

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
UNIVERSITE MOULOUD MAMMARI DE Tizi-Ouzou



FACULTE DU GENIE ELECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT D'ELECTRONIQUE

Mémoire de Fin d'Etudes de MASTER PROFESSIONNEL
Domaine : Sciences et Technologies
Filière : Génie électrique
Spécialité : Electronique industrielle

Présenté par M^{elle} : TAKILT HAKIMA
M^{elle} : SEBAOUI FERROUDJA

Thème

**Conception et réalisation d'un système antivol à base
d'une carte Arduino**

Soutenu publiquement le : 27 / 09 /2016 Devant le jury :

M^r. Mourad Lazri Maitre de conférences classe A à l'U.M.M.T.O Président

M^r. Fethi Ouallouche Maitre de conférences classe B à l'U.M.M.T.O Encadreur

M^r. Slimane Hameg Maitre Assistant de classe A à l'U.M.M.T.O Examineur

M^{elle}. Samia Slimani Maitre de conférences classe B à l'U.M.M.T.O Examineur

Promotion : 2015 / 2016

Remerciement

- ❖ *Nous remercierons d'abord le **bon dieu** tout puissant qui nous à procuré le courage et la volonté pour la réalisation de ce travail.*
- ❖ *Nous tenons à remercier notre promoteur **M^r** : **F.OUALLOUCHE** pour avoir accepté de diriger notre travail et pour toute l'aide et les conseils qui nous a prodiges.*
- ❖ *Nous remercierons tous les **enseignants** qui nous on encouragés et conseillés durant cursus universitaire*
- ❖ *Nous remercions aussi la maison **TIZI-TRONIC***
- ❖ *Nous tenons également à remercier les **membres du jury** d'avoir pris le temps de juger notre travail et une pensée à tous les enseignants qui ont de près ou de loin contribué à notre formation.*

Dedicaces

Je dédie ce modeste travail à :

- ❖ *ma très chère mère et mon cher père*
- ❖ *mes frères et sœurs ainsi a son marie sans oublie notre adorable rayane et nesrine et mouhamedsaid*
- ❖ *tous mes ami (e) s*

Takilt Hakima

Je dédie ce modeste travail à :

A mes parents pour leurs amour inestimable, leurs confiance, leurs soutien, Leurs sacrifices et leurs encouragements.

A ma très chère famille à laquelle je dois tout.

A mes très chers amis (es).

A tous les enseignants qui ont contribué à ma réussite durant mon parcours Educatif.

A tous mes sœurs et mes frères.

A mon fiançailles et toute sa famille.

Sebaoui ferroudja

Sommaire

Introduction	1
Chapitre I : généralit sur la carte Arduino	
I-1.Préambule	2
I-2. Historique de projet Arduino.....	2
I-3.Definition de la carte Arduino	3
I-4. Les types des cartes Arduino.....	6
I-5. Les diffirentes cartes Arduino.....	6
I-6. les principaux avantages du système Arduino.....	8
I-7. les étapes d'utilisation de la carte Arduino.....	9
I-8. Programmation de la carte Arduino	9
I-8-1.Interface de programmation	10
I-8-2. structure d'un programme.....	13
I-8-3. Bibliothèques de programmation	15
I-9. Domaines d'utilisation de la carte Arduino	15
I-10. protection de la carte Arduino	16
I-11. protection du port USB.....	16
I-12. protection des entrées numériques	16
I-13. protection des compsants.....	17
I-14. la carte Arduino Uno.....	17
I-14-1. Définition.....	17
I-14-2. Description de la carte Arduino Uno.....	17
I-14- 3. Caractéristiques principales.....	19
I-14-4. Schéma structurel de la carte Arduino Uno	19
I-14-5. Alimentation de la carte Arduino Uno	20
I-14-6. Les entrées/sorties	21
I-14-7. la communication	24
I-14-8. Gestion des mémoires.....	25
I-14-9.Dimensions de la carte	25

I-14-10. protection du port USB contre la surcharge en intensité	25
I-14-11. Le microcontrôleur Atmel ATmega328	26
I-14-12. Les caractéristiques principales de l'ATmega328.....	26
I-14-13. Architecteur interne de l'Atmega328	29
Discussion	30

Chapitre II : généralit sur les capteurs

II-1. Préambule.....	31
II-2. Les capteurs.....	31
II-2-1. Définition	31
II-2-2.Les principes de mesure d'un capteur	32
II-2-3.les caractéristiques d'un capteur.....	33
II-2-4. Les types d'erreurs classiques	33
II-2-5. Les différentes familles de capteurs	36
II-2-6. Classification selon les signaux	38
II-2-7. Caractéristiques d'une chaine de mesure numérique	41
II-2-8. Les critères de choix d'un capteur	42
II-2-9. Caractéristiques métrologiques	43
II-2-10. Grandeurs d'influence.....	44
II-2-11. Le capteur a effet Hall	45
II-11-1. Définition	45
II-11-2. Principe de fonctionnement	45
II-11-3. un capteur effet Hall UGN3503	46
II-11-4.Caractéristiques	47
II-11-5.Utilisation	47
Discussion	47

Chapitre III : Réalisation pratique

III-1. Préambule	48
III-2. Matériel à utiliser	48
III-3. Schéma Bloc	49
III-4. Tracé du circuit imprimés et schéma électrique	50
III-5. Fonctionnement du schéma électrique	51
III-6. Organigramme global du système	52
III-7. Le schéma général de la réalisation	53
Discussion	54
Conclusion	55

Liste des figures

Figure-1.1 : Exemple de la carte électronique.

Figure 1- 2 : la fenêtre principale de l'environnement de programmation Arduino .

Figure 1- 3 :La carte Arduino Mega.

Figure1- 4 : La carte Arduino DUE

Figure1- 5 : la carte Arduino nano

Figure1- 6 : les différentes parties de la fenêtre principale du logiciel Arduino

Figure1- 6-1 : La barre des Menus.

Figure1-6-2 : la barre des boutons

Figure 1-7 : structure d'un programme

Figure 1-8 : Description de la carte Arduino Uno

Figure 1-9 : structurel de la carte Arduino Uno

Figure 1-10 : Brochage de la carte Arduino Uno

Figure 1-11 : le microcontrôleur ATmega328

Figure 1-12 : Architecteur interne de l'Atmega328

Figure 2-1 : un capteur Figure2- 3-a : l'erreur de zéro

Figure2- 3-b : L'erreur d'échelle

Figure 2-3-c : L'erreur de linéarité

Figure 2-3-d: L'erreur due au phénomène d'hystérésis

Figure2- 3-e: L'erreur de quantification

Figure2- 4 : un capteur de composition.

Figure 2-5 : le signal TOR

Figure 2-6-a : le signal continu

Figure 2-6-b : le signal formel.

Figure2- 6-c : le signal fréquentiel

Figure2-7-a : le signal train d'impulsion

Figure 2-7-b : le signal d'échantillonnage

Figure 2-7 : la chaîne de mesure.

Figure 2-8: le principe de fonctionnement d'un capteur à effet Hall.

Figure2-9 : un capteur effet Hall UGN3503

Figure 3-1 : circuit imprimé

Figure3- 2 : le schema electrique

Figure3-3: Schéma Bloc

Figure 3- 3 : Organigramme global du système

Figure 3-5 : le schéma général de la réalisation



Introduction

INTRODUCTION

L'évolution de l'électronique a produit un développement considérable en ce qui concerne le mode de vie de l'humanité. En effet, l'électronique a permis de faciliter pratiquement toutes les tâches qui étaient difficiles à accomplir autrefois grâce à des circuits programmables tels que les microcontrôleurs [1].

Le microcontrôleur peut être programmé pour analyser et produire des signaux électriques, de manière à effectuer des tâches très diverses comme la domotique. La carte Arduino peut être utilisée pour construire des objets interactifs indépendants (prototypage rapide), ou bien peut être connecté à un ordinateur pour communiquer avec des logiciels et ainsi effectuer la tâche voulue.

Dans le domaine de la surveillance et la sécurité électronique, plusieurs circuits et montages sont proposés, réalisés et commercialisés à grande échelle ces dernières années. Ces systèmes utilisent des principes de fonctionnement différent selon le but recherché.

Toutefois, le composant clé d'un tel système est le capteur utilisé. En effet, dans les systèmes largement commercialisés, on utilise un capteur infrarouge ou sonore. Afin de tester l'utilisation l'effet Hall, nous avons intégré ce capteur dans une carte programmable.

Le système que nous proposons est un système antivol compact qui peut protéger une maison ou un véhicule, il offre la possibilité de déjouer des tentatives d'effraction.

Afin de présenter notre travail, nous avons divisé le présent mémoire en trois chapitres.

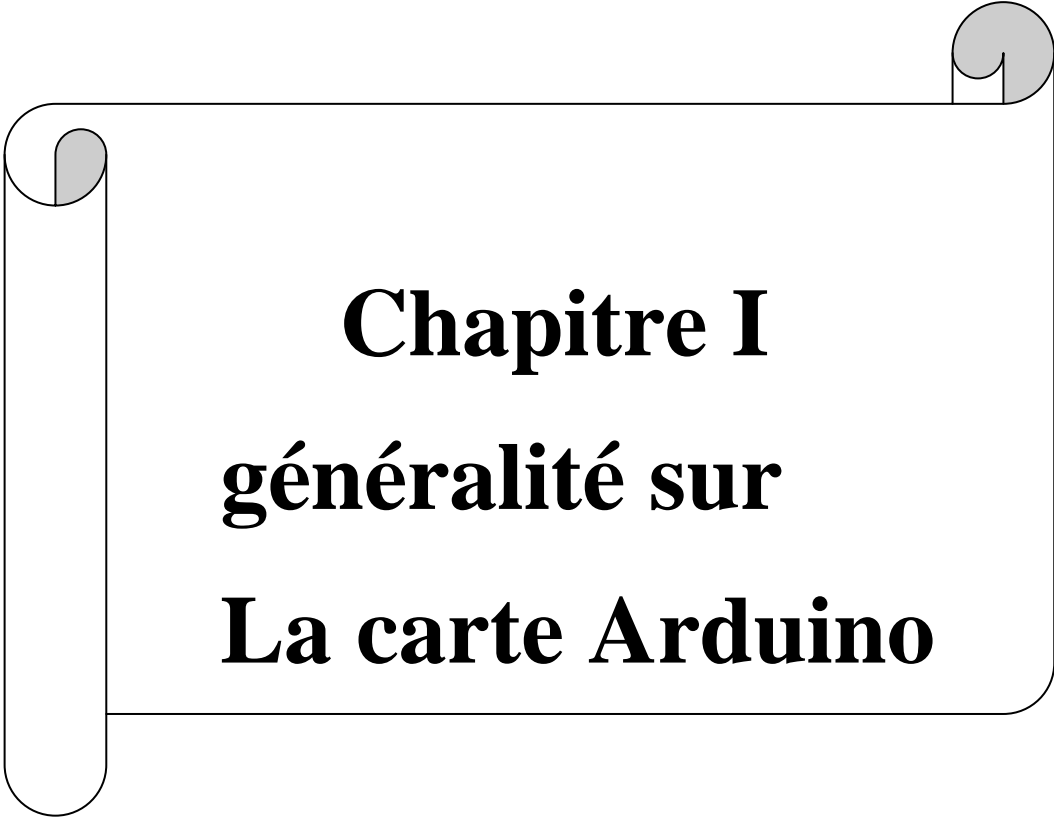
Le premier chapitre est consacré aux généralités sur la carte Arduino.

INTRODUCTION

Dans le deuxième chapitre nous présenterons des généralités sur les capteurs puis une étude sur le capteur a effet Hall qui est utilisée dans notre réalisation.

Dans le troisième chapitre, nous exposerons la réalisation pratique du système et son fonctionnement.

Enfin, nous terminons le présent mémoire par une conclusion générale et une bibliographie.



Chapitre I
généralité sur
La carte Arduino

I-1.Préambule :

Arduino est une petite carte électronique programmable à l'aide d'un logiciel multiplateformes, qui peut être accessible à tout un chacun dans le but de créer facilement des systèmes électroniques.

La famille Arduino est apparue en 2006 sous forme d'une carte programmable avec un port série. Puis de nouvelles versions sont apparues, avec différents formats : petites, grandes, plus puissantes, plus économiques, connexion USB, Ethernet[2], ...etc .

Ces cartes peuvent être utiliser pour récupérer les informations de capteurs, contrôler des moteurs, communiquer avec un ordinateur ou un téléphone portable, envoyer et lire des messages sur Internet et encore bien d'autres usages.

I-2. Historique de projet Arduino : [3]

Le projet Arduino est issu d'une équipe d'enseignants et d'étudiants de l'école de Design d'Interaction d'Ivrea (Italie). Ils rencontraient un problème majeur à cette période (avant 2003- 2004) : les outils nécessaires à la création de projets d'interactivité étaient complexes et onéreux. Ces coûts souvent trop élevés rendaient difficiles le développement par les étudiants de nombreux projets et ceci ralentissait la mise en œuvre concrète de leur apprentissage.

En 2003, Hernando Barragan, pour sa thèse de fin d'études, avait entrepris le développement d'une carte électronique dénommée Wiring, accompagnée d'un environnement de programmation libre et ouvert. Pour ce travail, Hernando Barragan réutilisait les sources du projet Processing. Basée sur un langage de programmation facile d'accès et adaptée aux développements de projets de designers, la carte Wiring a donc inspiré le projet Arduino (2005).

Massimo Banzi enseigne dans une école de Design à Ivrea en Italie, et souvent ses étudiants se plaignent de ne pas avoir accès à des solutions bas prix pour accomplir leurs projets de robotique. Banzi en discute avec David Cuartielles, un ingénieur Espagnol spécialisé sur les microcontrôleurs... Ils décident de créer leur propre carte en embarquant dans leur histoire un des étudiants de Banzi, David Mellis qui sera chargé de créer le langage de programmation allant avec la carte. En deux jours David écrira le code. Trois jours de plus et la carte était créé... Ils décidèrent de l'appeler Arduino (un bar fréquenté par les élèves à proximité de l'école)...

Il devient un hit tout de suite auprès des étudiants. Tout le monde arrive à en faire quelque chose très rapidement sans même avoir de connaissances particulière ni en électronique ni en informatique. Investissent 3000 euros pour créer les premiers lots de cartes:

Les 50 premières partent directement à des élèves de l'école. En 2006, 5 000 cartes vendues... En 2007 plus de 30 000, En 2011 plus de 120 000, sans compter les clones.

Arduino est aussi le nom d'un roi italien, personnage historique de la ville « Arduin d'Ivrée », ou encore un prénom italien masculin qui signifie « l'ami fort ».

I-3. Définition de la carte Arduino :

Le système Arduino est une plate-forme open-source d'électronique programmée qui est basée sur une simple carte à microcontrôleur (de la famille AVR) comportant un certain nombre d'entrées et de sorties (les ports) permettant la connexion de capteurs, ou d'actionneurs, et un logiciel, véritable environnement

du développement intégré, pour écrire, compiler et transférer le programme vers la carte à microcontrôleur.

Arduino est une plateforme matérielle et logicielle de développement d'applications embarquées.

- **Le côté matériel** : L'Arduino est un circuit imprimé composé de plusieurs composants électroniques, et permettant de recevoir, d'analyser et de produire des signaux électriques. Sa simplicité et son prix réduit le rend accessible à n'importe qui désireux de se lancer dans l'électronique ou voulant développer des composants électriques avancés.



Figure 1-1 : Exemple de la carte électronique.

- **le logiciel** : Le logiciel de programmation des modules Arduino est une application JAVA[4], libre et multi-plateformes, Le langage de programmation utilisé est un mélange de C et C++, est un langage permettant à un être humain d'écrire un ensemble d'instructions (code source) qui seront directement converties en langage machine grâce à un compilateur, l'exécution d'un programme s'effectue de manière séquentielle.

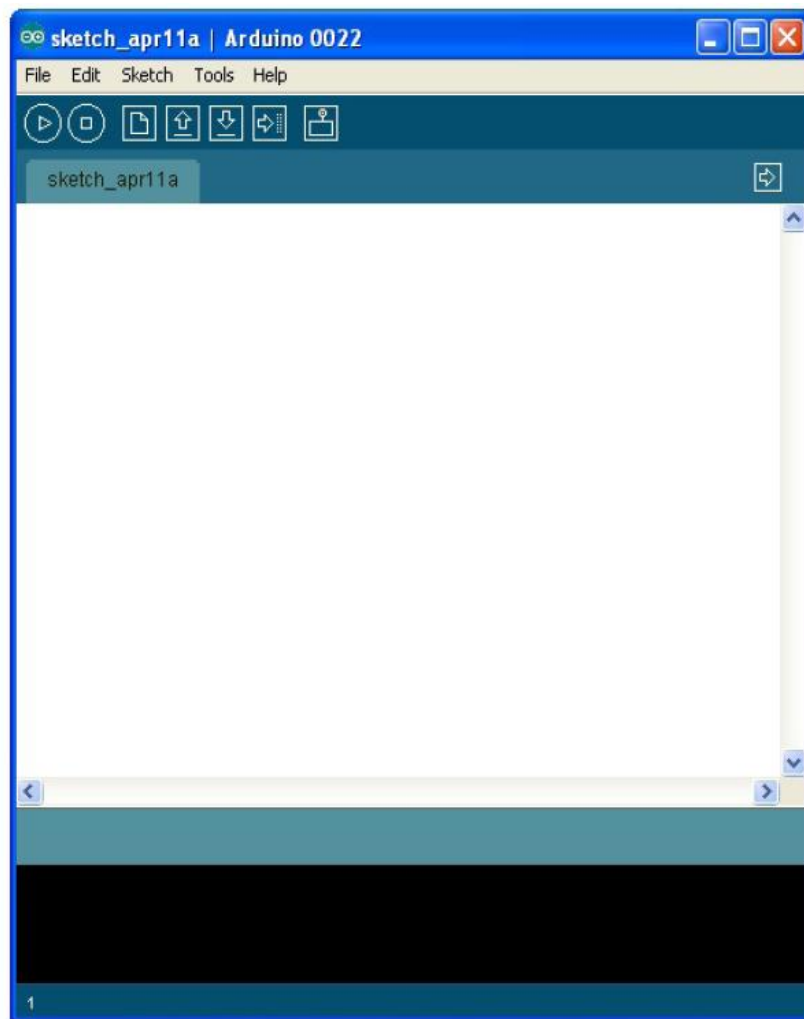


Figure 1- 2 : la fenêtre principale de l'environnement de programmation Arduino .

I-4. Les types des cartes Arduino : [5]

Il y a trois types de cartes :

- «Officielles » qui sont fabriquées en Italie par le fabricant officiel : Smart Projects
- « Compatibles » qui ne sont pas fabriqués par Smart Projects, mais qui sont totalement compatibles avec les Arduino officielles.
- Les « autres » fabriquées par diverse entreprise et commercialisées sous un nom différent (Freeduino, Seeduino, Femtoduino, ...).

I-5. Les différentes cartes Arduino :[6]

Il existe plusieurs version des carte arduino par exemple :

➤ **La carte Arduino MEGA :**

La carte ARDUINO MEGA est la carte la plus diffusée après la carte ARDUINO UNO. Elle offre un nombre d'entrées/sorties beaucoup plus important (54 contre 14), un processeur plus puissant doté d'une mémoire plus vaste qui permet d'exploiter des algorithmes plus complexes.

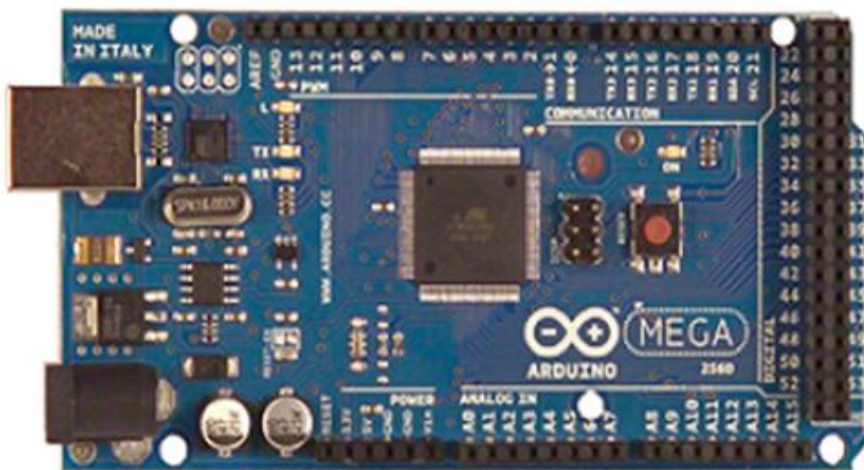


Figure 1- 3 :La carte Arduino Mega.

➤ **La carte Arduino DUE :**

La carte ARDUINO DUE est une évolution de la carte ARDUINO MEGA et offre des performances réputées 3 fois supérieures. Elle permet de manipuler rapidement des algorithmes lourds particulièrement utiles dans le monde de la robotique par exemple.

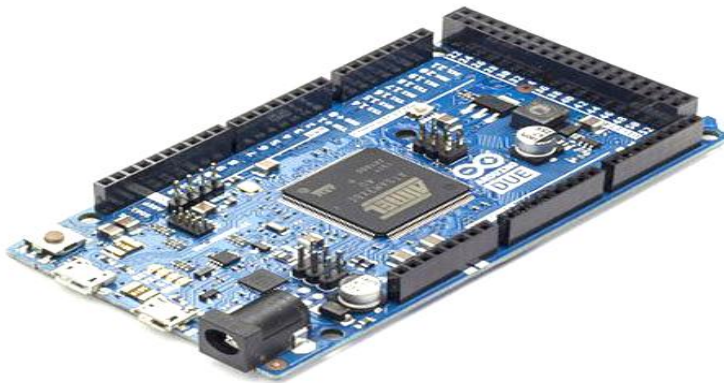


Figure1- 4 : La carte Arduino DUE .

➤ **La carte Arduino NANO :**

La carte ARDUINO NANO la même qu'une carte ARDUINO UNO miniaturisée. Sa taille et son poids réduits la destinent à une utilisation dans des espaces réduits (en textile par exemple) ou dans des applications de robotique ou de modélisme pour lesquels le poids et la taille sont des facteurs déterminant (hélicoptères, drones...)



Figure1- 5 : la carte Arduino nano.

I-6. les principaux avantages du système Arduino :[7]

- **matériel peu onéreux** : les carte Arduino sont relativement peu couteuses comparativement aux autres plateformes de développement.
- **matériel Open source et extensible** : les schémas des modules sont publiés sous une licence Créative Commons, et il est possible de réaliser nos propres versions des cartes Arduino, en les complétant et en les améliorant.
- **logiciel gratuit et Open Source** : le logiciel Arduino et le langage Arduino sont publiés sous licence open source, disponible pour être complété par des programmeurs expérimentés. Le langage peut être aussi étendu à l'aide de bibliothèques C++.
- **logiciel multi-plateforme** : le logiciel Arduino, écrit en JAVA, tourne sous les systèmes d'exploitation Windows, Macintosh et Linux.
- **modules « shield »** : cartes supplémentaires se connectant sur le module Arduino pour augmenter les possibilités : afficheur graphique couleur, interface ethernet,...
- **pas cher**: les cartes Arduino sont relativement peu coûteuses comparativement aux autres plateformes.
- **un environnement de programmation clair et simple**: l'environnement de programmation Arduino est facile à utiliser pour les débutants, tout en étant assez flexible pour que les utilisateurs avancés puissent en tirer profit également.
- **Récupère les données des capteurs** pour les transmettre à l'interface de commande (pc) et traduire les instructions pour faire fonctionner les actionneurs.

I-7. les étapes d'utilisation de la carte Arduino:

Dans le cas de l'utilisation de la carte Arduino dans un circuit électronique, on suit les étapes suivantes :

- On conçoit ou on ouvre un programme existant avec le logiciel Arduino.
- On vérifie ce programme avec le logiciel Arduino (compilation).
- Si des erreurs sont signalées, on modifie le programme.

- On charge le programme sur la carte.

- On câble le montage électronique.

- L'exécution de programme est automatique après quelques secondes.

- On alimente la carte soit par le port USB, soit par une source d'alimentation autonome (pile 9 volts par exemple).

- On vérifie que notre montage fonctionne.

I-8. Programmation de la carte Arduino :[8]

I-8-1.Interface de programmation :

Le logiciel Arduino a pour fonctions principales :

- De pouvoir écrire et compiler des programmes pour la carte Arduino
- De se connecter avec la carte Arduino pour y transférer les programmes
- De communiquer avec la carte Arduino

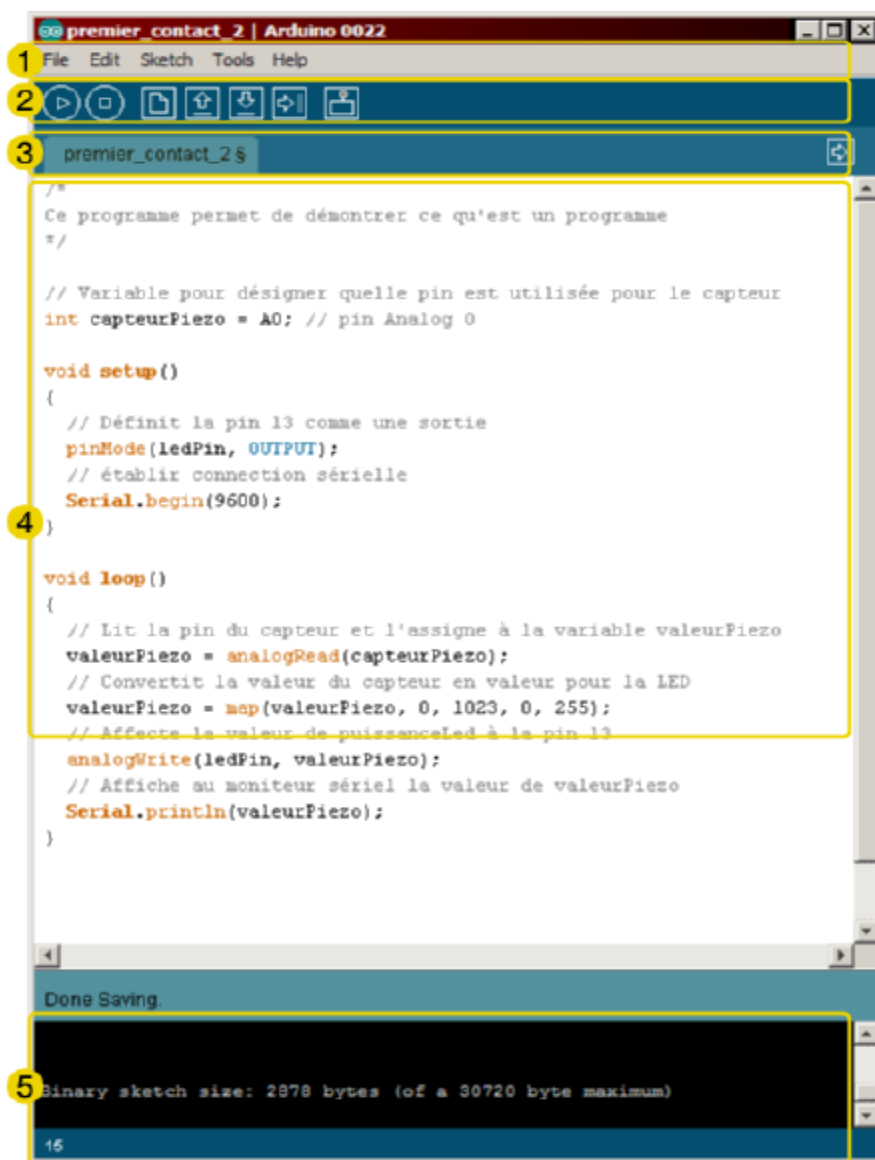


Figure1- 6 : les différentes parties de la fenêtre principale du logiciel

Arduino.

La fenêtre de l'application Arduino comporte les éléments suivants

1) La barre des Menus :

C'est principalement ce menu que l'on va utiliser le plus. Il dispose d'un certain nombre d'applications très utiles. Il a été traduit en français progressivement.

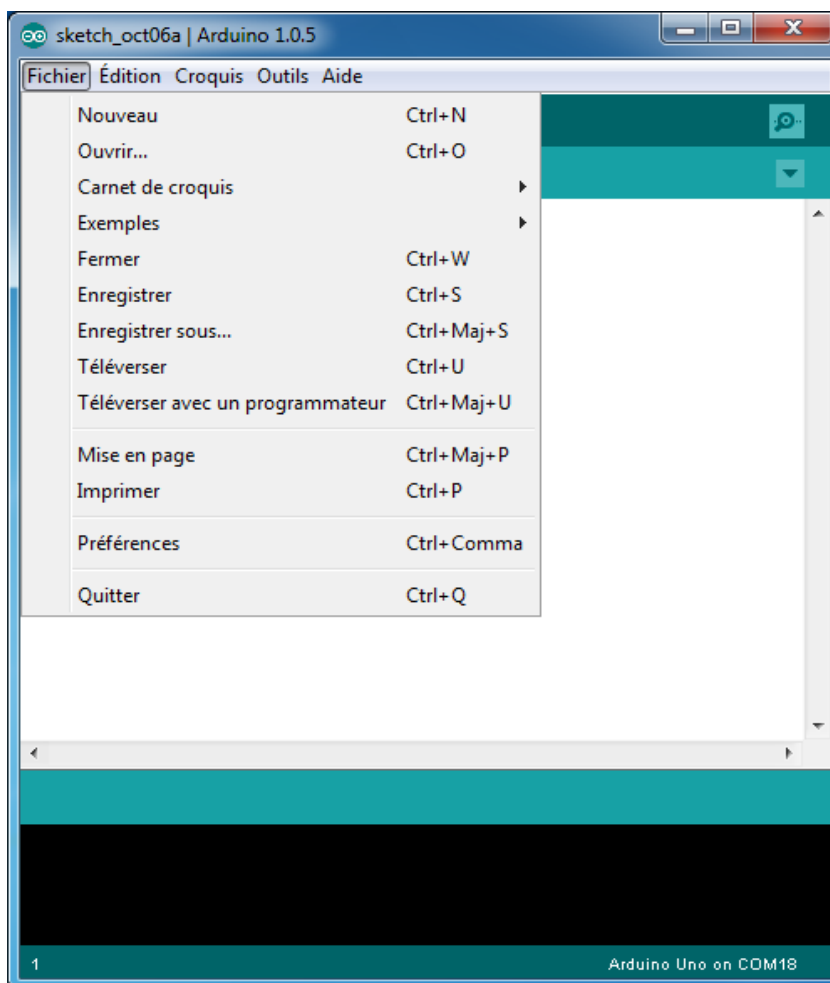


Figure1- 6-a : La barre des Menus.

Cette barre contient les icônes suivantes :[9]

- **File (Fichier) :** ce menu contient les différentes options de création, d'ouverture, de sauvegarde, d'impression d'un programme, ou l'ouverture d'un exemple parmi les exemples qui accompagnent le logiciel Arduino.

- **Edite (Editer) :** ce menu contient les options de copier/coller, sélection, et les options de recherche.
- **Sketch (programme ou séquence) :** ce menu contient les différentes fonctions de la barre des boutons, ainsi que les options d'ajout de bibliothèques ou de fichiers.
- **Tools (Outils) :** c'est dans ce menu qu'on sélectionne le type de carte à programmer, et le port série utilisé ainsi que la fonction de chargement du bootloader dans l'ATmega.
- **Help(Aide) :** ce menu est fait pour donner de l'aide concernant les différents problèmes rencontrés au niveau du logiciel Arduino.

2) la barre des boutons :[10]

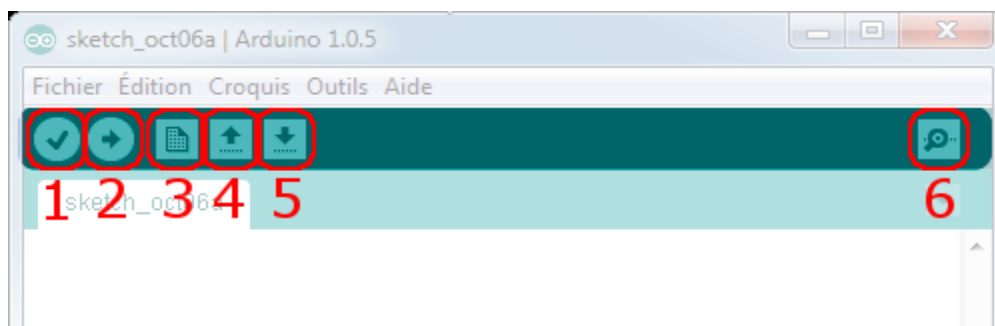


Figure1-6-b: la barre des boutons.

- **vérifier/compiler (Bouton 1) :** Ce bouton permet de vérifier et compiler s'il ya des erreurs dans le programme en sélectionnant la ligne où se trouve l'erreur.
- **téléverser (Bouton 2) :** transférer le programme vers la carte Arduino
- **Nouveau (Bouton 3) :** en appuyant sur cette icône, une fenêtre d'édition vide s'ouvre pour la saisie d'un nouveau programme
- **Ouvrir (Bouton 4) :** il suffit de cliquer sur ce bouton pour accéder à la liste de tous les programmes qui se trouvent dans le "livre de

programmes “.cliquer sur l’un des programmes l’ouvre dans la fenêtre courante.

- **Enregistrer (Bouton 5)** : avant de quitter le programme saisi, il faut l’enregistrer, et ca peut se faire en cliquant sur cette icône.
- **Moniteur série (Bouton 6)** : Ouvre le moniteur série il suffit d’appuyer sur ce bouton, ce moniteur est utilisé pour la visualisation ou les données (message, valeurs, caractères) transmises vers l’ordinateur via le câble USB connecté à la carte Arduino et envoyer des caractères vers la carte Arduino.

3) un ou plusieurs ongles correspondant aux sketches

4) **un Editeur** : c’est dans cette espace ou’ on va écrire notre programme.

5) **une Console Texte**: elle permet d’afficher les informations et erreurs de compilation et de léléversement du programme.

I-8-2. structure d’un programme : Un programme utilisateur Arduino est suite d’instructions élémentaires sous forme textuelle, ligne par ligne.la carte lit puis effectue les instructions les unes après les autres, dans l’ordre défini par lignes de code. Comporte trois parties :

➤ **la définition des constantes et de variables :**

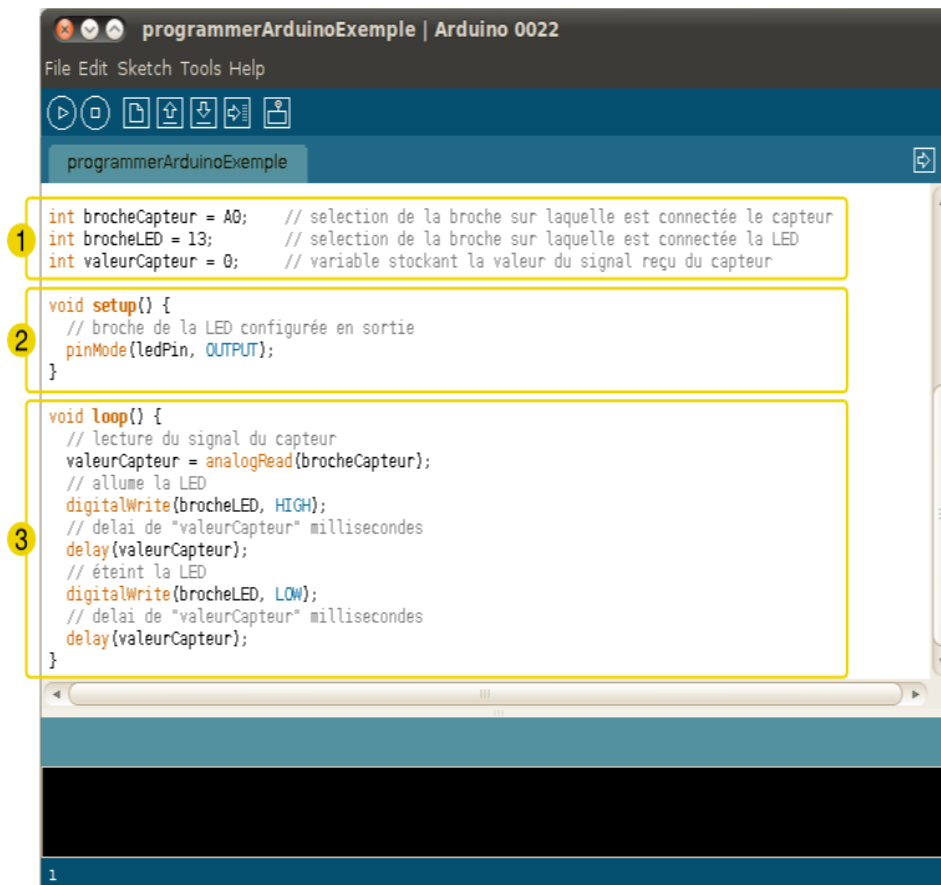
Elle permet également l’inclusion des bibliothèques utilisées dans le programme au moyen de # include.

➤ **la configuration des entrées et sorties (void stup()) :[11]**

la fonction stup() contient les instructions d’initialisation ou de configuration des ressources de la carte comme par exemple, la configuration en entrée ou sorties des broches d’E /S numériques, la définition de la vitesse de communication de l’interface série,etc...cette fonction n’est exécutée qu’ une seule fois juste après le lancement du programme.

➤ **la programmation des interactions et comportements (viod loop()) :[12]**

La fonction loop() contient les instructions du programme à proprement parlé. Cette fonction sera répétée indéfiniment tant que la carte Arduino restera sous tension.



```
programmerArduinoExemple | Arduino 0022
File Edit Sketch Tools Help
programmerArduinoExemple

1 int brocheCapteur = A0; // selection de la broche sur laquelle est connectée le capteur
  int brocheLED = 13; // selection de la broche sur laquelle est connectée la LED
  int valeurCapteur = 0; // variable stockant la valeur du signal reçu du capteur

2 void setup() {
  // broche de la LED configurée en sortie
  pinMode(ledPin, OUTPUT);
}

3 void loop() {
  // lecture du signal du capteur
  valeurCapteur = analogRead(brocheCapteur);
  // allume la LED
  digitalWrite(brocheLED, HIGH);
  // delai de "valeurCapteur" millisecondes
  delay(valeurCapteur);
  // éteint la LED
  digitalWrite(brocheLED, LOW);
  // delai de "valeurCapteur" millisecondes
  delay(valeurCapteur);
}

1
```

Figure 1-7 : structure d'un programme .

Pour place des commentaires sur une ligne ou en fin de ligne, il faut utiliser la syntaxe suivante :

//cette ligne est un commentaire sur une seule ligne

-Pour place des commentaires sur plusieurs lignes :

/* commentaires, sur plusieurs lignes qui sera ignoré par le programme, mais pas par celui qui lit le code*/

I-8-3. Bibliothèques de programmation :[13]

Une bibliothèque set un ensemble de fonctions utilitaires mises à disposition des utilisateurs de l'environnement Arduino. Les fonctions sont regroupées en fonction de leur appartenance à un même domaine conceptuel (mathématique, graphique, tris, etc). L'environnement de programmation Arduino (l'IDE) comporte par défaut plusieurs bibliothèques externes. L'instruction suivante sera ajouté au début de programme. #include <la_bibliothèque.h>

I-9. Domaines d'utilisation de la carte Arduino :[14]

La carte Arduino est utilisée dans divers domaines telles que :

- **Prototypage rapide de projets innovants utilisant l'électronique :** Arduino facilitant l'expérimentation en amont de la phase d'industrialisation.
- **Production artisanale d'objets numériques et de machines-outils à faible cout** dans la perspective d'une culture d'appropriation technologique favorisant le bricolage et la débrouille.
- **Captation et analyse de données scientifiques** (environnement, énergie, etc..) à des fins éducatives, de recherche ou d'appropriation citoyenne.
- **Spectacle vivant :** grâce aux nombreuses fonctions d'interaction offertes par Arduino, il est possible de créer des performances de VJing. Utiliser le mouvement des danseurs pour générer en temps réel des effets sonores et visuels dans un spectacle.
- **Installations d'arts numériques :** Arduino permettant de réaliser des œuvres d'art interagissant de manière autonome avec le public.
- **Mode et design textile :** plusieurs stylistes et designers investissant ce domaine créatif en exploitant les possibilités offertes par l'intégration de l'électronique notamment dans des vêtements(e-textile).

- **Projets pédagogiques :** à destination d'étudiants, de professionnels ou du grand public selon les porteurs de ces initiatives : écoles supérieures, centres de formation spécialisée ou des Media Labs.

I-10. protection de la carte Arduino :[15]

Chacune des entrées /sorties de la carte Arduino ne peut pas délivrer plus de 20 mA. cette quantité de courant est relativement faible mais permet, par exemple, de contrôler une diode électroluminescente LED ainsi que des actionneurs de faible puissance tel qu' un piézoélectrique ou encoure un petit servomoteur.

I-11. protection du port USB :[16]

Le port USB d'ordinateur délivre une quantité limitée de courant électrique. En général un port USB peut fournir au maximum 500 mA en 5v. Si on vous réaliser un montage qui nécessite plus de courant. Il s'agira de prévoir une alimentation externe suffisamment puissante.

I-12. protection des entrées numériques :[17]

Comme leur configuration dépend du programme résident dans la carte Arduino. Ce que l'on considère dans un montage comme des entrées numériques peuvent accidentellement être configurées en sortie (avec la commande pinMode) et réglées à 0 v (LOW). Dans ce cas, si cette « entrée » reçoit du 5 v,il se produira un court-circuit qui risque d'endommager partiellement ou intégralement la carte.

Pour éviter ce genre de mésaventure, lorsqu'on cable une entrée numérique, il est prudent d'y ajouter une résistance de 100 Ω qui limite le courant en cas de fausse manœuvre.

I-13. protection des composants :[18]

Chaque composant possède ses propres conventions d'utilisation. Par exemple , il existe des composants qui possèdent un sens de branchement à respecter, l'on dit que ces composants sont polarisés, c'est le cas des LEDs, de certains condensateurs, des diodes,ect...

La plupart des composants ne peuvent pas fonctionner seuls, par exemple une LED a besoin d'une résistance appropriée pour ne pas s'user ou « bruler ».cette résistance permet de limiter le courant qui traverse la LED. Le courant supprimé est alors dissipé en chaleur par la résistance.

I-14. la carte Arduino Uno :**I-14-1. Définition :[19]**

La carte Arduino Uno est basée sur un ATmega328 cadencé à 16 MHz. C'est la plus récente et la plus économique carte à microcontrôleur d'Arduino. Des connecteurs situés sur les bords extérieurs du circuit imprimé permettent d'enficher une série de modules complémentaires.

Elle peut se programmer avec le logiciel Arduino. Le contrôleur ATmega328 contient un bootloader qui permet de modifier le programme sans passer par un programmeur.

I-14-2. Description de la carte Arduino Uno :[20]

Le modèle Uno de la société Arduino est une carte électronique dont le noyau est un microcontrôleur ATMEL de référence ATmega328. Le microcontrôleur ATmega328 est un microcontrôleur 8 bits de la famille AVR dont la programmation peut être réalisée en langage C. cette carte possède 14 entrées/sorties numériques (dont 6 peuvent être utilisées comme étant des sorties PWM (Pulse Width Modulation)), 6 entrées analogiques avec un convertisseur Analogique /Numérique de 10 bits de résolution, 1 résonateur céramique (quartz) de 16 MHz, 1 connecteur ICSP(In Circuit Serial Programmation) qui permet

d'injecter le bootloader à l'intérieur du microcontrôleur, 1 connecteur jack pour une alimentation extérieur, un bouton de reset pour mettre le processus à zéro.

L'avantage de cette carte c'est qu'elle n'a pas besoin de pilote pour faire la conversion FTDI USB/Série, elle a juste un petit microcontrôleur ATmega8 (pour la version 2) programmé comme convertisseur USB/Série.

Elle contient tout ce que le microcontrôleur a besoin pour fonctionner, il faut seulement la connecter avec un câble USB à un ordinateur ou avec une alimentation externe pour commencer. la figure suivante représente les différents éléments de la carte Arduino Uno :

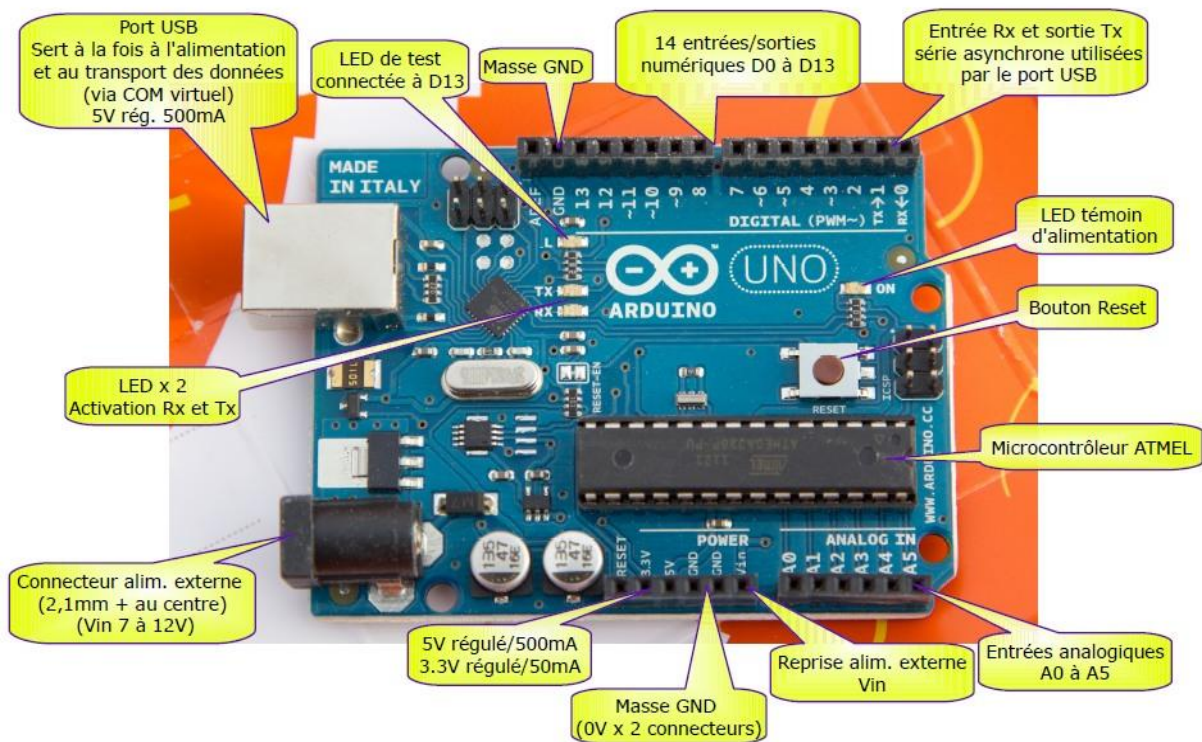


Figure 1-8 : Description de la carte Arduino Uno.

I-14- 3. Caractéristiques principales :[21]

- Tension de fonctionnement : 5v
- tension d'alimentation (recommandée): via port USB ou 7 à 12 V sur connecteur alim.
- tension d'alimentation (limites) : 6 à 20v.
- microprocesseur: ATMega328.
- d'un bouton de réinitialisation (reset).
- vitesse d'horloge 16MGz.
- d'une connexion USB.
- d'un connecteur d'alimentation jack.
- d'un connecteur ICSP (programmation "in-circuit").

I-14-4. Schéma structurel de la carte Arduino Uno :[22]

Tous les composants d'une carte électronique ne sont pas forcément reliés entre eux. Le câblage des composants suit un plan spécifique à chaque carte électronique, qui se nomme le schéma électronique :

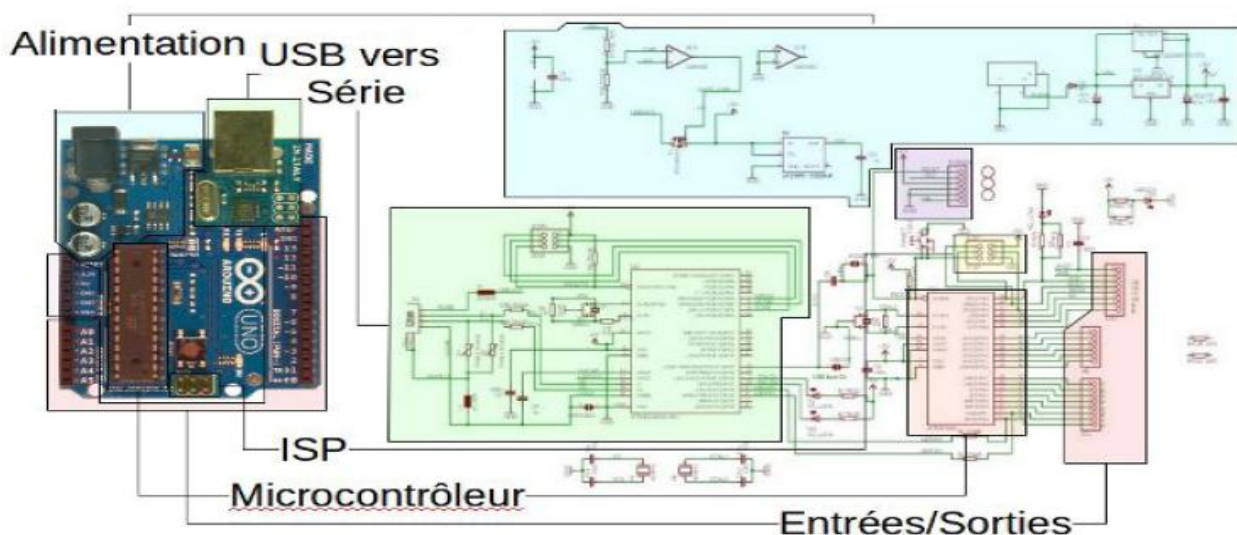


Figure 1-9 : structurel de la carte Arduino Uno.

I-14-5. Alimentation de la carte Arduino Uno : [23]

La carte Arduino Uno peut-être alimentée soit via la connexion USB (qui fournit 5V jusqu'à 500 mA) ou à l'aide d'une alimentation externe. La source d'alimentation est sélectionnée automatiquement par la carte.

L'alimentation externe (non-USB) peut être soit un adaptateur secteur (pouvant fournir typiquement de 3V à 12V sous 500 mA) ou des piles (ou des accus).

L'adaptateur secteur peut être connecté en branchant une prise de 2.1mm, dédié du pôle positif au centre dans le connecteur jack de la carte. Les fils en provenance d'un bloc de piles ou d'accus peuvent être insérés dans les connecteurs des broches de la carte appelées Gnd (masse ou 0V) et Vin (Tension positive en entrée) du connecteur d'alimentation.

La carte peut fonctionner avec une alimentation externe de 6 à 20 volts. Cependant, si la carte est alimentée avec moins de 7V, la broche 5V pourrait fournir moins de 5V et la carte pourrait être instable. Si on utilise plus de 12V, le régulateur de tension de la carte pourrait chauffer et endommager la carte. Aussi, la plage idéale recommandée pour alimenter la carte Uno est entre 7V et 12V.

➤ Les broches d'alimentation sont les suivants: [24]

- **VIN.** La tension d'entrée à la carte Arduino quand il est à l'aide d'une source d'alimentation externe (par opposition à 5 volts de la connexion USB ou une autre source d'alimentation régulée). Vous pouvez fournir une tension à travers cette broche, ou, si l'alimentation en tension par l'intermédiaire de la prise d'alimentation, y accéder via cette broche.
- **5V.** la tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (pour info : les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite " tension régulée" obtenue à l'aide d'un composant appelé un régulateur et qui est intégré

dans la carte Arduino).le 5v régulé fourni par cette broche peut donc provenir soit de la tension d'alimentation VIN via le régulateur de la carte, ou bien de la connexion USB (qui fournit du 5v régulé) ou de tout autre source d'alimentation régulée.

- **3V3.** Une alimentation de 3,3 volts générés par le régulateur à bord. Consommation de courant maximale est de 50 mA.
- **GND.** Les broches au sol.
- **IOREF.** Cette broche sur la carte Arduino fournit la référence de tension avec laquelle le microcontrôleur fonctionne. Un écran correctement configuré peut lire la tension de la broche IOREF et sélectionnez la source d'alimentation appropriée ou activer des traducteurs de tension sur les sorties pour travailler avec le 5V ou 3,3V.

I-14-6. Les entrées/sorties :[25]

La carte Arduino Uno disponible de 14E/S numériques et de 6 entrées analogiques.

➤ **Les Entrées/sorties numériques :**

Chacune des 14 broches numériques (repérées 0 à13) peut être utilisée en entrée (input) ou en sortie (output) en utilisant les instructions `pinMode()`, `digitalWrite()` et `digitalRead()` du langage Arduino. Ces broches fonctionnent en 5v. chaque broche peut fournir ou recevoir un maximum de 40 mA d'intensité et dispose d'une résistance interne "résistance de rappel (pull-up)"(déconnectée par défaut)de 20 à 50 KOhms. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction `digitalWrite (brocge,HIGH)` .

En outre, certaines broches ont des fonctions spécialisées:

- **Communication Série:** broches 0 (RX) et 1 (TX). Permet de recevoir (RX) et transmettre (TX) les données série de niveau TTL.ces broches sont connectées aux broches correspondantes du circuit intégré ATmega8U2 programmé en

convertisseur USB-vers-série de la carte (composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur). On fait appel à la transmission série à travers ces broches avec l'instruction `Serial.print()`, à condition que le câble USB soit déconnecté, sinon il va y avoir un chevauchement.

- **Interruptions externes** : broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur faible, un front montant ou descendant, ou un changement de valeur. Voir la fonction `attachInterrupt()` pour plus de détails.
- **Impulsion PWM (largeur d'impulsion modulée): broches 3, 5, 6, 9, 10 et 11.** Fournissent une impulsion PWM 8 bits à l'aide de l'instruction `analogWrite()`.
- **SPI (Interface Série Périphérique): broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la bibliothèque pour communication SPI. Les broches SPI sont également connectées sur le connecteur ICSP (In Circuit Serial Programming).
- **I2C** : broches 4(SDA) et 5(SCL), Supportent les communications de protocole I2C, disponible en utilisant la bibliothèque `Wire` /I2C.
- **LED** : Broche 13. Il y a une LED incluse dans la carte connectée à la broche 13. Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte.

➤ **Les entrées analogiques :**

La carte Uno dispose de 6 entrées analogiques(A0 marqué par A5), dont chacun fournissent 10 bits de résolution (ie 1024 valeurs différentes soit de 0 à 1023) à l'aide de la très utile fonction `analogRead()` du langage Arduino . Par défaut, ils mesurent à partir du sol(0v) à 5 volts(valeur 1023), est cependant possible de changer l'extrémité supérieure de sa gamme en utilisant la broche AREF et la l'instruction `analogReference()` du langage Arduino.

En outre, certaines broches ont des fonctionnalités spécialisées:

- **TWI: broche A4 ou SDA et A5 ou une broche SCL** .Soutien communication TWI en utilisant la bibliothèque Wire.

Il y a quelques autres broches de la carte:

- **AREF**. Tension de référence pour les entrées analogiques. Utilisé avec l'instruction `analogReference()`.s'utilise pour réduire
- **Reset** : Mettre cette broche au niveau BAS entraîne la réinitialisation (= le redémarrage) du microcontrôleur. Typiquement, cette broche est utilisée pour ajouter un bouton de réinitialisation sur le circuit qui bloque celui présent sur la carte.

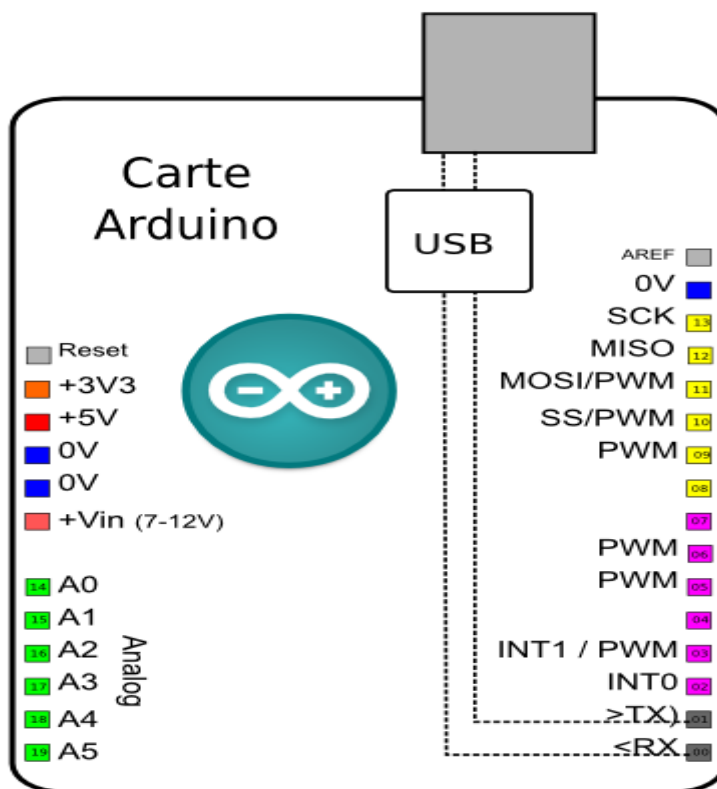


Figure 1-10 : Brochage de la carte Arduino Uno.

I-14-7. la communication :[26]

L'Arduino Uno dispose d'un certain nombre de moyens pour communiquer avec un ordinateur, un autre Arduino, ou d'autres microcontrôleurs. Le ATmega328 fournit UART (Universal Asynchronous Receiver Transmitter ou émetteur-récepteur universel asynchrone en français) pour la communication série de niveau TTL (5V) et qui est disponible sur les broches numériques 0 (RX) et 1 (TX).

Un ATmega8U2 sur la carte assure la connexion entre cette communication série vers le port USB de l'ordinateur et apparaît comme un port COM virtuel pour le logiciel sur l'ordinateur. Le code «firmware »utilisé pour programmer l'ATmega8U2 utilise le driver standard USB COM, et aucun driver externe n'est nécessaire.

Le logiciel Arduino comprend un moniteur de série qui permet aux données textuelles simples pour être envoyés vers et depuis la carte Arduino. Les voyants RX et TX sur la carte clignote lorsque des données sont transmises via le circuit intégré USB-vers-série et la connexion USB vers l'ordinateur (mais pas pour la communication série sur les broches 0 et 1).

Une bibliothèque SoftwareSerial permet une communication série sur l'une des broches numériques de la Uno.

Le ATmega328 supporte également la communication par protocole I2C (TWI) et SPI :

- Le logiciel Arduino inclut la librairie **Wire** qui simplifie l'utilisation du bus I2C.
- Pour utiliser la communication SPI (Interface Série Périphérique), la librairie pour communication SPI est disponible, il suffit de la faire inclure dans le programmer au niveau du logiciel Arduino lors de la programmation.

I-14-8. Gestion des mémoires :[27]

Le microcontrôleur ATmega328 dispose de 32 ko de mémoire de programme Flash(dont 0.5 ko également utilisés par le bootloader). Il contient aussi de 2ko de mémoire vive (SRAM). Cette mémoire est généralement utilisée pour stocker les résultats temporaires lors de calculs. Elle peut être lue et écrite à tout instant par le microcontrôleur mais son contenu est perdu dès que la carte n'est plus alimentée. L'ATmega328 dispose également 1KO mémoire EEPROM. Le contenu de cette mémoire est accessible grâce aux fonctions de la librairie <<EEPROM>>.

I-14-9. Dimensions de la carte :[28]

Les longueurs et largeurs maximales de la carte Arduino Uno sont respectivement 6.86 cm et 50.33 cm, avec le connecteur USB et le connecteur d'alimentation Jack s'étendant au-delà des dimensions de la carte. Quatre trous de vis permettent à la carte d'être fixée sur une surface ou dans un boîtier (pour l'embarquer sur un système). Noter que la distance entre les broches 7 et 8 est de 0.16 pouces, et 0.1 pouces séparant les autres broches.

I-14-10. protection du port USB contre la surcharge en intensité :[29]

La carte Arduino Uno intègre un poly-fusible réinitialisable qui protège le port USB de l'ordinateur contre les surcharges en intensité (le port USB est généralement limité à 500 mA en intensité). Bien que la plupart des ordinateurs aient leur propre protection interne, le fusible de la carte fournit une couche supplémentaire de protection. Si plus de 500mA sont appliqués au port USB, le fusible de la carte coupera automatiquement la connexion jusqu'à ce que le court-circuit ou la surcharge soit stoppé.

I-14-11. Le microcontrôleur Atmel ATmega328 :[30]

Le microcontrôleur utilisé sur la carte Arduino Uno est un microcontrôleur ATmega328.c'est un microcontrôleur ATMEL de la famille AVR 8bits.

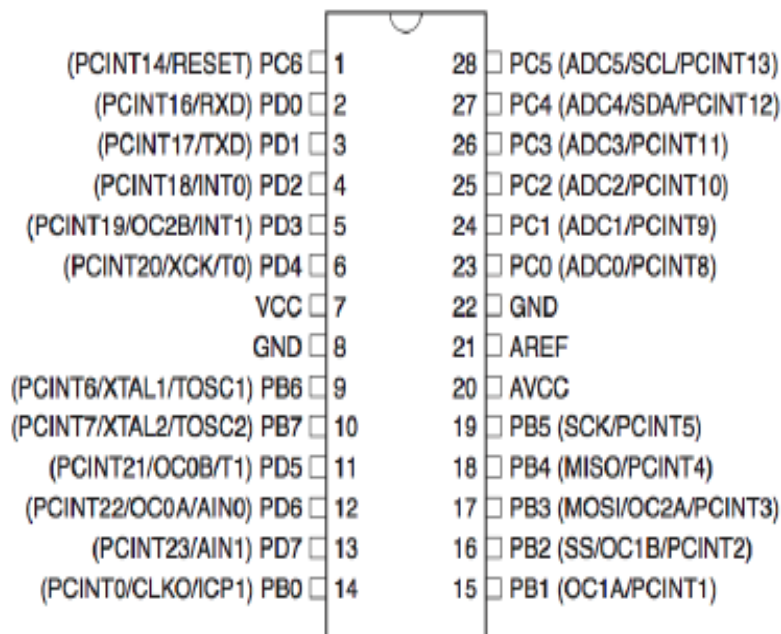


Figure 1-11 : le microcontrôleur ATmega328.

I-14-12. Les caractéristiques principales de l'ATmega328 :[31]

Le microcontrôleur ATmega328 de Atmel dispose :

- de 14 broches numériques d'entrées/sorties (dont 6 peuvent être utilisées en sorties PWM (largeur d'impulsion modulée)).réparties selon l'ordre suivant : OC0A(PD6), OC0B(PD5), OC1A(PB1), OC1B(PB3), OC2A(PB3), OC2B(PD3), et 2 (0et1) pour réception/émission série).
- de 6 entrées analogiques 10 bits (qui peuvent également être utilisées en broches entrées/sorties numériques).ces entrées/sorties sont réparties sur trois ports :PortB, PortC, PortD(23 broches E/S en tout).
- intensité max disponible par broche E/S (5v): 40 mA (200mA cumulé pour l'ensemble des broches).

- intensité max sur sortie 3.3v=50mA.
- mémoire programme Flash: 32 ko dont 0.5ko sont utilisés par le bootloader.
- mémoire SRAM (mémoire volatile): 2 kB.
- mémoire EEPROM (mémoire non volatile): 1 kB.

Il contient aussi trois compteurs (Timer0,Timer1,Timer2),le Timer0 et le Timer2 sont à comptage 8 bits, le Timer1 il est à comptage 16 bits. Chaque Timer peut être utilisé pour générer deux signaux PWM.certaines broches peuvent avoir plusieurs fonctions différentes choisies par programmation :

- ✓ **PWM** =pour l'utilisation de la PWM, le ATmega a 6 broches qui peuvent servir à cette fonction qui sont les broches OC0A(PD6), OC0B(PD5), OC1A(PB1), OC1B(PB3) OC2A(PB3), OC2B(PD3).
- ✓ **Convertisseur Analogique/Numérique** = le ATmega328 possède un convertisseur Analogique/Numérique d'une résolution de 10 bits, ce convertisseur peut être utilisé à travers 6 entrées multiplexées de ADC0(PC0) jusqu' à ADC5(PC5).
- ✓ **Gestion bus I2C**= ce bus est exploité via les deux broches SDA(PC5)/SCL(PC4).
- ✓ **Port série (USART)** = émission/réception série via les broches TXD(PD1)/RXD(PD0).
- ✓ **Comparateur Analogique** = le comparateur analogique intégré dans le ATmega peut être utilisé à travers les deux broches AIN0(PD6) et AIN1(PD7),ce comparateur peut déclencher une interruption.

- ✓ **Watchdog Timer programmable** : l'ATMega possède un compteur dit de chien de garde programmable pour générer des interruptions à la fin de son comptage et il peut être utilisé comme étant un simple compteur.

- ✓ **Gestion d'interruptions (24 sources possibles) : en résumé**
 - interruptions liées aux entrées INT0 (PD2) et INT1(PD3).
 - interruptions sur changement d'état des broches PCINT0 à PCINT23.
 - interruptions liées aux Timers 0,1 et 2(plusieurs causes configurables).
 - Interruption liée au comparateur analogique.
 - Interruption de fin de conversion ADC.
 - Interruptions du port série USART.
 - Interruption du bus I2C.

I-14-13. Architecteur interne de l'Atmega328 :

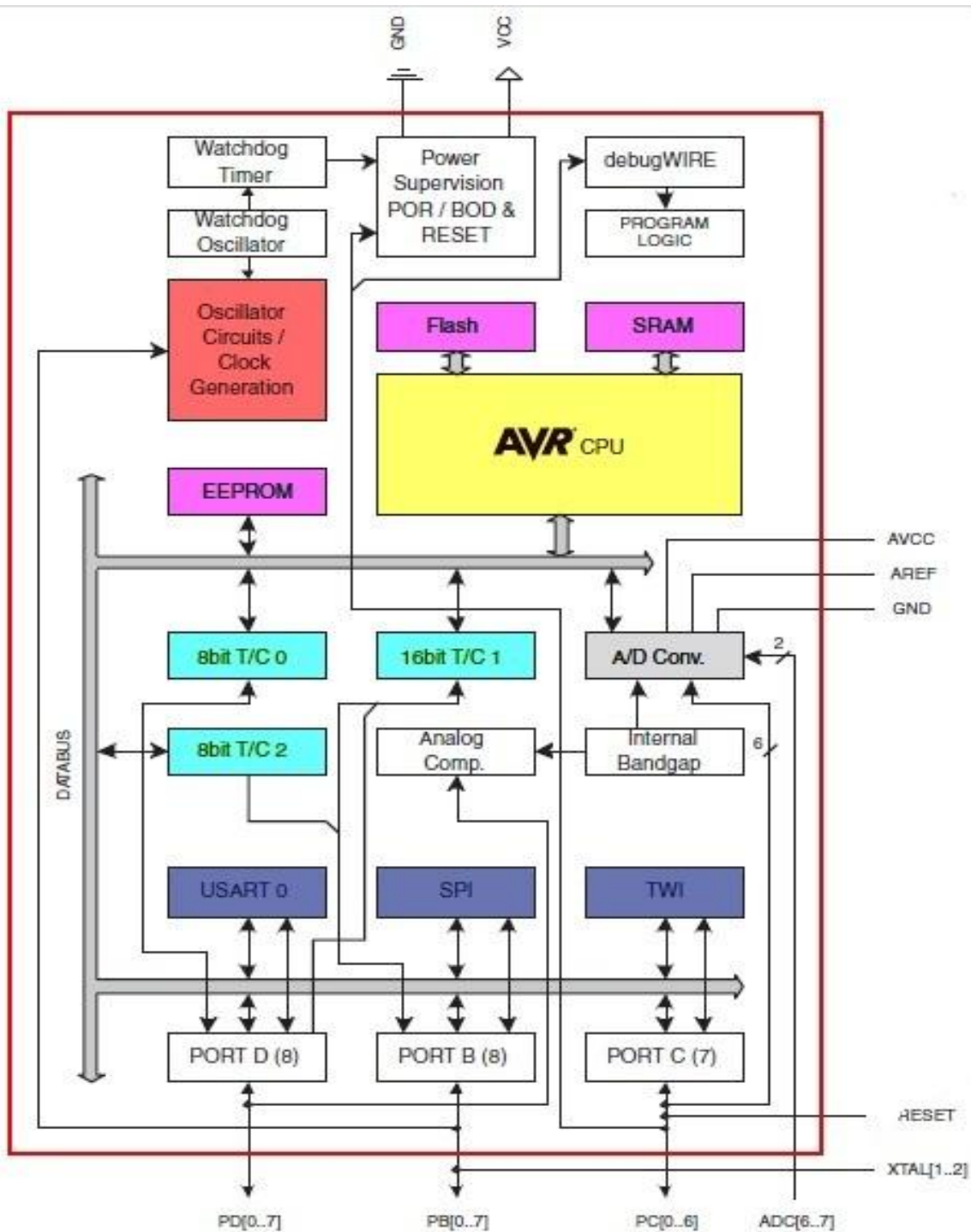
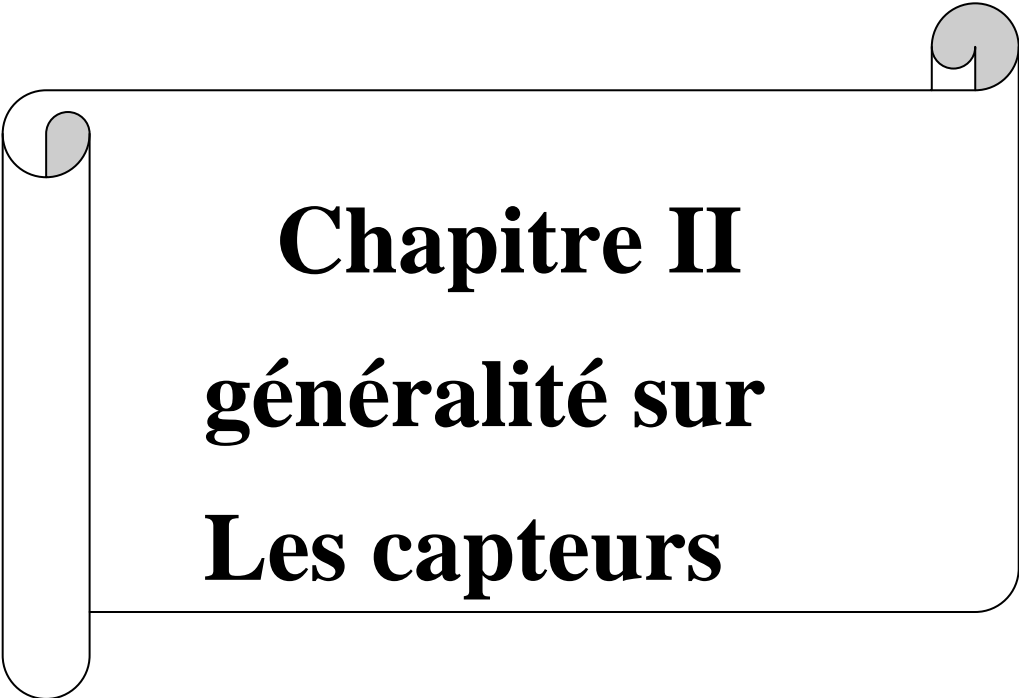


Figure 1-12 : Architecteur interne de l'Atmega328.

Discussion :

L'Avènement de la carte Arduino a permis le développement de diverses applications embarquées. Cela permet de remplacer plusieurs réalisations électroniques par une seule carte à plusieurs fonctions.

Afin d'utiliser la carte Arduino, il faut maîtriser le logiciel de programmation Arduino. La même carte peut être utilisée pour deux réalisations différentes. Il faut juste modifier le programme.



Chapitre II
généralité sur
Les capteurs

II-1. Préambule :

Un antivol est un dispositif ou un mécanisme destiné à dissuader, décourager, retarder, signaler ou empêcher le vol d'un objet.

L'antivol peut être distinct de l'objet qu'il est destiné à protéger, ou bien être intégré à cet objet. Il est conçu de sorte que son retrait ou sa désactivation ne puisse être effectué que par le propriétaire de l'objet (à l'aide d'une clef, d'un code...).

Un antivol C'est une petite centrale d'alarme à une entrée, équipée d'une puissante sirène, est adaptée pour fonctionner avec n'importe quel capteur ou groupe des capteurs filaires. C'est un système idéal pour protéger de petits locaux, comme un box, une cave, un studio etc...

II-2. Les capteurs :[33]**II-2-1. Définition :**

Un capteur est un organe de prélèvement d'information qui élabore à partir d'une grandeur physique, une autre grandeur physique de nature différente (très souvent électrique). Cette grandeur représentative de la grandeur prélevée est utilisable à des fins de mesure ou de commande.

Un capteur est composé de deux éléments :

- **Corps d'épreuve** : est un élément mécanique qui régit sélectivement à la grandeur à mesurer. Il a pour rôle de transformer la grandeur à mesurer en une autre grandeur physique dit mesurable. Cette grandeur constitue la réaction du corps d'épreuve.

- **Détecteur (élément sensible)** : est un élément lié au corps d'épreuve, il traduit les réactions du corps d'épreuve en une grandeur électrique constituant le signal de sortie.

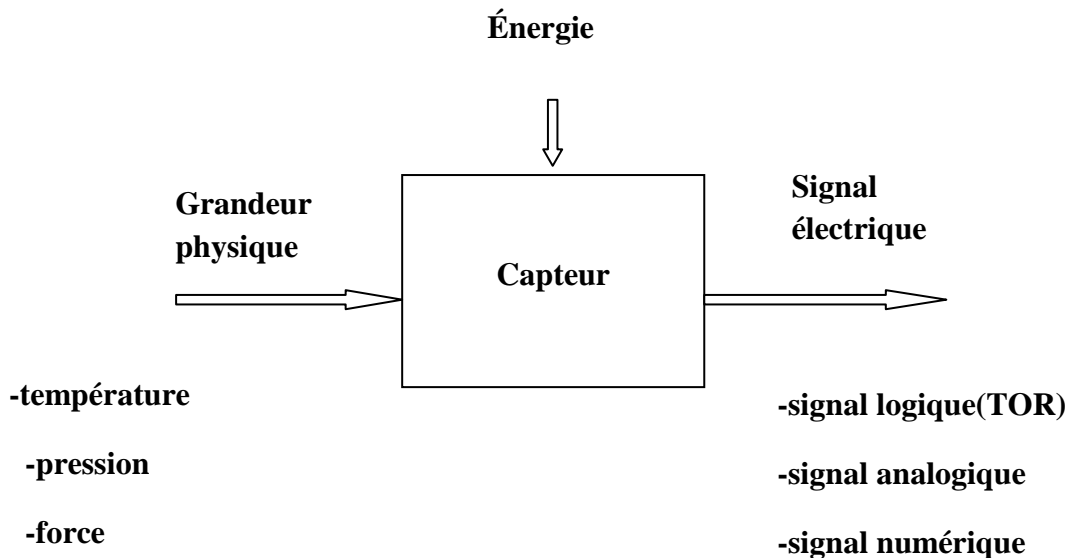


Figure 2-1 : fonctionnement d'un capteur.

II-2-2. Les principes de mesure d'un capteur : [34]

- Le mesurande (X): c'est l'objet de la mesure (la grandeur à mesurer).
- Le mesurage : c'est l'ensemble des opérations pour déterminer la valeur du mesurande.
- La mesure(x) : c'est le résultat du mesurage.
- Erreur absolue(e) : résultat du mesurage moins la valeur vraie du mesurande. Une erreur absolue s'exprime dans l'unité de la mesure.

$$e = x - X \dots \dots \dots (1)$$

- Erreur relative (e_r) : rapport de l'erreur de mesure à une valeur vraie de mesurande, une erreur relative s'exprime généralement en pourcentage de la grandeur mesurée.

$$e_r = e / X \dots\dots\dots(2)$$

- L'incertitude (dx) : le résultat de la mesure x d'une grandeur X n'est pas complètement défini par un seul nombre, il faut au moins la caractériser par un couple (x,dx) et une unité de mesure. dx est l'incertitude sur x, les incertitudes proviennent des différentes erreurs liées à la mesure.

$$x-dx < X < x+dx\dots\dots\dots(3)$$

II-2-3.les caractéristiques d'un capteur :[36]

- Etendue de mesure : valeurs extrêmes pouvant être mesurée par le capteur.
- Résolution : plus petite variation de grandeur mesurable par le capteur.
- Sensibilité : variation du signal de sortie par rapport à la variation du signal d'entrée.
- Précision : aptitude du capteur à donner une mesure proche de la valeur vraie.
- Rapidité : temps de réaction du capteur, la rapidité est liée à la bande passante.
- Linéarité : représente l'écart de sensibilité sur l'étendue de mesure.

II-2-4. Les types d'erreurs classiques :

- **L'erreur de zéro (offset) :**

Erreur de zéro = la valeur de x quand X=0

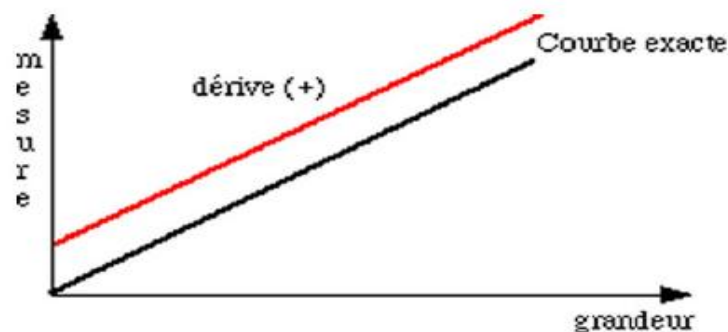


Figure2- 3-a : l'erreur de zéro.

➤ **L'erreur d'échelle (gain) :**

C'est une erreur qui dépend de façon linéaire de la grandeur mesurée

$$\text{Erreur de gain} = 20 \log(\Delta x / \Delta X) \dots \dots \dots (4)$$

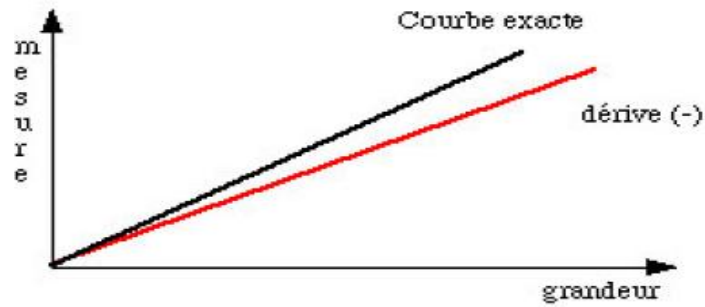


Figure2- 3-b : L'erreur d'échelle.

➤ **L'erreur de linéarité :**

Les caractéristique n'est pas une droite

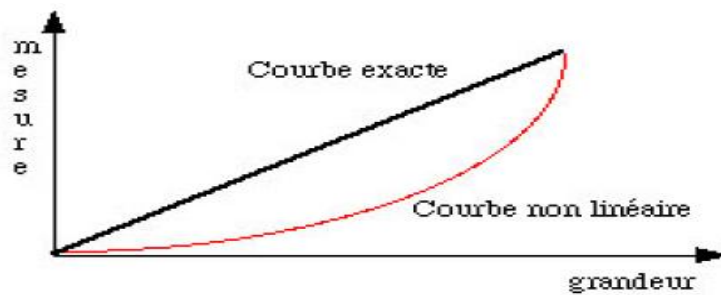


Figure 2-3-c : L'erreur de linéarité.

➤ **L'erreur due au phénomène d'hystérésis :**

Il y a phénomène d'hystérésis lorsque le résultat de la mesure dépend de la précédente mesure

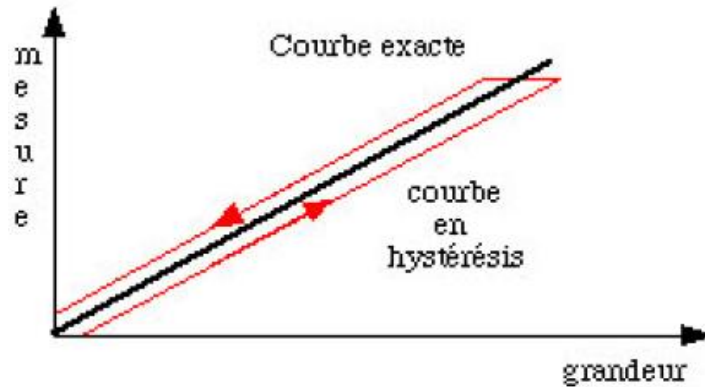


Figure 2-3-a: L'erreur due au phénomène d'hystérésis.

➤ **L'erreur de quantification :**

La caractéristique est en escalier, cette erreur est souvent due à une numérisation du signal.

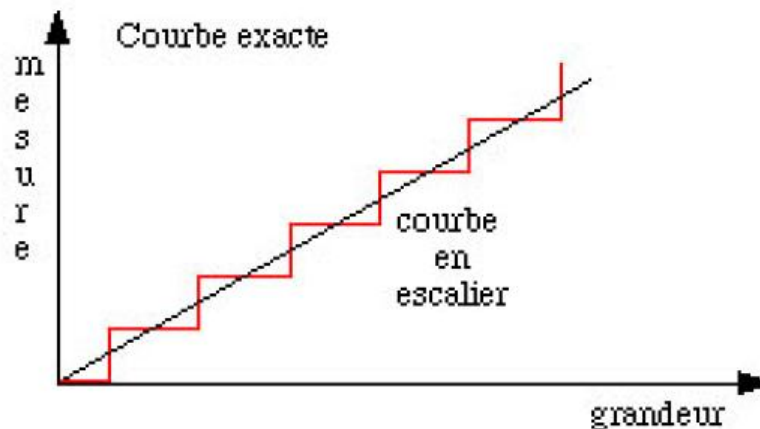


Figure2- 3-e: L'erreur de quantification.

II-2-5. Les différentes familles de capteurs :[37]

On peut classer les capteurs en quatre familles :

➤ **Les capteurs actifs :**

Les capteurs actifs sont des capteurs qui fonctionnent en générateur, un capteur actif est généralement fondé dans son principe sur un effet physique qui assure la conversion en énergie électrique de la forme d'énergie propre à la grandeur physique à mesurer (énergie thermique, mécanique ou de rayonnement).

- Les effets physiques les plus classiques sont :
 - ✓ **Effet thermoélectrique**) : Un circuit formé de deux conducteurs de nature chimique différente, dont les jonctions sont à des températures T_1 et T_2 , est le siège d'une force électromotrice d'origine thermique $e(T_1, T_2)$.
 - ✓ **Effet piézo-électrique** : L'application d'une contrainte mécanique à certains matériaux dits piézo-électriques (le quartz par exemple) entraîne l'apparition d'une déformation et d'une même charge électrique de signe différent sur les faces opposées.
 - ✓ **Effet d'induction électromagnétique** : La variation du flux d'induction magnétique dans un circuit électrique induit une tension électrique (détection de passage d'un objet métallique).
 - ✓ **Effet photo-électrique** : la libération de charges électriques dans la matière sous l'influence d'un rayonnement lumineux ou plus généralement d'une onde électromagnétique.
 - ✓ **Effet Hall** : Un champ magnétique B et un courant électrique I créent dans le matériau une différence de potentiel.
 - ✓ **Effet photovoltaïque**: des électrons et des trous sont libérés au voisinage d'une jonction PN illuminée, leur déplacement modifie la tension à ses bornes.

➤ **Les capteurs passifs :**

Il s'agit généralement d'impédances (résistance, inductance, capacité) dont l'un des paramètres déterminants est sensible à la grandeur mesurée. La variation d'impédance résulte :

- d'une variation de dimension du capteur (capteurs de position, potentiomètre, inductance à noyaux mobile, condensateur à armature mobile)
- d'une déformation résultant d'une force ou d'une grandeur s'y ramenant (pression accélération). Exemple : armature de condensateur soumise à une différence de pression.

➤ **Les capteurs composites :[38]**

Un capteur composite est un capteur constitué d'un corps d'épreuve et d'un capteur actif ou passif. Le corps d'épreuve, quant à lui, est un capteur qui, soumis au mesurande, donne une grandeur physique non électrique appelée mesurande secondaire qui elle va être traduite en une grandeur électrique par un capteur.

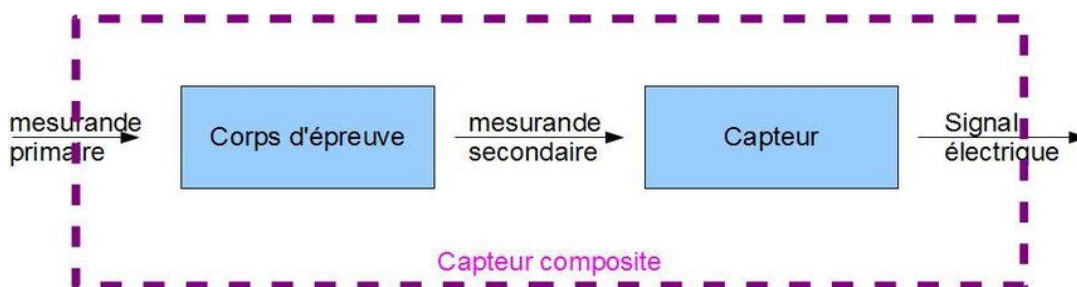


Figure2- 4 : un capteur de composition.

➤ **Les capteurs intégrés[39]**

Un capteur intégré est un capteur qui utilise la microélectronique. Ce capteur est constitué d'une plaque en silicium dans lequel on a fixé le capteur, le corps d'épreuve si besoin et d'autres composants électroniques qui peuvent servir à linéariser, amplifier, convertir le courant en tension, etc.

Ce type de capteur est très utile vu qu'il fournit un signal linéaire avec une grande sensibilité, une miniaturisation et un coût faible.

II-2-6. Classification selon les signaux :[40]

- les capteurs Tout Ou Rien (TOR)
- les capteurs analogiques
- les capteurs numériques

• **Le capteur Tout Ou Rien (TOR) :**

Ce type de capteur permet de **détecter** un événement ou un objet lié au fonctionnement du système technique.

Le signal électrique en sortie de ce capteur est de type logique (signal acceptant 2 niveaux : niveau logique 0 ou niveau logique 1)

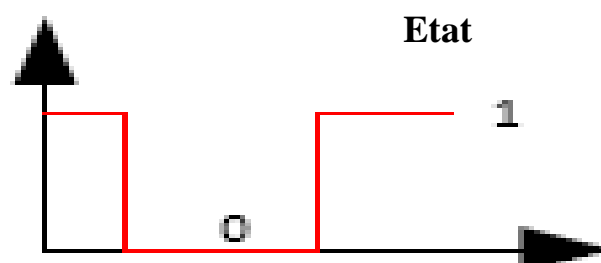


Figure 2-5 : le signal TOR.

- **Capteurs analogiques :**

La sortie est une grandeur électrique dont la valeur est une fonction de la grandeur physique mesurée par le capteur. La sortie peut prendre une infinité de valeurs continues

- ✓ **Signal continu** : c'est un signal qui varie lentement dans le temps par exemple (mesure de température)

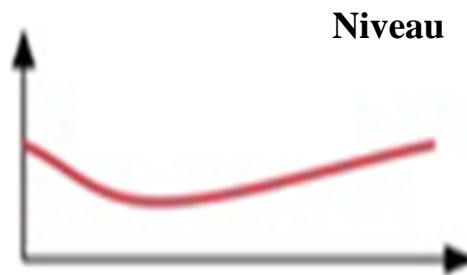


Figure 2-6-a : le signal continu.

- ✓ **Signal formel** : c'est la forme de ce signal qui est l'information importante par exemple (pression cardiaque)

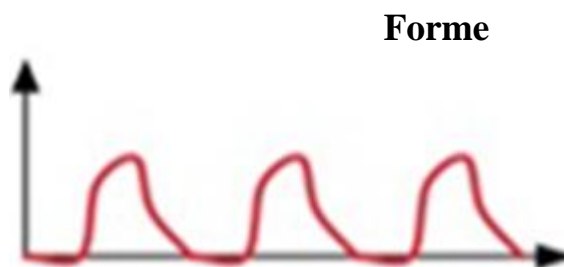


Figure 2-6-b : le signal formel.

- ✓ **Signal fréquentiel** : c'est le spectre fréquentiel qui transporte l'information désirée par exemple (analyse vocale).

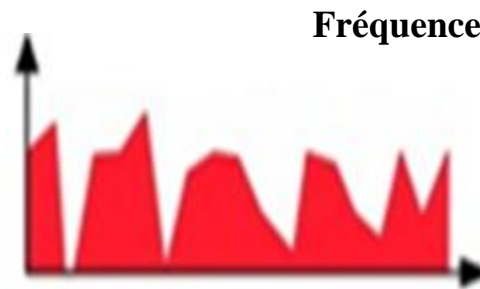


Figure2- 6-c : le signal fréquentiel.

- **Signal numérique :**

Un signal est numérique si l'amplitude de la grandeur physique le représentant ne peut prendre qu'un nombre fini de valeurs. On a trois cas :

- ✓ Signal Tout Ou Rien
- ✓ Train d'impulsion : chaque impulsion est l'image d'un changement d'état par exemple (un codeur incrémental donne un nombre fini et connu d'impulsion par tour)

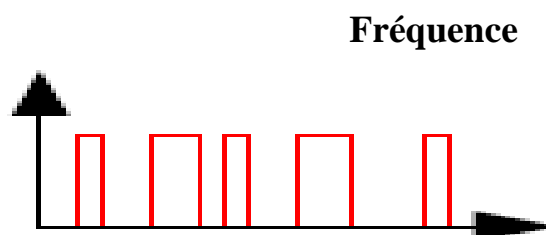


Figure2-7-a : le signal train d'impulsion.

- ✓ L'échantillonnage : c'est l'image numérique d'un signal analogique exemple (débit, niveau)

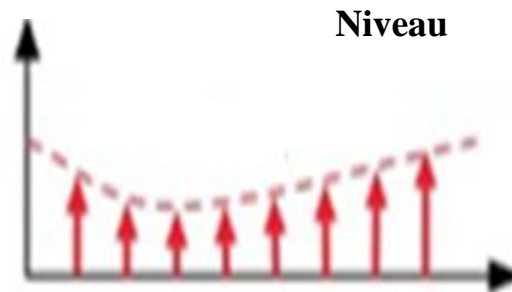


Figure 2-7-b : le signal d'échantillonnage.

II-2-7. Caractéristiques d'une chaîne de mesure numérique :[41]

C'est l'ensemble des traitements du signal issu du capteur qui va permettre l'interprétation ou affichage correct du mesurande, chaîne de mesure simple : on parle aussi de conditionnement du signal.

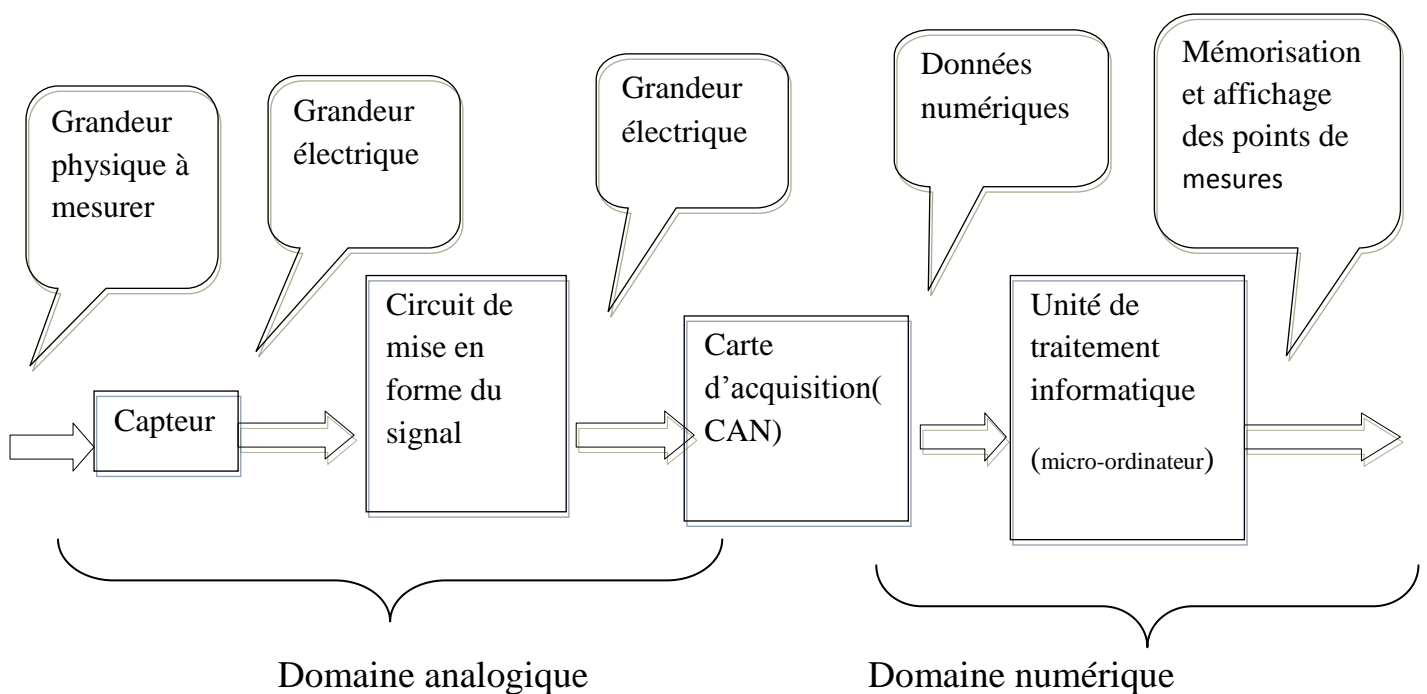


Figure 2-7 : la chaîne de mesure.

La structure de base d'une chaîne de mesure comprend au minimum quatre étages :

- Un capteur sensible aux variations d'une grandeur physique et qui à partir de ces variations délivre un signal électrique.
- Un conditionneur de signal dont le rôle principal est l'amplification du signal délivré par le capteur pour lui donner un niveau compatible avec l'unité de numérisation, cet étage peut parfois intégrer un filtre qui réduit les perturbations présentes sur le signal.
- Une unité de numérisation qui va échantillonner le signal à intervalles réguliers et affecter un nombre (image de la tension) à chaque point d'échantillonnage.
- L'unité de traitement informatique peut exploiter les mesures qui sont maintenant une suite de nombres (enregistrement, affichage de courbes,...etc).

II-2-8. Les critères de choix d'un capteur :[42]

- Grandeur physique à acquérir
- Température, pression, signaux visuels...
- Fonction de transfert grandeur électrique = f (grandeurs physiques)
- Plage d'acquisition
- Gamme de mesure de la grandeur physique
- Plage de restitution
- La grandeur de sortie
- Précision
- Erreur maximale sur la mesure (exprimée en pourcentage de la valeur mesurée)
- Fidélité
- Prix

- Variation minimale de la grandeur d'entrée qui provoque une variation détectable de la grandeur de sortie.

II-2-9. Caractéristiques métrologiques :[43]

Pour bien choisir un capteur, il est important de connaître ses caractéristiques métrologiques. En effet les caractéristiques métrologiques permettent de savoir quel capteur utiliser, dans quelle gamme, avec quelle précision le résultat nous sera donné et beaucoup d'autres précisions qui peuvent s'avérer très utile dans le choix du capteur.

➤ Les erreurs :

Le capteur et toute la chaîne de traitement de la mesure introduisent des erreurs : bruit, décalage, ...etc

L'erreur globale de mesure ne peut être qu'estimée, une conception rigoureuse de la chaîne de mesure permet de réduire les erreurs et donc l'incertitude sur le résultat.

➤ Etalonnage :

L'étalonnage permet d'ajuster et de déterminer, sous forme graphique ou algébrique, la relation entre le mesurande et la grandeur électrique de sortie. Très souvent l'étalonnage n'est valable que pour une seule situation d'utilisation du capteur.

➤ Limites d'utilisation :

Les contraintes mécaniques, thermiques ou électriques auxquelles un capteur est soumis entraînent, lorsque leurs niveaux dépassent des seuils définis, une modification des caractéristiques du capteur, au-dessus d'un certain seuil, l'étalonnage n'est plus valable, au-dessus d'un autre plus grand le capteur risque d'être détruit.

➤ **Sensibilité :**

Plus un capteur est sensible plus la mesure pourra être précise, c'est une caractéristique importante pour l'exploitation et l'interprétation des mesures.

➤ **Rapidité- temps de réponse :**

La rapidité est la spécification d'un capteur qui permet d'apprécier de quelle façon la grandeur de sortie suit dans le temps les variations du mesurande.

➤ **Finesse :**

La présence du capteur peut perturber le phénomène physique mesuré.

La finesse est une spécification qui permet d'estimer l'influence de la présence du capteur et de ses liaisons sur la valeur du mesurande. La finesse est d'autant plus grande que l'influence du capteur est faible.

II-2-10. Grandeurs d'influence :

Les grandeurs d'influence, ce sont toutes les grandeurs physiques autres que le mesurande dont la variation peut modifier la réponse du capteur :

- **Température** : modifications des caractéristiques électriques, mécaniques et dimensionnelles
- **Pression, vibrations** : déformations et contraintes pouvant altérer la réponse.
- **Humidité** : modification des propriétés électriques (constante diélectrique ou résistivité). dégradation de l'isolation électrique.
- **Champs magnétiques** : création de f.é.m d'induction pour les champs variables ou modifications électriques (résistivité) pour les champs statiques.
- **Tension d'alimentation** : lorsque la grandeur de sortie du capteur dépend de celle-ci directement (amplitude ou fréquence).

- **nécessité de :**
 - ✓ **Réduire les grandeurs d'influence** (tables anti-vibration, blindages magnétiques...)
 - ✓ **Stabiliser les grandeurs d'influence** à des valeurs parfaitement connues.
 - ✓ **Compenser** l'influence des grandeurs parasites par des montages adaptés (pont de Wheastone)

II-2-11. Le capteur a effet Hall :

II-11-1. Définition :[45]

Un capteur a effet All donne un signal lorsqu'il détecte un champs magnétique ou une pièce métallique, la tension de Hall et amplifié dans le capteur.

II-11-2. Principe de fonctionnement :[46]

Si un courant I_0 traverse un barreau en matériau conducteur ou semi-conducteur, et si un champ magnétique d'induction B est appliqué perpendiculairement au sens de passage du courant, une tension V_h , proportionnelle au champ magnétique et au courant I_0 , apparait sur les faces latérales du barreau. Les électrons sont déviés par le champ magnétique, créant une différence de potentiel appelée tension de Hall. Le champ magnétique déforme la trajectoire

des électrons car il engendre une force de LORENTZ(e)

$$V_h = K_h * B * I_0 \dots \dots \dots (4)$$

Avec K_h : constante de Hall, qui dépend du matériau utilisé

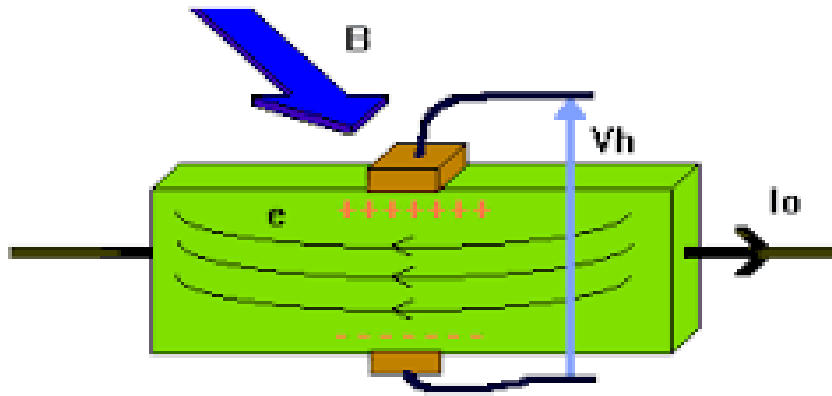


Figure 2-8: le principe de fonctionnement d'un capteur a effet Ahll.

II-11-3. un capteur effet Hall UGN3503 :[47]

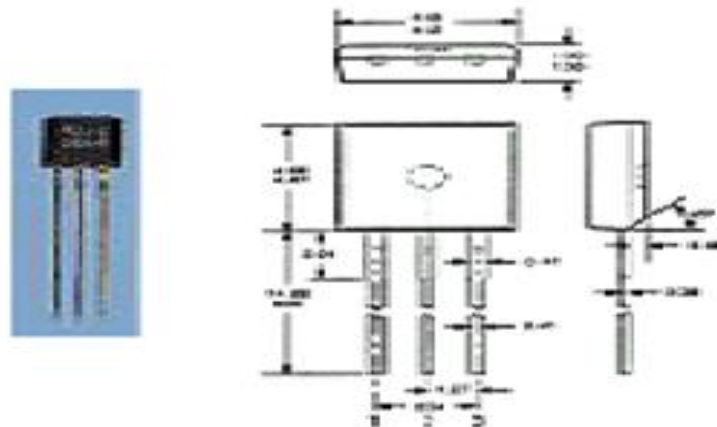


Figure2-9 : un capteur effet Hall UGN3503

Brochage :

- 1VCC
- 2Gnd
- 3Sortie

II-11-4.Caractéristiques :[48]

- Capteur à effet Hall UGN3503 de Allegro Micro Systems
- Très haute sensibilité
- Alimentation : 4,5 à 6 V

II-11-5.Utilisation :[49]

- Réalisation de détecteurs de position / proximité / mouvement, de compteurs de rotation, d'ampèremètres à induction, ...
- Mise en œuvre simple

Discussion :

Les capteurs sont des éléments essentiels dans une réalisation électronique. En effet, avec un capteur on aura un signal électrique à partir d'une grandeur physique. Ce signal électrique peut être exploité selon les besoins (amplification, filtrage, ...etc). Le choix du capteur à utiliser dans une réalisation dépend de plusieurs paramètres dont la grandeur physique à mesurer, le prix, la précision, ...etc.



Chapitre III
Réalisation pratique

III-1.Préambule :

Après l'étude théorique des différents blocs de système antivol compact, nous passerons à la réalisation pratique.

Dans ce chapitre, nous allons présenter les étapes permettant de réaliser ce système antivol simple compact.

Il s'agit d'un petit système d'alarme, constituée d'une carte Arduino Uno et d'un capteur effet hall. Cette alarme procède à la surveillance permanente d'un objet métallique (voiture), le système d'alarme réalisé sera positionné devant une sortie de garage permettant de détecter un éventuel passage d'un véhicule.

III-2.Matériel à utiliser :

- Carte Arduino UNO
- Capteur a effet hall UGN3503
- Haut parleur
- Câble USB
- Breadboard
- 3 Résistances de $1K\Omega$
- Transistor 2n3904
- Diode 1N914
- Fiche Jack.
- LED

III-3.Schéma Bloc :

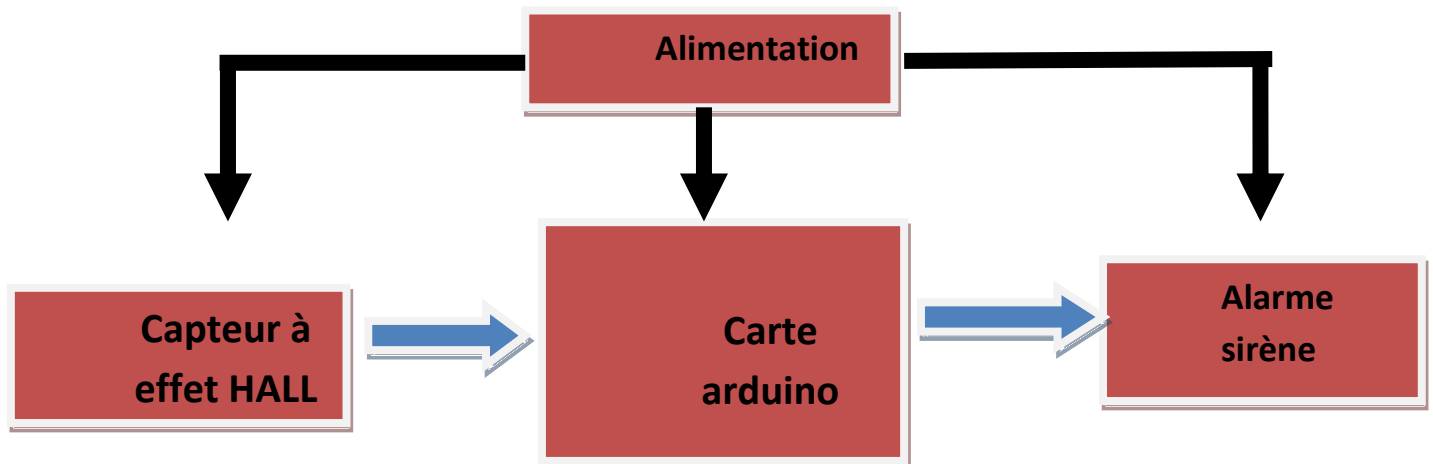


Figure3-3 : Schéma Bloc.

➤ **Alimentation :**

Pour l'alimentation on a utilisé une batterie de 12V avec un diviseur de tension pour obtenir une tension de 5v. Celle-ci sert à alimenter la carte ARDUINO et le capteur a effet hall. Pour le haut-parleur (sirène) on l'a alimenté avec l'alimentation de 12v qui est transmise à la sirène quand le transistor est saturé lequel est excité par la sortie de la carte ARDUINO.

III-4. Tracé du circuit imprimés et schéma électrique:

Nous avons utilisé le logiciel proteus(ISIS 7 professional) pour faire le circuit imprimé et le schéma électrique comme illustré par les figures suivantes :

➤ **Le circuit imprimé :**

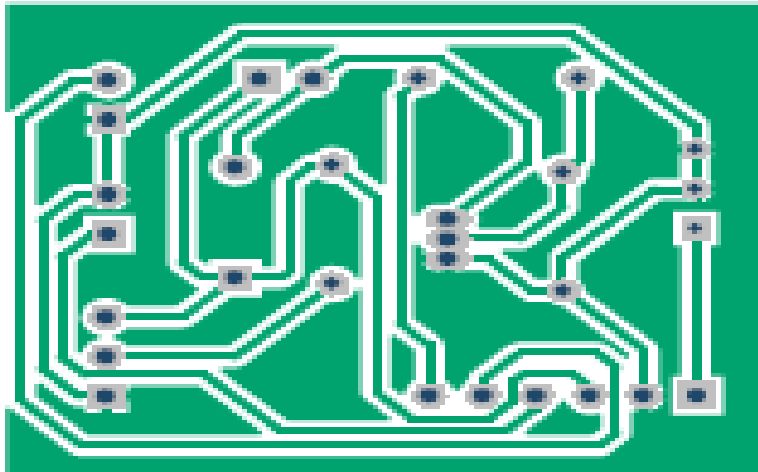


Figure 3-1 : circuit imprimé.

➤ **le schéma électrique correspondant :**

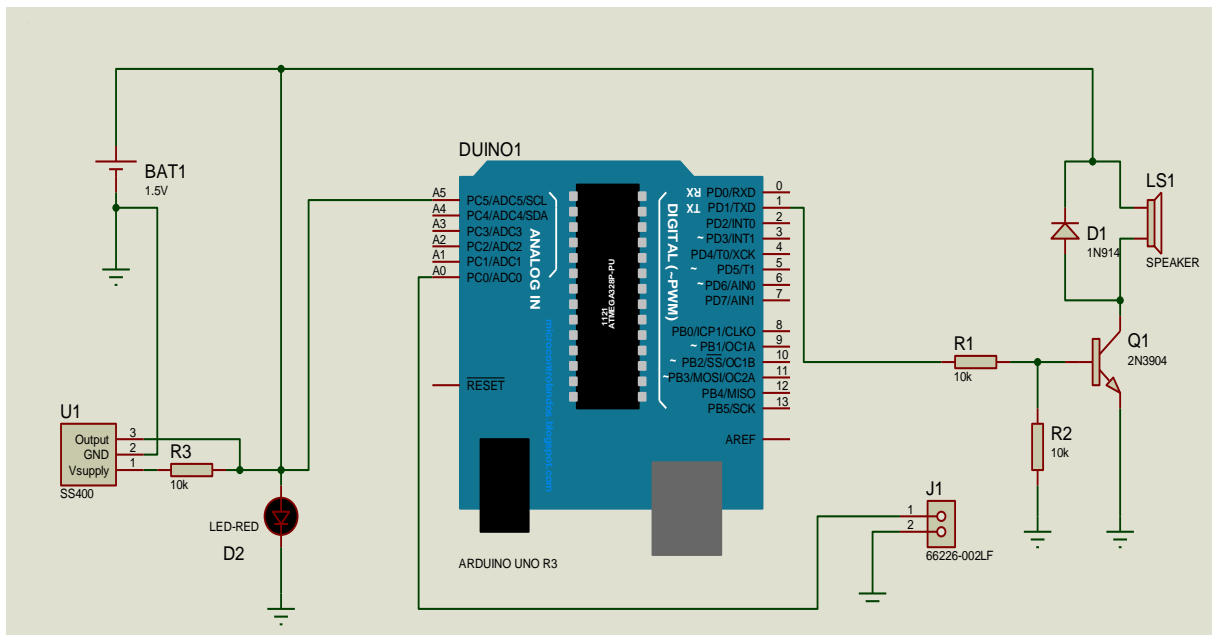


Figure3- 2 :le schema électrique .

III-5.Fonctionnement du schéma électrique :

Le circuit ci-dessous représente une alarme anti-intrusion commandé par une carte ArduinoUNOet un capteur à effet hall. Pour activer l'alarme et la rendre capable de détecter les signaux du capteur il faut insérer le jack équipée de sa résistance et faisant office d'une clef reliée à la fiche femelle de la fiche jack. La clef étant insérée, l'alarme devient opérationnelle cette situation est matérialisée par l'extinction de la led et par émission d'une brève note acoustique de la part de la sirène à partir de ce moment le capteur à effet HALL de type UGN3503 est utilisé pour capter un éventuel champs magnétique émis par une quelle conque source considérée comme une source d'intrusion, de ce faite le capteur convertit le champ magnétique en une tension électrique qui est récupérer par la carte ARDUINO UNO sur le port de l'entrée analogique de ce faite une impulsion est générer sur la base du transistor Q1 2N3904 préalablement qui se trouve dans un état bloqué a travers la sortie numérique de la carte, alors le transistor devient saturé, un courant I_c traverse le haut parleur et cela créer un signal sonore, une diode est utilisée sur le collecteur du transistor parallèlement au haut parleur comme une protection contre les courant inverse qui pouvaient être générer par le bobinage du haut parleur . La sirène étant enclenchée et reste ainsi pour une durée allant de 10 seconde à 2 minutes. Si au terme de cette temporisation le capteur détecte toujours, il se produit un nouveau cycle et la sirène recommence à sonne.

Dans le cas contraire si le capteur ne détecte plus, l'alarme passe au repos.

Une attention particulière doit être portée a la phase de désactivation de la centrale d'alarme

On admet que l'alarme a été mise en service et que le propriétaire s'était absenter. Si, a son retour il insère la clef pour désactiver la centrale et que, durant son absence, l'alarme a été déclenchée, la diode LED s'allume et la

sirène émet quatre tonalités. Par contre, si aucun déclenchement ne s'est produit, la diode LED s'allume et la sirène émet seulement deux notes.

III-6. Organigramme global du système :

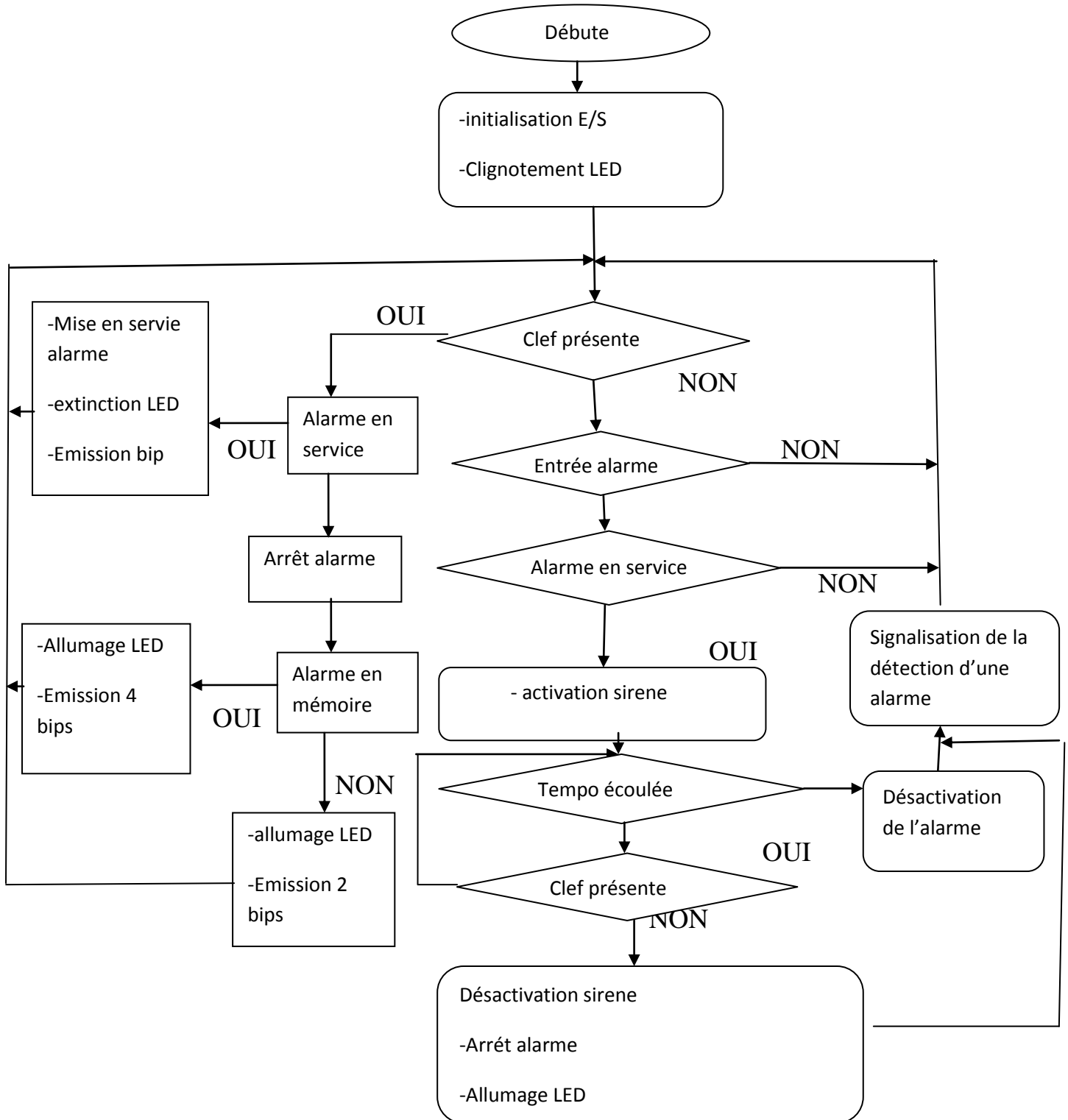


Figure III- 3 : Organigramme global du système.

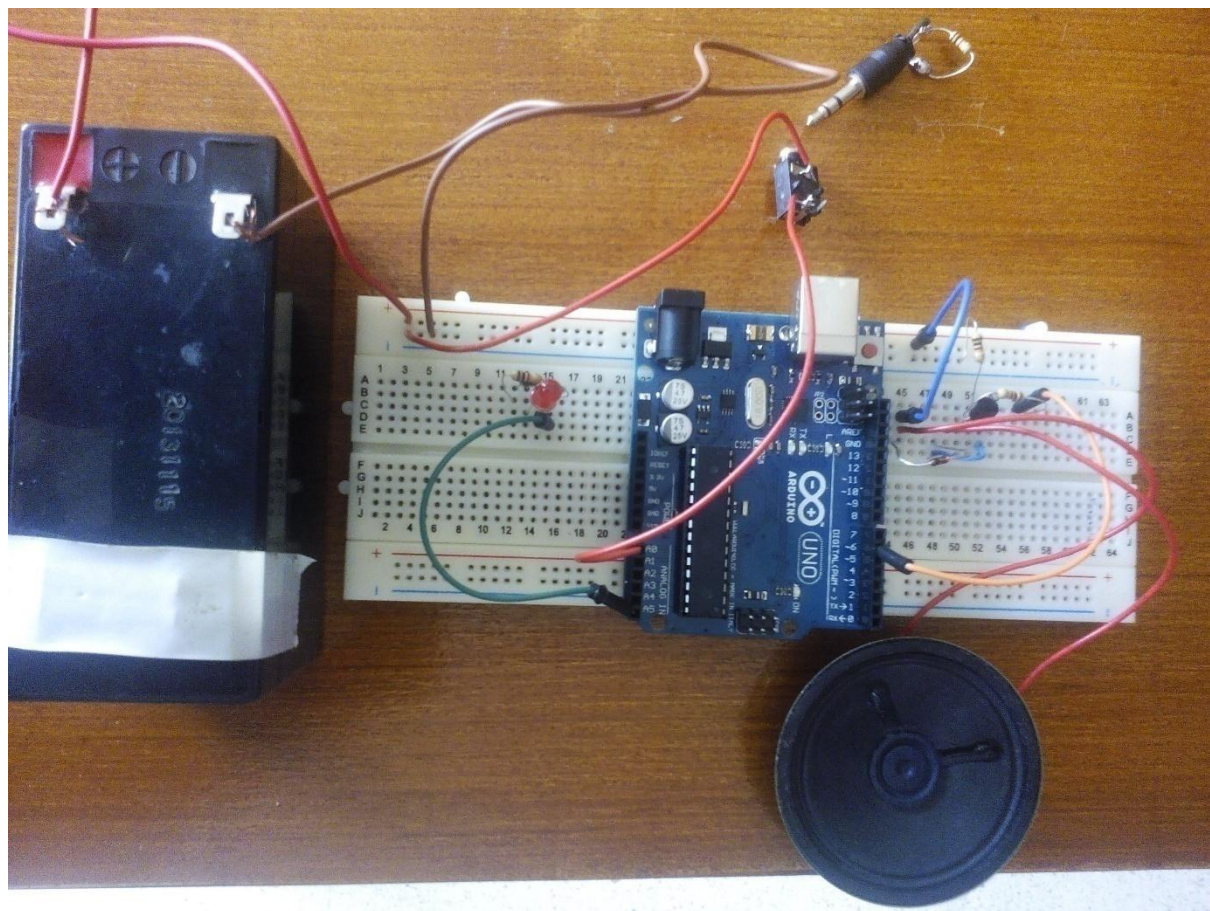
III-7. Le schéma général de la réalisation :

Figure 3-5: le schéma général de la réalisation.

Discussion :

Dans ce chapitre nous nous sommes intéressés à l'aspect pratique de notre étude concernant l'inti vole basé sur un capteur à effet HALL et une carte ARUINO, une conception qui peut être utilisée pour détecter tout matériaux pouvaient générer un champ magnétique une étude théorique préalablement effectuée pour chaque partie de notre circuit électrique afin de satisfaire notre cahier des charges et cela en connaissant toutes les caractéristique électrique de chaque composant comme, le courant de fonctionnement du capteur et cela en dimensionnant l'alimentation et la résistance R_4 et la tension délivrée par ce dernier qui doit être suffisante pour qu'il soit lu par la carte ARDUINO, ainsi la même opération est effectuer pour Q1 en faisant un choix sur le type de transistor en prenant en considération le courant délivrer par la carte ARDUINO qui doit être suffisant pour que le transistor Q1 puisse change d'état afin de générer un signal sonore et cela en se référant aux datasheets,



Conclusion

Conclusion

Le travail présenté concerne l'étude et la réalisation d'un système antivol compact. Ce système nous prévient d'une quelconque tentative d'effraction et peut être utilisé pour la protection des véhicules, des habitations, des magasins et autres.

Afin de réaliser notre alarme, nous avons fait le choix du capteur. Plusieurs alarmes utilisent soit un capteur infrarouge ou sonore. Alors, nous avons voulu tester un autre type de capteurs qui est le capteur à effet Hall. Celui-ci est connecté à une carte programmable qui est Arduino.

Ensuite, nous avons conçu une carte électronique et vérifié son bon fonctionnement avec Proteus.

Une fois la simulation effectuée, nous avons programmé notre carte Arduino Uno. Puis, nous avons réalisé notre carte électronique.

Ce travail a été bénéfique à plus d'un titre, dans la mesure où il nous a permis de mettre en pratique et approfondir nos connaissances acquises durant notre cursus et de mieux comprendre le fonctionnement et l'utilisation des composants électroniques et surtout la carte Arduino.

Comme perspectives, nous proposons d'associer à ce capteur à effet Hall un autre capteur infrarouge.



Bibliographie

bibliographiques

➤ Mémoires :

[1] Etude et réalisation d'une alarme antivols présentée par : M^r djebrane hameg et M^r hacene manseur universitaire MOULOUD MAMMERRI tizi-ouzou (2007/2008).

[9] [20][28][31] conception et réalisation d'un système de télésurveillance médicale à base d'une carte Arduino « transmission bluetooth » présenté par : M^r nachef mhenna universitaire MOULOUD MAMMERRI tizi-ouzou (2012/2013)

➤ [11][12] TP INITIATION ARDUINO FEGI-Département Electronique Master M1 Electronique Biomédicale. (2015/2016)

➤ Sites internet :

[2] file:///E:/arduino/chapI/D%C3%A9buter%20avec%20une%20carte%20Arduino%20-%20PoBot.html

[3][14][15][16]17[18] arduino.pdf published :2011-12-22 license :None

[4] file:///E:/arduino/chapI/Arduino%20-%20wikipédia.htm

[5] [10] Arduino : Premiers pas en informatique embarquée Auteurs : Simon Landrault (Eskimon) et Hippolyte Weisslinger (olyte)
Le blog d'Eskimon Édition du 01 juin 2014

[6] file:///E:/arduino/chapI%20différentes%20cartes%20Arduino.html

[7][8] http://www.arduino.cc

[13] http://www.arduino.cc/en/Reference/Libraries

[19] [26] file:///E:/arduino/Arduino%20carte%20Arduino%20uno.htm

[21][23][25][27][29] file:///H:/arduino%20uno.htm

[22] file:///D:/_capiter%20Micro.controleu%20_%20Arduino.htm

[24] file:///E:/arduino/Arduino%20uno.htm

[30]Microcontrôleurs EI3Option agi B.Cottenceau B311ISTIA
bertrand. cottenceau@univ-angers.fr

[32]file:///D:/arduino%20%20uno.htm

[33][34][35][36][37][40][41][44] <http://cbissprof.free.fr>

[38][39]file:///E:/les%20capteur/les%20capteur/Capteur_Introduction%20%E2%80%94%20Wikiversit%C3%A9.html

[42]<http://www.uvt.rnu.tn/resources-uvt/cours/Distillation/chapitre2/Chapitre-3/Section-3-5.html>

[43]<https://fr.wikiversity.org/wiki/Capteur/Introduction>

[45][46]file:///H:/les%20capteuuur/Capteurs-Actionneurs%20_%20technologie%20-%20Les%20capteurs%20%C3%A0%20effet%20Hall.html

[47]file:///H:/capteuuuur/dossier%20a%20effet%20Ahl/Th%C3%A9ories%20Sonde%20%C3%A0%20effet%20Hall.html

[48][49]file:///H:/capteuuuur/dossier%20a%20effet%20Ahl/Lot%20de%202%20Capteurs%20%C3%A0%20Effet%20Hall%20Ultra-Sensibles%20UGN3503%20IC0010%20_%20RADIOELEC%20_%20Composants%20et%20Modules%20Electroniques,%20sp%C3%A9cialiste%20Audiophiles%20et%20Radioamateurs.html

➤ **logiciels électroniques utilisés :**

[1] googel

[2] proteus (ISIS7 professional)

[3] arduino