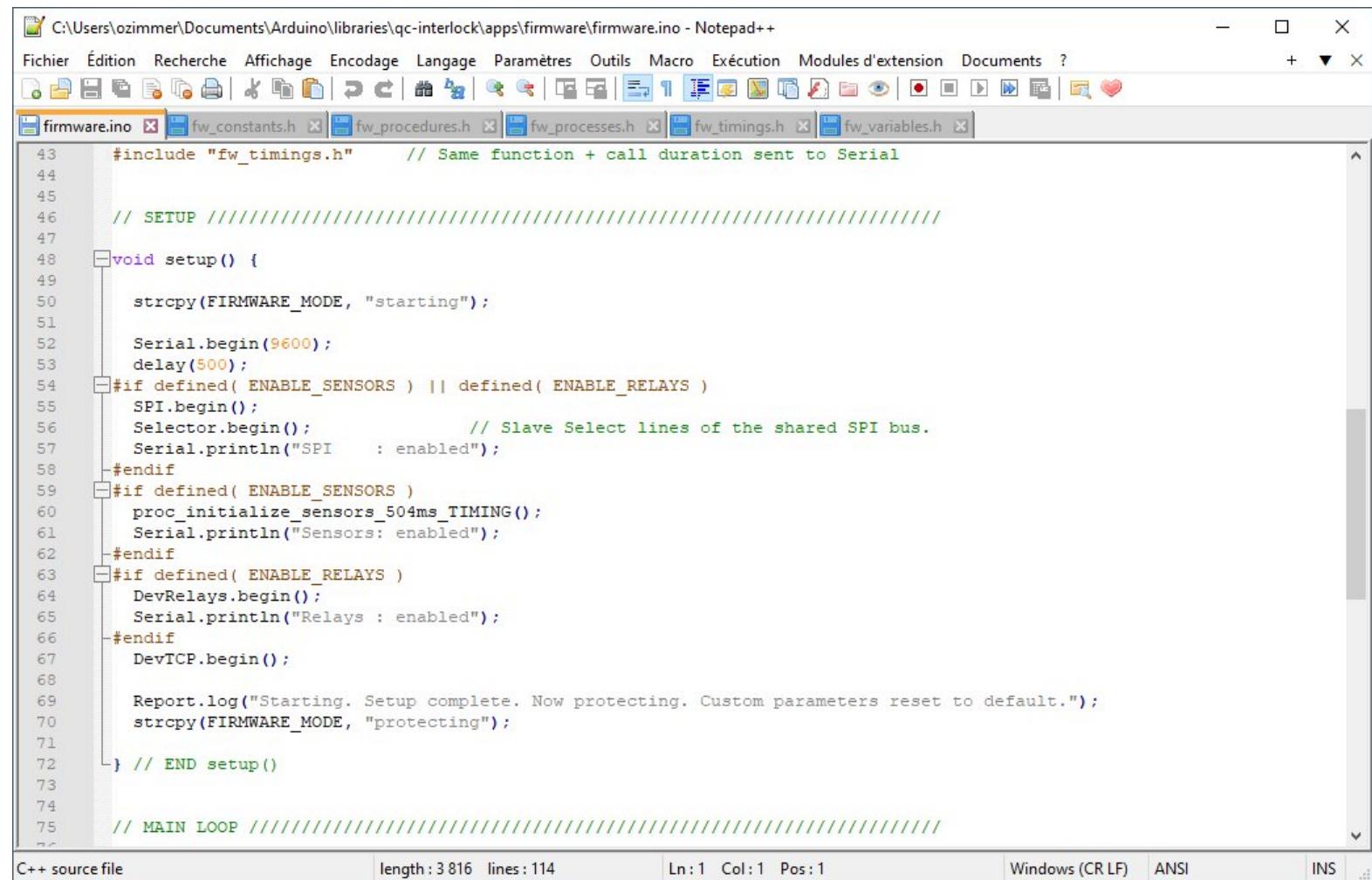


Organisation d'un grand projet logiciel sous Arduino IDE 1.8.19 (legacy)"



Utilisez Notepad++ pour créer/éditer les fichiers sources.



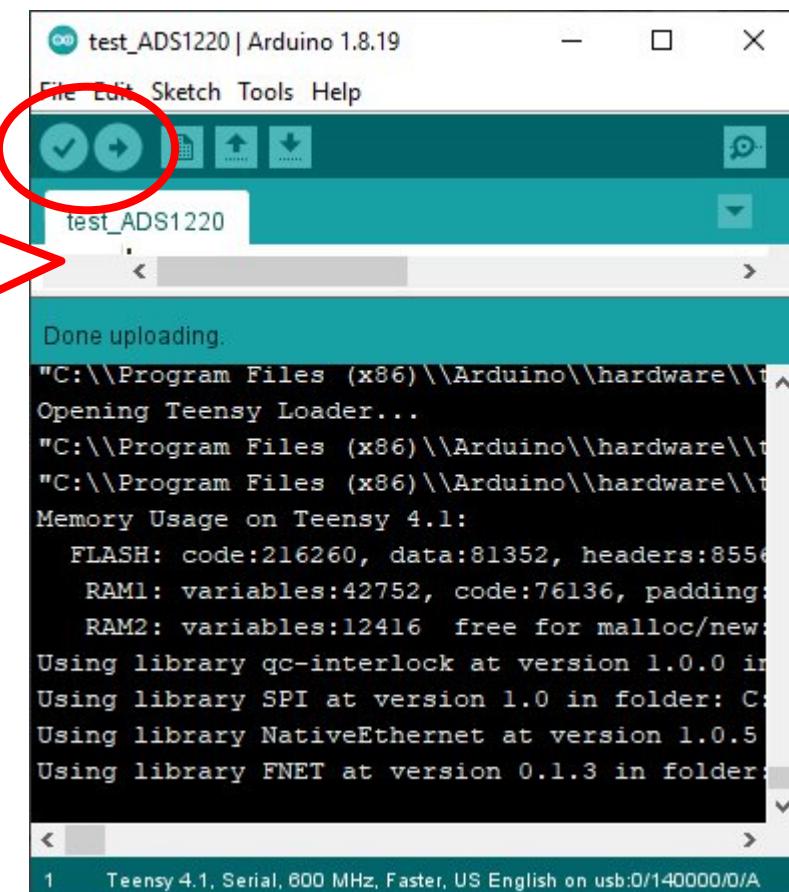
The screenshot shows the Notepad++ interface with the following details:

- Title Bar:** C:\Users\ozimmer\Documents\Arduino\libraries\qc-interlock\apps\firmware\firmware.ino - Notepad++
- Menu Bar:** Fichier, Édition, Recherche, Affichage, Encodage, Langage, Paramètres, Outils, Macro, Exécution, Modules d'extension, Documents, ?
- Toolbar:** Includes icons for file operations like Open, Save, Print, Find, Replace, and various document formats.
- Tab Bar:** firmware.ino (active), fw_constants.h, fw_procedures.h, fw_processes.h, fw_timings.h, fw_variables.h
- Code Editor:** Displays the Arduino source code for the `firmware.ino` file. The code includes setup and main loop sections with conditional compilation directives (#if, #endif) for sensors and relays. It also includes SPI.begin() and Selector.begin() calls, along with serial communication and log statements.
- Status Bar:** C++ source file, length: 3816 lines: 114, Ln:1 Col:1 Pos:1, Windows (CR LF), ANSI, INS.

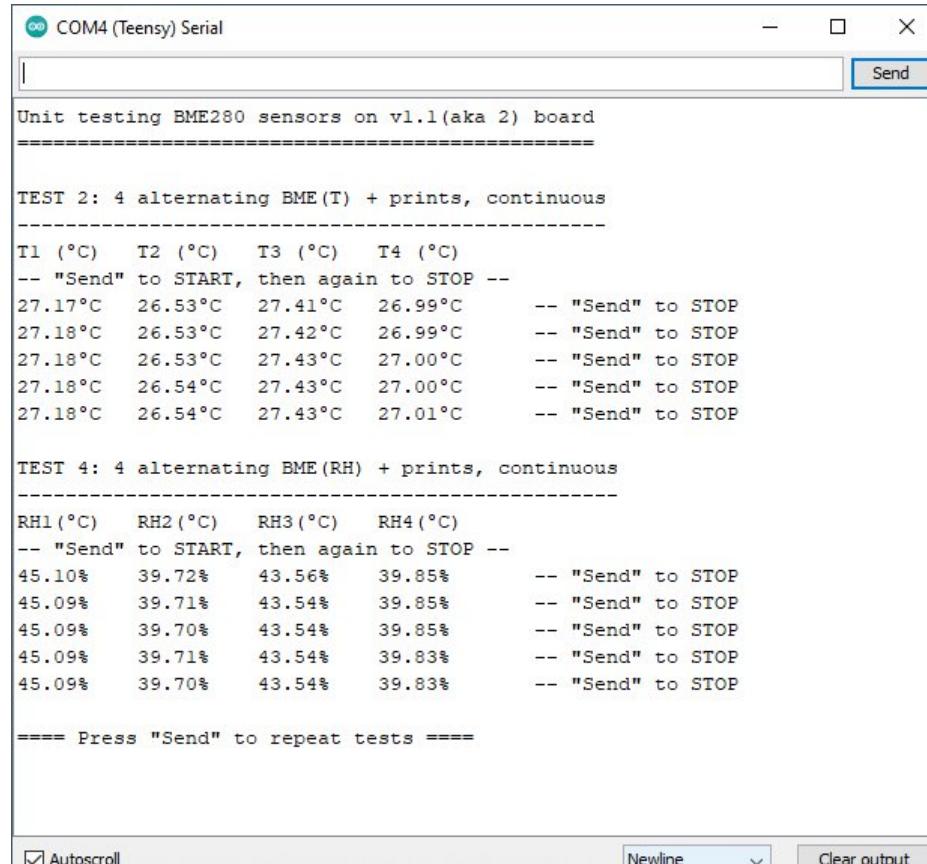
Pour compiler, identifier les erreurs et flasher le programme, utilisez Arduino IDE.

Le bouton compile automatiquement la dernière version du projet affiché *à partir du fichier sur le disque*.

(donc pas besoin de le recharger après édition dans Notepad++)



Pour les interactions utilisateur en mode développement ou les informations de débogage, utiliser la série sur USB et le moniteur Arduino IDE.



The screenshot shows the Arduino Serial Monitor window titled "COM4 (Teensy) Serial". The window displays the following text:

```
Unit testing BME280 sensors on v1.1(aka 2) board
=====
TEST 2: 4 alternating BME(T) + prints, continuous
-----
T1 (°C) T2 (°C) T3 (°C) T4 (°C)
-- "Send" to START, then again to STOP --
27.17°C 26.53°C 27.41°C 26.99°C -- "Send" to STOP
27.18°C 26.53°C 27.42°C 26.99°C -- "Send" to STOP
27.18°C 26.53°C 27.43°C 27.00°C -- "Send" to STOP
27.18°C 26.54°C 27.43°C 27.00°C -- "Send" to STOP
27.18°C 26.54°C 27.43°C 27.01°C -- "Send" to STOP

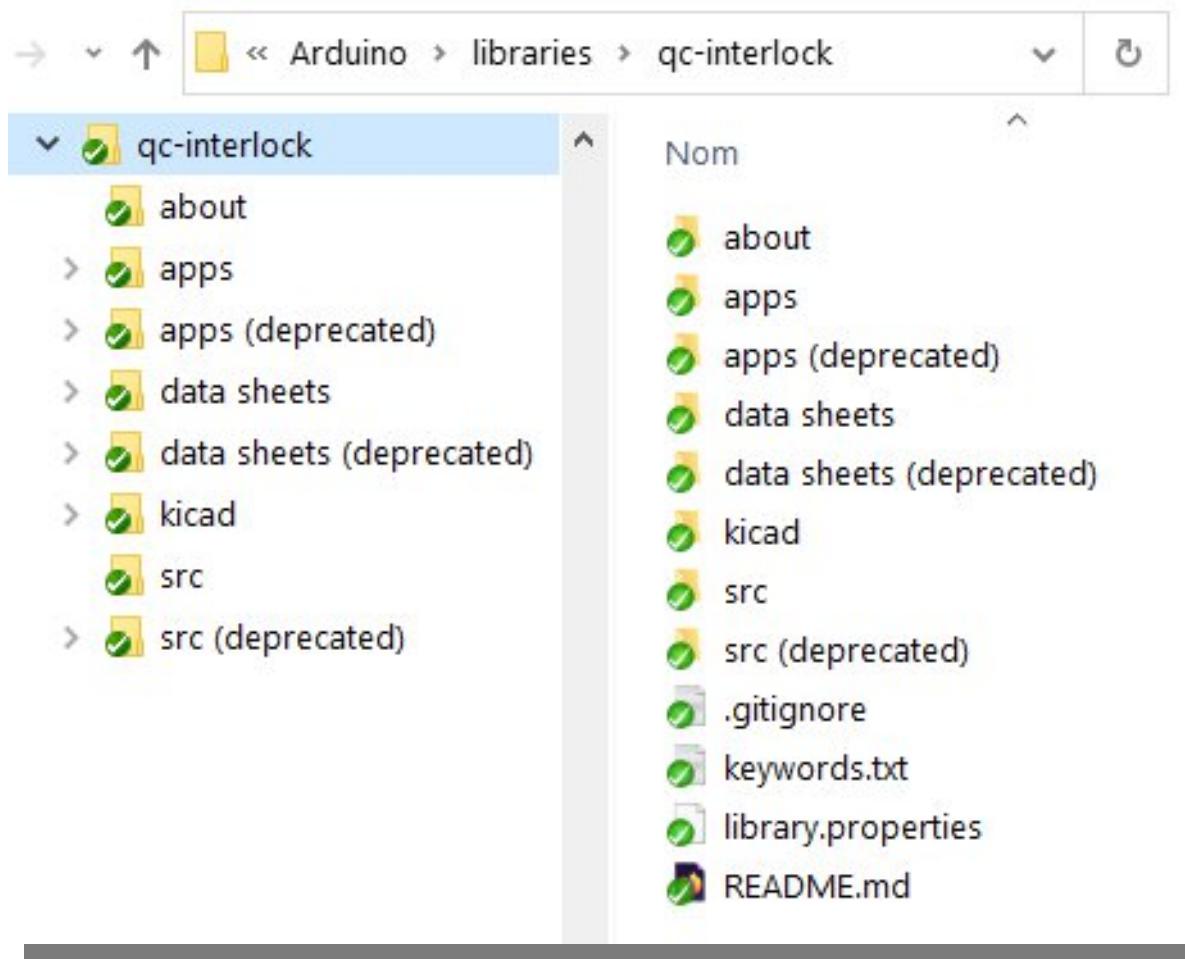
TEST 4: 4 alternating BME(RH) + prints, continuous
-----
RH1(°C) RH2(°C) RH3(°C) RH4(°C)
-- "Send" to START, then again to STOP --
45.10% 39.72% 43.56% 39.85% -- "Send" to STOP
45.09% 39.71% 43.54% 39.85% -- "Send" to STOP
45.09% 39.70% 43.54% 39.85% -- "Send" to STOP
45.09% 39.71% 43.54% 39.83% -- "Send" to STOP
45.09% 39.70% 43.54% 39.83% -- "Send" to STOP

==== Press "Send" to repeat tests ====

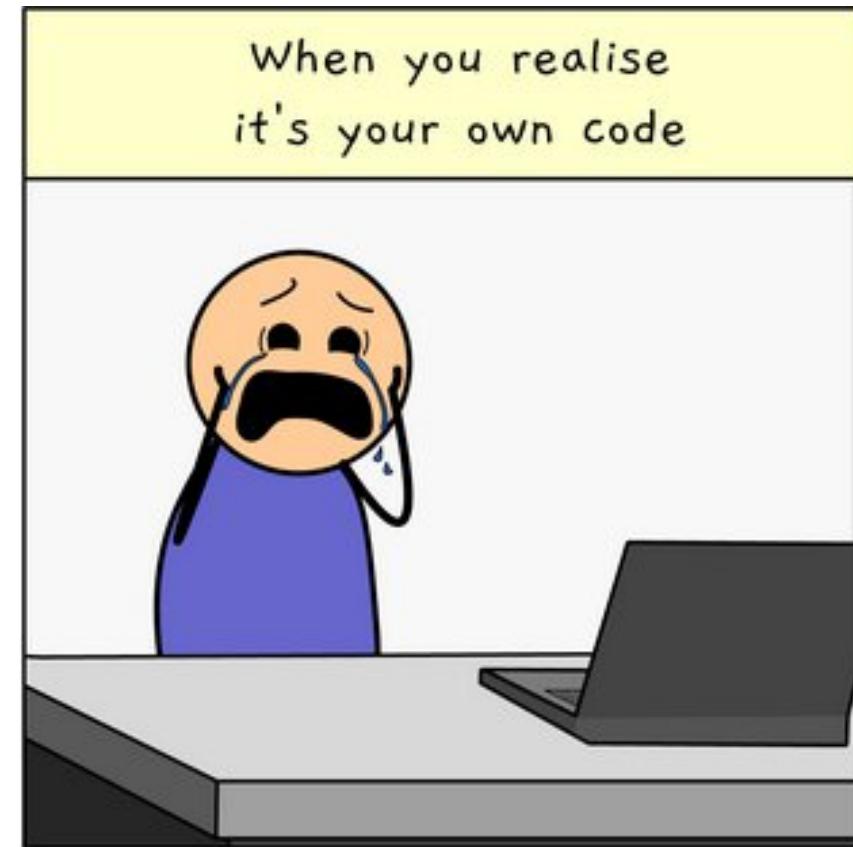
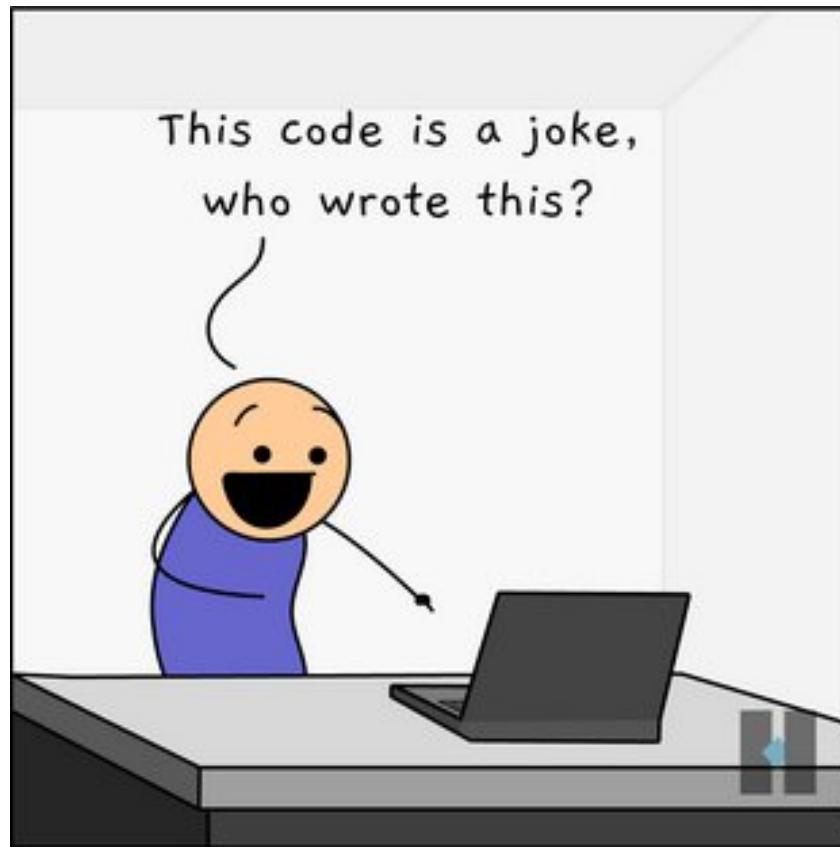
```

At the bottom of the window, there are three buttons: "Autoscroll" (checked), "Newline", and "Clear output".

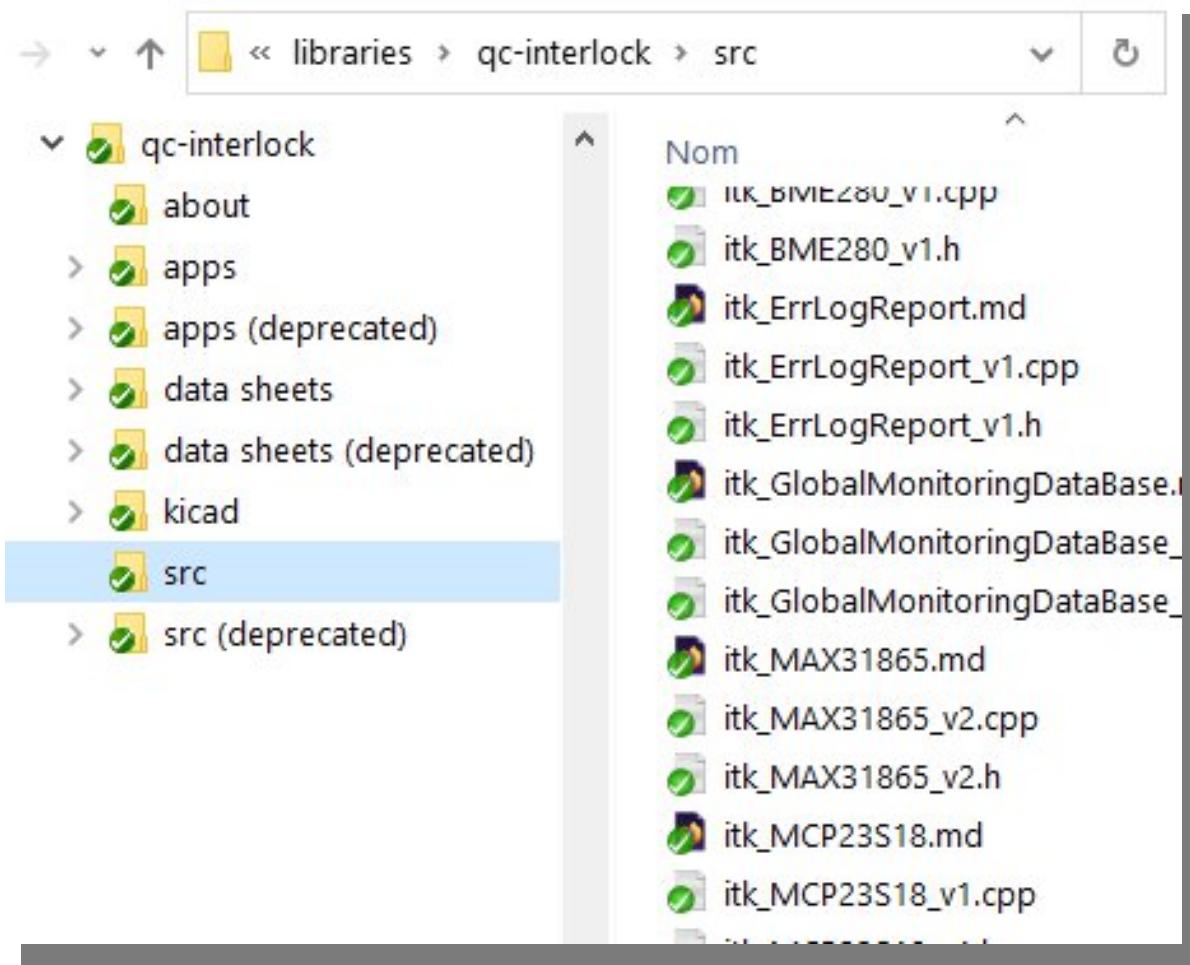
Assemblez dans une seule "bibliothèque utilisateur" Arduino l'ensemble des fichiers du projet.



- fichier de plus haut niveau (.ino) du programme principal
- fichier de plus haut niveau (.ino) des programmes de test unitaires ou intégrés
- modules utilisateur de code utilisés par les programmes
- documentation des matériels
- documentation du programme
- autres fichiers

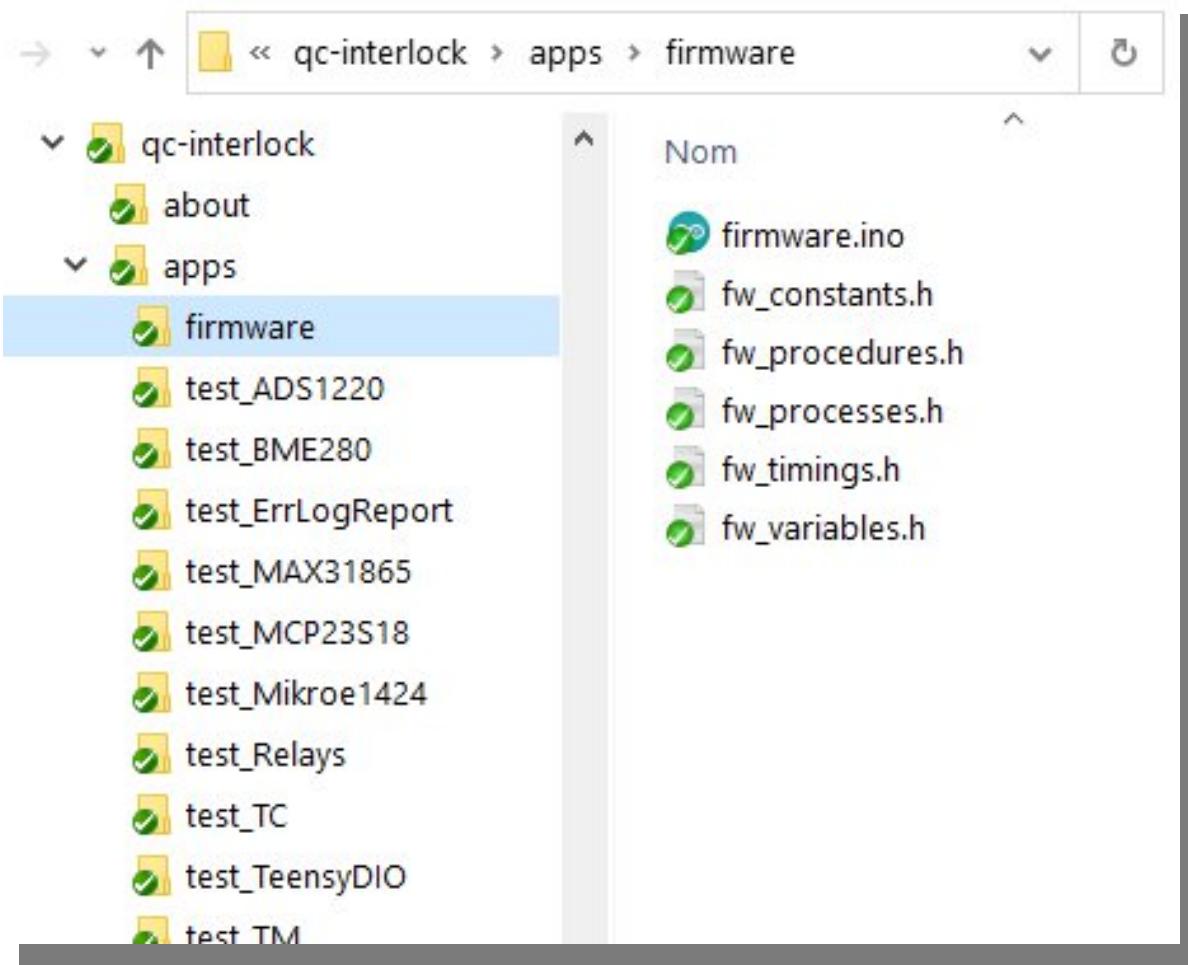


Adoptez le "nouveau format" (1.5+) de bibliothèque Arduino.



- Le dossier de la bibliothèque est de fait votre "dossier projet".
- Il doit être placé dans "%UserProfile%\Documents\Arduino\libraries"
- Modulariser votre code sous la forme de bibliothèques utilisateur.
- Réunissez ces bibliothèques dans le sous-dossier "<projet>\src".

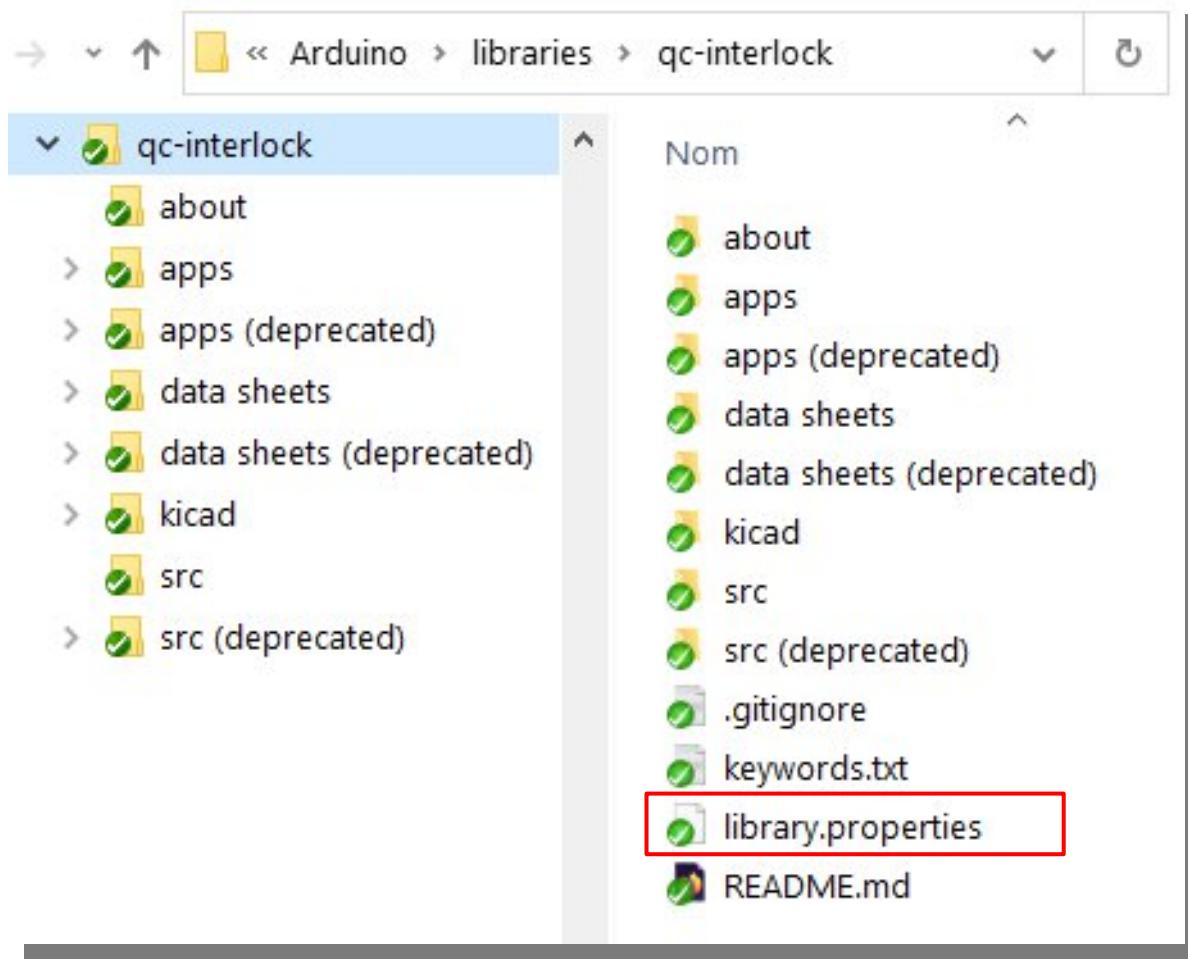
Adoptez le "nouveau format" (1.5+) de bibliothèque Arduino.



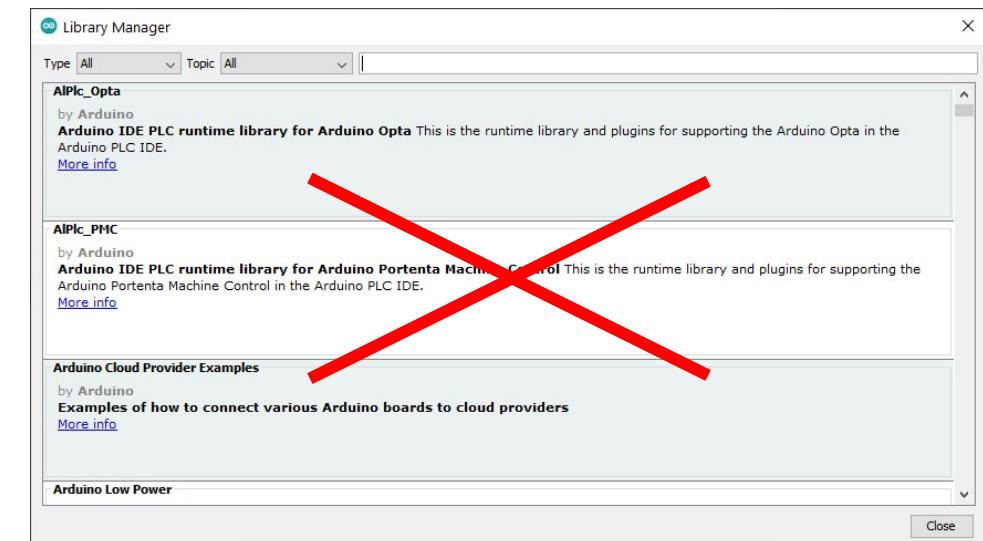
- Placez les dossiers des programmes* principal et de test dans un ou plusieurs sous-dossiers du projet.
Ils apparaîtront comme "exemples" dans Arduino IDE.
Ils ne pourront pas être modifiés "trop facilement" dans Arduino IDE qui n'autorise pas leur modification avec son éditeur.

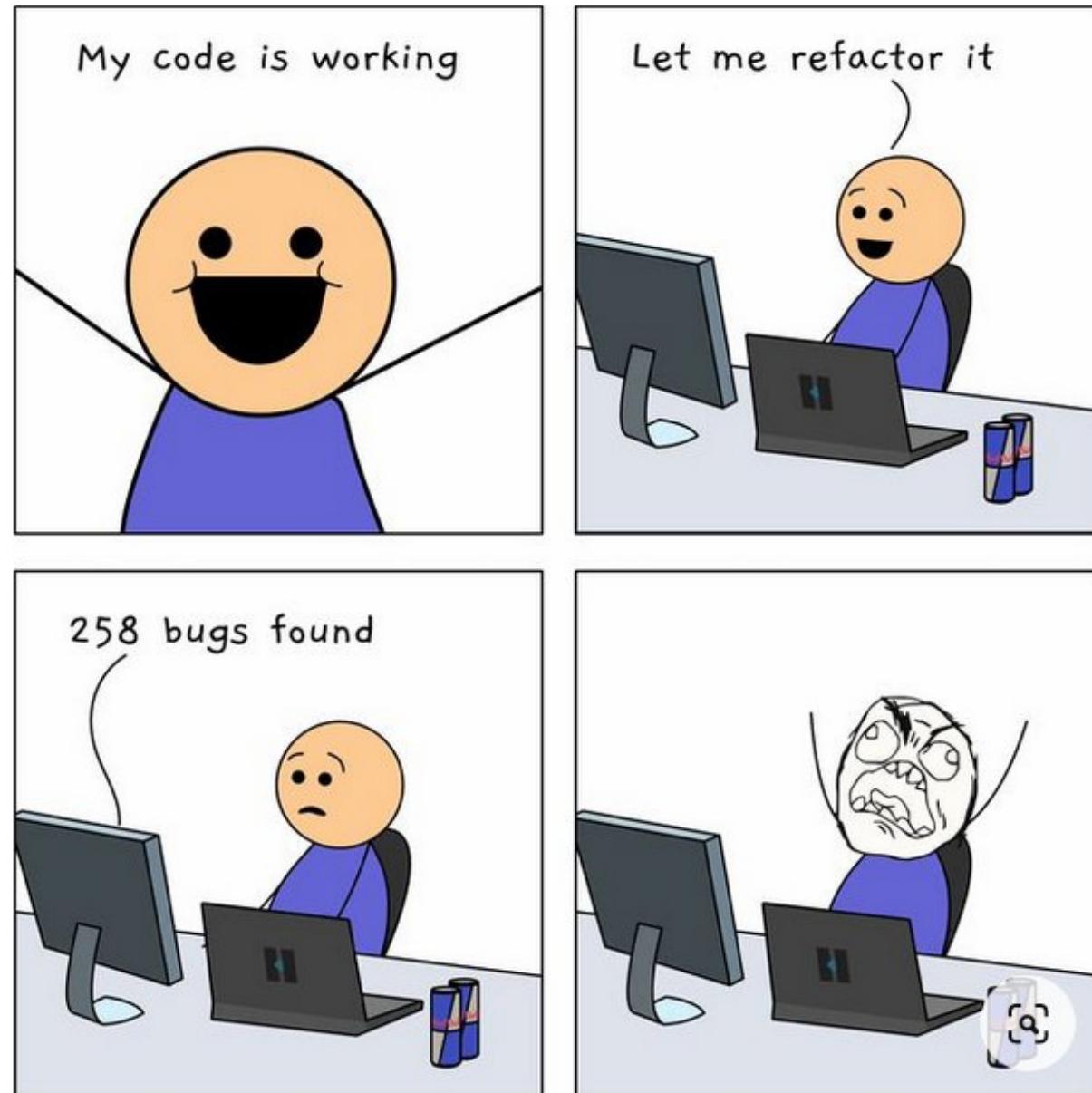
* rappel : avec Arduino IDE, 1 programme.ino = 1 dossier homonyme obligatoire.

Ne cherchez pas à intégrer le gestionnaire de bibliothèques Arduino IDE.

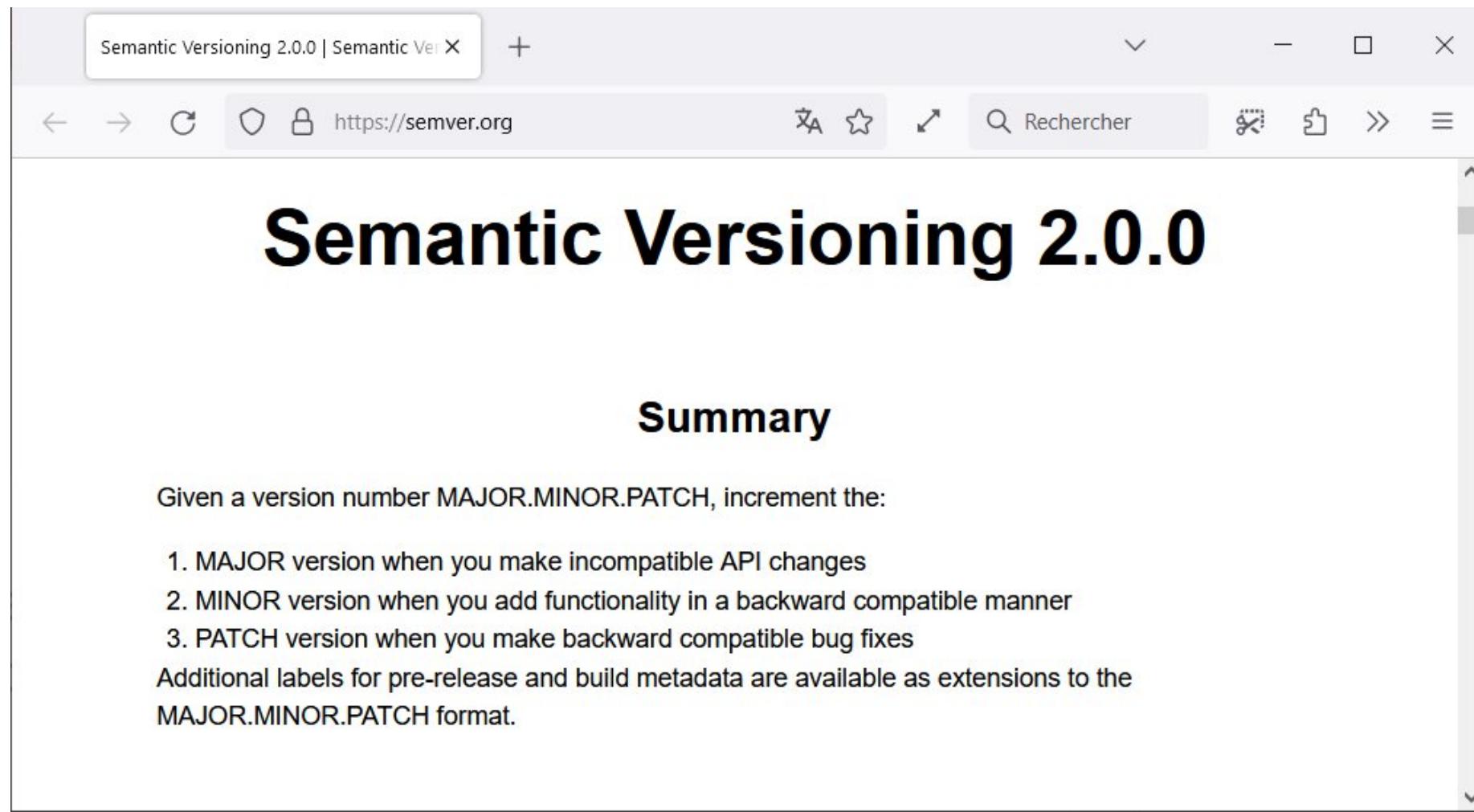


- Note : vous devez néanmoins inclure un fichier "library.properties" minimum.





Utilisez un versionnage "semver" (.org) pour vos bibliothèques* et préfixez le nom des fichiers avec la version majeure.



The screenshot shows a web browser window with the title "Semantic Versioning 2.0.0 | Semantic Ver X". The address bar displays "https://semver.org". The main content area features a large, bold heading "Semantic Versioning 2.0.0" followed by a "Summary" section. Below the summary, there is a list of rules for incrementing version numbers and a note about pre-release and build metadata.

Semantic Versioning 2.0.0

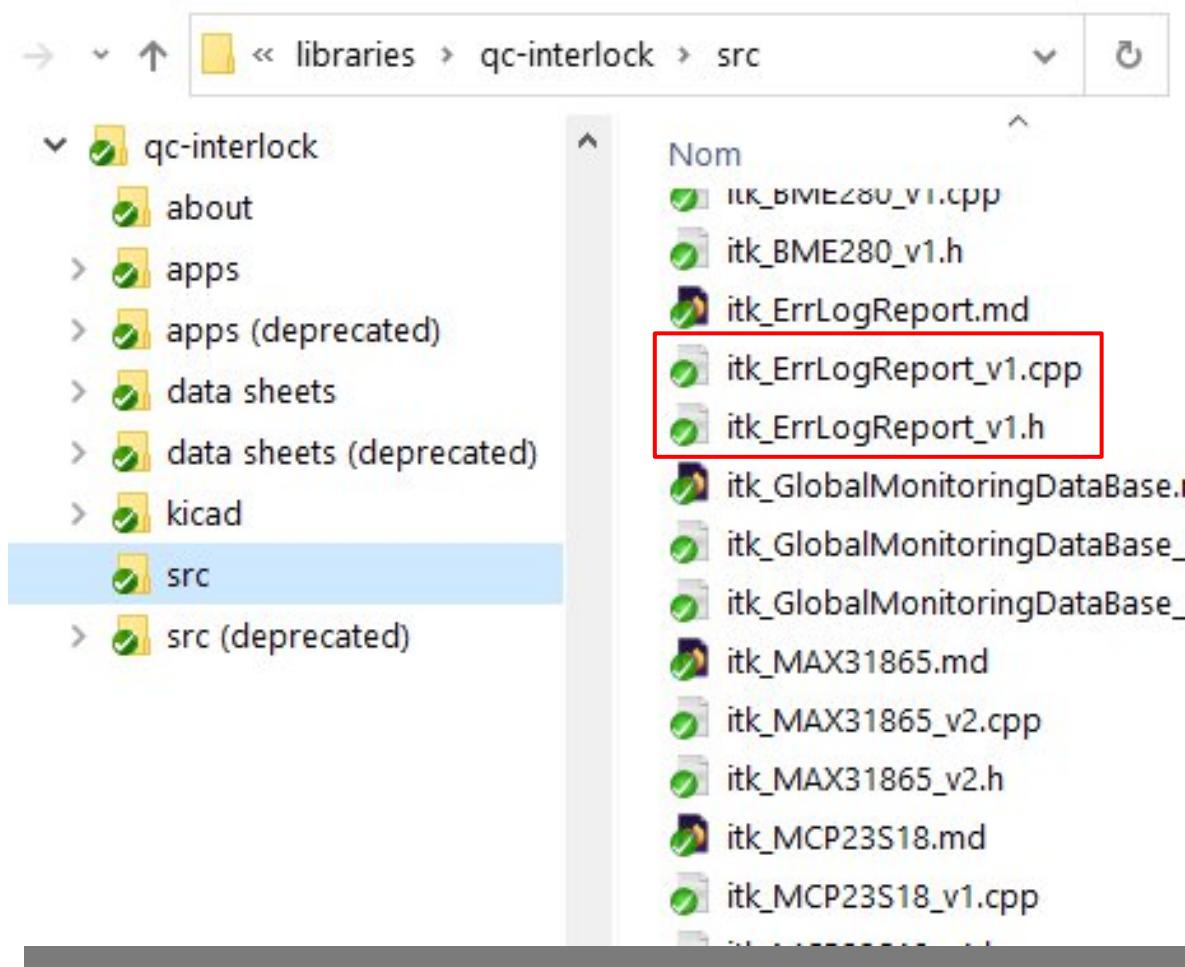
Summary

Given a version number MAJOR.MINOR.PATCH, increment the:

1. MAJOR version when you make incompatible API changes
2. MINOR version when you add functionality in a backward compatible manner
3. PATCH version when you make backward compatible bug fixes

Additional labels for pre-release and build metadata are available as extensions to the MAJOR.MINOR.PATCH format.

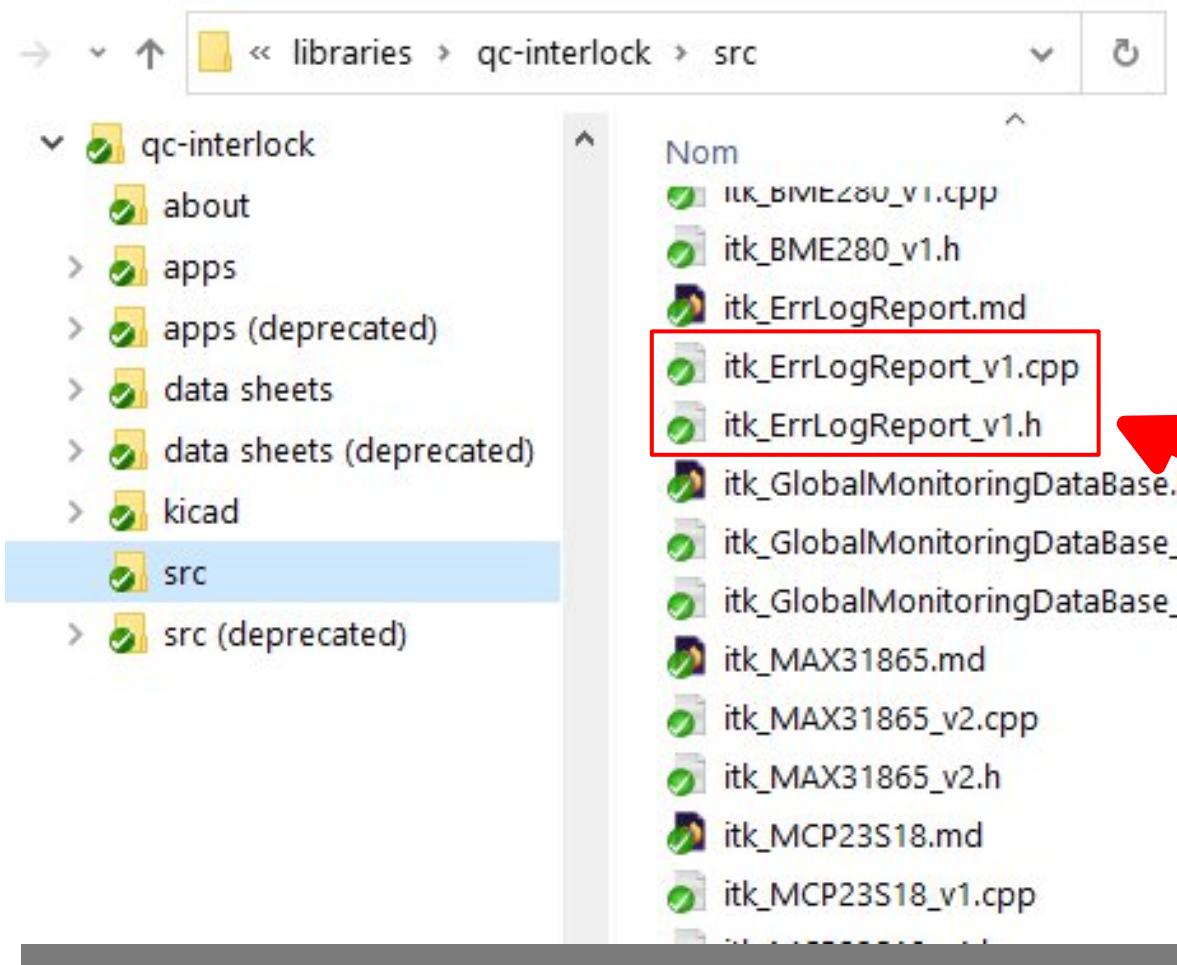
Utilisez un versionnage "semver" (.org) pour vos bibliothèques* et préfixez le nom des fichiers avec la version majeure.



- Permet la coexistence de deux versions majeures distinctes.

* une bibliothèque = 1 fichier.h et 1 fichier.cpp.

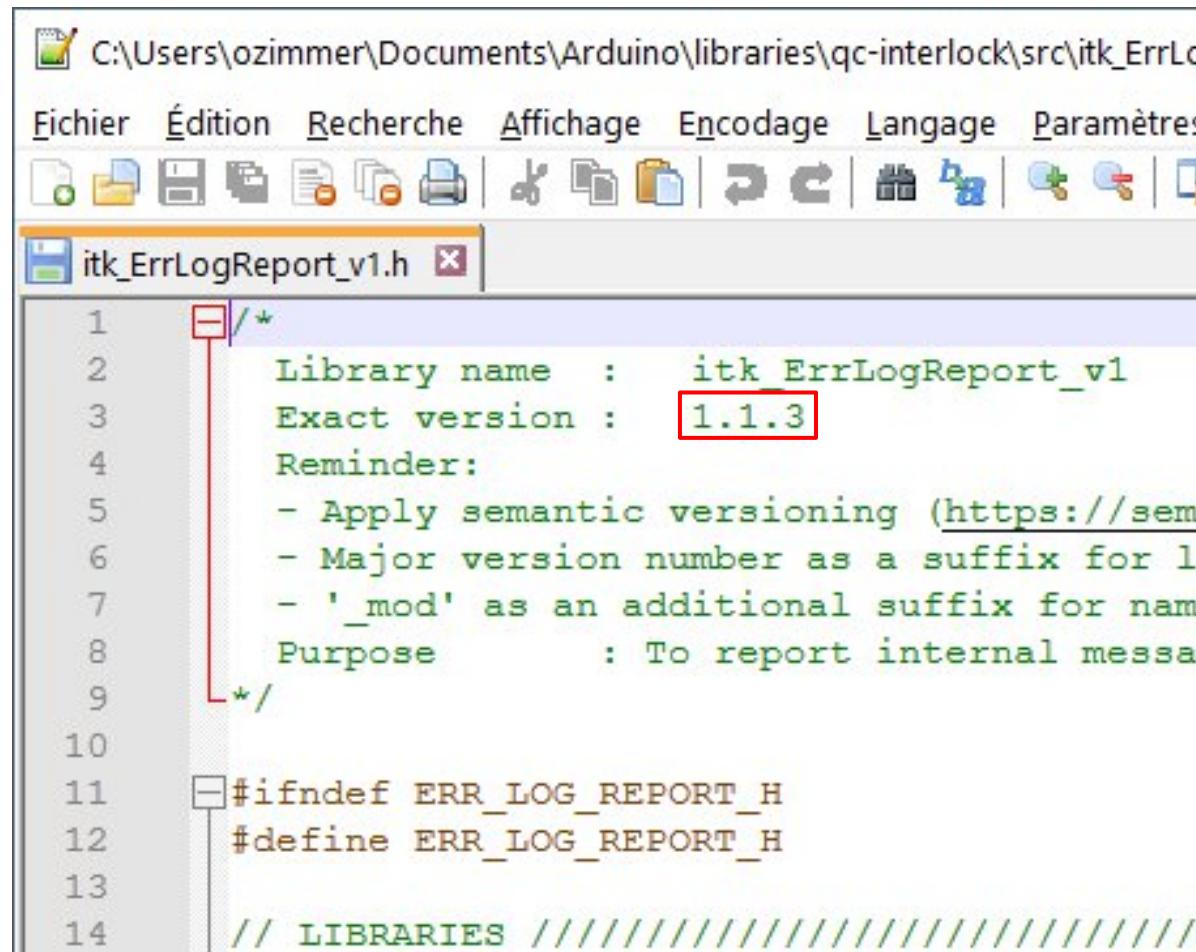
Utilisez un versionnage "semver" (.org) pour vos bibliothèques* et préfixez le nom des fichiers avec la version majeure.



- Si des modifications "cassantes" sont apportées, la nouvelle version majeure devra être visée explicitement dans chaque appelant → impose de vérifier l'impact sur l'appelant

```
C:\Users\ozimmer\Documents\Arduino\libraries\qc-interlock\apps\test_ErrLogReport.ino
Fichier Édition Recherche Affichage Encodage Langage Paramètres ...
test_ErrLogReport.ino
1 /* 
2  * Unit tests for the ErrLogReport library.
3  */
4
5 #include <itk_ErrLogReport_v1.h>
6 // Notice: you must properly configure flavor
```

Utilisez un versionnage "semver" (.org) pour vos bibliothèques* et préfixez le nom des fichiers avec la version majeure.



```
C:\Users\ozimmer\Documents\Arduino\libraries\qc-interlock\src\itk_ErrLogReport_v1.h

Fichier Édition Recherche Affichage Encodage Langage Paramètres
itk_ErrLogReport_v1.h

1  /* 
2   * Library name : itk_ErrLogReport_v1
3   * Exact version : 1.1.3
4   * Reminder:
5   * - Apply semantic versioning (https://semver.org/)
6   * - Major version number as a suffix for library name
7   * - '_mod' as an additional suffix for names
8   * Purpose : To report internal messages
9  */
10
11 #ifndef ERR_LOG_REPORT_H
12 #define ERR_LOG_REPORT_H
13
14 // LIBRARIES //////////////////////////////////////////////////////////////////
```

- Permet d'ajouter des fonctionnalités ou des correctifs (modifications "non-cassantes") sur une version majeure donnée.
Le nom ne change pas → le nouveau fichier sera automatiquement prise en compte par tous les fichiers utilisant cette version majeure .

* une bibliothèque = 1 fichier.h et 1 fichier.cpp.

Arduin (en italien : *Arduino*), né vers 955 à Pombia et mort le 14 décembre 1015 à l'abbaye de Fruttuaria, est un prince de la maison d'Ivrée (Anscarides), fils du comte Dadon de Pombia. Il fut marquis d'Ivrée de 990 jusqu'à sa mort. Usurpateur du trône d'Italie au début du xi^e siècle, il fut l'unique roi indépendant d'Italie durant cette période entre 1002 et 1004.



Placez votre dossier projet sous contrôle de version avec Git et GitLab.



firmware.ino

fw_constants.h

fw_procedures.h

fw_pro

fw_tima

fw_varia

The screenshot shows a web browser displaying a GitLab project page for 'qc-interlock'. The page includes a header with a cloud icon, a sidebar with project navigation, and a main area showing commit history, files, and settings. On the left, a TortoiseGit context menu is open over a file, with a red arrow pointing from the 'Git Commit -> "master"' option to the 'History' button in the GitLab interface.

Name	Last commit	Last update
about	add note about temperature calculation from NTC...	2 months ago
apps (deprecated)	Big code-base restructuring (re-integrating all libr...	4 months ago
apps	For november 2023 release.	2 weeks ago
data sheets (deprecated)/National Se...	Big code-base restructuring (re-integrating all libr...	4 months ago
data sheets	Add chip NTC datasheet.	2 months ago
kicad	Big code-base restructuring (re-integrating all libr...	4 months ago
src (deprecated)	Big code-base restructuring (re-integrating all libr...	4 months ago

Documentez le code au moyen de fichiers texte markdown.md.

The **itk_ErrLogReport_vX** library

Author(s)	Purpose	Notices
O.Zimmermann	To report internal messages (errors, etc) to the user. This library creates the <code>Report</code> shared, global object. Callers shall use this object to produce error or log reports. It may instantly push them to user (debug), or rather store them in a FIFO for further pulling and transmission (nominal). It may flush messages to serial (debug), or format them as JSON strings and store them to the dedicated global variables (nominal).	

itk_BME280_v1.h
itk_ErrLogReport.md (highlighted with a red box)
itk_ErrLogReport_v1.cpp
itk_ErrLogReport_v1.h
itk_GlobalMonitoringDataBase.i
itk_GlobalMonitoringDataBase_...
itk_GlobalMonitoringDataBase_...
itk_MAX31865.md
itk_MAX31865_v2.cpp
itk_MAX31865_v2.h
itk_MCP23S18.md
itk_MCP23S18_v1.cpp

```
1 [ << src files ] ( . )
2
3 The `itk_ErrLogReport_vX` library
4
```

Author(s)	Purpose	Notices
O.Zimmermann	****To report internal messages to the user.** This library creates the `Report` shared, global object. Callers shall use this object to produce error or log reports. It may instantly push them to user (debug), or rather store them in a FIFO for further pulling and transmission (nominal). It may flush messages to serial (debug), or format them as JSON strings and store them to the dedicated global variables (nominal).	

Release notes

1.1.3 (September 20, 2023)
- fix : now using "itk_GlobalMonitoringDataBase_v2".

1.1.2 (March 12, 2023)
- bug : flush() didn't filter out forbidden characters when flushing to JSON.

1.1.1 (March 8, 2023)
- fix : relocate flavour switches into cpp file (1. altering a switch there will force Arduino IDE to recompile the whole library with its new value, 2. keep them next to the source code)

Utilisez la fonctionnalité de "Release" de GitLab pour signaler et mettre à disposition les versions marquantes du projet à l'intention des utilisateurs.

The screenshot shows a web browser window displaying the GitLab 'Releases' page for the qc-interlock project. The URL in the address bar is <https://gitlab.cern.ch/lpsc-itk/ob-cell-loading/qc-interlock/-/releases>. The page title is 'Releases - lpsc-itk / ob-cell-loading'. The main content area shows a release titled 'november-2023-1.0.1'. Under 'Assets', there are five download links: 'Source code (zip)', 'Source code (tar.gz)', 'Source code (tar.bz2)', and 'Source code (tar)'. Below this is an 'Other' section containing a link to '231206-16h52-firmware.ino.hex.zip'. Under 'Evidence collection', there is a file named 'november-2023-1.0.1-evidences-15666.json' with a size of '4b2fff2a'. A note indicates it was 'Collected 6 days ago'. At the bottom, there is a 'What's in this release' section.

Utilisez la fonctionnalité de "Release" de GitLab pour signaler et mettre à disposition les versions marquantes du projet à l'intention des utilisateurs.

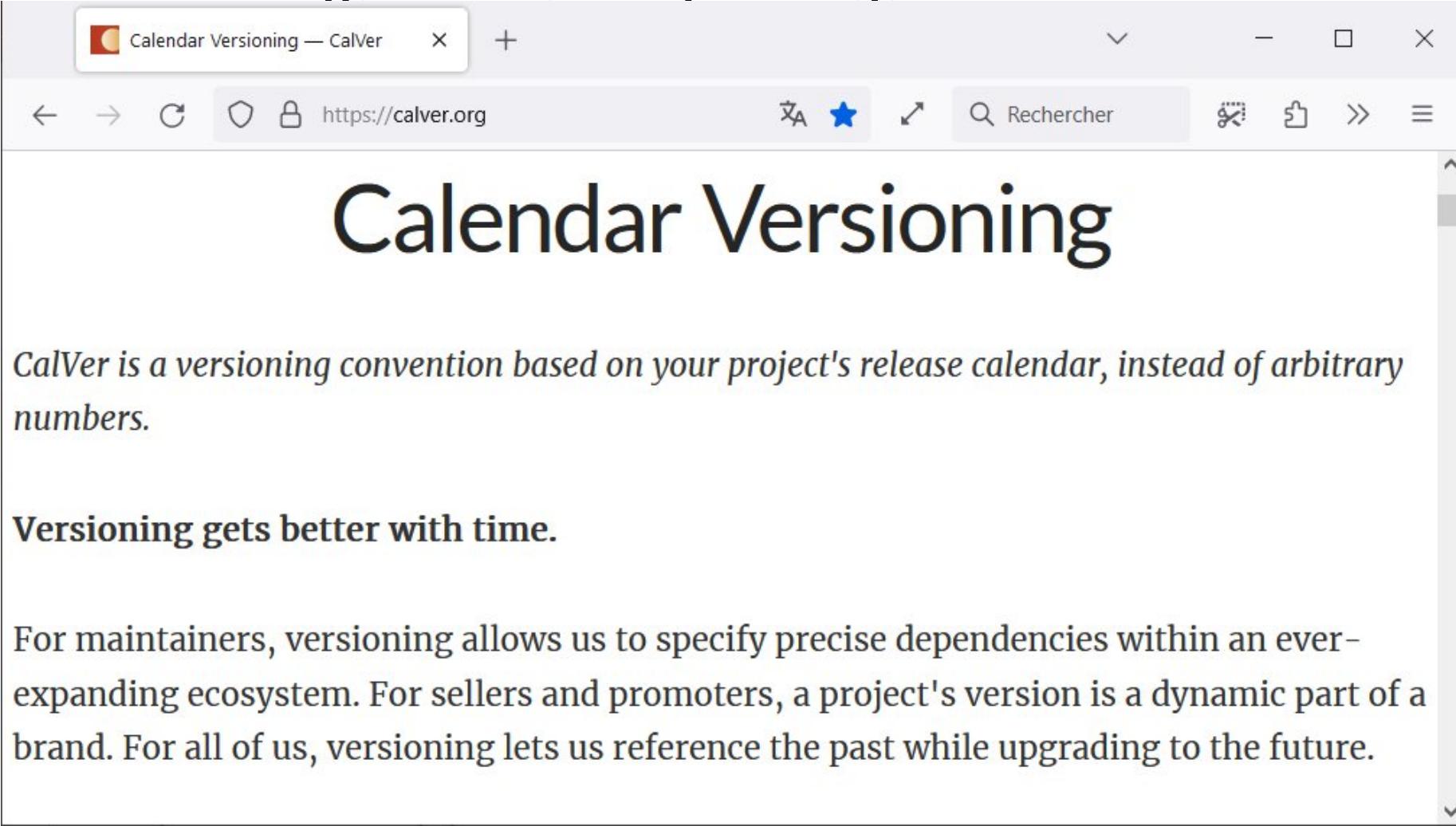
Possibilité de joindre par URL des "artefacts" ou fichiers annexes (par exemple binaire compilé de la release) stockés dans un cloud (...ou dans GitLab ?).

Utilisez la fonctionnalité de "Release" de GitLab pour signaler et mettre à disposition les versions marquantes du projet à l'intention des utilisateurs.

The screenshot shows a GitLab interface for a project named 'qc-interlock'. On the left, there's a sidebar with project navigation. The main area displays a release titled 'november-2023-1.0.1'. This release contains five assets: 'Source code (zip)', 'Source code (tar.gz)', 'Source code (tar.bz2)', 'Source code (tar)', and a file named '231206-16h52-firmware.ino.hex.zip'. Below the assets, there's an 'Evidence collection' section with a JSON file named 'november-2023-1.0.1-evidences-15666.json'. A red callout bubble points from the right towards the release title, containing the following text:

Adoptez un versionnage souple mais bien réfléchi pour aider les utilisateurs à se repérer dans vos releases (voir "calver.org").

Utilisez la fonctionnalité de "Release" de GitLab pour signaler et mettre à disposition les versions marquantes du projet à

A screenshot of a web browser window titled "Calendar Versioning — Calver". The URL in the address bar is "https://calver.org". The main content of the page is a large, bold title "Calendar Versioning" followed by a descriptive paragraph: "CalVer is a versioning convention based on your project's release calendar, instead of arbitrary numbers. Versioning gets better with time. For maintainers, versioning allows us to specify precise dependencies within an ever-expanding ecosystem. For sellers and promoters, a project's version is a dynamic part of a brand. For all of us, versioning lets us reference the past while upgrading to the future."

Calendar Versioning — Calver

https://calver.org

Calendar Versioning

CalVer is a versioning convention based on your project's release calendar, instead of arbitrary numbers.

Versioning gets better with time.

For maintainers, versioning allows us to specify precise dependencies within an ever-expanding ecosystem. For sellers and promoters, a project's version is a dynamic part of a brand. For all of us, versioning lets us reference the past while upgrading to the future.

