



P5

Imprimante 3D à Chocolat

Projet IMA4
2015/2016

Hugo Vandebunder
Sylvain Verdonck

Tuteurs :
Emmanuelle Pichonat
Alexandre Boé
Xavier Redon
Thomas Vantroys

SOMMAIRE

Remerciements.....	3
Introduction.....	4
Solutions envisagées & retenues.....	5
Extrusion.....	5
Imprimante.....	5
Contrôle de la température.....	5
Développement technique.....	7
Liste du matériel utilisé.....	7
Contrôle de la température.....	7
Capteur de température.....	7
Chauffage du réservoir.....	7
Contrôle de la pression.....	8
Capteur de pression.....	8
Imprimante 3D.....	9
Impression & achat des pièces.....	9
Assemblage, configuration et calibration.....	9
Programmes.....	10
Arduino dédié à la régulation.....	10
Arduino dédié à l'impression.....	11
Communication entre les deux systèmes.....	12
Tests.....	13
Améliorations.....	14
Isolation.....	14
Matériaux.....	14
Gestion de pression.....	14
Capteur de pression.....	14
Commande Compresseur.....	14
Refroidissement.....	14
Arduino.....	15
Imprimante.....	15
Conclusion.....	16
Annexes.....	17
Raccord tuyau de cuivre.....	17
Autres modifications du firmware.....	18
Diagramme Labview.....	19

REMERCIEMENTS

Nous tenons à remercier nos tuteurs de leur soutien continu et les réponses qu'ils ont su nous apporter lorsque nous étions bloqués.

Et nous remercions chaleureusement Thierry Flament et Antoine Urquizar qui nous ont éclairés respectivement sur le chocolat et de nombreuses questions d'électroniques, et sur les imprimantes 3D.

INTRODUCTION

Le domaine des imprimantes 3D est en pleine expansion, avec un marché qui explose, une communauté d'adepte qui grandit et de plus en plus d'applications variées, allant du spatial au culinaire.

Le but de ce projet est de construire une imprimante 3D à bas coût permettant d'utiliser des matériaux comestibles, le chocolat comme exemple, qui sera notre premier objectif.

On peut d'ores et déjà retrouver des machines imprimant des matériaux comestibles dans le commerce, telles que la Foodini ou encore la Choc Edge. La première permettant d'imprimer tout type de "matériaux" liquide et même comestibles, alors que la seconde se concentre sur l'impression du chocolat. Néanmoins ces deux imprimantes nécessitent de remplir une cartouche d'un matériau déjà liquide. C'est une limite que nous essayerons de dépasser au cours de ce projet. Notre but est de concevoir un système capable de chauffer et faire fondre le chocolat (ou le matériau qu'on veut utiliser quel qu'il soit), pour qu'il puisse être extrudé par la suite.

SOLUTIONS ENVISAGEES & RETENUES

Pour commencer, nous divisons la réalisation de ce projet en différentes thématiques dont nous devons envisager les solutions. Nous achèterons ensuite le matériel nécessaire à la réalisation de la solution technique qui a notre préférence. Nous nous contentons d'explorer une solution, puisque le temps alloué pour ce projet ne nous permet pas d'en faire plus sans bâcler le travail.

EXTRUSION

Le chocolat, ou tout autre matériau culinaire, présente des caractéristiques qui font que les méthodes d'extrusion d'une imprimante classique ne sont pas applicables. De manière générale, le problème est le suivant : il nous faut un système pour "pousser" le matériau en dehors de notre réservoir. De plus, si l'on souhaite l'utilisation de matériaux variés, il faut que la poussée soit réglable.

Nous réfléchissons donc à un système de vis sans fin, de piston avec un vérin ou encore un système à air comprimé. Les deux premiers rendent le système moins accessible, et donc plus difficile à démonter et nettoyer entre les utilisations. Il y a un risque d'encrassement qui remet en cause le fonctionnement normal. C'est pourquoi nous avons opté pour la dernière option. Bien qu'elle nous ajoute de grosses contraintes d'étanchéité, nous estimons cette solution plus viable pour la durabilité du système. C'est aussi la méthode la plus propre, puisque de l'air poussera le matériau.

IMPRIMANTE

La réalisation du projet nécessite une imprimante 3D fonctionnelle, et bien que notre sujet propose la construction d'une imprimante 3D à partir de pièces de récupérations, une telle entreprise constituerait un projet en soit. Néanmoins, acheter une imprimante de toutes pièces nous était impossible, puisque le but du projet est de proposer une solution à bas coûts.

C'est pourquoi nous avons saisi l'opportunité d'utiliser un kit acheté par le Fabricarium de l'école qui n'avait pas été exploité. Il comprenait toutes les pièces électroniques d'une imprimante modèle Prusa i3, les parties mécaniques restant à acheter ce qui limitait le coût et le temps de construction de manière non négligeable.

CONTROLE DE LA TEMPERATURE

Nous ne souhaitons pas nous limiter à l'impression d'un chocolat préalablement fondu ou encore aux matériaux liquides à températures ambiantes, c'est pourquoi un contrôle de la température est nécessaire. Pour réguler un système, nous avons besoin de mesurer la valeur qu'on veut modifier, et pouvoir agir dessus. Nous avons envisagé d'utiliser des modules Peltier ou du fil résistif pour le chauffage, et une sonde de température type PT100 pour une mesure de la température.

Les modules Peltier et sondes PT100 impliquant un coût relativement élevé, nous nous décidons pour le fil résistif et un composant électronique capteur de température.

Les solutions que nous avons retenues nous permettent d'établir le schéma de principe suivant :

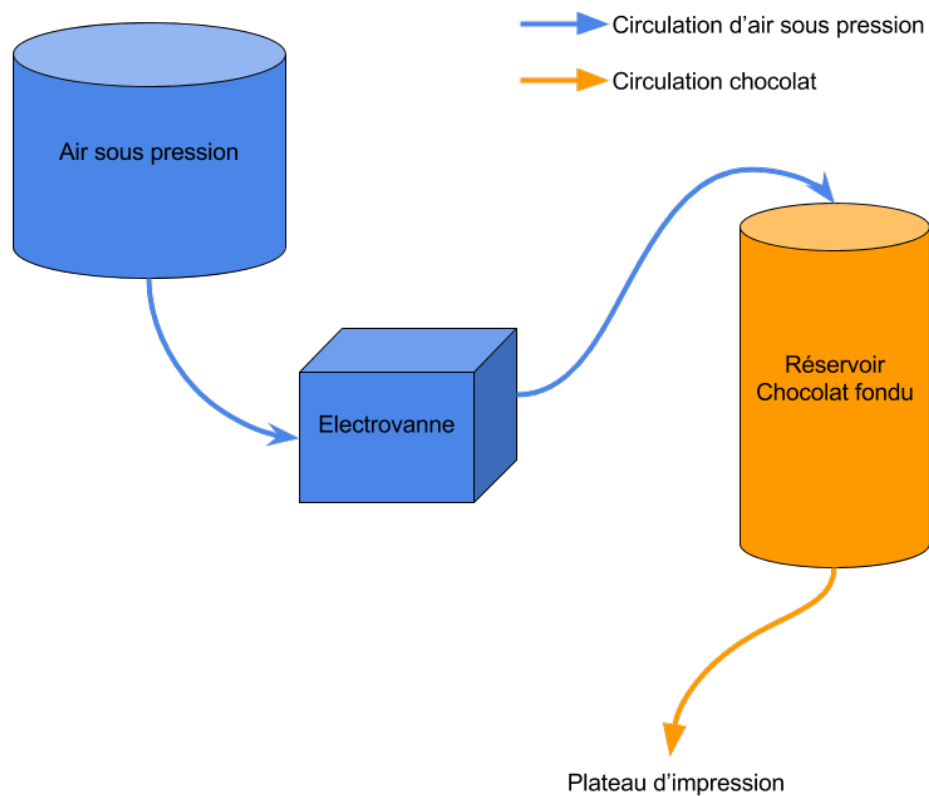


FIGURE 1 - SCHEMA DE PRINCIPE

Nous avons un réservoir qui contient de l'air comprimé. A l'aide de l'électrovanne, on injecte de la pression dans le réservoir à chocolat qui est chauffé par le fil résistif. C'est en régulant l'ouverture de l'électrovanne, que l'on contrôle la vitesse d'extrusion, le chocolat sera poussé à l'extérieur du réservoir.

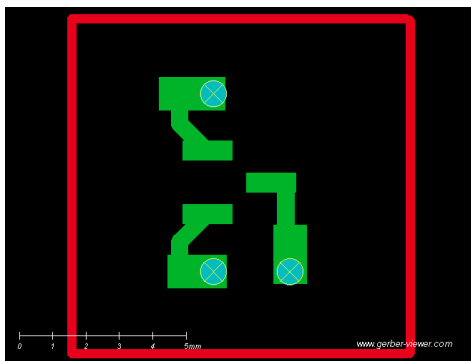
DEVELOPPEMENT TECHNIQUE

LISTE DU MATERIEL UTILISE

Compresseur	Résistances x2
Tuyau pneumatique	Relais électrique
Raccords en té	Capteur température
Bouteille de soda (réservoir à air comprimé)	(Capteur de pression)
Électrovanne	Transistors x2
Tuyau en cuivre	Papier sulfurisé
Fil résistif	
Mousse isolante	
Pièces 3D imprimées	
Arduino	

CONTROLE DE LA TEMPERATURE

CAPTEUR DE TEMPERATURE



Pour utiliser le capteur de température acheté, nous avons conçu une carte électronique qui nous permet de relier plus facilement les pins et de l'enficher dans la pièce d'extrusion pour avoir la mesure la plus précise possible. L'idée est de faire passer les fils par l'arrière de la carte, pour que le capteur soit au plus proche du produit dans notre réservoir.

FIGURE 2 - PCB CAPTEUR DE TEMPÉRATURE

CHAUFFAGE DU RESERVOIR

On utilise un tube en cuivre comme réservoir, autour duquel on enroule le fil résistif. Pour limiter le rayonnement du fil, et concentrer la chaleur vers le tube, on entoure ce montage d'une mousse isolante récupérée. Le tuyau étant en cuivre, conducteur d'électricité, il provoque un court-circuit avec le fil résistif qui l'empêche de chauffer. Pour isoler électriquement ces deux éléments, tout en conservant la conduction thermique, nous utilisons du papier sulfurisé. Pour une solution plus permanente, nous proposons d'enduire le cuivre de vernis ou encore d'utiliser une feuille de mica.

On fixe ensuite ce tuyau à l'imprimante via une pièce imprimée. Cette pièce prend le tuyau dans une pince serrée par une vis.

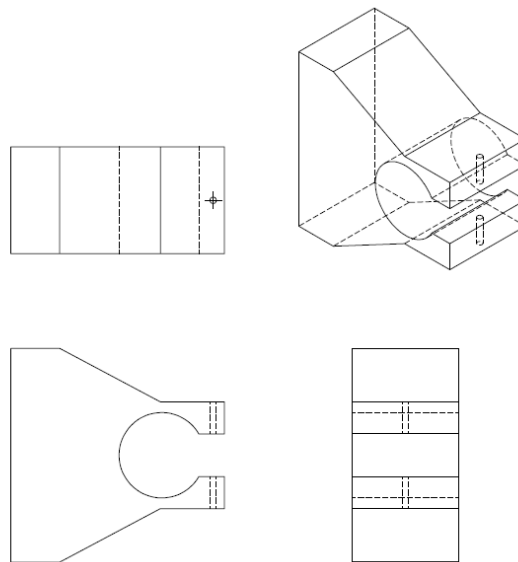


FIGURE 3 - FIXATION DE L'EXTRUDEUR

CONTROLE DE LA PRESSION

CAPTEUR DE PRESSION

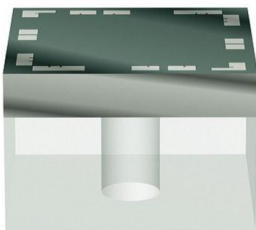


FIGURE 4 - CAPTEUR DE PRESSION

Le capteur de pression acheté s'est avéré de dimensions trop petites pour les technologies de soudure dont nous disposons. Notre tuteur M. Boé a néanmoins essayé de le souder sous microscope à la colle argent, mais cela s'est avéré trop fragile et le capteur n'a pas pu être utilisé.

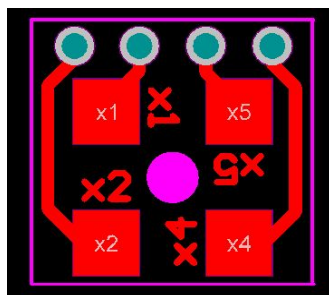


FIGURE 5 - PCB CAPTEUR DE PRESSION

Nous avons prévu une carte pour utiliser ce capteur, qui permettait de tendre des fils entre le dessus du composant et la carte. Le capteur étant collé sur la carte. Son entrée d'air, sous le composant, était alimentée par dessous.

IMPRIMANTE 3D

IMPRESSION & ACHAT DES PIÈCES

Après un inventaire complet des pièces dont nous disposions, nous avons commencé à imprimer les pièces disponibles gratuitement sur internet. Ces pièces constituent une bonne partie de la mécanique de l'imprimante, et c'est un moyen efficace de diminuer les coûts de construction de notre imprimante. Il a néanmoins été nécessaire d'acheter quelques compléments, telles que les tiges métallique, vis et écrous. Enfin, le châssis et le plateau ont été réalisés en bois, découpé à la découpeuse laser du Fabricarium, une fois de plus pour réduire le coût de notre machine.

ASSEMBLAGE, CONFIGURATION ET CALIBRATION.

Après avoir répertorié nos pièces, nous avons entamé le montage de l'imprimante en suivant le guide de montage et une vidéo. Nous n'avons pas monté l'extrudeur sur l'imprimante, puisqu'il nous est inutile dans notre projet.

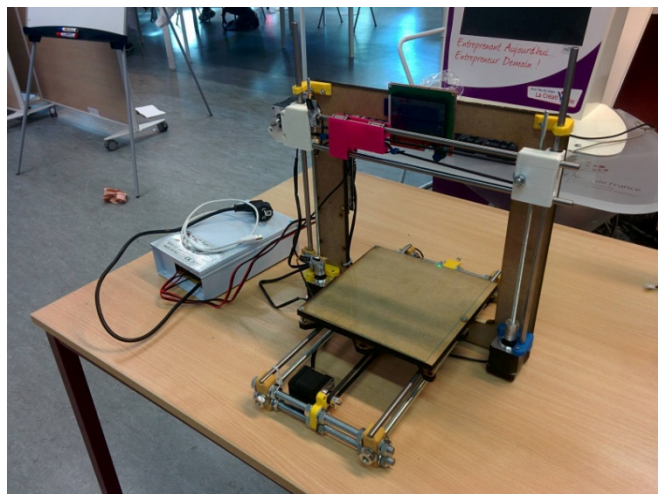


FIGURE 6 - NOTRE IMPRIMANTE 3D

Ensuite, une configuration du firmware interne est nécessaire. Disponible en OpenSource, le Marlin s'est imposé sur le marché de l'imprimante 3D, et est facilement adaptable à de nombreux types d'imprimantes. La configuration des parties mécaniques est la seule nécessaire, n'ayant ni extrudeur, ni plateau chauffant, la gestion de température est totalement omise dans notre système. Cette configuration se développe en définissant les positions maximales sur chaque axe, la vitesse et le nombre de pas des moteurs. Il était toutefois nécessaire de vérifier le sens des moteurs, et de changer leur valeur dans le firmware ou d'inverser les branchements, dans le mauvais cas.

Pour vérifier la calibration et le bon fonctionnement de l'imprimante, on attache de façon rudimentaire un crayon de bois à la partie mobile, on fixe une feuille de papier sur le plateau et on adapte la hauteur. On lance alors l'impression d'un cube. Résultat concluant : un cube se dessine sur la feuille de papier. Pour vérifier la bonne calibration, on imprime le même cube en 3D et on compare les dimensions des deux cubes. L'expérience est concluante, donc notre imprimante est fonctionnelle.

PROGRAMMES

ARDUINO DEDIE A LA REGULATION

PARTIE ELECTRONIQUE

Sur un Arduino Uno que l'on contrôle grâce à LabView, on régule l'impression via la température et la pression. Un transistor nous permet d'alimenter ou non l'électrovanne à 12V, ce qui la met en position ouverte.

Pour le fil résistif, nous devons combiner un transistor et un relais, puisque nous faisons passer 5 Ampères pour le faire chauffer. Le transistor alimente le relais qui permet l'alimentation du fil.

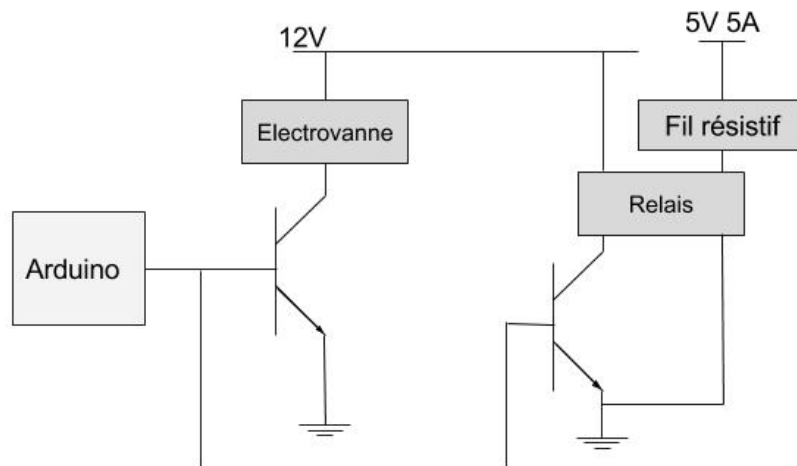


FIGURE 7 - SCHEMA DU MONTAGE D'ALIMENTATION

PARTIE LABVIEW

On utilise le logiciel Labview, largement utilisé en entreprise pour ce type de régulation. On le couple avec un Arduino puisque les composants habituels qui vont avec ce logiciel sont excessivement chers. Le programme Labview se compose de deux parties : une face avant qui fait l'interface avec l'utilisateur, et un diagramme qui forme le programme.

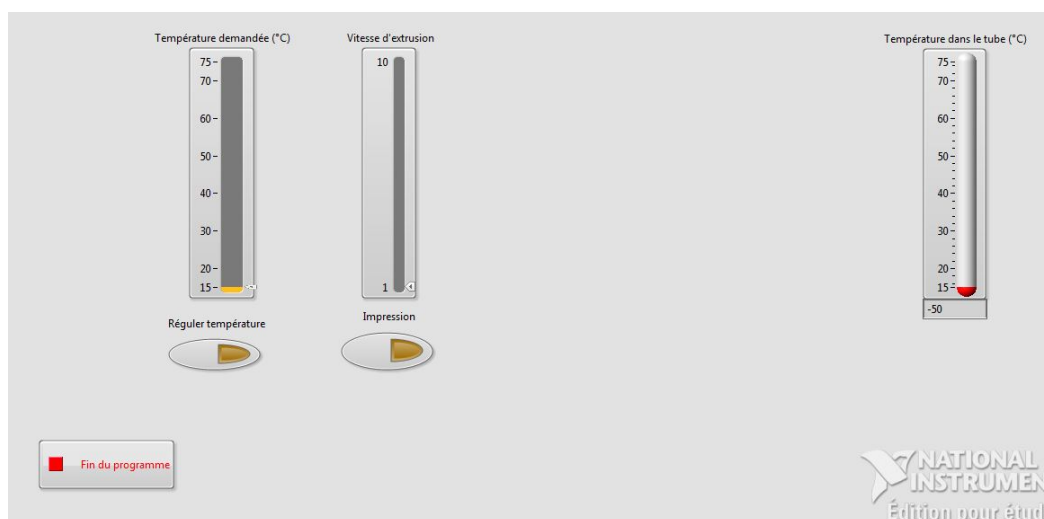


FIGURE 8 - FACE AVANT LABVIEW

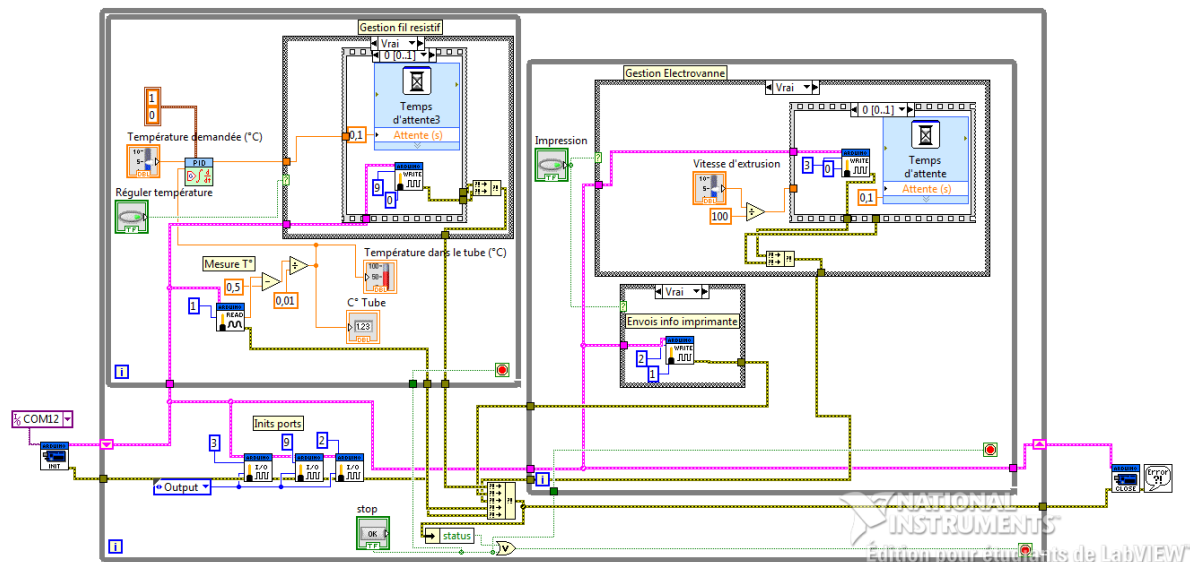


FIGURE 9 - DIAGRAMME LABVIEW

Le programme a un fonctionnement simple :

On initialise l'Arduino par le bloc tout à gauche, et on entre ensuite dans une boucle qui ne s'arrête qu'à la pression du bouton « Fin du programme ».

On initialise ensuite les ports qu'on va utiliser, et on entre alors dans deux sous-programmes différents :

Le programme de gestion de la température. On trouve ici la lecture de la valeur du port analogique connecté au capteur de pression. Par un calcul on retrouve la valeur de la température. Cette valeur est comparée via un PID à la valeur demandée par l'utilisateur. Le PID permet d'envoyer une valeur de 0 à 10 en fonction de l'approche de la valeur demandée par la valeur mesurée. Ce qui nous permet de ralentir progressivement la chauffe. Ensuite, si l'utilisateur a allumé la régulation de température, la valeur fournie par le PID permet de laisser chauffer le fil plus ou moins longtemps par rapport à son temps de repos. C'est un fonctionnement proche d'un système de PWM.

Le programme de gestion de la pression fonctionne sur un principe similaire. Si l'utilisateur a activé l'impression, on envoie un 1 à l'Arduino s'occupant de cette partie, et on alimente également l'électrovanne. On reste sur un principe de PWM, où l'électrovanne reste au repos un certain temps, et l'utilisateur choisi de la laisser ouverte plus ou moins longtemps.

En appuyant sur le bouton stop, on ferme proprement la communication avec l'Arduino via le bloc "close" à droite.

ARDUINO DEDIE A L'IMPRESSION

Fourni avec le kit, un Arduino Méga pour contrôler l'impression. On installe dessus le firmware Marlin, disponible en OpenSource. C'est ce programme que l'on modifie pour ajuster l'impression classique à celle du chocolat. Ces changements doivent être activés par l'utilisateur avant le lancement dans l'impression, pour cela nous avons ajouté l'option « enable/disable chocolate » dans le menu « prepare », grâce aux manipulations suivantes.

```
static void lcd_prepare_menu() {
  START_MENU();
  MENU_ITEM(back, MSG_MAIN, lcd_main_menu);
  #if defined( SDSUPPORT ) && defined( MENU_ADDAUTOSTART )
    MENU_ITEM(function, MSG_AUTOSTART, lcd_autostart_sd);
  #endif
  MENU_ITEM(gcode, MSG_DISABLE_STEPPERS, PSTR("M84"));
  MENU_ITEM(gcode, MSG_AUTO_HOME, PSTR("G28"));
  MENU_ITEM(function, MSG_SET_HOME_OFFSETS, lcd_set_home_offsets);
  //MENU_ITEM(gcode, MSG_SET_ORIGIN, PSTR("G92 X0 Y0 Z0"));
  #ifndef CHOCO
    if (chocolate_mode)
      MENU_ITEM(function, MSG_DIS_CHOCO, lcd_set_choco);
    else MENU_ITEM(function, MSG_EN_CHOCO, lcd_set_choco);
  #endif
}
```

FIGURE 10 - LCD_PREPARE_MENU

```
#ifndef CHOCO
int chocolate_mode = 0; //default : chocolate printed is disabled

void lcd_set_choco(){ //Changes chocolate_mode when called
  if (chocolate_mode)
    {chocolate_mode = 0; SERIAL_ECHOLNPGM("chocolate_mode=0\n");}
  else {chocolate_mode = 1; SERIAL_ECHOLNPGM("chocolate_mode=1\n");}
}
#endif
```

FIGURE 11 - LCD_SET_CHOCO

Dans le fichier `ultralcd.cpp` et plus précisément, dans la fonction `lcd_prepare_menu`, si le mode chocolat est activé, on affiche un message pour le désactiver, et vice-versa. Dans les deux cas, ce choix renvoie vers la même fonction : `lcd_set_choco`. Nous avons défini les messages `MSG_EN_CHOCO` et `MSG_DIS_CHOCO` dans le fichier de langage : `language_en.h` (à retrouver en annexe). Nous n'avons pas défini ces messages dans les autres langues.

Cette fonction change l'état d'une variable externe, globale à tous les fichiers, qui permet de savoir si l'utilisateur a choisi ou non d'imprimer du chocolat : `chocolate_mode`.

COMMUNICATION ENTRE LES DEUX SYSTEMES

Pour que l'Arduino Uno qui régule la pression et la température puisse donner l'ordre d'imprimer à l'Arduino Méga, nous définissons une des pins du dernier en lecture, et nous ajoutons ce code dans la boucle du fichier « `Marlin_main.cpp` » :

```
// Check pressure status every n milliseconds
#ifdef CHOCO
  if (chocolate_mode && card.sdprinting)
  {
    int print_choco = digitalRead(pinChoco);
    while (print_choco == 0)
    {
      gcode_M25(); //pause SD Print
      print_choco = digitalRead(pinChoco);
    }
    gcode_M24(); //Resume SD Print
  }
#endif //CHOCO
```

FIGURE 12 - LOOP

On vérifie d'abord que le mode d'impression du chocolat a bien été sélectionné, et que l'on est en train d'imprimer. On lit la valeur sur la pin de communication. Lorsqu'on trouve un ordre d'arrêt d'impression (0), on lance l'instruction `pause`. L'impression est alors bloquée, jusqu'à ce qu'un ordre d'impression (1) soit lu, alors on lance l'instruction de reprise d'impression.

Il a été nécessaire de trouver un pin non utilisées par l'Arduino Méga pour effectuer cette liaison, ce qui ne s'est pas révélé simple puisque le montage électronique de cet Arduino n'a pas été conçu par nos soins.

Toutes les modifications apportées au firmware sont encapsulées dans une définition, qu'il suffit de commenter pour désactiver tous changements liés à l'impression du chocolat.

TESTS



FIGURE 13 - TEST D'EXTRUSION DE CHOCOLAT

Lors des derniers jours, nous avons assemblé tout notre système pour tenter des impressions de chocolat. Nous en avons extrudé un peu, mais très vite l'étanchéité s'est révélée problématique. Néanmoins, à partir de ces essais, nous avons pu valider notre modèle.

AMELIORATIONS

Même si notre système est fonctionnel sur le principe, il est améliorable pour le rendre tout à fait utilisable.

ISOLATION

Le problème principal pour imprimer du chocolat, ou tout autre matériau aussi visqueux, c'est l'étanchéité. Pour commencer, nous avons modélisés nous mêmes certaines pièces de notre extrudeur, que nous avons ensuite imprimés en 3D. Le PLA étant imprimé par couche, il n'est pas totalement étanche, d'autant plus avec les défauts d'impressions. Ces pièces comportaient donc de grosses fuites. Nous avons essayé de pallier à ce problème en augmentant l'épaisseur de nos pièces et en augmentant les surfaces de contact entre les pièces et le tuyau, mais cette solution n'était pas suffisante. La solution logique serait de réaliser ces pièces en métal, en y ajoutant un pas de vis, pour garantir une meilleure étanchéité entre les éléments du système.

Pour s'assurer d'une étanchéité maximale, nous proposons d'incorporer de vrais joints entre les différentes pièces de notre système. Ceux-ci garantiront l'absence de fuites.

MATERIAUX

Si le but est d'imprimer de la nourriture, c'est-à-dire d'ingérer les pièces imprimées, il sera nécessaire de remplacer le cuivre et le PLA par des métaux alimentaires, comme l'acier inoxydable. Pour des raisons économiques et de simplicité d'acquisition, nous avons opté pour le cuivre, qui suffisait amplement pour nos tests.

GESTION DE PRESSION

CAPTEUR DE PRESSION

Le capteur de pression acheté ne convenait pas au système, et c'est pourquoi actuellement, nous utilisons le manomètre du compresseur pour mesurer la pression. Cependant, intégrer un capteur utilisable dans le système et lire sa valeur sur l'Arduino Uno est tout à fait envisageable.

COMMANDE COMPRESSEUR

De plus, pour automatiser totalement la régulation de la pression, nous préconisons l'utilisation d'un compresseur à commande numérique. En modifiant légèrement le code Labview, nous pouvons générer les signaux de commandes pour ce compresseur en fonction de la pression. (i.e. mesure pression trop faible : pause impression – activer compresseur)

REFROIDISSEMENT

Faute de temps nous n'avons pas du tout incorporé de solutions pour refroidir l'impression. De ce fait, notre système pourrait imprimer des objets avec ratio volume/surface faible, tel que des décorations de gâteaux. Si l'utilisateur souhaite imprimer des structures réellement en 3D, alors un système de refroidissement devient nécessaire.

Nous proposons deux pistes de solutions : En premier lieu, il faudrait refroidir le chocolat directement en sortie d'extrudeur, pour cela un petit ventilateur attaché à l'extrudeur et bien orienté, tel qu'on en trouve sur un extrudeur classique, sera tout à fait efficace. Ensuite, si l'imprimante se trouve dans une pièce à température ambiante relativement élevée, il devient cohérent d'isoler l'imprimante à l'aide d'une enceinte thermique refroidie.

ARDUINO

Pour faciliter le développement de notre projet, nous n'avons développé séparément la régulation de température et pression du contrôle de l'impression. Cependant, comme LabView fonctionne avec un programme à implémenter sur l'Arduino, il est envisageable de rassembler les deux, et ainsi, de n'utiliser qu'un seul Arduino, l'Arduino Méga qui a la mémoire suffisante pour accueillir les deux programmes..

IMPRIMANTE

L'imprimante est fonctionnelle, néanmoins rajouter un peu d'habillage pour éviter que les fils électriques ainsi que les pièces bougent. Un vrai support pour l'écran LCD serait un plus.

Actuellement il manque une pièce pour attacher notre extrudeur à la pièce mobile sur les axes.

Enfin, lorsque l'Arduino Uno ordonne d'arrêter l'impression, il y a un délai avant l'arrêt. Il est dû au fait que l'Arduino Méga charge une série d'instructions qu'il exécute, avant de poursuivre son code. Ce délai, qui provient du firmware original, est une problématique intéressante puisqu'il impliquerait des erreurs d'impression.

CONCLUSION

Nous obtenons une imprimante fonctionnelle, et un extrudeur dont nous avons validé le principe. Néanmoins, certaines améliorations sont nécessaires pour que l'extrusion de chocolat, ou autre matériaux alimentaires, soit opérationnel. Nous espérons avoir construit les bases et proposé les problématiques de futurs projets.

L'imprimante 3D est une technologie fascinante, et ce projet a été avant tout l'occasion de s'y familiariser, par le biais d'impression de pièces, de construction et calibration de l'imprimante et de la compréhension et modification de son firmware. De plus, l'utilisation de Labview et du langage de programmation C++, rarement pratiqués dans le cadre de nos études, constituent une bonne pratique de l'apprentissage par soit même, compétence nécessaire aux ingénieurs.

Enfin, c'était aussi une première réelle expérience de gestion de projet, et de ses aléas. Nous avons notamment rencontrés des problèmes avec un de nos fournisseurs, ce qui nous a retardé. Nous avons néanmoins su nous adapter pour avancer dans le projet.

Suite à notre travail sur le firmware de notre imprimante, l'un de nous a eu l'opportunité de trouver son stage dans l'impression 3D, dans le développement du firmware d'une imprimante 3D à béton.

ANNEXES

RACCORD TUYAU DE CUIVRE

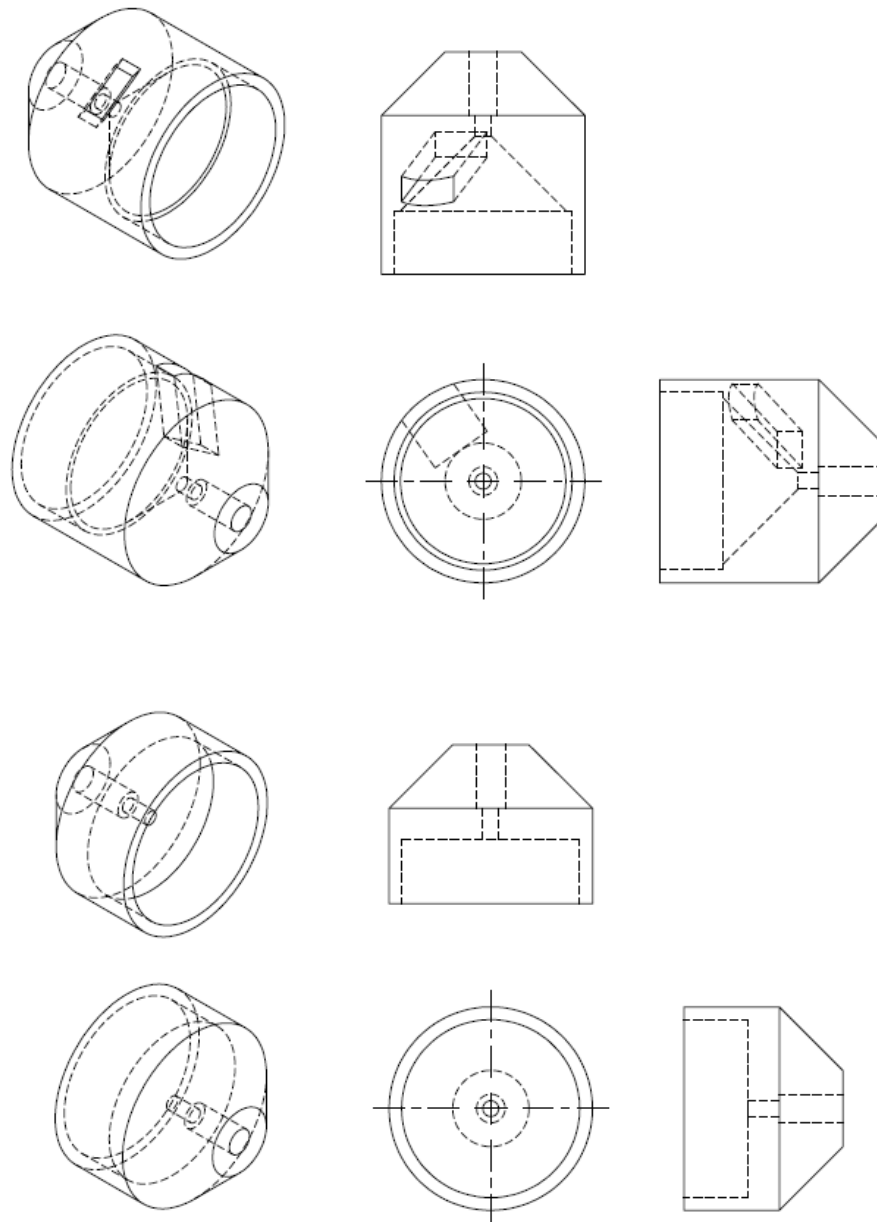


FIGURE 14 - PIÈCES DE RACCORD DU TUYAU DE CUIVRE

AUTRES MODIFICATIONS DU FIRMWARE

Le define autorise nos changements du firmware, et on définit une variable externe.

```
//*****
//*****CHOCOLATE PRINTING*****
//*****
//To permanently disable chocolate printing settings and functions, comment the next line
#define CHOCO

//if the user enables : chocolate_mode = 1 ; default = 0.
#ifndef CHOCO
  extern int chocolate_mode;
#endif
```

FIGURE 15 - FICHIER CONFIGURATION.H

On définit les variables globales, et on utilise la variable externe. Dans la fonction setup, on définit la pin utilisée pour la communication comme une entrée.

```
#ifndef CHOCO
  extern int chocolate_mode;
  int pinChoco=4; //Pin to read if we allow extrusion (com with Arduino Uno)
#endif

void setup() {
  setup_killpin();
  setup_filrunoutpin();
  setup_powerhold();
  MYSERIAL.begin(BAUDRATE);
  SERIAL_PROTOCOLLNPGM("start");
  SERIAL_ECHO_START;

  #ifndef CHOCO
    pinMode(pinChoco, INPUT);
  #endif
```

FIGURE 16 - FICHIER MARLIN_MAIN.CPP

Pour définir les messages à afficher dans le menu « prepare » par l'item correspondant.

```
#ifndef CHOCO
#define MSG_EN_CHOCO          "enable chocolate"
#define MSG_DIS_CHOCO        "disable chocolate"
#endif
```

FIGURE 17 - FICHIER LANGUAGE_EN.H

DIAGRAMME LABVIEW

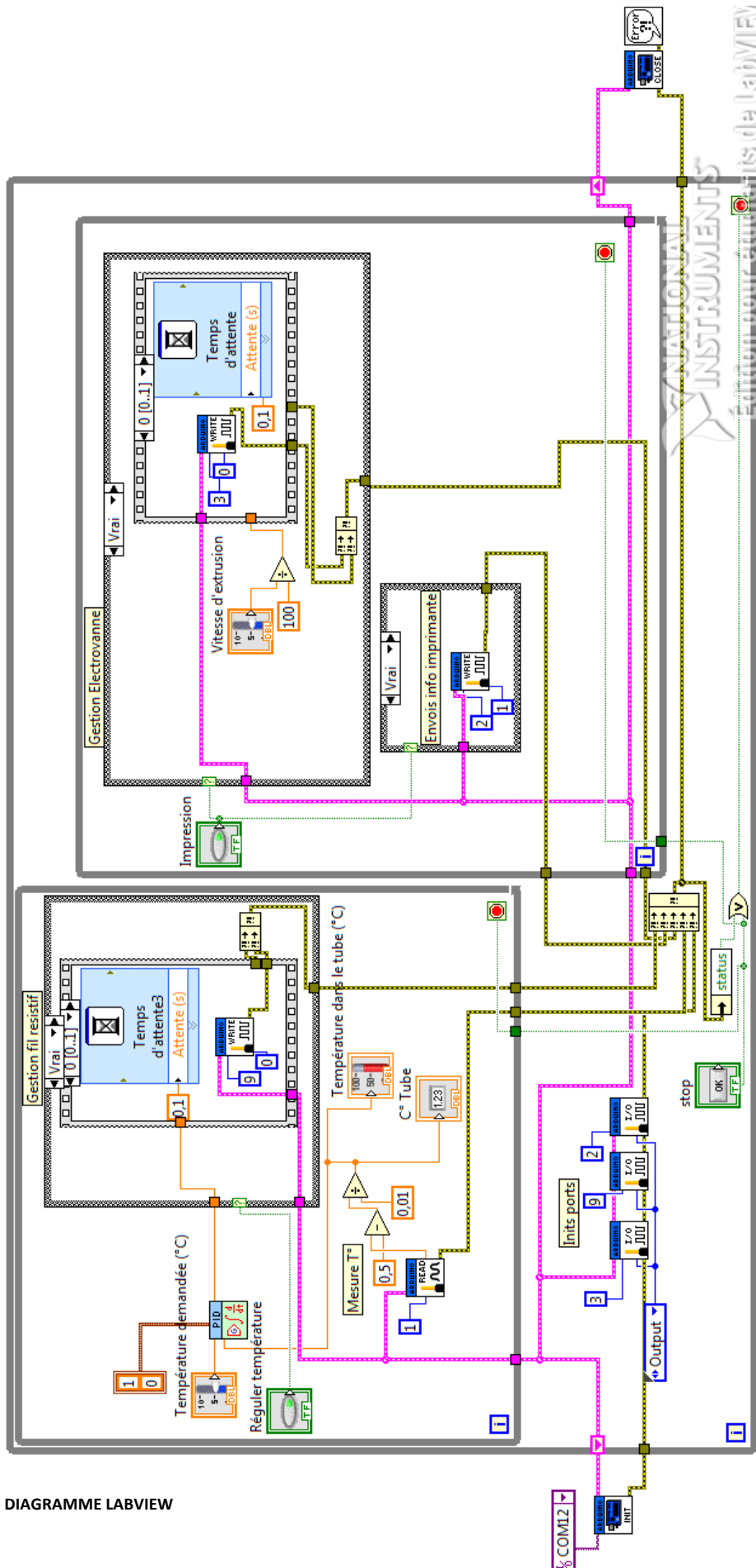


FIGURE 18 - DIAGRAMME LABVIEW