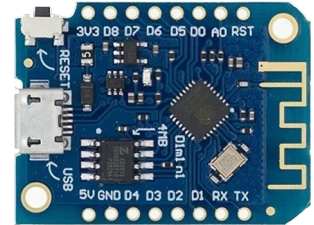




WeTimer

Installation logicielle et Programmation de la carte.

Pour être utilisée en tant que minuterie pour planeur de vol libre, la carte microcontrôleur Wemos D1 mini doit être chargée avec le programme WeTimer depuis un ordinateur (Windows, macOS ou Linux) à l'aide du logiciel Arduino IDE.



(Les copies d'écran de cette notice ont été réalisées sous Linux).

Installation du logiciel Arduino IDE :

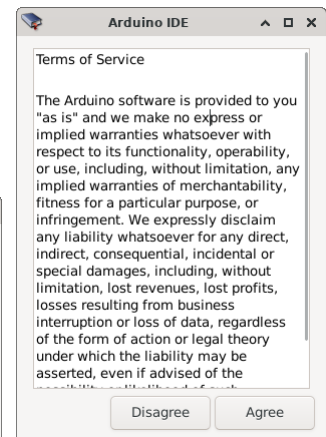
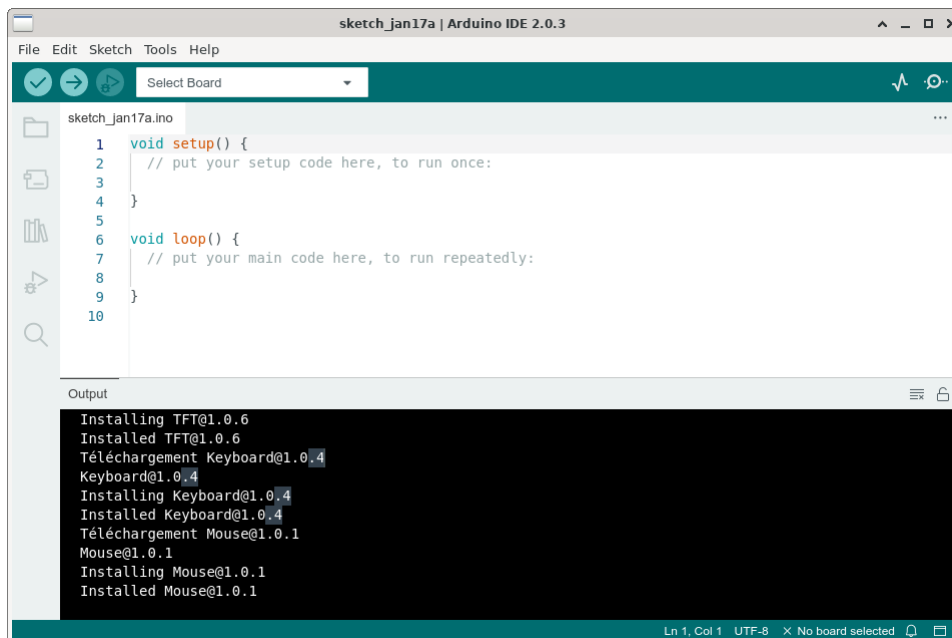
Le logiciel Arduino IDE est un logiciel open source pour la programmation de microcontrôleurs en langage C et C++. C'est ce logiciel qui est utilisé pour le développement du programme de la minuterie WeTimer.

Pour commencer, si vous ne l'avez pas déjà, téléchargez gratuitement la dernière version d'Arduino IDE ici : <https://www.arduino.cc/en/software>

Installez le sur votre ordinateur comme n'importe quel autre logiciel.

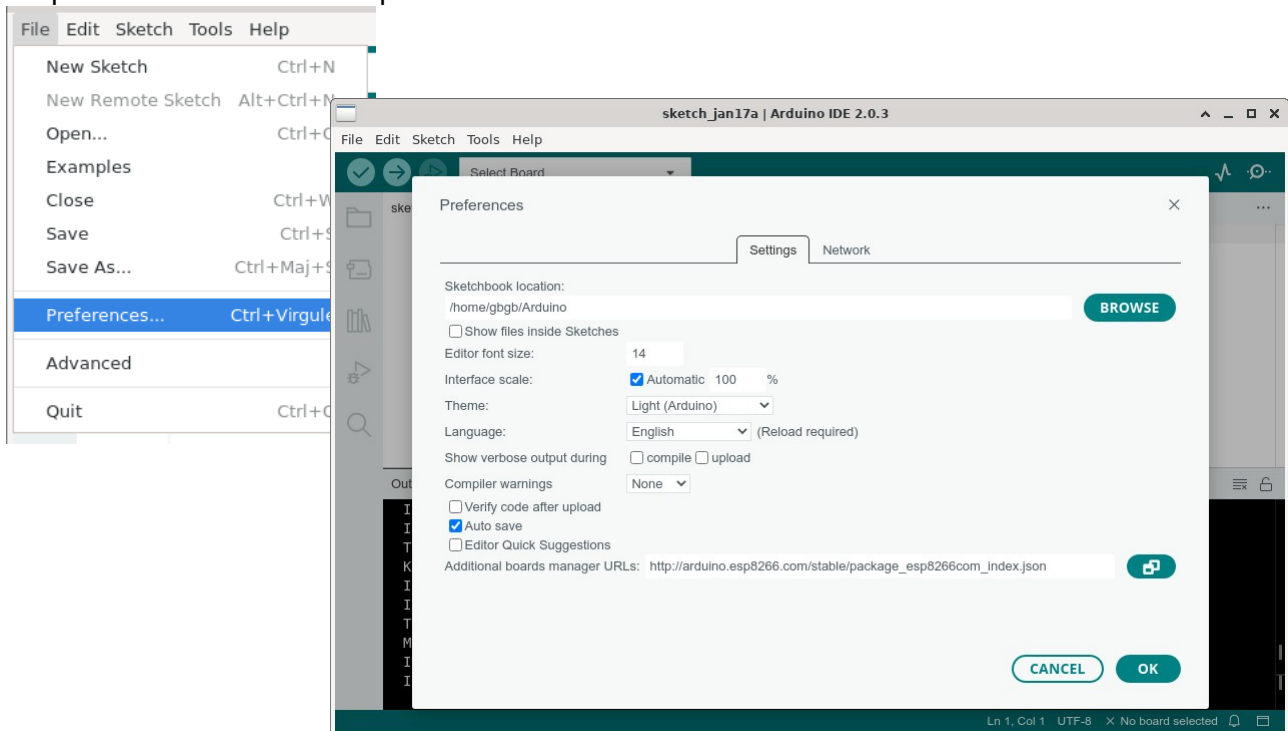
Au premier lancement du logiciel, vous devez accepter les termes de service en cliquant sur « Agree ».

Vous obtiendrez une fenêtre ressemblant à ceci :



Prise en charge des cartes ESP8266 dans l'IDE Arduino

Cliquez sur le menu « File » puis « Preferences » :

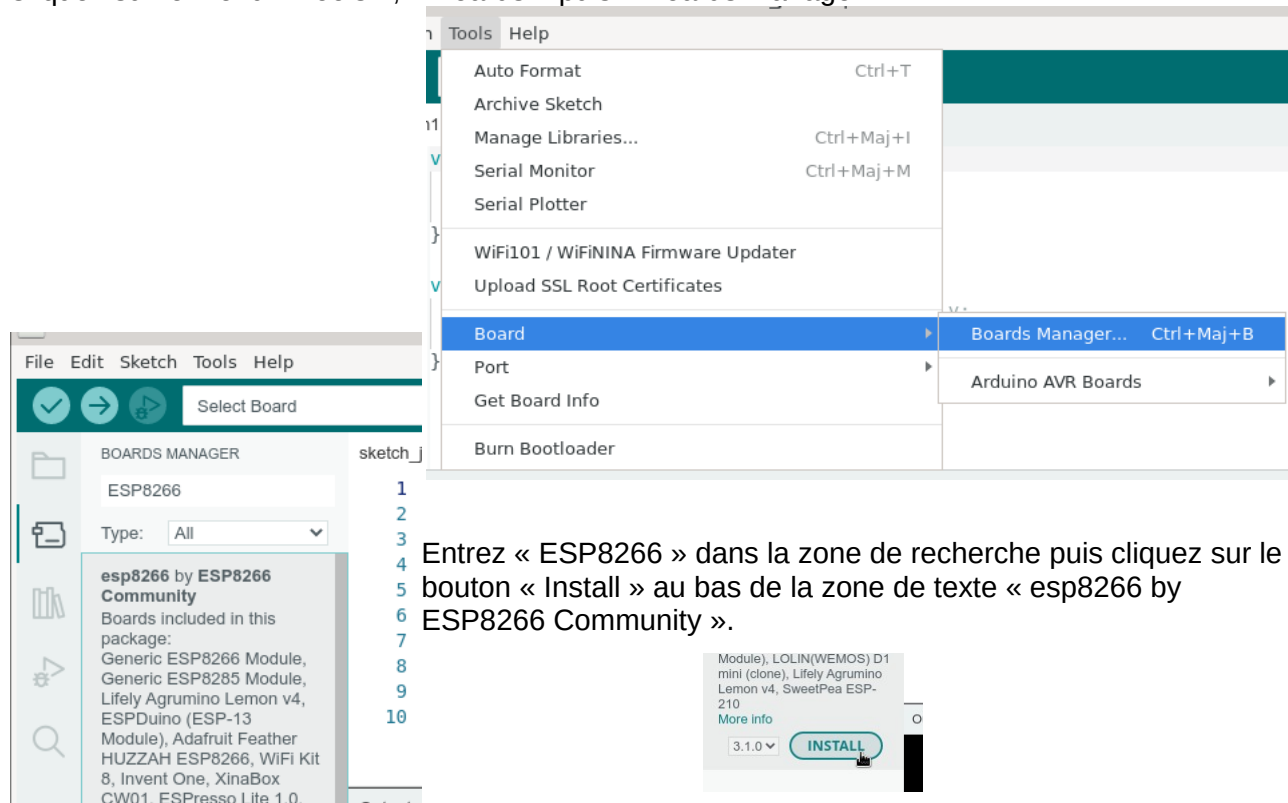


Dans la zone de saisie « Additional boards manager URLs : » ajoutez l'URL suivante :

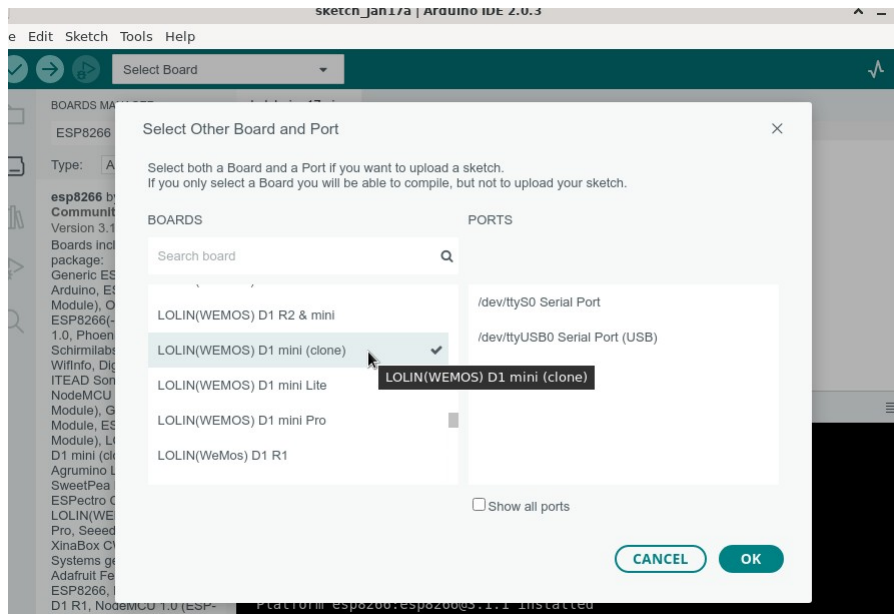
http://arduino.esp8266.com/stable/package_esp8266com_index.json

Note : On peut entrer plusieurs URLs de cartes en les séparants par des virgules.

Cliquez sur le menu « Tools », « Boards » puis « Boards manager

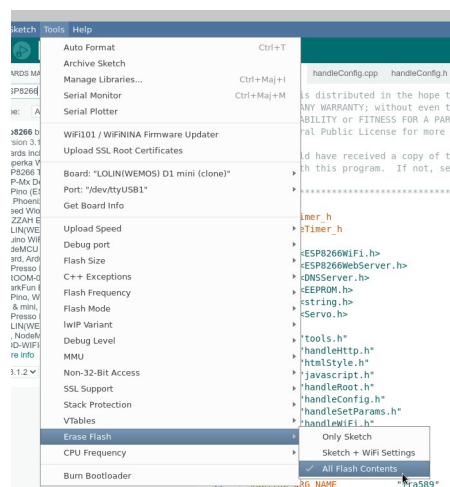


Après avoir cliqué sur « Select Board », faire défiler la liste des cartes qui s'affiche, puis sélectionnez la carte « LOLIN(WEMOS) D1 mini (clone) » :



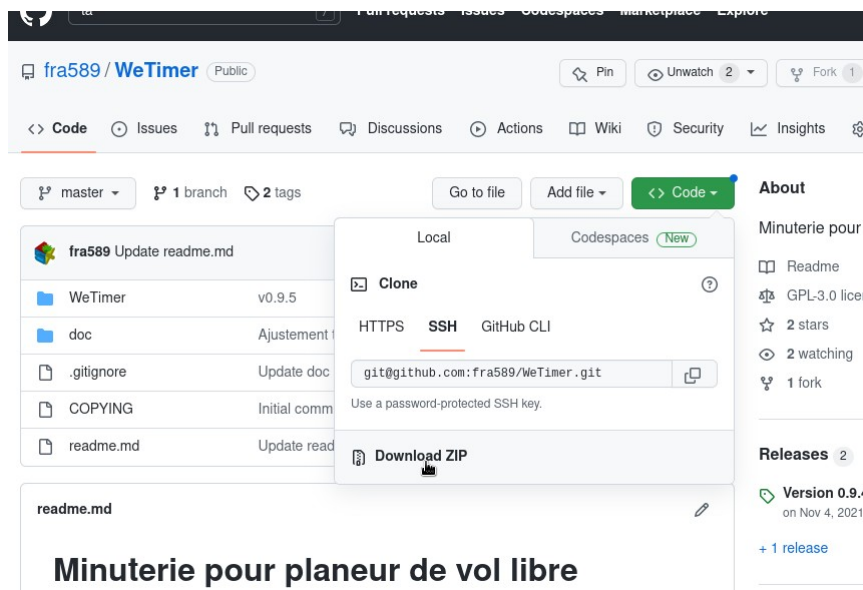
Sélectionnez également le port sur lequel vous avez connecté la carte, puis, cliquez sur « OK ».

Dans le menu « Tools », cliquez sur « Erase Flash », puis sélectionnez « All Flash Contents » pour une réinitialisation correcte de la carte lors de la programmation de WeTimer.

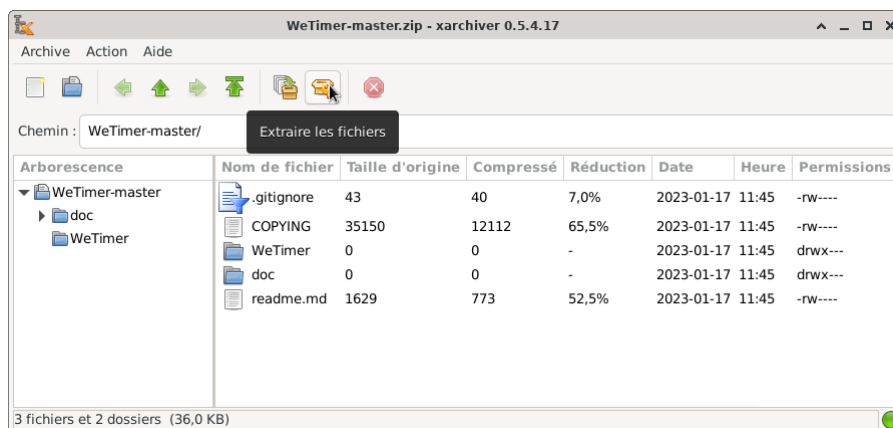


Programme WeTimer :

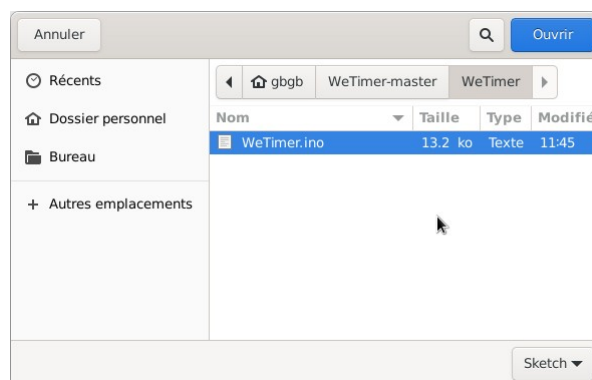
Téléchargez le programme WeTimer depuis Github ici : <https://github.com/fra589/WeTimer>
Cliquez sur le bouton vert « <> Code ▾ », puis sur « Download ZIP ».



Décompressez le fichier téléchargé à l'endroit de votre choix sur votre ordinateur en conservant l'organisation et le nom des dossiers à l'intérieur du ZIP.



Dans l'IDE Arduino, cliquez sur le menu « File » puis « Open » et sélectionnez le fichier « Wetimer.ino » dans le dossier décompressé « WeTimer-Master/WeTimer/ ».

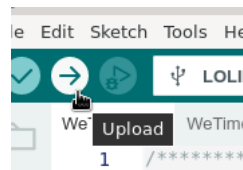


Une nouvelle fenêtre contenant les sources du programme de la minuterie s'ouvre :

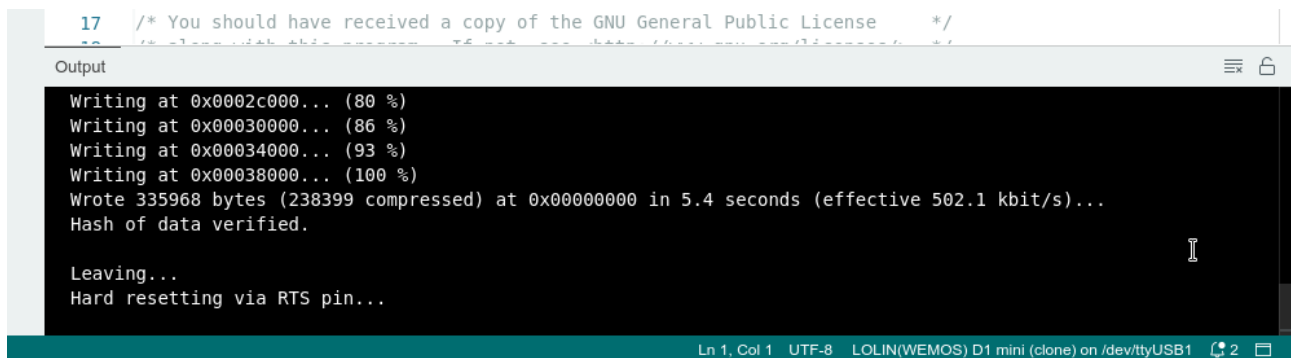


```
1  /*****  
2  /*  
3  /* Copyright (C) 2021 Gauthier Briere (gauthier.briere "at" gmail.com) */  
4  /*  
5  /* This file: WeTimer.ino is part of WeTimer  
6  /*  
7  /* WeTimer is free software: you can redistribute it and/or modify it  
8  /* under the terms of the GNU General Public License as published by  
9  /* the Free Software Foundation, either version 3 of the License, or  
10 /* (at your option) any later version.  
11 /*  
12 /* WeTimer is distributed in the hope that it will be useful, but  
13 /* WITHOUT ANY WARRANTY; without even the implied warranty of  
14 /* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
15 /* GNU General Public License for more details.  
16 /*  
17 /* You should have received a copy of the GNU General Public License  
18 /* along with this program. If not, see <http://www.gnu.org/licenses/>. */  
19 /*  
20 /*  
21 /*  
22 #include "WeTimer.h"  
23  
24 //-----  
25 // Variables globales  
26 //-----  
27  
28 char cli_ssid[32] = DEFAULT_CLI_SSID;  
29 char cli_pwd[63] = DEFAULT_CLI_PWD;
```

Cliquez sur le bouton « → » (Upload) pour compiler et envoyer le programme dans la carte.



A la fin du traitement, lorsque vous voyez le message :



```
17 /* You should have received a copy of the GNU General Public License */  
18 /*  
Output  
Writing at 0x0002c000... (80 %)  
Writing at 0x00030000... (86 %)  
Writing at 0x00034000... (93 %)  
Writing at 0x00038000... (100 %)  
Wrote 335968 bytes (238399 compressed) at 0x00000000 in 5.4 seconds (effective 502.1 kbit/s)...  
Hash of data verified.  
  
Leaving...  
Hard resetting via RTS pin...  
Ln 1, Col 1 UTF-8 LOLIN(WEMOS) D1 mini (clone) on /dev/ttyUSB1 2
```

Cela indique que le téléchargement est termine sur la carte microcontrôleur et que celle ci à été redémarrée.

La minuterie est maintenant fonctionnelle !