

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي و البحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكر بلقايد- تلمسان

Université Aboubakr Belkaïd-Tlemcen

كلية التكنولوجيا

Faculté de Technologie

Département de Génie Electrique et Electronique (GEE)

Filière : Electronique



MASTER INSTRUMENTATION

PROJET DE FIN D'ETUDES

Présenté par : BENSFAFA Fatima Zohra & HADJI Hanane

Intitulé du Sujet

Réalisation d'une balance électronique

Soutenu en 2019, devant le jury composé de :

M^r HAMDOUNE Abdelkader

Pr

Univ. Tlemcen

Président

M^r BECHAR Hassan

MAA

Univ. Tlemcen

Encadrant

M^r NEMMICHE Ahmed

MCB

Univ. Tlemcen

Examineur

Année Universitaire 2018-2019

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Remerciements

Louange à ALLAH le tout Puissant, le Miséricordieux, le Clément, pour nous avoir accordé à vie, la santé, la force et la patience d'aller jusqu'au bout du rêve.

Nous tenons d'abord à remercier très chaleureusement notre respectueux encadreur Mr BECHAR Hassan, maitre-assistant classe A à l'Université Abou Bekr Belkaid pour son soutien permanent, ses conseils, ses orientations, sa disponibilité, son aide et sa solidarité.

C'est un plaisir et un honneur que nous avoir fait bénéficier de ses expériences et ses compétences.

Nous remercions également Mr HAMDOUNE A, professeur à l'Université Abou Bekr Belkaid, qui a bien voulu nous faire l'honneur de présider le jury et de juger notre thèse. Veuillez trouver dans ce travail, la reconnaissance et le témoignage de notre profond respect.

Nos respects et nos remerciements vont à Mr NEMMICHE A, maitre de conférences classe B à l'Université Abou Bekr Belkaid. Vous nous avez honorés d'accepter avec grande sympathie de siéger parmi notre jury de soutenance de notre PFE, votre présence et vos remarques nous honorent.

Merci à nos familles, en particulier nos parents, nos frères et sœurs pour leur présence, leur soutien et leurs conseils.

Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce modeste travail.

Dédicace

Je dédie ce modeste travail...

A mes chers parents

Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien être. Je vous remercie pour tout le soutien et l'amour que vous me portez depuis mon enfance et j'espère que votre bénédiction m'accompagne toujours.

Que ce modeste travail soit l'exaucement de vos vœux tant formulés, fruit de vos innombrables sacrifices, bien que je ne vous en acquitte jamais assez.

Puisse Dieu, le Très Haut, vous accorder santé, bonheur et longue vie.

A mes chères et adorables sœurs

Amel, Soumia, Karima, ma jumelle Wissam

A mon cher frère

Mohammed

Aucune dédicace ne saurait exprimer tout l'amour que j'ai pour vous, Votre joie et votre gaieté me comblent de bonheur. Puisse Dieu vous garder, éclairer votre route et vous aider à réaliser vos vœux les plus chers.

A mes petites nièces Wissam et Samar, mon neveu Ayoub

Que dieu vous garde et illumine vos chemins.

A mes grandes mères, mon grand père

Que dieu leurs procure bonne santé et longue vie.

A mes chers oncles, tantes, leurs époux et épouses

A mes chers cousins et cousines

Un remerciement particulier et sincère pour Meriem la prunelle de mes yeux

A Mon binôme et chère amie Hanane

En souvenir de notre sincère et profonde amitié et des moments agréables que nous avons passés ensemble. Je t'aime ma sœur

A mes chères amies

B.Anissa, B.Salima, B.Abir, je vous aime énormément vous êtes comptées comme mes vraies sœurs.

A toute la promotion « Instrumentation 2018 »

Je ne peux trouver les mots sincères pour vous exprimer mon affection et mes pensées, vous êtes pour moi des amis sur qui je peux compter. Je vous souhaite une vie pleine de santé et de bonheur.

A mes enseignants de la faculté de technologie

Vous nous avez formés et guidés. Je vous en remercie sincèrement.

« BENSIFA Fatima Zohra »

Que ce travail témoigne de mes respects :

A mes parents,

Grâce à leurs tendres encouragements et leurs grands sacrifices, ils ont pu créer le climat affectueux et propice à la poursuite de mes études. Aucune dédicace ne pourrait exprimer mon respect, ma considération et mes profonds sentiments envers eux. Que dieu leur procure bonne santé et longue vie.

A mes chères sœurs

Fatiha, Hadjer, Sabrina

Ma petite sœur Sabrina en lui souhaitant la réussite au BAC

A mes frères

Saïd, Mohamed, Othmane

A tous les membres de ma famille

Tantes oncles cousins et cousines, qui m'ont soutenue de loin et de près

A tous mes chères Amies,

L. Fatima, B. Abir, K. Asma, B. Dounai, B. Noura, B. Abir, B. Salima

*Sans oublier mon binôme **Fatima Zohra** et toute sa famille.*

A toute la promotion « Instrumentation 2018 »

....

Et tant d'autres,

Merci pour tous les bons moments passés ensemble.

« HADJI Hanane »

Résumé :

La balance mécanique est apparue au monde après la révolution française en 1799. Avec l'évolution technologique, on a découvert la balance électronique dont l'évaluation de la charge est effectuée par un capteur de force et traitée électroniquement. Cet instrument était l'objet d'intérêt de notre travail.

Ce projet de fin d'étude consiste en la conception et la réalisation d'un prototype d'une balance électronique à partir d'une jauge de contrainte, d'un module nommé HX711 comprenant la partie amplification associée à un CAN (convertisseur analogique numérique) 24bits, d'un afficheur OLED (Diode électroluminescente organique) et d'un clavier numérique, le tout piloté par une carte de type Arduino Uno avec un microcontrôleur ATmega328. Nous avons organisé ce mémoire en trois parties :

- Une partie, a été consacrée à une vision globale sur les balances.
- Une autre, a été consacrée, à l'étude théorique des différents éléments.
- Dans la dernière nous avons détaillé la réalisation pratique d'un point de vue Hardware et Software.

Mots clés : balance électronique, CAN, OLED, microcontrôleur.

ملخص:

بقيام الثورة الفرنسية سنة 1799 ظهر في العالم جهاز جديد لقياس الوزن الا وهو الميزان الميكانيكي. وبمرور الوقت ومع التطور التكنولوجي تم اكتشاف الميزان الالكتروني المستعمل حاليا الذي يقوم بقياس القوة التي يحدثها ثقل الحمل من خلال جهاز استقبال يسمى خلية قياس جهد الثقل. ولقد كان هذا النوع من الموازين محل اهتمام مذكرة تخرجنا. يتوقف مشروعنا النهائي على تصميم وانشاء نموذج من ميزان الكتروني المتكون أساسا من مقياس الانفعال ووحدة تسمى HX711 بما في ذلك جزء التضخيم المرتبط بمحول تناظري رقمي 24bits، وصمام ثنائي باعث للضوء العضوي، ولوحة مفاتيح رقمية، حيث يتم التحكم في كل ذلك بواسطة اردوينو اينو مع المتحكم الدقيق ATmega328. لقد قمنا بقسيم هذه الاطروحة الى ثلاثة أجزاء:

- جزء خصص لدراسة عامة حول الموازين.
- جزء اخر تمحور حول دراسة نظرية لمختلف العناصر المستعملة في مشروعنا.
- الجزء الخبير تناول الجانب التطبيقي.

الكلمات المفتاحية: ميزان الكتروني، محول تناظري رقمي، صمام، وحدة تحكم دقيقة.

Abstract:

The mechanical scale appeared to the world after the French revolution in 1799. With the technological evolution, one discovered the electronic scale where the load evaluation is carried out by a load cell and treated electronically. This instrument is the object of interest of this work.

Our end of studies project consists in designing and producing a prototype of an electronic scale using a strain gauge, a module named HX711 contained the amplification part associated to a 24bits ADC (analog to digital converter), an Oled (Organic Light-Emitting Diode) display and a digital keyboard, all of that will be piloted by an Arduino Uno board with an ATmega328 microcontroller. This thesis is organized in three parts:

- A part is devoted to a global vision on scales.
- Another part is devoted to the theoretical study of the different elements used.
- In the last part we detailed the practical achievement, Software/Hardware point of

view

Key words: electronic scale, ADC, OLED, microcontroller.

Listes des abréviations

GND : ground

CAN : Convertisseur analogique numériques

GPIO: General Purpuse Input /Output

OLED: Organic Light –Emitting Diode

LED: light emitting diode.

LCD: liquid Cristal display

I2C: Inter-Integrated Circuit

SPI: Serial Peripheral Interface

SCL: Serial Clock

SDA: Serial Data

MISO: Master Input, Slave Output

MOSI: Master Output, Slave Input

PGA : amplificateur à gain programmable

AOP : Amplificateur opérationnel

CLK : CLock

DAT : Data

Liste des figures

Figure 1.1:les différentes parties de la balance romaine	3
Figure1. 2:Romaine savoyarde ou piémontaise en fer forgé.....	4
Figure 1.3:Pesons cylindriques	5
Figure 1.4:Peson triangulaire	5
Figure 1.5:Peson demi-lune.....	5
Figure 1.6:Peson à cadran annulaire	5
Figure 1.7:Bascule décimale	6
Figure 1.8:Balance Roberval de grand luxe, signée Béranger & Cie	6
Figure 1.9:Grande balance de Roberval, de marque Frey & Studer à Strasbourg	7
Figure 1.10:Roberval de luxe datable vers 1870-1880	7
Figure 1.11:Principe de fonctionnement d'une balance à compensation électromagnétique	8
Figure 1.12:Principe du capteur hydraulique	8
Figure 1.13:Principe du capteur pneumatique.....	9
Figure 1. 14:Structure d'une jauge de contrainte	10
Figure2.1:Schéma bloc d'une balance électronique commerciale	
Figure2.2:Structure interne du capteur	13
Figure2.3:Diagramme détaillé des jauges de contrainte sur les capteurs de force à barres lorsque la force est appliquée	15
Figure2.4:les différents modèles de jauge de contrainte	15
Figure2.5:Déformation du corps d'épreuve	16
Figure 2 .6:Schéma électrique simplifié du capteur	16
Figure 2.7:le montage quart de pont.....	17
Figure 2.8:le montage demi-pont	17
Figure 2.9:le pont complet	18
Figure 2.10:Principe de fonctionnement du pont de Wheatstone	18
Figure 2.11:Schéma de principe du montage pont simple	19
Figure 2.12:Schéma de principe du montage demi-pont.....	20
Figure 2.13:Schéma de principe du montage pont complet	20
Figure 2.14:Circuit intégré INA125P.....	22
Figure 2.15:Amplificateur à point d'alimentation unique.....	23
Figure 2.16:Symbole d'un CAN à N bits	23
Figure 2.17:(1) signal analogique (2) signal échantillonné (3) quantifié.....	24
Figure 2.18:CAN simple rampe	25
Figure 2.19:Fonctionnement du CAN à double rampe	26
Figure 2.20:Schéma de principe d'un convertisseur à approximation successives.....	26
Figure 2.21:Structure d'un convertisseur Flash	27
Figure 2.22:Architecture interne d'un microcontrôleur	29
Figure 2.23:Photo d'un afficheur à LED.....	31
Figure 2.24:Photo d'un afficheur LCD	32
Figure 2 .25:Photo afficheur OLED	32
Figure 2.26:Photo d'unclavier à 16 touches.....	33

Figure 2.27:Support d'une carte SD.....	34
Figure 2.28:Photo d'une balance commerciale avec imprimante	35
Figure3.1 :Schéma synoptique du circuit.....	36
Figure 3.2:Schéma structurel de la carte Arduino Uno	37
Figure 3.3:Schémas simplifié de la carte Arduino Uno	38
Figure 3.4:brochage de l'ATMega328	38
Figure 3.5:La correspondance entre les broches de l'ATMega328 et les broches de l'Arduino	39
Figure 3.6:Circuit imprimé du hx711.....	41
Figure 3.7:Schéma fonctionnel du montage typique d'une balance	41
Figure 3.8:Les différentes broches du HX711 [19].....	42
Figure 3.9:Câblage du HX711 avec la jauge de contrainte et l'Arduino	44
Figure 3.10:Brochage du clavier	44
Figure 3.11:Schéma interne du clavier.....	45
Figure 3.12:Câblage du clavier avec Arduino.....	46
Figure 3.13:Structure d'une OLED	48
Figure 3.14:Câblage de l'OLED avec Arduino.....	50
Figure 3.15:Photo d'une carte Shield	
Figure 3.16:Câblage de la carte SD avec l'Arduino Uno.....	52
Figure 3.17:Interface de la plateforme Arduino.....	53
Figure 3.18:Barre de boutons Arduino.....	54
Figure 3.19:Fenêtre d'ouverture du logiciel PROTEUS.....	55
Figure 3.20:Schéma électrique du circuit sur ISIS.....	56
Figure 3.21:les étapes de la réalisation de circuit	

Liste des tableaux

Tableau3.1 : Spécifications de la carte Arduino Uno.....	39
Tableau3.2 : Brochage du HX711.....	42
Tableau3.3 : la correspondance entre la jauge, le HX711 et la carte Arduino.....	43
Tableau3.4 : Branchement d'OLED et Arduino.....	49

Table des matières

Résumé	V
Liste des abréviations	VI
Liste des figures	VII
Liste des tableaux	VIII
Introduction générale	1
Chapitre1 : Généralités sur les balances	
1. Introduction	2
1.1 Généralités sur les balances	2
1.1.1 Définitions	2
1.1.2 Historique de la mesure du poids	3
1.1.2.1 Balance mécanique	3
1.1.2.2 Balance électronique	7
1.1.3 La précision	10
1.1.4 Unité de mesure	11
1.2 Conclusion	11
Chapitre2 : les différents éléments d'une balance électronique	
2. Introduction	12
2.1 Description des différents blocs	12
2.1.1 Capteurs de force à jauge de contrainte	12
2.1.1.1 Constitution	13
2.1.1.2 Principe de pesage par la jauge de contrainte	14
2.1.1.3 Montages et choix des jauges de contrainte	15
2.1.1.4 Conditionnement des jauges	16
2.1.1.5 Le pont de Wheatstone	17
2.1.1.6 Principe de fonctionnement du pont de Wheatstone	18
2.1.1.7 Equilibrage du pont	19
2.1.2 Amplification :	21
2.1.3 Convertisseur analogique numérique :.....	23
2.1.3.1 Principe de la conversion analogique numérique :.....	23
2.1.3.2 Principales structures des C.A.N:.....	24

2.1.3.3 Caractéristiques des C.A.N :	27
2.1.4 La carte microcontrôleur :	29
2.1.4.1 Description de la carte :	29
2.1.4.2 L'utilisation des microcontrôleurs :	30
2.1.4.3 Avantages et inconvénients des microcontrôleurs :	30
2.1.5 Affichage :	31
2.1.6 Clavier :	33
2.1.7 Carte SD :	34
2.1.8 Imprimante :	35
2.2. Conclusion :	35

Chapitre3 : Etude et Réalisation

3. Introduction :	36
3.1 Schéma synoptique du circuit :	36
3.2 Description des différents composants :	37
3.2.1 La carte microcontrôleur de type Arduino Uno :	37
3.2.1.1 Caractéristiques de la carte Arduino Uno :	39
3.2.1.2 Les entrées / sorties numériques D0 à D13 :	40
3.2.1.3 Les entrées analogiques A0 à A5 :	40
3.2.2 Le module HX711 :	41
3.2.2.1 Fonctionnement du moduleHX711 :	41
3.2.2.2 Configuration des broches du module HX711 :	42
3.2.2.3 Caractéristiques et spécifications du HX711 :	43
3.2.2.4 Connexion du module avec la jauge de contrainte et Arduino Uno :	43
3.2.3 Le clavier :	44
3.2.3.1 Principe du clavier matriciel :	44
3.2.3.2 Connexion du clavier avec la carte Arduino Uno :	46
3.2.4 L'afficheur OLED :	46
3.2.4.1 Module I2C OLED :	46
3.2.4.2 Caractéristiques des afficheurs OLED :	47
3.2.4.3 Fonctionnement de la technologie OLED :	47
3.2.4.4 Avantages et inconvénients de la technologie OLED :	48
3.2.4.5 Connexions de l'afficheur avec la carte Arduino Uno :	49

3.2.5 La carte SD : (mémoire flash)	50
3.2.5.1 Description de la carte :.....	50
3.2.5.2 Présentation de la carte shield pour la carte SD :.....	50
3.2.5.3 Connexions avec la carte Arduino Uno :	52
3.3 Présentation de l'espace de développement intégré :.....	53
3.3.1 Les étapes de développement :.....	54
3.4 Définition du logiciel PROTEUSE :	54
3.4.1 Présentation de l'environnement ISIS :.....	55
3.4.2 Simulation du circuit :	55
3.5 Test du projet final :	56
3.6 Organigramme :.....	58
3.7 Conclusion :.....	61
Conclusion générale	62

Références bibliographiques

Annexes



Introduction Générale

Introduction Générale

Comparer la masse de deux objets ou définir celle d'un objet par rapport à une référence est un besoin qui remonte à plusieurs millénaires avant notre ère. Au fil du temps et grâce aux progrès techniques, la balance s'est améliorée afin de répondre, en fonction des différents usages, aux besoins toujours plus importants en termes de reproductibilité, d'exactitude et de précision de la mesure.

Aujourd'hui, les balances électroniques grand public sont à des tarifs tout à fait bon marché. Leur procédé consiste à détecter des variations de courant électrique en fonction de la pression exercée. Les balances haut de gamme arrivent même à calculer le taux de masse grasse du corps. Ils sont omniprésents sous toutes formes : balances de laboratoire, médicale, industrielles et des balances commerciales qui sont l'intérêt de notre travail.

En effet, ce projet consiste à étudier, à concevoir et à réaliser une balance électronique commerciale à partir d'un capteur de force qui est la jauge de contrainte, du module HX711 et de la carte microcontrôleur de type Arduino Uno qui est le cerveau du circuit à réaliser

Ce rapport présente le déroulement du projet et permet de suivre la progression de notre travail ainsi que les résultats obtenus et les améliorations possibles.

Notre but dans ce travail est de réaliser un prototype d'une balance électronique qui peut mesurer un poids de 5Kg. Et pour cela ce mémoire de fin d'étude est composé de trois chapitres.

- Le premier chapitre sera consacré à une présentation sur les balances. L'historique et les différents types de capteur de pesage et modes de fonctionnement, aussi nous avons abordé quelques définitions qui sont en rapport avec notre thème de projet fin d'étude.

- Le deuxième chapitre représente une étude théorique sur les différents éléments constituant une balance électronique tels que le capteur de pesage, le convertisseur analogique-numérique, l'amplificateur, la carte microcontrôleur, le clavier, la carte SD et l'imprimante si c'est nécessaire.

- Enfin, le troisième chapitre présente notre étude et réalisation dont laquelle nous avons analysé et décrit les différents composants de notre propre balance électronique avec bien sûr le traitement du programme du circuit réalisé par le logiciel Proteus. Et nous clôturons notre mémoire avec une conclusion générale et des perspectives.



Chapitre 1

**Généralités sur la mesure
du poids**

1. Introduction :

La mesure est l'action de déterminer la ou les valeurs numériques d'une grandeur (longueur, masse, etc.) par comparaison avec une grandeur constante de même espèce prise comme thème de référence (étalon ou unité). Dans ce chapitre on s'intéresse à étudier la mesure du poids qu'elle se fait par deux types d'instruments :

- **La balance mécanique :**

Cette balance fonctionne par comparaison de masse et de poids connu avec la masse d'une substance de poids inconnu. Une masse inconnue est déposée sur le plateau de pesée. La pesée se fait en enlevant des masses connues du contrepoids à l'aide d'un système mécanique de came jusqu'à ce que l'équilibre soit atteint. Elle a l'avantage d'être résistante et durable.

- **La balance électronique :**

La balance électronique est aujourd'hui le standard sur le marché de la balance. Elle fonctionne selon le principe suivant : une masse inconnue placée sur le plateau de la balance entraîne le déplacement ou la torsion d'un élément mécanique. L'équilibre initial est rétabli par une force électromagnétique proportionnelle à la masse de l'objet à peser dans le premier cas, dans le deuxième une mesure du déséquilibre causé par la torsion de l'élément mécanique permet de déterminer le poids.

La balance électronique bénéficie, contrairement à la balance mécanique, d'une précision particulièrement élevée.

De plus la balance électronique permet l'addition et l'ajout de fonctions supplémentaires à l'action de peser simplement.

1.1 Généralités sur les balances :

1.1.1 Définitions :

Le terme **balance** prend son origine du latin bis (deux fois) et lanx (plateau), est un instrument de mesure qui sert à évaluer des masses par comparaison avec des poids dont les masses sont connues.

Un capteur est un dispositif transformant l'état d'une grandeur physique observée en une grandeur utilisable généralement électrique. Pour le capteur de force il est conçu pour convertir une force « poids » appliquée sur un objet en un signal électrique.

Le pesage en tant que discipline, peut être défini comme la pratique et la connaissance de l'ensemble des techniques qui permettent de mesurer, imprimer et communiquer la valeur de la masse d'un objet ou d'un produit dans le cadre d'un usage particulier.

La masse est une propriété physique fondamentale d'un corps. Elle rend compte, dans une bonne approximation, de la quantité de matière contenue dans ce corps indépendamment de l'endroit où il se trouve. Elle est exprimée en kilogramme (Kg).

Le poids est la mesure de la force que la pesanteur exerce sur la masse d'un corps. Il est défini comme le produit de la masse par la force de gravitation, c'est donc une force exprimée en Newton (N).

La tare en physique est une masse de référence permettant de déduire la masse d'un objet.

1.1.2 Historique de la mesure du poids :

La balance est passée par plusieurs générations comme suit :

1.1.2.1 Balance mécanique :

a) Balance romaine :

• Principe :

Balance à fléau simple, à bras inégaux où la charge est équilibrée par le déplacement d'un poids curseur le long d'un bras gradué (en coches équidistantes) et pouvant être à une ou deux portées (dites alors "fort" et "faible"). Dans ce dernier cas, les romaines sont pourvues de deux appareillages de suspension (positionnés selon des bras de levier différents) et la permutation se fait par retournement du fléau, le crochet de charge pouvant passer d'un côté à l'autre en enfourchant la tête de la balance. [1]

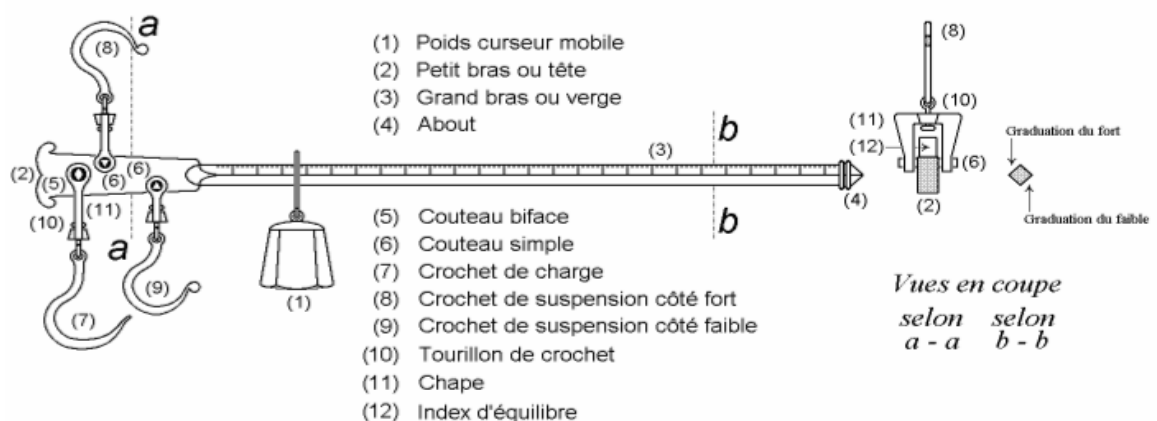


Figure1.1 : les différentes parties de la balance romaine [1]



Figure1.2 : Romaine savoyarde ou piémontaise en fer forgé [1]

b) Balance à ressort :

• Principe :

De tous les instruments de pesage présentés ici, ce sont les seuls à ne pas comparer une charge avec une ou plusieurs masses étalonnées mais à compenser l'action de la pesanteur terrestre sur cette masse à déterminer, par la déformation élastique d'un ressort.

Le déplacement d'un index devant une graduation permet de lire directement la valeur obtenue.

Rapides (peu ou pas d'oscillations), pratiques (faible encombrement, pas de poids autonomes), ces instruments sont peu précis et leurs performances varient dans le temps et selon le lieu (gravité terrestre).

Même si l'appellation "dynamomètre" est correcte pour les instruments de pesage à ressort, leur dénomination courante est "peson", celle de "dynamomètre" étant en pratique réservée aux instruments de même principe (et gradués avec les mêmes unités) mais destinés à la mesure de forces de traction ou de compression à l'exception de celle induite par la gravité terrestre. [1]



Figure1.3 : Pesons cylindriques [1]



Figure1.4 : Peson triangulaire [1]



Figure1.5 : Peson demi-lune [1]



Figure1.6 : Peson à cadran annulaire [1]

c) Les bascules décimales :

- **Principe :**

Une bascule à rapport est une balance de forte portée à fléau et à bras inégaux, équipée d'un petit plateau suspendu pour les poids et d'un plateau de grande taille pour la charge, souvent appelé "tablier".

Ce plateau de charge repose sur deux leviers de forme triangulaire (l'embase du plateau constitue l'un d'entre eux) qui sont combinés de façon à assurer aux points d'appui du plateau (deux contacts avec un levier, un avec l'autre) un déplacement vertical identique mais de très faible amplitude.

Selon que la bascule est à rapport décimal ou centésimal, la distance entre le pivot et le point de suspension des poids vaut dix ou cent fois celle entre le pivot et le point de suspension "arrière" du plateau de charge.

Jusqu'au 31 décembre 1839, il était permis d'utiliser des poids spéciaux, de masse égale au dixième (bascules décimales) ou du centième (bascules centésimales) de la valeur nominale qu'ils représentaient à condition de porter mention à la fois de la masse réelle et de la valeur nominale.

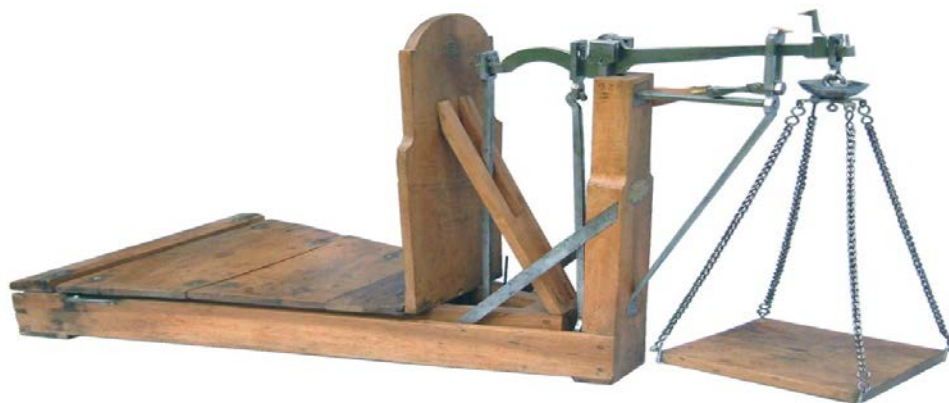


Figure1.7 : Bascule décimale [1]

d) Balance de ROBERVAL :

- **Principe :**

Fléau principal à bras égaux, relié à un levier annexe situé en dessous et appelé "contre-fléau", l'ensemble formant un parallélogramme.

Chaque bassin est supporté par une tige verticale reposant via une traverse sur les couteaux de l'extrémité du fléau et reliée à une extrémité du contre-fléau.



Figure1.8 : Balance Roberval de grand luxe, signée Béranger & Cie [1]



Figure1.9 : Grande balance de Roberval, de marque Frey & Studer à Strasbourg [1]



Figure1.10 : Roberval de luxe datable vers 1870-1880.

1.1.2.2 Balance électronique :

Depuis les années 2000, on peut dire que le pesage mécanique a pratiquement disparu. Il a été remplacé par le pesage dit tout électronique.

Dans le domaine du pesage industriel, il existe plusieurs types de capteur :

a. Capteur à compensation électromagnétique des forces :

- **Principe :**

Le principe est basé sur la transformation électrodynamique. Le courant est conduit par une bobine placée entre les deux pôles d'un aimant permanent. Au repos, le système est équilibré. Si le système est déséquilibré par une force externe, il est remis à l'équilibre par une régulation du courant de la bobine.

La construction en deux dimensions assure une résistance contre les coups et les torsions. La résistance mécanique du capteur est augmentée en utilisant des leviers et une protection intégrée. [2]

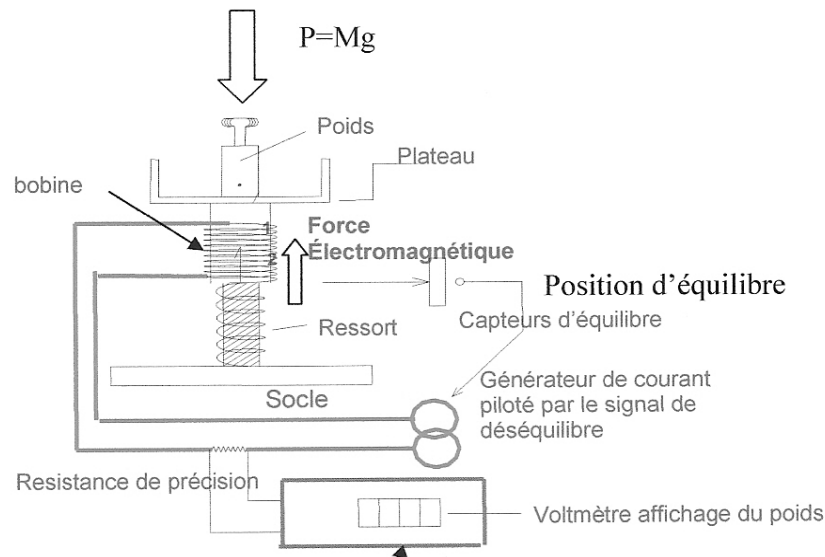


Figure1.11 : Principe de fonctionnement d'une balance à compensation électromagnétique [2]

• **Caractéristiques principales :**

- ✓ Haute résolution.
- ✓ Bonnes performances métrologiques.
- ✓ Réduction de force par leviers.
- ✓ Sophistiquée mais relativement chère.

b. Capteurs de force hydrauliques :

Les capteurs de force hydrauliques utilisent un agencement classique de piston et de cylindre pour transmettre une variation de pression due au mouvement du piston et un agencement de diaphragme qui produit un changement de pression sur un tube de Bourdon connecté aux capteurs de charge. [2]

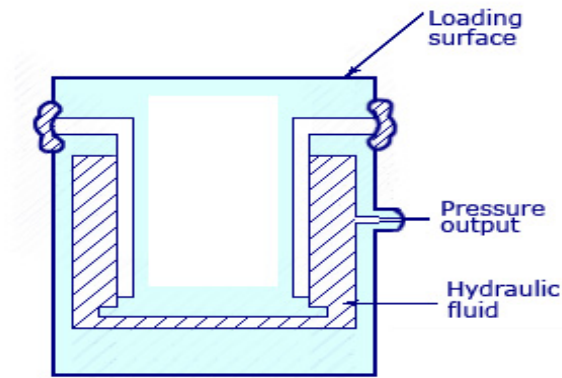


Figure1.12 : Principe du capteur hydraulique [3]

c. Capteurs pneumatiques :

Les capteurs pneumatiques utilisent une pression d'air appliquée à l'une des extrémités d'un diaphragme. Cette dernière s'échappe par la buse située au bas du capteur, qui comporte un manomètre à l'intérieur de la cellule.

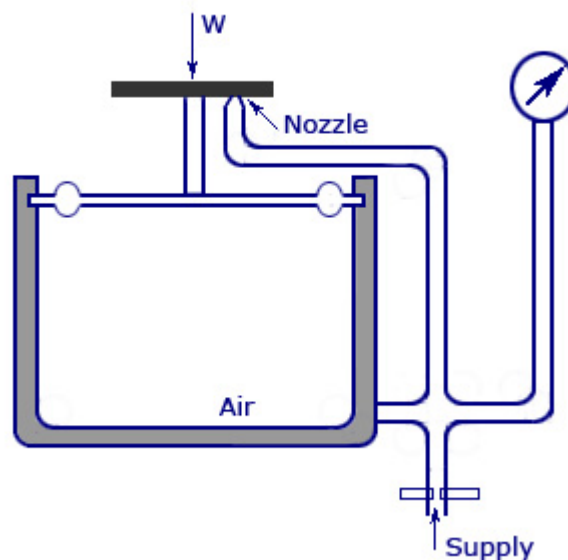


Figure1.13 : Principe du capteur pneumatique [3]

d. Capteur à jauge de contrainte :

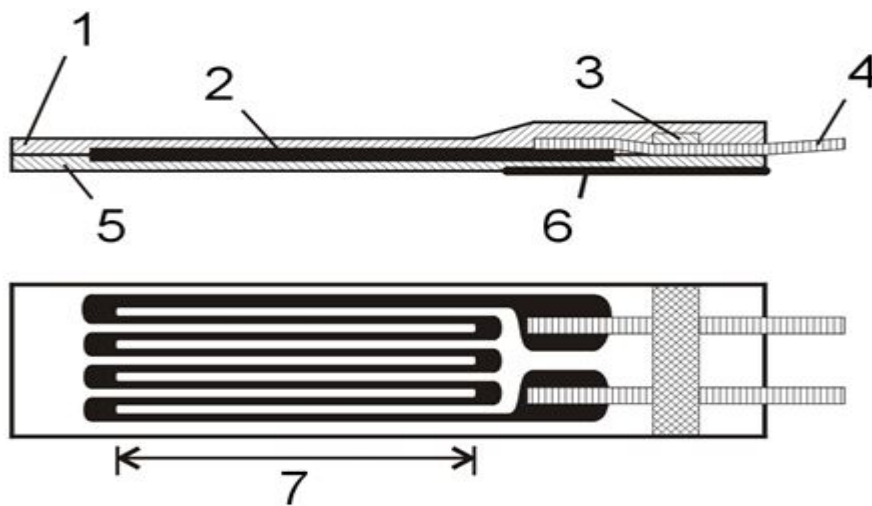
L'origine des jauges à résistance électrique trouve sa source dans une série d'expérience conduites par William Thomson au sujet des propriétés électromécaniques des métaux. Une de ses découvertes (1856) était que la résistance électrique de certains fils varie lorsqu'ils sont étirés. Depuis ce phénomène a été quantifié comme étant le facteur de jauge qui est donné par

$$\text{la relation suivante } \mathbf{k} = \frac{\Delta R}{R} \times \frac{1}{\epsilon} \text{ pour que } \epsilon = \frac{\Delta L}{L}$$

Le développement des premières jauges en Amérique du Nord remonte aux années 1930. Elles furent fabriquées par Simmons et Ruge dans leur laboratoire du MIT. Ces premières jauges consistaient en un fil enroulé autour d'un support de papier.

En 1952, les anglais Sanders and Roe introduisirent les jauges à trame pelliculaire qui sont fabriquées par un procédé de photogravure. Depuis lors, c'est devenu la technique expérimentale d'analyse des contraintes la plus populaire. [2]

Ce capteur qui représente plus de 80 % du marché sera étudié en détail dans le chapitre 2.1.1 car c'est le capteur qui sera utilisé par notre réalisation pratique.



- 1 Couverture
- 2 Grille de mesure
- 3 Fixation des bandes de raccordement
- 4 Bandes de raccordement
- 5 Support
- 6 Raidisseur de la base
- 7 Longueur de la grille de mesure

Figure1.14 : Structure d'une jauge de contrainte [2]

• **Caractéristiques principales :**

- Technologie simple, largement éprouvée.
- Faible coût de fabrication, basse consommation de courant.
- Apte pour les balances extra-plates.

- Existe en multiples versions : capteurs de flexion, capteurs annulaires de charges...

1.1.3 La précision :

Selon la plus petite masse qu'elles peuvent mesurer, les balances électroniques sont classées en fonction de la plus petite division et de la capacité typique :

- ❖ Balance analytique :
 - ◆ Ultramicroanalytique (0,1 μg / 3 g)
 - ◆ Microanalytique (0,001 mg / 3 g)
 - ◆ Semimicroanalytique (0,01 mg / 30 g)
 - ◆ Macroanalytique (0,1 mg / 160 g)
- ❖ Balance de précision (1 mg / 160 g - 60 kg) [4]

1.1.4 Unité de mesure :

Une balance mesure le poids pas la masse et il strictement parlant, incorrect d'afficher un résultat en kilogramme. Mais étant donné que pour que la plupart des applications la force de gravitation peut être considérée comme constante partout sur terre (9.80665 m. s⁻²), la communauté scientifique à autoriser d'exprimer le poids en kilogramme.

Lors de la 26ème réunion de la Conférence générale des poids et mesures, prévue le 16 novembre 2018, la définition du kilogramme a été remplacé par une constante universelle de la physique qui elle est immatérielle.

Pour plus de précision, les métrologistes ont défini le kilogramme par la constante de Planck, introduite en 1900 par Max Planck, père de la physique quantique, pour caractériser les échanges d'énergie entre la matière et le rayonnement.

La constante de Planck vaut $6,626069934 \times 10^{-34} \text{ kg.m}^2.\text{s}^{-1}$ à 13×10^{-9} près.

1.2 Conclusion :

Dans ce chapitre, nous avons présenté brièvement l'historique des balances de mesures de poids depuis l'antiquité jusqu'aux nouvelles techniques de pesage qui se sont modernisées de plus en plus.

Par la suite dans notre travail on va détailler plus la balance électronique à base de jauge de contrainte.



Chapitre 2

Les différents éléments
d'une balance électronique

2. Introduction :

Dans ce chapitre nous allons présenter les différentes parties qui constituent une balance électronique commerciale en se basant sur ce schéma bloc :

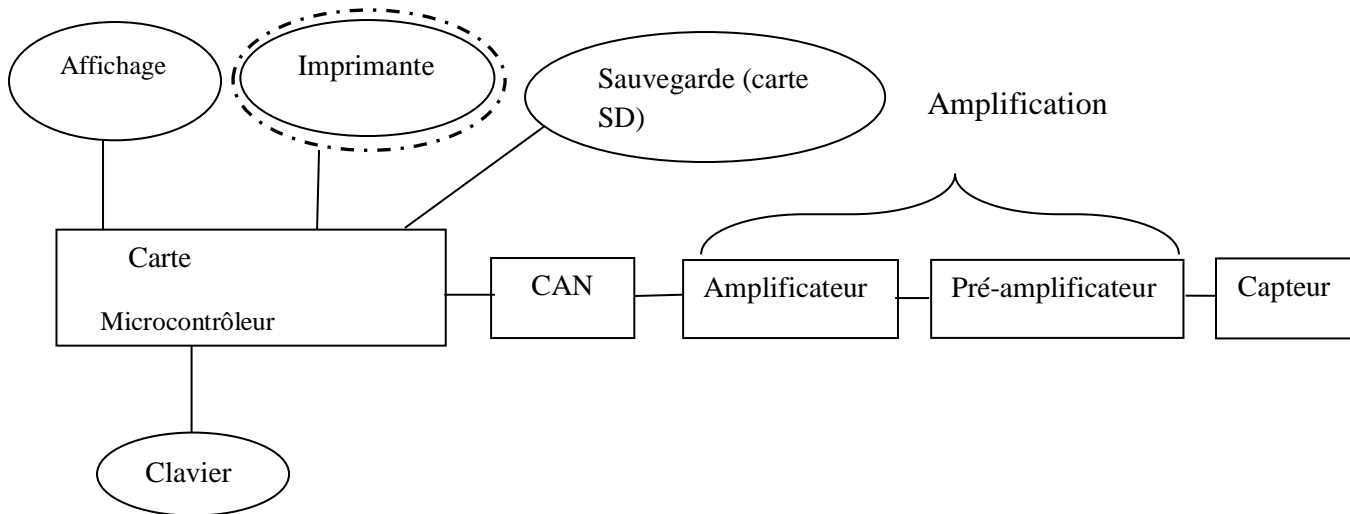


Figure2.1 : Schéma bloc d'une balance électronique commerciale

Dans un premier temps la sortie analogique du capteur est amplifiée en deux étapes, vu que le signal est très faible nous utilisons d'abord un pré-amplificateur et ensuite un amplificateur. Par la suite cette tension est convertie en un signal numérique à l'aide d'un CAN de bonne résolution pour une meilleure précision de notre balance. Nous faisons entrer ce signal numérique dans une carte à microcontrôleur pour pouvoir afficher le poids, le sauvegarder sur une mémoire flash ou bien imprimer un ticket de caisse si c'est nécessaire.

2.1 Description des différents blocs :

2.1.1 Capteurs de force à jauge de contrainte :

La jauge de contrainte est un capteur dont la résistance varie en fonction de sa déformation. Elle permet la mesure indirecte d'une force, d'une pression, d'un poids, etc... par déformation du corps d'épreuve sur lequel elle est solidaire.

2.1.1.1 Constitution :

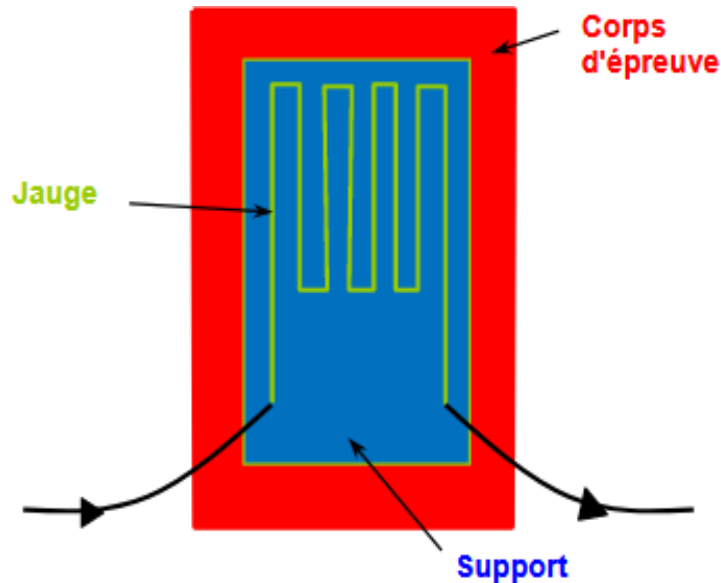


Figure2.2 : Structure interne du capteur [4]

➤ **La jauge** : Un petit élément résistif qui est collé sur une pièce au point où on veut mesurer la déformation. Cet élément est fait d'un fil fin enroulé selon une direction préférentielle et collé à la pièce par l'entremise d'un support d'isolation. Lorsque la pièce est soumise au chargement, sa déformation est transmise à travers la colle et le support à la jauge. Un changement proportionnel de la résistance en résulte. [4]

➤ **Le support** : Le support fait le lien entre le corps d'épreuve et la pièce déformée. Il doit donc répondre à des caractéristiques bien spécifiques : déformation facile, bonne aptitude au collage et un coefficient de variation relativement faible. On peut ici utiliser des résines époxyde ou des polyamides. Une étape de préparation de surface est nécessaire avant de procéder au collage. Celle-ci permet de maximiser l'accroche qu'aura la colle sur le matériau en augmentant les défauts de régularité sur la surface du matériau, ce dans le but d'éviter d'arracher la jauge par cisaillement lors de l'essai. [4]

➤ **Le corps d'épreuve** : Le corps d'épreuve est la partie qui subira les déformations. Il est donc préférable d'utiliser un matériau facilement déformable afin d'obtenir un signal de forte amplitude. Il faut également éviter de sortir de la gamme de déformation élastique de celui-ci pour éviter tout risque de déformation permanente. [4]

2.1.1.2 Principe de pesage par la jauge de contrainte :

Si le cylindre en acier est comprimé, il raccourcit tout en s'élargissant. Quand il est tiré en plus de s'allonger il s'amincit. L'importance de ces changements dépend de la masse même de l'acier. Il est évident que si le corps d'épreuve est une masse importante il va falloir des forces plus élevées pour pouvoir le comprimer comme une masse plus fine. Cette notion est essentielle lorsqu'on construit des capteurs de force pour différents objectifs de mesure : pour de faibles forces nominales on utilise des petits capteurs à l'inverse pour des forces plus grandes les capteurs seront plus gros. C'est pourquoi la caractéristique force nominale définit la charge maximum prévue pour le capteur.

Un capteur de force contient généralement quatre jauges, reliés entre-elles dans un montage que l'on nomme pont de Wheatstone, que nous allons expliquer plus en détail par la suite. Ce qui est très important est que les jauges soient correctement collées sur le corps d'épreuve afin de subir les mêmes déformations que lui. Quand l'acier se déforme, la résistance de la jauge de contrainte varie, comme mentionnée ci-dessus. Ainsi, le signal de sortie du pont de jauges délivre des informations sur l'importance de cette déformation. À partir de là, nous pouvons calculer la force agissant sur les jauges. C'est ainsi que le capteur de force fonctionne.

D'un point de vue mathématique, il est intéressant de voir que le capteur de force fonctionne seulement selon le principe de rapports linéaires. Par conséquent, la force est proportionnelle à l'effort mécanique (σ). Ce σ est proportionnel à la contrainte. Le changement relatif de résistance dépend proportionnellement de la contrainte. En conclusion, le signal de sortie du pont de Wheatstone est linéairement proportionnel à la variation relative de la résistance des jauges. [5]

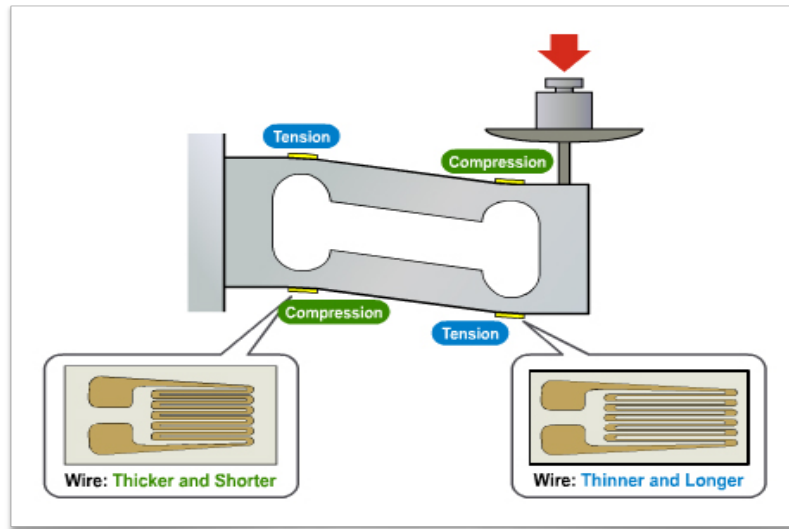


Figure2.3 : Diagramme détaillé des jauges de contrainte sur les capteurs de force à barres lorsque la force est appliquée [5]

2.1.1.3 Montages et choix des jauges de contrainte :

En fonction du type de matériau du corps d'épreuve, le concepteur choisit les jauges de contraintes adaptées à son application parmi les différents modèles de jauge de contrainte par les principaux fabricants de jauges qui sont : HBM et VISHAY.

La jauge uniaxiale est utilisée dans les capteurs de flexion, la jauge biaxiale dans les capteurs de compression et la jauge de cisaillement dans les capteurs de cisaillement. [6]

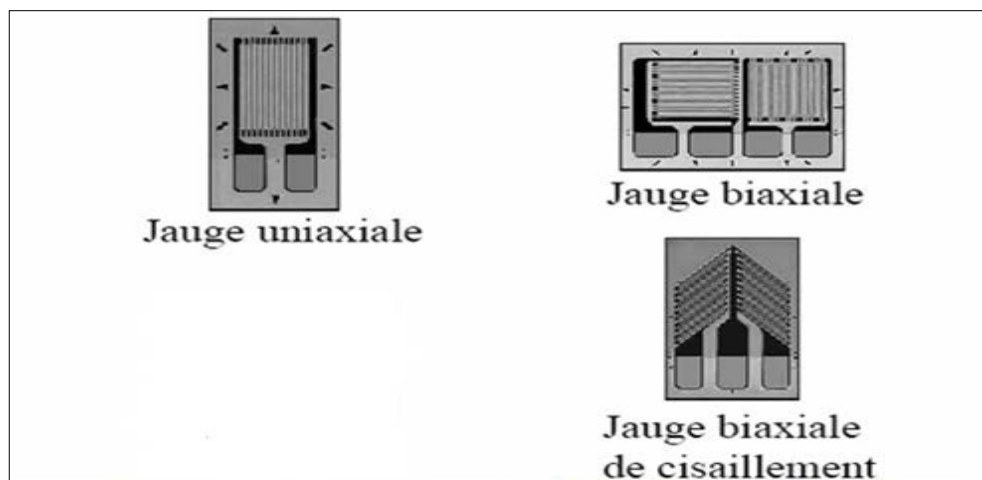


Figure2.4 : les différents modèles de jauge de contrainte [6]

Chapitre2 : Les différents éléments d'une balance électronique

Ces jauges sont représentées par les résistances appelées J1, J2, J3, J4 (généralement au nombre de 4) sont collées sur le corps d'épreuve puis câblées en pont de Wheatstone.

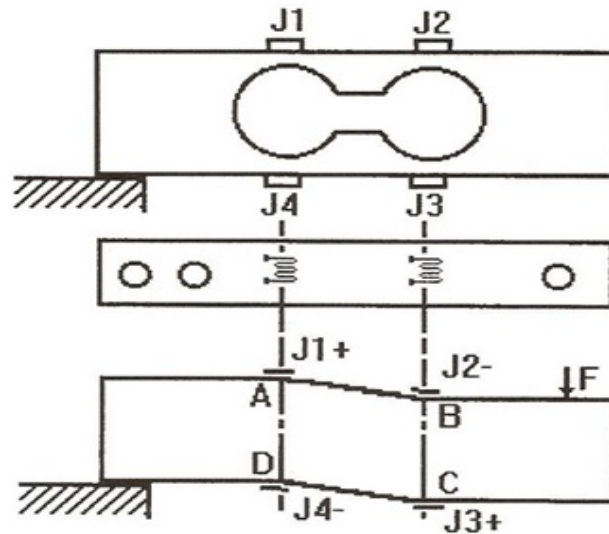


Figure2.5 : Déformation du corps d'épreuve [6]

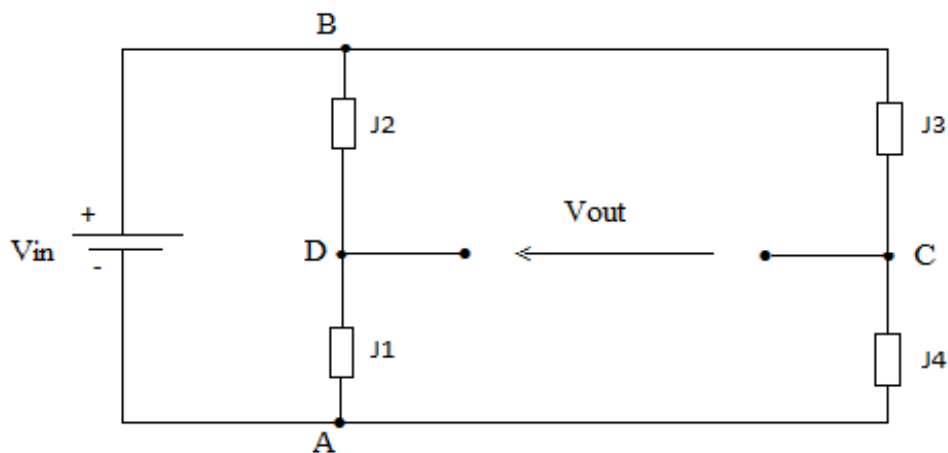


Figure2.6 : Schéma électrique simplifié du capteur

2.1.1.4 Conditionnement des jauges :

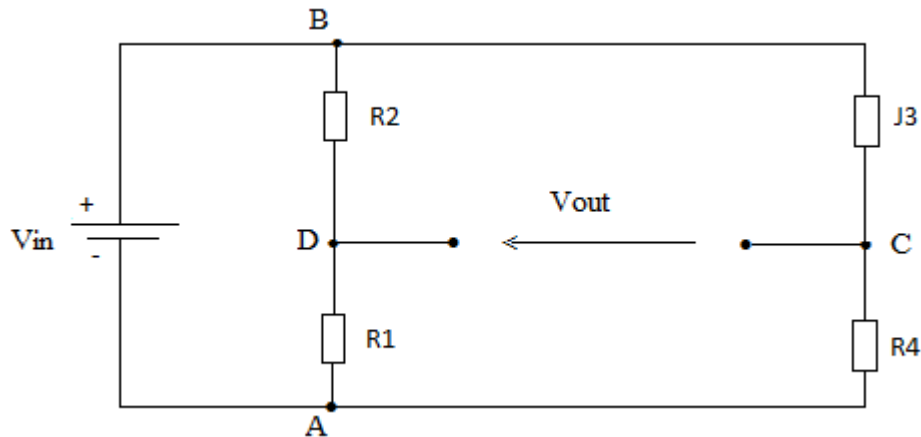
Les variations de résistances liées à la petite déformation du conducteur dans une jauge de contraintes sont très faibles. Le meilleur système pour en déduire une force appliquée sur la cellule est de mesurer la variation relative de résistance et de la convertir en une différence de potentielle proportionnelle à cette variation, mesurée par un potentiomètre.

Le circuit le plus adapté pour cette mesure est le pont de Wheatstone. [7]

2.1.1.5 Le pont de Wheatstone :

a) Montage quart de pont :

Dans le montage en quart de pont, on ne dispose que d'une jauge, trois résistances viennent en



complément.

Figure2.7 : le montage quart de pont

Ce montage est le plus simple et le moins cher, mais il représente quelques inconvénients :

- ✓ La jauge étant éloignée des autres résistances, il faut prendre en compte la résistance des fils de liaison. Celle-ci dépend de la température ambiante, d'où une mesure imprécise.
- ✓ Le signal sortant de la jauge est détérioré à cause de la résistance des fils. [7]

b) Montage demi pont :

Chapitre2 : Les différents éléments d'une balance électronique

Pont de Wheatstone composé de 2 jauges et 2 résistances. Il est plus sensible que le montage en quart de pont, par contre il est plus cher. [7]

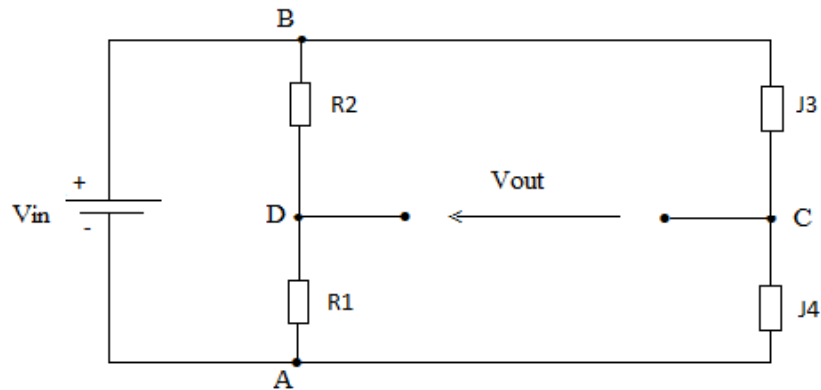


Figure2.8 : le montage demi-pont

c) Montage complet :

Le pont complet de Wheatstone est composé de 4 jauges placées sur le corps d'épreuve. Il est plus précis que le montage en demi pont et encore plus que celui en quart de pont (4 fois plus sensible). Mais c'est le plus cher des 3 montages. [7]

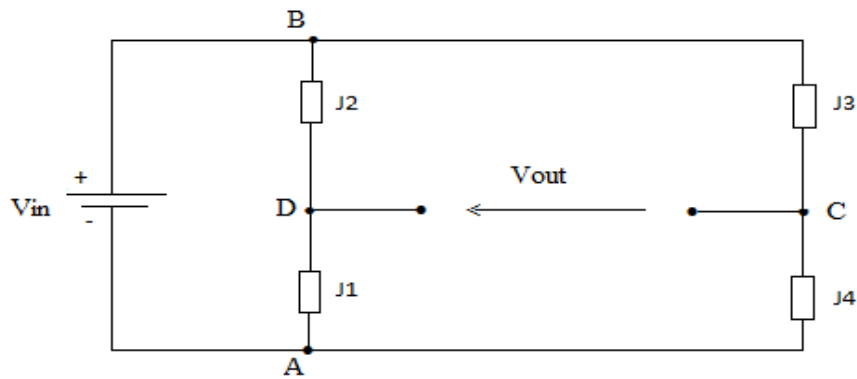


Figure2.9 : le pont complet

2.1.1.6 Principe de fonctionnement du pont de Wheatstone :

Le pont de Wheatstone est constitué de quatre résistances disposées en quadrilatère. Le pont fonctionne selon le principe représenté dans la figure ci-dessous.

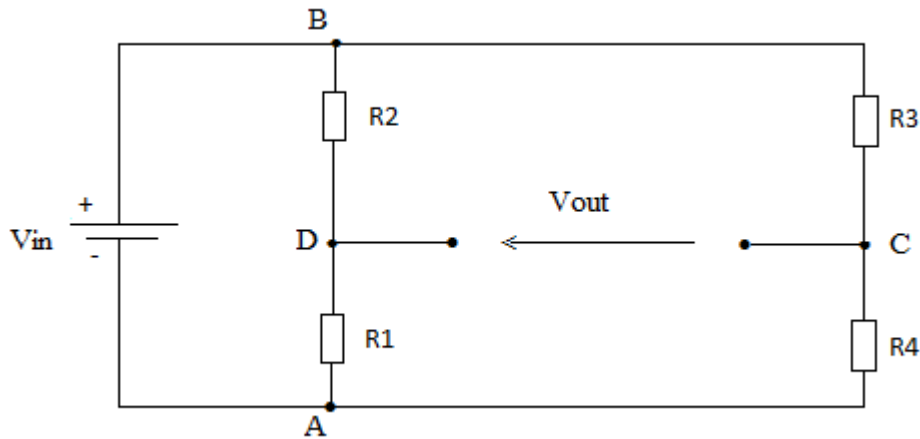


Figure2.10 : Principe de fonctionnement du pont de Wheatstone

Ce qui caractérise le montage est le pont entre les points C et D, contenant un ampèremètre qui détecte le courant lorsque A et B sont reliés à une source de tension ou de courant.

Le pont est un moyen très précis de mesurer des résistances. [7]

2.1.1.7 Equilibrage du pont :

En appliquant le théorème de Millman en C et en D, on a :

$$V_C = \frac{\frac{V_A}{R_4} + \frac{V_B}{R_3}}{\frac{1}{R_4} + \frac{1}{R_3}} = \frac{V_A R_3 + V_B R_4}{R_3 + R_4}$$

$$V_D = \frac{\frac{V_A}{R_1} + \frac{V_B}{R_2}}{\frac{1}{R_1} + \frac{1}{R_2}} = \frac{V_A R_2 + V_B R_1}{R_2 + R_1}$$

Avec :

$$V_{in} = V_B - V_A \quad \text{et} \quad V_{out} = V_D - V_C$$

On a l'équation du pont :

$$V_{out} = V_{in} \frac{(R_1 R_3 - R_2 R_4)}{(R_1 + R_2)(R_3 + R_4)}$$

On constate que $V_{out} = 0$ quand $(R_1 R_2 - R_2 R_4) = 0$. On dit alors que le pont est à l'équilibre.

a. Quart de pont (pont simple) :

On a initialement $R_1 = R_2 = R_3 = R_4 = R$ avec J_1 ou J_3 varie de $+dR$. [7]

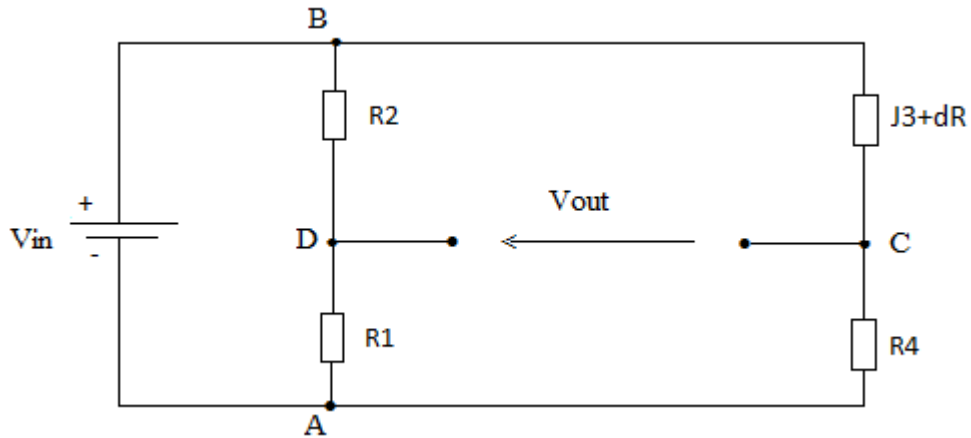


Figure2.11 : Schéma de principe du montage pont simple

L'équation générale devient alors :

$$V_{out} = \frac{V_{in} dR}{4R}$$

b. Demi-pont :

Le pont est initialement équilibré mais J_3 varie de $+dR$ et J_4 de $-dR$. [7]

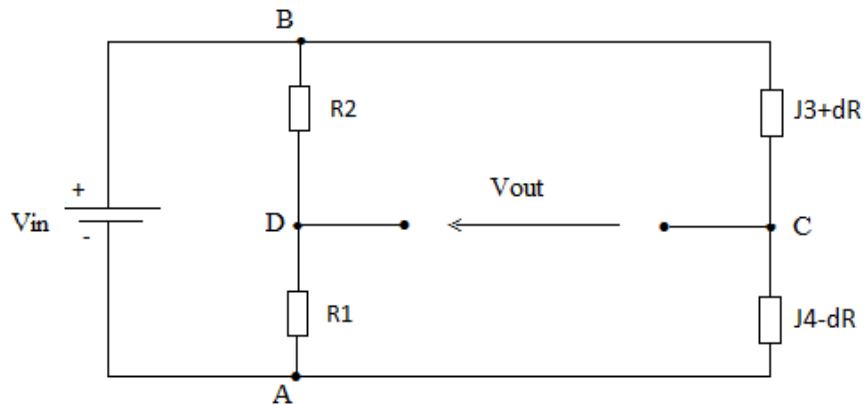


Figure2.12 : Schéma de principe du montage demi-pont

On trouve :

$$V_{out} = \frac{V_{in} dR}{2R}$$

c. Pont complet :

Le pont est initialement équilibré mais J_1 et J_3 varie de $+dR$ et J_2, J_4 de $-dR$ et $J_3=J_4=J_2=J_1=R$. [7]

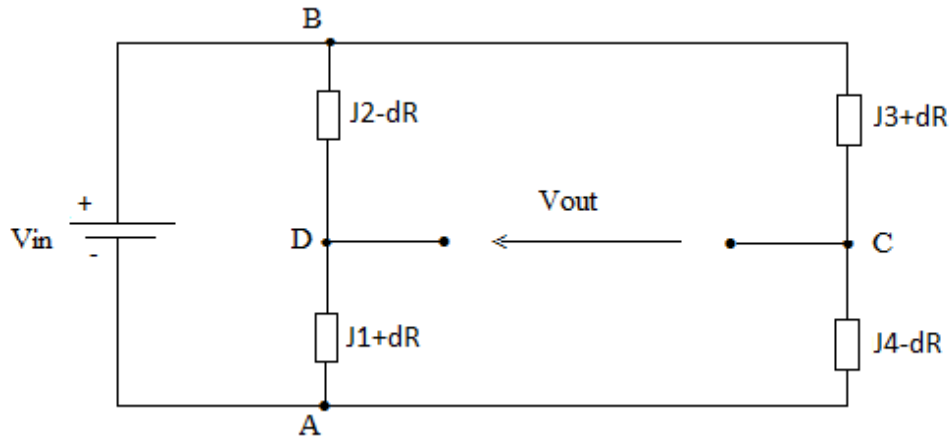


Figure2.13 : Schéma de principe du montage pont complet

Donc l'équation de base devient alors :

$$V_{out} = \frac{V_{in} dR}{2R}$$

La tension de mesure est dans ce cas une fonction rigoureusement linéaire des variations de résistance.

Le pont de Wheatstone réalise l'auto compensation des termes quadratique et par conséquent présente une réponse parfaitement linéaire.[7]

2.1.2 Amplification :

Dans une balance électronique à base de jauge de contrainte le signal délivré à la sortie de celle-ci est très petit c'est pourquoi on ajoute un bloc d'amplification pour augmenter le signal pour qu'on puisse lire facilement les changements de la mesure.

a) Le préamplificateur :

Un préamplificateur ou préamplificateur de commande est un amplificateur électronique qui reçoit et adapte un signal avant de le transmettre à l'amplificateur principal.

Un préamplificateur joue le rôle du premier étage d'amplification situé au plus près de la source de signal. Conformément au théorème de « Friis », c'est lui qui est le garant d'un bon rapport signal sur bruit en sortie du système puisqu'il doit réussir à extraire un très faible

Chapitre2 : Les différents éléments d'une balance électronique

signal électrique. Grâce à sa proximité à la source du signal, il permet de limiter les dégradations de celui-ci par des interférences parasites ou par son atténuation lors du transport.

Les préamplificateurs se rencontrent donc dans toutes les applications où l'on désire mesurer électriquement un signal physique en aval d'un transducteur quelconque :

- ✓ Applications audio : microphones, électrophone.
- ✓ Applications radiofréquences : antennes.
- ✓ Autres applications : photodétecteurs, détecteur infrarouge, instruments scientifiques, etc... [8]

b) L'amplificateur :

Le type d'amplificateur utilisé dans les balances c'est l'amplificateur de mesure ou bien plus précisément l'amplificateur d'instrumentation.

L'amplificateur de mesure est un élément essentiel dans la partie de conditionnement d'une chaîne d'acquisition : il permet le traitement des signaux issus de capteurs de mesure.

Il est généralement réalisé à partir d'un ou de plusieurs amplificateurs opérationnels (AOP), de telle manière qu'il améliore leurs caractéristiques intrinsèques : composante continue, dérive, bruit d'amplification, gain en boucle ouverte, taux de réjection du mode commun, impédance d'entrée. [8]

Parmi les amplificateurs d'instrumentations en circuits intégrés, on trouve le INA125P qui est un circuit permet d'amplifier de très faibles signaux électriques qui ne sont pas assez puissants pour être directement compatible avec notre microcontrôleur (Arduino, etc..) pour les mesurer par conversion analogique / numérique.

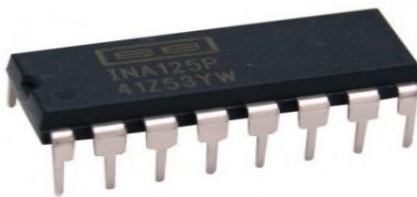


Figure2.14 : Circuit intégré INA125P

i) Quelques exemples d'utilisation du dispositif INA125P :

- Construction de balances de pesage à jauges de contraintes montées en pont.
- Amplificateur de ponts de mesure de température, de pression.
- Acquisition de données multi canaux.
- Instrumentation avec des thermocouples.

Cet amplificateur d'instrumentation de forte précision fournit l'alimentation du pont de mesure avec une tension de référence de précision. Il amplifie ensuite le signal de mesure différentiel avec un fort gain, ajustable de 4 à 10 000. [9]

ii) Caractéristiques :

- Circuit intégré INA125P, Texas Instruments.
- Boîtier DIP 16 (2x8 pins) à souder.
- Tension d'alimentation unique « 2,7 V à 36 V » ou double « $\pm 1,35$ V à ± 18 V ».
- Tension de référence très précise 0.5% max.
- Tension d'excitation ajustable pour utiliser différents types de capteurs à amplifier.
- Très faible courant de repos 460 μ A.
- Tension de référence de précision 1.24 V, 2.5 V, 5 V ou 10V.
- Faible tension d'offset 250 μ V max.
- Faible dérive en température 2 μ V/ $^{\circ}$ C max.
- Protection d'entrée 40V.
- Faible bruit.
- Le gain d'amplification se règle en ajoutant une résistance entre pin8 et pin9.
- Température de fonctionnement de -40 $^{\circ}$ C à 85 $^{\circ}$ C.
- Un mode sommeil permet d'économiser de l'énergie.
- Compatible avec Arduino en 5V. [9]

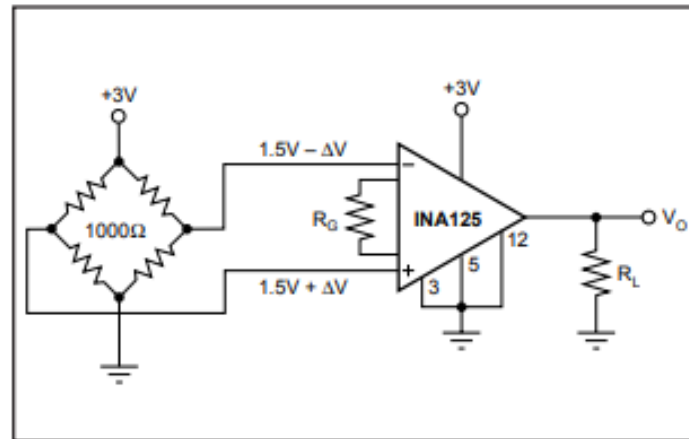


Figure2.15 : Amplificateur à point d'alimentation unique [9]

2.1.3 Convertisseur analogique numérique :

2.1.3.1 Principe de la conversion analogique numérique :

Un convertisseur analogique-numérique (CAN) est un dispositif électronique permettant la conversion d'un signal analogique en un signal numérique.

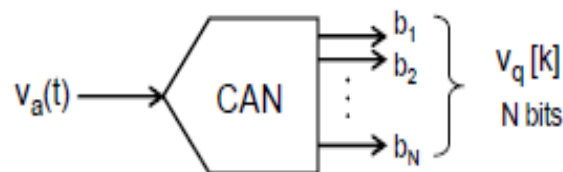


Figure2.16 : Symbole d'un CAN à N bits

Cette première définition pour être complète on appelle deux autres, celles des signaux analogiques et numériques :

- **Signal analogique** : signal continu en temps et en amplitude.
- **Signal numérique** : signal échantillonné et quantifié, discret en temps et en amplitude.

➤ Description :

Conceptuellement, la conversion analogique-numérique peut être divisée en trois étapes : **l'échantillonnage temporel, la quantification et le codage.**

La figure ci-dessous (**figure2.13**) présente successivement ces trois étapes pour un CAN dont la sortie du signal numérique est sur 3 bits :

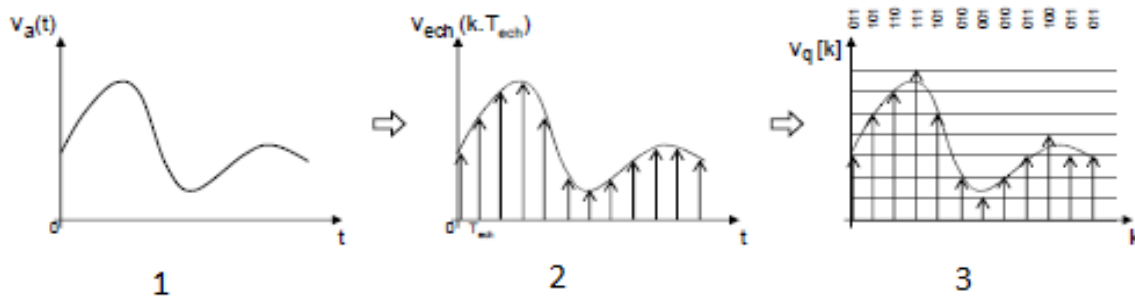


Figure2.17 : (1) signal analogique (2) signal échantillonné (3) quantifié

Un signal analogique, $v_a(t)$ continu en temps et en amplitude (1) est échantillonné à une période d'échantillonnage constante T_{ech} .

- On obtient alors un signal échantillonné $V_{ech}(k.T_{ech})$ discret en temps et continu en amplitude (2).
- Ce dernier est ensuite quantifié, on obtient alors un signal numérique $V_q(k)$ discret en temps et en amplitude (3). La quantification est liée à la résolution du CAN (son nombre de bits).
- La figure (3) présente également le code numérique sur trois bits (en code binaire naturel) associé à $V_q(k)$ en fonction du temps. [10]

2.1.3.2 Principales structures des C.A.N:

a) CAN à simple rampe :

On réalise au moyen d'un compteur et d'un convertisseur numérique-analogique une rampe de tension. Un comparateur arrête le compteur quand la tension créée par le CAN atteint la tension à convertir. Le compteur indique alors le résultat sur N bits, qui peut être stocké ou traité.

Ce type de convertisseur est simple, il est réservé aux applications pour lesquelles une dizaine de mesures par seconde est suffisante.

De plus, leur temps de conversion qui évolue avec la tension à convertir les rend totalement absents du monde de l'électronique. [11]

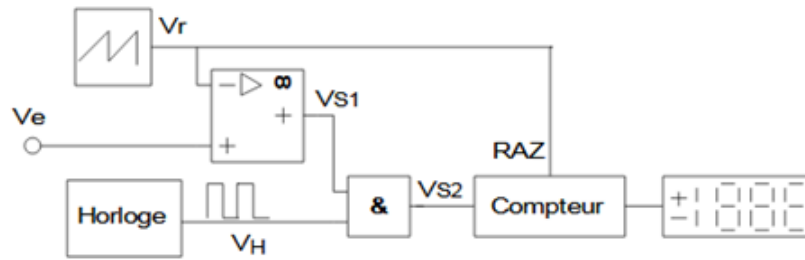


Figure2.18 : CAN simple rampe [11]

b) CAN à double rampe :

Cette évolution des convertisseurs à simple rampe sert à s'affranchir de la dérive naturelle des composants qui le composent. Son fonctionnement repose sur une comparaison entre une référence et le signal à convertir.

La conversion se fait en 3 étapes :

- On charge une capacité avec un courant proportionnel au signal à convertir pendant un temps fixe (le temps du comptage complet du compteur).
- On décharge ensuite la capacité, avec un courant constant issu de la tension de référence, jusqu'à annulation de la tension à ses limites. Quand la tension devient nulle, la valeur du compteur est le résultat de la conversion.
- On annule enfin la tension aux limites de la capacité par une série convergente de charge et de décharge. On parle généralement de phase de relaxation.

Ce type de convertisseur est précis, simple et peu onéreux, par contre le temps de conversion est très long. Souvent, ce convertisseur forme le circuit de base des voltmètres numériques. [11]

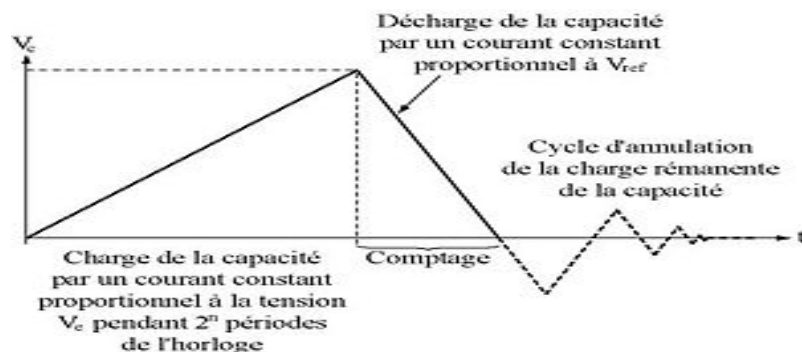


Figure2.19 : Fonctionnement du CAN à double rampe [11]

c) Convertisseur à approximations successives :

Chapitre2 : Les différents éléments d'une balance électronique

Ce type de convertisseur est très proche en termes de composition des CAN à simple rampe. Le CAN à approximations successives utilise un processus de dichotomie pour traduire numériquement une tension analogique.

Un séquenceur (généralement appelé SAR pour Successive Approximation Register) couplé à un CNA, génère une tension analogique qui est comparée au signal à convertir.

Le résultat de cette comparaison est alors introduit dans le SAR, qui va le prendre en compte, pour la suite de dichotomie, jusqu'à complétion.

Le convertisseur réalise par conséquent sa conversion en positionnant en premier le bit de poids fort (MSB) et en descendant progressivement jusqu'au LSB.

Les convertisseurs à approximations successives ont des temps de conversion de l'ordre de dizaine de microsecondes, pour des résolutions d'une douzaine de bits à peu près. [11]

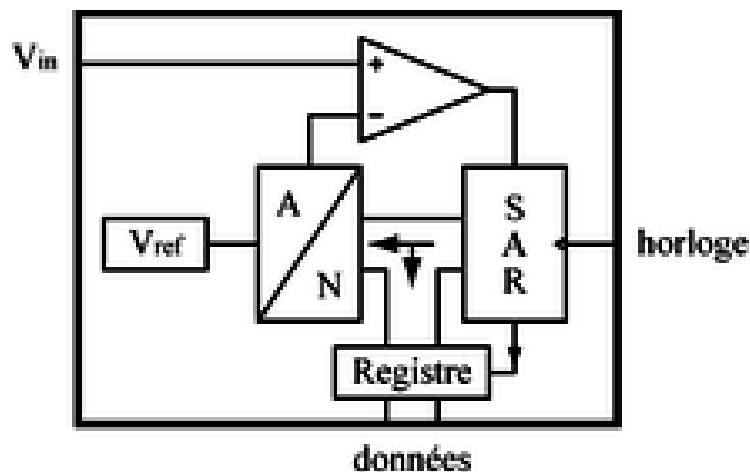


Figure2.20 : Schéma de principe d'un convertisseur à approximation successive [11]

d) Convertisseur parallèle (flash) :

Le principe est de générer $2^N - 1$ tensions analogiques au moyen d'un diviseur de tension à 2^N résistances. Les $2^N - 1$ tensions obtenues aux limites de chacune des résistances sont ensuite comparées dans $2^N - 1$ comparateurs au signal à convertir. Un bloc logique combinatoire relié à ces comparateurs donnera le résultat codé sur N bits en parallèle. Cette technique de conversion est particulièrement rapide, mais coûteuse en composants et par conséquent utilisée pour les applications critiques comme la vidéo.

Les CAN Flash ont des temps de conversion inférieurs à la microseconde mais une résolution assez faible (de l'ordre de 8 bits) et sont fréquemment très chers. [11]

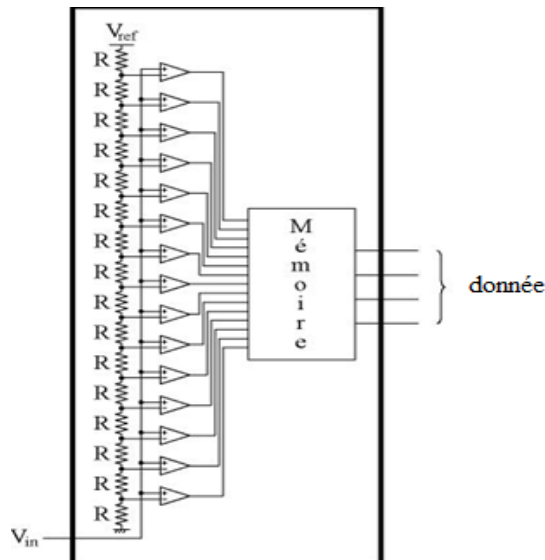


Figure 2.21 : Structure d'un convertisseur Flash [11]

2.1.3.3 Caractéristiques des C.A.N :

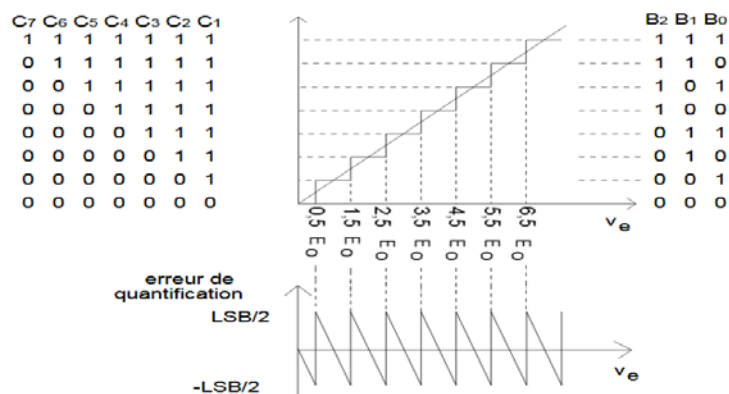
✚ Résolution :

C'est la tension d'entrée qui fait passer le nombre en sortie de zéro à un.

Pour les convertisseurs à sortie BCD, la résolution s'exprime en "digit" ou en points (ex : voltmètre 2000 points). [12]

✚ Erreur de quantification :

C'est la différence entre la réelle et l'idéale (droite passant par l'origine). [12]



✚ Erreur de facteur d'échelle :

Si pour obtenir le nombre maximum en sortie, la tension d'entrée doit être différente de V_o (tension pleine échelle), il y a erreur de facteur d'échelle, cette erreur s'exprime en fraction de LSB.

Chapitre2 : Les différents éléments d'une balance électronique

Ex : pour $V_o = 10V$ et un CAN de 8 bits : $q = 39mV$

L'erreur max : $1/2 \text{ LSB} = 19,5mV$. [12]

✚ Absence de code :

C'est l'équivalent de l'erreur de monotonie pour le C.N.A., une augmentation de q mV à l'entrée ne se traduisant pas par une augmentation du nombre en sortie. [12]

✚ Non linéarité :

Elles sont définies de la même façon que pour le C.N.A. Si $\Delta V_s > q$ il y a absence de code. [12]

✚ Tension de décalage :

Le constructeur indique sa valeur maximale. (Ex : 10LSB), cette tension de décalage peut être réglée mais elle est sensible à la température. [12]

✚ Impédance d'entrée :

Elle varie d'un CAN à l'autre : de quelques $k\Omega$ à $10^{12} \Omega$. [12]

2.1.4 La carte microcontrôleur :

2.1.4.1 Description de la carte :

Un microcontrôleur est défini comme une unité de traitement de l'information de type microprocesseur à laquelle on a ajouté des périphériques internes permettant de réaliser des montages sans nécessiter l'ajout de composants externes.

Sa structure est celle des systèmes à base de microprocesseurs. Il est composé des éléments suivants :

- ✓ Un microprocesseur (CPU : Central Processing Unit).
- ✓ Une horloge (avec éventuellement un quartz externe).
- ✓ De la mémoire (ROM, RAM, EEPROM, EPROM).

Chapitre2 : Les différents éléments d'une balance électronique

- ✓ Des interfaces parallèles pour la connexion des entrées et des sorties.
- ✓ Des interfaces séries (liaison RS232, liaison I2C).
- ✓ Convertisseur analogique-numérique.
- ✓ Convertisseur numérique-analogique.
- ✓ Compteur (timer).
- ✓ Chien de garde (watchdog).
- ✓ Bus (bus d'adresse, bus de données, bus de commande). [13]

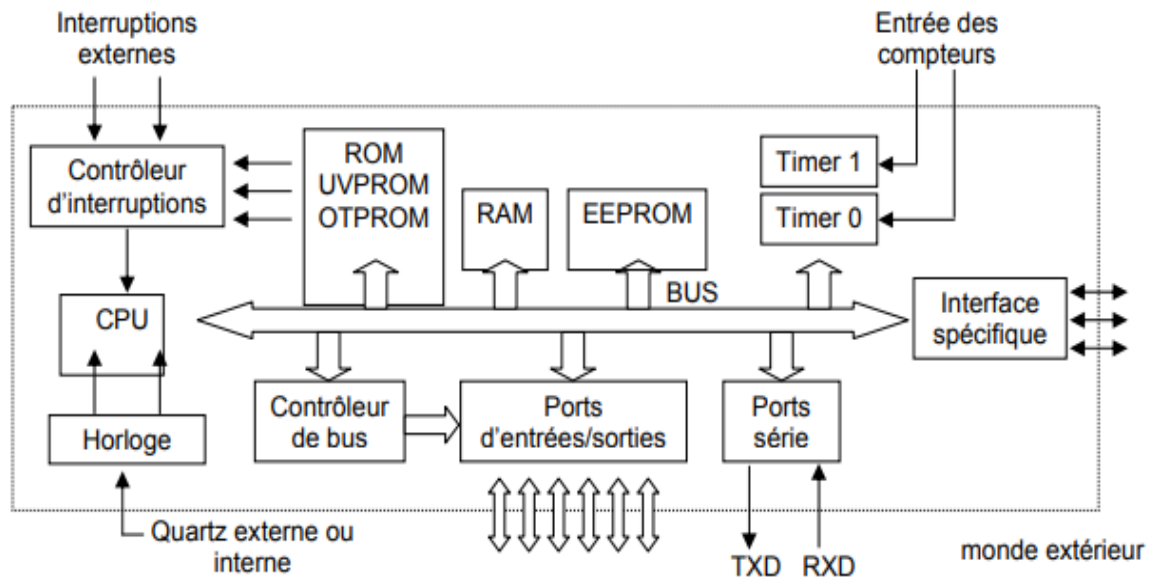


Figure2.22 : Architecture interne d'un microcontrôleur [13]

2.1.4.2 L'utilisation des microcontrôleurs :

Les microcontrôleurs sont aujourd'hui implantés dans la plupart des applications de grand public ou professionnelles. Voici quelques utilisations des microcontrôleurs :

- ✓ Télécommunications : cartes FAX et MODEM, téléphones portables (interfaces homme machine, gestion d'écrans graphiques) ...
- ✓ Industriels : automates programmables, contrôle de processus divers, supervision...
- ✓ Commercial : électroménager, domotique...
- ✓ Automobile : tableau de bord, contrôle des sièges, des vitres...
- ✓ Militaire et spatial : sonde, lanceurs de fusées, missile, robots... [13]

2.1.4.3 Avantages et inconvénients des microcontrôleurs :

Ci-dessous, on trouvera les principaux avantages et inconvénients des microcontrôleurs :

(+)

- Diminution de l'encombrement du matériel et du circuit imprimé.
- Simplification du tracé du circuit imprimé :
 - > Augmentation de la fiabilité du système.
 - > Nombre de composants.
 - > Connexions composants/supports et composant circuit imprimé.
- Intégration en technologie MOS, CMOS, ou HCMOS :
 - > diminution de la consommation.
- Le microcontrôleur contribue à réduire les coûts à plusieurs niveaux :
 - > Moins cher que les composants qu'il remplace.
 - > Diminution des coûts de main d'œuvre (conception et montage).
- Environnement de programmation et de simulation évolués. [13]

(-)

- Le microcontrôleur est souvent surdimensionné devant les besoins de l'application.
- Investissement dans les outils de développement.
- Écrire les programmes, les tester et tester leur mise en place sur le matériel qui entoure le microcontrôleur.
- Incompatibilité possible des outils de développement pour des microcontrôleurs de même marque.
- Les microcontrôleurs les plus intégrés et les moins coûteux sont ceux disposant de ROM programmables par masque. [13]

2.1.5 Affichage :

Un afficheur numérique est un instrument qui affiche un signal d'entrée, soit en forme numérique soit en graphique à barres. De nombreux afficheurs de panneaux numériques comprennent également des relais pour la commutation des alarmes, et la retransmission analogique de la valeur d'affichage, ainsi que la capacité de connecter et transférer des données vers un ordinateur. [14]

Les afficheurs les plus utilisés pour un système à microcontrôleurs sont :

✚ Afficheurs à LED :

Ce sont des dispositifs électroniques composés de diodes électroluminescentes qui permettent de diffuser de l'information (lettres, chiffres et autres caractères) en temps réel ou stockée. Les afficheurs à LED permettent un affichage nocturne car la source lumineuse de ce dispositif est interne.

Leurs domaines d'application sont très nombreux : industries, écoles, administrations, restauration, salles d'accueil, panneaux de destination des véhicules de transport public, etc....



Figure2.23 : Photo d'un afficheur à LED

✚ Afficheur LCD :

Les afficheurs à cristaux liquides, autrement appelés afficheurs LCD, sont des modules compacts intelligents et nécessitent peu de composants externes pour un bon fonctionnement.

Un afficheur LCD est capable d'afficher tous les caractères alphanumériques usuels et quelques symboles supplémentaires. Pour certains afficheurs, il est même possible de créer ses propres caractères. Chaque caractère est identifié par son code ASCII.



Figure2.24 : Photo d'un afficheur LCD

✚ OLED :

OLED est l'acronyme de Organic Light Emitting Diode qui signifie diode électroluminescente organique. C'est un composant électronique qui produit de la lumière à partir d'une substance organique dont la propriété est s'illuminer lorsqu'elle est traversée par un courant électrique.

Puisque nous allons utiliser cet afficheur dans notre réalisation pratique, plus de détails seront introduire dans le prochain chapitre.

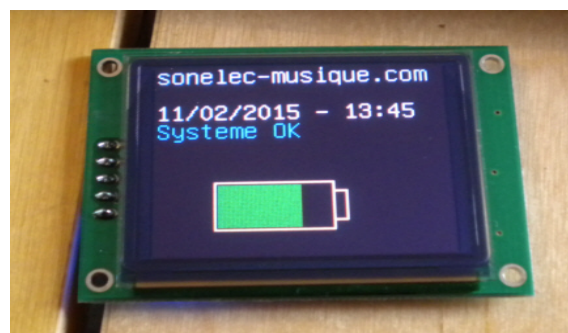


Figure2.25 : Photo afficheur OLED

2.1.6 Clavier :

Un clavier est très essentiel pour une balance commerciale. Il a l'objectif de faire beaucoup d'opérations :

- Allumer/Eteindre la balance.
- Rentrer des prix et enregistrer les ventes.
- Tarer, faire un total.
- Appel d'un prix mémorisé...

Chapitre2 : Les différents éléments d'une balance électronique

Un clavier matriciel est composé essentiellement de combinaisons de boutons poussoirs arranger de façon à former des lignes et des colonnes. De cette façon, le nombre de broches d'entrée/sortie nécessaires pour leur connexion à un microcontrôleur est réduit.

Les claviers sont largement utilisés comme dispositifs d'entrée/sortie dans beaucoup d'application dans notre vie quotidienne tel que les appareils comme le téléphone, le distributeur automatique de billets, la serrure électronique, la calculatrice, la minuterie... [15]



Figure 2.26 : Photo d'un clavier à 16 touches

2.1.7 Carte SD :

Une carte SD (Secure Digital) est une carte mémoire amovible de stockage de données numériques créée en janvier 2000 par une alliance formée entre les industriels Panasonic, SanDisk et Toshiba.

Ce genre de mémoire sont constitué à base de mémoire électronique qu'on appelle mémoire morte reprogrammable de type flash.

La mémoire flash est un type d'EEPROM qui permet la modification de plusieurs espaces mémoires en une seule opération. Elle est donc plus rapide lorsque le système doit écrire à plusieurs endroits en même temps.

La mémoire flash utilise comme cellule de base un transistor MOSFET possédant une grille flottante enfouie au milieu de l'oxyde de grille, entre le canal et la grille. L'information est stockée grâce au piégeage d'électrons dans cette grille flottante. Deux mécanismes sont utilisés pour faire traverser l'oxyde aux électrons :

- L'injection d'électrons chauds.

Chapitre2 : Les différents éléments d'une balance électronique

- L'effet tunnel obtenu en appliquant une haute tension sur la vraie grille (appelée grille de contrôle). [16]

En effet, la technique flash se décline sous deux principales formes : flash NOR et NAND.

✚ **Flash NOR** : Les temps d'effacement et d'écriture sont longs mais elle possède une interface d'adressage permettant un accès aléatoire et rapide à n'importe quelle position. Le stockage des données est 100 % garanti par le fabricant. Elle est adaptée à l'enregistrement de données informatiques destinées à être exécutées directement à partir de cette mémoire. Cette caractéristique est appelée XIP (eXecute In Place). [16]

✚ **Flash NAND** : La mémoire Flash NAND est en fait une mémoire de type EEPROM dans laquelle chaque cellule est un transistor à porte flottante capable de stocker un bit (mémoire dite SLC), deux bits (mémoire MLC) ou trois bits (mémoire TLC). Toutefois, la performance et la fiabilité ont tendance à diminuer avec le nombre de bit par cellule.

Les mémoires Flash sont considérées comme très fiables puisqu'elles supportent entre 10 000 et 100 000 cycles d'écriture/effacement selon la qualité de fabrication des cellules. [16]

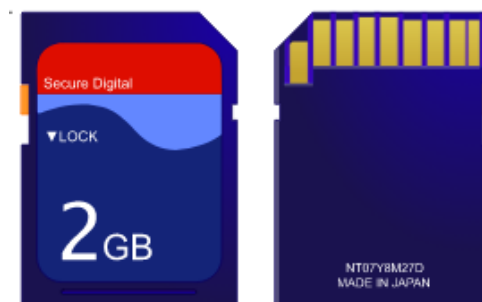


Figure2.27 : Support d'une carte SD

2.1.8 Imprimante :

Généralement pour les balances électroniques commerciales on utilise des imprimantes thermiques.

Ces derniers se composent d'une rangée de petites résistances électriques de la largeur du papier utilisé qui chauffent instantanément, sous le principe matriciel. En effet, le papier est

Chapitre2 : Les différents éléments d'une balance électronique

sensible à la chaleur. C'est donc en chauffant ce papier aux endroits voulus que le texte apparaisse.

Ce type d'imprimantes a pour avantage d'être très rapide, silencieuse et de ne pas nécessiter d'encre.



Figure2.28 : Photo d'une balance commerciale avec imprimante

2.2 Conclusion :

Au cours de ce chapitre, nous avons étudié les différentes parties qui constituent une balance électronique.

Pour le chapitre suivant, nous allons aborder la réalisation de notre propre balance électronique en utilisant des composants plus modernes.



Chapitre 3

Etude et Réalisation

3. Introduction :

Dans ce dernier chapitre nous allons entamer plus précisément la partie qui concerne la réalisation du prototype de notre balance électronique en décrivant ses différents composants-avec lesquels nous avons travaillé.

3.1 Schéma synoptique du circuit :

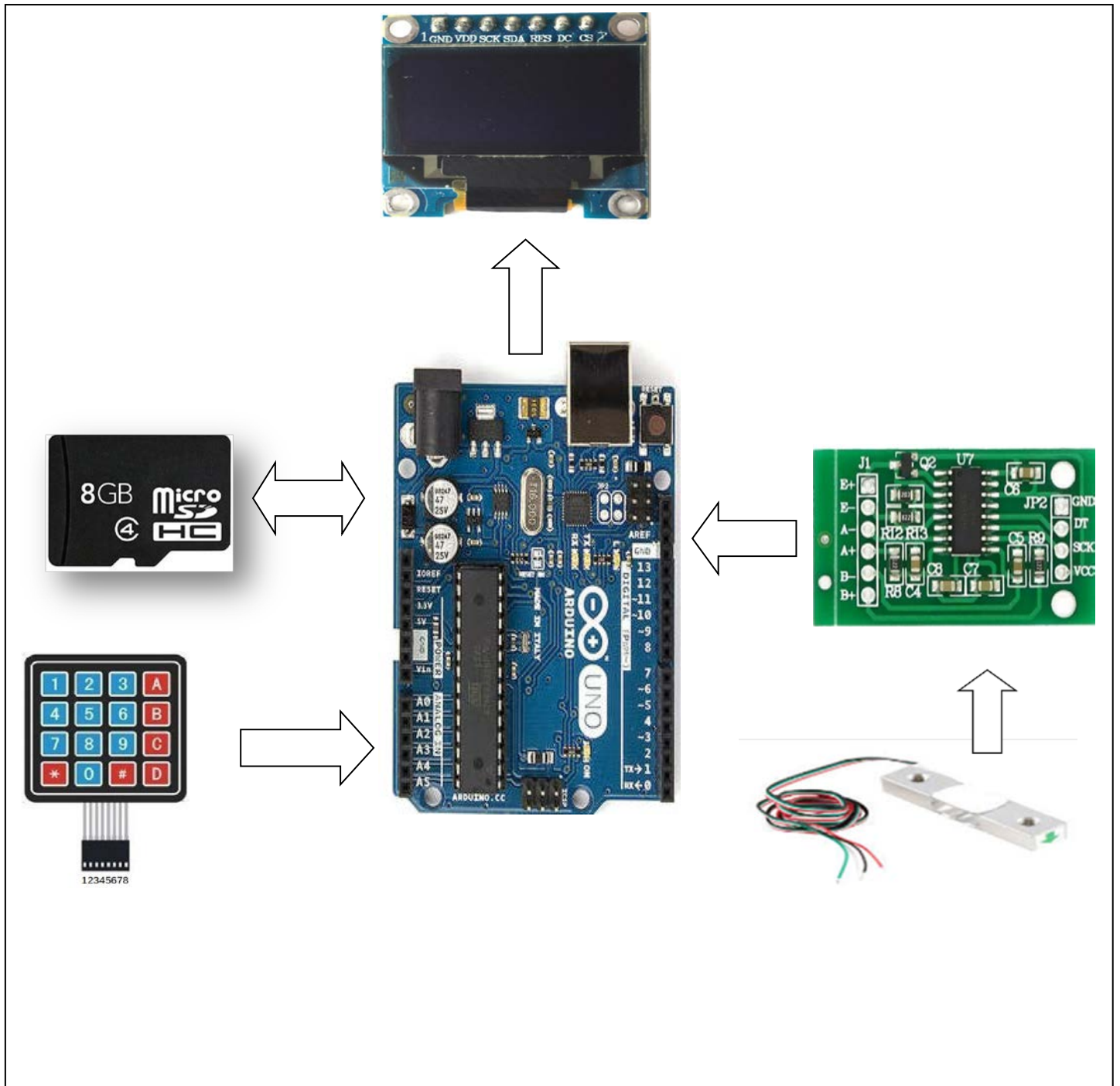


Figure3.1 : Schéma synoptique du circuit

Chapitre3 : Etude et Réalisation

Dans une première étape, notre capteur de pesage à base de jauge de contrainte va délivrer un signal électrique proportionnel au poids déposé sur le plateau. Ce signal est une tension de l'ordre de quelques millivolts, c'est pourquoi nous avons choisi le module HX711 dont l'élément principale est un amplificateur spécialement dédié aux jauges de contraintes pour la mesure de poids.

La carte Arduino Uno est le cerveau de notre balance, c'est elle qui va gérer tout le système électronique de notre balance et afficher le poids, le prix... sur un afficheur OLED et faire enregistrer les données sur une carte SD.

3.2 Description des différents composants :

3.2.1 La carte microcontrôleur de type Arduino Uno :

Le modèle Uno de la société Arduino est une carte électronique dont le cœur est un microcontrôleur ATMEL de référence ATmega328. Ce dernier est un microcontrôleur 8bits de la famille AVR (figure 2.6).

La carte permet comme son nom l'indique, de contrôler des éléments mécaniques : systèmes, lumières, moteurs, etc. Cette carte électronique permet donc à son utilisateur de programmer facilement des choses et de créer des mécanismes automatisés, sans avoir de connaissances particulières en programmation. Elle est un outil pensé est destiné aux inventeurs, artistes ou amateurs qui souhaitent créer leur propre système automatique en le codant de toute pièce.

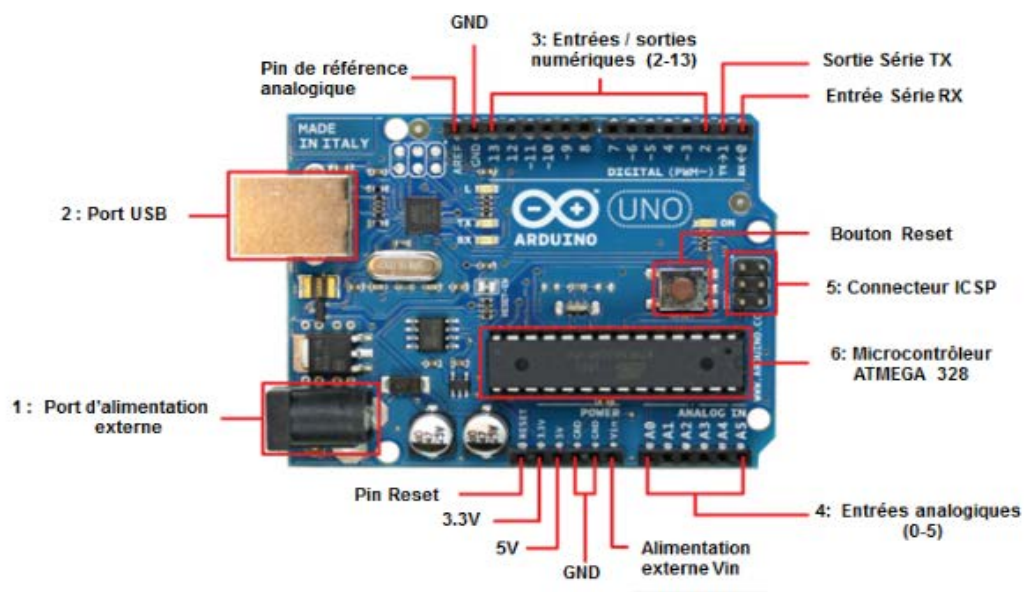


Figure3.2 : Schéma structurel de la carte Arduino Uno

Chapitre3 : Etude et Réalisation

Les signaux d'entrée-sortie du microcontrôleur sont reliés à des connecteurs selon la figure3.7 en bas.

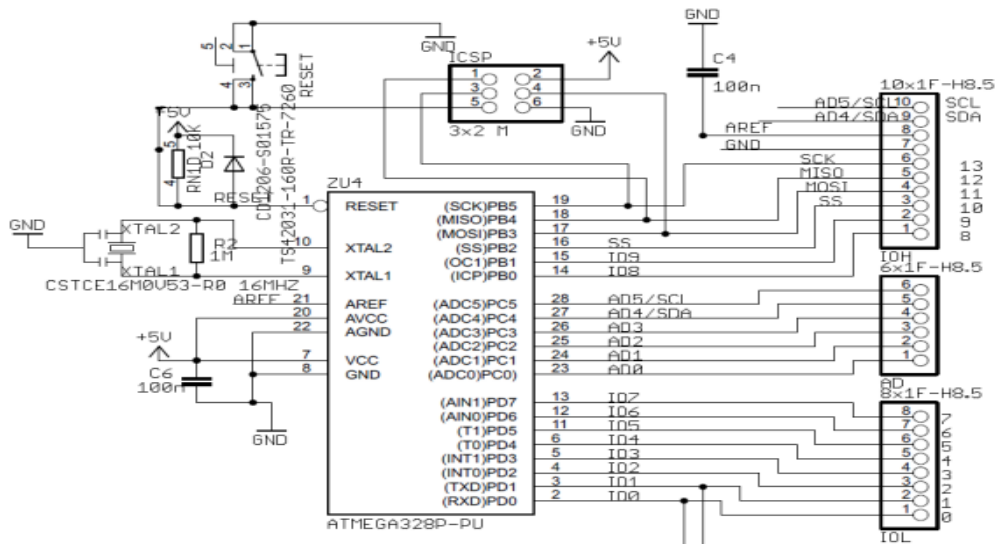


Figure3.3 : Schémas simplifié de la carte Arduino Uno [17]

Le brochage du microcontrôleur est donné dans la figure3.8 ci-dessous.

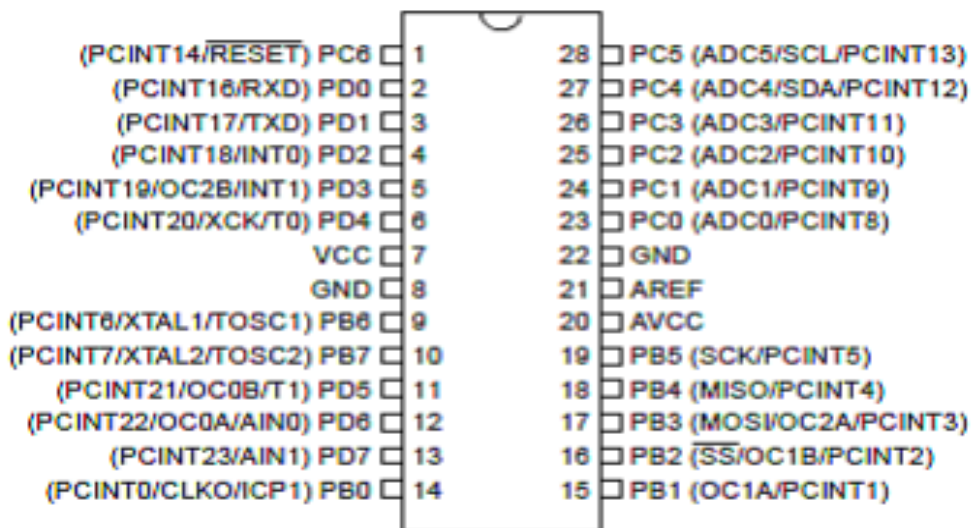


Figure3.4 : Brochage de l'ATMega328

La figure2.9 dans la page suivante montre la correspondance entre les broches de l'ATMega328 et les broches de l'Arduino.

Chapitre3 : Etude et Réalisation



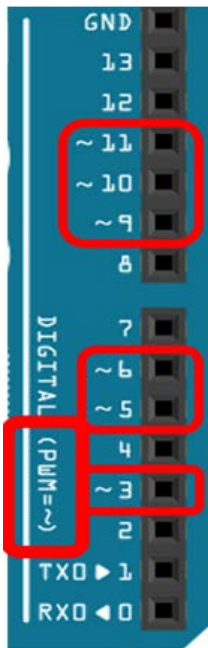
Figure3.5 : La correspondance entre les broches de l'ATMega328 et les broches de l'Arduino [17]

3.2.1.2 Caractéristiques de la carte Arduino Uno :

Caractéristique	Description
Microcontrôleur	ATmega328
Tension d'alimentation	5V
Tension d'entrée (recommandée)	7 à 12V
Tension d'entrée (limites)	6 à 20V
Ports numériques	14 broches d'E/S (6sorties commutables en MLI-modulation de largeur d'impulsion)
Ports analogiques	6 entrées analogiques
Courant max par broches d'E /S	40 mA
Courant max sur sortie 3,3V	50 mA
Mémoire Flash	32 Ko
Mémoire SRAM	2 Ko
Mémoire EEPROM	1 Ko
Fréquence d'horloge	16 MHZ
Timers : timer0, timer2 ; timer1	Comptage 8bits ; comptage 16bits
CAN (résolution 10bits)	6entrées multiplexées ADC0(PC0) à ADC5 (PC5)
Port série (USART)	Emission/réception série via les broches TXD et RXD

Tableau3.1 : Spécifications de la carte Arduino Uno

3.2.1.3 Les entrées / sorties numériques D0 à D13 :



Chacun des connecteurs D0 à D13 peut être configuré par programmation en entrée ou sortie. Les signaux véhiculés par ces connecteurs sont des signaux logiques, c'est –à-dire qu'ils ne peuvent prendre que deux états HAUT (5 volts) ou BAS (0 volts).

En pratique, les connecteurs D0 et D1 réservés pour la liaison série asynchrone (port COM virtuel via le câble USB) ne sont pas exploités pour d'autres utilisations. À noter que chacun des connecteurs ne peut fournir ou absorber un courant supérieur à 40 mA environ (200mA pour l'ensemble des connecteurs) Certains connecteurs peuvent être spécialisés comme sorties PWM.

Les entrées numériques D0 à D7 sont reliées avec les entrées numériques du microcontrôleur PD0 à PD7 et les entrées numériques D8 à D13 sont reliées avec les entrées numériques du microcontrôleur PB0 à PB5 respectivement. [18]

3.2.2.1 Les entrées analogiques A0 à A5 :



Par défaut et contrairement aux entrées /sorties numériques qui ne peuvent prendre que deux états HAUT et BAS, ces six entrées peuvent admettre toute tension analogique comprise entre 0 et 5 volts. Pour pouvoir être traitées par le microcontrôleur, ces entrées analogiques sont prises en charge par un CAN dont le rôle est de convertir une tension en une grandeur numérique binaire sur 10 bits.

Les entrées analogiques A0 à A5 sont reliées avec les entrées analogiques microcontrôleur PC0 à PC5 respectivement. [18]

3.2.2 Le module HX711 :

Pour mesurer le poids à l'aide d'un capteur de force, nous avons généralement besoin d'une source d'alimentation régulée, d'un amplificateur de sortie et d'un convertisseur CAN.

Les trois fonctions nécessitent trois circuits et composants distincts, ce qui augmente les coûts et la complexité.

Chapitre3 : Etude et Réalisation

Nous pouvons donc utiliser le **circuit intégré HX711** doté de toutes les caractéristiques ci-dessus qui supprime simplement les circuits complexes.

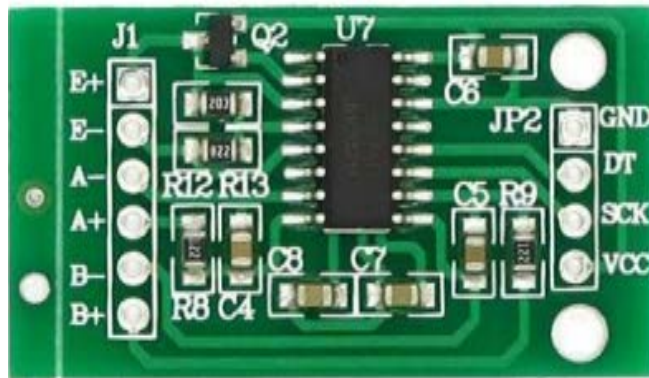


Figure3.6 : Circuit imprimé du hx711

3.2.2.1 Fonctionnement du module HX711 :

Pour comprendre l'utilisation de cet amplificateur de cellules de charges « HX711 », considérons le circuit d'application présenté ci-dessous.

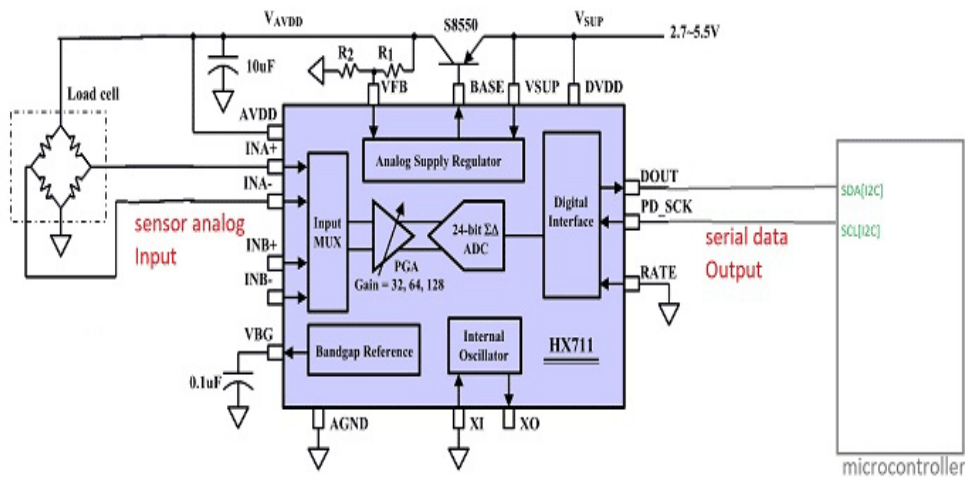


Figure3.7 : Schéma fonctionnel du montage typique d'une balance [19]

Comme indiqué dans le circuit ci-dessus, pour connecter la cellule de pesée à un microcontrôleur à l'aide de HX711, nous avons besoin de quelques composants supplémentaires. Ces composants remplissent diverses fonctions et sont obligatoires.

La cellule de charge est alimentée par la puce HX711, de sorte qu'aucune autre source d'alimentation distincte n'est nécessaire. Elle produit un signal de tension analogique à la sortie en fonction du poids appliqué sur sa surface. Ce signal analogique est transmis à un PGA (amplificateur à gain programmable) via un multiplexeur. Le PGA amplifie le signal et fournit

Chapitre3 : Etude et Réalisation

une sortie vers une interface numérique. Cette dernière convertit le signal analogique en valeur numérique et fournit des données séries à la sortie.

Le HX711 utilise une interface à deux lignes : donnée (DOUT) et horloge (SCK) pour la communication. Les broches GPIO de tous les microcontrôleurs devraient fonctionner, et de nombreuses bibliothèques ont été écrites, facilitant ainsi la lecture des données du HX711 sur 24 bits. [19]

3.2.2.2 Configuration des broches du module HX711 :

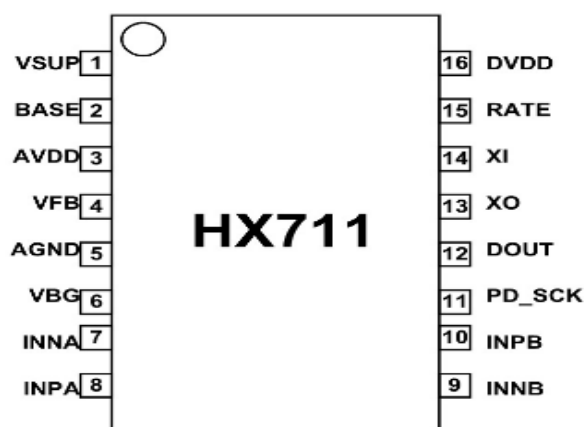


Figure3.8 : Les différentes broches du HX711 [19]

Le HX711 est un dispositif à 16 broches dont La description de chaque broche est donnée dans le tableau3.1 ci-dessous.

Pin n°	Nom de broche	Fonction	Description
1	VSUP	Puissance	Alimentation du régulateur : 2.7 à 5.5V
2	BASE	Sortie analogique	Sortie de contrôle du régulateur
3	AVDD	Puissance	Alimentation analogique : 2.6 à 5.5V
4	VFB	Entrée analogique	Entrée de contrôle du régulateur
5	AGND	Sol	Masse analogique
6	VBG	Sortie analogique	Sortie by-pass de référence
7	DANS UN-	Entrée analogique	Entrée négative du canal A
8	INA +	Entrée analogique	Canal A entrée positive
9	INB-	Entrée analogique	Entrée négative du canal B
9	INB+	Entrée analogique	Entrée positive du canal B
11	PD_SCK	Entrée numérique	Mise hors tension et entrée d'horloge série
12	DOUT	Sortie numérique	Sortie de données série
13	XO	E / S numérique	Crystal I / O
14	XI	Entrée numérique	Crystal I / O ou entrée d'horloge externe, le code PIN est faible : utilisez un oscillateur intégré
15	TAUX	Entrée numérique	Contrôle du débit de données de sortie, Pin est Bas : 10Hz ; Pin est élevé : 80Hz
16	DVDD	Puissance	Alimentation numérique : 2.6 à 5.5V

Tableau3.2 : Brochage du HX711

3.2.2.4 Caractéristiques et spécifications du HX711 :

- Deux canaux d'entrée différentiels sélectionnables.
- PGA actif à faible bruit sur puce avec gain sélectionnable de 32, 64 et 128.
- Régulateur d'alimentation sur puce pour alimentation analogique à cellules de charge et ADC.
- Oscillateur sur puce ne nécessitant aucun composant externe avec cristal externe en option.
- Réinitialisation sur puce.
- Commande numérique simple et interface série : commandes à broche, aucune programmation nécessaire.
- Débit de données de sortie sélectionnable 10SPS ou 80SPS.
- Réjection simultanée de 50 et 60 Hz.
- Consommation électrique : fonctionnement normal <1,5mA, mise hors tension <1uA.
- Plage de tension d'alimentation de fonctionnement : 2.6V à 5.5V.
- Plage de température de fonctionnement : -40 à + 85°C. [19]

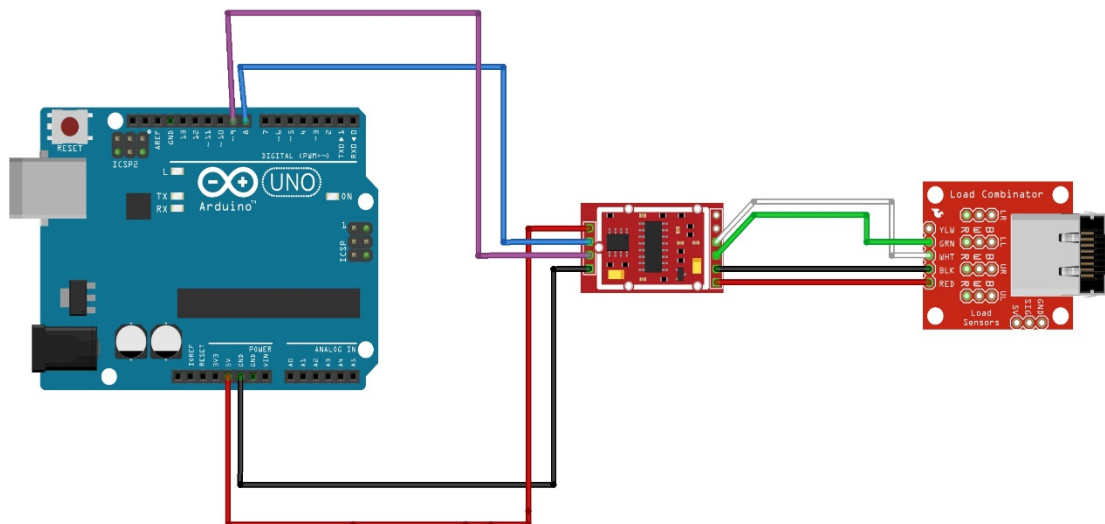
3.3.1 Connexion du module avec la jauge de contrainte et Arduino Uno :

Les cellules de charge utilisent une configuration en pont Wheatstone à quatre fils pour se connecter au Hx711. Ceux-ci sont généralement colorés en Rouge, Noir, Blanc, Vert.

La connexion entre les broches des 3composants est montrée dans le tableau ci-dessous.

La jauge vers HX711	HX711 vers Arduino Uno
Rouge : E+	Vcc vers 5V
Noire : E-	GND vers GND
Blanc : A-	SCK vers D8
Vert : A+	DOUT vers D9

Tableau3.3 : la correspondance entre la jauge, le HX711 et la carte Arduino



fritzing

Figure3.9 : Câblage du HX711 avec la jauge de contrainte et l'Arduino

3.2.3 Le clavier :

Typiquement, le brochage d'un clavier matriciel est le suivant :

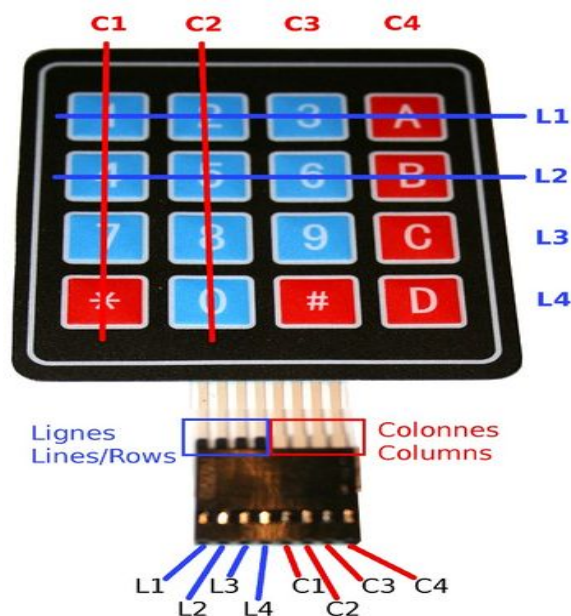


Figure3.10 : Brochage du clavier [20]

3.2.3.1 Principe du clavier matriciel :

Le clavier est organisé en une matrice et c'est l'intersection d'une ligne avec une colonne qui permet d'identifier la touche enfoncée. Avec un clavier matriciel, le nombre de lignes d'entrée/sortie est égal à la somme du nombre de colonnes et du nombre de lignes.

L'identification de la touche enfoncée peut se faire de deux façons :

Chapitre3 : Etude et Réalisation

- Nous générons un niveau bas sur la 1^{ère} colonne tandis que les autres colonnes restent au niveau haut. Puis, nous décalons le niveau bas sur la 2^{ème} colonne en laissant un niveau haut sur les autres colonnes. Nous balayons indéfiniment et rapidement toutes les colonnes avec ce niveau bas et le microcontrôleur scrute en permanence le niveau sur les lignes. Lorsqu'une touche est enfoncée, la colonne et la ligne correspondante à la touche enfoncée sont toutes les deux au niveau bas. Des résistances de tirage sont nécessaires sur les lignes afin d'imposer un niveau haut sur le port d'entrée/sortie du microcontrôleur lorsqu'aucune touche n'est enfoncée.

- Nous envoyons un niveau bas simultanément sur chaque colonne et nous lisons le niveau sur chaque ligne. Si une touche est enfoncée, un niveau bas apparaîtra sur la ligne correspondante. Nous renvoyons alors le code lu des lignes vers les colonnes, ce qui fera apparaître un niveau bas dans l'une d'elles. Les positions des deux "0" lus donnent les coordonnées de la touche recherchée. [21]

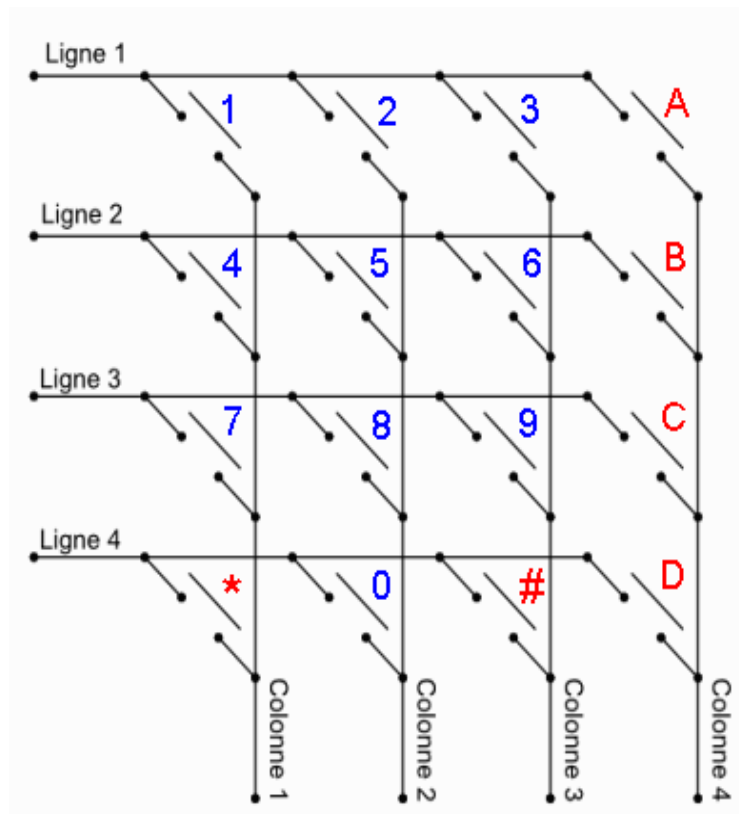


Figure3.11 : Schéma interne du clavier [21]

3.2.3.2 Connexion du clavier avec la carte Arduino Uno :

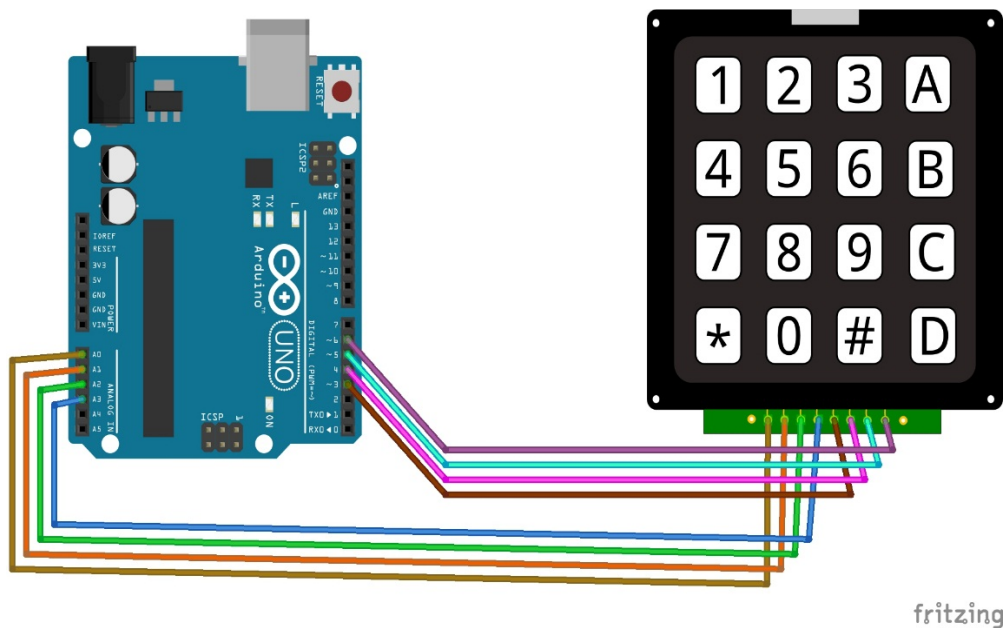


Figure3.12 : Câblage du clavier avec Arduino

3.2.4 L'afficheur OLED :

Une diode électroluminescente organique est un dispositif d'affichage doté d'une technologie auto-émettrice de lumière composée d'un mince film organique multicouche placé entre une anode et une cathode.

Contrairement à la technologie LCD, OLED ne nécessite pas de rétro-éclairage. OLED offre une large plage de visualisation, presque 180 degrés de gauche à droite et de haut en bas, et consomme également moins d'énergie que les écrans LCD existants.

Les écrans OLED ont été utilisés pour moniteurs, téléphones portables, assistants numériques personnels, etc.

3.2.4.1 Module I2C OLED :

Ces écrans sont petits, seulement 0,96 pouce de diamètre mais ils sont très lisibles en raison du contraste élevé d'un écran OLED. Cet écran est composé de 128×64 pixels OLED blanc individuels, chacun d'eux étant activé ou désactivé par la puce du microcontrôleur.

Aucun rétro-éclairage n'est requis pour cet affichage car il crée son propre éclairage, ce qui réduit la puissance requise pour faire fonctionner l'OLED. Son angle visuel est supérieur

à 160 °. Il n'a besoin que de 2 ports d'E / S à contrôler car cet écran utilise l'interface commune I2C pour une utilisation Plug-and-Play facile ce qui signifie que on peut le connecter sans avoir besoin de fils.

L'écran est livré avec un joli protecteur d'écran avec une languette pour pouvoir l'enlever facilement après l'installation. Le circuit imprimé est préparé avec 4 trous de taille raisonnable dans les coins pour la fixation et une embase à 4 broches déjà installée pour alimenter et communiquer avec l'affichage. Il peut être connecté directement à une carte Arduino sans aucune pièce supplémentaire. [22]

3.2.4.2 Caractéristiques des afficheurs OLED :

- Il s'agit d'une interface OLED, I2C (ou IIC) 0,96" 128×64 avec pilote SSD1306, compatible avec 3.3 / 5V.
- Un processus de fabrication simplifié comparé au TFT-LCD.
- Lumière auto-émettant, contrairement au rétro-éclairage requis pour l'écran LCD TFT.
- Haute luminosité.
- Léger et mince.
- Capable de larges angles de vision > 160.
- Basse tension de fonctionnement et consommation électrique.
- Réponse rapide (<deuxième niveau).
- Large gamme de températures de fonctionnement (-30 ° C à 70 ° C). [22]

3.2.4.3 Fonctionnement de la technologie OLED :

Le fonctionnement des OLED repose sur le phénomène d'électroluminescence. L'émission de la lumière résulte de l'action d'un champ électrique ou d'un courant qui passe à travers une substance donnée. Dans le cas des OLED, il s'agit de l'injection de charges électriques dans une couche de matière organique. La lumière provient de l'émission d'un photon suscitée par la recombinaison d'un exciton c'est-à-dire d'un couple électron-trou à l'intérieur de la couche émettrice en polymère.

La structure de base d'une OLED comporte quatre couches :

- Le **substrat transparent** en plastique ou en verre : c'est le support de la diode.
- L'**anode transparente** crée les défauts d'électrons (les trous) lors du passage du courant électrique : elle est faite d'oxyde d'indium-étain.

Chapitre3 : Etude et Réalisation

- La **cathode métallique** constitue la couche émettrice d'électrons : elle est composée principalement de métaux comme l'aluminium, le calcium ou le magnésium.
- La **couche conductrice organique** se trouve entre l'anode et la cathode.

En appliquant une différence de potentiel entre les deux électrodes, l'anode et la cathode émettent respectivement des trous positifs et des électrons qui migrent dans la couche conductrice organique et se recombinent entre eux. Lors de cette recombinaison, le surplus d'énergie apporté par les électrons permet l'émission de photon et donc, de lumière.

En réalité, la couche conductrice organique est elle-même constituée de plusieurs couches organiques ayant pour rôle de faciliter le transport des lacunes ou des électrons vers la couche émettrice de lumière.[23]

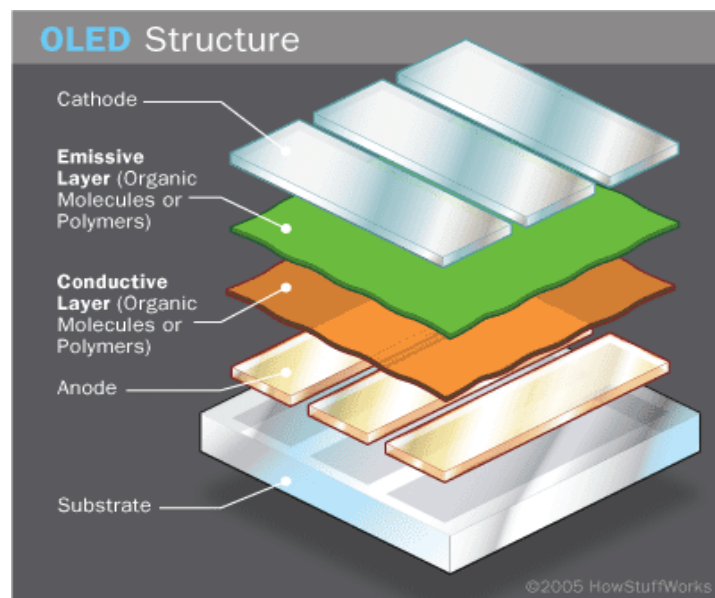


Figure3.13 : Structure d'une OLED [23]

3.2.4.4 Avantages et inconvénients de la technologie OLED :

La technologie OLED comme toute technologie a ces points forts et ces points faibles, on notera ci-dessous quelques-unes :

(+)

- Un affichage OLED n'a pas besoin de rétro-éclairage, ce qui a pour conséquence une économie d'énergie non négligeable ainsi qu'un encombrement réduit et un avantage certain concernant la flexibilité des écrans.

Chapitre3 : Etude et Réalisation

- Le processus et le coût de fabrication des écrans OLED apparaissent comme étant plus rentables comparé à d'autres technologies.

- Le contraste, le rendu des couleurs, le temps de réponse ainsi que l'angle de vision sont également profitables à la technologie OLED. Cela peut s'expliquer par une lumière directement émise par chaque pixel qui compose l'écran, proposant ainsi une meilleure restitution des couleurs et une diffusion de la lumière proche de 90°.

- L'écran affiche un véritable noir, en ce sens qu'il ne correspond à aucune émission de lumière, à l'inverse du LCD utilisant un rétro-éclairage qui a tendance à filtrer à travers la dalle dans les noirs. [24]

(-)

- La durée de vie assez limitée. Bien que des techniques dérivées travaillent à améliorer cette longévité, pour le moment OLED ne peut rivaliser avec ses concurrents à ce niveau.

- La grande sensibilité des semi-conducteurs organiques à l'humidité est un frein supplémentaire à la commercialisation en masse de cette technologie. Cette sensibilité est encore plus importante dans le cas des écrans flexibles, le confinement des semi-conducteurs étant moins sécurisé,

- Le développement de la technologie OLED n'est pas facilité par le fait qu'il s'agit d'une technologie propriétaire, les brevets principaux appartiennent à Kodak, Idemitsu et Sony. [24]

3.2.4.5 Connexions de l'afficheur avec la carte Arduino Uno :

Pour faire la connexion de l'afficheur OLED avec la carte Arduino Uno, nous avons relié les broches entre eux comme suit :

Pins (Oled)	Pins (Arduino)
DC	D4
RESET	D5
SDA	D6
SCL	D7
CS (inutilisé)	A5

Tableau3.4 : Branchement d'OLED et Arduino

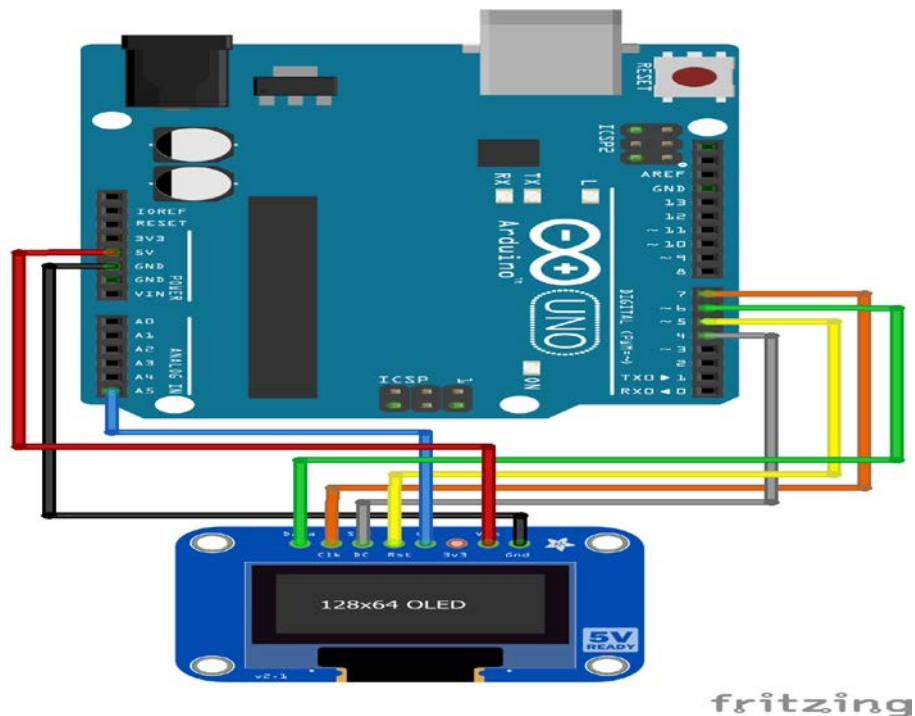


Figure3.14 : Câblage de l'OLED avec Arduino

3.2.5 La carte SD : (mémoire flash)

Cet outil est indispensable pour une balance électronique commerciale car elle permet la sauvegarde et la mémorisation des données de la pesée : la marchandise vendue et le prix correspondant.

3.2.5.1 Description de la carte :

La mémoire flash est une mémoire à semi-conducteurs, non volatile et réinscriptible, c'est-à-dire une mémoire possédant les caractéristiques d'une mémoire vive mais dont les données ne se volatilisent pas lors d'une mise hors tension. Ainsi la mémoire flash stocke les bits de données dans des cellules de mémoire, mais les données sont conservées en mémoire lorsque l'alimentation électrique est coupée. [25]

3.2.5.2 Présentation de la carte shield pour la carte SD :

La carte shield SD Card est une carte d'interface compatible Arduino permettant d'ajouter un espace de stockage sur notre projet de balance électronique. Elle supporte les cartes SD standards et les cartes micro-SD (via adaptateur inclus).

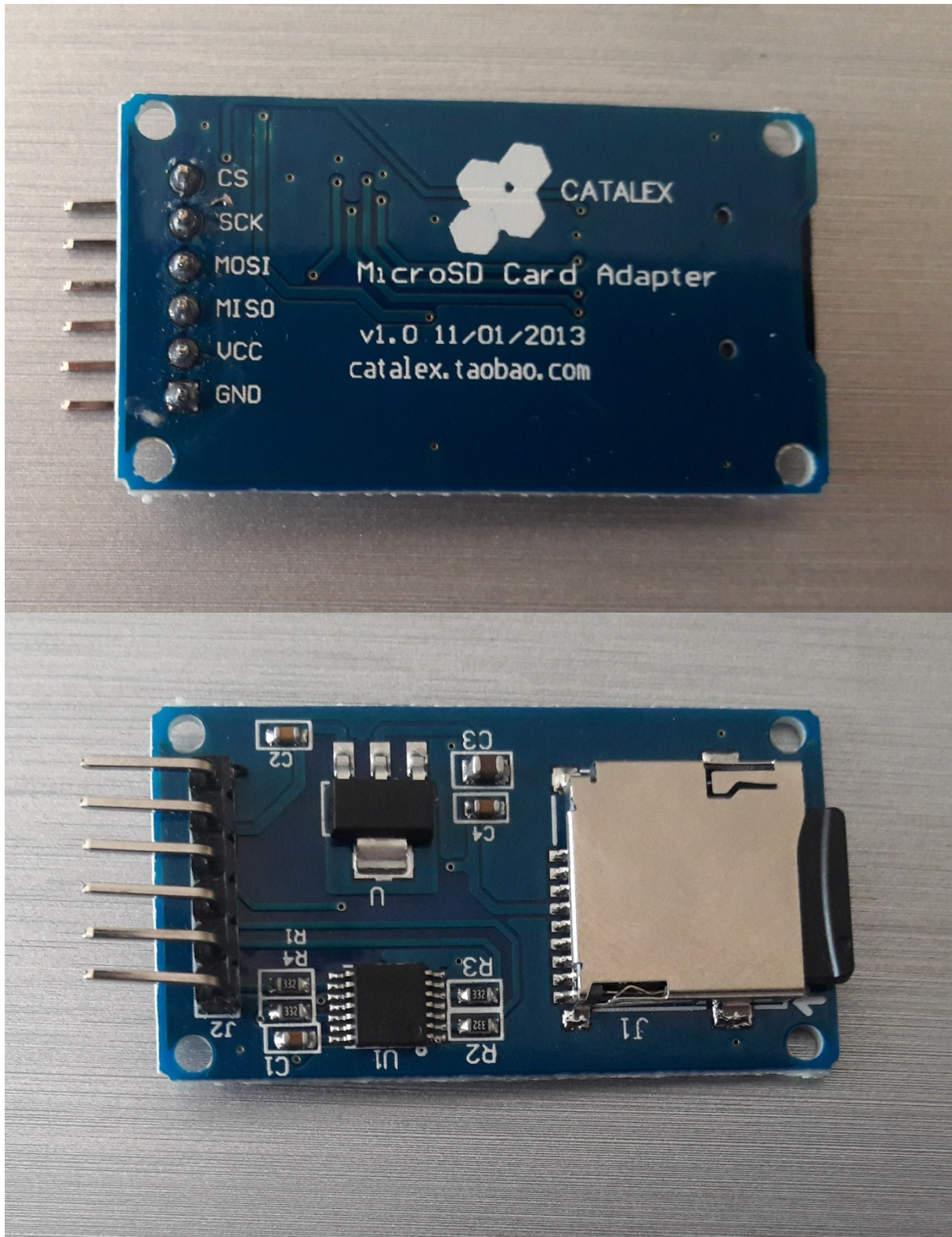


Figure3.15 : Photo d'une carte Shield

Le module de la carte SD présente 6 broches pour permettre d'établir la connexion. 2connexions pour l'alimentation et 4 pour établir la liaison SPI.

Chapitre3 : Etude et Réalisation

- 5V ou 3.3V pour l'alimentation du module.
- GND la masse du module.
- CS ou Chip Select pour activer la communication.
- MISO broche de transmission équivalente à la borne TX d'un port série. Sortie du module.
- MOSI broche de réception équivalente à la borne RX d'un port série. Entrée du module.
- SCK horloge permettant de synchroniser la communication. [25]

3.2.5.3 Connexions avec la carte Arduino Uno :

La plupart des modules pour carte SD embarquent des régulateurs de tension qui permettent d'utiliser 3.3v ou 5v sans problèmes.

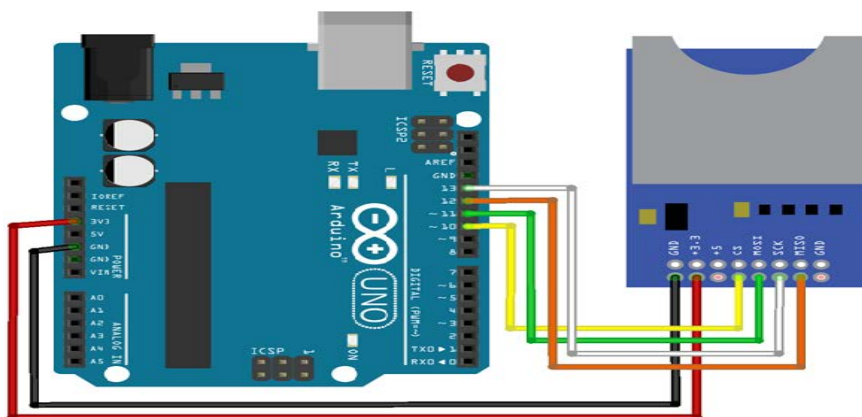


Figure3.16 : Câblage de la carte SD avec l'Arduino Uno

3.3 Présentation de l'espace de développement intégré :

L'interface de l'IDE Arduino est plutôt simple, il offre une interface minimale et épurée pour développer un programme sur les cartes Arduino. Il est doté d'un éditeur de code avec coloration syntaxique et d'une barre d'outils rapide. Ce sont les deux éléments les plus importants de l'interface, c'est ceux que l'on utilise le plus souvent. On retrouve aussi une barre de menus, plus classique qui est utilisé pour accéder aux fonctions avancées de l'IDE.

Enfin, une console affichant les résultats de la compilation du code source, des opérations sur la carte, etc. [26]

Chapitre3 : Etude et Réalisation



Le langage Arduino est inspiré de plusieurs langages. On retrouve notamment des similarités avec le C, le C++, le Java et le Processing. Le langage impose une structure particulière typique de l'informatique embarquée.

- ✚ La fonction « Setup » contiendra toutes les opérations nécessaires à la configuration de la carte (directions des entrées sorties, débits de communications série, etc.).
- ✚ La fonction « Loop » elle, est exécutée en boucle après l'exécution de la fonction setup. Elle continuera de boucler tant que la carte n'est pas mise hors tension, redémarrée (par le bouton reset). Cette boucle est absolument nécessaire sur les microcontrôleurs étant donné qu'ils n'ont pas de système d'exploitation. En effet, si l'on omettait cette boucle, à la fin du code produit, il serait impossible de reprendre la main sur la carte Arduino qui exécuterait alors du code aléatoire. [26]

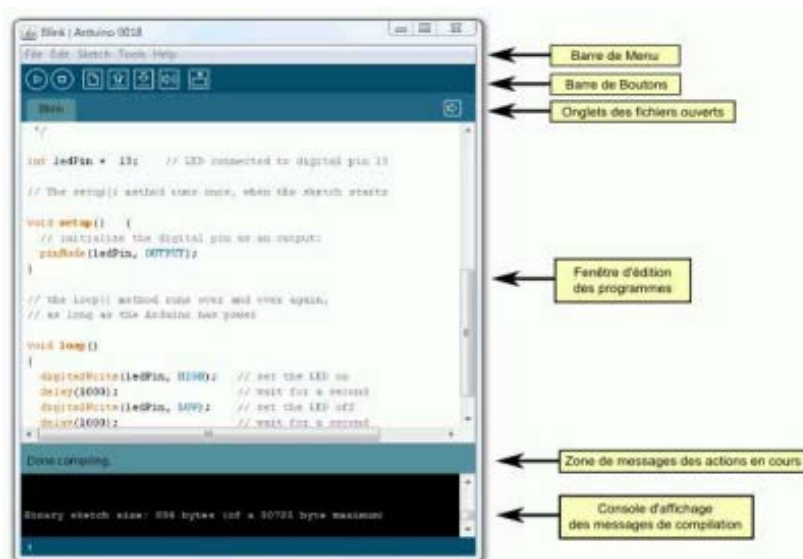


Figure3.17 : Interface de la plateforme Arduino [26]

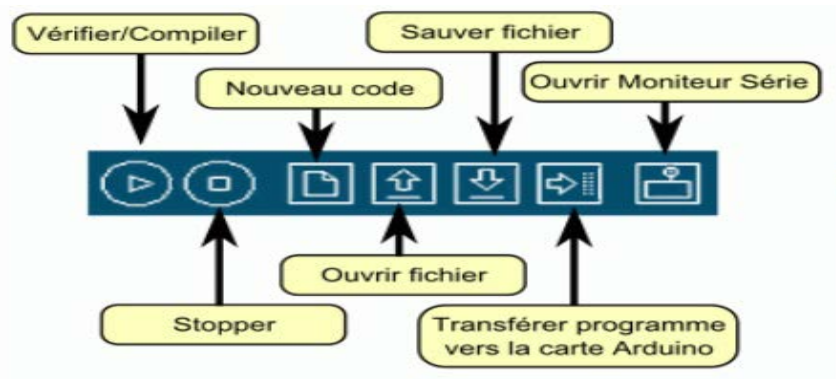


Figure3.18 : Barre de boutons Arduino [26]

3.3.1 Les étapes de développement :

1. On conçoit ou on ouvre un programme existant avec le logiciel ARDUINO.
2. On vérifie ce programme avec le logiciel ARDUINO (compilation).
3. Si des erreurs sont signalées, on modifie le programme.
4. On câble le montage électronique.
5. On charge le programme sur la carte.
6. L'exécution de programme est automatique après quelques secondes.
7. On alimente la carte soit par le port USB, soit par une source d'alimentation
8. autonome (pile 9 volts par exemple).
9. On vérifie que notre montage fonctionne.[26]

3.4 Définition du logiciel PROTEUSE :

Proteus est une suite logicielle permettant la CAO électronique éditée par la société Labcenter Electronics. Proteus est composé de deux logiciels principaux : ISIS, permettant entre autres la création de schémas et la simulation électrique, et ARES destiné à la création de circuits imprimés.

Grâce à des modules additionnels, ISIS est également capable de simuler comportement d'un microcontrôleur (PIC, Atmel, 8051, ARM, HC11...) et son interaction avec les composants qui l'entourent.



Figure3.19 : Fenêtre d'ouverture du logiciel PROTEUS

3.4.1 Présentation de l'environnement ISIS :

ISIS « Intelligent Schematic Input System » est principalement connue pour éditer des schémas électriques. Il permet aussi de simuler les schémas réalisés auparavant. Grâce à lui, nous pouvons également contrôler la majorité de l'aspect graphique du circuit.

3.4.2 Simulation du circuit :

Afin d'atteindre notre objectif principal qui est la réalisation pratique de notre prototype, nous avons d'abord simulé le circuit en question qui est représenté dans la figure3.20 dans la page suivante sur l'environnement ISIS pour tester son fonctionnement.

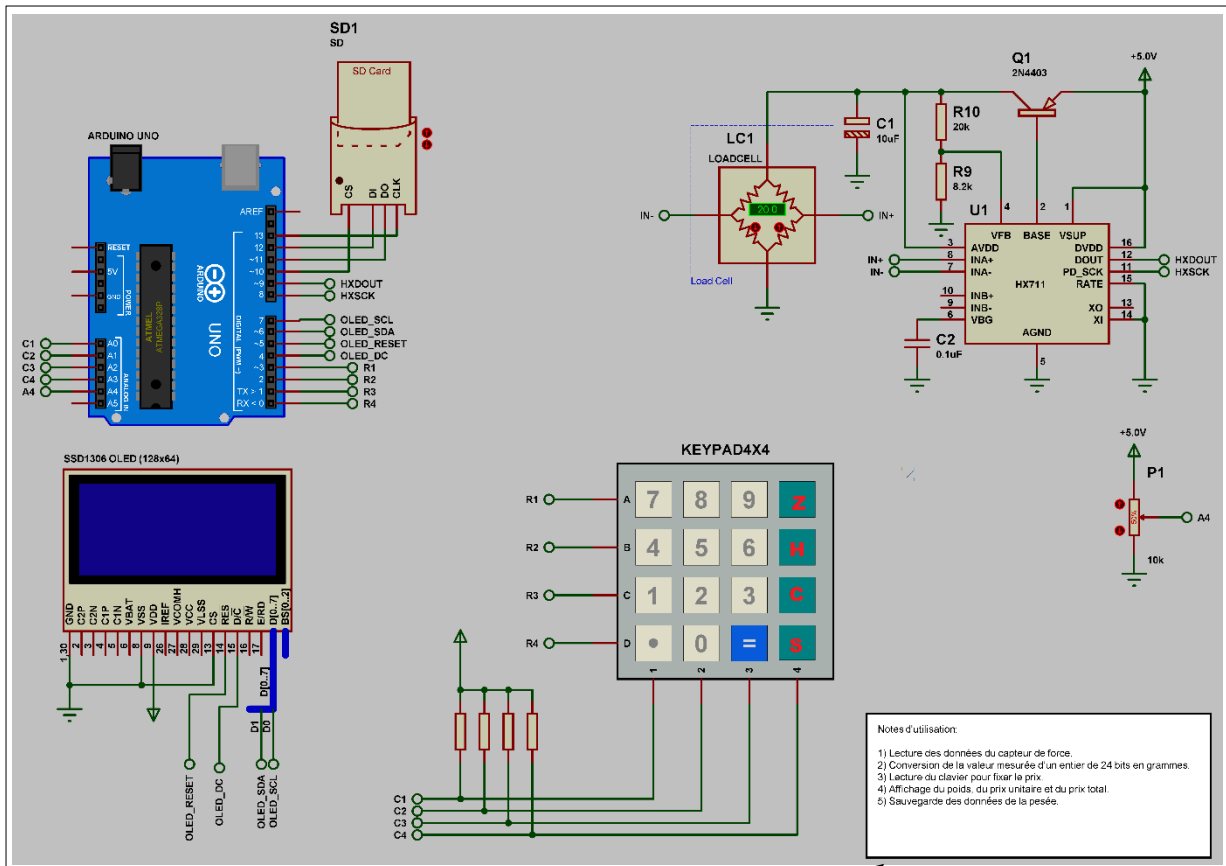


Figure3.20 : Schéma électrique du circuit sur ISIS

3.5 Test du projet final :

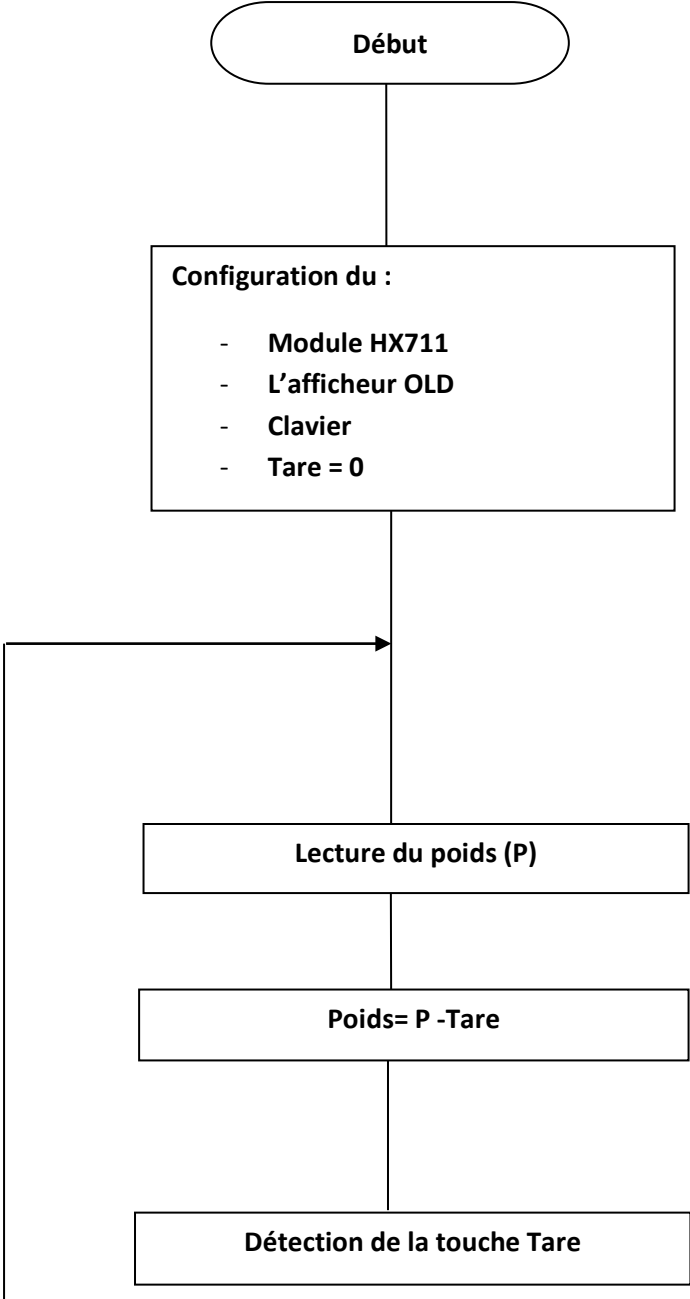
Après avoir câblé les différents composants entre eux, on a fait un test de fonctionnement du circuit sur une plaque d'essai et pour que notre maquette soit bien présentée nous avons construit une balance en verre composée de trois cercles superposés reliés avec la tige filetée.

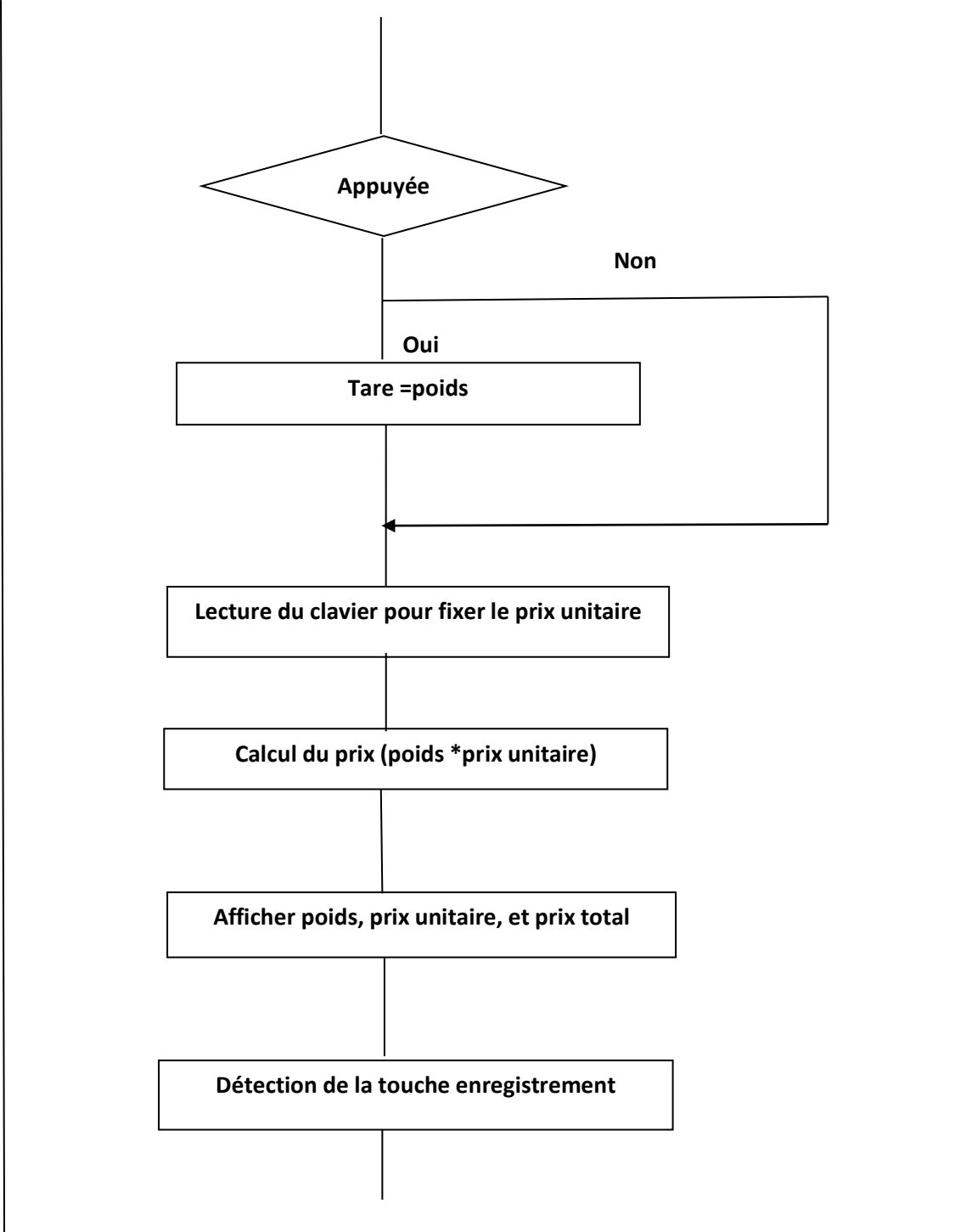
Pour faire la mesure du poids nous avons collé la jauge avec le premier cercle en haut et pour le reste des composants on les a mis en bas comme il est illustré dans la figure3.21.

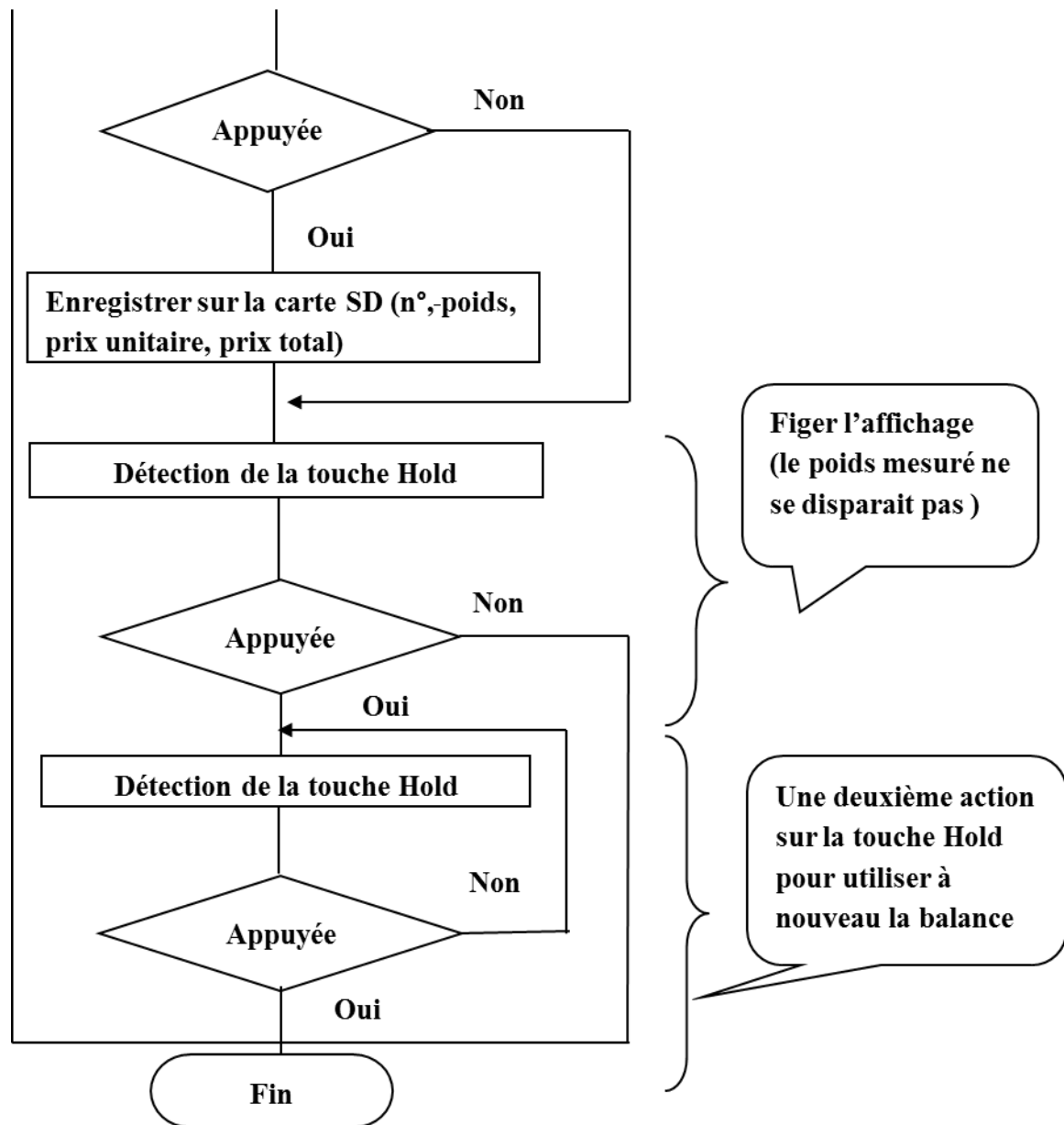


Figure3.21 : les étapes de la réalisation du circuit

3.6 Organigramme :





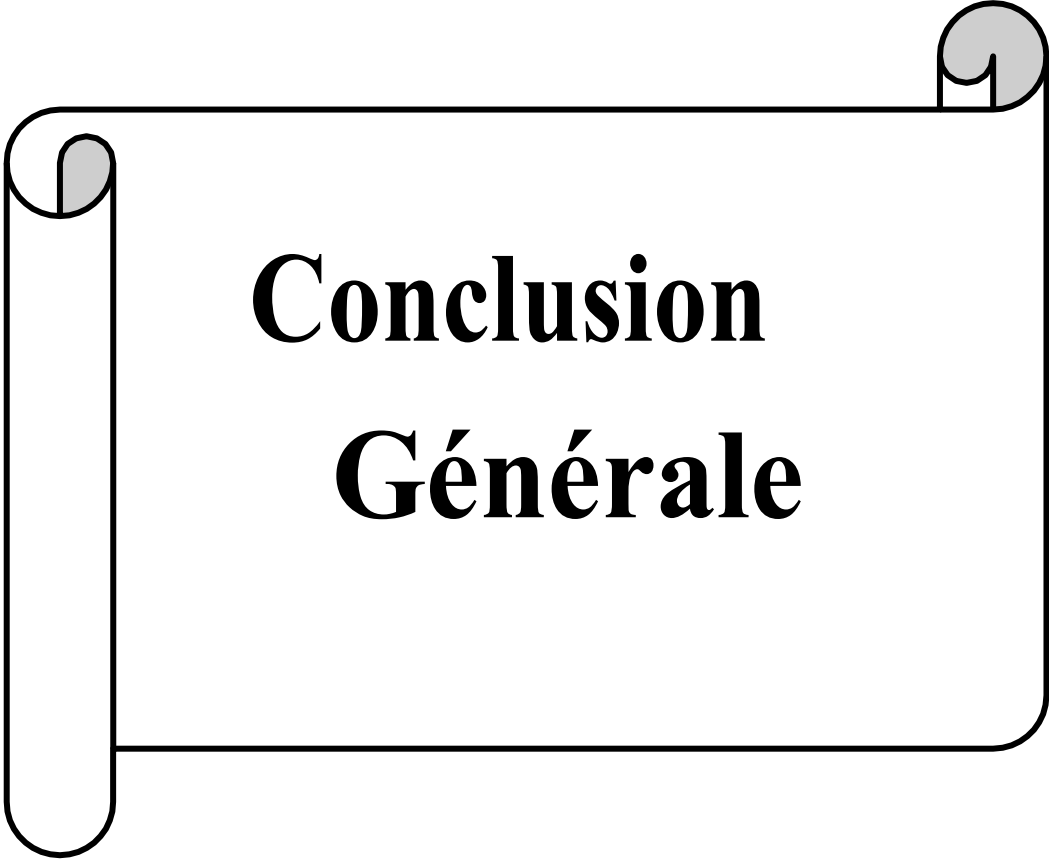


La calibration peut se faire à tout moment à l'aide du potentiomètre P1.

1. Appuyer sur tare pour avoir zéro.
2. Placer un poids de référence exemple 200g.
3. Régler le potentiomètre P1 pour avoir l'affichage 200.
4. Enlever le poids pour revenir à zéro.
5. Remettre le poids pour vérifier la mesure.

3.7 Conclusion :

Dans ce chapitre, nous avons fait d'abord une petite présentation du schéma synoptique de notre montage, puis nous avons étudiée détails tous les étages qui constituent notre balance. Par la suite nous avons commencé par simuler son fonctionnement à l'aide du logiciel Proteus et pour terminer notre projet nous avons réalisé notre prototype de la balance électronique commerciales programmé le microcontrôleur de la carte Arduino avec le programme fonctionnel.



**Conclusion
Générale**

Conclusion Générale

Dans ce projet, nous avons pu réaliser un dispositif concret assurant la fonction d'une balance électronique qui a pour objectif de mesurer un poids allant jusqu'à 5kg en plus de l'affichage on peut mémoriser les données de la pesée.

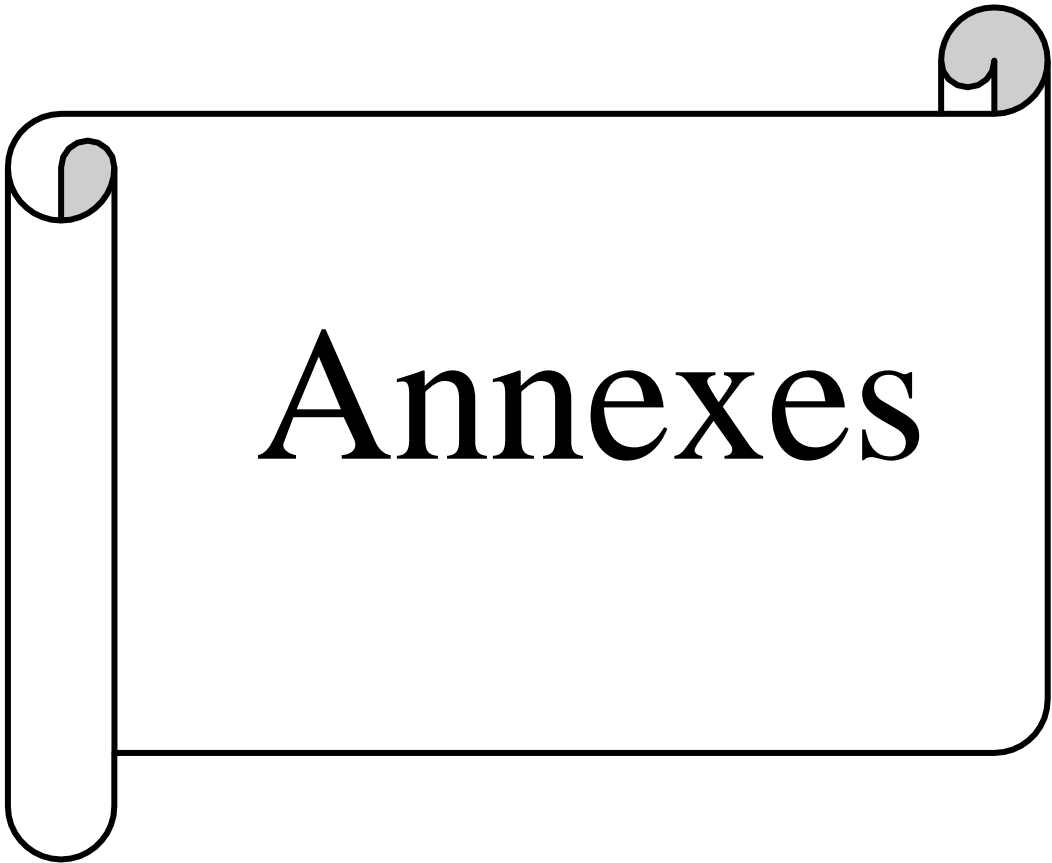
Le projet unissait la pratique et des outils de programmation, cela nous a été d'un grand intérêt dans la mesure où nous avons pu approfondir nos connaissances sur les deux aspects. Ceci dit, approfondir ses connaissances n'est pas tout savoir...

Ce projet était intéressant pour nous pour plusieurs raisons. D'abord, la balance est un outil répandu dans la vie courante qui nous est familier. Mais surtout cela nous a permis de travailler en groupe, et en collectivité. Cette expérience nous a donné, d'ores et déjà, une idée du monde du travail.

Pour améliorer le projet on peut ajouter une horloge temps réel DS1307. Ce module est capable de gérer l'heure (heures, minutes, secondes) et la date (jours, mois, année) tout en s'occupant des mois de 30 ou 31 jours, des années bissextiles, etc. Le calendrier intégré dans le module DS1307 est valable de l'an 2000 à l'an 2100, ce qui devrait être suffisant pour la plupart des projets.

La communication avec le microcontrôleur maître se fait via un bus I2C. Le DS1307 dispose de tout le nécessaire pour garder en mémoire l'heure en cas de coupure d'alimentation grâce à une batterie externe. Une simple pile bouton permet de garder l'heure et la date à jour durant plusieurs années sans alimentation.





Annexes

24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales

DESCRIPTION

Based on Avia Semiconductor's patented technology, HX711 is a precision 24-bit analog-to-digital converter (ADC) designed for weigh scales and industrial control applications to interface directly with a bridge sensor.

The input multiplexer selects either Channel A or B differential input to the low-noise programmable gain amplifier (PGA). Channel A can be programmed with a gain of 128 or 64, corresponding to a full-scale differential input voltage of $\pm 20\text{mV}$ or $\pm 40\text{mV}$ respectively, when a 5V supply is connected to AVDD analog power supply pin. Channel B has a fixed gain of 32. On-chip power supply regulator eliminates the need for an external supply regulator to provide analog power for the ADC and the sensor. Clock input is flexible. It can be from an external clock source, a crystal, or the on-chip oscillator that does not require any external component. On-chip power-on-reset circuitry simplifies digital interface initialization.

There is no programming needed for the internal registers. All controls to the HX711 are through the pins.

FEATURES

- Two selectable differential input channels
- On-chip active low noise PGA with selectable gain of 32, 64 and 128
- On-chip power supply regulator for load-cell and ADC analog power supply
- On-chip oscillator requiring no external component with optional external crystal
- On-chip power-on-reset
- Simple digital control and serial interface: pin-driven controls, no programming needed
- Selectable 10SPS or 80SPS output data rate
- Simultaneous 50 and 60Hz supply rejection
- Current consumption including on-chip analog power supply regulator:
 - normal operation < 1.5mA, power down < 1uA
- Operation supply voltage range: 2.6 ~ 5.5V
- Operation temperature range: -40 ~ +85°C
- 16 pin SOP-16 package

APPLICATIONS

- Weigh Scales
- Industrial Process Control

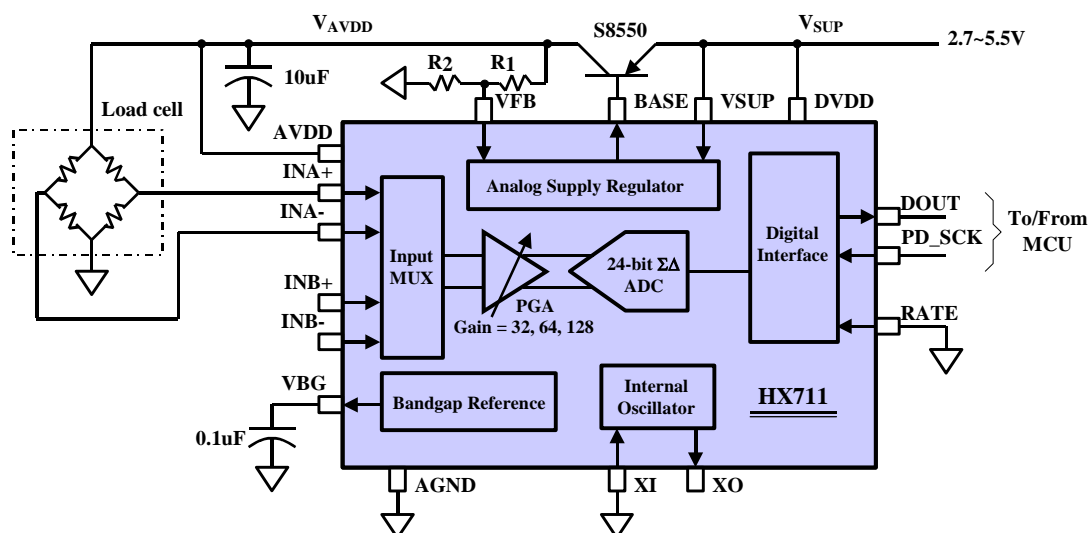
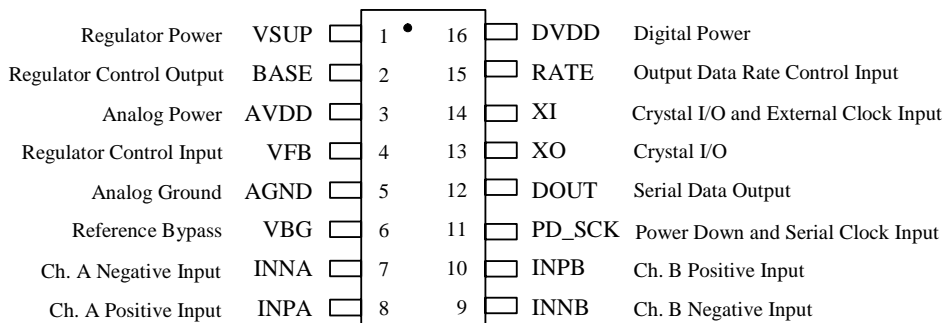


Fig. 1 Typical weigh scale application block diagram

Pin Description


SOP-16L Package

Pin #	Name	Function	Description
1	VSUP	Power	Regulator supply: 2.7 ~ 5.5V
2	BASE	Analog Output	Regulator control output (NC when not used)
3	AVDD	Power	Analog supply: 2.6 ~ 5.5V
4	VFB	Analog Input	Regulator control input (connect to AGND when not used)
5	AGND	Ground	Analog Ground
6	VBG	Analog Output	Reference bypass output
7	INA-	Analog Input	Channel A negative input
8	INA+	Analog Input	Channel A positive input
9	INB-	Analog Input	Channel B negative input
10	INB+	Analog Input	Channel B positive input
11	PD_SCK	Digital Input	Power down control (high active) and serial clock input
12	DOUT	Digital Output	Serial data output
13	XO	Digital I/O	Crystal I/O (NC when not used)
14	XI	Digital Input	Crystal I/O or external clock input, 0: use on-chip oscillator
15	RATE	Digital Input	Output data rate control, 0: 10Hz; 1: 80Hz
16	DVDD	Power	Digital supply: 2.6 ~ 5.5V

Table 1 Pin Description

KEY ELECTRICAL CHARACTERISTICS

Parameter	Notes	MIN	TYP	MAX	UNIT
Full scale differential input range	V(inp)-V(inn)	$\pm 0.5(AVDD/GAIN)$			V
Common mode input		AGND+1.2		AVDD-1.3	V
Output data rate	Internal Oscillator, RATE = 0	10			Hz
	Internal Oscillator, RATE = DVDD	80			
	Crystal or external clock, RATE = 0	$f_{clk}/1,105,920$			
	Crystal or external clock, RATE = DVDD	$f_{clk}/138,240$			
Output data coding	2's complement	800000		7FFFFFF	HEX
Output settling time ⁽¹⁾	RATE = 0	400			ms
	RATE = DVDD	50			
Input offset drift	Gain = 128	0.2			mV
	Gain = 64	0.4			
Input noise	Gain = 128, RATE = 0	50			nV(rms)
	Gain = 128, RATE = DVDD	90			
Temperature drift	Input offset (Gain = 128)	± 6			nV/°C
	Gain (Gain = 128)	± 5			ppm/°C
Input common mode rejection	Gain = 128, RATE = 0	100			dB
Power supply rejection	Gain = 128, RATE = 0	100			dB
Reference bypass (V _{BG})		1.25			V
Crystal or external clock frequency		1	11.0592	20	MHz
Power supply voltage	DVDD	2.6		5.5	V
	AVDD, VSUP	2.6		5.5	
Analog supply current (including regulator)	Normal	1400			μ A
	Power down	0.3			
Digital supply current	Normal	100			μ A
	Power down	0.2			

(1) Settling time refers to the time from power up, reset, input channel change and gain change to valid stable output data.

Table 2 Key Electrical Characteristics

Analog Inputs

Channel A differential input is designed to interface directly with a bridge sensor's differential output. It can be programmed with a gain of 128 or 64. The large gains are needed to accommodate the small output signal from the sensor. When 5V supply is used at the AVDD pin, these gains correspond to a full-scale differential input voltage of $\pm 20\text{mV}$ or $\pm 40\text{mV}$ respectively.

Channel B differential input has a fixed gain of 32. The full-scale input voltage range is $\pm 80\text{mV}$, when 5V supply is used at the AVDD pin.

Power Supply Options

Digital power supply (DVDD) should be the same power supply as the MCU power supply.

When using internal analog supply regulator, the dropout voltage of the regulator depends on the external transistor used. The output voltage is equal to $V_{AVDD} = V_{BG} * (R1 + R2) / R1$ (Fig. 1). This voltage should be designed with a minimum of 100mV below VSUP voltage.

If the on-chip analog supply regulator is not used, the VSUP pin should be connected to either AVDD or DVDD, depending on which voltage is higher. Pin VFB should be connected to Ground and pin BASE becomes NC. The external 0.1uF bypass capacitor shown on Fig. 1 at the VBG output pin is then not needed.

Clock Source Options

By connecting pin XI to Ground, the on-chip oscillator is activated. The nominal output data rate when using the internal oscillator is 10 (RATE=0) or 80SPS (RATE=1).

If accurate output data rate is needed, crystal or external reference clock can be used. A crystal can be directly connected across XI and XO pins. An external clock can be connected to XI pin, through a 20pF ac coupled capacitor. This external clock is not required to be a square wave. It can come directly from the crystal output pin of the MCU chip, with amplitude as low as 150 mV.

When using a crystal or an external clock, the internal oscillator is automatically powered down.

Output Data Rate and Format

When using the on-chip oscillator, output data rate is typically 10 (RATE=0) or 80SPS (RATE=1).

When using external clock or crystal, output data rate is directly proportional to the clock or crystal frequency. Using 11.0592MHz clock or crystal results in an accurate 10 (RATE=0) or 80SPS (RATE=1) output data rate.

The output 24 bits of data is in 2's complement format. When input differential signal goes out of the 24 bit range, the output data will be saturated at 800000h (MIN) or 7FFFFFFh (MAX), until the input signal comes back to the input range.

Serial Interface

Pin PD_SCK and DOUT are used for data retrieval, input selection, gain selection and power down controls.

When output data is not ready for retrieval, digital output pin DOUT is high. Serial clock input PD_SCK should be low. When DOUT goes to low, it indicates data is ready for retrieval. By applying 25~27 positive clock pulses at the PD_SCK pin, data is shifted out from the DOUT output pin. Each PD_SCK pulse shifts out one bit, starting with the MSB bit first, until all 24 bits are shifted out. The 25th pulse at PD_SCK input will pull DOUT pin back to high (Fig.2).

Input and gain selection is controlled by the number of the input PD_SCK pulses (Table 3). PD_SCK clock pulses should not be less than 25 or more than 27 within one conversion period, to avoid causing serial communication error.

PD_SCK Pulses	Input channel	Gain
25	A	128
26	B	32
27	A	64

Table 3 Input Channel and Gain Selection

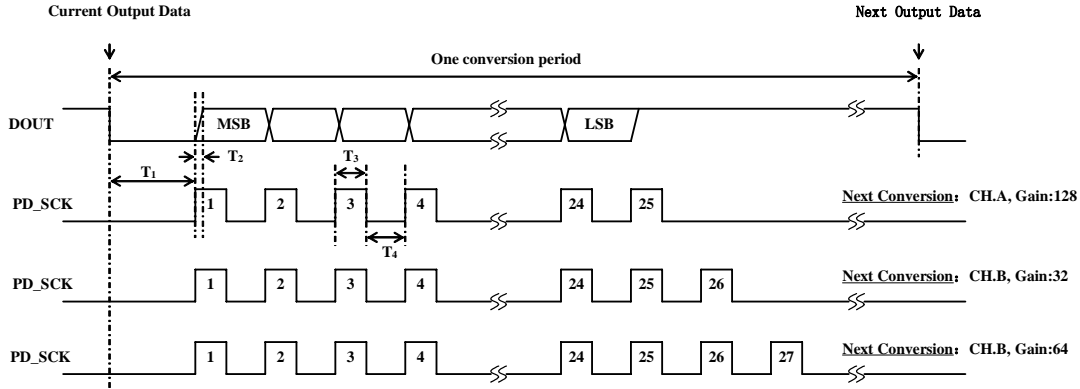


Fig.2 Data output, input and gain selection timing and control

Symbol	Note	MIN	TYP	MAX	Unit
T ₁	DOUT falling edge to PD_SCK rising edge	0.1			μs
T ₂	PD_SCK rising edge to DOUT data ready			0.1	μs
T ₃	PD_SCK high time	0.2	1	50	μs
T ₄	PD_SCK low time	0.2	1		μs

Reset and Power-Down

When chip is powered up, on-chip power on rest circuitry will reset the chip.

Pin PD_SCK input is used to power down the HX711. When PD_SCK Input is low, chip is in normal working mode.

powered down. When PD_SCK returns to low, chip will reset and enter normal operation mode.

After a reset or power-down event, input selection is default to Channel A with a gain of 128.

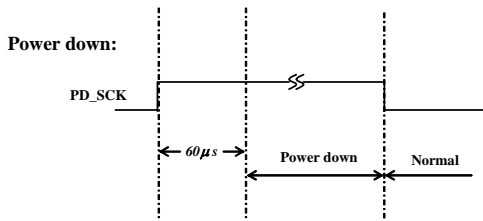


Fig.3 Power down control

When PD_SCK pin changes from low to high and stays at high for longer than 60μs, HX711 enters power down mode (Fig.3). When internal regulator is used for HX711 and the external transducer, both HX711 and the transducer will be

Application Example

Fig.1 is a typical weigh scale application using HX711. It uses on-chip oscillator (XI=0), 10Hz output data rate (RATE=0). A Single power supply (2.7~5.5V) comes directly from MCU power supply. Channel B can be used for battery level detection. The related circuitry is not shown on Fig. 1.

Reference PCB Board (Single Layer)

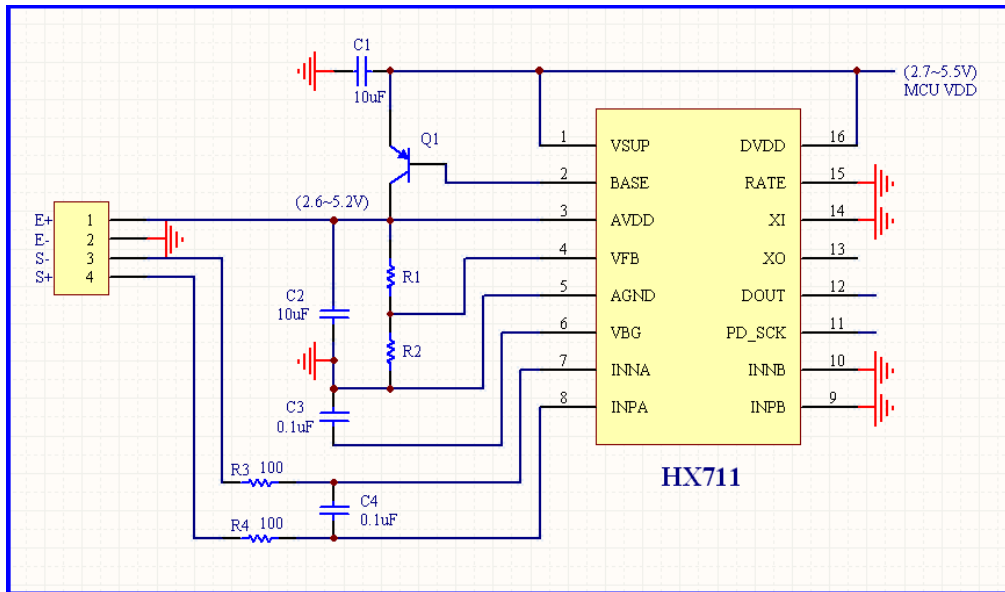


Fig.4 Reference PCB board schematic

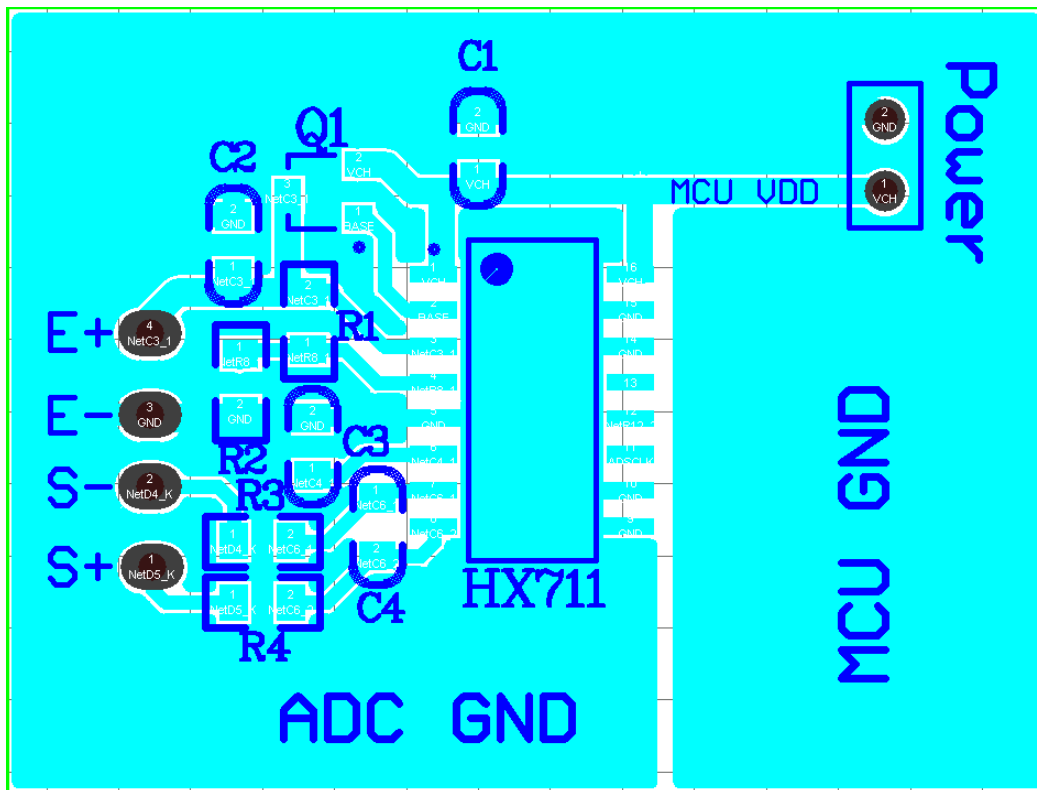


Fig.5 Reference PCB board layout



Reference Driver (Assembly)

```

/*-----
Call from ASM:      LCALL   ReaAD
Call from C:       extern unsigned long ReadAD(void);
                    .
                    .
                    unsigned long data;
                    data=ReadAD();
                    .
                    .
-----*/

PUBLIC      ReadAD
HX711ROM    segment code
rseg       HX711ROM

sbit       ADD0 = P1.5;
sbit       ADSK = P0.0;
/*-----
OUT:   R4, R5, R6, R7   R7=>LSB
-----*/

ReadAD:
    CLR    ADSK           //AD Enable (PD_SCK set low)
    SETB   ADD0          //Enable 51CPU I/O
    JB     ADD0,$        //AD conversion completed?
    MOV    R4,#24

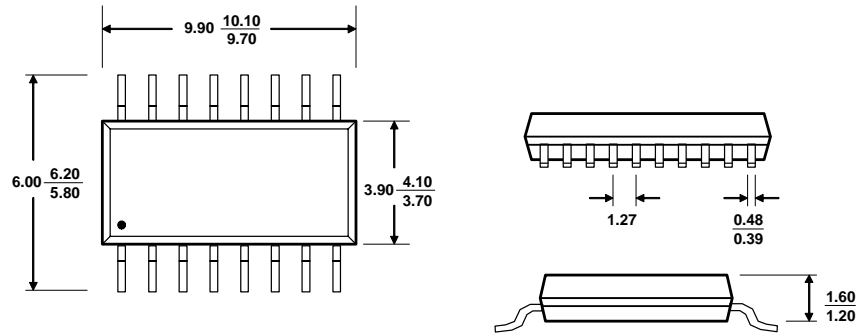
ShiftOut:
    SETB   ADSK          //PD_SCK set high (positive pulse)
    NOP
    CLR    ADSK          //PD_SCK set low
    MOV    C,ADD0        //read on bit
    XCH   A,R7           //move data
    RLC   A
    XCH   A,R7
    XCH   A,R6
    RLC   A
    XCH   A,R6
    XCH   A,R5
    RLC   A
    XCH   A,R5
    DJNZ  R4,ShiftOut    //moved 24BIT?
    SETB   ADSK
    NOP
    CLR    ADSK
    RET
    END

```

Reference Driver (C)

```
//-----  
sbit ADD0 = P1^5;  
sbit ADSK = P0^0;  
unsigned long ReadCount(void) {  
    unsigned long Count;  
    unsigned char i;  
    ADD0=1;  
    ADSK=0;  
    Count=0;  
    while(ADD0);  
    for (i=0;i<24;i++) {  
        ADSK=1;  
        Count=Count<<1;  
        ADSK=0;  
        if(ADD0) Count++;  
    }  
    ADSK=1;  
    Count=Count^0x800000;  
    ADSK=0;  
    return(Count);  
}
```

Package Dimensions



Typ $\frac{\text{MAX}}{\text{MIN}}$ Unit: mm

SOP-16L Package

Code source

```
// This software is made by (c)Elektor Labs
#include <U8glib.h>
#include "HX711.h"
#include <SPI.h>
#include <SD.h>

/* SD card attached to SPI bus as follows:
   MOSI - pin 11
   MISO - pin 12
   CLK - pin 13
   CS - pin 10 */
// clavier
#include "TimerOne.h"
const uint8_t no_key = 0U;
const uint8_t keys_val[4][4] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
volatile uint8_t key_main_access;
volatile uint8_t key_state;
volatile uint8_t old_col;
volatile uint8_t new_col;
volatile uint8_t new_row;
volatile uint8_t main_read;
volatile uint8_t main_interrupted;
volatile uint8_t key1;
volatile uint8_t key2;
const int row_1 = 3;
const int row_2 = 2;
```

```
const int row_3 = 1;
const int row_4 = 0;
const int col_1 = 17;
const int col_2 = 16;
const int col_3 = 15;
const int col_4 = 14;
volatile uint32_t old_time, new_time, diff = 0U;
//clavier
#define DOUT 9
#define CLK 8
HX711 scale(DOUT, CLK);
U8GLIB_SSD1306_128X64 u8g(7, 6, 19, 4, 5); //scl,sda,cs(unused),dc,res
File myFile;
//variable declaration
float calibration_factor = -7050.0; //This value is obtained using the SparkFun_HX711_zerocalibration
sketch
float gram;
float punit = 0;
float ptotal;
unsigned long numero = 0;
float factor;
int button1 = 2;
int button2 = 3;
int bar = 0;
bool button_zero = 0;
bool button_hold = 0;
uint8_t c[] = //copyright symbol
{
    0b00000011, 0b11100000,
    0b00001100, 0b00011000,
    0b00010000, 0b10000100,
```

```

0b00100011, 0b01100010,
0b00100100, 0b00010010,
0b01001000, 0b00000001,
0b01001000, 0b00000001,
0b01010000, 0b00000001,
0b01001000, 0b00000001,
0b01001000, 0b00000001,
0b00100100, 0b00010010,
0b00100011, 0b01100010,
0b00010000, 0b10000100,
0b00001100, 0b00011000,
0b00000011, 0b11100000,
};
//----- SETUP -----
void setup()
{
  //Serial.begin(9600);
  //Serial.println("OLED SSD1306 Display");
  //calibrate the the scale of the hx711 and reset the zero point
  scale.set_scale(calibration_factor);
  scale.tare();
  pinMode(10, OUTPUT); // CS SDCard
  //clavier
  /* init all vars for keypad in default state */
  init_all_vars__key_c();
  /* 20ms timer interrupt */
  Timer1.initialize(20000);
  Timer1.attachInterrupt(isr_key_callback);
  /* all colums inout high */
  pinMode(col_1, INPUT_PULLUP);
  pinMode(col_2, INPUT_PULLUP);

```

```

pinMode(col_3, INPUT_PULLUP);
pinMode(col_4, INPUT_PULLUP);
/* all rows output HIGH */
pinMode(row_1, OUTPUT);
pinMode(row_2, OUTPUT);
pinMode(row_3, OUTPUT);
pinMode(row_4, OUTPUT);
digitalWrite(row_1, HIGH);
digitalWrite(row_2, HIGH);
digitalWrite(row_3, HIGH);
digitalWrite(row_4, HIGH);
//clavier
//calibrate the OLED screen and draw the initial screen
u8g.setColorIndex(1);
u8g.firstPage();
do {
    draw(0);
} while (u8g.nextPage());
u8g.firstPage();
delay(1000);
}
//----- FIN SETUP -----
// INTERRUPTION CLAVIER
void isr_key_callback()
{
    //old_time = micros();
    uint8_t check;

    if (OU == key_main_access) /* if main is not accessing keypad variable, will happen if main wants to
reset keypad variables */
    {
        if (OU == key_state) /* ground all rows & check for any col low */

```

```

{
    digitalWrite(row_1, LOW); digitalWrite(row_2, LOW); digitalWrite(row_3, LOW);
digitalWrite(row_4, LOW);

    if (!digitalRead(col_1)) /* if found store the col & move to next state */
    {
        old_col = 1U;
        key_state = 1U;
    }
    else if (!digitalRead(col_2))
    {
        old_col = 2U;
        key_state = 1U;
    }
    else if (!digitalRead(col_3))
    {
        old_col = 3U;
        key_state = 1U;
    }
    else if (!digitalRead(col_4))
    {
        old_col = 4U;
        key_state = 1U;
    }
    else
    {
        } }
    else if (1U == key_state) /* ground one row at a time & check for col low */
    {
        check = 0U; /* assume row 1 is found to be low */

        digitalWrite(row_1, LOW); digitalWrite(row_2, HIGH); digitalWrite(row_3, HIGH);
digitalWrite(row_4, HIGH);

        if (!digitalRead(col_1)) /* if found store the col & row */

```



```

{
    new_col = 1U;
    new_row = 1U;
}
else if (!digitalRead(col_2))
{
    new_col = 2U;
    new_row = 1U;
}
else if (!digitalRead(col_3))
{
    new_col = 3U;
    new_row = 1U;
}
else if (!digitalRead(col_4))
{
    new_col = 4U;
    new_row = 1U;
}
else
{
    check = 1U; /* initial assumption that this row found to be false, so now check next row */
}
if (1U == check) /* repeat same as row 1 */
{
    check = 0U;

    digitalWrite(row_1, HIGH); digitalWrite(row_2, LOW); digitalWrite(row_3, HIGH);
digitalWrite(row_4, HIGH);

    if (!digitalRead(col_1))
    {
        new_col = 1U;

```

```

    new_row = 2U;
}
else if (!digitalRead(col_2))
{
    new_col = 2U;
    new_row = 2U;
}
else if (!digitalRead(col_3))
{
    new_col = 3U;
    new_row = 2U;
}
else if (!digitalRead(col_4))
{
    new_col = 4U;
    new_row = 2U;
}
else
{
    check = 1U;
} }
if (1U == check) /* repeat same as row 1 */
{
    check = 0U;

    digitalWrite(row_1, HIGH); digitalWrite(row_2, HIGH); digitalWrite(row_3, LOW);
    digitalWrite(row_4, HIGH);

    if (!digitalRead(col_1))
    {
        new_col = 1U;
        new_row = 3U;
    }
}

```

```

else if (!digitalRead(col_2))
{
    new_col = 2U;
    new_row = 3U;
}
else if (!digitalRead(col_3))
{
    new_col = 3U;
    new_row = 3U;
}
else if (!digitalRead(col_4))
{
    new_col = 4U;
    new_row = 3U;
}
else
{
    check = 1U;
} }
if (1U == check) /* repeat same as row 1 */
{
    check = 0U;
    digitalWrite(row_1, HIGH); digitalWrite(row_2, HIGH); digitalWrite(row_3, HIGH);
digitalWrite(row_4, LOW);
    if (!digitalRead(col_1))
    {
        new_col = 1U;
        new_row = 4U;
    }
    else if (!digitalRead(col_2))
    {

```

```

new_col = 2U;
new_row = 4U;
}
else if (!digitalRead(col_3))
{
new_col = 3U;
new_row = 4U;
}
else if (!digitalRead(col_4))
{
new_col = 4U;
new_row = 4U;
}
else
{
check = 1U;
} }
if (0U == check) /* a row low has been found */
{
if (new_col == old_col) /* if new column & old column has been equal */
{
if ((new_col >= 1U) && (new_col <= 4) && (new_row >= 1) && (new_row <= 4)) /* if new col &
new row are within array boundaries */
{
if (0U == main_read) /* if main is not reading the key state */
{
key1 = keys_val[new_row - 1U][new_col - 1U]; /* store the key value */
}
else
{
key2 = keys_val[new_row - 1U][new_col - 1U]; /* mainline got interrupted */

```

```

        main_interrupted = 1U;
    }

    key_state = 2U;    /* key found successfully, now move to next state */
}

else
{
key_state = 0U;    /* if new col & new row are not within array boundaries, something wrong, move
to default state */
    } }

else /* if new col & old col are not same, something wrong, move to default state */
{
    key_state = 0U;
} }

else /* if no row low has been found, something wrong,move to default state */
{
    key_state = 0U;
} }

else if (2U == key_state) /* wait for all keys to open again, by grounding all rows, & wait for col to
be high */
{
    digitalWrite(row_1, LOW); digitalWrite(row_2, LOW); digitalWrite(row_3, LOW);
digitalWrite(row_4, LOW);

    if (digitalRead(col_1) && digitalRead(col_2) && digitalRead(col_3) && digitalRead(col_4))
    {
        key_state = 0U;    /* all col found high, move to default state */
    } }
else
{
} }

/*new_time = micros();
if(diff < (new_time-old_time))
{

```

```

    diff = new_time-old_time;

    Serial.println(diff);
  */
}

// INTERRUPTION CLAVIER
//----- AFFICHAGE -----
//main function to draw the pages on the OLED screen
void draw(int page)
{
  //the outer border of the screen
  for (int i = 0; i < 2; i++)
  {
    // u8g.drawFrame(i,i,128-2*i,64-2*i);
    u8g.drawFrame(i, i, 128 - 2 * i, 64 - 2 * i);
  }
  //sets the font for the text
  u8g.setFont(u8g_font_unifont);
  //initial screen
  if (page == 0)
  {
    u8g.drawStr(4, 30, "Instrumentation");
    u8g.drawBitmap(85, 40, 2, 15, c);
  }
  //visualization of the measured weight in gram and ounce
  if (page == 1 || page == 2 || page == 7)
  {
    u8g.drawStr(15, 14, "MELN:");
    u8g.setPrintPos(10, 28);
    u8g.print("poids:");
    u8g.print(gram, 0);
    u8g.print(" g");
  }
}

```

```

u8g.setPrintPos(10, 43);
u8g.print("PU:");
u8g.print(punit, 0);
u8g.print(" DA");
u8g.setPrintPos(10, 57);
u8g.print("prix:");
u8g.print(ptotal, 0);
u8g.print(" DA");
//bar graph of the weight in gram
u8g.drawStr(108, 17, "0");
u8g.drawStr(108, 56, "5"); //for use with a different max load, change the string to the new max
and change the first number(-8 for each added character)
u8g.drawFrame(118, 10, 7, 44);
//makes sure the bar does not surpass the graph and makes sure the height of the bar isn't less then 1
if (bar >= 1 && bar <= 40)
{
u8g.drawBox(120, 12, 3, bar);
}
else if (bar > 40)
{
u8g.drawBox(112, 12, 3, 40);
u8g.drawStr(10, 57, "Too much!");
}
//displays wether or not the program is in hold mode
if (page == 2)
{
u8g.drawStr(64, 14, "Hold");
}
//displays "SAVE" is in save mode
if (page == 7)
{ u8g.drawStr(64, 14, "Save");

```

```

    } }
//tells the user that the scale has been calibrated
if (page >= 3 && page <= 6)
{
    u8g.drawStr( 10, 20, "calibrating");
    if (page == 4)
    {
        u8g.drawStr( 10, 33, ".");
    }
    if (page == 5)
    {
        u8g.drawStr( 10, 33, "..");
    }
    if (page == 6)
    {
        u8g.drawStr( 10, 33, "...");
    } } }

//----- FIN AFFICHAGE -----
//----- LOOP -----

//main loop
void loop()
{
    // clavier
    uint8_t temp;
    temp = key_read();
    if (no_key != temp)
    {
        //Serial.print("KEY: ");
        //Serial.println((char)temp);
        if (temp >= 48 && temp <= 57)
        {

```



```

    punit = punit * 10 + temp - 48;
}
else if (temp == 67) // Touche C : clear
{
    punit = 0;
}
else if (temp == 65) // Touche A : Zero & Calibration
{
    scale.tare();
    for (int i = 3; i < 7; i++)
    {
        u8g.firstPage();
        do {
            draw(i);
        } while (u8g.nextPage());
        delay(250);
    } }
else if (temp == 66) // Touche B : Hold
{
    u8g.firstPage();
    do {
        draw(2);
    } while (u8g.nextPage());
    delay(1000);
    while (1)
    {
        if (key_read() == 66)
        {
            delay(20);
            break;
        } } }

```

```

else if (temp == 68) // Touche D : Save
{
  if (SD.begin())
  {
    //Serial.println("initialization done.");
    myFile = SD.open("recette.txt", FILE_WRITE);
    if (myFile)
    {
      // afficher "Save" sur l'afficheur OLED
      u8g.firstPage();
      do {
        draw(7);
      } while (u8g.nextPage());
      delay(1000);
      //Serial.print("Writing to recette.txt...");
      numero = numero + 1;
      myFile.print(String(numero));
      myFile.print("----");
      myFile.print(String(gram, 0));
      myFile.print("----");
      myFile.print(String(punit, 0));
      myFile.print("----");
      myFile.println(String(ptotal, 0));
      myFile.close();
      //Serial.println("done.");
    } else
    {
      //Serial.println("error opening recette.txt");
    }
  } else
  {

```

```

    //Serial.println("initialization failed!");
} } }
//clavier
//measures an average ADC value of the potmeter
for (int i = 0; i < 100; i++)
{
    factor += analogRead(4);
    //Serial.print("analogRead=");
    //Serial.println(factor);
}
//the measurement is calculated in a factor for the weight
factor /= 100;
factor += 22500;
//the actual weight is calculated in gram and ounce(avoirdupois)
gram = (-scale.get_units(10) * int(factor) / 1000);
ptotal = gram * punit;
//calculates the height of the bar in the graph. max height is 40 pixels and max weight is 5000g
bar = gram * 40 / 5000;
//initiates the draw function for the visualisation of the weight
u8g.firstPage();
do {
    draw(1);
} while (u8g.nextPage());
}
//=====key_read=====
uint8_t key_read(void)
{
    uint8_t temp;
    main_read = 1U; /* set var for main line read & read the key value */
    temp = key1;
    key1 = no_key;
}

```

```

main_read = 0U;
if (1U == main_interrupted) /* if main line gets interrupted, again read the key var */
{
    temp = key2;
    key2 = no_key;
    main_interrupted = 0U;
}
return temp;
}

//=====clear_key_vars=====
void clear_key_vars(void)
{
    key_main_access = 1U; /* clear all previous key variables & start again */
    key_state = 0U;
    old_col = 1U;
    new_col = 1U;
    new_row = 1U;
    main_read = 0U;
    main_interrupted = 0U;
    key1 = no_key;
    key2 = no_key;
    key_main_access = 0U;
}

//=====init_all_vars__key_c=====
void init_all_vars__key_c(void)
{
    key_main_access = 0U; /* init ll vars to default state */
    key_state = 0U;
    old_col = 1U;
    new_col = 1U;
    new_row = 1U;
}

```

```
main_read = 0U;  
main_interrupted = 0U;  
key1 = no_key;  
key2 = no_key;  
}
```