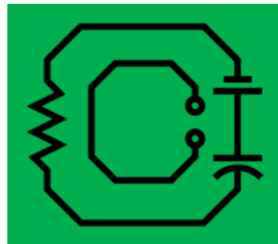


UNIVERSITÉ DE SHERBROOKE  
Groupe technique



Compétition de  
Conception de  
Circuits Imprimés

# Documentation C3I

Inch2Cent

Rédigé par :  
Équipe #5,  
Élodie Beauchemin

En date du : 27 décembre 2024



## Table des matières

1. Introduction .....	3
1.1. Contexte d'application .....	3
1.2. Contraintes du projet .....	3
1.2.1. Contraintes technologiques .....	3
1.2.2. Contraintes de temps et de budget .....	4
1.3. Description du produit réalisé .....	4
1.3.1. Présentation globale .....	4
2. Développement .....	6
2.1. Conception électronique .....	6
2.1.1. Choix technologiques .....	6
2.1.2. Schémas électriques et conception PCB .....	6
2.1.3. Parties non-fonctionnelles .....	11
2.2. Conception informatique .....	13
2.2.1. Présentation globale du code .....	13
2.2.2. Librairies .....	13
2.3. Gestion .....	14
2.3.1. Temps .....	14
2.3.2. Budget .....	14
3. Conclusion .....	16
4. Références .....	17
Annexe A <b>Schéma électrique</b> .....	18



# 1. Introduction

## 1.1. Contexte d'application

Inch2Cent permet de convertir des dimensions en pouces (po) afin de les avoir en centimètres (cm). Ainsi, un clavier permet d'indiquer une longueur qu'il est souhaité de convertir et l'écran indique le résultat.

De cette façon, la conversion de longueur s'effectue directement. En effet, cette règle évite d'avoir à calculer par soi-même la conversion. Ainsi, le processus est accéléré et l'utilisation d'un outil supplémentaire comme une calculatrice n'est plus nécessaire.

Cet outil peut être utilisé pour toutes les envergures de projet nécessitant une conversion de longueur. En effet, il est adapté pour de courtes longueurs comme dans les projets scolaires et pour de plus grandes longueurs comme dans la construction.

## 1.2. Contraintes du projet

### 1.2.1. Contraintes technologiques

Lors de ce projet, le groupe technique a émis quelques contraintes dans les règlements de la compétition [1]. En effet, les dimensions du prototype doivent être de 15,5 cm par 3,3 cm maximale. Également, afin d'éviter des pénalités lors du pointage, il ne faut pas utiliser de DevBoard. Finalement, il est fortement suggéré de produire un circuit imprimé (PCB) de deux couches seulement.

En plus de ces contraintes, certains des composants choisis imposent certaines limitations. Notamment, l'afficheur à sept segments ne comporte que 3 plages d'affichage. Ainsi, ce ne sont pas tous les nombres demandés qui peuvent être convertis. De cette façon, si la longueur en centimètres comprend quatre chiffres ou plus, l'affichage n'est pas effectué. Dans le projet, cette limitation impose que toutes les longueurs supérieures à 390 po ne soient pas affichées. Également, toutes les valeurs converties sous 390 po présentent différents degrés de précision en fonction du nombre de décimales qu'il est possible d'afficher.

Aussi, le microcontrôleur choisi, ATMEGA328P, fournit seulement une tension égale à VCC. Ici, elle correspond à 3 V. Ainsi, cette alimentation doit être suffisante pour les composants contrôlés par le microcontrôleur.

### 1.2.2. Contraintes de temps et de budget

Lors de la planification du projet, quelques dates importantes ont été prises en compte [1]. Entre autres, le projet doit être remis pour le 27 décembre. À ce moment-là, toute la documentation, ainsi que les documents de conception du projet doivent être remis. Ainsi, l'ensemble de la planification est organisé autour de cette date butoir. Également, l'idée de projet doit être remise pour le 21 octobre. Finalement, la commande des pièces et du PCB doit être faite pour le 22 novembre. De cette façon, l'ensemble de la conception électrique doit être effectuée pour cette date. Au terme de la compétition, une journée d'évaluation est prévue le 5 janvier afin de présenter le travail accompli.

En plus des échéances imposées, un budget prédéfini est imposé à l'équipe [1]. En effet, au terme du projet, la production d'un PCB doit être inférieure à 25 \$. Ce montant doit prendre en compte tous les composants utilisés dans le produit final. Pour le développement du projet, un budget de 100 \$ est alloué à l'équipe. Ainsi, toutes les dépenses liées au surplus d'inventaire pour les composants entrent dans ce budget.

## 1.3. Description du produit réalisé

### 1.3.1. Présentation globale

La figure 1 présente le schéma bloc du projet Inch2Cent. Les différentes parties du projet y sont présentées.



Figure 1 : Schéma bloc de Inch2Cent

La conception de ce projet est basée sur l'utilisation d'un microcontrôleur. Pour l'ensemble du circuit électrique, l'alimentation utilisée est une batterie au lithium de 3 V.

Ce prototype est composé d'un clavier à douze boutons, soit les chiffres 0 à 9, un point et la touche d'entrer. Ce clavier permet d'indiquer le nombre qu'il est souhaité de convertir. Lorsque l'ensemble du nombre est indiqué, il est possible d'appuyer sur entrer pour démarrer la conversion. Le microcontrôleur effectue la conversion, puis le résultat est affiché sur l'afficheur à sept segments. Ce résultat est affiché pendant 15 secondes avant que le prototype soit réinitialisé. De cette façon, il est possible d'indiquer un nouveau nombre à convertir. Pendant la période d'affichage, la lecture des boutons n'est pas effectuée. Ainsi, il faut absolument attendre la réinitialisation pour indiquer un nouveau nombre. Simultanément à tout ceci, une diode électroluminescente (DEL) allume afin d'indiquer que



le prototype est allumé. Lorsque le bouton sélectionné est lu par le programme, la DEL clignote. Ainsi, le bouton doit être maintenu jusqu'à ce clignotement. Aussi, lorsque l'affichage du résultat se termine, la DEL peut partager deux messages soit elle clignote, soit elle s'éteint. Elle clignote dans deux cas.

1. Si le nombre ne peut pas être affiché parce qu'il est composé de quatre chiffres ou plus.
2. Si la longueur en centimètres est supérieure à 100.

En fait, la DEL clignote afin d'indiquer une erreur ou un arrondissement du résultat. La DEL s'éteint s'il n'y a pas eu d'arrondissement majeur sur le résultat.



## 2. Développement

### 2.1. Conception électronique

#### 2.1.1. Choix technologiques

Pour ce prototype, un microcontrôleur est sélectionné afin de permettre de lire les données transmises par le clavier et de pouvoir contrôler l'affichage. En effet, le type de contrôle sélectionné doit être capable de passer à travers l'ensemble des étapes de la séquence présenté dans la section 2.2.1. Si les étapes ne peuvent pas être suivies, le fonctionnement du prototype ne peut pas être atteint. Également, le microcontrôleur est une option de petite taille et à faible coût qui permet d'atteindre le résultat escompté. Étant donné la grosseur du prototype et le budget alloué, ce sont des aspects importants à prendre en compte pour le choix de contrôle.

Lors de la sélection des pièces, le type d'afficheur est l'une des décisions majeures. En effet, l'équipe veut une stratégie à faible coût, fiable et facile à comprendre. De cette façon, elle s'arrête sur l'afficheur à sept segments. L'équipe doit choisir un nombre de positions qui semble permettre l'affichage d'un grand nombre de résultats sans pour autant compromettre la précision de ceux-ci. Également, l'équipe doit prendre en compte la dimension du composant. Plus le nombre de positions est grand, plus la pièce est grosse. Toutefois, l'espace sur le PCB est tout de même limité. Ainsi, en prenant en compte tous ces facteurs, l'équipe décide que l'afficheur à sept segments de trois positions est la meilleure option pour le projet.

#### 2.1.2. Schémas électriques et conception PCB

L'ensemble du schéma électrique conçu par l'équipe est présenté à l'annexe A. Une description détaillée des différentes sections du schéma est présentée ci-dessous.

Le schéma électrique de l'alimentation du circuit et ses protections est présenté à la figure 2. Le circuit électrique est alimenté par une batterie au lithium de 3 V. Étant donné le type de batterie utilisé, une protection contre les inversions de polarité n'est pas utile. En effet, lorsque ce type de batterie est branché à l'envers, aucune alimentation n'est transmise au circuit. Aussi, aucune protection contre la surtension n'est utilisée, car les batteries peuvent atteindre un maximum de 3,3 V [2]. Étant donné que la différence de tension est négligeable, les composants peuvent supporter cette faible surtension. Toutefois, un fusible est utilisé afin de protéger le circuit électrique en cas de court-circuit. Celui-ci est dimensionné en fonction de la charge totale du circuit, soit 0,245 A. Ici, la charge du circuit doit correspondre maximale à 80 % du courant admissible par la protection [3]. Ainsi, un fusible de 0,37 A est sélectionné.

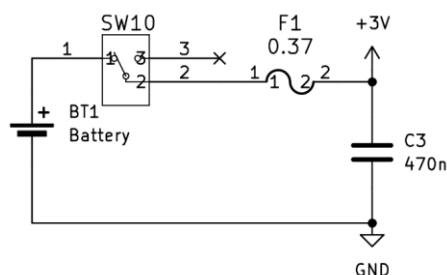


Figure 2 : Schéma électrique de l'alimentation

Les branchements des différents boutons qui composent le clavier sont présentés à la figure 3. Ici, un branchement nécessitant une seule broche du microcontrôleur est sélectionné. En effet, le microcontrôleur, ATMEGA328P, possède un nombre de broches analogique limité. Ainsi, cette stratégie de branchement permet de réduire grandement l'utilisation du microcontrôleur. Toutefois, il faut tout de même être capable de reconnaître quel bouton est sélectionné par l'utilisateur. Pour ce faire, un diviseur de tension est calculé avec 2-1 [4].

$$R_1 = \frac{(V_{CC} - V_{IO}) \cdot R_2}{V_{IO}} \quad 2-1$$

La tension présente à la patte commune de tous les boutons permet de l'identifier. Étant donné qu'ils sont associés à des résistances différentes, cette tension change selon le bouton sélectionné. Les résistances sont choisies en fonction des tensions désirées pour chacun d'entre eux. Ainsi, les tensions entre 0,25 et 3 V sont réparties sur douze boutons avec un intervalle de 0,25 V. Le microcontrôleur possède une résolution de 10 bits, ce qui permet de calculer avec l'équation 2-2 que la précision de l'ADC est 0,003 V [5]. Ainsi, l'écart est suffisant pour que le microcontrôleur puisse repérer la différence entre les boutons.

$$\text{Résolution} = \frac{V_{CC}}{2^n} \quad 2-2$$

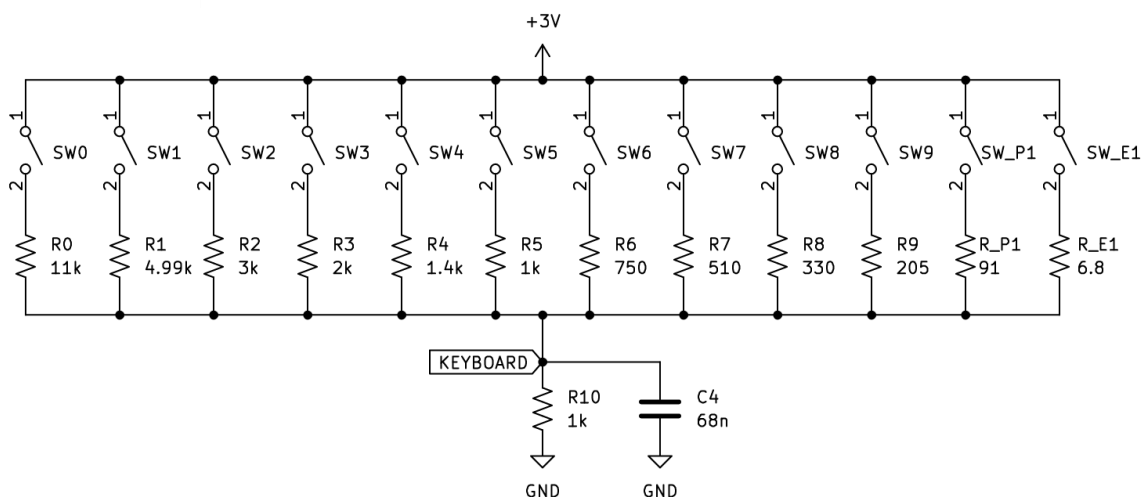


Figure 3 : Schéma électrique du clavier

Présentement, aucun traitement n'est effectué pour les mauvaises utilisations du clavier. En effet, si l'utilisateur appuie sur plusieurs boutons simultanément, le programme considère tout de même la tension à la broche. Ainsi, la tension mesurée ne correspond à aucun des boutons sélectionnés. Par exemple, si les boutons 3 et 4 sont appuyés en même temps, le convertisseur comprend 6. Les deux boutons sélectionnés au même moment créent une nouvelle tension qui correspond à une valeur différente. À refaire ce projet, l'équipe mettrait en place une protection contre ce type d'utilisation afin d'assurer un usage optimisé de l'outil. Malheureusement, à ce moment-ci, elle n'est pas en moyen d'expliquer la méthode qui serait utilisée.

Les branchements associés à l'afficheur à sept segments sont présentés dans la figure 4. Sachant que les DELs de l'afficheur engendrent une perte de 2,6 V maximum par segment, il est calculé avec la loi d'Ohms que la résistance nécessaire est 39  $\Omega$ . Chacun des signaux de gauche sur l'image permettent d'activer les différents segments. En effet, lorsqu'ils sont à HIGH, le segment allume. Pour contrôler la position du chiffre qui allume, ce sont plutôt les signaux de droite. Lorsque la position est éteinte, le signal est placé à HIGH. Lorsqu'il est ouvert, il est à LOW afin de créer une différence de potentiel dans la DEL.

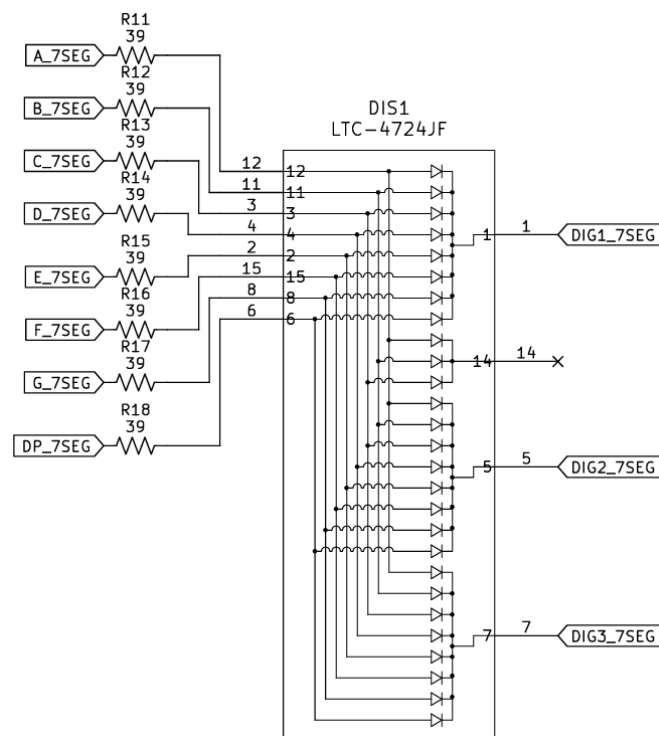


Figure 4 : Schéma électrique de l'afficheur à 7 segments

Le schéma électrique de la DEL est présenté à la figure 5. Ici, une résistance est placée en série avec la DEL pour que l'ensemble du 3 V soit épuisé lorsqu'on est rendu à la broche du microcontrôleur. Toutefois, si c'était à refaire, la logique de branchement serait inversée.



Présentement, la DEL est toujours allumée, mais ce n'était pas le résultat escompté. En effet, il aurait fallu que le branchement soit tel qu'il est présenté à la figure 6. De cette façon, la DEL allume seulement lorsque le clignotement est nécessaire.

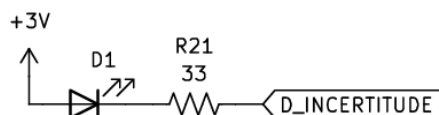


Figure 5 : Schéma électrique de la DEL

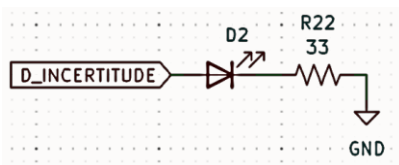


Figure 6 : Schéma électrique souhaité pour la DEL

Les branchements du microcontrôleur sont présentés à la figure 7. Ainsi, toutes les sections présentées précédemment sont associées à l'aide de ce schéma.

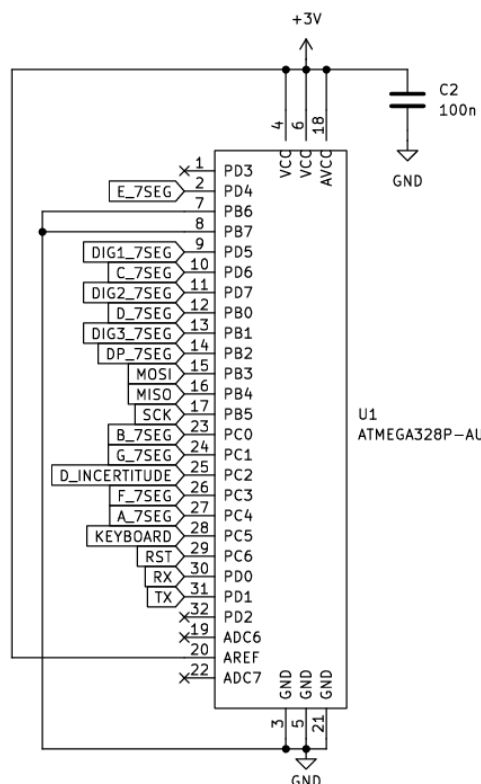


Figure 7 : Branchement du microcontrôleur

Pour permettre la programmation du PCB, un connecteur AVR-ISP-6 est nécessaire. Son branchement est présenté à la figure 8. Également, l'accès au *reset* du microcontrôleur peut être nécessaire lors de l'utilisation. Celui-ci est présenté à la figure 9.

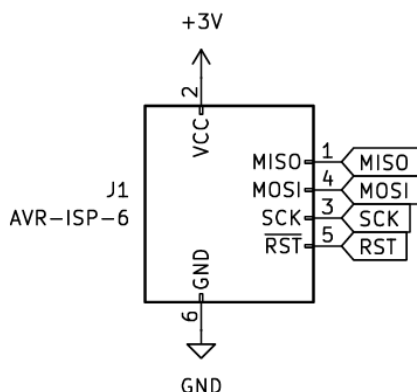


Figure 8 : Schéma électrique du AVR-ISP-6

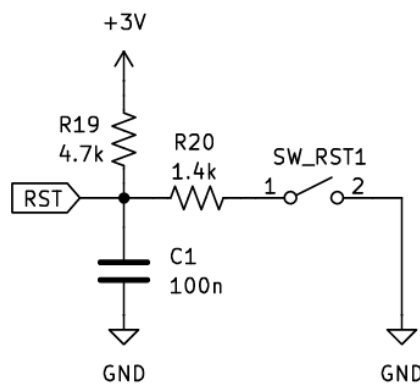


Figure 9 : Branchement pour le *reset* du microcontrôleur

À l'aide de toutes les sections du schéma électrique, le PCB présenté à la figure 10 et à la figure 11 est obtenu. Sur le PCB, toutes les pièces des différentes sections sont situées proches les unes des autres. D'ailleurs, le condensateur de découplage du microcontrôleur est situé près de celui-ci afin de réduire le bruit provenant des autres pièces. Mis à part les condensateurs de découplage, il y a une seule section dont la disposition des composants est importante, le clavier. En effet, les boutons doivent être disposés de façon logique afin que l'utilisation soit intuitive. Ainsi, les cinq premiers boutons à partir de la gauche sur les deux rangées sont les chiffres en ordre croissant et les deux boutons restants sont le point et la touche d'entrer.

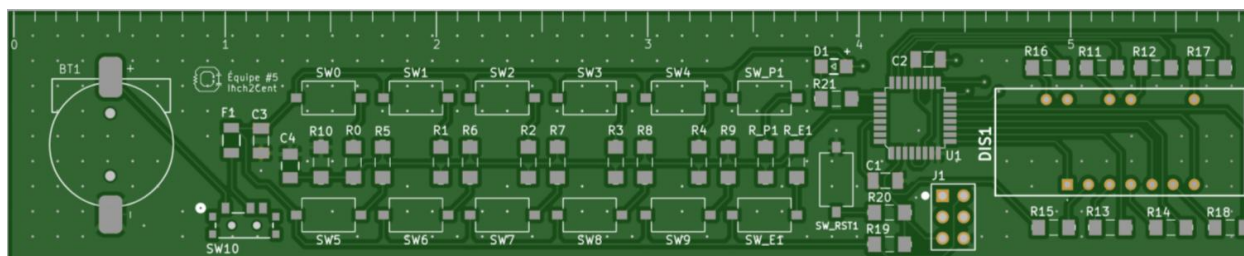


Figure 10 : PCB vu du dessus

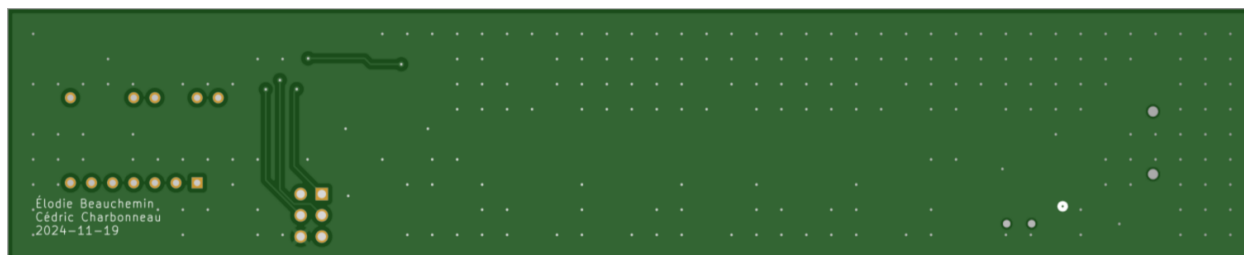


Figure 11 : PCB vu du dessous

### 2.1.3. Parties non fonctionnelles

Au terme de ce projet, seulement un élément ne fonctionne pas et un, est incomplet. En effet, les segments A et F de l'afficheur ne s'allument pas. La figure 12 présente la position des différents segments sur l'afficheur à sept segments.

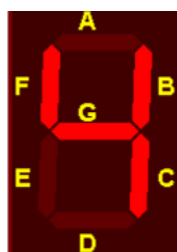


Figure 12 : Position des différents segments de l'afficheur [6]

Le fonctionnement de ces segments est affecté par le type de broche sélectionné. Pour sa bonne utilisation, des broches numériques sont nécessaires. Toutefois, ils ont été placés sur des broches analogiques. Lors de la conception, l'équipe s'est référée à la figure 13 pour attribuer les broches. Ainsi, lorsqu'ils ont été assignés aux broches 19 et 22, l'équipe pensait que les broches permettaient d'être analogique et numérique.

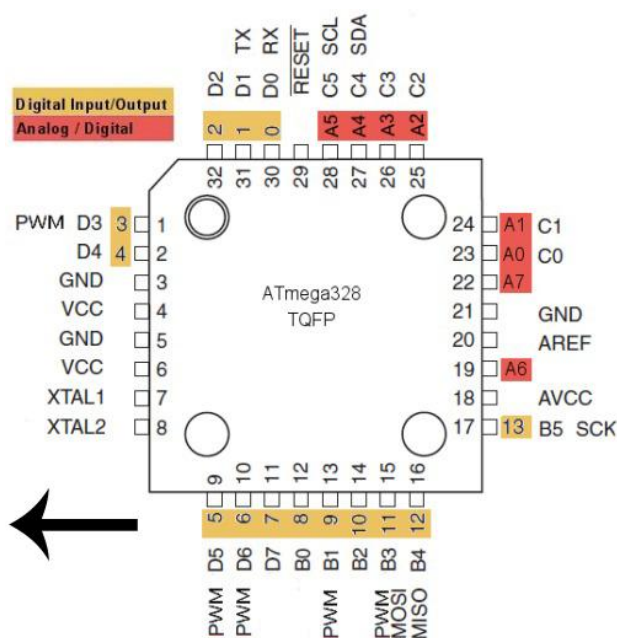


Figure 13 : Source erronée pour l'attribution des broches [7]

Toutefois, lors de la programmation, elle s'est rendu compte que les broches sont seulement analogiques. Tel que la figure 14 le montre, ces broches correspondent à A6 et A7 sans avoir d'équivalent numérique.

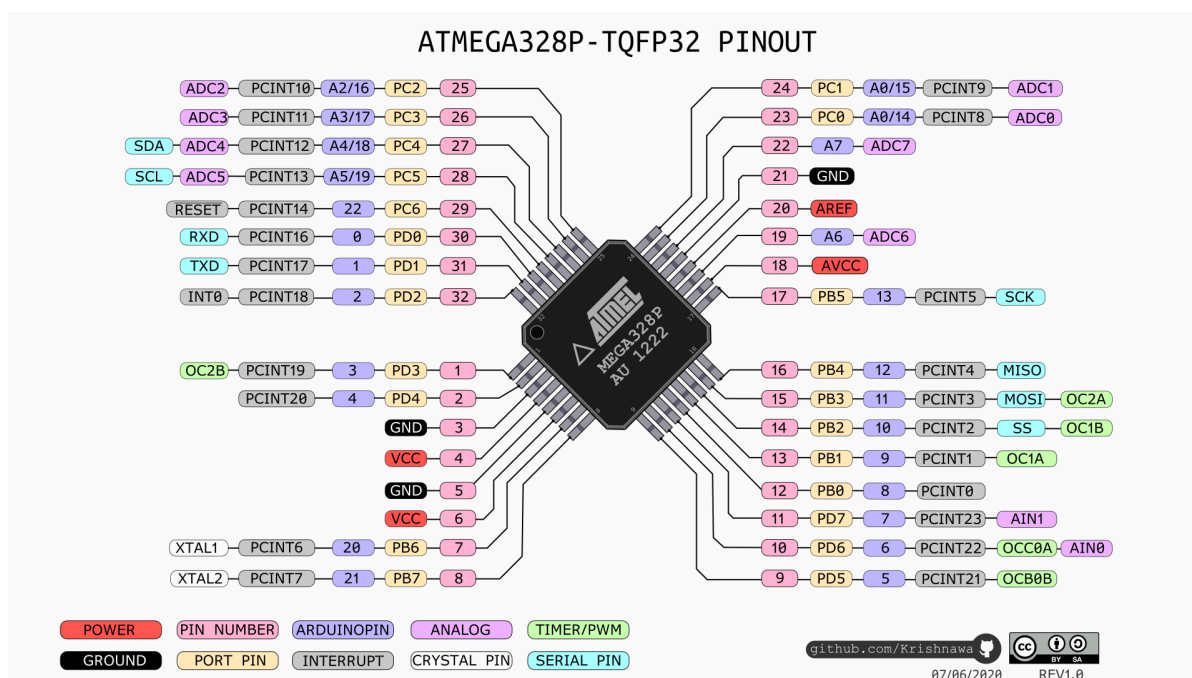


Figure 14 : Les broches du ATMEGA328P-AU [8]

Pour les faire fonctionner, il faudrait déplacer les traces de ces segments vers les broches 26 et 27 (comme présenté dans le schéma électrique de l'annexe A). Malheureusement, l'équipe a manqué de temps pour effectuer cette correction. Ainsi, les segments sont constamment fermés. Cette erreur engendre une difficulté dans la lecture du résultat affiché.

Un deuxième élément crée une difficulté dans la lecture du résultat. En effet, la lueur du chiffre à la position précédente est présente à la position actuelle (*ghosting*). La figure 15 présente un exemple de *ghosting*.



Figure 15 : Exemple de *ghosting* [9]

Bien que l'équipe ait tenté différentes méthodes pour retirer cet effet, il reste présent. Notamment, l'ensemble de l'afficheur est éteint entre l'affichage de chaque position afin d'essayer de réduire cet effet. À refaire le projet, davantage de temps serait investi afin de régler le problème.

## 2.2. Conception informatique

### 2.2.1. Présentation globale du code

La figure 16 présente le schéma bloc général du fonctionnement du code de Inch2Cent.

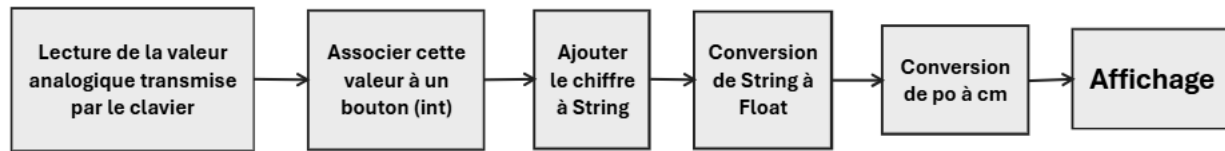


Figure 16 : Schéma bloc général du code de Inch2Cent

Le fonctionnement du programme repose sur six étapes différentes. Celles-ci permettent de lire la longueur en pouce et d'afficher la longueur en centimètres correspondante.

1. Lors de la sélection d'un bouton, une valeur analogique est transmise au microcontrôleur. Chaque bouton possède une valeur qui lui est spécifique. Le code permet de lire cette valeur.
2. La valeur analogique lue est associée à une variable (de type int) qui correspond au bouton sélectionné. Le code identifie celui-ci à l'aide d'une vérification par rapport aux intervalles de valeurs analogiques pour chaque bouton.
3. Cette variable est ajoutée à la fin du String qui compose la valeur d'entrée. De cette façon, il est possible de construire un nombre à partir de chiffres. Par exemple, 1-2-3 correspond à 123.
4. Cette String est convertie en variable (de type float) à l'aide d'une fonction préétablie par Arduino. De cette façon, le nombre peut être utilisé pour des opérations mathématiques.
5. Le nombre obtenu est multiplié par 2,54 afin que la longueur devienne en centimètres.
6. La longueur en centimètres est affichée sur l'afficheur à sept segments à l'aide de trois fonctions.
  - a. La première permet de gérer l'affichage des chiffres sur une seule position.
  - b. La deuxième permet de gérer la position d'affichage.
  - c. La troisième gère l'ensemble de l'affichage pour les trois positions de façon simultanée.

### 2.2.2. Librairies

Lors de ce projet, la seule librairie utilisée est Arduino.h. L'ensemble des autres fonctions sont créées par l'équipe afin qu'elles puissent répondre aux différentes demandes spécifiques du projet.

La librairie fournie par Arduino est utilisée afin de pouvoir accéder à des fonctions et à des classes qui lui sont spécifiques. Notamment, la classe String permet de convertir une String



en float. Également, l'équipe n'a pas les connaissances nécessaires afin de programmer en registres. Ainsi, elle s'appuie sur l'information déjà disponible.

## 2.3. Gestion

### 2.3.1. Temps

Lors de la planification du projet, les échéances présentées dans la section 1.2.2 sont prises en compte. Pour l'ensemble de ces dates, un jeu de quelques jours est appliqué afin de prévenir les imprévus. De cette façon, les dates importantes de l'équipe sont comme suit :

- la commande de pièces et de PCB est prête pour le 18 novembre;
- le prototype est fonctionnel pour le 23 décembre;
- la documentation est rassemblée et remplie pour le 27 décembre;

Bien qu'il n'y ait pas d'échéance, il est prévu que la programmation soit effectuée en attendant l'arrivée de la commande de pièces.

Au terme du projet, il est possible de voir que l'ensemble de ces dates ne sont pas respectées. Comme prévu, la commande de pièces est effectuée pour le 22 novembre avec quelques jours d'avance. Également, la recherche liée à la programmation est commencée en attendant les pièces. Toutefois, la programmation de l'ensemble du code n'est pas terminée avant l'arrivée de la commande le 2 décembre. La programmation s'étire jusqu'au 14 décembre à cause de certains problèmes de logiciels. De cette façon, l'assemblage du PCB peut débuter durant la semaine du 17 décembre. La phase de tests du prototype débute le 19 décembre. Malheureusement, certains imprévus repoussent la date de complétion partielle du prototype au 27 décembre. Finalement, l'ensemble de la documentation peut tout de même être remise à temps pour le 27 décembre.

À la suite de ce projet, il est noté que l'équipe doit prévoir davantage de temps pour les tests et le débogage. Sinon, le projet peut rapidement tomber en retard et ne pas atteindre ses échéances. Également, toutes les phases du projet doivent être planifiées avec des dates de début et de fin. De cette façon, il y a un objectif clair et le glissement du projet sera contrôlé.

### 2.3.2. Budget

En début de projet, l'équipe ne s'impose pas de limitations supplémentaires à celles fournies par le groupe technique tel que présenté dans la section 1.2.2. Toutefois, elle essaie tout de même de choisir intelligemment les composants achetés afin de réduire au maximum les dépenses liées au projet. Ainsi, la commande de pièces est effectuée avec des composants à faible coût en prévoyant des pièces de surplus en cas d'imprévus.

Au terme du projet, il est évalué que les coûts liés à la production d'un prototype (en comprenant le PCB) sont de 20,65 \$. Pour les pièces seulement, les dépenses s'élèvent à



16,65 \$. Ainsi, l'équipe réussit à rester sous le budget de 25 \$ alloué. Pour le développement de ce prototype, seulement une commande de pièces s'élevant à 46,62 \$ est effectuée. De cette façon, le budget de 100 \$ est respecté.

Lors de la compétition, une seule commande de PCB est effectuée. Celui-ci comporte une correction qui aurait pu être modifiée lors d'une deuxième commande, mais cela n'a pas été fait. La réparation nécessaire est présentée dans la section 2.1.3.



### 3. Conclusion

En terme général, le projet est resté le même que celui initialement établi. En effet, les fonctionnalités choisies étaient réalistes en fonction du temps donné et des compétences de l'équipe. De cette façon, l'équipe n'a pas eu à sacrifier de fonctionnalités en cours de route. Toutefois, l'équipe a tout de même noté une fonctionnalité qui aurait été intéressante à ajouter au prototype. En plus de la réinitialisation du programme après 15 secondes, il serait bien d'ajouter une réinitialisation manuelle pour l'utilisateur. Malheureusement, l'équipe n'avait pas suffisamment de temps pour effectuer cette modification.





#### 4. Références

- [1] J. Turcotte et M. Caisse, *Document d'information de la compétition d'automne 2024*, 2024.
- [2] Energizer, *Energizer CR1220*.
- [3] P. Sagveen, «Fuse Tutorial,» 03 10 2017. [En ligne]. Available: <https://www.digikey.ca/en/articles/fuse-tutorial>. [Accès le 15 11 2024].
- [4] Digikey, «Calculateur de diviseur de tension,» 2024. [En ligne]. Available: <https://www.digikey.ca/fr/resources/conversion-calculators/conversion-calculator-voltage-divider#:~:text=Un%20diviseur%20de%20tension%20est,pour%20la%20mesure%20de%20tensions..> [Accès le 11 11 2024].
- [5] Arrow, «Analog-to-digital converters basics,» 17 04 2023. [En ligne]. Available: <https://www.arrow.com/en/research-and-events/articles/engineering-resource-basics-of-analog-to-digital-converters>. [Accès le 11 11 2024].
- [6] sonelec-musique, «Afficheur LEDs 7 segments 006,» 29 08 2010. [En ligne]. Available: [https://www.sonelec-musique.com/electronique\\_realisations\\_afficheur\\_leds\\_7seg\\_006.html](https://www.sonelec-musique.com/electronique_realisations_afficheur_leds_7seg_006.html). [Accès le 25 11 2024].
- [7] D. Chen, «Using atmega328p,» 22 11 2015. [En ligne]. Available: <https://dankc.com/using-atmega328p-8-16-20mhz/>. [Accès le 15 11 2024].
- [8] krishnawa, «ATMEGA328P TQFP32 PINOUT,» 2019. [En ligne]. Available: [https://www.reddit.com/r/arduino/comments/gyrdii/atmega328p\\_tqfp32\\_pinout/](https://www.reddit.com/r/arduino/comments/gyrdii/atmega328p_tqfp32_pinout/). [Accès le 03 12 2024].
- [9] JesperB, «Max7219 behaves strange when resetting arduino,» 04 2019. [En ligne]. Available: <https://forum.arduino.cc/t/max7219-behaves-strange-when-resetting-arduino/585204>. [Accès le 27 12 2024].



## Annexe A Schéma électrique

