

# D1-Mini / MEGA-WiFi installation

## Installation commune sous Windows

### 1) Informations générales

Cette réalisation est un exemple proposé sans aucune garantie et reste sous l'entière responsabilité de son utilisateur. L'auteur de cette présentation ne peut être tenu responsable de la mauvaise utilisation de **cette réalisation qui demande un minimum de compétences techniques**. L'emploi de vieux logiciels et matériels : PC 32 bits, Smartphone ou de vieille tablette peuvent présenter des réactions lentes ou incompatibles et ne sont pas recommandés. Les informations dans ce document peuvent changer sans préavis selon les mises à jour des logiciels et n'engagent aucunement la responsabilité de l'auteur.

Ce logiciel **WDD6** vous permet de conduire vos locomotives, de programmer les **CV**, et grâce à l'ajout d'un **TCO** entièrement configurable selon votre réseau, de commander vos accessoires et aiguillages et de lire un bus de rétro-signalisation compatible à la norme **S88** en temps réel. En supplément **DMC** commande les trains depuis un Smartphone. Tous les fichiers nécessaires au fonctionnement du logiciel sont contenus dans la mémoire Flash de l'ESP8266 qui sert de serveur et donc cette **centrale DCC WiFi** est autonome.

**Aucune connexion Internet est nécessaire pour son fonctionnement sauf si vous chargez certains fichiers depuis Internet.**

**WDD6**, **TCOwifi** et **DMC** sont totalement compatibles avec la bibliothèque **DCCppS88** et partiellement compatible avec les bibliothèques **DCCpp** et **DCC++**. Toutes les fonctions de base sont compatibles sauf la rétro-signalisation **S88** qui est une option exclusive de **DCCppS88**.

**Configuration recommandée pour le PC : une version 64-bit avec Windows 10.**

Ce projet a été compilé avec l' **IDE Arduino 1.8.19 ou 2.3.2** et testé avec les matériels suivants :

- **Windows 10** avec un **PC** 64-bit Xeon à 3.2GHz (4-core 8-thread), RAM 8Mo DDR3 et une interface réseau à 100Mb/s avec une Box **WiFi**
- **Windows 10** avec un **PC** portable 64-bit I5 à 1.6GHz (4-core 8-thread), RAM 8Mo DDR4 et une interface réseau WiFi une Box **WiFi**

**Centrale DccppS88** sur :

- Genuine Arduino Mega 2560 + D1-Mini + L298
- Clone Mega2560 WiFi R3 + L298
- Clone Mega 2560 + LM18200
- Clone Mega 2560 + DigitalBooster DB-4 de LDT (CDE)

**Client WiFi :**

- iPad1, Smartphone, PC portable ASUS, ...

**Navigateur Internet :**

- **Chrome, Edge, Firefox, Opera, Safari**

**Attention** : sur la carte **MEGA WiFi R3** avec une antenne externe, prenez garde de ne pas être trop proche du routeur de votre Box afin de conserver un niveau de réception compris dans cette amplitude, **-75 dBm < RSSI < -45 dBm**, pour ne pas saturer la partie radio du routeur de votre Box, ni avoir un niveau de réception trop faible. Ceci se traduirait par un mauvais téléchargement des fichiers conduisant à un dysfonctionnement du logiciel. Cette valeur peut varier selon le routeur utilisé. Si vous êtes trop loin, cela ne fonctionnera pas non plus. Cette remarque de proximité ne concerne pas le **D1-Mini**.

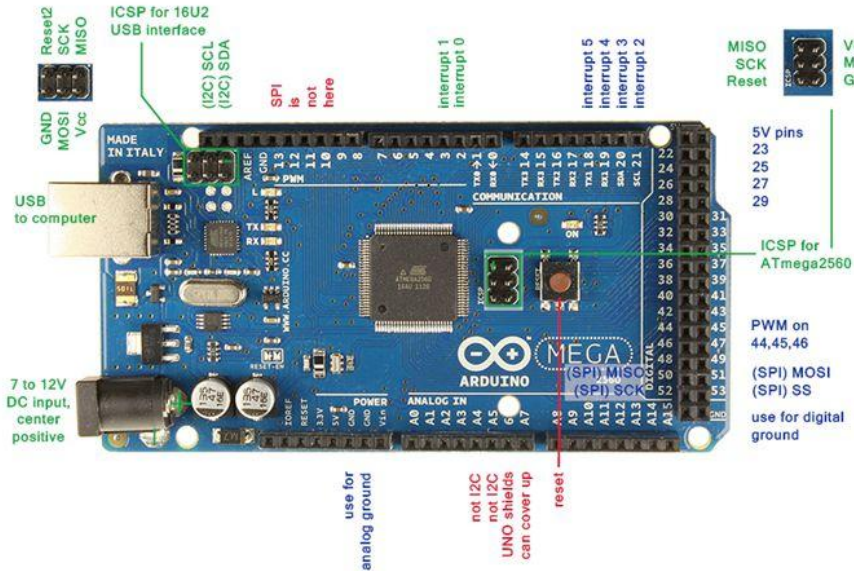
*Note : Il se peut que certains Smartphones ou autres équipements ne fonctionnent pas comme prévu. Essayez avec votre matériel.*

## D1-Mini / MEGA-WiFi installation

## 2) Matériel nécessaire

Le **MEGA2560** génère le **DCC** sous le contrôle de **DCCppS88** qui reçoit les commandes depuis l'interface **WiFi**. Cette interface **WiFi** est réalisée avec un **ESP8266** qui gère les communications réseau entre les différents intervenants extérieurs et le **MEGA2560**. L'**ESP8266** fait aussi office de serveur de fichiers avec le contenu de sa mémoire Flash où sont stockés les fichiers des logiciels **WDD6**, **TCOWifi**, **DMC**, **etc**, qui pilotent la centrale **DCC**.  
**Attention : certains clones posent des problèmes de communication sur la broche Rx0.**

### 2.1) Version 1, *D1-Mini + MEGA2560*



### Figure 1 : Arduino MEGA2560

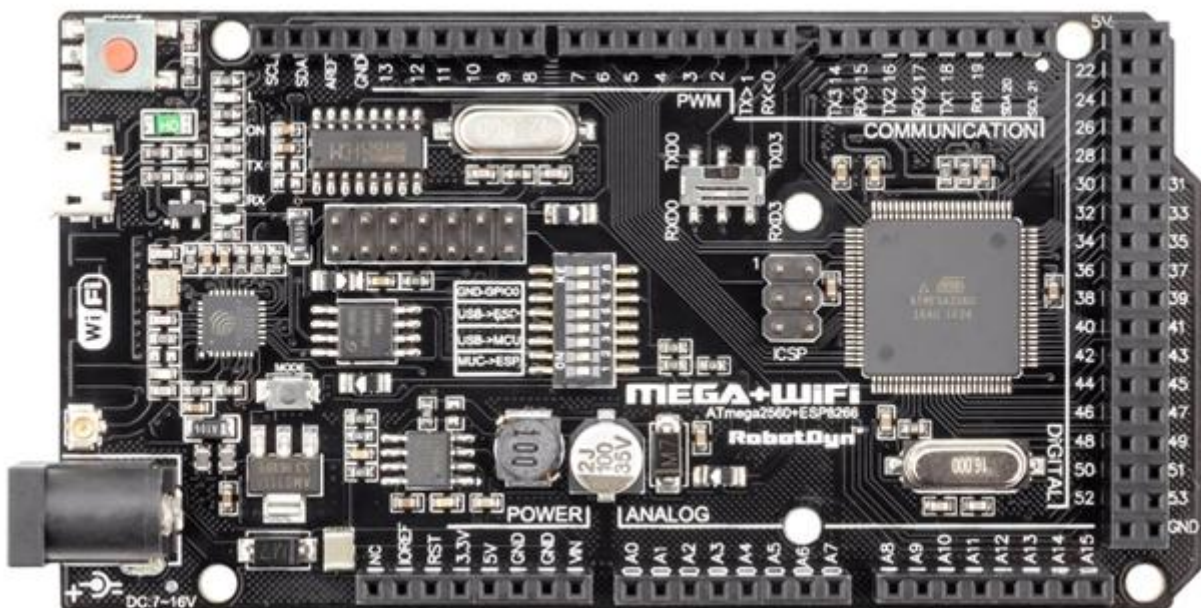


Figure 2 : WEMOS D1-Mini

**MEGA2560** classique avec **D1-Mini** qui lui sert d'interface WiFi.

*ou*

## 2.2) Version 2, *Mega WiFi R3*



### Figure 3 : MEGA WiFi R3

Cette carte regroupe un **MEGA2560** et un **ESP8266**.

# D1-Mini / MEGA-WiFi installation

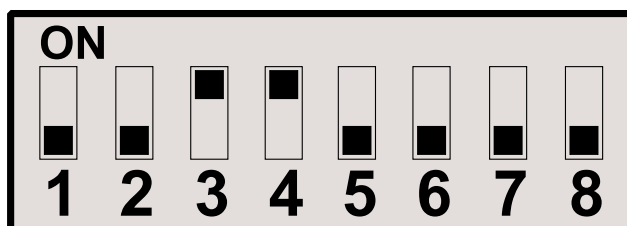
## 3) Procédure d'installation ou de mise à jour de l'Arduino Mega2560

Dans le répertoire **Mes Documents**, créer un sous-répertoire **Arduino** dans lequel vous créez un sous-répertoire **DCCppS88** dans lequel dé-zippez les fichiers contenus dans **DCCppS88.zip**.

A) Ouvrir l'**IDE Arduino** et sélectionner le répertoire **DCCppS88** ci-dessus. Ouvrir le fichier **DCCppS88.ino**.

B) Dans le menu **Outils – Type de carte** sélectionnez **Arduino/Genuino Mega ou Mega2560**

C) Sur un port **USB** de votre **PC**, brancher votre carte **Mega2560** ou **Mega WiFi R3** en configurant ses commutateurs DIP pour téléverser le code dans le **MEGA2560** (**DIP3 et DIP4 : ON**). Appuyer sur Reset pendant 3 secondes.



Allez dans **Outils – Type de carte – Gestionnaire de carte**

D) Dans le menu **Outils**, avec une liaison **USB** sélectionnez le port correspondant à la ligne Port **COMx**.

E) Compiler le code avec l'**IDE Arduino** et téléverser le code dans votre **MEGA2560**. (la liaison série fonctionne à **115200 Bauds**).

F) Vérification :

Une fois chargé dans le **MEGA2560**, ouvrir l'interface de communication de l'**IDE Arduino** configurée pour **115200 baud sans fin de ligne** et vérifier que le Moniteur affiche bien les informations suivantes :

```
Initialisation de la liaison serie USB2COM 115200 bauds
Programme de conduite DCCppS88 pour Arduino UNO/MEGA2560 + L298N
Adaptation par Philippe Chavatte - 31 juillet 2018 - lormedy@free.fr
Option S88 par Philippe Chavatte - 25 mai 2019 - lormedy@free.fr
VERSION DCCpp library: 1.3.2 + S88
Compatible avec CDM_Rail, CDT, JMRI et Rocrail
----- Module DCC initialisé avec serial port -----
*** DCCpp LIBRARY ***
VERSION DCC++:      2.0.1
VERSION DCCpp library: 1.3.5
VERSION S88 library:  1.3.0
COMPILED:  Sep 14 2019 19:07:19

DCC SIG MAIN(DIR): 12
  DIRECTION: 255
  ENABLE(PWM): 3
  CURRENT: 54/A0

DCC SIG PROG(DIR): 2
  DIRECTION: 255
  ENABLE(PWM): 11
  CURRENT: 55/A1

S88_Clock_PIN:  5
S88_Load_PS_PIN: 6
S88_Reset_PIN:  7
S88_DataL_PIN:  8
S88_DataR_PIN:  9

USE EEPROM:
  S88  M: 8

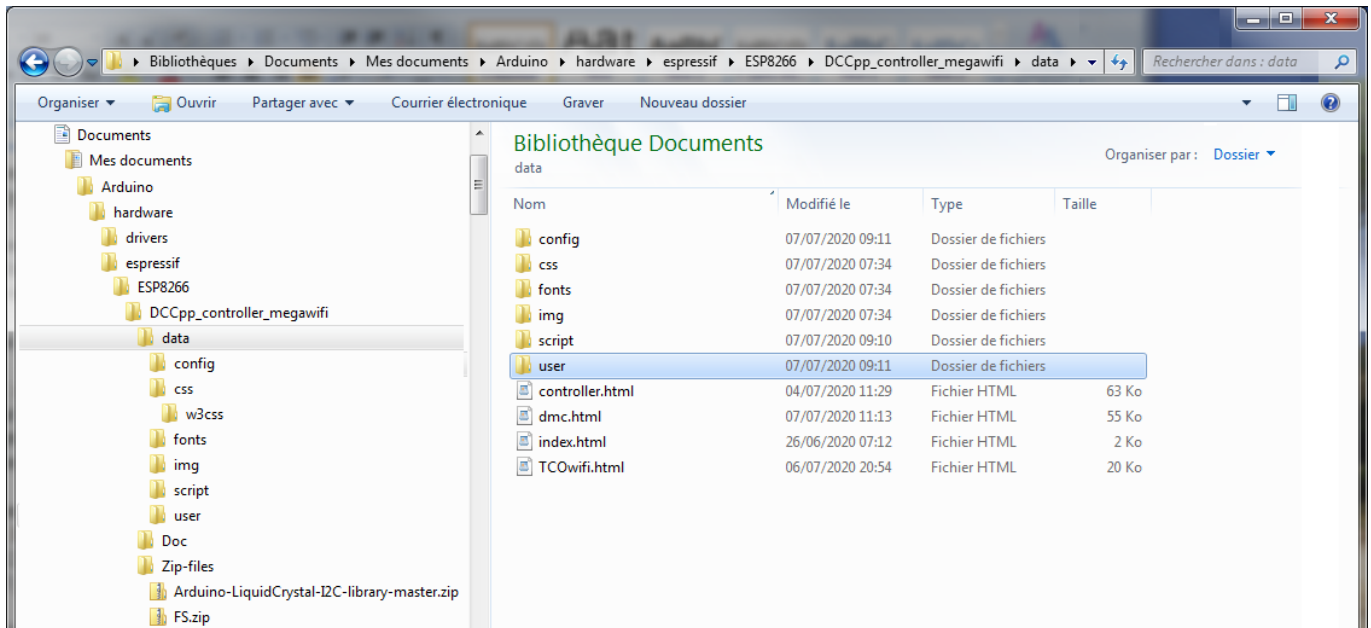
INTERFACE:  SERIAL
DCCppS88 ready
```

\*\*\*\*\* N'oubliez pas de fermer cette fenêtre "Moniteur" pour pouvoir programmer les Data dans la Flash. \*\*\*\*\*

# D1-Mini / MEGA-WiFi installation

## 4) Architecture des répertoire du projet WDD6/TCOwifi/DMC

Dans le répertoire **Mes Documents**, créer un sous-répertoire **Arduino** dans lequel vous créez un sous-répertoire **hardware** dans lequel vous créez un sous-répertoire **espressif** dans lequel vous créez un sous-répertoire **ESP8266**.



Dézipper **WDD6.zip** dans ce répertoire **ESP8266**. Vous venez de créer le projet **DCCpp\_controller\_BTWifi** dans lequel se trouvent tous les fichiers nécessaires au fonctionnement de cette centrale **DCC WiFi**.

### Note importante :

Tous les fichiers personnels se trouvent dans le répertoire **user/**. Ce sont les seuls fichiers que vous pouvez éditer si besoin. Vous pouvez ajouter ici des photos de locos, format maxi recommandé 300x270px, dans la limite de **700Ko** environ. La longueur totale des noms des fichiers ne doit pas excéder **27 caractères**. Pour préserver la place dans mémoire Flash, il est préférable de stocker ses photos sur un autre serveur de fichiers ou sur Internet. **Ne pas stocker ici de fichiers temporaires au risque de perturber le téléchargement du projet.**

On y trouve essentiellement :

- **user/locos.json** dans lequel sont enregistrés tous les **CV** et paramètres des locos
- **user/Layout.js** dans lequel est enregistré le dessin du **TCO**

## 5) Procédure d'installation ou de mise à jour du code de l'ESP8266

Pour pouvoir compiler les **ESP8266**, suivre la procédure décrite sur la page Internet :  
<http://lormedy.free.fr/Mega-Wifi-R3-setup.html>

- A) Démarrer l'**IDE Arduino** et ouvrir les fichiers du projet **DCCpp\_controller\_BTWifi.ino** qui se trouve dans le répertoire éponyme.
- B) Dans le fichier **I2C\_LCD.cpp** du projet **DCCpp\_controller\_BTWifi.ino** indiquez l'adresse de votre afficheur LCD aux lignes 17-19.
- C) **Ancienne version** : Editer **WifiConfig.h** et indiquez le **nom** de votre point d'accès **WiFi** ainsi que son **mot-de-passe** pour autoriser la connexion **WiFi**. Si vous utilisez un D1-Mini, la première ligne doit être mise en commentaire.



# D1-Mini / MEGA-WiFi installation

**Nouvelle version :** Editer **conf.json** dans le repertoire « **data** » et indiquez le **nom** de votre point d'accès **WiFi** ainsi que son **mot-de-passe** pour autoriser la connexion **WiFi**.

Si vous utilisez un D1-Mini, éditer **WifiConfig.h** du projet, la ligne 18 doit être mise en commentaire.

Si vous utilisez un MEGA R3 WIFI, éditer **WifiConfig.h** du projet, la ligne 18 ne doit pas être mise en commentaire.

**D)** Ouvrir l'onglet **Outils - Type de carte** et sélectionnez **Generic ESP8266 Module**. (ou **Lolin D1 R2 & mini** pour la 1ère version)

**E)** Dans l'onglet **Outils - Flash size** et sélectionnez **Flash size: 4M (3M SPIFFS)** puis à la ligne **Upload Speed** à **512000** et enfin **Builtin Led** indiquer **"14"** pour la carte MEGA R3 WIFI et **"2"** pour le D1-Mini.

**F)** Dans le menu **Outils**, avec une liaison **USB** sélectionnez le port correspondant à la ligne Port **COMx**. Si vous utilisez **OTA**, sélectionnez le Port **IP** dont l'adresse doit être visible en bas de l'onglet **Outils - Port**. Dans ce cas vous n'avez pas besoin de modifier les commutateurs DIP.

Outils	Aide
Formatage automatique	Ctrl+T
Archiver le croquis	
Réparer encodage & recharger	
Gérer les bibliothèques	Ctrl+Maj+I
Moniteur série	Ctrl+Maj+M
Traceur série	Ctrl+Maj+L
WiFi101 / WiFinINA Firmware Updater	
ESP8266 Sketch Data Upload	
Type de carte: "Generic ESP8266 Module"	▶
Upload Speed: "512000"	▶
CPU Frequency: "80 MHz"	▶
Flash Frequency: "40MHz"	▶
Flash Mode: "DOUT (compatible)"	▶
Flash Size: "4M (3M SPIFFS)"	▶
Crystal Frequency: "26 MHz"	▶
Reset Method: "ck"	▶
Debug port: "Disabled"	▶
Debug Level: "Rien"	▶
lwIP Variant: "v2 Lower Memory"	▶
VTables: "Flash"	▶
Exceptions: "Disabled"	▶
Builtin Led: "14"	▶
Erase Flash: "Only Sketch"	▶
Espressif FW: "nonos-sdk 2.2.1 (legacy)"	▶
SSL Support: "All SSL ciphers (most compatible)"	▶
Port: "COM6"	▶
Récupérer les informations de la carte	
Programmeur: "AVRISP mkII"	
Graver la séquence d'initialisation	

Outils	Aide
Formatage automatique	Ctrl+T
Archiver le croquis	
Réparer encodage & recharger	
Gérer les bibliothèques	Ctrl+Maj+I
Moniteur série	Ctrl+Maj+M
Traceur série	Ctrl+Maj+L
WiFi101 / WiFinINA Firmware Updater	
ESP32 Sketch Data Upload	
ESP8266 Sketch Data Upload	
Type de carte: "LOLIN(WEMOS) D1 mini (clone)"	
Upload Speed: "512000"	
CPU Frequency: "80 MHz"	
Flash Size: "4MB (FS:3MB OTA:~512KB)"	
Flash Mode: "DOUT (compatible)"	
Flash Frequency: "40MHz"	
Debug port: "Disabled"	
Debug Level: "Rien"	
lwIP Variant: "v2 Lower Memory"	
VTables: "Flash"	
C++ Exceptions: "Disabled (new aborts on oom)"	
Stack Protection: "Disabled"	
Erase Flash: "Only Sketch"	
SSL Support: "All SSL ciphers (most compatible)"	
MMU: "32KB cache + 32KB IRAM (balanced)"	
Non-32-Bit Access: "Use pgm_read macros for IRAM/PROGMEM"	
Port: "COM5"	
Récupérer les informations de la carte	
Programmeur	
Graver la séquence d'initialisation	

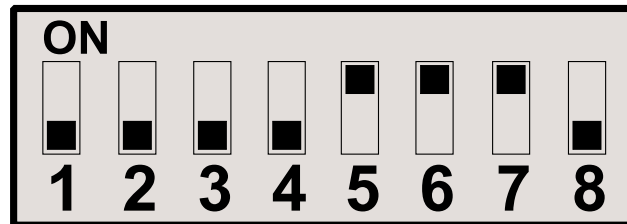
# D1-Mini / MEGA-WiFi installation

**G)** Compiler les fichiers avec l'**IDE Arduino** et ensuite téléverser le code dans votre **ESP8266**. (la liaison série fonctionne à **512000 Bauds**).

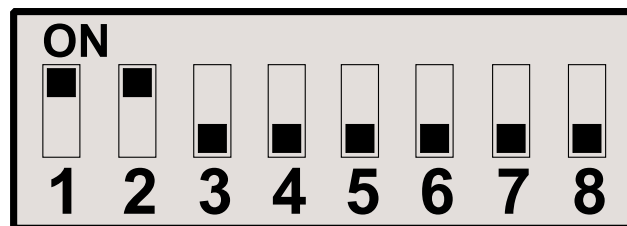
**H1) version D1-Mini)** Sur un port **USB** de votre **PC**, brancher votre carte **D1-Mini** et connecter D3 avec GND. Appuyer sur **Reset** pendant 2 secondes. Après 3 secondes, libérer D3 puis téléverser le code dans l'**ESP8266**.

ou

**H2) version Mega WiFi R3)** Sur un port **USB** de votre **PC**, brancher votre carte **Mega WiFi R3** et configurer ses commutateurs DIP (**DIP5, DIP6 et DIP7 : ON**). Appuyer sur **Reset** pendant 2 secondes puis téléverser le code dans l'**ESP8266**.



**I) Vérification :** Une fois chargé dans le **ESP8266**, configurer ses commutateurs DIP pour Booter l'**ESP8266**.



Ouvrir l'interface de communication de l'**IDE Arduino** configurée pour **115200 baud sans fin de ligne**. Vérifier que le Moniteur affiche bien les informations suivantes au démarrage :

```
ESP8266 D1 mini:
Initialisation de la liaison Bluetooth à 9600 bauds
Initialisation de la liaison série 0 à 115200 bauds
Initialisation de la liaison série 1 à 115200 bauds
I2C Scanning...
I2C device found at address 0x3F
I2C scan done

Project:      Controller DCCppS88 WiFi + Bluetooth
Version:      V1.23
Compiled:     Jul  3 2022 13:48:11

Wifi diags:
Access Point  "DCCppS88" started
IP address:    192.168.4.1

Connecting.....
Station connected to STA "freebox_menthalo"
IP address:    192.168.0.155
RSSI:         -51 dBm

mDNS responder started: http://dccpps88.local

OTA ready

SPIFFS started
WebSocket server started
HTTP server started

Info système:
Fréq. CPU      : 80 MHz
Flash Speed    : 40 MHz
Flash total    : 16777216
Flash real size : 4194304
```

# D1-Mini / MEGA-WiFi installation

```
Taille sketch      : 398016
Flash Mode         : 3
Sketch maxi        : 647168
RAM total          : 81920
RAM libre          : 38552
SPIFFS total       : 2949250
SPIFFS utilisée    : 2493434
SPIFFS blockSize   : 8192
SPIFFS pageSize    : 256
SPIFFS maxOpenFiles : 5
SPIFFS maxPathLength : 32
Adresse MAC        : 50:02:91:BA:B9:E1

ESP8266 ADC0 = 3300 mV (1024)
ESP8266 Serial COM is now connected to MEGA2560 serial interface
ESP8266-D1-mini + MEGA2560 + DCCppS88 ready
ou
MEGA-R3-WIFI with ESP8266 + MEGA2560 + DCCppS88 ready
```

*Note : les noms de fichier et les chiffres indiqués ci-dessus correspondent à une configuration, la votre est différente.*

Dans ces lignes, repérer l'adresse **IP** que votre Box a donné à votre **centrale DCC WiFi**. Elle servira pour appeler le logiciel **WDD6**, **TCOWifi** et **DMC** depuis le navigateur de votre **PC**.

A partir de cette étape vous pourrez utiliser la mise à jour de l'**ESP8266** en **WiFi** avec **OTA**.

**\*\*\* N'oubliez pas de fermer cette fenêtre "Moniteur" pour pouvoir programmer les Data dans la Flash. \*\*\***

**J) Affichage LCD** en option (adresse par défaut 0x27 ou 0x3F)

L'afficheur **LCD** (LCD2004) en **I2C** affiche 4 lignes d'information détaillées ci-dessous :

1. Nom et le version du code de l'**ESP8266**
2. Adresse **IP** de la **centrale DCC WiFi**
3. Log du boot de l'**ESP8266** puis adresse **IP** du client à la connexion. Ensuite toutes commandes **DCC** reçues.
4. Log du boot du **MEGA** puis mode de connexion. Ensuite toutes commandes **DCC** acquittées par le **MEGA**.

## 6) Procédure d'installation ou de mise à jour des data de l'application HTML/JS

**Avant de commencer vous devez vous assurer de l'installation dans Windows 10 de Java 8.0 pour un PC 64-bit.**

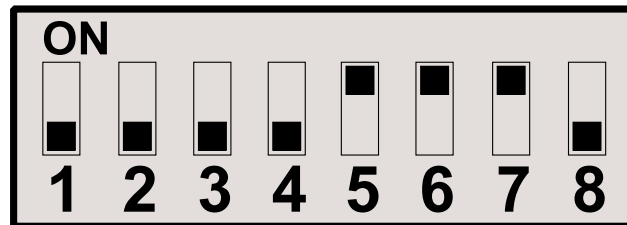
Lisez la rubrique "**Comment transférer un fichier dans la mémoire SPIFFS**" sur cette page Internet :  
<https://byfeel.info/utilisation-de-la-memoire-spiffs-sur-esp8266/>

**A1) version D1-Mini)** Sur un port **USB** de votre **PC**, brancher votre carte **D1-Mini** et c'est tout.

**A2) version Mega WiFi R3)** Sur un port **USB** de votre **PC**, brancher votre carte **Mega WiFi R3** et configurer ses commutateurs DIP pour téléverser le code dans l'**ESP8266**. (**DIP5**, **DIP6** et **DIP7** : **ON**).

Si vous utilisez **OTA**, sélectionnez le Port **IP** dont l'adresse doit être visible en bas de l'onglet **Outils - Port**. Dans ce cas vous n'avez pas besoin de modifier les commutateurs DIP.

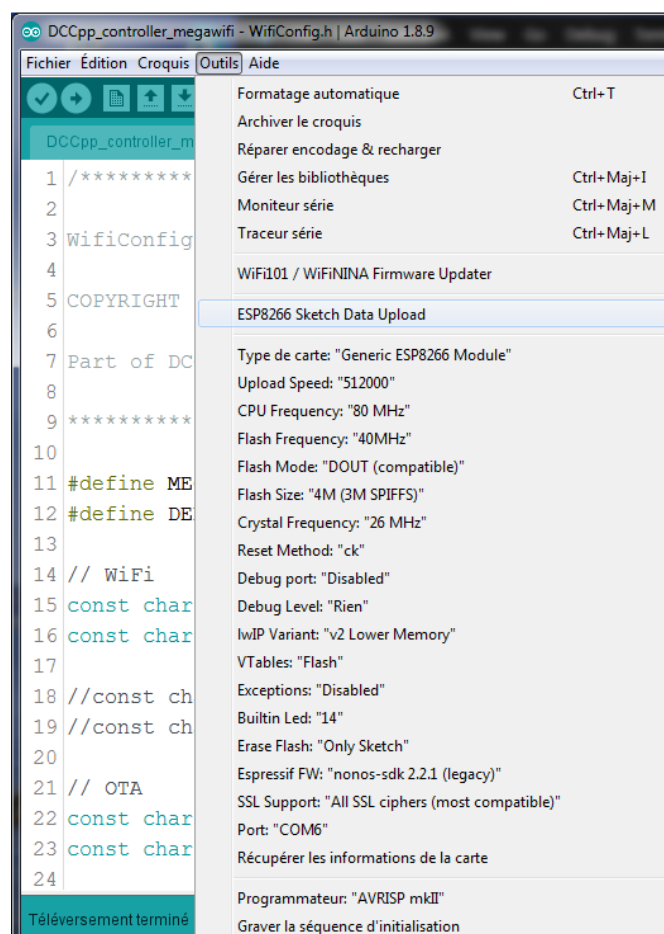
# D1-Mini / MEGA-WiFi installation



- B) Démarrer l'**IDE Arduino** et ouvrir le fichier **DCCpp\_controller\_BTwifi.ino** qui se trouve dans le répertoire éponyme.
- C) Dans l'onglet **Outils - Flash size** sélectionnez **Flash Size: 4M (3M SPIFFS)**.
- D) Avec l'**IDE Arduino** à l'onglet **Outils**, sélectionner la ligne **ESP8266 Sketch Data Upload**. Tous les fichiers de l'application HTML contenu dans le dossier "data" seront téléversés dans la mémoire Flash de la carte. (la liaison série fonctionne à **512000 Bauds**)

**Note:** le projet doit être compilé avec la même valeur de "**Flash Size**" pour la compilation du code et pour le téléchargement des Data afin qu'il retrouve ses Data dans la mémoire Flash.

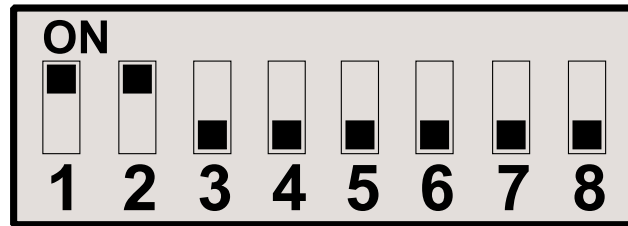
Si l'installation ne fait pas apparaître la ligne "**ESP8266 Sketch Data Upload**", voir : <http://wei48221.blogspot.com/2018/09/how-to-add-sketch-data-upload-function.html> à la section "**For ESP8266**".





# D1-Mini / MEGA-WiFi installation

E) Sur votre carte **Mega WiFi R3**, maintenant configurer ses commutateurs DIP pour une communication entre **MEGA** et **ESP8266** sur RX0/TX0. (**DIP0** et **DIP1** : ON)



---

Ce projet contient 3 logiciels : TCOwifi, WDD6 et DMC pour conduire les trains.

L'installation est finie et vous allez pouvoir utiliser WDD6 avec la centrale DCC.

---

# D1-Mini / MEGA-WiFi installation

## 7) Pour lancer l'application WDD6, TCOWifi et DMC

Connecter vos 2 Boosters **L298N** à votre carte **Mega WiFi R3** et aux rails comme indiqué sur le schéma en utilisant une carte fille de prototypage pour réaliser l'interconnexion.

Brancher le système de mesure de courant de votre choix.

Connecter vos modules **S88-N** via le câble **RJ45** sur les pins indiquées sur le schéma et un connecteur **RJ45**.

Brancher une alimentation **16/19VDC** en plaçant si nécessaire un fusible en série pour alimenter vos Boosters et une alimentation adaptée pour le **MEGA2560**.

Ouvrir un navigateur Internet compatible : **Chrome, Edge, Firefox, Opera, Safari, ...**

Dans la barre d'adresse en haut, tapez l'**IP** de votre module **WiFi** lue sur l'écran **LCD** ou sur le moniteur de l'**IDE Arduino** (ex: 192.168.0.152) ou tapez **dccpps88wifi.local** puis entrée.

Si la connexion est réussie vous voyez apparaître un menu dans lequel vous allez sélectionner : **controller.html** ou **TCOWifi.html** ou **dmc.html**.

Après un temps nécessaire au téléchargement des fichiers depuis la Flash en **WiFi**, vous voyez apparaître un tableau de bord avec un bande supérieure rouge et des onglets horizontaux en bandes bleues. Sur la droite de l'écran se trouvent des boutons pour ouvrir ou refermer les différents onglets qui affichent les fonctions de l'application.

En cas de mauvais chargement de la page sur navigateur, déplacer l'antenne **WiFi** pour trouver un endroit où la communication **WiFi** ne sature pas la partie radio de votre Box. Mais pas trop éloigné de votre Box WiFi.

La première bande supérieure rouge signifie que le **DCC** n'alimente pas les rails. Cette bande passe au vert quand vous cliquez sur le bouton **Power** indiquant que le **DCC** alimente maintenant les rails. Si vous cliquez une deuxième fois elle repassera au rouge et ainsi de suite. En cas de court-circuit, cette bande passera immédiatement au rouge et l'alimentation des rails en **DCC** sera coupée jusqu'à la suppression de la cause du court-circuit. A ce moment vous pourrez cliquer à nouveau sur le bouton **Power** pour rétablir l'alimentation des rails en **DCC**.

### Lecture du courant sur voie principale :

La lecture du courant consommé sur les voies principales s'obtient en cliquant sur le bouton **Main current** situé tout en haut de l'écran. La réponse **DCC** apparaît sur la ligne **Log** en haut et à droite de l'écran à côté de sa valeur en mA. Chaque clic sur ce bouton réactualisera la lecture du courant en affichant une nouvelle valeur.

La ligne **Log** affiche le retour de toutes les commandes exécutées par le **MEGA**. S'il n'a pas d'affichage sur cette ligne, vérifier que le **MEGA** est bien connecté et a "Booté" correctement en appuyant 3 secondes sur son bouton **RESET**.

*Note : toutes les valeurs utilisées sur cet affichage graphique sont en notation décimale.*

**Remarque : au lieu d'utiliser le booster L298N, il est possible d'utiliser le booster LMD18200 en adaptant son câblage voir la page : <http://lormedy.free.fr/WiFiDCCcpp.html>**

---

A tous les modélistes, bon train.

Philippe Chavatte