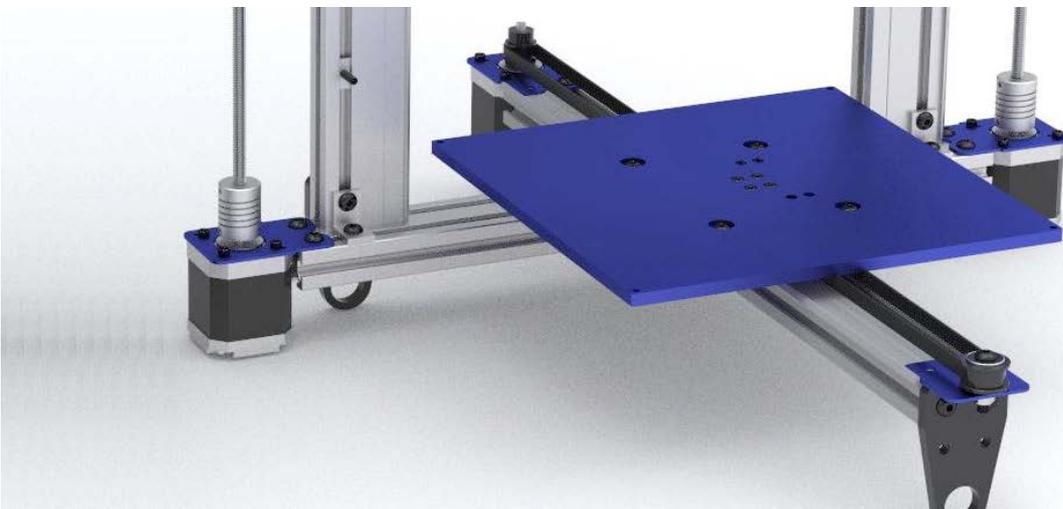
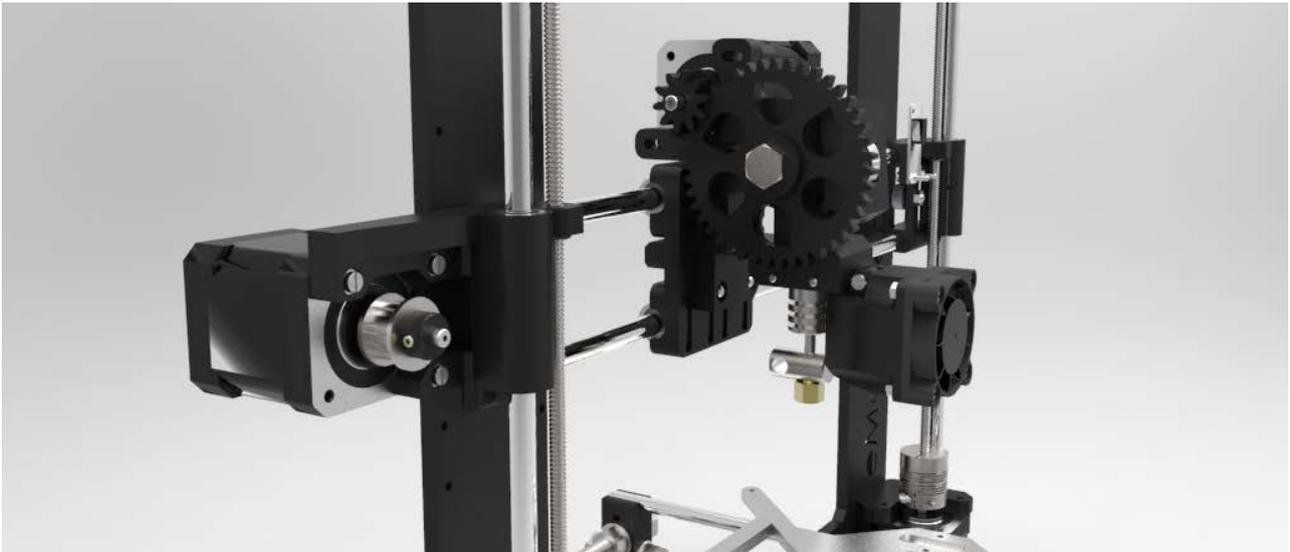




IMPRIMANTES REPRAP



MISE EN ROUTE

Sommaire

I. Installation des drivers et du firmware	3
II. Transfert du firmware via Arduino	5
III. Connexion avec Repetier-Host	6
IV. Réglage du firmware	10
A. Configuration via Repetier	10
B. Configuration manuelle du firmware Marlin	12
1. Fonctionnement des axes / Mechanical Settings	13
2. Capteurs de fin de course / Mechanical Settings	13
V. Vérification de votre configuration et pré-réglages	14
A. Déplacement manuel des axes	14
B. Commande du déplacement des axes	15
C. Inverser la direction de votre moteur	15
D. Vérifier ses capteurs de fin de course « endstops »	16
E. Vérification du fonctionnement des parties chauffantes	17

I. Installation des drivers et du firmware

En premier lieu, après avoir fixé le shield (RAMPS) sur l'Atmega2560 (Arduino), connectez l'alimentation 12V, puis l'USB.

Installer le logiciel Arduino qui se trouve dans votre CD. Vérifier que la carte Arduino est reconnue par votre PC. Pour cela, faites le raccourci « Windows + Pause », puis « Gestionnaire de périphériques » dans la colonne de gauche. La carte Arduino Mega 2560 doit apparaître dans la liste des périphériques.

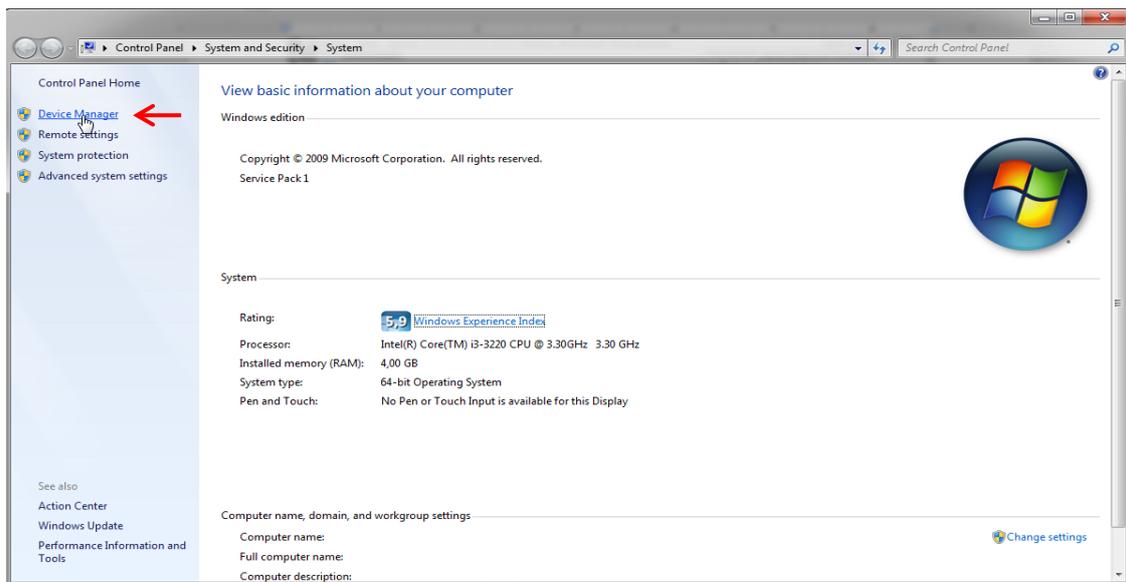


Figure 1 : Panneau "Système" de Windows

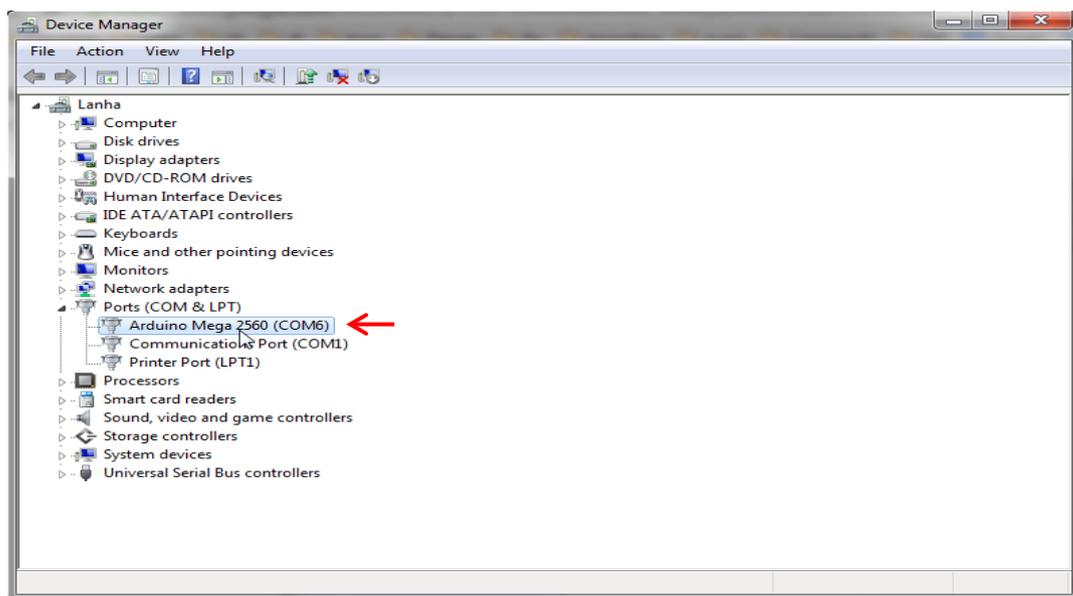


Figure 2 : Gestionnaire de périphériques

Si elle n'est pas reconnue, installer les drivers manuellement en faisant un clic-droit dessus. Le pilote de contrôle de la carte Arduino se trouve dans le répertoire « drivers » de votre logiciel Arduino installé sur votre PC.

Relevez le port COM correspondant à votre carte (Arduino Atmega 2560). Pour cela, aller dans le « Gestionnaire de périphériques » comme précédemment. La carte Arduino Mega 2560 doit apparaître ainsi que le numéro du port COM correspondant. Lancez le logiciel [Arduino](#).

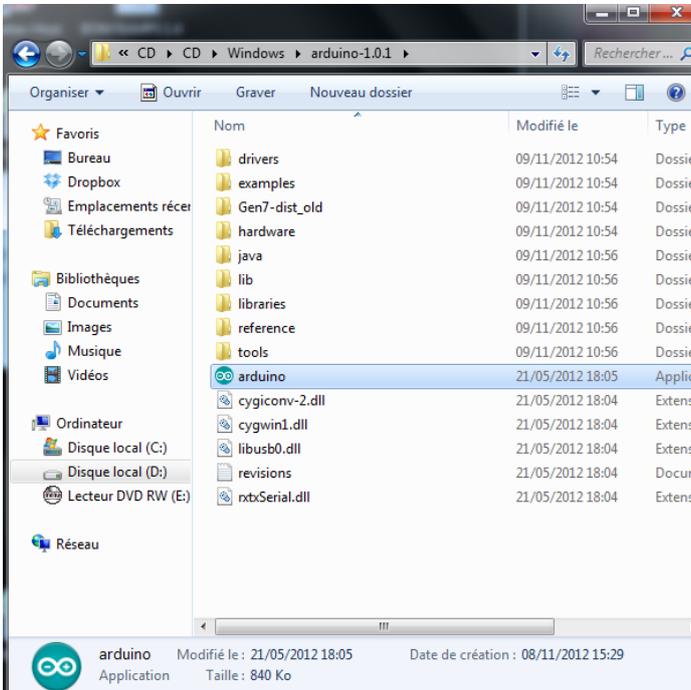


Figure 3 : Lancement du logiciel Arduino

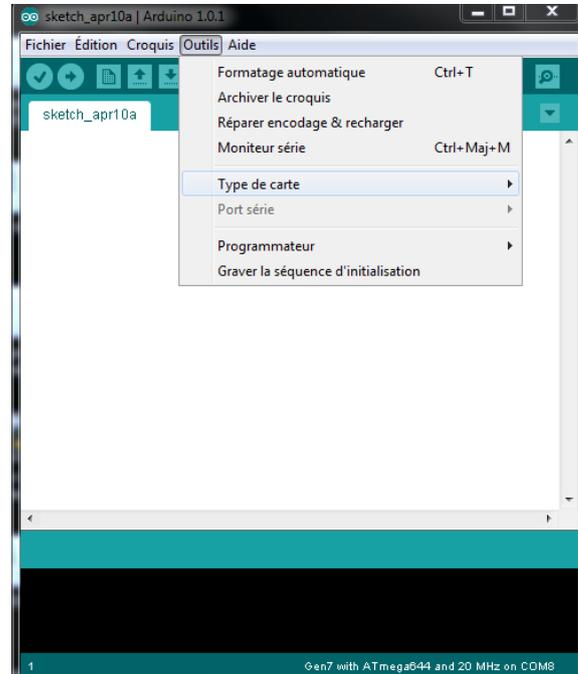


Figure 4 : Sélection de la carte

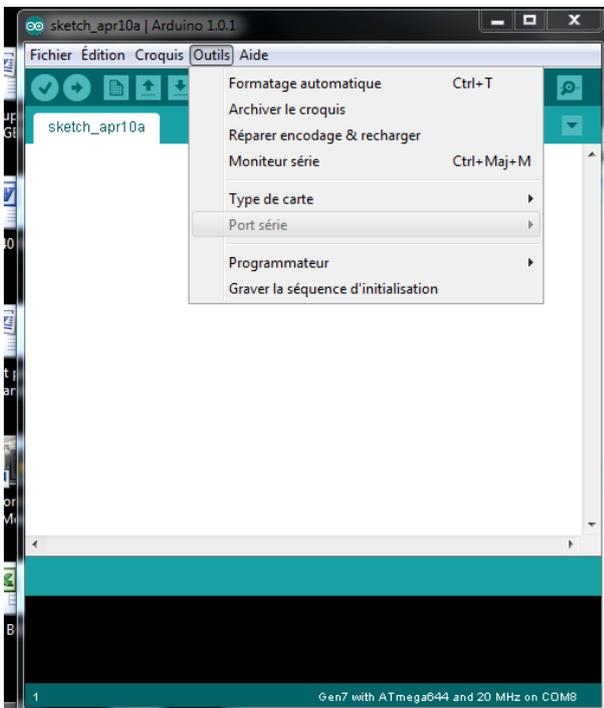


Figure 5 : Sélection du port COM

Une fois le logiciel Arduino lancé, sélectionnez votre carte : Cliquez sur Outils/Type de carte (figure 2).

- Si vous avez un kit précision et/ou une Ramps 1.4 sélectionnez Arduino Mega 2560.

Sélectionnez ensuite le port COM correspondant (relevé précédemment).

II. Transfert du firmware via Arduino

Ouvrez ensuite avec Arduino (fichier/ouvrir) le dossier firmware du CD dans lequel il vous faut sélectionner selon le modèle de votre imprimante (Prusa Air 2, ORD Bot ou Prusa i3 Rework) :

- Marlin.pde (Arduino avant 1.0) ou Marlin.ino (Arduino après 1.0)

Cliquez ensuite sur téléverser.

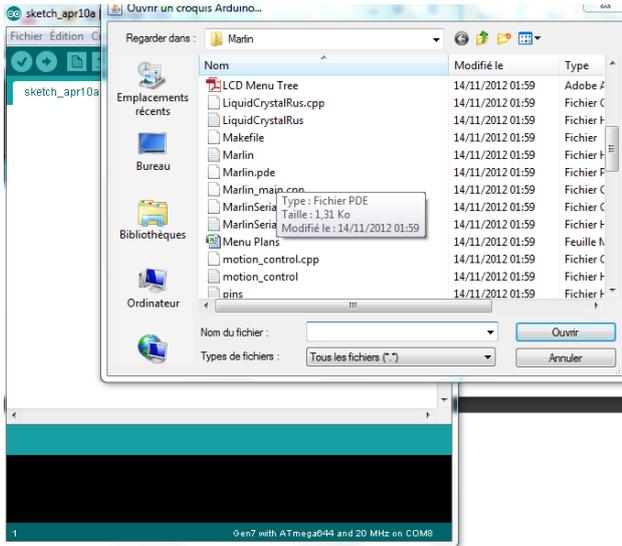


Figure 6 : Ouverture du firmware sous Arduino

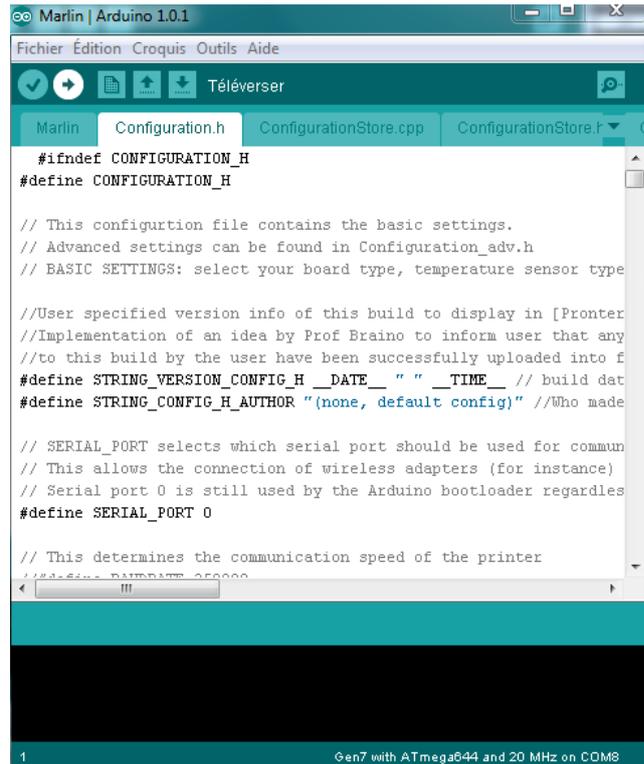


Figure 7 : Aperçu du fichier Configuration.h du firmware

Le transfert est terminé lorsque « Completed » est affiché dans la barre d'information inférieure du logiciel.

III. Connexion avec Repetier-Host

Avant d'attaquer ce tutorial il vous faudra avoir rempli les prérequis suivants :

- parties électroniques et mécaniques prêtes à fonctionner (voir notice de montage) ;
- carte détectée et pilote de l'Arduino reconnu par votre ordinateur (voir guide Arduino) ;
- firmware Repetier-firmware uploadé via Arduino.

Tout d'abord nous allons régler la partie communication de notre logiciel hôte Repetier-Host.

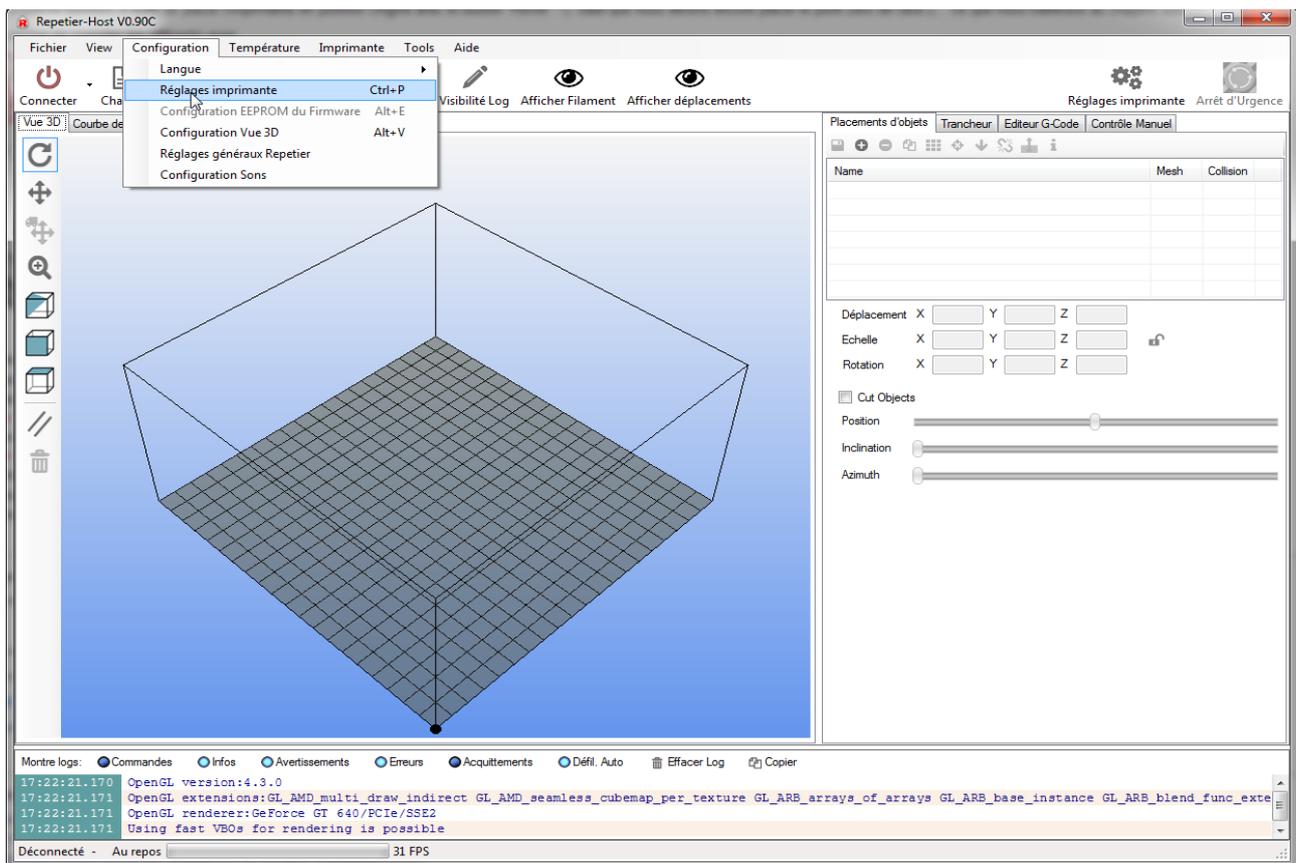


Figure 8 : Interface du logiciel Repetier-Host et ouverture des réglages de l'imprimante

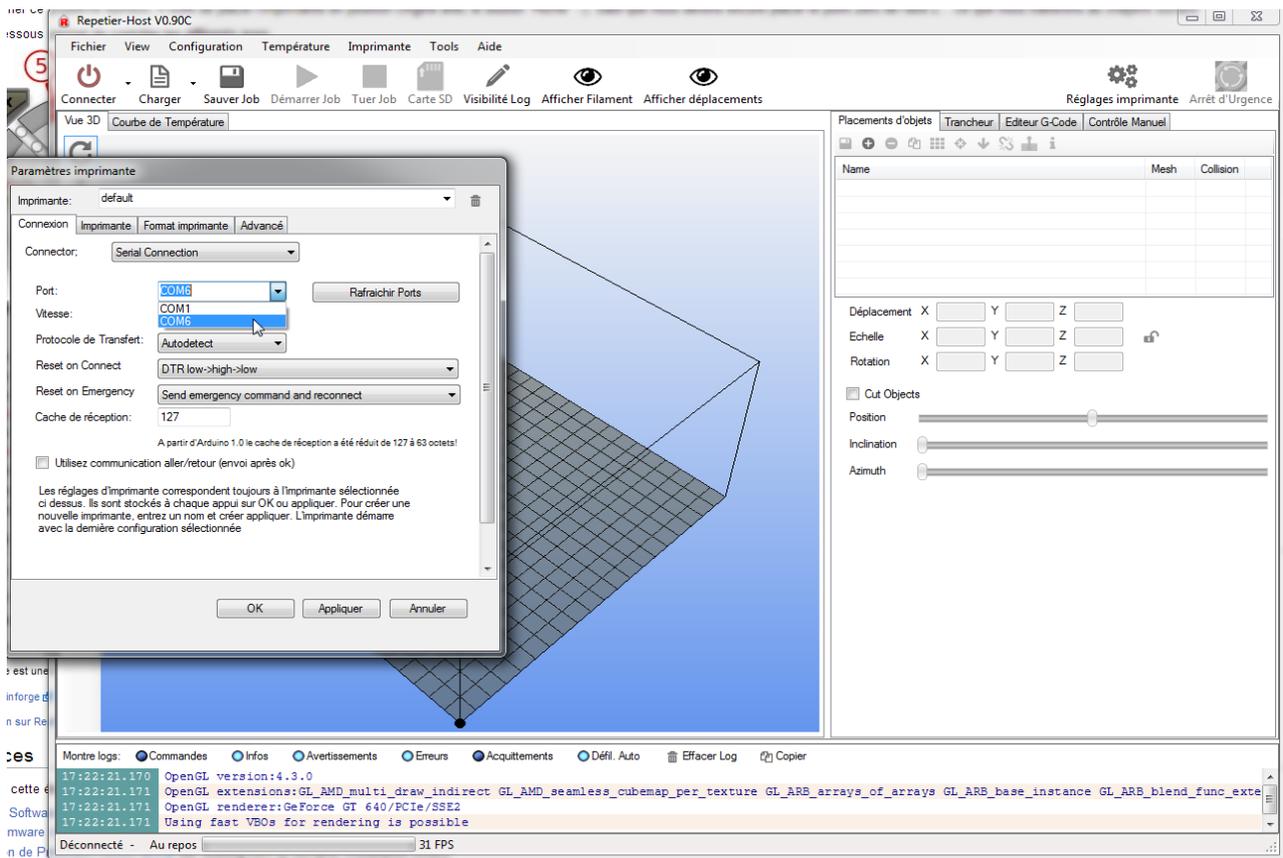


Figure 9 : Choix du port COM dans les paramètres de l'imprimante

Nous changeons le port de communication correspondant à votre Reprap. Le port COM a été relevé dans la première étape de la mise en route de l'imprimante (voir partie d'installation des drivers).

Nous allons maintenant régler la vitesse de communication de notre Arduino, sous Repetier-firmware nous sommes à 115200 BAUD par défaut (sous Marlin nous pourrions être à 250000 BAUD).

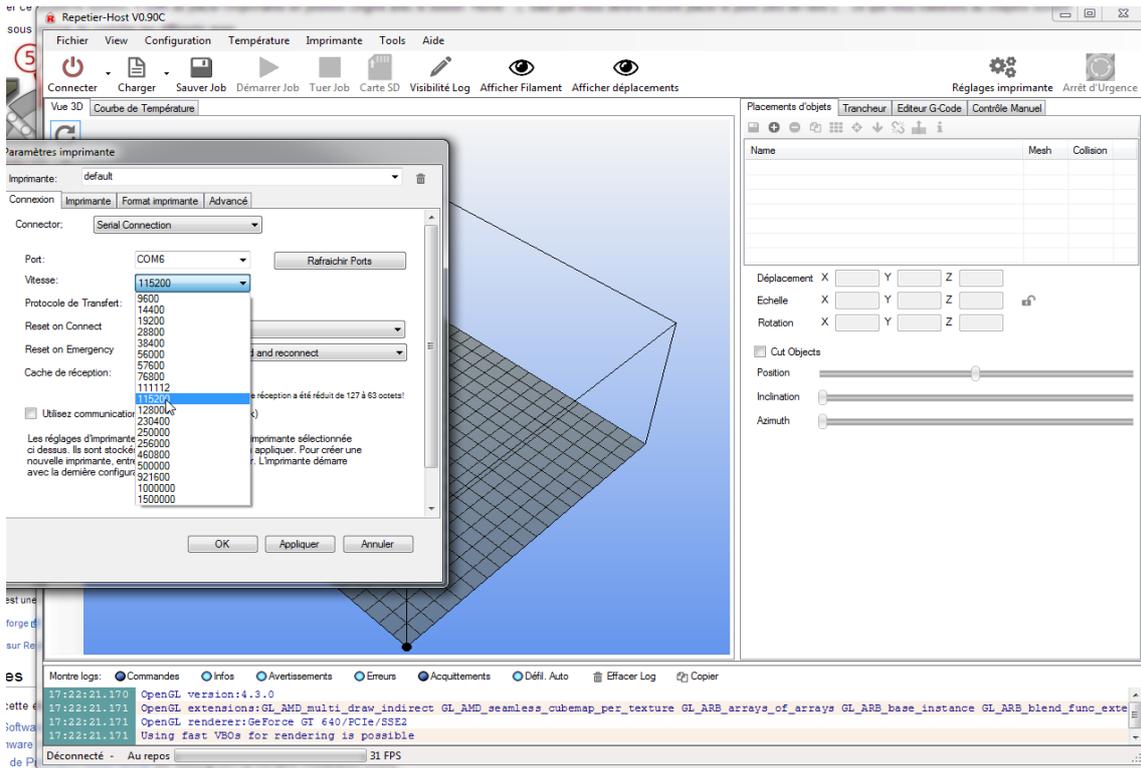


Figure 10 : Réglage de la vitesse de communication

Nous enclenchons la communication ping-pong qui permet la vérification et la correction des paquets d'instructions envoyés à notre Arduino.

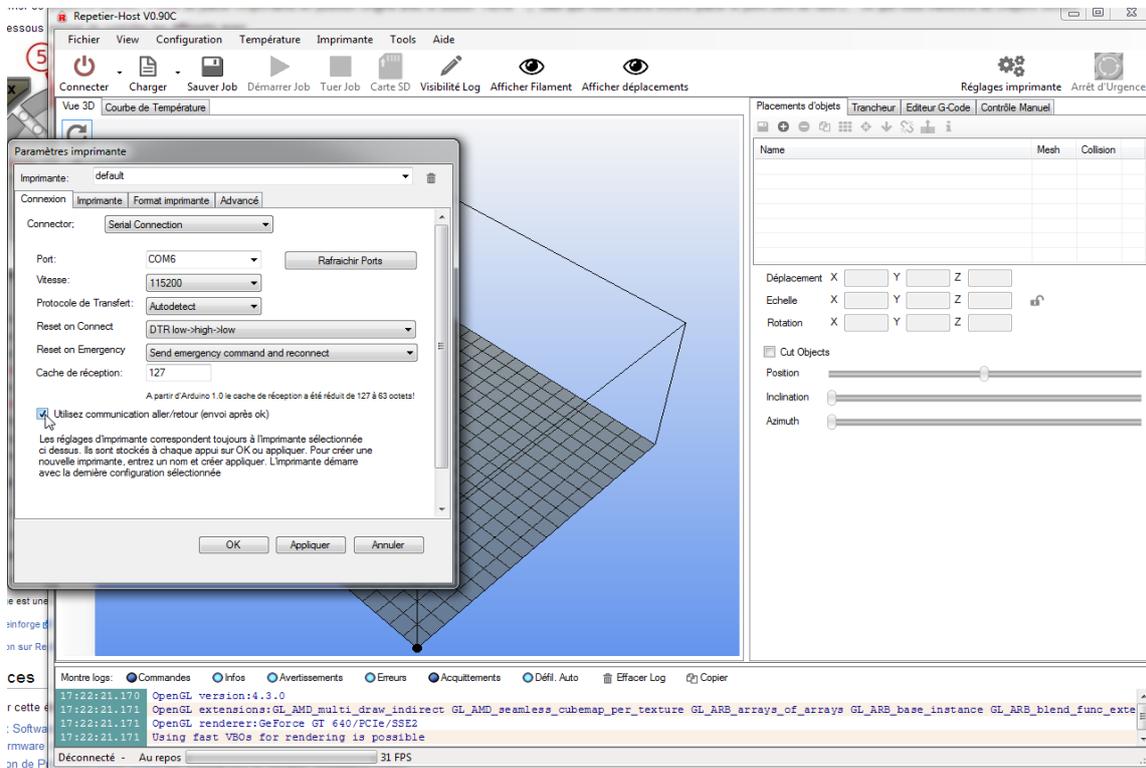


Figure 11 : Activation de la communication aller-retour

Vous pouvez donc vous connecter maintenant à votre imprimante 3D à l'aide du bouton connecter (choisir l'imprimante à l'aide de la flèche si nécessaire). Il faut savoir que le logiciel Repetier-Host permet de communiquer avec plusieurs imprimantes (une fenêtre par imprimante).

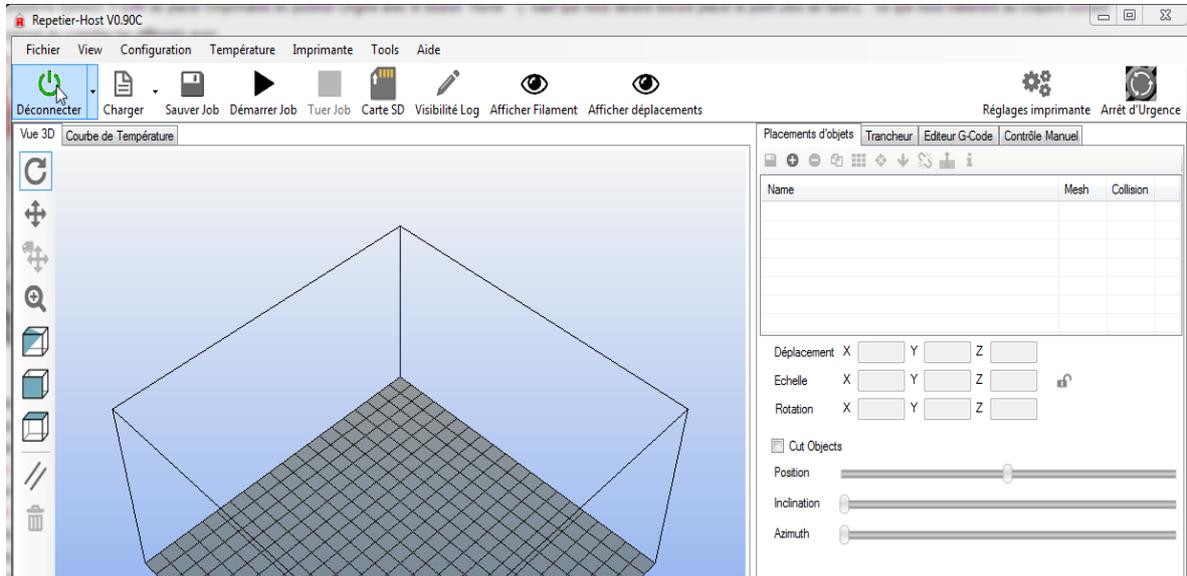
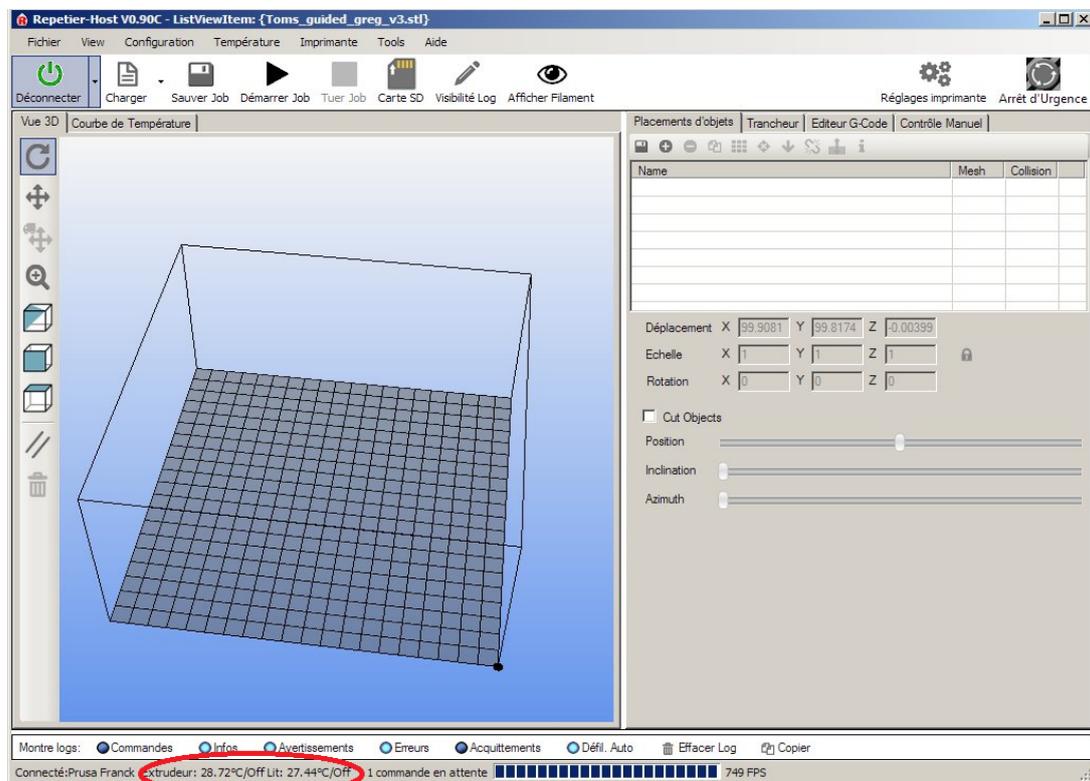


Figure 12 : Connexion à l'imprimante

Félicitation, vous communiquez maintenant avec votre imprimante !



Vous devez normalement lire les températures de votre extrudeur et lit chauffant mesurées par les thermistances qui devraient avoisiner à +/- 5°C la température ambiante. Si ce n'est pas le cas, alors vous avez un défaut de câblage ou de configuration de votre de votre thermistance dans le firmware. Si vous voyez 0°C votre thermistance n'est pas connectée.

IV. Réglage du firmware

Nous allons maintenant procéder aux réglages du micrologiciel (firmware) de votre Arduino. En effet, il n'est pas nécessaire de remplir les valeurs de pas par mm, vitesse et accélération dans le firmware avant de lancer le logiciel Repetier Host. Repetier a un accès direct à la mémoire de l'Arduino qui permet de modifier à la volée les variables stockées dans notre Arduino (et cela même durant l'impression, ce qui est bien pratique pour l'étape de calibration).

Néanmoins, les notions indispensables à la compréhension des principaux réglages mécaniques du firmware seront détaillées en prévision de l'étape suivante de ce manuel qui est la vérification du fonctionnement de l'imprimante.

A. Configuration via Repetier

Pour se faire ouvrez les réglages EEPROM sous Repetier (sous le firmware Marlin, l'interface EEPROM ne fonctionne que si on l'active au préalable en annulant le commentaire EEPROM_SETTINGS et EEPROM_CHITCHAT dans le firmware pour permettre l'édition de l'EEPROM via Repetier-host).

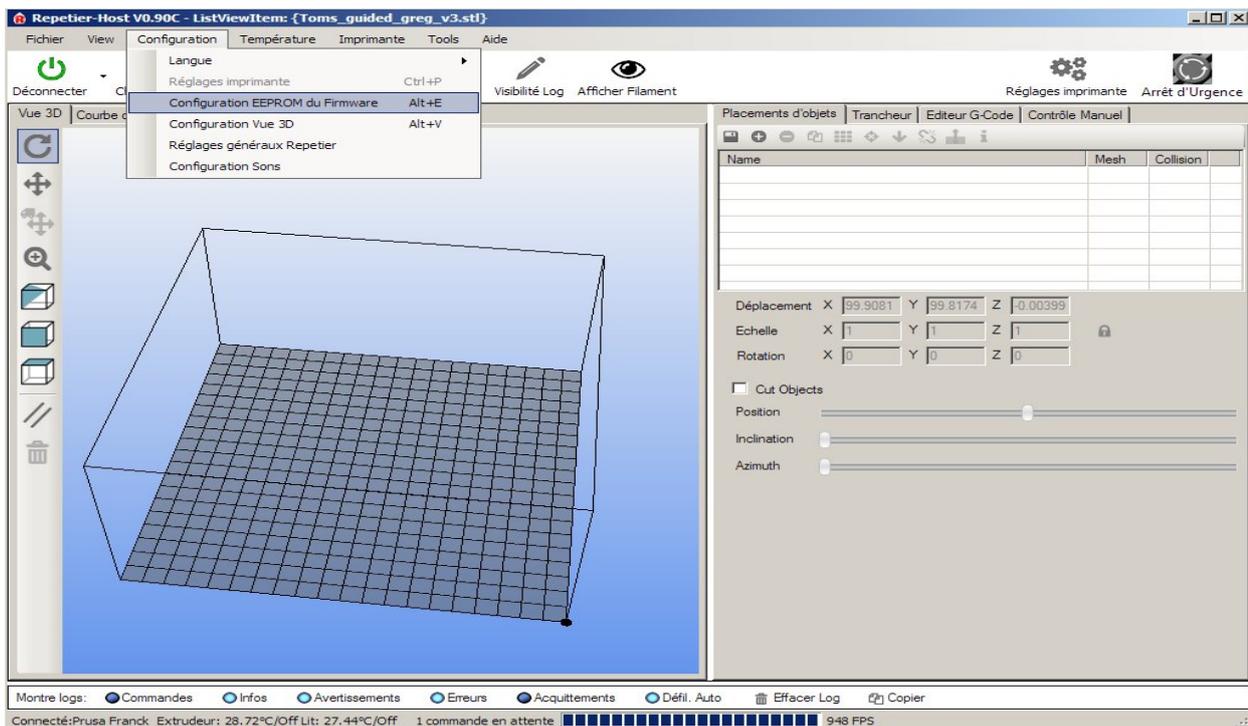
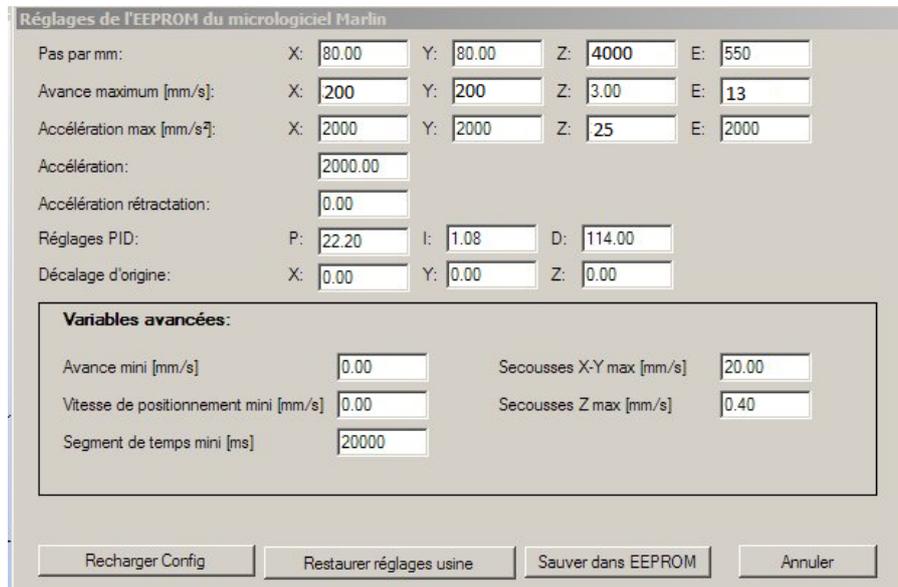


Figure 13 : Configuration du firmware sous Repetier-Host

Modifions le nombre de pas par mm dans le cas où nous utilisons une courroie GT2 sur une poulie de 20 dents avec une valeur de microstepping de 16 (tous les « jumpers » posés sur le RAMPS).

Dans le cas où nous utilisons des poulies GT2 16 dents, le nombre de pas par mm est de 100 (le nombre de dent est inscrit sur la poulie.)

Si vous avez choisi le **firmware Marlin**, vous trouverez la configuration suivante :



Réglages de l'EEPROM du micrologiciel Marlin

Pas par mm: X: 80.00 Y: 80.00 Z: 4000 E: 550

Avance maximum [mm/s]: X: 200 Y: 200 Z: 3.00 E: 13

Accélération max [mm/s²]: X: 2000 Y: 2000 Z: 25 E: 2000

Accélération: 2000.00

Accélération rétractation: 0.00

Réglages PID: P: 22.20 I: 1.08 D: 114.00

Décalage d'origine: X: 0.00 Y: 0.00 Z: 0.00

Variables avancées:

Avance mini [mm/s] 0.00 Secousses X-Y max [mm/s] 20.00

Vitesse de positionnement mini [mm/s] 0.00 Secousses Z max [mm/s] 0.40

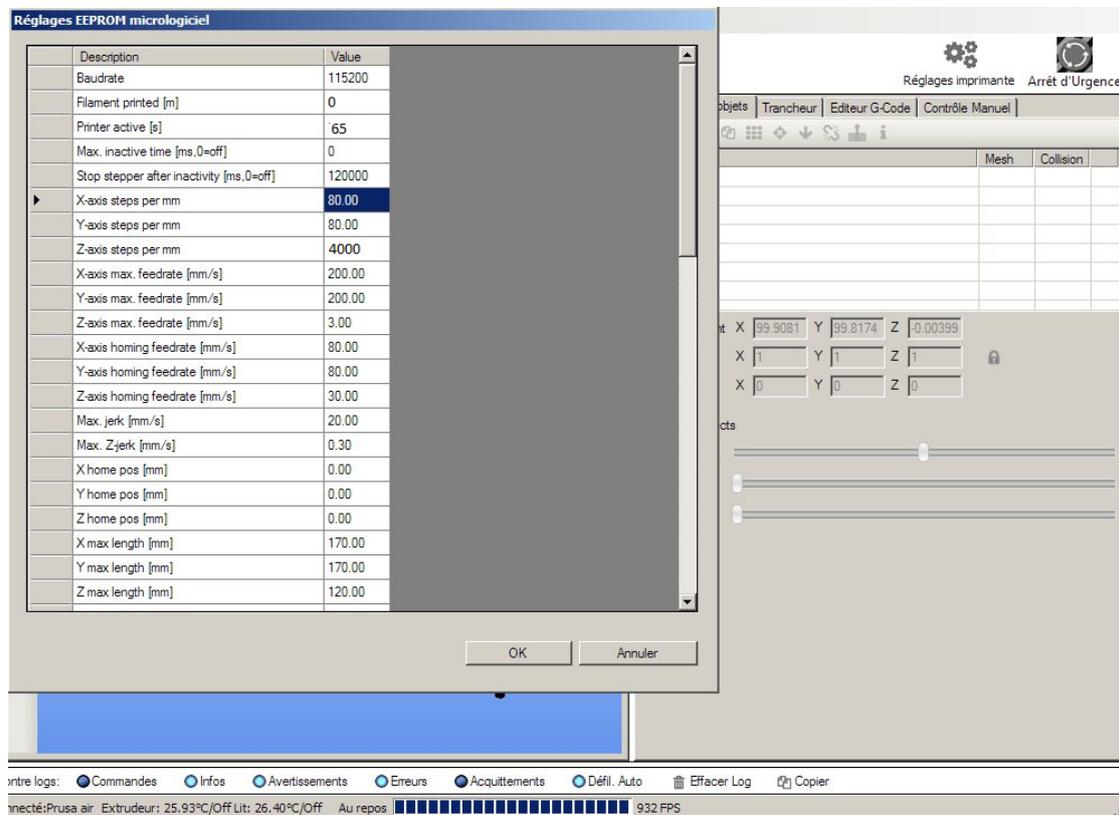
Segment de temps mini [ms] 20000

Recharger Config Restaurer réglages usine Sauver dans EEPROM Annuler

Figure 14 : Réglages EEPROM dans le cas d'un firmware Marlin

Le calcul des paramètres est donné par défaut pour une carte Ramps avec tous les cavaliers en place sous les pilotes stepsticks des moteurs pas à pas.

Dans le cas du choix d'un **firmware Repetier** :



Réglages EEPROM micrologiciel

Description	Value
Baudrate	115200
Filament printed [m]	0
Printer active [s]	65
Max. inactive time [ms,0=off]	0
Stop stepper after inactivity [ms,0=off]	120000
X-axis steps per mm	80.00
Y-axis steps per mm	80.00
Z-axis steps per mm	4000
X-axis max. feedrate [mm/s]	200.00
Y-axis max. feedrate [mm/s]	200.00
Z-axis max. feedrate [mm/s]	3.00
X-axis homing feedrate [mm/s]	80.00
Y-axis homing feedrate [mm/s]	80.00
Z-axis homing feedrate [mm/s]	30.00
Max. jerk [mm/s]	20.00
Max. Zjerk [mm/s]	0.30
X home pos [mm]	0.00
Y home pos [mm]	0.00
Z home pos [mm]	0.00
X max length [mm]	170.00
Y max length [mm]	170.00
Z max length [mm]	120.00

OK Annuler

Réglages imprimante Arrêt d'Urgence

objets Trancheur Editeur G-Code Contrôle Manuel

Mesh Collision

Position: X 99.9081 Y 99.8174 Z -0.00399

Axis: X 1 Y 1 Z 1

Axis: X 0 Y 0 Z 0

cts

entre logs: Commandes Infos Avertissements Erreurs Acquittements Défil. Auto Effacer Log Copier

connecté:Prusa air Extrudeur: 25.93°C/Off Lit: 26.40°C/Off Au repos 932 FPS

Figure 15 : Réglages EEPROM dans le cas d'un firmware Repetier

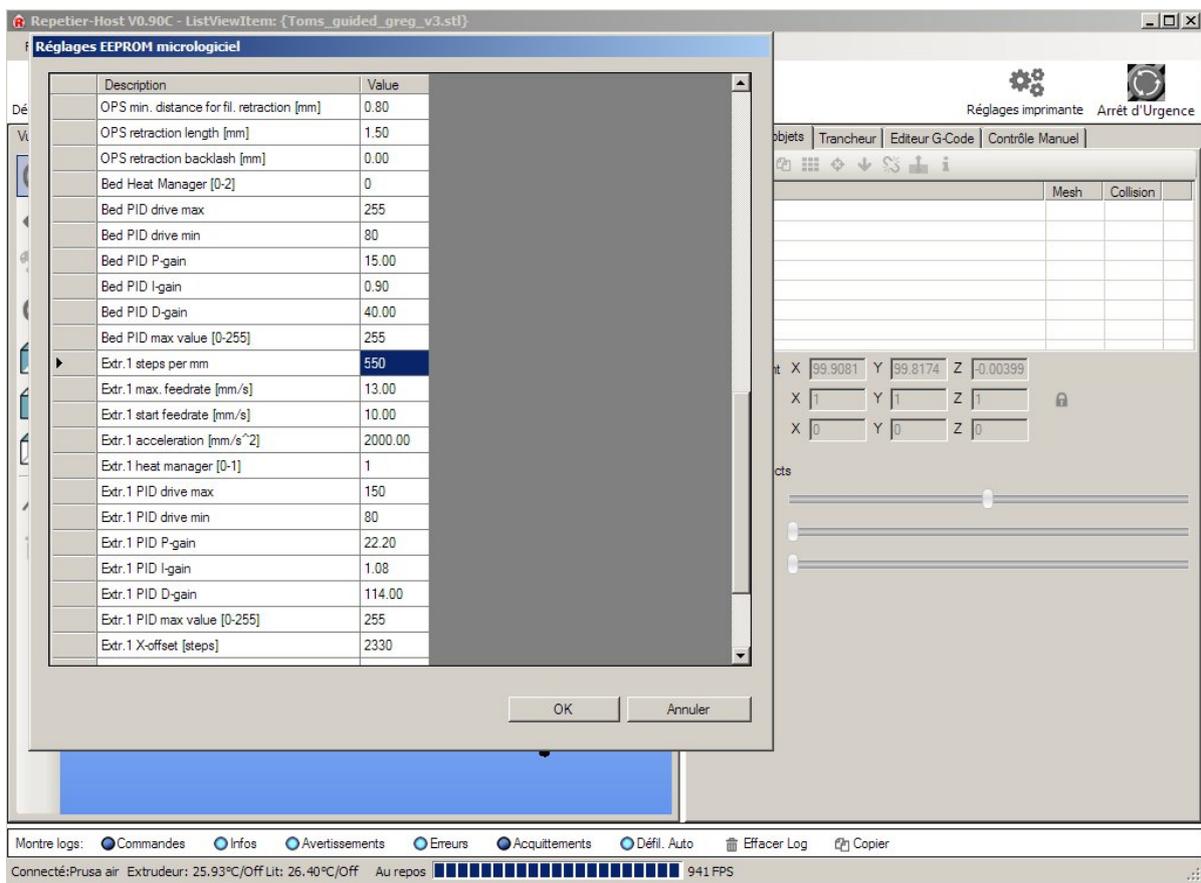


Figure 16 : Réglages EEPROM pour l'extrudeur

L'extruder sera mis entre 500 et 800 steps par mm dans le cas normal d'utilisation d'un extruder Greg Wade avec une démultiplication 9/47 utilisé ici chez eMotion-Tech.

B. Configuration manuelle du firmware Marlin

Le firmware Marlin est constitué de différents fichiers textes possédant l'extension .h. Ce firmware est ouvrable et modifiable à l'aide du logiciel Arduino comme nous l'avons vu précédemment. Celui-ci est normalement pré-configuré pour être directement utilisable mais il se peut que vous ayez à configurer certains paramètres du firmware pour optimiser le fonctionnement de votre imprimante.

Pour cela, le seul fichier à modifier est nommé : **Configuration.h**

Pour modifier le firmware, enregistrer le dossier « Marlin » en local, effectuer les modifications, sauvegarder le firmware puis le re-téléverser.

[Attention, les notions que nous allons aborder ne sont utiles que si vous rencontrez des problèmes dans la partie suivante : Vérification de votre configuration et pré-réglages.](#)

Le fichier Configuration.h possède plusieurs grandes parties définissant le fonctionnement de l'imprimante. Nous allons seulement aborder ce qui concerne les capteurs de fin de course et le fonctionnement des axes moteurs et extrudeur.

1. Fonctionnement des axes / Mechanical Settings

Ce réglage permet de changer le sens de rotation des moteurs (déplacement à l'opposé du sens voulu ou moteur de l'extrudeur fonctionnant dans le mauvais sens).

Ce paragraphe est accessible à la ligne **301** (numéro visible en bas à gauche du logiciel Arduino). Il suffit de changer le réglage booléen : par exemple « true » par « false » pour le moteur de l'extrudeur (ligne INVERT_E0_DIR).

```
#define INVERT_X_DIR true // for Mendel set to false, for Orca set to true
#define INVERT_Y_DIR true // for Mendel set to true, for Orca set to false
#define INVERT_Z_DIR true // for Mendel set to false, for Orca set to true
#define INVERT_E0_DIR true // for direct drive extruder v9 set to true, for geared extruder set to false
#define INVERT_E1_DIR false // for direct drive extruder v9 set to true, for geared extruder set to false
#define INVERT_E2_DIR false // for direct drive extruder v9 set to true, for geared extruder set to false
```

2. Capteurs de fin de course / Mechanical Settings

Ce réglage permet de définir la position d'origine d'un axe. Celui-ci est par exemple nécessaire lorsque le plateau s'éloigne du capteur de fin de course alors qu'on demande la prise d'origine de l'axe Y.

Ce paragraphe est accessible à la ligne **308** (numéro visible en bas à gauche du logiciel Arduino). Il suffit de changer la valeur « -1 » en « 1 ».

Attention, effectuer cette opération qu'après avoir vérifié que vos branchements moteurs sont corrects.

```
// ENDSTOP SETTINGS:
// Sets direction of endstops when homing; 1=MAX, -1=MIN
#define X_HOME_DIR -1
#define Y_HOME_DIR -1
#define Z_HOME_DIR -1

#define min_software_endstops true // If true, axis won't move to coordinates less than HOME_POS.
#define max_software_endstops true // If true, axis won't move to coordinates greater than the defined lengths below.
// Travel limits after homing
#define X_MAX_POS 180
#define X_MIN_POS 0
#define Y_MAX_POS 180
#define Y_MIN_POS 0
#define Z_MAX_POS 180
#define Z_MIN_POS 0
```

Le paragraphe suivant permet de définir les dimensions utiles d'impression. Modifier les bornes maximum (ici 180 mm) pour pouvoir balayer la surface entière du plateau. Ces valeurs doivent correspondre aux valeurs configurables dans le logiciel Repetier (configuration de l'imprimante).

V. Vérification de votre configuration et préréglages

Nous allons maintenant procéder à la première vérification matérielle de votre machine. Pour se faire, il est nécessaire d'avoir validé la partie précédente.

A. Déplacement manuel des axes

Première étape :

- poussez vos chariots X et Y au centre de leurs axes, si vous sentez une résistance - **Ne pas forcer !** -, il suffit de stopper le blocage des moteurs :

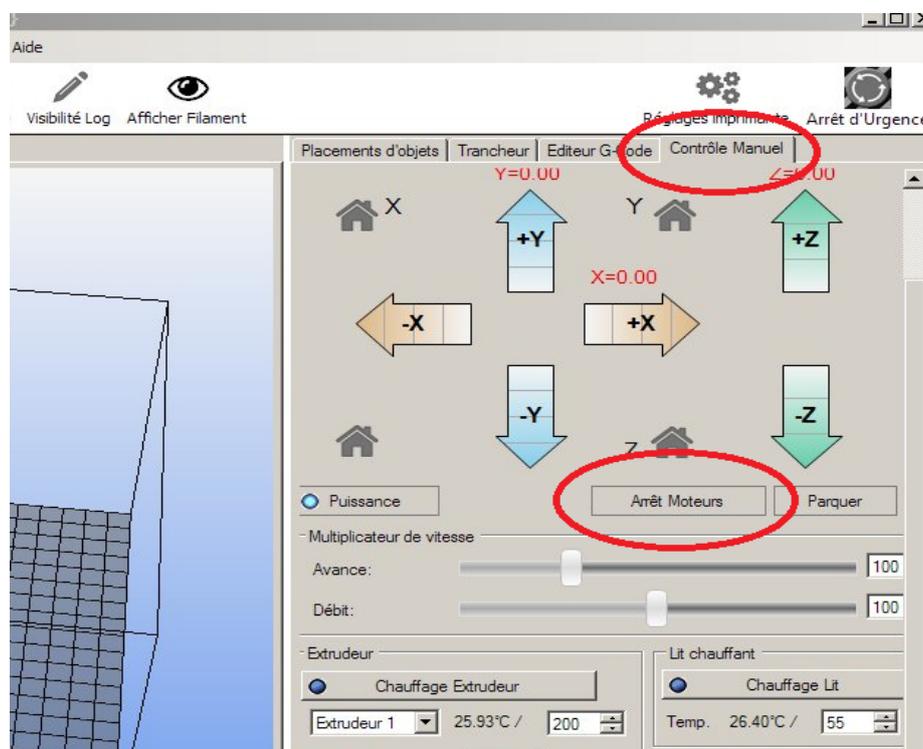


Figure 17 : Arrêt des moteurs pour déplacement manuel

Pour l'axe Z essayez de monter la buse d'extrusion hors de portée du plateau (au moins 20 mm) en tournant les deux coupleurs simultanément et de rester aussi horizontal que possible pour ne pas nuire à la structure plastique.

Vous êtes maintenant prêt à faire vos premiers mouvements sans risques.

B. Commande du déplacement des axes

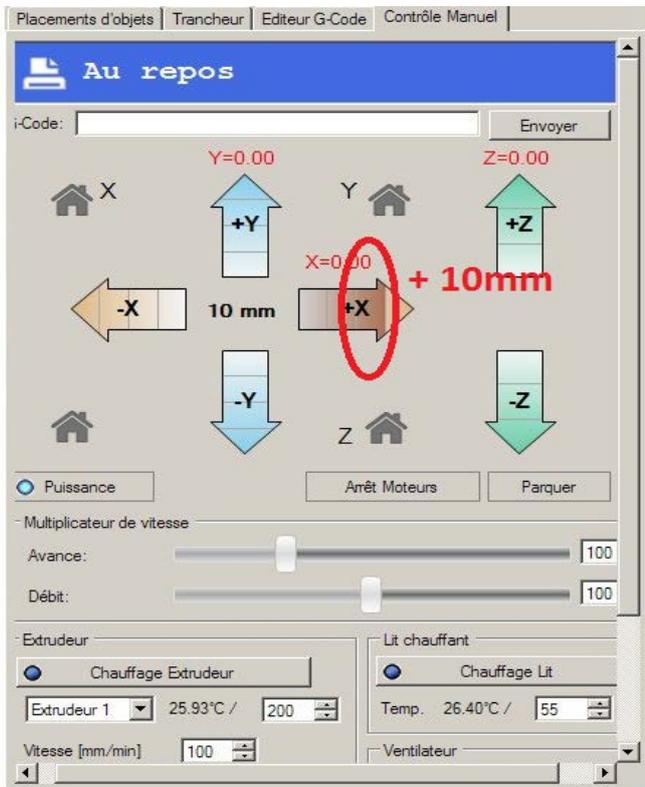


Figure 18 : Déplacement de l'axe X

Commander le déplacement de l'axe X de 10 mm dans le sens X+. S'il ne se passe rien, vérifiez le câblage de l'alimentation du moteur de l'axe X.

Le déplacement doit donc être de 10 mm à l'opposé de votre endstop X. S'il se rapproche de votre endstop, vous avez alors une inversion de direction à corriger.

C. Inverser la direction de votre moteur

Il existe 2 différents moyens d'inverser le sens de rotation de votre moteur :

- Mécaniquement, en effet le branchement de votre moteur est constitué dans l'ordre du contact 1 et 2 pour la première bobine et du contact 3 et 4 de votre seconde bobine. **Intervertir les fils d'une bobine revient à inverser le sens du moteur.** *Petite astuce : suivant comment est câblé votre moteur, il est possible qu'inverser simplement la prise de votre moteur sur votre Ramps change le sens de rotation de votre moteur (cela dépend du câblage interne du moteur en fonction du fournisseur).*
- Logiciellement, il suffit de se rendre dans le fichier « configuration.h » du firmware Marlin ou Repetier et de modifier le « #define INVERT_X_DIR » pour inverser sa valeur, « false » à la place de « true » ou inversement (voir partie ci-avant).

Procédez de la même façon pour les axes Y et Z.

D. Vérifier ses capteurs de fin de course « endstops »

La première étape est de vérifier si l'endstop est fonctionnel. Pour se faire, vérifiez que votre **la LED rouge est allumée** lors d'un contact (liaison USB nécessaire au minimum).

Maintenant posez votre doigt sur le bouton reset de votre Ramps, (**sans appuyer dessus !**), **appuyez sur l'origine X (maison) sans lâcher le bouton reset (il vous permettra de tout arrêter si vous avez un problème sans rien endommager)** et vérifiez que le chariot X s'arrête après avoir touché le capteur de fin de course. Si vous voyez/sentez un problème, n'hésitez pas à appuyer sur le bouton reset de la carte.

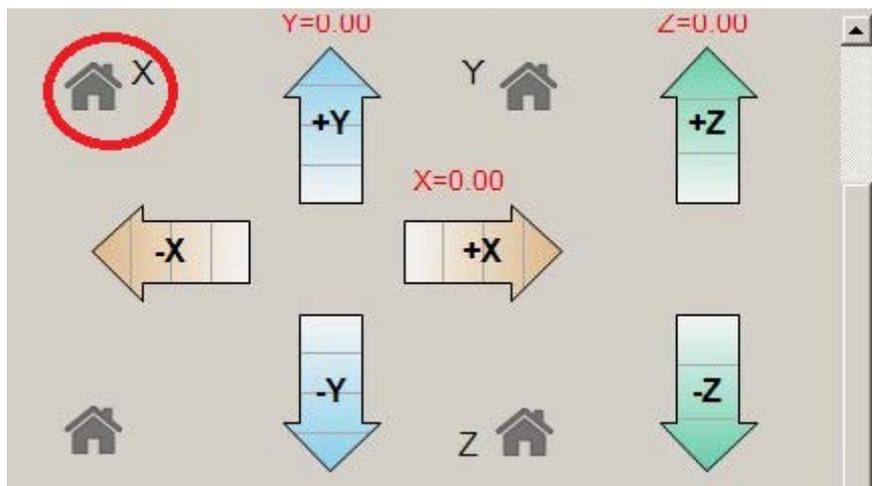


Figure 19 : Position d'origine X

Si l'axe se dirige à l'opposé du capteur de fin de course, effectuer la modification des constantes booléennes dans le firmware Marlin concernant les capteurs de fin de course (voir partie ci-avant).

Faites les mêmes opérations pour l'axe Y.

Montez l'endstop Z pour être sûr que la buse ne touche pas le lit chauffant durant l'opération, puis effectuez la même opération.

E. Vérification du fonctionnement des parties chauffantes

Procédez au lancement de la chauffe pour l'extrudeur :

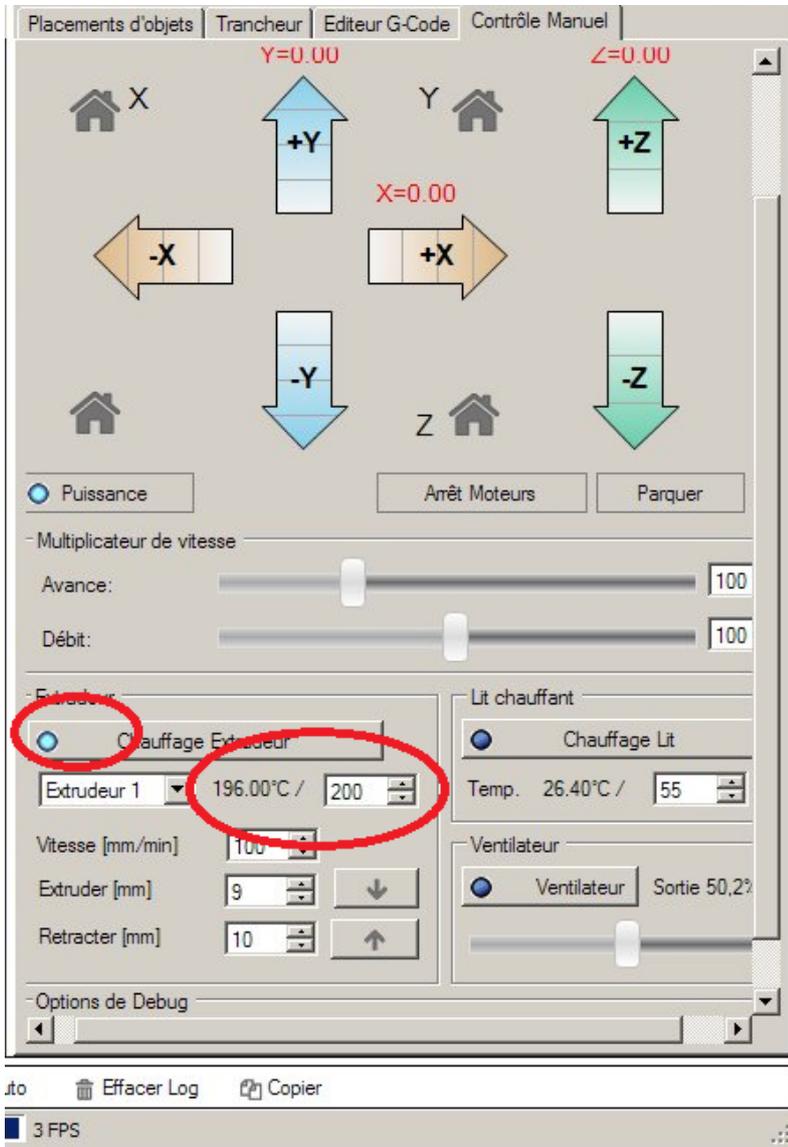


Figure 20 : Chauffe de l'extrudeur

Vérifiez maintenant si la température de l'extruder augmente et non celle du lit chauffant.

Si jamais rien ne change, contrôlez le voyant en bas de la carte Ramps D10. Si celui-ci est allumé, vérifiez votre câblage. Votre résistance chauffante ou cartouche de chauffe est mal connectée (pensez à utiliser un multimètre pour mesurer la résistance aux bornes de la partie chauffante, vous devriez avoisiner une valeur de 5 Ohms).

Faire de même pour le lit chauffant.

Félicitation, votre machine est réglée et prête à l'emploi !