

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكر بلقايد - تلمسان

Université Aboubakr Belkaïd – Tlemcen –

Faculté de TECHNOLOGIE



MÉMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

En : Télécommunications

Spécialité : Réseaux et Télécommunications

Par: Yahi Abderrezzak

Sujet

Surveillance à distance de la consommation d'énergie via un système IoT

Soutenu publiquement, le 18/06/2018, devant le jury composé de :

Président

R. MERZOUGUI

Univ. Tlemcen

Examinatrice

H. BENOSMAN

Univ. Tlemcen

Encadreur

M. HADJILA

Univ. Tlemcen

Année Universitaire : 2017-2018

Dédicace

Dédicaces

On remercie Allah le tout puissant de nous avoir donné la santé et la volonté d'entamer et de terminer ce mémoire.

A l'homme de ma vie, mon exemple éternel, mon soutien moral et source de joie et de bonheur, celui qui s'est toujours sacrifié pour me voir réussir, que Allah te garde dans son vaste paradis, à toi mon père.

A la lumière de mes jours, la source de mes efforts, la flamme de mon cœur, ma vie et mon bonheur maman que j'adore.

Aux personnes dont j'ai bien aimé la présence dans ce jour, à tous mes frères Mohammed, Abderrahim et ma sœur Fatima. Je dédie ce travail dont le grand plaisir leurs revient en premier lieu pour leurs conseils, aides, et encouragements.

Aux personnes qui m'ont toujours aidé et encouragé, qui étaient toujours à mes côtés, et qui m'ont accompagnaient durant mon chemin d'études supérieures, mes aimables amis, collègues d'étude, sans oublier les profs du master en général.

Remerciements

Avant tout remerciement, louange à Allah.

C'est avec un grand honneur que je réserve l'ouverture de mon projet en signe de gratitude et de reconnaissance à l'égard de tous ceux qui m'ont donné assistance, de près ou de loin, pour la réalisation de mon projet de fin d'étude. Je tiens à adresser mon vif remerciements à mon encadreur **Mr.HADJILA Mourad** enseignant et conférencier à l'université Abou-bekr-Belkaid de Tlemcen pour sa présence, son encadrement, ses conseils fournis de façon efficace tout au long de la période de réalisation. Je veux aussi exprimer tous mes sincères remerciements à tous les professeurs de l'université Abou Bekr Belkaid de Tlemcen qui ont fait beaucoup d'efforts pour nous transmettre leurs savoirs. Leurs compétences incontestables ainsi que leurs humanités qui vaut notre admiration et notre respect. Mon sincères remerciements s'adressent aussi pour votre patience et votre encadrement durant toutes ces années. Mes remerciements s'adressent également aux membres du Jury **M. H. BENOSMAN** et **Mr. MERZOUGUI Rachid** qui nous font l'honneur de participer à notre soutenance.

Titre Surveillance à distance de la consommation d'énergie via un système IoT.

Résumé

Le but des systèmes embarqués vise à trouver des solutions aux problèmes qui nous sont en face dans notre vie, afin de faciliter la tâche, d'assurer la fiabilité, la rapidité, et la flexibilité des ressources du monde réel en temps réel avec moins des contraintes.

Dans ce contexte, ce projet de fin d'études consiste à surveiller à distance la consommation d'énergie électrique à travers un système IoT équipé d'un système embarqué via le WiFi ou l'ADSL.

Mots-clés : Systèmes embarqués, arduino, NodeMCU, électricité, cloud, WiFi, capteur de courant, relais.

Title Remote monitoring of energy consumption via an IoT system.

Abstract

The goal of embedded systems is to find solutions to the problems we face in our lives, to make it easier, to ensure the reliability, speed, and flexibility of real-world resources in real time with less constraints.

In this context, this end-of-studies project consists in remotely monitoring the consumption of electrical energy through an IoT system equipped with an embedded system via WiFi or ADSL.

Keywords: Embedded systems, arduino, NodeMCU, electricity, cloud, WiFi, current sensor, relay.

العنوان المراقبة عن بعد لاستهلاك الطاقة عبر نظام إنترنت الأشياء.

ملخص

الهدف من الأنظمة المدمجة هو إيجاد حلول للمشاكل التي نواجهها في حياتنا ، لتسهيل الأمر ، لضمان موثوقية وسرعة ومرونة موارد العالم الحقيقي في الوقت الفعلي بأقل من القيود.

في هذا السياق ، يتكون مشروع نهاية الدراسات هذا من مراقبة استهلاك الطاقة الكهربائية عن بعد من خلال نظام إنترنت الأشياء مجهز بنظام مضمن عبر WiFi أو ADSL.

كلمات البحث: الأنظمة المدمجة ، arduino ، الكهرباء ، السحابة ، WiFi ، الاستشعار الحالية ، التتابع.

Table des matières

Dédicaces.....	i
Remerciements	ii
Résumé	iii
Tables des matières.....	iv
tables des figures.....	vii
Liste des abréviations.....	x
Introduction générale	12

Chapitre I : Les systèmes embarqués

I.1 Introduction.....	3
I.2 Historique des systèmes embarqués	3
I.2.1 Les anciens Génération.....	3
I.2.1.1 La première génération - les tubes à vides (1945-1955).....	3
I.2.1.2 Deuxième génération - les transistors (1955-1965).....	4
I.2.1.3 Troisième génération - les circuits intégrés (1965-1980).....	4
I.2.1.4 Quatrième génération - les microprocesseurs 4004 (1970-2000).....	5
I.2.1.5 Cinquième génération - PSoC (2000 – 2010).....	6
I.2.1.6 Sixième génération (2010 – aujourd’hui).....	7
I.3. Objets communicants & Internet des Objets (Internet Of Things).....	7
I.3.1 Objets communicants.....	7
I.3.2 Internet des Objets (Internet of Things).....	8
I.3.3 Des exemples dans le domaine d’application.....	9
I.3.4 La sécurité des IoT.....	12
I.4 Les Systèmes Cyber-Physiques (SCP).....	13
I.4.1 Exemple d’application dans les systèmes cyber-physique	13
I.5 Les données massives (Big Data).....	14
I.6 L’informatique en nuage (Cloud Computing).....	16
I.6.1 Principe du cloud computing	17
I.6.2 Services.....	19
I.7 L’Intelligence Artificielle (AI).....	22
I.8 l’Architecture générale et modes de fonctionnement des systèmes embarqués	23
I.9 Principales caractéristiques des systèmes embarqués temps réel	24
I.9.1 Exemples d’applications des systèmes embarqués temps réel.....	24

I.10 Les logiciels libres et les systèmes embarqués.....	25
I.11 Conclusion.....	25

Chapitre II : Etude et concept d'arduino

II.1 Introduction.....	23
II.2 arduino.....	23
II.2.1 Historique du projet arduino.....	23
II.2.2 Que-est-ce qu'un arduino ?.....	24
II.2.3 Architecture d'arduino.....	26
II.2.3.1 Qu'est-ce qu'un microcontrôleur ?.....	26
II.2.3.2 Structure de microcontrôleur.....	28
II.2.3.3 Carte arduino Mega2560.....	31
II.2.4 Famille d'arduino.....	33
II.2.4.1 arduino Uno.....	34
II.2.4.2 Arduino Esplora.....	34
II.2.4.3 Boarduino V2.0.....	35
II.2.4.4 Arduino Nano :.....	36
II.2.4.5 Arduino LilyPad.....	37
II.3 Exemple des projets arduino.....	38
II.4 Environnement du développement Intégré (IDE).....	40
II.4.1 Installation et setup.....	40
II.4.2 Structure d'un programme arduino.....	45
Par défaut, pendant le lancement d'un nouveau projet, on trouve la fonction void setup ().....	45
II.4.3 Démonstration réelle.....	46
II.5 Conclusion.....	49

Chapitre III : Contrôle à distance de l'énergie électrique

III.1 Introduction :.....	46
III.2 Etude sur l'électricité.....	47
III.2.1 Le flux d'électrons.....	47
III.2.2 Le courant.....	48
III.2.3 La tension :.....	49
III.2.4 Résistance.....	51
III.2.5 La loi d'Ohm.....	52
III.2.6 la puissance.....	52

III.2.7 L'énergie.....	53
III.3 Les compteurs intelligents (smart meter).....	53
III.4 Composants matériel et logiciel requis :.....	53
III.4.1 Connexion des composants à la carte Arduino Méga :.....	59
III.4.2 Tester la connexion du matériel.....	61
III.4.3 Envoi des données à google docs :.....	66
III.5 Conclusion.....	69
Conclusion générale.....	70
Bibliographie.....	71

Liste des figures

Chapitre I

Figure I- 1 : Le schéma théorique d'un tube à vide avec des tubes à vide réel.	4
Figure I- 2 : Premier transistor fabriqué en 1947	4
Figure I- 3 : Transistor actuel.....	4
Figure I- 4 : Circuits intégrés	5
Figure I- 5 : Micro-Processeur INTEL 4004.	5
Figure I- 6 : PSoC 1 c Chips PSoC 5LP Development kit	6
Figure I- 7: L'internet des objets.	8
Figure I- 8: Histoire de la technologie la connectivité des choses.	9
Figure I- 9: La maison intelligent (smart house).....	10
Figure I- 10: Transport intelligent.	11
Figure I- 11 : Smart Watch.....	11
Figure I- 12 : Compteur intelligent.	12
Figure I- 13 : Les systèmes cyber-physiques (scp) dans l'industrie.	13
Figure I- 14 : Cobot d'assistance à l'effort – Source : Usine digitale.....	14
Figure I- 15 : Les différents domaine dans les big data.	16
Figure I- 16 : Le Nuage.....	17
Figure I- 17 : Les services du Nuage.....	20
Figure I- 18 : Intelligence artificielle.	22
Figure I- 19 : Architecture générale des systèmes embarqué	24

Chapitre II

Figure II- 1 : Microconroleur ATMega328p 1.....	25
Figure II- 2: Carte arduino uno.	26
Figure II- 3 : Schéma fonctionnel d'un microcontrôleur.	30
Figure II- 4 : Schéma explicatif de microcontrôleur.....	30
Figure II- 5 : Schéma explicatif de la notion des ports.	31
Figure II- 6 : Carte arduino Mega2560.	32
Figure II- 7 : Notion des entrées et sorties dans la carte arduino.	33
Figure II- 8 : Carte Arduino uno.	34
Figure II- 9 : Carte Arduino.	35
Figure II- 10 : la carte Boarduino avec ces composantes.....	36
Figure II- 11 : Carte Arduino Nano.	37
Figure II- 12 : Carte Arduino Lilypad.....	37
Figure II- 13 : Projet ArduCAM camera 360° degree.	38
Figure II- 14 : Capture avec le projet ArduCam.	39
Figure II- 15 : Projet mesure de l'atmosphère.	39

Figure II- 16 : Drone avec arduino.	40
Figure II- 17 : Téléchargement d'IDE Arduino	41
Figure II- 18 : Téléchargement d'IDE Arduino et l'enregistrement sur le pc	41
Figure II- 19 : Début d'installation avec le programme dans le bureau.....	41
Figure II- 20 : Installation du programme partie1	42
Figure II- 21 : Installation du programme partie 2.	42
Figure II- 22 : Partie finale de l'installation.	42
Figure II- 23 : Le programme Arduino après l'ouverture.	43
Figure II- 24 : Barre des menus.....	43
Figure II- 25 : Barre des menus avec ces sous-menus partie1.....	43
Figure II- 26 : Barre des menus avec ces sous-menus partie 2.....	44
Figure II- 27 : Barre des icons.	44
Figure II- 28 : Editeur de texte.....	45
Figure II- 29 : Barre des erreurs.....	45
Figure II- 30 : Structure d'un programme dans l'IDE.	45
Figure II- 31 : Organigramme des fonction setup et Loop.	46
Figure II- 32 : Schéma explicatif de la simulation par fritzing.	47
Figure II- 33 : Choix de la carte dans IDE.	47
Figure II- 34 : Le code explicatif d'allumage de la LED.....	48
Figure II- 35 : Démonstration réelle.	48
Figure II- 36 : Allumage de la LED.....	49

Chapitre III

Figure III- 1 : Le flux de courant.....	47
Figure III- 2 : Parcours à travers un conducteur en cuivre sur une durée d'un second.....	48
Figure III- 3 : Faible flux d'électrons.....	48
Figure III- 4 : Fort courant dans le conducteur.	49
Figure III- 5 : Les électrons se déplacent en raison d'une différence de potentiel.	49
Figure III- 6 : La différence de charge ne peut être compensée, faute de liaison.....	50
Figure III- 7 : Une compensation de la différence de charge se produit	50
Figure III- 8 : Evolution d'une tension continue au cours du temps	51
Figure III- 9 : Evolution d'une tension alternative au cours du temps	51
Figure III- 10 : La résistance freinant le flux d'électrons.....	52
Figure III- 11 : Compteur intelligent (smart meter).....	53
Figure III- 12 : Module relais SDR-05VDC-SL-C.....	54
Figure III- 13 : Capteur du courant ACS712 brick.....	55
Figure III- 14 : Lampe 75watt.....	55
Figure III- 15 : le câble avec la duit utiliser.....	56
Figure III- 16 : Carte NodeMCU Dev Kit avec ses entrées et sorties.....	57
Figure III- 17 : Ajouter la carte NodeMCU à l'Arduino IDE.....	58
Figure III- 18 : Choix de la carte NodeMCU dans Arduino	58
Figure III- 19 : Connexion de VCC et le GND a la plaque d'essai.....	59
Figure III- 20 : Branchement de relais module avec la carte arduino.....	59
Figure III- 21 : Branchement de relais module avec la carte arduino.....	60

Figure III- 22 : Branchement du NODEMCU avec la carte arduino	60
Figure III- 23 : Schéma explicatif de la connexion des modules à la carte arduino.	61
Figure III- 24 : Organigramme du capteur de courant dans la carte Arduino et dans le NodeMCU.....	63
Figure III- 25 : Organigramme de la méthode de fonctionnement de NodeMCU.	64
Figure III- 26 : Schéma explicatif d’envoi du client server avec http.....	65
Figure III- 27 : Envoi du client vers le serveur.....	65
Figure III- 28 : Envoi des données a google docs	66
Figure III- 29 : Création d’une feuille a google drive	66
Figure III- 30 : Création d’un formulaire.	67
Figure III- 31 : Relier le formulaire avec la feuille 1	67
Figure III- 32 : Relier le formulaire avec la feuille 2	68
Figure III- 33:Obtenir le DEVID.....	68
Figure III- 34 : Ajouter DEVID a notre code.....	68
Figure III- 35 : Etape sans fermeture de switch dans relais.	69
Figure III- 36 : Etape avec fermeture de switch dans relais.	69

Liste des abréviations

NTICs : Nouvelle Technologies Information Communications.

GSM: Global system for mobile communication.

NASA: National Aeronautics and Space Administration.

IBM: International Business Machines.

PABX: Private Automatic Branch Exchange.

MIT: Massachusetts Institute of Technology.

RFID: Radio Frequency Identifier.

GBIF: Global Biodiversity Information facility.

SVI: Switch Virtual Interface.

ITU: International Telecommunication Union.

TCP/IP: Transport control protocol / internet protocol.

3D: Three dimension.

CDC: Construction Development Company.

Bull: c'est un nom de famille.

Fortran: FORmula TRANslator.

CAN: Controller Area Network.

CAN: Convertisseur analogique numérique.

IDE : Intelligent Developpement Environnement

RAM: Random-Access Memory

ROM: Read Only Memory

MP3: MPEG-1 Audio Layer III

GND: Ground

Wifi: Wireless Fidelity

LED: Light emitting diode

Introduction générale

Jusqu'à nos jours, la Sonalgaz envoie ses agents ou des personnes responsables pour enregistrer la quantité d'électricité consommée par les citoyens dans les grandes villes ou dans la campagne. Pour certaines maisons, le compteur d'électricité se trouve à l'intérieur, ce qui rend l'enregistrement difficile en l'absence des abonnés. Dans les grandes villes, l'employeur déploie un grand effort et un temps important pour effectuer cette opération. Pour remédier à ce problème, on fait appel à l'internet des objets qui va nous faciliter cette tâche.

Nous pouvons gagner du temps pour accomplir d'autres travaux ou projets, même s'inspirer les ressources humaines, penser aux idées innovantes. Nous pouvons facilement faire des analyses dont le but d'améliorer la consommation de cette énergie afin d'éviter le gaspillage de cette ressource sachant que l'Algérie est un pays très vaste et nous pouvons utiliser cette dernière pour améliorer notre économie en ajoutant des panneaux solaires dans le désert ou en utilisant l'énergie éolienne.

L'utilisation des capteurs indique en temps réel l'évolution de la consommation d'électricité des appareils des entreprises et des ménages, et ses effets pour le prix du courant sur le réseau électrique. Chacun peut programmer ses appareils pour qu'ils consomment moins d'énergie ou se déconnectent du réseau pendant les pics de consommation sur les lignes : on évite ainsi une montée en flèche du prix du courant ou même une baisse de tension sur le réseau, et l'on reçoit un crédit sur sa facture d'électricité du mois suivant.

Dans ce mémoire, on propose une solution à ce problème qui consiste à créer un compteur intelligent pour la télégestion de la consommation électrique. Le rôle principal de ce dernier se focalise sur le calcul de la consommation d'électricité et de l'envoyer à des stations de calcul ou à des serveurs pour collecter toutes les informations de l'abonné afin de construire le bulletin de la consommation automatiquement, rapidement et avec précision.

Environ 3,81 milliards de personnes, soit 51% de la population mondiale utilisent l'internet. Demain, nous serons des milliards de milliards. En plus, tous les objets dans notre vie seront connectés, il y aura ainsi en 2020 entre 50 et 80 milliards d'objets connectés en circulation dans le monde grâce à la révolution de l'internet qui rend notre monde une petite ville, dont l'avancement des technologies de l'information ; on cite un débit très important, des petits dispositifs avec un rendement intéressant.

Introduction Générale

Les machines deviennent très petites, puissantes et accessibles à tout le monde avec un prix raisonnable dans le but de faciliter la tâche à un être humain bien que pour gagner du temps et avancer dans le domaine de l'innovation.

Les avancées technologiques récentes confortent la présence de l'informatique et de l'électronique au cœur du monde réel, de plus en plus d'objets se voient ainsi équipés de processeurs et de moyens de communication mobiles, leur permettant de traiter des informations mais également de les transmettre. Si on prend, par exemple notre pays, on est les derniers dans cet avancement rapide, il faut le rattraper avant que nous serons très loin de tout ça.

Cette évolution s'inscrit dans le cadre de **l'information pervasive**, plus connue sous le nom **d'ubiquitous computing**. Un des objectifs de ce domaine est de combler le fossé entre les mondes réel et virtuel en rendant les objets intelligents, dont le but de faire une collaboration humain-machine pour trouver des solutions à des problèmes souvent complexes.

Notre motivation pour ce sujet c'est l'analyse des données pour trouver les meilleures solutions avec les ressources qu'on dispose en Algérie aussi que l'implémentation des objets connectés qui constituera l'avenir après la disparition du pétrole.

Notre mémoire est organisé en trois chapitres :

Le premier chapitre présente des généralités sur les systèmes embarqués en temps réel, l'internet des objets, les systèmes cyber-physiques, big data, le cloud (nuage), l'intelligence artificielle, et le domaine d'application pour chaque technologie.

Le deuxième chapitre se divise en trois parties :

- La première concerne une présentation sur l'historique de la carte Arduino.
- La deuxième traite les familles de l'arduino avec des explications de chaque carte et ses caractéristiques, et les projets réalisés par la carte arduino.
- La dernière partie consiste en une démonstration réelle avec nos matériels ; il s'agit d'allumer une LED.

Le dernier chapitre décrit la notion de l'électricité et la réalisation pratique de notre projet ayant pour but d'utiliser et de programmer la carte arduino et la carte NodeMCU ainsi que l'envoi des données à Google cloud. Ce chapitre contient :

- Une partie électronique dans laquelle nous allons décrire l'ensemble des modules électroniques utilisés tels que le relais, le capteur du courant, et le NodeMCU
- Une partie informatique consacrée à programmer la carte arduino et le NodeMCU, pour lire le courant électrique puis calculer la puissance afin de déduire l'énergie.
- Finalement, vient l'étape qui consiste à connecter le cloud avec notre système pour l'envoi des informations nécessaires.

Chapitre I

Les systèmes embarqués



Chapitre 1 : Les systèmes embarqués

I.1 Introduction

Un système embarqué (SE) est un système informatisé spécialisé qui constitue une partie intégrante dans un autre large système ou un dans une machine. C'est un système sur un processeur avec une mémoire en quelque kilo-octets dont les programmes sont stockés sur la ROM. On peut dire que SE est un système ayant une interface digitale comme une montre, une voiture, une camera etc.

Certains SE ont des systèmes d'exploitation comme Raspberry Pi qu'on peut l'ajouter dans une carte mémoire. Il y a plusieurs systèmes d'exploitation dédiés au système embarqués citons : linux windows10 IOT, RTEMS, FreeRTOS, chibiOS /RT etc...., et d'autres sans systèmes d'exploitation mais programmable comme arduino qu'on va introduire dans le prochain chapitre car toute leur logique peut être implantée en un seul programme ou une seule fonction.

D'une autre façon, un système embarqué est une combinaison de logiciel et matériel, avec des capacités fixes ou programmables, qui est spécialement conçu pour un type d'application particulier comme les distributeurs automatiques de boissons, les automobiles, les équipements médicaux, les caméras, les téléphones ...

Les SE programmables sont dotés d'interfaces de programmation et leur programmation est une activité spécialisée en utilisant plusieurs langages de programmation tels que le c, c++ etc.

Il existe des systèmes embarqués qui nécessitent un traitement en temps réel dédiés à la télégestion, au contrôle ou à la supervision d'un phénomène bien précis. Ces systèmes assurent la mobilité, une faible consommation d'énergie, un poids léger, un volume compact et un cout faible.

Avant d'aborder les détails sur des systèmes embarqués, on parle un peu sur l'historique de cette technologie [1].

I.2 Historique des systèmes embarqués

I.2.1 Les anciens Génération

I.2.1.1 La première génération - les tubes à vides (1945-1955)

En 1904, John Flemming invente le tube à vide ; c'est un interrupteur électronique caractérisé par les pannes fréquentes, et il est encombrant .

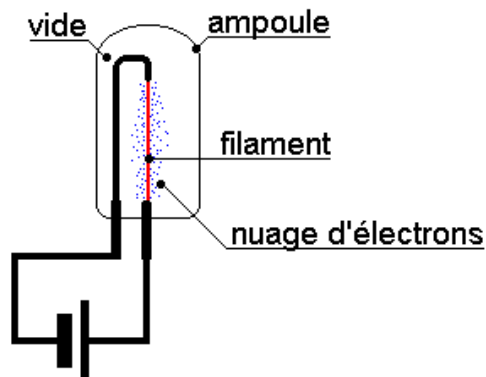


Figure I- 1 : Le schéma théorique d'un tube à vide avec des tubes à vide réel.

I.2.1.2 Deuxième génération - les transistors (1955-1965)

L'année 1948 a connu l'invention du transistor aux Bell Labs par Bardeen et Shockley avec l'interrupteur commandé électroniquement. Le transistor est beaucoup moins encombrant que le tube à vide. Cette génération est caractérisée par la programmation en langage machine, la microprogrammation, la représentation des nombres en virgule flottante. Pour les logiciels, il y a l'apparition des systèmes d'exploitation, naissance du langage de programmation en assembleur. Le premier langage de haut niveau était le **FORTRAN**. Cette génération a connu le début des grandes firmes telles que **IBM, BULL, DEC, CDC** etc. [2].



Figure I- 2 : Premier transistor fabriqué en 1947



Figure I- 3 : Transistor actuel

I.2.1.3 Troisième génération - les circuits intégrés (1965-1980)

En 1961, est apparu le premier circuit intégré sur silicium fabriqué par R. Noyce. Ce dernier causa un moindre encombrement, moindre consommation avec un gain en performances.

D'autres part, cette génération est marquée par la compatibilité des machines d'une même gamme (**IBM : System/360**), l'introduction du parallélisme (multiprogrammation), l'apparition des

systèmes à multi-processeurs pour faciliter la gestion des tâches, l'introduction de plusieurs langages de programmation tels que pascal, c, basic etc. Il y a plusieurs exemples d'architectures, on cite quelques-uns : IBM 360, CDC 6000, PDP 11 [3] ...

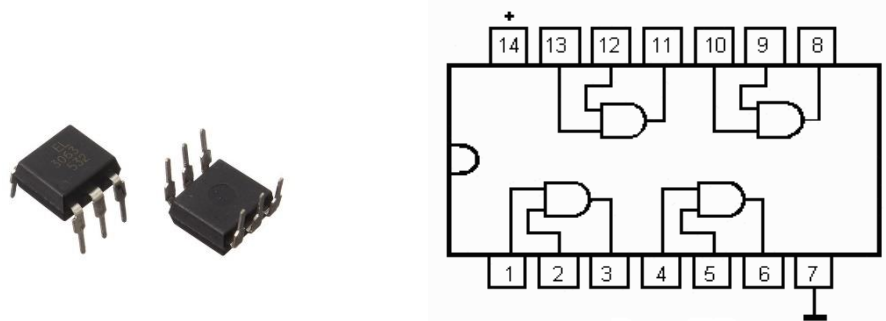


Figure I-4 : Circuits intégrés

I.2.1.4 Quatrième génération - les microprocesseurs 4004 (1970-2000)

La présence des systèmes embarqués dans notre vie est rendue possible grâce à une évolution dans le domaine d'électronique, spécialement dans la numérisation dans les années 70.

Les processeurs deviennent très rapides, puissants avec un coût abordable. Cette évolution dans ce domaine guide les chercheurs et les scientifiques à d'autres perspectives dans le domaine de technologie.

En 1971, Intel produit le premier microprocesseur 4004, c'est un circuit générique, programmé, il a été implémenté dans les calculatrices commerciales. L'apparition du processeur a donné une naissance aux ordinateurs portables qui sont devenus les machines actuelles.

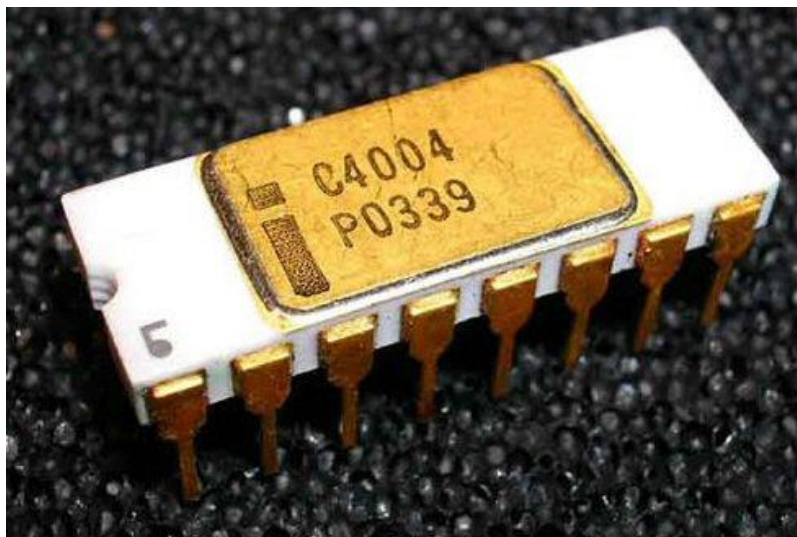


Figure I-5 : Micro-Processeur INTEL 4004.

I.2.1.5 Cinquième génération - PSoC (2000 – 2010)

PSoC est l'acronyme de Programmable System on Chip. C'est un système électronique propriétaire de Cypress. Les circuits PSoC ont été introduits au début des années 2000 et sont conçus pour remplacer à la fois le microcontrôleur et les circuits périphériques d'un système embarqué.

Les PSoC contiennent des blocs analogiques (majoritairement à capacités commutées et numériques) configurables par l'utilisateur permettant de les intégrer.

Donc, l'idée c'est de remplacer le microcontrôleur et les circuits interfaces analogiques (CAN et CNA, filtres, amplificateurs ...) ou numériques (compteurs, timers, bus du drivers ...) associés par un circuit unique. On intègre ainsi un système électronique embarqué complet afin de réduire le nombre de composants électronique[4].

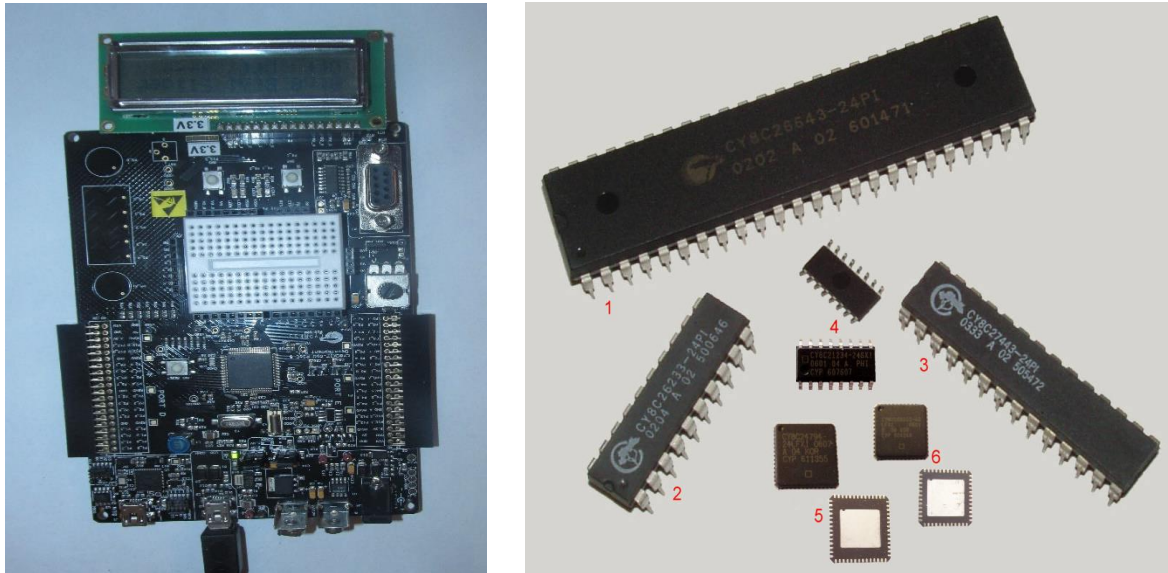


Figure I- 6 : PSoC 1/c Chips PSoC 5LP Development kit

La figure I-6 illustre quelques exemples du PSoC :

- 1 CY8C26643-24PI (48 pin DIP)
- 2 CY8C26633-24PI (20 pin DIP)
- 3 CY8C27443-24PI (28 pin DIP)
- 4 CY8C21234-24SWI (16 pin DIP)
- 5 CY8C24794-24LFXI (56 pin QFN)
- 6 CYWUSB6953-48LFXC (48 pin QFN)

I.2.1.6 Sixième génération (2010 – aujourd’hui)

Dans ces dernières années, il y a une appellation qu’on entend beaucoup le terme de l’innovation et toute vers entrepreneuriat. Grâce au développement rapide dans le domaine de l’information et technologies, les appareils deviennent très petits et accessibles à tout le monde. Avec l’avènement des circuits programmables et la facilité de programmer un circuit, une personne qui ne sait ni programmation ni électronique, peut construire son propre projet afin de le commercialiser dans le marché. Cette évolution nous guide à continuer de développer des nouvelles idées.

I.3. Objets communicants & Internet des Objets (Internet Of Things)

Aujourd’hui, on utilise de nombreux objets communicants dans notre quotidien. Nous sommes entourés de ces appareils pouvant communiquer avec leur environnement et échanger des données, qui nous offrent de plus en plus de services facilitant nos activités, et avec lesquels nous interagissons fréquemment. Ils sont en train de transformer nos habitudes, nos comportements, et plus globalement nos sociétés en général. Les objets communicants sont d’une grande variété : ils peuvent aller d’une simple carte à puce, comme les cartes sans contact qu’on utilise pour accéder aux transports en commun, ou bien les Smartphones, qui sont aujourd’hui de véritables petits bijoux de technologie offrant beaucoup plus de services que la simple téléphonie, ou encore les télévisions connectées qui sont capables d’aller chercher des contenus sur Internet pour notre plus grand bonheur. Ces objets communicants vont se retrouver dans tous les domaines, allant du domaine grand public tels les objets de nos maisons, qui deviennent de plus en plus intelligents, jusqu’à la ville intelligente d’une manière générale.

Le monde industriel va être révolutionné grâce à ces nouveaux objets, qui vont offrir plusieurs opportunités économiques pour un marché colossal se chiffrant en milliards d’euros, et qui intéresse tous les acteurs des NTICs [5].

I.3.1 Objets communicants

Pour bien comprendre le terme objet connecté ou communicant, on répond à cette question : qu’appelle-t-on un objet communicant ?

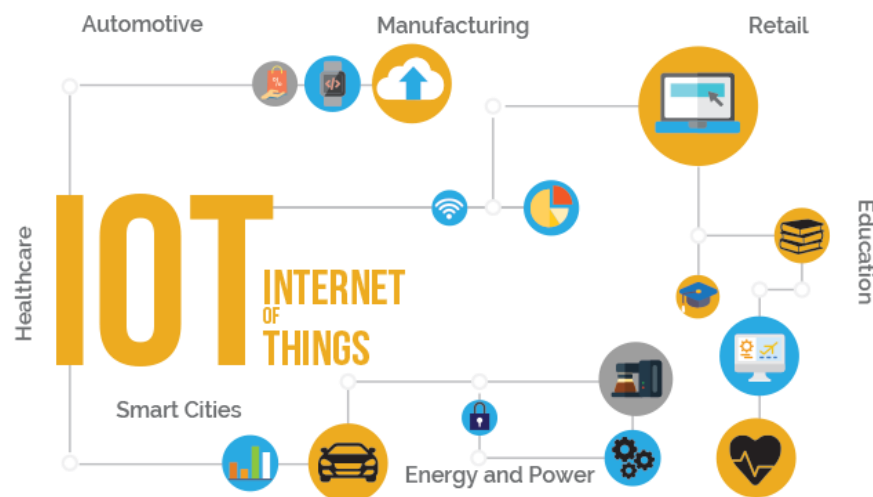
On peut définir un objet communicant comme un appareil pouvant interagir avec son environnement en répondant à des stimuli, et échanger avec ses pairs en utilisant un support de communication. Un objet communicant est doté d’une unité de calcul, plus ou moins sophistiquée

selon l'objectif souhaité, et d'un ou plusieurs périphériques de communication qui vont permettre à cet objet d'envoyer et de recevoir des données de l'extérieur.

En plus d'un ensemble de capteurs qui vont jouer le rôle de périphériques d'entrée, tels un capteur de luminosité ou un capteur de mouvement, ils possèdent également un ensemble d'actionneurs, ou périphériques de sortie, comme le bras motorisé d'un robot, ou tout simplement un écran d'affichage sur un Smartphone. Les périphériques de communication peuvent être filaires ou sans fil. On s'attend à avoir plusieurs dizaines de milliards d'objets communicants à l'horizon de l'an 2020. Ils élargiront l'accès à l'information des périphériques classiques, et dans ce cas, on va plutôt parler « d'Internet des Objets » [5].

I.3.2 Internet des Objets (Internet of Things)

L'internet des objets est un ensemble de concepts, d'entités réseau, de technologies et d'architectures de communications et de services permettant la connexion du monde physique/réel au monde digital.



We strive to transform life and improve performance and efficiencies through IOT.

Figure I- 7: L'internet des objets.

Nous distinguons deux visions. Une première vision à court et moyen terme caractérisée par l'extension du modèle de communication Internet **TCP/IP** vers de nouveaux objets, les objets du quotidien dotés d'une interface de communication, ou nouveaux objets conçus pour l'acquisition d'informations de l'environnement. Une deuxième vision à long terme caractérisée par la création de nouveaux modèles de communication adaptés aux nouvelles contraintes de l'Internet des

Objets, les objets avec ressources réduites comme l'énergie, explosion du nombre d'objets connecté. Quand on parle de l'Internet des Objets, on retrouve l'historique du terme qui démarre au MIT dans le cadre de travaux du AutoID lab sur les aspects digitalisation des processus dans les grandes chaînes de distribution. Ensuite, vient le premier rapport en 2005 du monde des télécommunications par l'ITU, l'organisme mondial de standardisation des télécommunications. L'évolution technologique autour de l'Internet des Objets, comme indiqué sur la figure I-7, montre bien les premières applications de l'IoT dans le milieu des chaînes de distribution avec l'introduction massive de la technologie **RFID**. Les prochaines applications sont autour de l'exploitation des technologies de capteurs, ensuite les services basés sur les informations de localisation, et sur l'horizon 2020, on attend des services de contrôle-commande à distance des objets connectés dans différents secteurs verticaux [5].

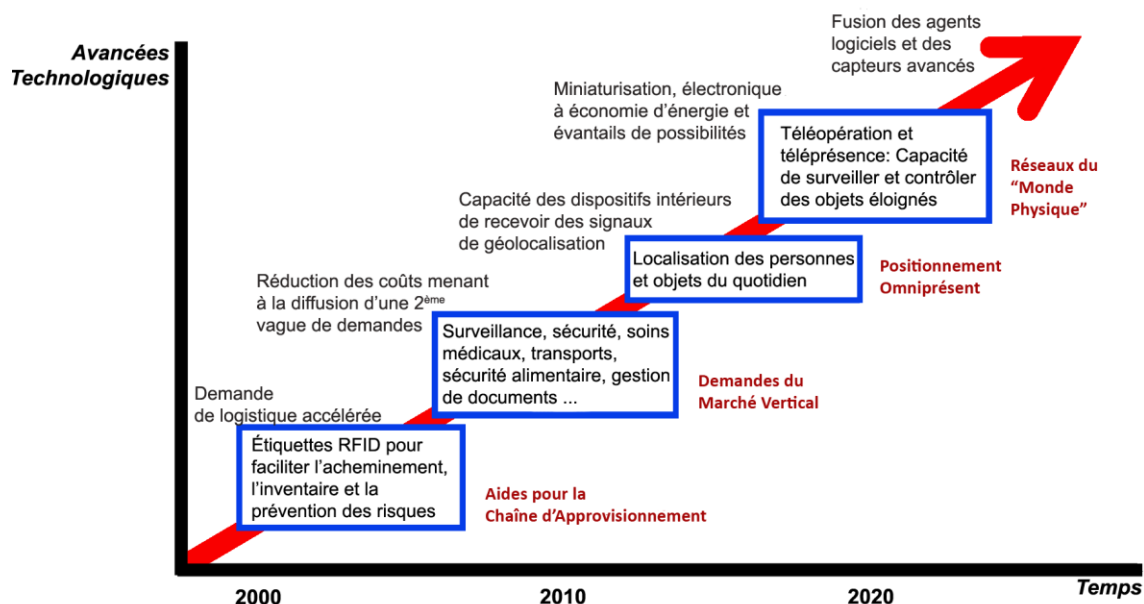


Figure I- 8: Histoire de la technologie la connectivité des choses.

I.3.3 Des exemples dans le domaine d'application

L'une des applications phares, qui touche un large public, est certainement la maison intelligente ou « bâtiment intelligent », avec ses nombreux objets communicants qui vont offrir de nouveaux services domotique, pour un meilleur contrôle des équipements et une utilisation optimale de l'énergie. La température ambiante intérieure sera contrôlée à la fois en fonction de la température extérieure et en fonction de la présence de personnes dans la maison. De même, éclairer, climatiser des locaux en l'absence de personne physique dans la maison sera inutile [6].



Figure I- 9: La maison intelligente (smart house)

Un autre exemple d'application est celui des transports intelligents. Nos voitures embarquant de nombreux équipements électroniques seront bientôt communicantes, et pourront dialoguer entre elles, ainsi qu'avec les infrastructures de la route. Cet échange permettra d'améliorer la sécurité routière en offrant des informations pertinentes en temps réel aux conducteurs, pour éviter des accidents ou des embouteillages. Cette meilleure gestion des transports aura un impact très fort sur l'économie. Le monde du travail et le monde industriel seront aussi impactés par les objets communicants connectés. Les usines intelligentes, truffées de capteurs communicants, collecteront des données en temps réel et surveilleront ainsi l'état des différents équipements entrant dans le processus de production [7].



Figure I- 10: Transport intelligent.

Internet des objets, avec les différents capteurs embarqués, soit à travers les Wearables ou des capteurs proches du corps, forme aujourd'hui une source d'information en temps réel d'une valeur inestimable pour l'innovation dans ce secteur. On peut citer la Startup Française Withings, c'est une montre intelligente entourée des capteurs (capteur cardiaque, capteur de localisation et autre pour calculer le mouvement ou l'activité donnée) qui a développé dès le début de l'IoT la balance intelligente avec un coach en ligne à travers une application mobile. Il est important de rappeler que l'explosion des services **IoT** autour de la personne ont été rendus possibles aussi grâce aux interfaces tactiles des Smartphones et tablettes, sur lesquelles des applications mobiles avec des interfaces très ludiques rendent l'interaction entre l'utilisateur et le service très favorable au déploiement de ces réseaux **IoT** [5].



Figure I- 11 : Smart Watch

ERDF a mis en place le programme LINKY, le compteur électrique intelligent qui mesure les données de consommation énergétique des utilisateurs et communique en temps réel ces informations. Un backoffice traite ces données aussi en temps réel et différents services novateurs

sont alors rendus possibles : la prédiction du volume d'énergie nécessaire dans une maison ou un quartier par exemple. Un autre aspect concerne l'intégration des différentes sources d'énergie renouvelable dans le réseau électrique, le dimensionnement intelligent de l'énergie globale, et sa distribution vers les clients finaux [5].

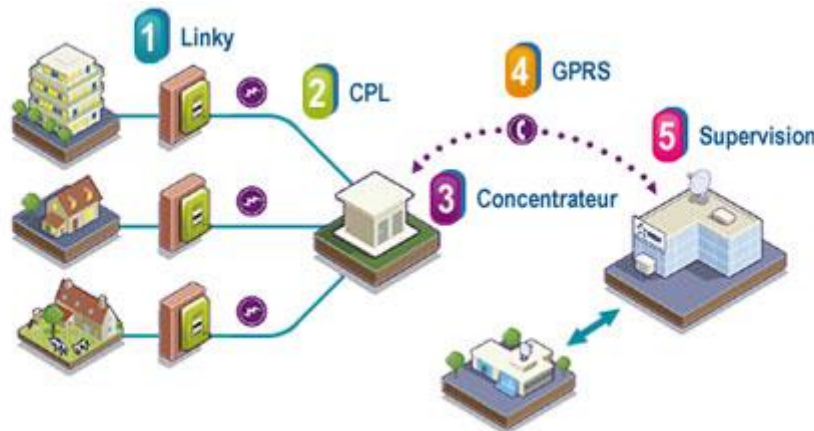


Figure I- 12 : Compteur intelligent.

I.3.4 La sécurité des IoT

Un attaquant ayant physiquement accès à un objet connecté est en mesure de recueillir beaucoup de ses informations sensibles. S'il réussissait par exemple à récupérer ses clés de chiffrement, il pourrait accéder à tout le trafic entrant et sortant de l'objet, et il pourrait aussi injecter du code malveillant destiné à d'autres objets du réseau. Chaque objet connecté apparaît ainsi comme un point critique dans l'architecture de l'IoT. Nous énumérons dans ce qui suit les différentes propriétés de sécurité et de protection de la vie privée qui devraient être garanties afin de sécuriser un objet connecté. Parmi les concepts importants utilisés dans la suite, nous rappelons la notion d'identité. Dans l'IoT, les objets intelligents sont considérés comme des entités indépendantes, capables d'agir au nom d'un utilisateur. Il existe plusieurs définitions dans la littérature, concernant principalement l'identité et l'identité partielle des objets intelligents. L'identité permet d'une part de distinguer les différents objets à l'intérieur du réseau, et d'autre part de vérifier leur origine. Dans toute architecture de gestion d'identité, l'établissement d'un environnement de confiance nécessite l'unicité des identités afin de pouvoir les authentifier. Les ressources contraintes des objets imposent cependant des extensions à la gestion traditionnelle d'identité, donc pour renforcer la sécurité, il faut utiliser les plus récents algorithmes de chiffrement, aussi l'utilisation de la sécurité en 3D [8].

I.4 Les Systèmes Cyber-Physiques (SCP)

Les Systèmes Cyber-Physiques sont des systèmes devant interconnecter le mode virtuel, l'internet et l'information, avec le monde réel, et ses caractéristiques physiques.

Les SCP sont une extension des systèmes embarqués, fortement distribués et communicants, réactifs et proactifs, adaptatifs à leurs environnements, y compris humains.

Les SCP possèdent de multiples capteurs et actionneurs et de fortes capacités de traitement et de stockage à travers l'Internet des Objets, toutes les bonnes propriétés nécessaires (e.g. performance, sécurité et résilience) pour mesurer, analyser et commander etc.[9].

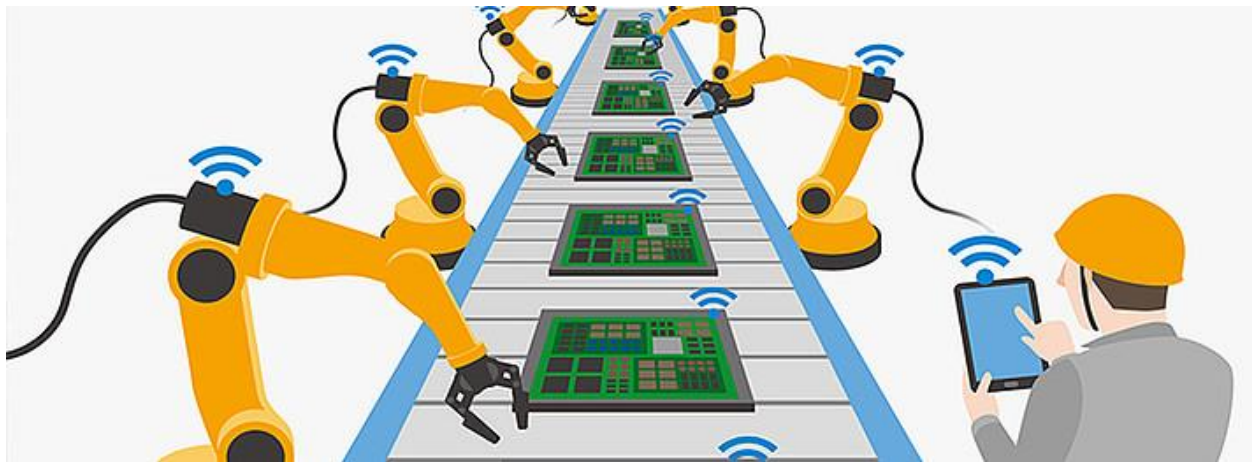


Figure I- 13 : Les systèmes cyber-physiques (scp) dans l'industrie.

I.4.1 Exemple d'application dans les systèmes cyber-physique

Le robot collaboratif est un système moins onéreux, moins contraignant et plus flexible que le robot classique ; son rôle est d'assister les humains dans leur travail.

La grande différence entre un robot industriel classique et un cobot, ce n'est pas le fait que le cobot dispose de deux bras et d'une tête équipée de caméras, mais sa capacité à travailler parmi les ouvriers, en toute sécurité. Qu'il s'agisse d'humanoïdes, de bras robots ou de portiques robotisés, cette faculté va vraisemblablement tout changer dans les ateliers [9].



Figure I- 14 : Cobot d'assistance à l'effort – Source : Usine digitale

I.5 Les données massives (Big Data)

Le Big Data, littéralement « grosses données » ou mégadonnées parfois appelées données massives, désigne des ensembles de données devenus si volumineux qu'ils dépassent l'intuition et les capacités humaines d'analyse et même celles des outils informatiques classiques de gestion de base de données ou de l'information. L'explosion quantitative et souvent redondante de la donnée numérique contraint à de nouvelles manières de voir et analyser le monde, de nouveaux ordres de grandeur concernent la capture, le stockage, la recherche, le partage, l'analyse et la virtualisation des données. Les perspectives du traitement des big data sont énormes et en partie encore insoupçonnées ; on évoque souvent de nouvelles possibilités d'exploration de l'information diffusée par les médias, de connaissance et d'évaluation, d'analyse tendancielle et prospective (climatiques, environnement ou encore sociopolitiques etc.) et la gestion des risques (commerciaux, assuranciers, industries, naturels) mais aussi en termes de génomique ou métagénomique, pour la médecine (compréhension du fonctionnement du cerveau, épidémiologie, éco épidémiologie...), la météorologie et l'adaptation aux changements climatiques, la gestion de réseaux énergétiques complexes (via les smartgrids, ou un futur « internet de l'énergie ») l'écologie (fonctionnement et dysfonctionnement des réseaux écologiques, des réseaux trophiques avec le **GBIF** par exemple), ou encore la sécurité et la lutte contre la criminalité. La multiplicité de ces applications laisse d'ailleurs déjà poindre un véritable écosystème économique impliquant, d'ores et déjà, les plus gros joueurs du secteur des technologies de l'information. Certains supposent que le big data pourrait aider les entreprises à réduire leurs risques et faciliter la prise de décision, ou créer la différence grâce à l'analyse prédictive et une « expérience client » plus personnalisée et contextualisée ^[10].

Divers experts, grandes institutions (comme le MIT aux États-Unis), administrations et spécialistes sur le terrain des technologies ou des usages considèrent le phénomène big data comme l'un des grands défis informatiques de la décennie 2010-2020 et en ont fait une de leurs nouvelles priorités de recherche et développement, qui pourrait notamment conduire à l'intelligence artificielle en étant exploré par des réseaux de neurones artificiels autoapprenants ^[11].

Big Data est un champ dédié à l'analyse, au traitement et au stockage de grandes collections de données provenant souvent de sources disparates. Les solutions et les pratiques Big Data sont généralement requises lorsque les technologies et techniques traditionnelles d'analyse, de traitement et de stockage des données sont insuffisantes. Plus précisément, les Big Data répondent à des exigences distinctes, telles que la combinaison de plusieurs ensembles de données sans rapport, le traitement de grandes quantités de données non structurées et la collecte d'informations cachées de manière sensible au temps. Bien que le Big Data puisse apparaître comme une nouvelle discipline, il se développe depuis des années. La gestion et l'analyse d'ensembles de données volumineux constituent un problème de longue date - des approches laborieuses des premiers efforts de recensement à la science actuarielle derrière les calculs des primes d'assurance. La science du Big Data a évolué à partir de ces racines. En plus des approches analytiques traditionnelles basées sur les statistiques, Big Data ajoute de nouvelles techniques qui tirent parti des ressources et des approches de calcul pour exécuter des algorithmes analytiques. Ce changement est important car les ensembles de données continuent à être plus grands, plus diversifiés, plus complexes et centrés sur le streaming. Alors que les approches statistiques ont été utilisées pour estimer les mesures d'une population par échantillonnage depuis les temps bibliques, les progrès de la science computationnelle ont permis le traitement d'ensembles de données entiers, rendant un tel échantillonnage inutile. L'analyse des ensembles de données Big Data est une entreprise interdisciplinaire qui combine les mathématiques, les statistiques, l'informatique et l'expertise en la matière. Ce mélange de compétences et de perspectives a conduit à une certaine confusion quant à ce qui constitue le domaine du Big Data et de son analyse, car la réponse que l'on reçoit dépendra de la perspective de celui qui répond à la question. Les limites de ce qui constitue un problème de Big Data changent également en raison du paysage en constante évolution de la technologie logicielle et matérielle. Ceci est dû au fait que la définition du Big Data prend en compte l'impact des caractéristiques des données sur la conception de l'environnement

de la solution elle-même. Il y a trente ans, un gigaoctet de données pouvait constituer un problème de Big Data et nécessiter des ressources informatiques spéciales. Maintenant, les gigaoctet de données sont monnaie courante et peuvent être facilement transmis, traités et stockés sur des appareils destinés aux consommateurs. Les données dans les environnements de Big Data s'accumulent généralement à partir d'être accumulées dans l'entreprise via des applications, des capteurs et des sources externes. Les données traitées par une solution Big Data peuvent être utilisées directement par les applications d'entreprise ou peuvent être introduites dans un entrepôt de données pour y enrichir les données existantes. Les résultats obtenus grâce au traitement des Big Data peuvent conduire à un large éventail d'idées et d'avantages, tels que l'optimisation opérationnelle, l'intelligence exploitable, l'identification de nouveaux marchés, les prévisions précises, la détection de fautes et de fraudes, les enregistrements, la meilleure prise de décision et des découvertes scientifiques. Évidemment, les applications et les avantages potentiels du Big Data sont vastes. Cependant, de nombreux problèmes doivent être pris en compte lors de l'adoption des approches analytiques Big Data ^[12].



Figure I- 15 : Les différents domaines dans les big data.

I.6 L'informatique en nuage (Cloud Computing)

Le cloud computing, ou l'informatique en nuage ou nuagique ou encore l'infonuagique (au Canada francophone), consiste à exploiter la puissance de calcul ou de stockage de serveurs informatiques distants par l'intermédiaire d'un réseau, généralement Internet. Les serveurs sont loués à la demande, le plus souvent par tranche d'utilisation, selon des critères techniques (puissance, bande passante, etc.), mais, également, au forfait. Le cloud computing se

caractérisé par sa grande souplesse selon le niveau de compétence de l'utilisateur client, il est possible de gérer soi-même son serveur ou de se contenter d'utiliser des applicatifs distants en mode SaaS.

Les principaux services proposés en cloud computing sont le SaaS (Software as a Service), le PaaS (Platform as a Service) et le IaaS (Infrastructure as a Service). En fonction du service, les systèmes d'exploitation, les logiciels d'infrastructure et les logiciels applicatifs seront de la responsabilité soit du fournisseur soit du client.

Les grandes entreprises du secteur informatique se sont lancées dans le cloud computing et ont investi massivement pour offrir de la puissance de calcul et de stockage. Cela constitue un important changement de paradigme des systèmes informatiques, jusque-là constitués de serveurs situés au sein même de l'entreprise [13].

I.6.1 Principe du cloud computing

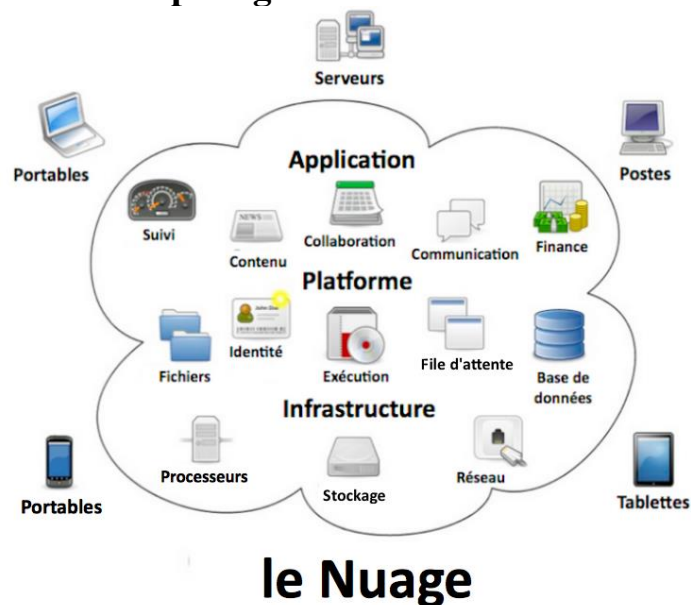


Figure I- 16 : Le Nuage.

Un cloud (« nuage ») est un ensemble de matériels, de raccordements réseau et de logiciels qui fournit des services que les individus et les collectivités peuvent exploiter à volonté depuis n'importe où dans le monde. Le cloud computing est un basculement de tendance : au lieu d'obtenir de la puissance de calcul par acquisition de matériel et de logiciel, le consommateur se sert de puissance mise à sa disposition par un fournisseur via Internet.

Les caractéristiques essentielles d'un nuage sont la disponibilité mondiale en libre-service, l'élasticité, l'ouverture, la mutualisation et le paiement à l'usage :

- Ressources en libre-service : et adaptation automatique à la demande. La capacité de stockage et la puissance de calcul sont adaptées automatiquement au besoin d'un consommateur. Cela contraste avec la technique classique des hébergeurs où le consommateur doit faire une demande écrite à son fournisseur en vue d'obtenir une augmentation de la capacité - demande dont la prise en compte nécessite évidemment un certain temps. En cloud computing, la demande est automatique et la réponse est immédiate ;
- Ouverture : les services de cloud computing sont mis à disposition sur Internet et utilisent des techniques standardisées qui permettent de s'en servir aussi bien avec un ordinateur qu'un téléphone ou une tablette ;
- Mutualisation : elle permet de combiner des ressources hétérogènes (matériel, logiciel, trafic réseau) en vue de servir plusieurs consommateurs à qui les ressources sont automatiquement attribuées. La mutualisation améliore l'évolutivité et l'élasticité et permet d'adapter automatiquement les ressources aux variations de la demande ;
- Paiement à l'usage : la quantité de service consommée dans le cloud est mesurée, à des fins de contrôle, d'adaptation des moyens techniques et de facturation.

Les nuages utilisent des technologies telles que la virtualisation du matériel informatique, les grilles, l'architecture orientée services et les services Web. Un nuage peut être public, privé ou communautaire. Un nuage public est mis à disposition du grand public. Les services sont généralement mis à disposition par une entreprise qui manipule une infrastructure lui appartenant. Un nuage privé est destiné exclusivement à une organisation qui peut le manipuler elle-même ou faire appel à des services fournis par des tiers. Dans un nuage communautaire, l'infrastructure provient d'un ensemble de membres qui partage un intérêt commun. Ce type de nuage est semblable à ceux montés par les milieux universitaires pour des études de grande envergure.

Un exemple de service grand-public, fourni en cloud computing, est le jeu à la demande (aussi appelé jeu sur demande, et, en anglais, gaming on demand (GoD) ou cloud gaming). Il permet de jouer normalement à des jeux vidéo sur son écran d'ordinateur, alors que le ou les logiciels de jeu tournent sur des serveurs à distance, qui renvoient la vidéo de ce qui a été joué en lecture en continu (en anglais streaming). Le jeu est hébergé et stocké sur des serveurs, dont l'utilisateur ne connaît

pas la localisation ni les caractéristiques. Il ne nécessite plus de supports, comme les CD, ou de matériel, comme les consoles de jeux. Les joueurs doivent seulement posséder un ordinateur relié à Internet, et les périphériques requis (clavier, souris, manette de jeu...).

La désignation « cloud computing » est issue de professionnels anglophones de l'informatique à la recherche d'un nom pour les nouveaux systèmes informatiques fonctionnant par l'action conjointe d'éléments disparates réunis indépendamment de leur localisation géographique et de l'infrastructure sous-jacente. Le nom vient du symbole en forme de nuage (« cloud ») servant à représenter Internet dans les schémas des réseaux informatiques.

I.6.2 Services

Du point de vue économique, le cloud computing est essentiellement une offre commerciale d'abonnement économique à des services externes. Selon le National Institute of Standards and Technology aux États-Unis, il existe trois catégories de services qui sont proposées en cloud computing : IaaS, PaaS et SaaS.

- **IaaS (infrastructure as a service)**

IaaS est un acronyme de « Infrastructure as a Service », en français, infrastructure en tant que service. C'est le service de plus bas niveau. Il consiste à offrir un accès à un parc informatique virtualisé. Des machines virtuelles sur lesquelles le consommateur peut installer un système d'exploitation et des applications. Le consommateur est ainsi dispensé de l'achat de matériel informatique. Ce service s'apparente aux services d'hébergement classiques des centres de traitement de données, et la tendance est en faveur de services de plus haut niveau, qui font davantage abstraction de détails techniques.

- **PaaS (platform as a service)**

PaaS est un sigle de « Platform as a Service », en français plate-forme en tant que service. Dans ce type de service, situé juste au-dessus du précédent, le système d'exploitation et les outils d'infrastructure sont sous la responsabilité du fournisseur. Le consommateur a le contrôle des applications et peut ajouter ses propres outils. La situation est analogue à celle de l'hébergement Web, où le consommateur loue l'exploitation de serveurs sur lesquels les outils nécessaires sont préalablement placés et contrôlés par le fournisseur. La différence étant que les systèmes sont mutualisés et offrent une grande élasticité - capacité de s'adapter automatiquement à la demande,

alors que, dans une offre classique d'hébergement Web, l'adaptation fait suite à une demande formelle du consommateur.

- **SaaS (software as a service)**

SaaS est le sigle de « Software as a Service », en français logiciel en tant que service. Dans ce type de service, des applications sont mises à la disposition des consommateurs. Les applications peuvent être manipulées à l'aide d'un navigateur Web ou installées de façon locative sur un PC, et le consommateur n'a pas à se soucier d'effectuer des mises à jour, d'ajouter des patches de sécurité et d'assurer la disponibilité du service. Gmail est un exemple de tel service. Il offre au consommateur un service de courrier électronique, et le consommateur n'a pas à se soucier de la manière dont le service est fourni. Autre exemple, Office 365 propose un ensemble de services en abonnement dont la suite logicielle Office, qui se met automatiquement à jour, l'utilisateur ne se soucie pas de racheter un nouveau logiciel ou de le mettre à jour. On parle ici de location de services hébergés par Microsoft. D'autres exemples de logiciels mis à disposition en SaaS sont Google Apps, Office Online ou LotusLive (IBM).

Un fournisseur de software as a service peut exploiter des services de type platform as a service, qui peut lui-même se servir d'infrastructure as a service.

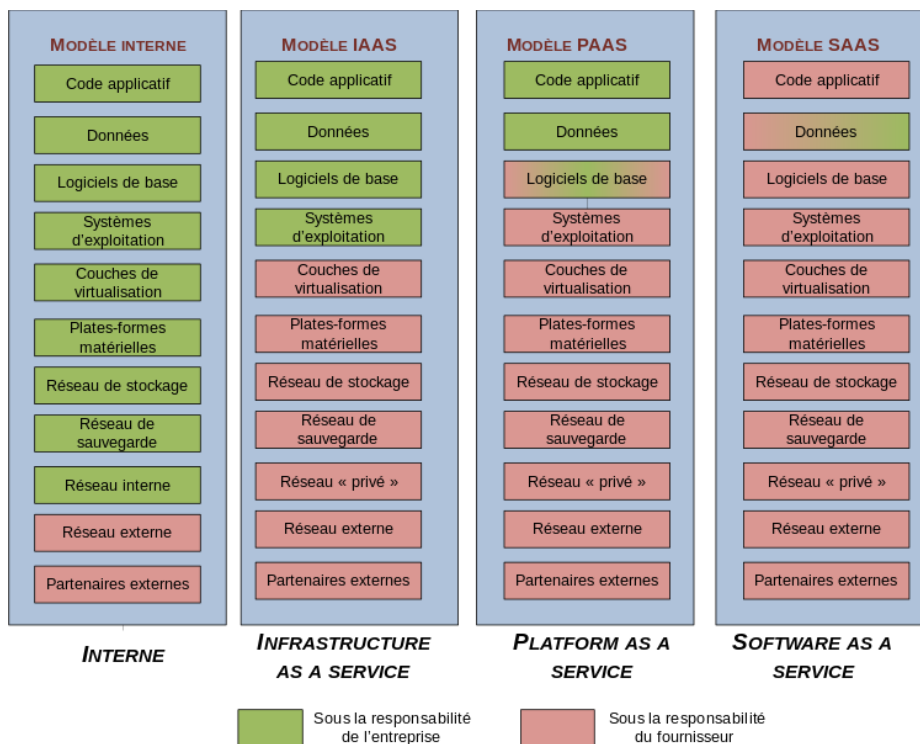


Figure I- 17 : Les services du Nuage.

D'autres services sont également disponibles :

- **Data as a service**

Ce service correspond à la mise à disposition de données délocalisées quelque part sur le réseau. Ces données sont principalement consommées par ce que l'on appelle des mashups.

- **Business process as a service (BPaaS)**

Il s'agit du concept de **BPaaS**, qui consiste à externaliser une procédure d'entreprise suffisamment industrialisée pour s'adresser directement aux managers d'une organisation, sans nécessiter l'aide de professionnels de l'informatique.

- **Desktop as a service (DaaS)**

Aussi appelé en français « bureau en tant que service », « bureau virtuel » ou « bureau virtuel hébergé » est l'externalisation d'une virtual desktop infrastructure auprès d'un fournisseur de services. Généralement, le desktop as a service est proposé avec un abonnement payant.

- **Network as a service (NaaS)**

Le network as a service correspond à la fourniture de services réseaux, suivant le concept de software defined networking (SDN).

- **Storage as a service (STaaS)**

STaaS : Storage as a Service correspond au stockage de fichiers chez des prestataires externes qui les hébergent pour le compte de leurs clients. Des services grand public tels que Microsoft OneDrive, SugarSync et Box.NET proposent ce type de stockage le plus souvent exemples : Microsoft SharePoint, Amazon S3, Dropbox, Google Drive, HubiC, iCloud, Ubuntu One, Windows Live Mesh, Wuala.

- **Communication as a service (CaaS)**

Il correspond à la fourniture de solutions de communication substituant aux matériels et serveurs locaux (**PABX, SVI...**) des ressources partagées sur Internet.

- **Workplace as a service (WaaS)**

Les caractéristiques du cloud sont qualifiées par les anglophones sous le vocable elastic computing capacity. Le National Institute of Standards and Technology en a donné une définition succincte qui reprend ces principes de base : « L'informatique dans les nuages est un modèle permettant d'établir un accès par le réseau à un réservoir partagé de ressources informatiques standard configurables (réseau, serveurs, stockage, applications et services) qui peuvent être rapidement

mobilisées et mises à disposition en minimisant les efforts de gestion ou les contacts avec le fournisseur de service [14].

I.7 L'Intelligence Artificielle (AI)

L'intelligence artificielle (IA) est « l'ensemble de théories et de techniques mises en œuvre en vue de réaliser des machines capables de simuler l'intelligence ». Elle correspond donc à un ensemble de concepts et de technologies plus qu'à une discipline autonome constituée.

Souvent classée dans le groupe des sciences cognitives, elle fait appel à la neurobiologie computationnelle (particulièrement aux réseaux neuronaux), à la logique mathématique (sous-discipline des mathématiques et de la philosophie) et à l'informatique. Elle recherche des méthodes de résolution de problèmes à forte complexité logique ou algorithmique. Par extension elle désigne, dans le langage courant, les dispositifs imitant ou remplaçant l'homme dans certaines mises en œuvre de ses fonctions cognitives.

Ses finalités et son développement suscitent, depuis toujours, de nombreuses interprétations, fantasmes ou inquiétudes s'exprimant tant dans les récits ou films de science-fiction que dans les essais philosophiques.

Dans notre vie, il y a plusieurs AI qui sont mis en œuvre on cite : Google Now, Alexa, Cortana, Siri.



Figure I-18 : Intelligence artificielle.

Le concept d'intelligence artificielle forte fait référence à une machine capable non seulement de produire un comportement intelligent, mais d'éprouver une impression d'une réelle conscience de

soi, de « vrais sentiments » (quoi qu'on puisse mettre derrière ces mots), et « une compréhension de ses propres raisonnements ».

L'intelligence artificielle forte a servi de moteur à la discipline, mais a également suscité de nombreux débats. En se fondant sur l'hypothèse, que tendent à confirmer les neurosciences et que des chercheurs n'hésitent pas à affirmer, que la conscience a un support biologique et donc matériel, les scientifiques ne voient généralement pas d'obstacle de principe à créer un jour une intelligence consciente sur un support matériel autre que biologique. Selon les tenants de l'IA forte, si à l'heure actuelle il n'y a pas d'ordinateurs ou de robots aussi intelligents que l'être humain, ce n'est pas un problème d'outil mais de conception. Il n'y aurait aucune limite fonctionnelle (un ordinateur est une machine de Turing universelle avec pour seules limites les limites de la calculabilité), il n'y aurait que des limites liées à l'aptitude humaine à concevoir les logiciels appropriés (programme, base de données...). Elle permet notamment de modéliser des idées abstraites [15].

I.8 l'Architecture générale et modes de fonctionnement des systèmes embarqués

Les systèmes embarqués utilisent généralement des microprocesseurs à basse consommation d'énergie ou des microcontrôleurs, dont la partie logicielle est en partie ou entièrement dans le matériel (on parle alors de firmware). Par définition, un système embarqué est un système complexe qui intègre du logiciel et du matériel conçus ensemble afin de fournir des fonctionnalités données. Il contient généralement un ou plusieurs microprocesseurs destinés à exécuter un ensemble de programmes définis lors de la conception et stockés dans des mémoires. D'autre part, c'est un système électronique et informatique autonome, souvent temps réel, spécialisé dans une tâche bien précise. Ses ressources sont limitées. Pour optimiser les performances et la fiabilité de ces systèmes, des circuits numériques programmables FPGA (Field Programmable Gate Array), des circuits dédiés à des applications spécifiques ASIC (Application Specific Circuits) ou des modules analogiques sont en plus utilisés.

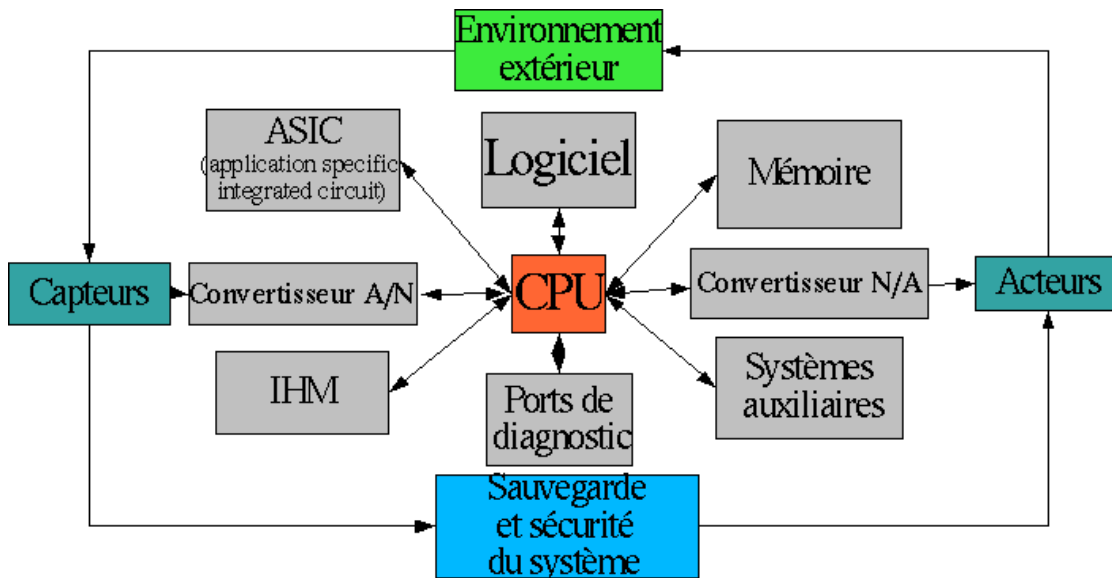


Figure I- 19 : Architecture générale des systèmes embarqués

I.9 Principales caractéristiques des systèmes embarqués temps réel

- Taille et complexité - Un système temps réel interagit avec un environnement extérieur souvent complexe et en évolution. Il doit respecter des échéances temporelles, garantir une fiabilité permanente. Il doit pouvoir interagir avec différents types d'éléments matériels.
- Implémentation efficace : restrictions dans l'utilisation des constructions du langage.
- Certification : garantir un fonctionnement conforme aux spécifications.
- Simulation, prototypage : vérification a priori de la conception.
- Plateformes d'essai : vérification a posteriori du bon fonctionnement.

I.9.1 Exemples d'applications des systèmes embarqués temps réel

Le temps réel touche des domaines très variés tels que le pilotage ou la navigation (de bateaux, d'avions, de trains, d'automobiles), le domaine des transactions bancaires, les transports (métro, aérospatiale, automobiles, etc.), les médias (décodeurs numériques), les services téléphoniques (terminal GSM, autocommutateur), la supervision médicale, le contrôle de processus industriels et systèmes de production industrielle (centrale nucléaire, chaîne de montage, usine chimique, etc.), et la robotique (ex. PathFinder : sonde lancée par la NASA en mars 1996, composée d'une station au sol et d'un robot mobile Séjourner).

I.10 Les logiciels libres et les systèmes embarqués

Les logiciels libres et en particulier GNU/Linux sont de plus en plus employés dans l'embarqué depuis qu'ils ont commencé à faire leur preuve il y a quelques années. Regardons Linux. Pourquoi retrouve-t-on Linux dans l'embarqué ? Tout d'abord pour ses qualités qu'on lui reconnaît maintenant dans l'environnement grand public :

- C'est un logiciel libre disponible gratuitement au niveau source.
- Il est stable et efficace.
- Il existe une aide rapide en cas de problèmes par la communauté Internet des développeurs Linux.
- Il y a un nombre de plus en plus important de logiciels libres disponibles.
- La connectivité IP chère aux systèmes embarqués est en standard.

I.11 Conclusion

Les systèmes embarqués sont à la base de tous les appareils. Donc, la conception de ces systèmes pose de nombreuses contraintes spécifiques : consommation d'énergie, encombrement, parcimonie des ressources matérielles disponibles, la gestion de ces ressources. Les logiciels libres ou les open source kits offrent une multitude de ressources pour aller plus loin dans ce domaine afin de développer d'autres solutions. Dans ce chapitre, on a exposé la notion des systèmes embarqués avec les futurs branches comme l'internet des objets, big data, cloud computing, intelligence artificielle. On a terminé avec l'architecture des systèmes embarqués. Le chapitre suivant sera consacré au concept d'arduino.

Chapitre II

Etude et concept d'arduino



Chapitre II : Etude et concept de l'arduino

II.1 Introduction

La conjonction de l'évolution des technologies de fabrication des circuits intégrés et de la nature du marché des systèmes électroniques fait que l'on est amené à concevoir des circuits de plus en plus complexes (plusieurs millions) de transistors en un temps de plus en plus court (quelques jours). Ce phénomène a entraîné une métamorphose du processus de conception allant de l'idée au produit.

Durant les dernières décennies, on est passé de la conception de circuits composés de quelques milliers de portes à des systèmes structurés et intégrés comme un réseau sur une même puce. Les puces modernes peuvent contenir plusieurs processeurs, de la mémoire et un réseau de communication complexe. Le principe de la conception reste le même ; il s'agit de générer une réalisation physique sous forme d'une puce en partant d'une spécification système. Par contre, les outils mis en oeuvre et l'organisation du travail durant le processus de conception ont beaucoup évolué. Partant d'une conception complètement manuelle où l'on dessinait les masques du circuit à réaliser sur du papier spécial, on est passé à une conception quasi automatique en partant d'une description du comportement du circuit sous forme d'un programme décrit dans un langage de haut niveau.

Dans le domaine de la conception de logiciel, de très nombreux travaux ont été et sont effectués sur la compilation de descriptions de haut niveau vers les langages machines (interprétés par les processeurs). Les habitudes de programmation pour les circuits programmables portes à croire que la compilation et l'édition de liens sont les seuls traitements à apporter au logiciel pour pouvoir le faire fonctionner sur une architecture cible. Le but de ce chapitre est de montrer l'utilité de ces circuits programmables libre et comment ils vont nous aider pour développer notre système.

II.2 arduino

II.2.1 Historique du projet arduino

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 durant cette période (avant 2003 - 2004). Les outils nécessaires à la création de projets d'interactivité étaient complexes et

onéreux (entre 80 et 100 euros). 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.

Jusqu'alors, les outils de prototypage étaient principalement dédiés à l'ingénierie, la robotique et aux domaines techniques. Ils sont puissants mais leurs processus de développement sont longs et ils sont difficiles à apprendre et à utiliser pour les artistes, les designers d'interactions et, plus généralement, pour les débutants. Leur préoccupation se concentra alors sur la réalisation d'un matériel moins cher et plus facile à utiliser. Ils souhaitaient créer un environnement proche de Processing, ce langage de programmation développé dès 2001 par Casey Reas et Ben Fry, deux anciens étudiants de John Maeda au MIT., lui-même initiateur du projet DBN .

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).

Comme pour Wiring, l'objectif était d'arriver à un dispositif simple à utiliser, dont les coûts seraient peu élevés, les codes et les plans « libres » (c'est-à-dire dont les sources sont ouvertes et peuvent être modifiées, améliorées, distribuées par les utilisateurs eux-mêmes) et, enfin, « multi-plates-formes » (indépendant du système d'exploitation utilisé).

Conçu par une équipe de professeurs et d'étudiants (David Mellis, Tom Igoe, Gianluca Martino, David Cuartielles, Massimo Banzai ainsi que Nicholas Zambetti), l'environnement arduino est particulièrement adapté à la production artistique ainsi qu'au développement de conceptions qui peuvent trouver leurs réalisations dans la production industrielle. Le nom arduino trouve son origine dans le nom du bar dans lequel l'équipe avait l'habitude de se retrouver. 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 » [16].

II.2.2 Que-est-ce qu'un arduino ?

arduino, et son synonyme Genuino, est une marque qui couvre des cartes matériellement libres sur lesquelles se trouve un microcontrôleur (d'architecture **Atmel AVR** comme l'AtMega328p, et d'architecture ARM comme le Cortex-M3 pour l'arduino Due). Les schémas de ces cartes sont

publiés en licence libre. Cependant, certains composants, comme le microcontrôleur par exemple, ne sont pas sous licence libre.

Le microcontrôleur peut être programmé pour analyser et produire des signaux électriques, de manière à effectuer des tâches très diverses comme la domotique (le contrôle des appareils domestiques - éclairage, chauffage...), le pilotage d'un robot, de l'informatique embarquée, etc.

C'est une plate-forme basée sur une interface entrée/sortie simple. Il était destiné à l'origine principalement mais pas exclusivement à la programmation multimédia interactive en vue de spectacle ou d'animations artistiques, ce qui explique en partie la descendance de son environnement de développement de Process, lui-même inspiré de l'environnement de programmation Wiring (l'un pensé pour la production d'applications impliquant des graphismes et l'autre pour pilotage de salles de spectacles).

arduino peut être utilisé pour construire des objets interactifs indépendants (prototypage rapide), ou bien peut être connecté à un ordinateur pour communiquer avec ses logiciels (ex. : Macromedia Flash, Processing, Max/MSP, Usine Hollyhock, Pure Data, SuperCollider). En 2011, les versions vendues sont pré-assemblées. Des informations sont fournies pour ceux qui souhaitent assembler ou construire une carte arduino eux-mêmes.

Le projet arduino a reçu un titre honorifique à l'Ars Electronica 2006, dans la catégorie Digital Communities[16].

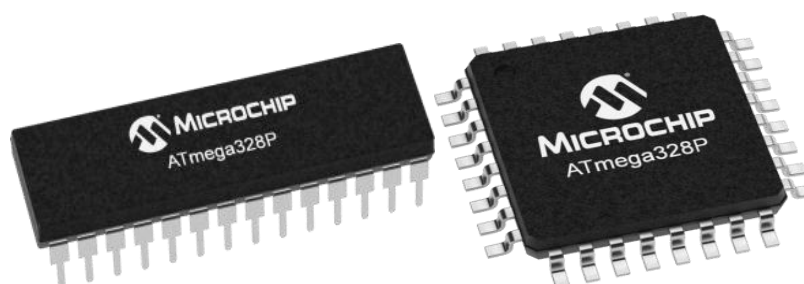


Figure II- 1 : Microconroleur ATmega328p 1.

La carte arduino est un circuit imprimé spécifiquement conçu pour héberger un microcontrôleur et donner accès à toutes ses entrées et sorties. Elle comprend aussi quelques autres composants électroniques qui permettent de faire fonctionner le microcontrôleur ou d'en étendre les fonctionnalités. Un microcontrôleur est un petit ordinateur confiné dans un unique circuit intégré (une puce). Il constitue un excellent moyen pour programmer et pour contrôler des équipements électroniques. Il existe une grande variété de telles cartes à microcontrôleur, certaines

des plus utilisées sont la platine Wiring, le PIC, le Basic Stamp et bien sûr arduino. Vous écrivez du code source dans l'environnement de développement arduino pour dire au microcontrôleur ce que vous souhaitez qu'il fasse. Par exemple, en écrivant une seule ligne de code (une instruction), vous pouvez faire clignoter une LED. Si vous connectez un bouton-poussoir, vous pouvez ajouter une autre ligne de code pour que la LED s'allume quand le bouton est enfoncé. Avec d'autres instructions, vous pouvez faire en sorte que la LED ne clignote que lorsque le bouton est enfoncé. Ainsi, vous pouvez facilement amener le système à se comporter d'une certaine manière, ce qui serait difficile à faire sans microcontrôleur. Comme un ordinateur de bureau, arduino peut assurer une multitude de fonctions, mais il ne sert pas à grand-chose tout seul. Il a besoin que quelque chose soit connecté sur au moins une de ses entrées et/ou de ses sorties pour être utile. Comme le clavier et la souris d'un ordinateur, ces canaux de communication permettent à l'arduino de sentir des objets du monde réel et d'agir dessus. Avant de continuer, il peut être utile de découvrir les grandes lignes de l'histoire d'arduino.

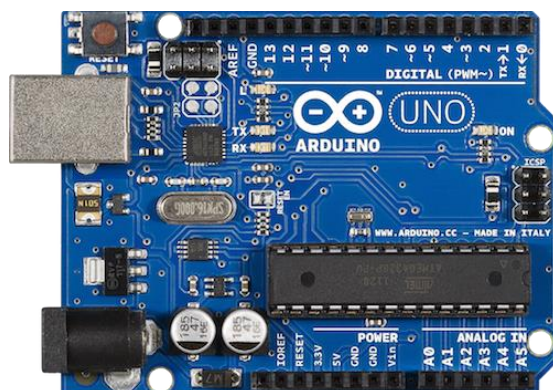


Figure II- 2: Carte arduino uno.

II.2.3 Architecture d'arduino

II.2.3.1 Qu'est-ce qu'un microcontrôleur ?

Un microcontrôleur est un circuit intégré (ou IC, Integrated Circuit), qui rassemble sur une puce plusieurs éléments complexes dans un espace réduit. Au temps des pionniers de l'électronique, on soudait un grand nombre de composants encombrants, tels que les transistors, les résistances ou les condensateurs, sur des cartes plus ou moins grandes. Aujourd'hui, tout peut loger dans un petit boîtier en plastique noir muni d'un certain nombre de broches. Ces dernières sont les connexions du circuit intégré au moyen desquelles s'effectue la communication. La figure II-1 montre un microcontrôleur ATmega328, qu'on trouve sur la carte arduino.

Avec ses dimensions réduites, il dispose pourtant d'une grande puissance de calcul. En fait, il suffit de le souder sur une carte et de le mettre sous tension pour pouvoir l'utiliser. Certes, il manque encore quelques composants (par exemple, des stabilisateurs de tension, des connexions pour la programmation, et d'autres sur lesquels nous reviendrons plus tard), mais il est cependant sous cette forme déjà (presque) prêt à l'emploi. A quoi sert-il ?

Maintenant, à quoi sert un microcontrôleur et ce qu'on peut faire avec. A cela, on peut répondre que les possibilités sont innombrables et dépendent uniquement de votre créativité.

Les microcontrôleurs jouent un rôle prépondérant dans les domaines suivants - cette liste est loin d'être exhaustive et sert surtout à se faire une idée des diverses possibilités d'utilisation.

- Fonctions de surveillance dans des environnements critiques, par exemple dans des cages thoraciques (température, humidité, fréquence cardiaque, pression sanguine du prématuré).
- Commande de chauffage : contrôle de la température externe ou interne pour le chauffage optimal de locaux.
- Stimulateurs cardiaques : surveillance de la fréquence cardiaque et, le cas échéant, stimulation du cœur.
- Appareils ménagers : par exemple, commande par programme enregistré dans des lave-linges ou lave-vaisselle modernes.
- Electronique de loisirs : lecteurs **MP3**, téléphones portables, appareils photo ...
- Robotique : par exemple, commande de robots industriels pour le montage de pièces automobiles.

Cette liste peut ainsi se poursuivre à l'infini, mais nous pouvons d'ores et déjà remarquer une chose les microcontrôleurs perçoivent des influences extérieures par le biais de capteurs, les traitent en interne à l'aide d'un programme, puis envoient des ordres de commande correspondants vers l'extérieur. Ils font donc preuve d'une certaine intelligence, qui dépend bien évidemment du programme mis en œuvre. Un microcontrôleur peut assurer des fonctions de mesure, de commande et de régulation. Regardons maintenant de plus près le fonctionnement d'une boucle de régulation. Elle se compose d'un processus en boucle fermée comportant une perturbation. Un capteur transmet cette dernière au microcontrôleur qui réagit alors en fonction de son programme.

Imaginez le scénario suivant. Vous vous trouvez au sein du système de contrôle de chauffage qui régule la température de votre local de travail. Le capteur dit au microcontrôleur : « Dis donc, il

faire plutôt chaud dans le local de travail ! » Le microcontrôleur réagit alors en régulant la température. Le chauffage apporte donc moins d'énergie sous forme de chaleur dans le local. Le capteur le remarque et dit au microcontrôleur : « La température est maintenant celle souhaitée, soit 20 °C. » De l'air froid provenant de l'extérieur rentre petit à petit. Le capteur donne l'alerte et dit au microcontrôleur : « Il commence à faire froid ici et mon bonhomme va tomber malade, il faut faire quelque chose ! » Le microcontrôleur augmente la température en conséquence. Vous voyez, c'est un jeu de ping-pong : ici, en l'occurrence, une boucle de régulation qui réagit à des influences perturbatrices extérieures liées à des variations de température [17].

II.2.3.2 Structure de microcontrôleur

Le microcontrôleur était déjà prêt à l'emploi. Mais où se trouve son programme et où stocke-t-il ses données ?

C'est un ordinateur complet sur un espace réduit au maximum, avec donc les éléments suivants :

- Unité Central (**CPU**)
- Mémoire de travail
- Mémoire de données
- Horloge interne
- Ports d'entrée et de sortie

Un microcontrôleur se divise grossièrement en trois parties :

- Unité centrale (**CPU**)
- Mémoires (**ROM** et **RAM**)
- Ports d'entrée et de sortie.

L'horloge interne, ou l'oscillateur qui permet de piloter l'unité centrale, a été laissée de côté pour le moment. Les éléments qui composent un microcontrôleur sont comparables aux périphériques d'un ordinateur. La différence réside dans le fait que les trois parties citées précédemment sont intégrées au microcontrôleur. Elles se trouvent toutes dans le même boîtier, ce qui est plus simple et plus compact.

Jetons maintenant un coup d'œil au schéma fonctionnel de notre microcontrôleur. Donc, pour cela et pour bien comprendre la méthode de fonctionnement de ces blocs dans ce schéma, on va vous expliquer ces blocs.

- **L'unité centrale (CPU)**

L'élément le plus important dans un microcontrôleur est l'unité centrale, appelée également **CPU** (Central Processing Unit). Sa fonction principale consiste à décoder et à exécuter des instructions. Elle peut adresser des mémoires, gérer des entrées ou sorties et réagir à des interruptions (interrupts). Une interruption (IRQ, ou Interrupt Request) est un signal qui demande au **CPU** d'interrompre un cycle de calcul en cours pour pouvoir réagir à un certain évènement.

- **Le bus de données**

Le bus de données sert à transporter les données d'un bloc à un autre. Par exemple, le **CPU** demande des données provenant de la mémoire, qui sont prises en charge par le bus et immédiatement mises à disposition pour traitement. Lorsque le résultat du calcul est disponible, il est à nouveau transféré sur le bus et transmis à un port de sortie qui, par exemple, pilote un moteur de robot pour atteindre un but précis. Cette structure de bus est une autoroute de données utilisable en commun par tous ceux qui sont desservis pour bien illustrer ce scénario. La figure II-4 nous explique ce passage.

La flèche rouge nous montre la direction de bus le prochain arrêt qui est la mémoire.

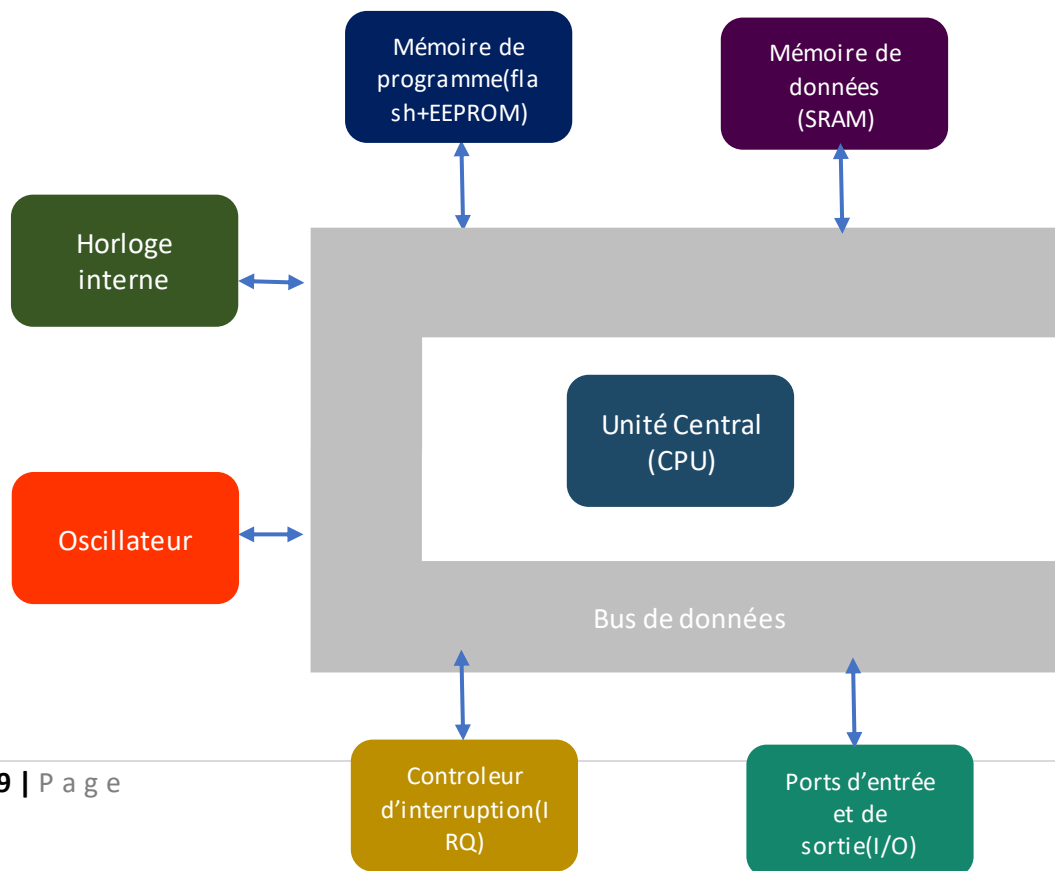


Figure II- 3 : Schéma fonctionnel d'un microcontrôleur.

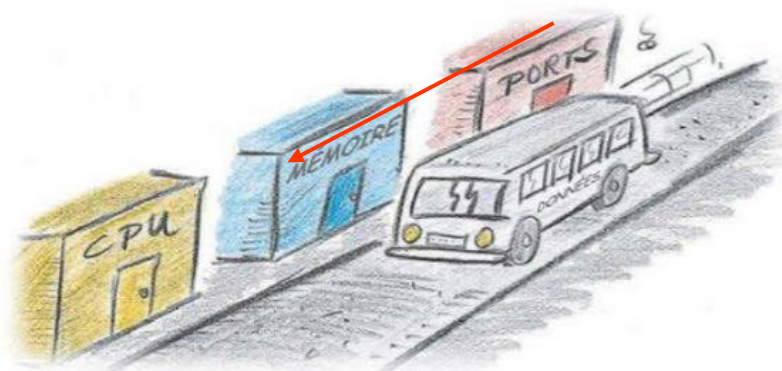


Figure II- 4 : Schéma explicatif de microcontrôleur.

- **Les zones de mémoire**

En principe, il existe deux types de mémoires d'un microcontrôleur :

- la mémoire de programme ;
- la mémoire de données.

La première accueille le programme que le **CPU** doit exploiter, alors que la seconde est utilisée pour gérer les résultats de calcul du moment.

- **Les ports d'entrées-sorties**

Les ports d'entrées-sorties sont les connexions qui relient le microcontrôleur au monde extérieur. Ils constituent une interface à laquelle la périphérie peut être connectée. On entend par périphérie tout ce qui peut être avantageusement raccordé à l'interface. Il peut s'agir, par exemple, des composants électroniques ou électriques suivants :

- LED (diode électroluminescente) ;
- bouton-poussoir ;
- commutateur ;
- LDR (Light Dépendant Résistor ou photorésistance) ;
- transistor ;
- résistance ;
- haut-parleur OU élément piézoélectrique ;

- etc.

Pour bien illustrer la notion des ports entrées sorties, imaginons ports d'une usine à deux garages, l'un pour l'arrivée des camions pour décharger ou charger (entrées) et l'autre pour la sortie[18].

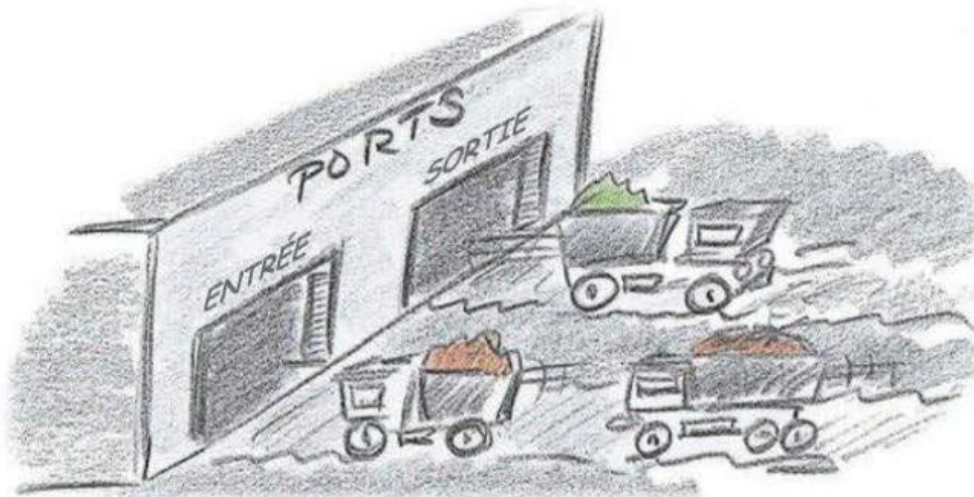


Figure II- 5 : Schéma explicatif de la notion des ports.

Donc à quoi sert à étudier le microcontrôleur et est-ce que l'arduino est un microcontrôleur ? Ce passage est consacré pour parler un peu sur le microcontrôleur et les principaux composants (unité centrale, mémoire, ports). Pour la réponse à la question, effectivement oui l'arduino c'est un microcontrôleur sans aucun doute, il possède bien tous les composants dont nous avons parlé et les réunit en son sein. Mais, il cohabite aussi avec d'autres composants sur une carte compacte, dont nous allons parler dans le prochain passage.

II.2.3.3 Carte arduino Mega2560

On va commencer par vous présenter la pièce maitresse de tous les montages de ce mémoire ; on va baser sur la carte Mega 2560 à cause de sa disponibilité en Algérie.

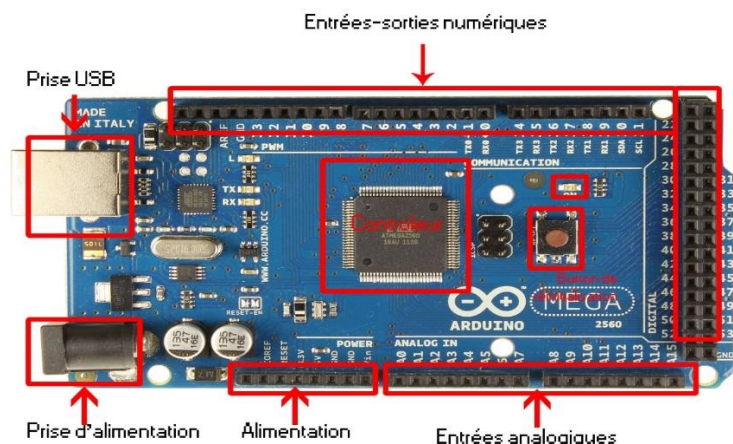


Figure II- 6 : Carte arduino Mega2560.

Au fil des années, plusieurs cartes arduino ont été développées. J'ai choisi de prendre l'exemple de la carte Mega. Les composants les plus divers y sont reconnaissables, lesquels feront l'objet d'une explication détaillée. Certains penseront certainement qu'on ne peut pas concevoir quelque chose de sérieux sur une surface aussi réduite. Mais grâce à la miniaturisation des composants ces dernières années, ce qui nécessitait auparavant cinq puces électroniques n'en requiert aujourd'hui plus qu'une.

Le plus gros élément qui saute directement aux yeux est le microcontrôleur proprement dit. Il est de type ATmega2560.

J'ai choisi la carte arduino Mega 2560 car, elle est disponible. Il existe d'autres familles comme : arduino Due ou arduino Yun, qu'on va les parler après ce passage.

Ces éléments sont les plus importants de la carte arduino mais, bien entendu, cela ne veut pas dire que les autres sont à négliger.

Voici les principales caractéristiques de la carte arduino :

- de 54 broches numériques d'entrées/sorties (dont 14 peuvent être utilisées en sorties PWM (largeur d'impulsion modulée)),
- de 16 entrées analogiques (qui peuvent également être utilisées en broches entrées/sorties numériques),
- de 4 UART (port série matériel),
- d'un quartz 16MHz,
- d'une connexion USB,
- d'un connecteur d'alimentation jack,

- d'un connecteur ICSP (programmation "in-circuit"),
- Et d'un bouton de réinitialisation (reset).

Comme vous pouvez le constater, un certain nombre d'entrées ou de sorties sont disponibles pour communiquer avec la carte arduino. Elles constituent l'interface avec le monde extérieur et permettent d'échanger des données avec le microcontrôleur, comme l'indique le schéma suivant :

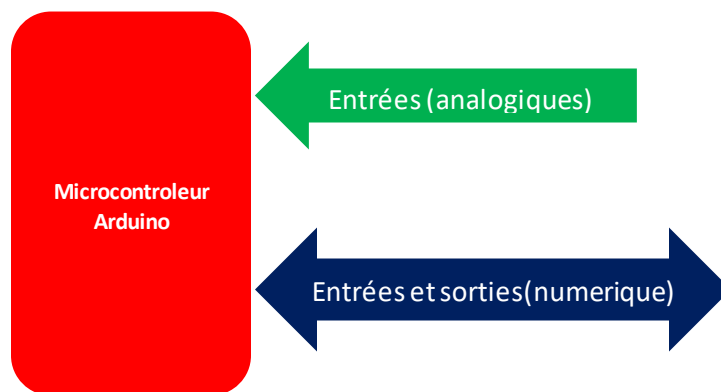


Figure II- 7 : Notion des entrées et sorties dans la carte arduino.

Le microcontrôleur arduino, représenté en rouge à gauche, peut communiquer avec nous via certaines interfaces. Certains ports servent d'entrées (flèche verte), et d'autres d'entrées et de sorties (flèche bleu). Un port est ici un chemin d'accès défini au microcontrôleur, pratiquement une porte vers l'intérieur qu'il est possible d'actionner[17].

II.2.4 Famille d'arduino

Les cartes arduino doivent satisfaire des exigences diverses et variées. Certains utilisateurs souhaiteront effectuer du prototypage et tester de nouveaux montages ou idées, la carte arduino Uno disposant d'assez d'entrées-sorties pour les projets d'envergure raisonnable. De par sa taille, la carte devrait aussi pouvoir être utilisée à l'avenir dans des projets plus ambitieux. D'autres auront besoin d'un grand nombre de ports afin de pouvoir raccorder de nombreux capteurs ou actionneurs. Une troisième catégorie d'utilisateurs ne cherchera qu'à attirer l'attention et à transmettre des signaux à leurs semblables à l'aide de diodes clignotantes. Ces exemples ne constituent qu'une partie des attentes que les maker auront vis-à-vis de leur carte arduino.

La forme, la taille ou les possibilités de connexion jouent un rôle décisif dans le choix de la carte adaptée. C'est pourquoi les développeurs d'arduino ont mis au point un vaste choix de cartes à microcontrôleur afin que chacun trouve le modèle qui réponde à ses besoins [19].

II.2.4.1 arduino Uno

La carte indispensable pour débuter demeure la carte arduino Uno dont le prix est très abordable. Elle s'est hissée au rang de standard de fait et elle convient parfaitement à tous ceux qui veulent faire leurs premières armes avec un microcontrôleur. C'est une valeur sûre.

- **Avantages :**

- carte arduino par excellence pour laquelle de nombreux exemples de montage sont disponibles sur Internet ;
- nombre suffisant de broches d'entrées ;
- sorties pour les projets élémentaires ;
- vaste choix de Shields ;
- bon marché

- **Inconvénients :**

- nombre insuffisant de broches d'entrées-sorties pour les projets ambitieux ;
- la mémoire disponible risque d'être un peu juste pour les gros projets ;
- ne peut pas être utilisée comme hôte USB pour simuler un clavier ou une souris.

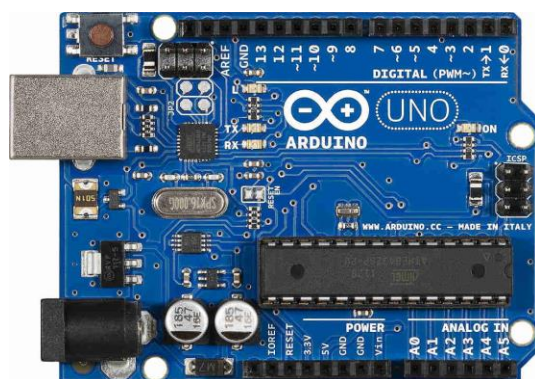


Figure II- 8 : Carte Arduino uno.

II.2.4.2 Arduino Esplora

L'Arduino Esplora est une carte sur laquelle est basée l'Arduino Leonardo. Si vous l'examinez de plus près, vous constaterez qu'elle est dotée d'un certain nombre de capteurs, qui ne sont pas présents sur la carte Arduino Uno ni sur d'autres cartes.

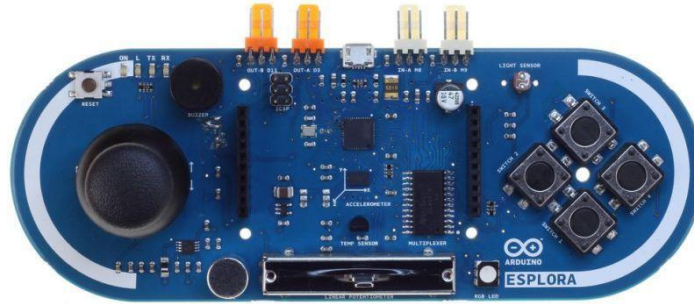


Figure II- 9 : Carte Arduino.

Les capteurs suivants sont préinstallés :

- un joystick à Yee un bouton-poussoir central ;
- un microphone ;
- un potentiomètre (linéaire) ;
- un capteur de température ;
- un accéléromètre 3 axes (x, y et z) ;
- quatre boutons poussoirs disposés en diamant ;
- un capteur de lumière.

La carte comporte aussi les sorties suivantes :

- un buzzer ;
- une LED RVB.

• **Avantages :**

- nombreux capteurs préinstallés ;
- connecteurs pour modules TinkerKit.

• **Inconvénients :**

- il n'est pas possible d'utiliser des Shields ;
- possibilités d'extension limitées ;
- relativement peu de circuits ou d'exemples de montages disponibles sur Internet (par rapport aux cartes Arduino Uno ou Arduino Mega 2560) ;
- prix assez élevé.

II.2.4.3 Boarduino V2.0

Si vous envisagez d'acheter une carte Boarduino, sachez que vous devez avoir un fer à souder sous la main, car elle est livrée en pièces détachées à assembler soi-même, à savoir :

- un microcontrôleur ;
- un support de circuit à 28 broches ;
- une carte ;
- des connecteurs.

J'avais annoncé que je ne présenterai que des modèles Arduino originaux, mais je vais tout de même faire une exception. Comme la plupart des clones, celui-ci est compatible avec Arduino. Il a été conçu pour être monté sur une plaque d'essais sans soudure. Vous trouverez des notes d'assemblage détaillées à la page indiquée par la référence [19].

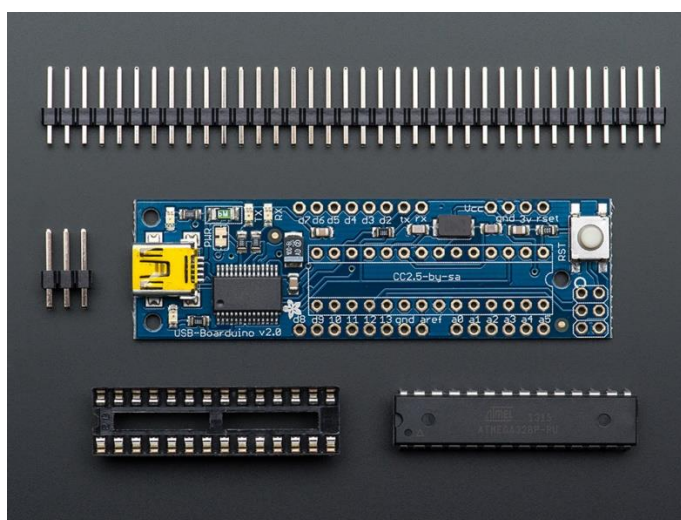


Figure II- 10 : la carte Boarduino avec ces composantes

- **Avantages :**
 - encombrement réduit ;
 - peut être enfichée directement sur la plaque d'essais.
- **Inconvénients :**
 - nécessite un peu de soudure avant d'être prête à l'emploi (ce n'est évidemment pas un problème pour les experts).

II.2.4.4 Arduino Nano

La carte Arduino Nano possède des connecteurs au dos qui permettent de l'enficher facilement sur une plaque d'essais, ce qui évite d'avoir recours à des cavaliers flexibles, comme pour l'Arduino Uno. Ne vous laissez pas abuser par les dimensions de cette minicarte dont les performances n'ont (presque) rien à envier à l'Arduino Uno.

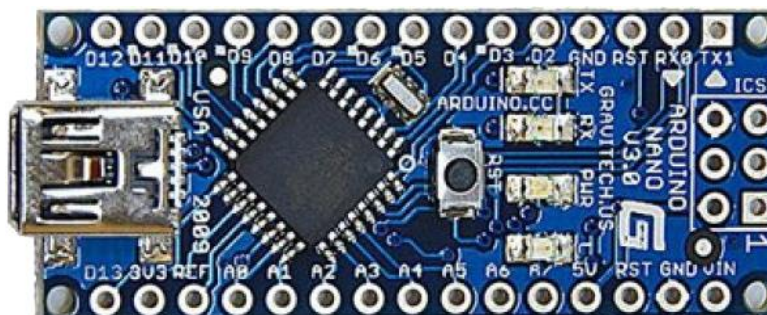


Figure II- 11 : Carte Arduino Nano.

• **Avantages :**

- encombrement réduit ;
- peut être enfichée directement sur la plaque d'essais.

• **Inconvénients :**

- il n'est pas possible d'utiliser des Shields.

II.2.4.5 Arduino LilyPad

La plateforme Arduino LilyPad est destinée aux plus créatifs d'entre nous qui veulent coudre des circuits électroniques sur leurs vêtements, par exemple. Les raccordements électriques ne se font pas par câbles, mais par des fils conducteurs. D'après le fabricant, le composant est lavable.

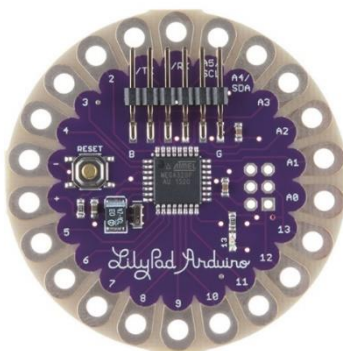


Figure II- 12 : Carte Arduino Lilypad.

• **Avantages :**

- fringable : carte extraplate conçue pour être intégrée à des vêtements.

• **Inconvénients :**

- la programmation ne se fait pas par USB, mais par un module adaptateur **FTDI** (5 V). Vous trouverez plus d'infos sur ce module en saisissant les critères de recherche suivante : LilyPad FTDI Basic Breakout - 5V DEV-10275

- Comme les raccordements électriques peuvent uniquement être soudés sur la plateforme Lily Pad, cette carte se prête davantage aux montages qui ne nécessitent pas d'ajustements fréquents.

II.3 Exemple des projets arduino

La photographie panoramique est une technique de photographie, utilisant un équipement ou un logiciel spécialisé, qui capture des images avec des champs de vision allongés. Il est parfois connu sous le nom de photographie grand format. Pour créer des photos panoramiques, on prend plusieurs photos à différents points de vue et les assemble ensuite. Il y a essentiellement deux façons d'obtenir ces photos, l'une consiste à utiliser une caméra et à tirer à partir d'une position différente, l'autre consiste à utiliser plusieurs caméras et à tirer en même temps. Ce dernier a un meilleur résultat de couture que l'ancien et nécessite moins d'expérience de la photographie, mais un équipement matériel plus complexe.

Avec la nature matérielle des blindages ArduCAM en particulier la carte ArduCAM-mini et Multi-Adapter pour Raspberry, vous pouvez connecter plusieurs caméras à une seule carte Arduino ou Raspberry Pi, même prendre des photos en même temps. On utilise 4 ArduCAM mini 5MP avec sa carte adaptateur et une carte Arduino UNO R3 pour créer panoramas ou visites virtuelles [20].

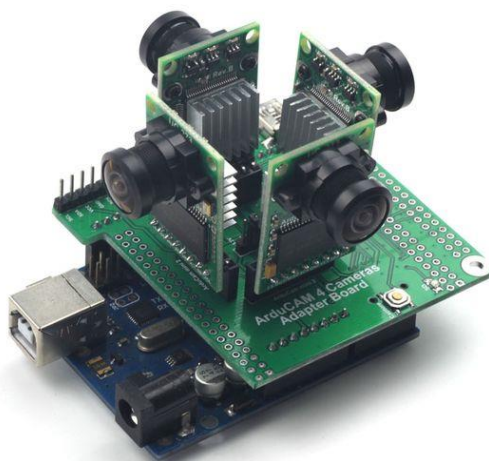


Figure II- 13 : Projet ArduCAM camera 360° degree.

Le but de ce projet est de capter l'image en 360° comme google. Cet appareil fait des captures des rues dans le monde afin de les montrer sous MAP en temps réel un autre projet. Pour Microsoft hologramme, il consiste à introduire les formes des objets en trois dimensions, pour cela, il a besoin de scanner quelques objets en plusieurs angles.

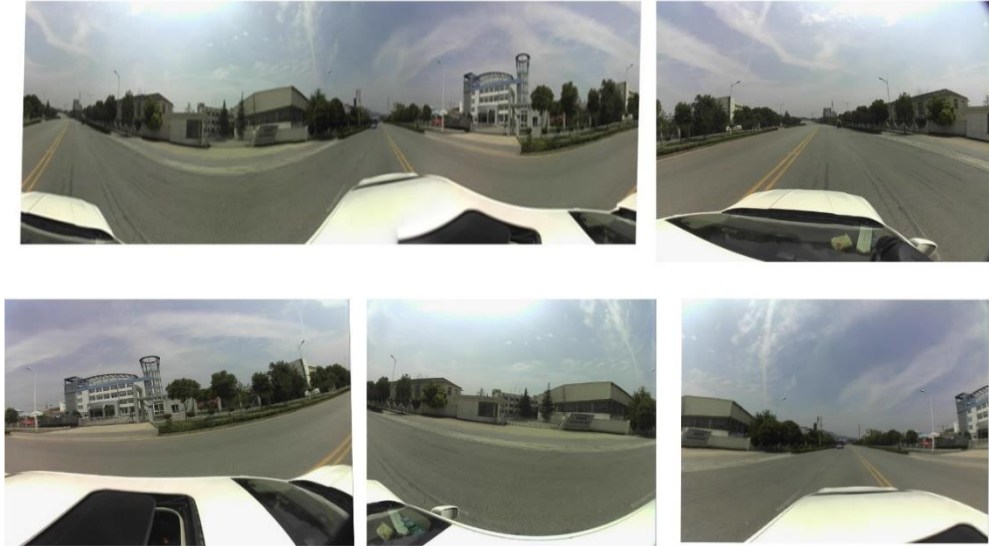


Figure II- 14 : Capture avec le projet ArduCam.

On cite un autre projet pour une station de météo pour connaître l'atmosphère d'extérieur avec la mesure de la température, la pression, altitude, et l'humidité à travers une application android programmée par java ou en kotlin. Cette application est utilisée à une distance de centaine de mètres car l'utilisation de la technologie WiFi, et pour alimenter ce kit, on utilise un panneau solaire avec une petite batterie. Ce projet nous aide à découvrir l'environnement à la sortie.

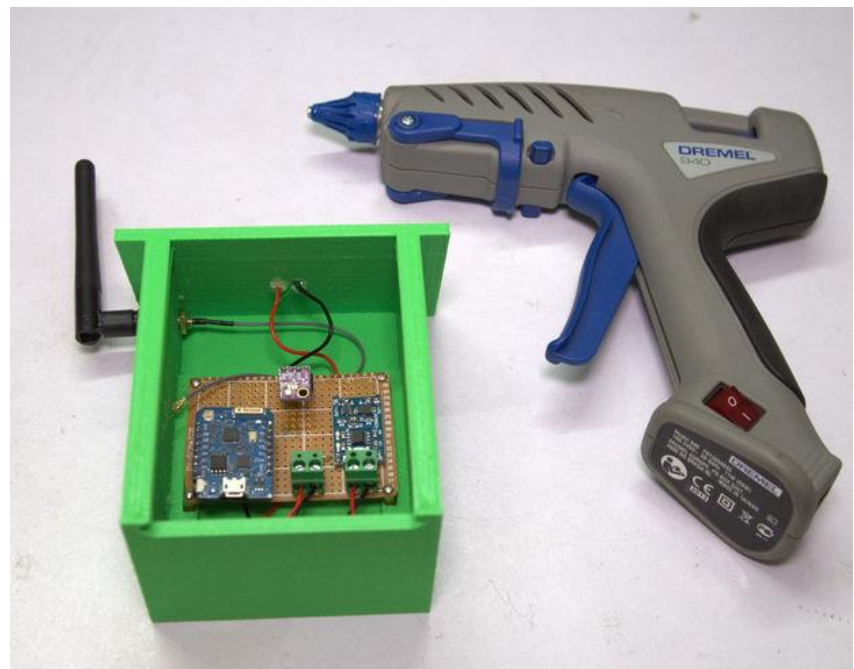


Figure II- 15 : Projet mesure de l'atmosphère.

Pour terminer, il y a les drones. Maintenant, on peut fabriquer un petit drone qui va nous servir à contrôler notre entourage ou aller dans des endroits difficilement accessibles par l'homme comme les grandes montagnes, ou pour suivre un phénomène comme par exemple au près d'un volcan. Dans ces types d'utilisations, on peut perdre un drone et non pas une vie humaine ou un scientifique.



Figure II- 16 : Drone avec arduino.

II.4 Environnement du développement Intégré (IDE)

Avant de commencer à travailler avec votre carte Arduino, vous devez installer sur votre ordinateur un logiciel indispensable. Il est semblable à ceux que vous avez sur votre ordinateur personnel, votre portable, votre tablette ou votre téléphone : il est requis pour utiliser votre matériel. Le logiciel Arduino est de type environnement du développement intégré (IDE). C'est un programme à interface graphique qui nous aide à créer notre code avec quelques langages de programmation comme le c, c++, python..., puis le charger dans la carte Arduino. Donc, on va commencer par vous montrer l'installation avec l'explication de l'environnement Arduino et un petit exemple consistant à allumer une LED par la carte Arduino.

II.4.1 Installation et setup

Tout d'abord, on va aller au site officiel du développement Arduino et on va télécharger le programme puis l'installer. On trouve des versions de ce logiciel pour Windows, Macintosh OS X et Linux. La figure si dessous nous explique comment installer le logiciel.

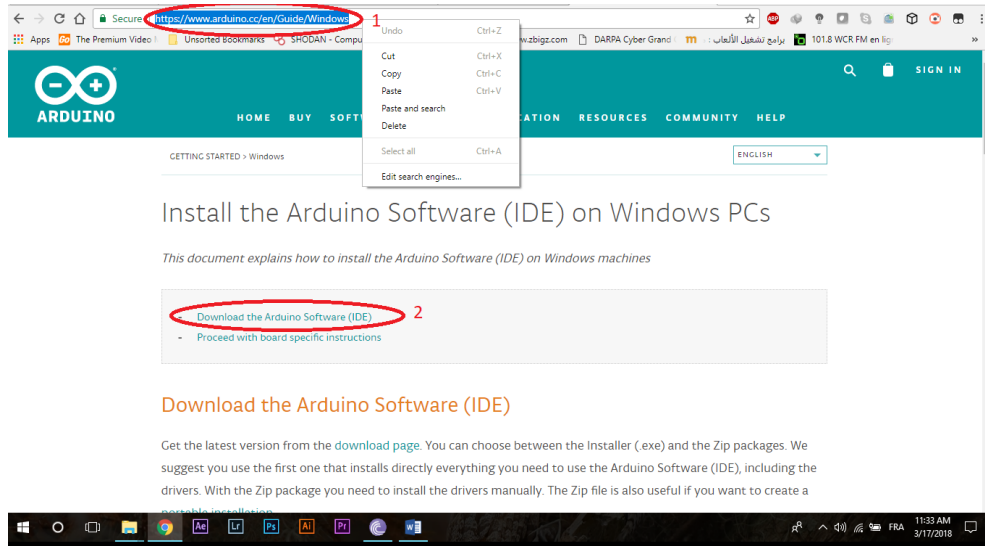


Figure II- 17 : Téléchargement d'IDE Arduino

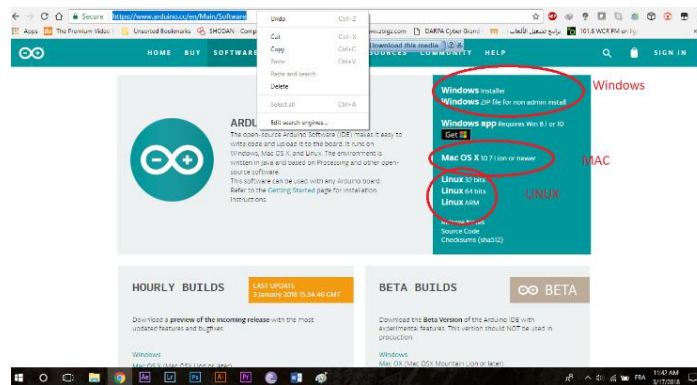


Figure II- 18 : Téléchargement d'IDE Arduino et l'enregistrement sur le pc .

Après le téléchargement, on va le mettre dans le bureau ou dans un dossier préféré pour commencer l'installation.



Figure II- 19 : Début d'installation avec le programme dans le bureau.

Double cliquer sur le programme, puis accepter les termes d'utilisation et les conditions de copyright de ce programme.

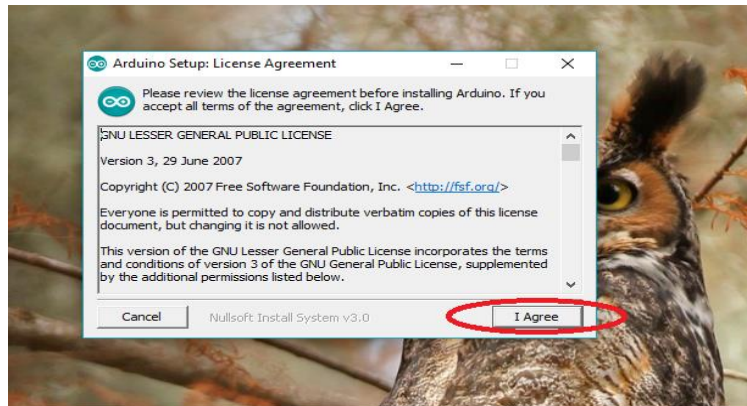


Figure II- 20 : Installation du programme partie1

Après, on clique sur suivant (ou Next) et on laisse l'installation par défaut pour installer tous les composants nécessaires.

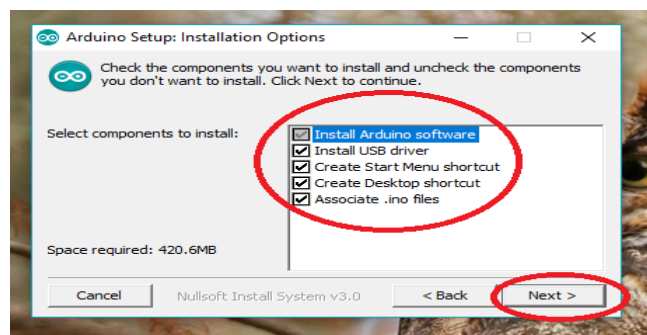


Figure II- 21 : Installation du programme partie 2.

Puis, on choisit le chemin d'installation dans notre disque dur. Par défaut, il est dans le c:\programme (X86)\arduino ; on peut changer le chemin comme on peut le laisser tel qu'il est, puis, on clique sur le Bouton Install.

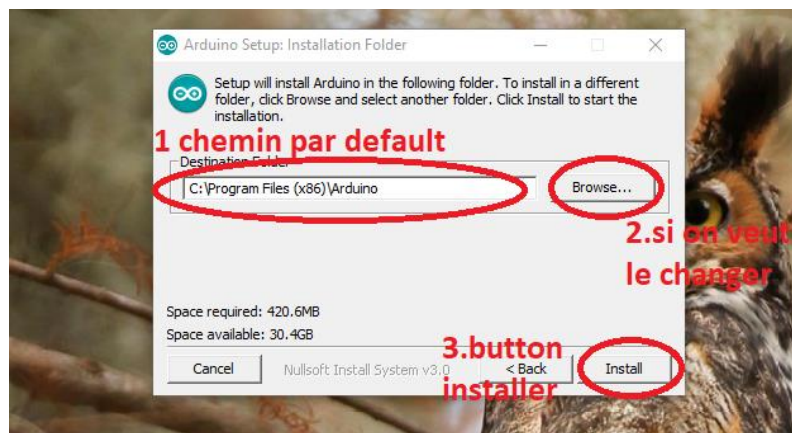


Figure II- 22 : Partie finale de l'installation.

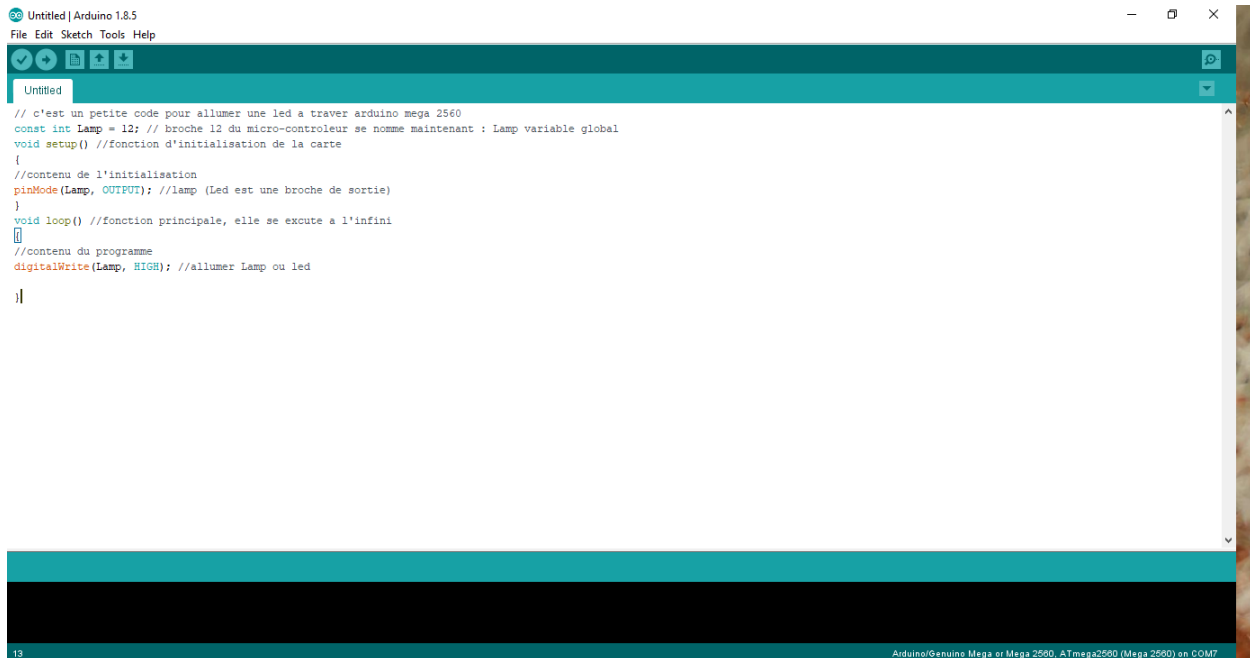


Figure II- 23 : Le programme Arduino après l'ouverture.

File Edit Sketch Tools Help

Figure II- 24 : Barre des menus.

Dans la barre des menus, on trouve le menu « File » contenant des options telles que créer un nouveau projet, le sauvegarder, ouvrir d'autres exemples ou un projet récent, imprimer un fichier et l'option pour quitter tout le programme.

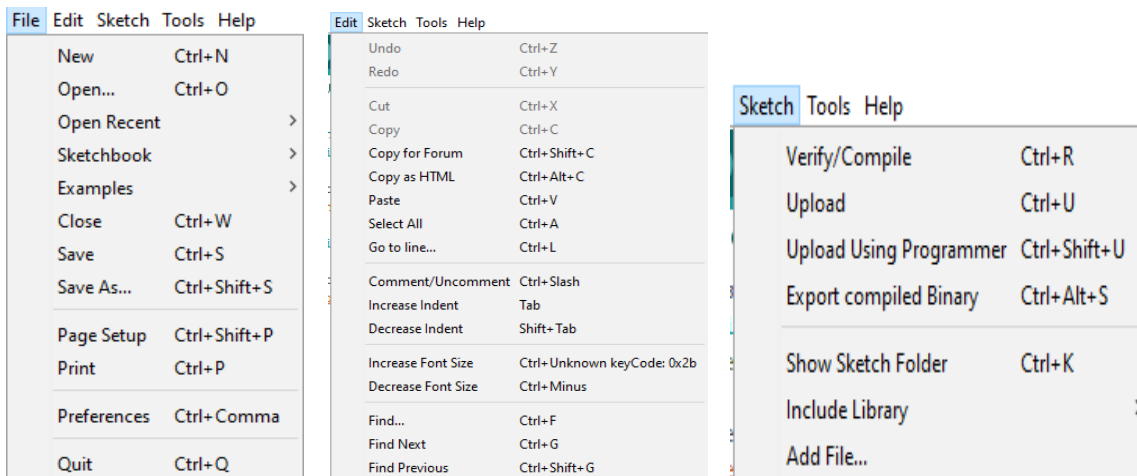


Figure II- 25 : Barre des menus avec ces sous-menus partie1.

Dans le menu « Edit », on trouve un sous menus qui contient les raccourcis (shortcut) du clavier pour sauvegarder ou lancer le programme sur la carte arduino. Dans la section sketch, on trouve aussi un autre sous menu qui contient la partie importante dans l'environnement commençons par

compile qui sert à compiler le code dans le programme pour voir s'il y a des problèmes puis on va le charger dans la carte utilisant Upload (téléverser), l'outil Upload using programmer fait la compilation et le chargement en même temps ; aussi on peut voir les dossiers où se trouve notre projet. On trouve aussi des bibliothèques déjà prêtes à l'utilisation directement sans les reprogrammer.

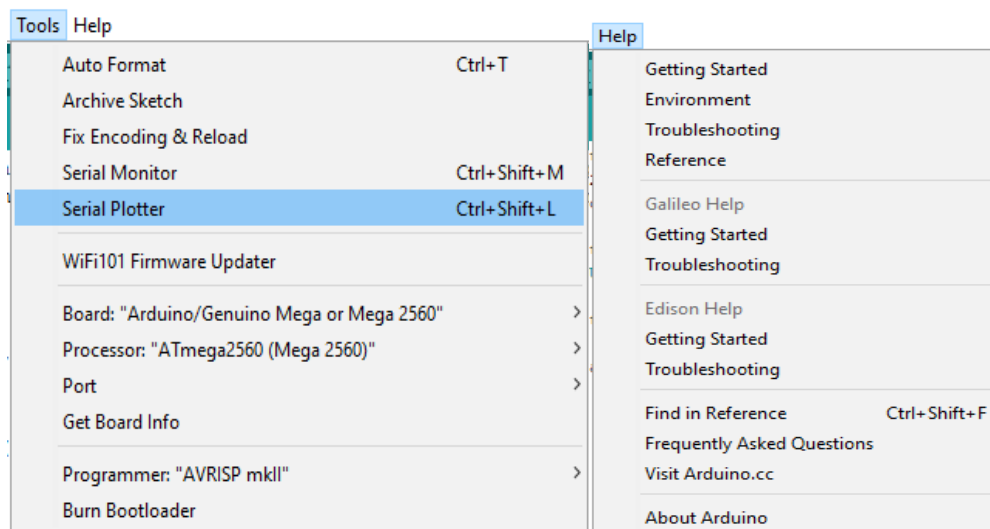


Figure II- 26 : Barre des menus avec ces sous-menus partie 2.

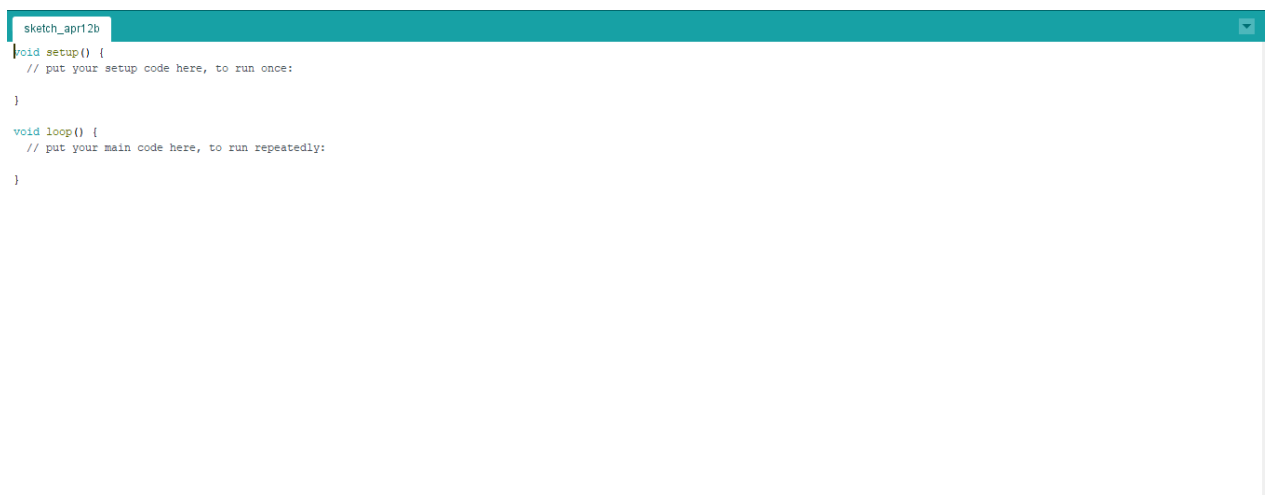
Dans la section « Tools », il y a la version ou le type d'Arduino que vous utilisez, le port de communication. Dans la dernière section help, on trouve la version de l'environnement utilisé, on peut contacter les constructeurs s'il y a un programme, il y a aussi le mode d'emploi pour bien comprendre le fonctionnement du programme.

Après la barre des menus, on trouve la barre des icônes qui contient l'icône Vérifier servant à vérifier la compilation, l'icône Téléverser pour charger le programme dans la carte Arduino, l'icône nouveau pour un Nouveau projet, l'icône Ouvrir pour ouvrir un projet déjà existant et l'icône Enregistrer pour sauvegarder ou enregistrer le projet.



Figure II- 27 : Barre des icônes.

Si on descend vers le bas, on trouve l'éditeur de texte dont on peut écrire notre code en c ou c++.

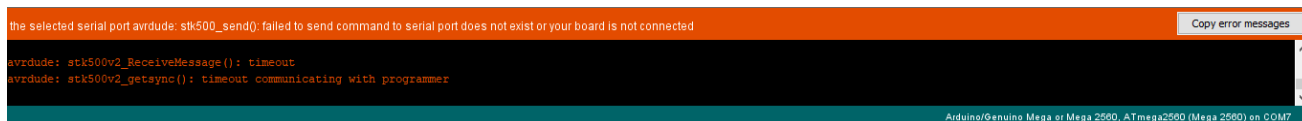


```
sketch_apr12b
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

Figure II- 28 : Editeur de texte.

S'il y a des erreurs dans le programme, dans la carte, ou pendant la transmission on peut les trouver dans la barre en bas de la fenêtre, elle signale où se trouve l'erreur, si au niveau de code ou dans le circuit.



```
the selected serial port avrdude: stk500_send() failed to send command to serial port does not exist or your board is not connected
Copy error messages
avrdude: stk500v2_ReceiveMessage(): timeout
avrdude: stk500v2_getsync(): timeout communicating with programmer
Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM7
```

Figure II- 29 : Barre des erreurs.

II.4.2 Structure d'un programme arduino

Par défaut, pendant le lancement d'un nouveau projet, on trouve la fonction `void setup ()` et la fonction `void loop()`.

```
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

Figure II- 30 : Structure d'un programme dans l'IDE.

La fonction **setup()** est appelée au démarrage du programme. Cette fonction est utilisée pour initialiser les variables, le sens des broches, les bibliothèques utilisées. La fonction `setup` n'est exécutée qu'une seule fois, après chaque mise sous tension ou reset (réinitialisation) de la carte Arduino.

La fonction **loop ()** (boucle en anglais) fait exactement ce que son nom suggère et s'exécute en boucle sans fin, permettant à votre programme de s'exécuter et de répondre. Utiliser cette fonction pour contrôler activement la carte Arduino.

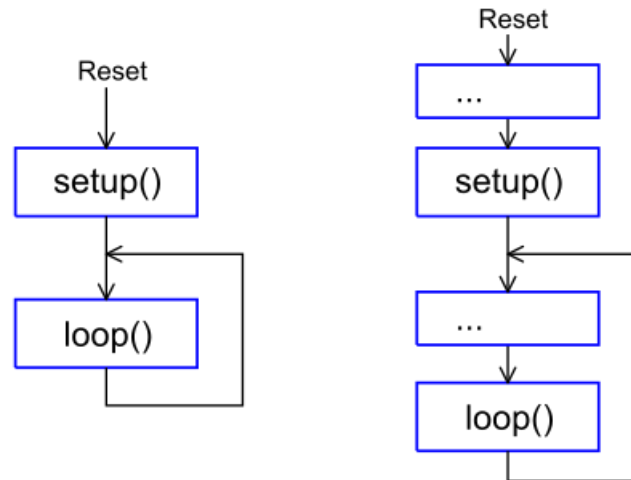


Figure II- 31 : Organigramme des fonction setup et Loop.

II.4.3 Démonstration réelle

Pour un début, on peut commencer par vous montrer une petite application consistant à allumer une LED à travers la carte Arduino. Pour cela, on va ouvrir l'IDE arduino, puis on crée un nouveau projet en allant dans file -> new. Avec arduino, nous devons utiliser un code minimal lorsque l'on crée un programme que nous allons créer en deux grosses parties : contenu de l'initialisation et contenu du programme. on fait le montage avec logiciel fritzing qui est un programme de simulation des circuits avec leur code.

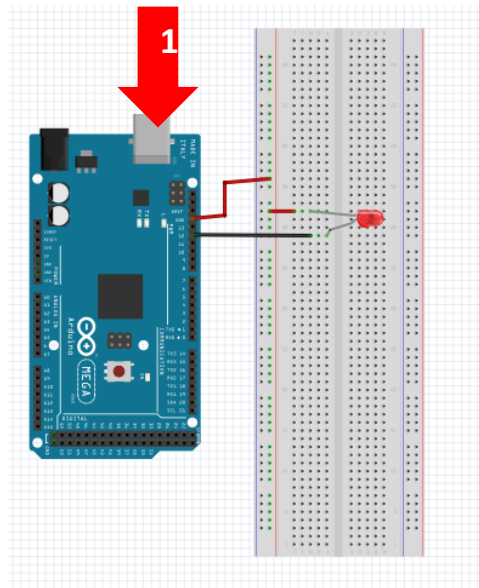


Figure II- 32 : Schéma explicatif de la simulation par fritzing.

On place le fil noir à l'entrée de l'alimentation ou le pin 12 pour créer notre circuit puis on ajoute la led ; après, on place le fil rouge à la sortie d'arduino ou le **GND**. On relie le port USB avec notre PC après avoir modifié le type de notre carte arduino de uno à Mega 2560 comme le montre la figure ci-dessous :

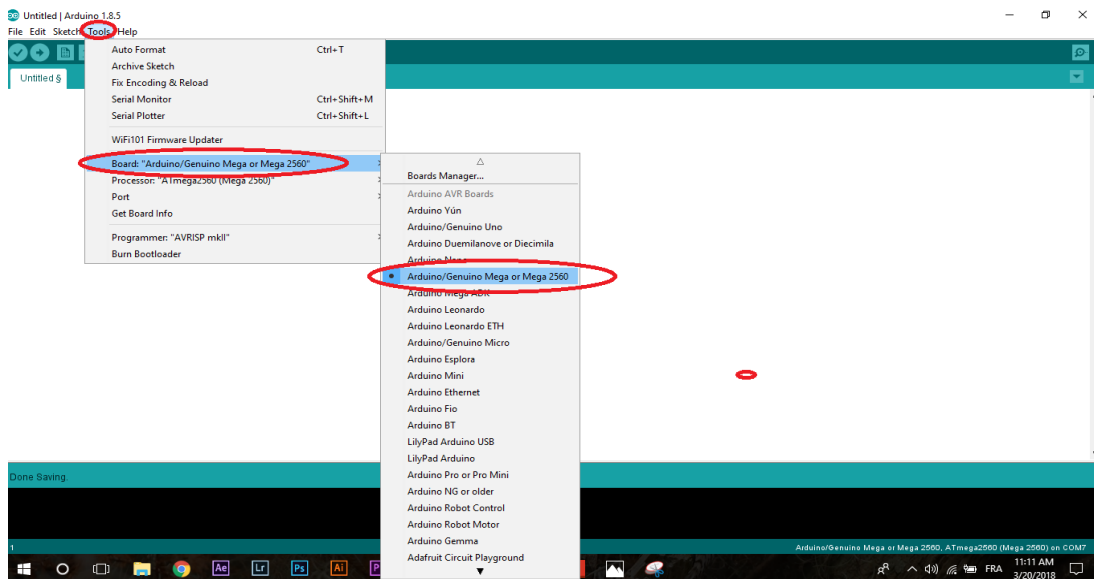


Figure II- 33 : Choix de la carte dans IDE.

Ensuite, on tape le code indiqué sur la figure suivante :


```

Untitled | Arduino 1.8.5
File Edit Sketch Tools Help

Untitled$
// c'est un petite code pour allumer une led a travers arduino mega 2560
const int Lamp = 12; // broche 12 du micro-contrôleur se nomme maintenant : Lamp variable global
void setup() //fonction d'initialisation de la carte
{
  //contenu de l'initialisation
  pinMode(Lamp, OUTPUT); //lamp (Led est une broche de sortie)
}
void loop() //fonction principale, elle se excute a l'infini
{
  //contenu du programme
  digitalWrite(Lamp, HIGH); //allumer Lamp ou led
}
    
```

Done Saving

19 Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM7 11:15 AM 3/20/2018

Figure II- 34 : Le code explicatif d'allumage de la LED.

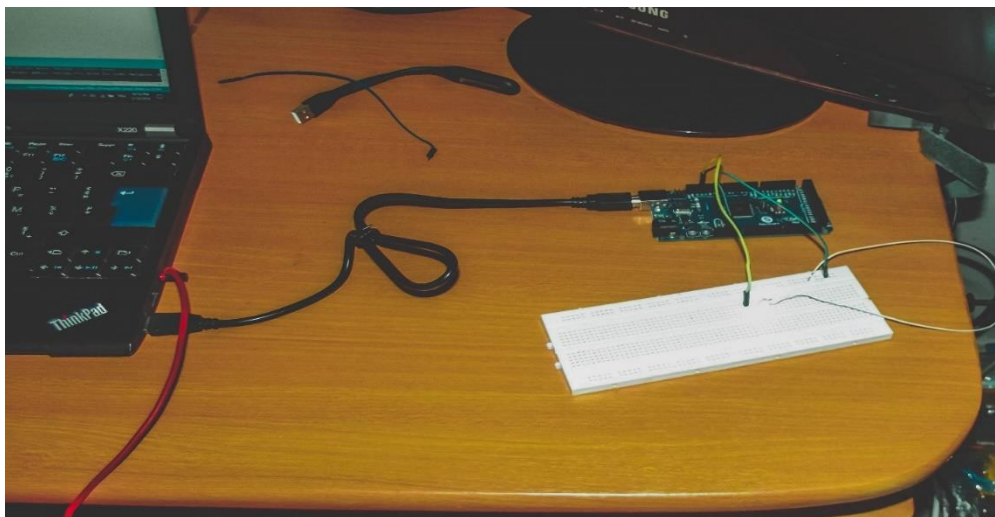


Figure II- 35 : Démonstration réelle.

La première ligne crée une variable de type int initialisée avec la valeur 12 (c'est la pin12 sur la carte arduino si on choisit la pin13 on l'initialise avec la valeur 13). void setup () est une fonction qui ne renvoie aucune valeur et s'exécute une seule fois, elle sert à initialiser la broche numérique comme sortie avec pinMode(lamp, output). La fonction void loop() est une fonction principale qui s'exécute en boucle infinie, et qui contient digitalWrite (lamp, Heigh) servant allumer la lampe ou la **LED**. Voici le montage dans la figure ci-dessous :

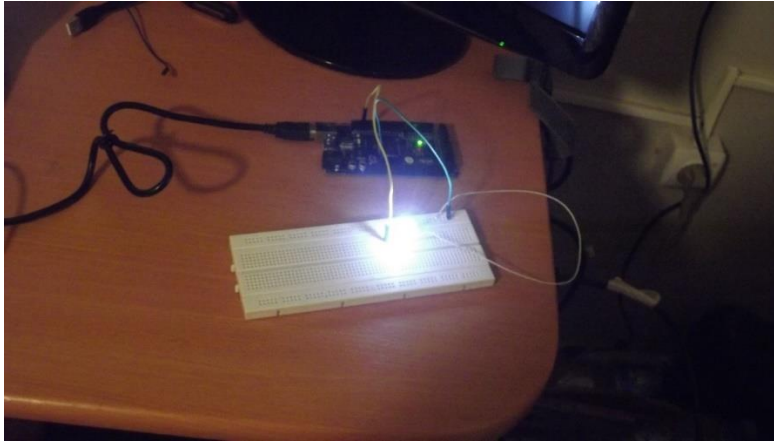


Figure II- 36 : Allumage de la LED.

II.5 Conclusion

Après la révolution de la technologie dans le domaine de l'IoT ces dernières années, il y a une augmentation très rapide car les gens peuvent devenir très riches par une simple application sans connaissance préalable ni de la programmation ni de l'électronique.

Dans ce chapitre, on a présenté la carte arduino de point de vu matériel, puis, on a montré l'environnement de développement intégré (**IDE**) en commençant par son installation suivie d'un exemple d'application qui allume une LED. Le chapitre suivant sera consacré sur la pratique et simulation .

Chapitre III

Contrôle à distance de l'énergie électrique



III.1 Introduction

L'apparition de l'électricité dans le XVIII^e siècle guide les scientifiques vers une révolution dans la technologie de tout appareil quel soit de télécommunications ou de l'électroménager. Aujourd'hui, les scientifiques pensent à l'utiliser au lieu du pétrole car c'est une énergie pure non polluante.

Le plus grand développement de l'énergie électrique se produit avec l'exploitation des centrales hydrauliques et le perfectionnement des turbines. L'énergie hydroélectrique se substitua peu à peu à celle du charbon ou du gaz, bien que les centrales thermiques continuent à fonctionner, car elles représentaient toujours un bon complément pour assurer un approvisionnement continu. En effet, les ressources hydroélectriques étaient dépendantes des pluies [21].

L'énergie électrique désigne toute énergie transférée ou stockée grâce à l'électricité. Cette énergie est transférée d'un système à un autre par un mouvement de charges.

Les systèmes susceptibles de fournir de l'énergie par transfert électrique sont les alternateurs ou des systèmes chimiques comme les piles notamment. Les systèmes susceptibles de transformer l'énergie issue de l'électricité sont par exemple les résistances électriques qui la transforment en chaleur, les moteurs qui la transfèrent en un travail mécanique, les lampes qui la transforment en rayonnement et en chaleur, et d'autres systèmes électrotechnique ou électronique. Le transport d'énergie électrique se fait au moyen d'un conducteur électrique, par exemple un métal ou une solution ionique.

L'énergie électrique ne peut pas être stockée en grande quantité (seule de petites quantités de charges électriques peuvent être stockées sous forme d'énergie dite électrostatique par exemple dans les condensateurs). L'expression « énergie électrique » est impropre en physique mais est une commodité de langage permettant d'indiquer que l'électricité nécessite et transporte de l'énergie. Pour stocker de l'énergie fournie par transfert électrique, il faut utiliser un convertisseur capable de stocker l'énergie reçue, par exemple en énergie chimique, dans les accumulateurs ou la convertir en énergie mécanique ou en énergie potentielle (par exemple dans une STEP ou un Barrage hydro-électrique) [22].

III.2 Etude sur l'électricité

L'électricité est une forme d'énergie. C'est un phénomène énergétique associé à la mobilité ou au repos de particules chargées positivement ou négativement.

Des phénomènes naturels, tels que la foudre, étaient déjà observés dès l'Antiquité, mais pendant très longtemps l'électricité a terrifié les hommes qui voyaient en elle une manifestation de la colère divine ou d'un pouvoir surnaturel.

Ce n'est qu'à partir de la fin du 16^e siècle qu'elle a commencé à être étudiée par les scientifiques pour en comprendre ses mécanismes et établir des lois. Leurs travaux successifs ont permis de créer artificiellement de l'électricité en transformant diverses sources d'énergies.

Aujourd'hui, cette électricité est produite par des centrales électriques, transportée et distribuée aux consommateurs. Comme le feu du temps des hommes préhistoriques, l'électricité a changé la vie de l'humanité. Elle est devenue indispensable à tout ce qui fait notre vie quotidienne : se nourrir, se chauffer, s'éclairer, se laver, soigner, communiquer, se déplacer, fabriquer [23].

III.2.1 Le flux d'électrons

Les électrons sont d'infimes particules élémentaires qui circulent à vitesse élevée dans un conducteur. Voici quelques-unes de ces propriétés caractéristiques :

- charge négative ($-1,602176 \times 10^{-19}$ C).
- pratiquement sans masse ($9,109382 \times 10^{-31}$ kg).
- stable (durée de vie supérieure à 1024 ans). Un conducteur traversé par un courant pour que

vous puissiez visualiser ces minuscules particules (voir figure III-1). Elles vont toutes dans le même sens et sont responsables de la circulation du courant.

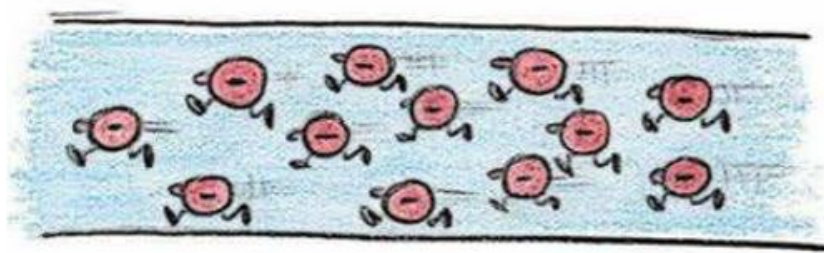


Figure III- 1 : Le flux de courant.

La charge négative de l'électron, la valeur $-1,602176 \times 10^{-19} \text{ C}$ était très petite. L'unité de mesure est le coulomb (C). La charge Q circulant en un temps donné à travers un conducteur ayant une certaine section est exprimée à l'aide de la relation suivante :

$$Q = I \cdot t$$

Il s'agit du produit entre l'intensité du courant I (en ampères) et le temps t (en secondes).

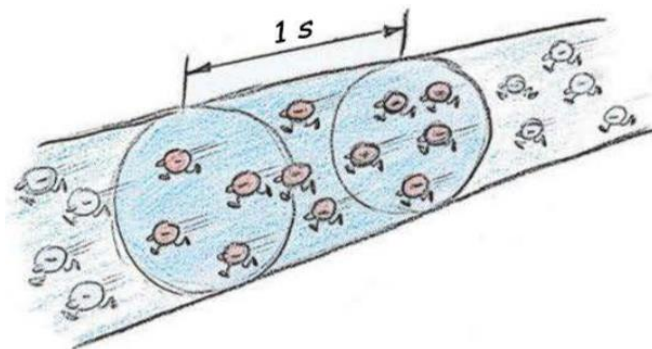


Figure III- 2 : Parcours à travers un conducteur en cuivre sur une durée d'un second

Sur cette vue de la migration des électrons à travers un conducteur en cuivre, on a marqué un tronçon que les électrons parcourent en 1 seconde. On peut retenir qu'une charge d'un coulomb a été transportée lorsqu'il y a eu le passage d'un courant d'un ampère pendant 1 seconde.

III.2.2 Le courant

Comme vous avez pu le voir dans la dernière formule, la charge et le courant sont reliés entre eux. On peut assimiler le courant à la circulation d'une charge électrique. Plus la charge circulant par unité de temps est importante, plus le courant électrique I est élevé.

La figure III-3 représente un faible flux d'électrons. Seuls quelques porteurs de charge circulent par unité de temps ; on aura donc un faible courant.

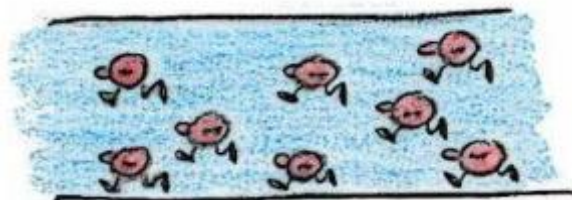


Figure III- 3 : Faible flux d'électrons.

A l'inverse, la figure III-4 représente beaucoup de porteurs de charge circulant par unité de temps, qui donnent naissance à un fort courant dans le conducteur.

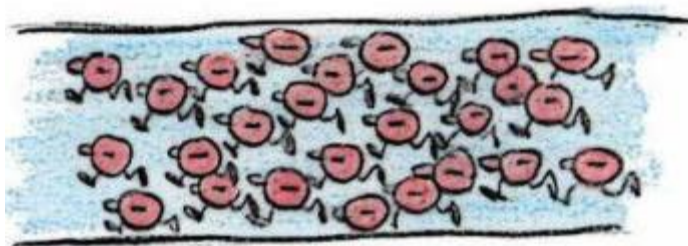


Figure III- 4 : Fort courant dans le conducteur.

L'intensité de courant I est mesurée en ampères (A) : l'ampère constitue une très forte intensité pour des microcontrôleurs. La charge maximale d'une sortie numérique de votre carte Arduino est de l'ordre de 40 mA (ou 40 mm ampères). Un milliampère représente un millième d'ampère ($1000\text{mA}=1\text{ A}$).

III.2.3 La tension

Si l'on jette un œil aux dessins précédents schématisant des électrons parcourant un conducteur, il y a quelque chose que nous ne prenons pas en considération : pourquoi se déplacent-ils ? Notre monde est ainsi fait : chaque action a sa raison ou sa finalité correspondante ; nos gestes sont toujours commandés ou motivés par quelque chose. Il en va de même pour les électrons : ils vont tous dans le même sens et vers le même but. Une force motrice doit par conséquent agir. La comparaison de ce phénomène avec de l'eau qui s'écoule est très parlante.

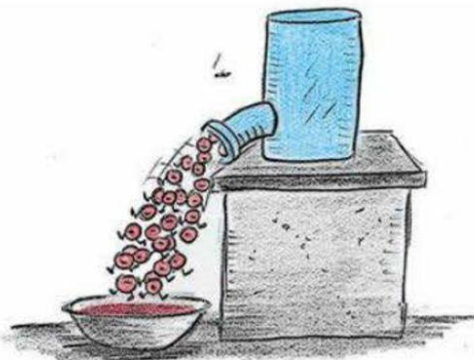


Figure III- 5 : Les électrons se déplacent en raison d'une différence de potentiel.

Ici, on parle de différence de charge et non de différence de potentiel. Les charges électriques tendent toujours à compenser les différences de charge.

Prenons, par exemple, une pile chargée. Elle a deux bornes (ou pôles) entre lesquelles il existe une différence de charge. L'une de ces bornes présente un excès de charge et l'autre un manque de charge. Faute de liaison électrique entre les deux pôles, aucune charge ne peut se déplacer pour

rétablir l'équilibre, si bien qu'aucun courant ne circule.

La tension U se mesure en volts (V) et sert à estimer la différence de potentiel entre deux points.

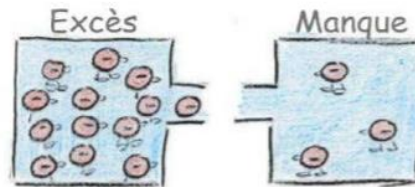


Figure III- 6 : La différence de charge ne peut être compensée, faute de liaison.

L'absence de liaison entre les deux potentiels empêche toute égalisation des charges et aucun courant ne circule.

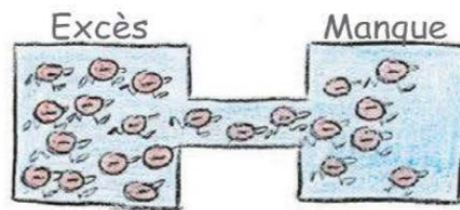


Figure III- 7 : Une compensation de la différence de charge se produit

C'est seulement quand une liaison est établie que les porteurs de charge peuvent se déplacer et qu'un courant peut circuler.

Le courant circule tant que la charge n'est pas équilibrée, autrement dit jusqu'à ce que les deux pôles aient autant de porteurs de charge l'un que l'autre. Si tous les électrons étaient passés du côté droit, il y aurait eu de nouveau un déséquilibre, et l'opération se serait répétée dans l'autre sens. Par ailleurs, une séparation de charge ne peut être rétablie que par un apport d'énergie après une égalisation de charge. Or, ce n'est pas le cas, et c'est donc aussi pour cette raison qu'une pile normale est vide après une égalisation de charge.

En électronique, il y a deux types de courant l'un est symbolisé par les lettres DC (Direct Current) ou par CC (courant continu). La figure III-8 montre l'évolution d'une tension continue au cours du temps. Et la figure III-9 montre l'évolution d'une tension alternative au cours du temps. Le courant alternatif est, quant à lui, noté AC (Alternating Current) ou CA (courant alternatif).



Figure III- 8 : Evolution d'une tension continue au cours du temps

L'axe horizontal des abscisses (X) représente le temps t et l'axe vertical des ordonnées (Y) indique la tension U . On voit que la valeur de tension ne varie pas au cours du temps. Analysons maintenant l'évolution de la tension alternative, représentée ici par une sinusoïde.

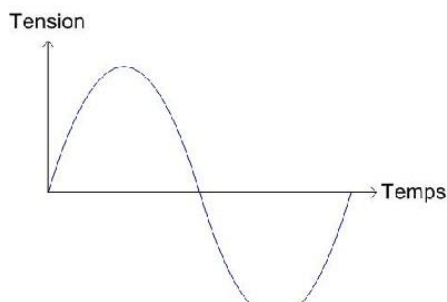


Figure III- 9 : Evolution d'une tension alternative au cours du temps

La valeur de la tension varie en permanence et oscille entre deux valeurs limites, l'une positive et l'autre négative. Notez que la lettre U indique qu'on a affaire à une tension. Sachez aussi qu'il existe, dans la plupart des cas, une relation de proportionnalité entre la tension et le courant.

III.2.4 Résistance

Les électrons circulant dans un conducteur et peuvent avoir plus ou moins de mal à arriver au bout. Sur leur chemin, ils rencontrent en effet des résistances très différentes les unes des autres.

On peut établir une classification des matériaux en fonction de leur conductibilité :

- isolants (très haute résistance), par exemple : la céramique ;
- mauvais conducteurs (haute résistance), par exemple : le verre ;
- bons conducteurs (faible résistance), par exemple : le cuivre ;
- très bons conducteurs (supraconductivité a de très basses températures OU la résistance électrique tend vers 0).
- semi-conducteurs (la résistance peut être commandée), par exemple : silicium ou germanium.

Il existe ainsi deux grandeurs électriques, inversement proportionnelles : la résistance R dont la

valeur est exprimée en ohms (Ω), elle se déduit à partir du code couleur peint directement sur son corps, et la conductance G .

Plus la résistance est élevée, plus la conductance est faible, et inversement. La relation qui lie une résistance à une conductance est donnée par :

$$R = \frac{1}{G}$$

La résistance est la valeur inverse de la conductance. Une résistance élevée s'oppose au passage des électrons ; on peut la comparer à un goulet d'étranglement.

Lorsque la résistance est élevée, le courant qui la traverse est faible. Imaginez que vous couriez sur une surface lisse. A avancer ne devrait pas vous poser trop de problèmes. En revanche, si vous essayez de courir dans du sable en gardant la même allure, c'est fatigant. On dépense de l'énergie sous forme de chaleur et votre vitesse diminue. Il en va de même pour des électrons qui doivent traverser par exemple du verre (isolant) au lieu du cuivre (conducteur) [17].

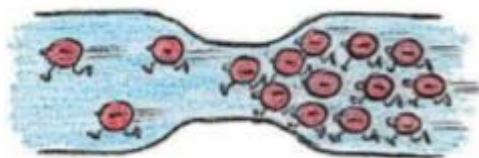


Figure III- 10 : La résistance freinant le flux d'électrons.

III.2.5 La loi d'Ohm

La loi d'Ohm décrit le rapport entre la tension U aux bornes d'une résistance R et le courant I qui la traverse. Voici la relation qui la définit :

$$R = \frac{U}{I}$$

La résistance s'obtient en divisant la tension par le courant : comme dit précédemment, elle s'exprime en ohms (Ω) et elle est désignée par la lettre R .

Nous utiliserons cette loi pour calculer la résistance de limitation pour une diode électroluminescente qui ne saurait fonctionner sans elle.

III.2.6 la puissance

La puissance reflète la vitesse à laquelle un travail est fourni. C'est la quantité d'énergie par unité de temps fournie par un système à un autre. C'est donc une grandeur scalaire. La puissance correspond donc à un débit d'énergie : si deux systèmes de puissances différentes fournissent le

même travail (la même énergie), le plus puissant des deux est celui qui est le plus rapide. La formule suivante nous montre la relation entre la tension et l'intensité du courant [24] :

$$P(\text{Watt}) = I(\text{Ampere}) * U(\text{volt})$$

III.2.7 L'énergie

Le terme énergie électrique désigne toute énergie transférée ou stockée grâce à l'électricité. Cette énergie est transférée d'un système à un autre par un mouvement de charges, elle est donnée par la relation suivante :

$$E(\text{Joul}) = P(\text{watt}) * t(\text{S}).$$

III.3 Les compteurs intelligents (smart meter)



Figure III- 11 : Compteur intelligent (smart meter).

Un smart meter ou compteur intelligent en français, est un compteur énergétique (électrique en général) capable de suivre en détail, et souvent en temps réel, la consommation électrique d'un bâtiment, d'une entreprise ou d'un foyer.

Ce compteur intelligent est en outre communicant et transmet par différents canaux (courant porteur, Internet, téléphone) les informations recueillies [25].

III.4 Composants matériel et logiciel requis :

La première partie de ce projet consiste à obtenir les composants nécessaires que nous allons utiliser pour notre projet de compteur d'énergie. Mis à part la carte Arduino Mega, qui sera le « cerveau » du projet, on devra disposer de deux parties principales sur notre bureau lors de la construction du projet. Ces composants sont le module relais, que nous allons utiliser pour allumer

et éteindre la lampe, et le capteur de courant analogique, qui est utilisé pour mesurer la puissance et, plus tard, la consommation d'énergie de la lampe.

Un relais est fondamentalement un commutateur électromagnétique utilisé dans les projets où nous devons commuter une très grande tension (110V ou 230V) en utilisant une petite tension comme le signal de commande (5V de la carte Arduino). Pour le relais, nous avons utilisé un module de relais 5V, qui peut passer à 10A et est plus que suffisant pour de nombreux appareils ménagers tels que les lampes. Le module lui-même est simplement un relais monté sur une carte de circuit imprimé avec les composants nécessaires pour faire fonctionner le relais et certains grands en-têtes et des traces à transporter jusqu'à 10A si nécessaire. Il utilise un relais SRD-05VDC-SL-C. L'image suivante représente le relais utilisé :

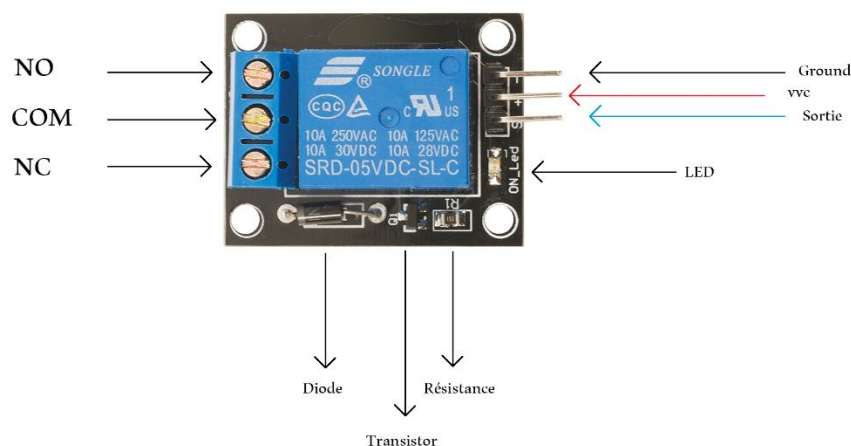


Figure III- 12 : Module relais SDR-05VDC-SL-C.

Bien sûr, on peut utiliser n'importe quel module de relais équivalent. Assurons simplement qu'il peut être allumé / éteint en utilisant un signal numérique 5V comme nous l'avons sur la carte Arduino Méga et qu'il peut changer au moins 5A, juste pour être sûr pour ce projet. La lampe que nous utilisons dans ce projet utilise seulement environ 340 mA, mais on peut connecter des appareils plus gros à notre projet plus tard. Si on veut notre propre module à partir d'un relais, on devrait simplement ajouter une diode en série avec le relais pour protéger notre carte Arduino lorsque le relais est en train de changer [26].

Ensuite, on a besoin d'un capteur de courant pour obtenir la consommation du courant instantanée de la lampe. Nous avons utilisé un module d'ITEAD Studio, qui est essentiellement une carte de dérivation pour le capteur ACS712 brick. Une carte de dérivation est simplement une carte qui est

composée d'une carte de circuit imprimé, la puce elle-même, et tous les composants nécessaires pour faire fonctionner la puce, tels que les résistances et les condensateurs. Ce capteur délivre un signal analogique en sortie proportionnel au courant mesuré. Ce signal peut ensuite être facilement converti en courant correspondant sur la carte Arduino Méga. Nous allons acquérir ce signal analogique en utilisant l'un des convertisseurs analogiques-numériques intégrés de la carte Méga. Il existe également des capteurs de courant non invasifs qu'on peut simplement couper autour du câble que nous souhaitons mesurer, mais ceux-ci sont généralement plus gros et ne s'intègrent pas bien avec les projets Arduino. Ce capteur est illustré à la figure suivante :

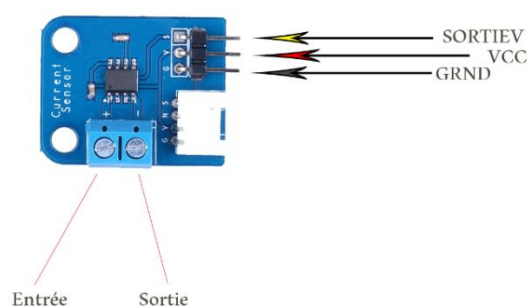


Figure III- 13 : Capteur du courant ACS712 brick

Tout comme pour le module relais, on peut utiliser n'importe quel capteur de courant équivalent pour ce projet. Les paramètres importants à prendre en compte sont le courant maximum pouvant traverser le capteur (5A) et la sensibilité du capteur (340 mV/A).

Connecter la lampe au projet d'une manière ou d'une autre. Bien sûr, un moyen serait de couper directement le câble d'alimentation de la lampe et de brancher la lampe directement sur notre projet.



Figure III- 14 : Lampe 75watt

On a utilisé un seul câble d'alimentation pour que je puisse connecter mon projet à la prise murale et connecter la lampe au projet avec ce dernier, comme je le ferais avec une prise male puis le

brancher à l'alimentation de la maison.



Figure III- 15 : le câble avec la duit utiliser.

On aura besoin aussi d'un module **WiFi** pour l'envoi des données à google docs. Ce module s'appelle ESP8266 et fait partie de la carte NodeMcu/Esp8266. **NodeMCU** est une plateforme IoT open source. Il comprend un microprogramme qui fonctionne sur le **SoC Wi-Fi ESP8266** d'Expressif System et du matériel basé sur le module ESP-12. Le terme "NodeMCU" désigne par défaut le firmware plutôt que les kits de développement. Le firmware utilise le langage de script Lua. Il est basé sur le projet eLua et sur le SDK Espressif non-OS pour ESP8266. Il utilise de nombreux projets open source, tels que lua-cjson et spiffs.

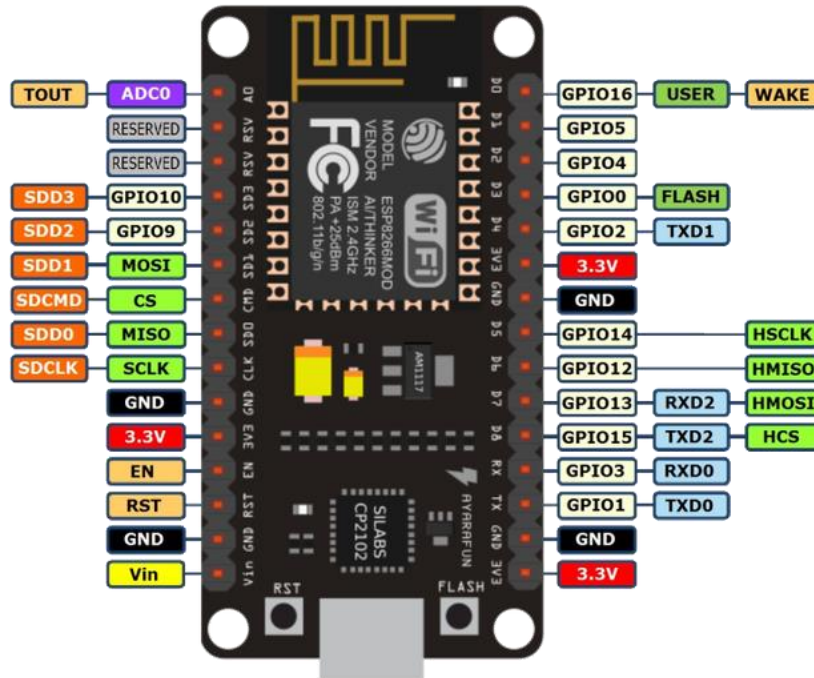


Figure III- 16 : Carte NodeMCU Dev Kit avec ses entrées et sorties.

ESP8266 est une conception de système sur puce (SoC) avec des composants comme la puce de processeur. Le processeur a environ 16 lignes GPIO, dont certaines sont utilisées en interne pour l'interface avec d'autres composants du SoC, comme la mémoire flash. Puisque plusieurs lignes sont utilisées en interne dans le SoC ESP8266, nous avons environ 11 broches GPIO restantes à des fins GPIO. Maintenant encore 2 pins sur 11 sont généralement réservés à RX et TX afin de communiquer avec un PC hôte à travers lequel le code objet compilé est téléchargé. Donc, finalement, cela ne laisse que 9 broches d'E/S à usage général, c'est-à-dire D0 à D8.

Comme indiqué dans la figure ci-dessus de NodeMCU Dev Kit, nous pouvons voir que les broches RX, TX, SD2, SD3 ne sont pas principalement utilisées comme GPIO puisqu'elles sont utilisées pour d'autres processus internes. Mais nous pouvons essayer avec la broche SD3 (D12) qui aiment surtout répondre aux fonctions GPIO / PWM / interruption.

Pour commencer à créer le code et le téléverser à la carte NodeMCU, on doit aller sur fichier->préférences -> ajouter le lien suivant :

http://arduino.esp8266.com/stable/package_esp8266com_index.json

comme le montre la figure ci-dessous :

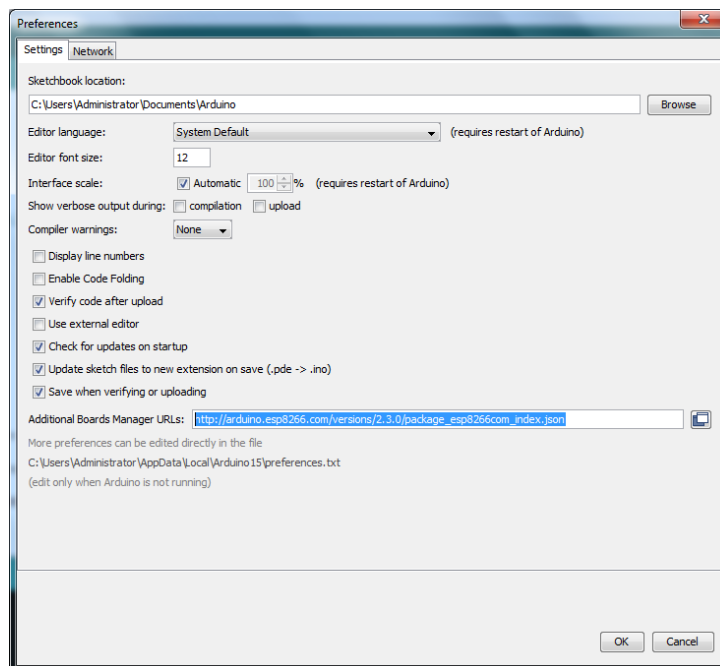


Figure III- 17 : Ajouter la carte NodeMCU à l'Arduino IDE

Cliquez sur OK pour fermer la fenêtre de préférences. Maintenant, allez dans Outils> Board & Select 'Board manager'.

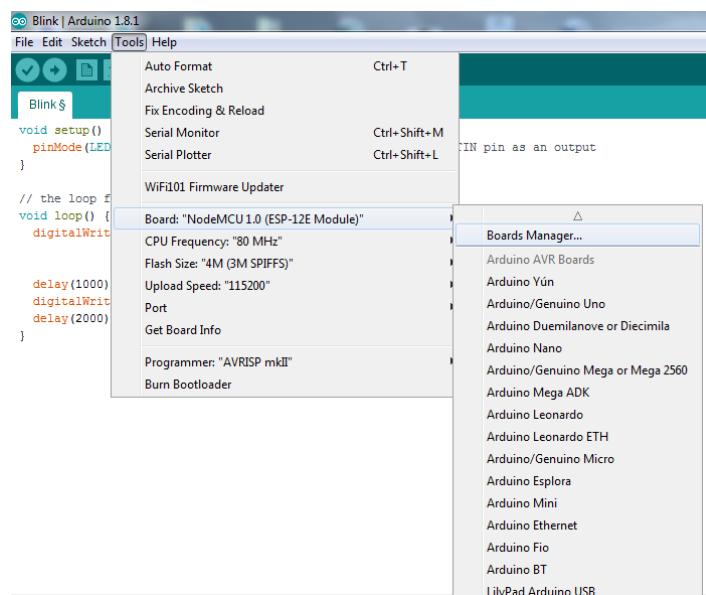


Figure III- 18 : Choix de la carte NodeMCU dans Arduino

Du côté du logiciel, on a besoin de la dernière version stable de l'IDE Arduino, qui est le seul qui supporte plusieurs cartes comme le NodeMcuEPS8266 (on a utilisé la version 1.8.5 dans notre projet) [27].

III.4.1 Connexion des composants à la carte Arduino Méga

Tout d'abord, on va commencer par brancher la tension de 5v ou le Vcc de la carte Arduino à la plaque d'essai. Ce dernier nous aide à brancher plusieurs dispositifs pour alimenter le capteur de courant et le relais. Pour cela, on va utiliser un câble rouge pour bien différencier entre d'autres branches, puis on branche le câble noir aux **GND** avec la plaque d'essai comme le montre la figure suivante :

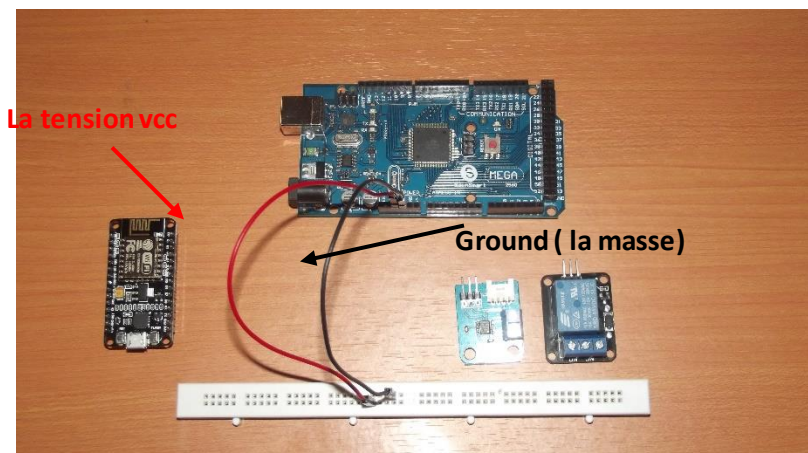


Figure III- 19 : Connexion de VCC et le GND a la plaque d'essai

Pour le module relais, il contient trois sorties la première s, c'est la sortie quand va l'utiliser pour commander notre relais (genre d'un Switch), on la branche avec le pin 8 et on va utiliser un câble bleu.

Le pin 8 signifie une sortie par rapport à la carte arduino, qu'on peut l'utiliser pour contrôler n'importe quel dispositif qui contient ce genre de sortie.

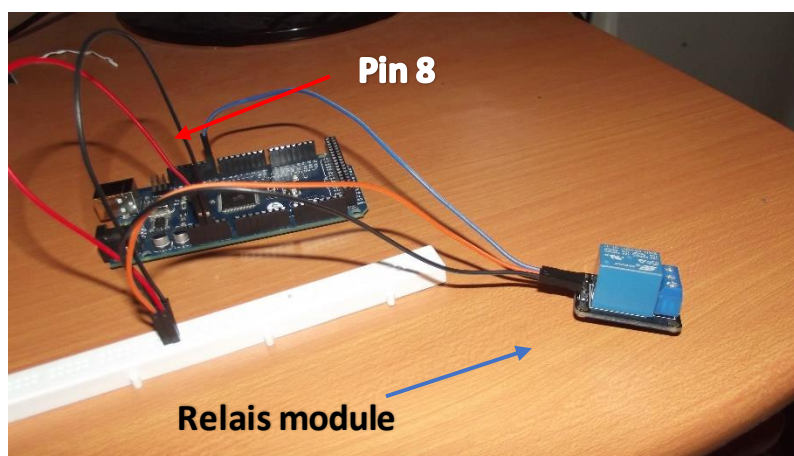


Figure III- 20 : Branchement de relais module avec la carte arduino

On refait la même chose avec le capteur de courant ; il contient aussi trois sorties mais pour ce genre de capteur, ou il y a une sortie quand va la placer sur ANALOG IN A0 on utilise un câble jaune c'est-à-dire en tant que entrée afin d'analyser le flux du courant entrant.

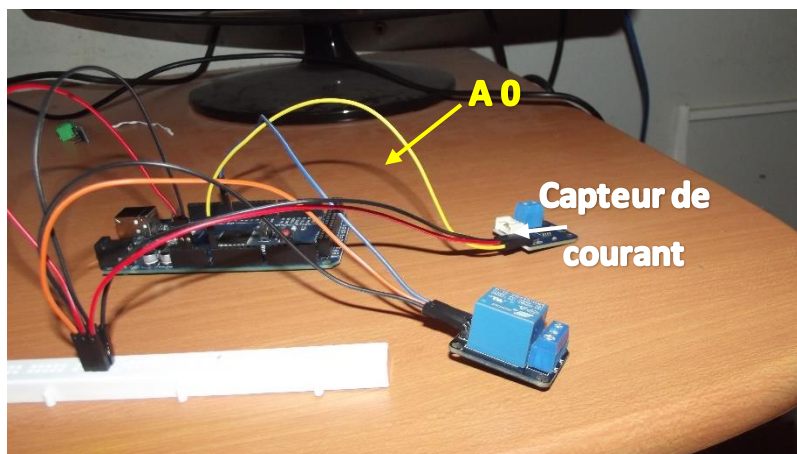


Figure III- 21 : Branchement de relais module avec la carte arduino

Pour terminer le branchement, on va ajouter notre module WiFi qu'est le NodeMCU, on le branche avec la sortie du capteur afin d'obtenir les informations nécessaires concernant le courant et calculer la valeur de l'énergie. On ajoute l'alimentation de 3.3v à la carte NodeMCU après le téléversement du programme à la carte, et on n'oublie pas de mettre le PIN A0 avec Arduino Mega aussi.

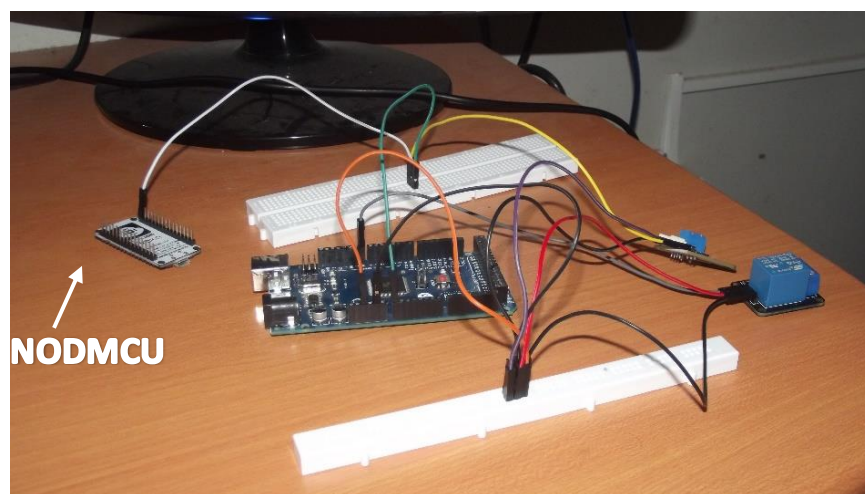


Figure III- 22 : Branchement du NODEMCU avec la carte arduino

Voici un schéma explicatif par **fritzing** qui nous montre la connexion entre les différents modules avec la carte arduino.

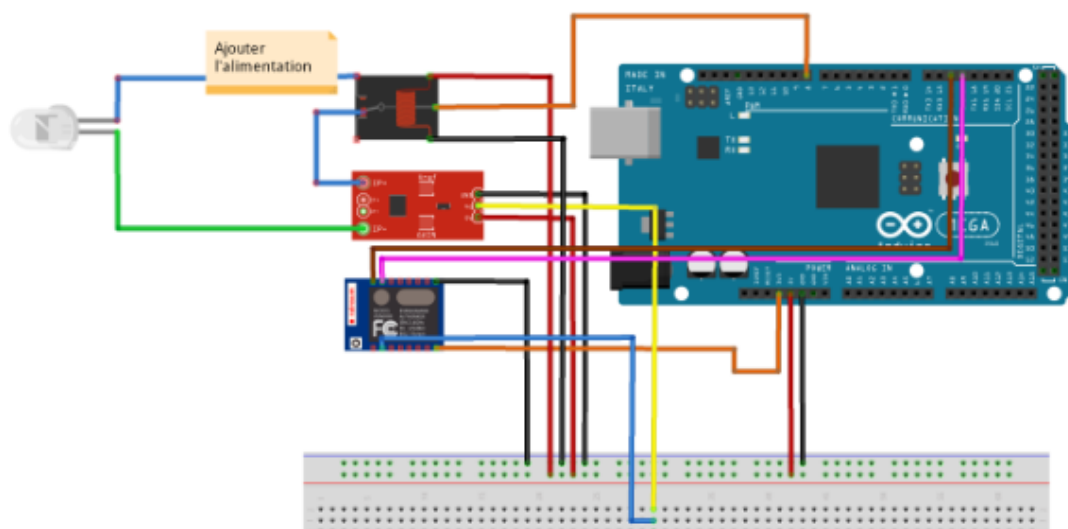


Figure III- 23 : Schéma explicatif de la connexion des modules à la carte arduino.

III.4.2 Tester la connexion du matériel

Maintenant que les connexions sont terminées, nous allons tout tester avant de commencer à envoyer des données de consommation d'énergie au cloud. Nous allons tester les différents modules comme si le projet était déjà opérationnel. Pendant toute la durée des tests, nous allons connecter le projet à la prise de courant dans le mur et à la lampe que nous voulons contrôler. De cette façon, nous ferons en sorte que toutes les connexions matérielles soient connectées avant de continuer à avancer.

Le relais, par exemple, sera contrôlé par une commande de switch pour allumer ou éteindre le switch. Fondamentalement, nous allons simplement envoyer une commande.

Pour le capteur de courant, nous allons simplement lire la valeur mesurée sur la broche analogique A0 en utilisant le convertisseur analogique-numérique de la carte Arduino méga et la convertir en un courant utilisable, puis calculer la valeur maximum et la valeur minimum du courant puis calculer la puissance que nous connaissons déjà la valeur de sa tension efficace (110V ou 230V selon l'endroit où vous vivez) afin de calculer l'énergie ou la consommation obtenue [26].

Pour le NODEMCU, aussi on va le connecter à notre modem puis on le relie au port série de la carte Arduino pour obtenir les informations de l'énergie électrique puis on crée notre doc Sheets à google drive et relie ce dernier avec la carte NODEMCU.

Commençons par une vue sur le code Arduino méga, il commence par une déclaration de variable pin 8 c'est-à-dire le port de module relais à l'extérieur des fonctions setup et loop :

```
const int relies_pin=8 ;
```

Dans la fonction setup, on déclare ce pin en tant que sortie utilisant la commande suivante :

```
pinMode (relies_pin, OUTPUT) ;
```

Dans la fonction loop, on donne un ordre ou une tension pour allumer la lampe par l'instruction suivante :

```
digitalWrite (relies_pin, HIGH) ;
```

Toutes ces déclarations sont dédiées au module relais pour le contrôler. Si on veut éteindre la lampe de notre circuit, on tape l'instruction suivante :

```
digitalWrite (relies_pin, LOW) ;
```

Pour le capteur de courant, on va déclarer le pin ANALOG A0 c'est-à-dire une entrée à la carte Arduino, ajoutons la déclaration d'une variable double puissance, et une autre pour la puissance crête à crête (la puissance maximale de capteur du courant), on les initialise par zéro comme suit :

```
int courantPin = 0; //le pin 0 (ANALOG A0)
```

```
double puissance = 0 ;
```

```
int puissance_crete= 0 ;
```

Un organigramme de fonctionnement du capteur de courant est donné par la figure ci-dessous :

Au début, on écrit le code de la fonction loop en commençant par déclarer les variables courant, courant_max, courant_min, pour tester s'il y a une tension aux bornes du capteur et on termine par le courant efficace, donc on va lire la valeur d'entrée sur le pin analogue A0, et on va faire un test sur la valeur du courant si elle est max c'est-à-dire qu'il y a du courant et par conséquent, on calcule la valeur du courant efficace puis celle de la puissance et on termine par calculer et afficher la valeur de l'énergie sur le port série, sinon, si la valeur du courant est inférieure au minimum du courant (il n'y a pas de courant), les valeurs de la puissance et de l'énergie sont nulles.

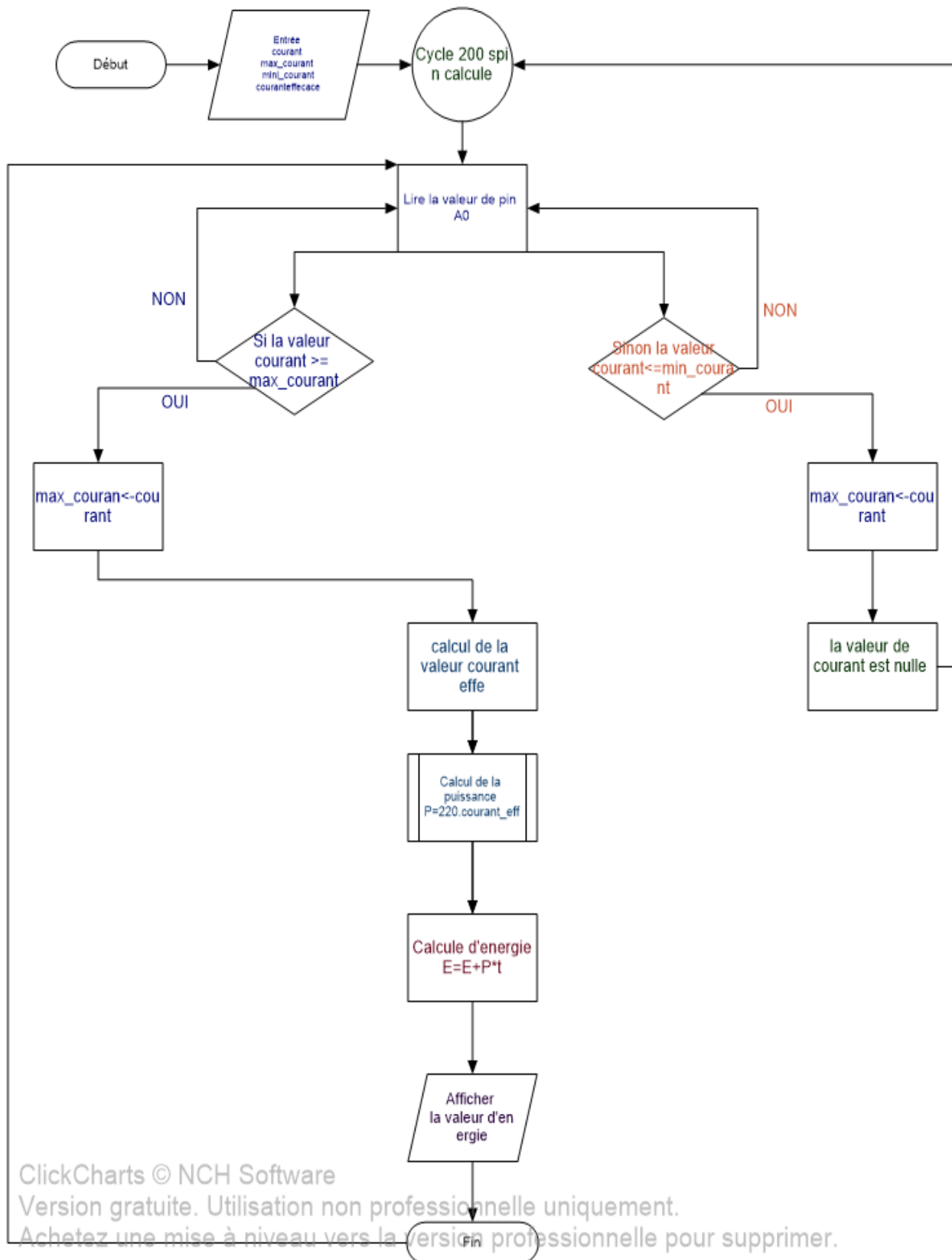


Figure III- 24 : Organigramme du capteur de courant dans la carte Arduino et dans le NodeMCU

Pour le NodeMCU, on établit la connexion au modem ou au point d'accès afin de connecter à l'internet si l'authentification est réalisée avec succès. On reçoit les données sur le port A0, puis, on envoie une requête HTTP à google server. L'opération s'effectue automatiquement chaque 20 seconde. On peut modifier cette valeur selon nos besoins.

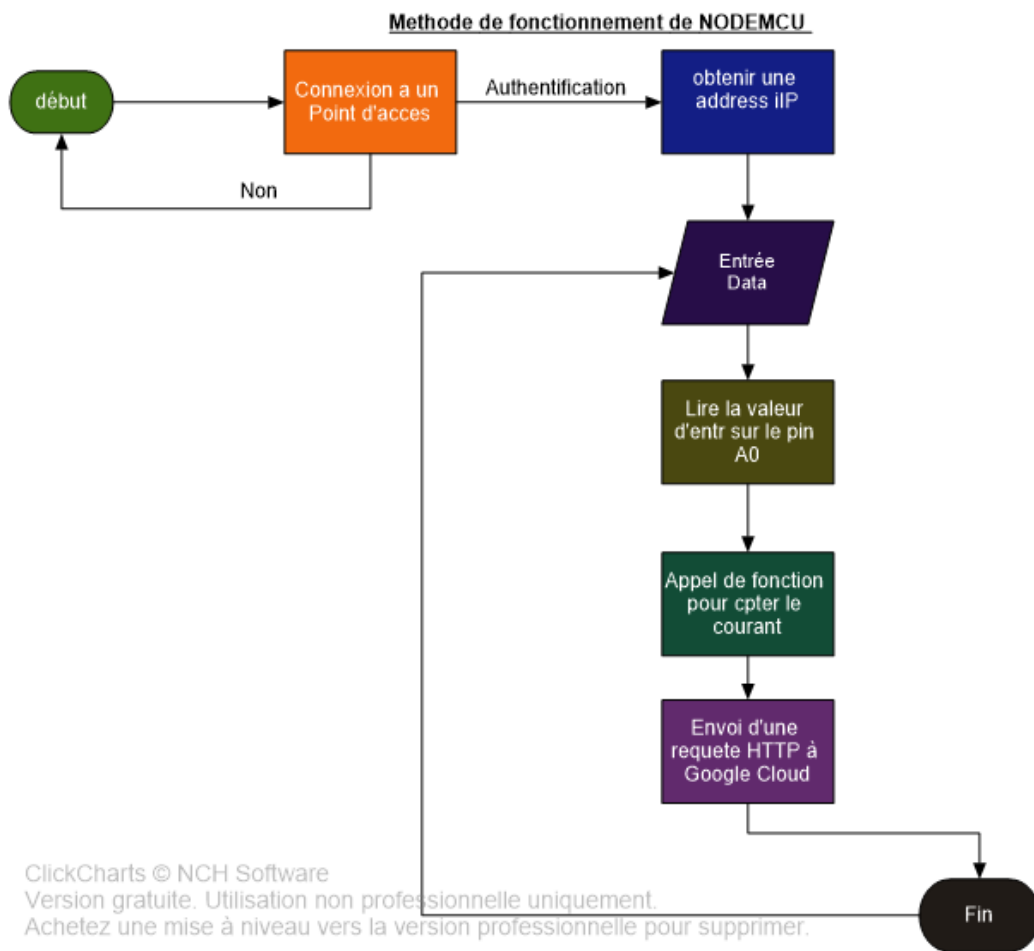


Figure III- 25 : Organigramme de la méthode de fonctionnement de NodeMCU.

HTTP est un espèce de langage qui va permettre au client (vous, par le biais de votre navigateur par exemple) de communiquer avec un serveur connecté au réseau (le serveur HTTP installé sur le serveur d'un site, par exemple [Apache](#)).

Dans le schéma suivant, les flèches représentent les requêtes HTTP :

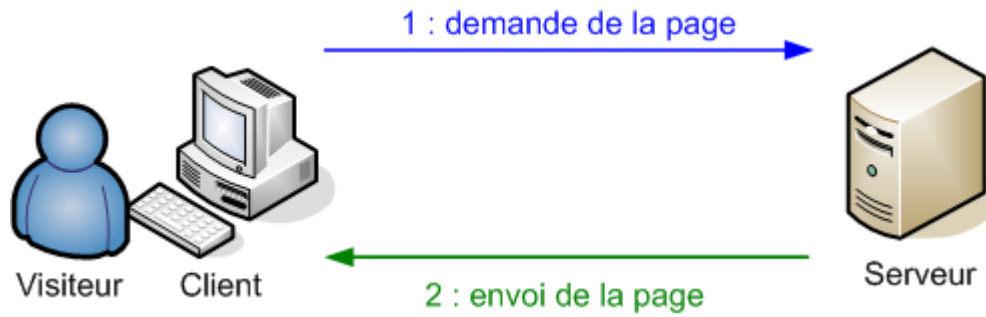


Figure III- 26 : Schéma explicatif d'envoi du client server avec http.

Les requêtes (les flèches du schéma) vont toujours par paires : la demande (du client) et la réponse (du serveur). Si ce n'est pas le cas, c'est qu'un problème est survenu à un endroit du réseau [28] Pour notre cas, on envoie des informations à un serveur. On a choisi google car c'est gratuit, en plus, il est disponible et peut supporter plusieurs requêtes à la fois.

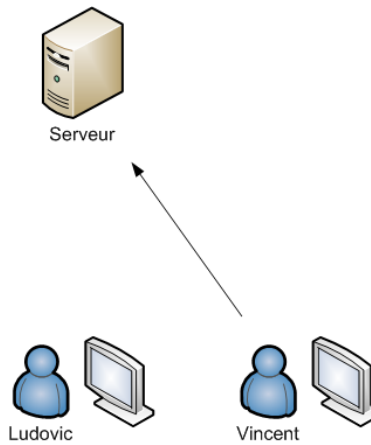


Figure III- 27 : Envoi du client vers le serveur

III.4.3 Envoi des données à google docs :

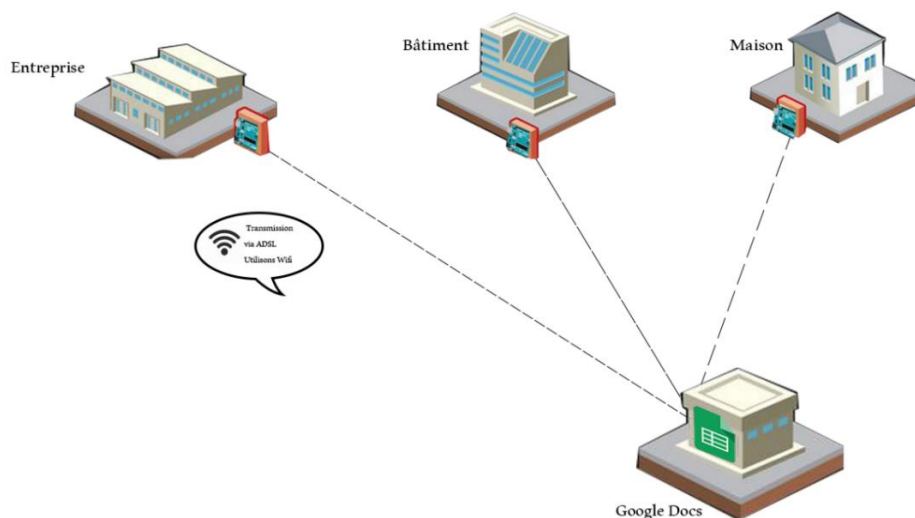


Figure III- 28 : Envoi des données a google docs

La première étape consiste à configurer une feuille de calcul Google Docs pour le projet. Créez une nouvelle feuille, vous lui donnez un nom (par exemple on la nomme Eneregie_monitoring, mais vous pouvez la nommer comme vous le souhaitez), et définissez un titre pour les colonnes que vous allez utiliser : Time, data, comme indiqué dans la capture d'écran suivante :

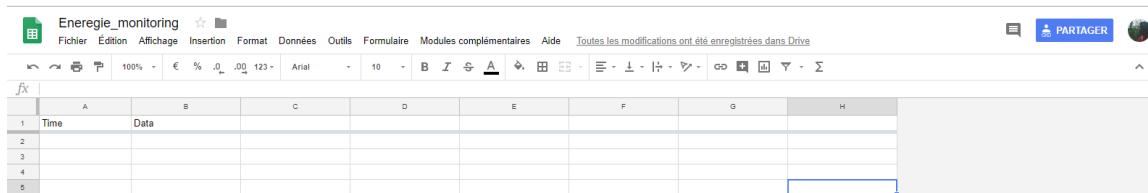


Figure III- 29 : Création d'une feuille a google drive

Cependant, dans notre cas, l'énergie a été calculée au niveau de la carte NodeMCU et elle est envoyée à un formulaire, et ce formulaire est relié avec le champ de la feuille à google docs. Après cette étape, on va créer un formulaire à remplir et on va relier ce formulaire avec la feuille de calcul Google Docs comme le montre la capture suivante :

Figure III- 30 : Création d'un formulaire.

Maintenant, on va relier le formulaire à la feuille de calcul précédente, et on clique sur l'onglet "Réponses". L'icône Google Sheet s'affiche juste au-dessus du bouton "Accepting Responses". On clique dessus et une fenêtre apparaîtra. On choisit "Sélectionner la feuille de calcul existante" et on clique sur "Sélectionner". Puis, choisissez la feuille de calcul que vous avez créée à l'étape 1.

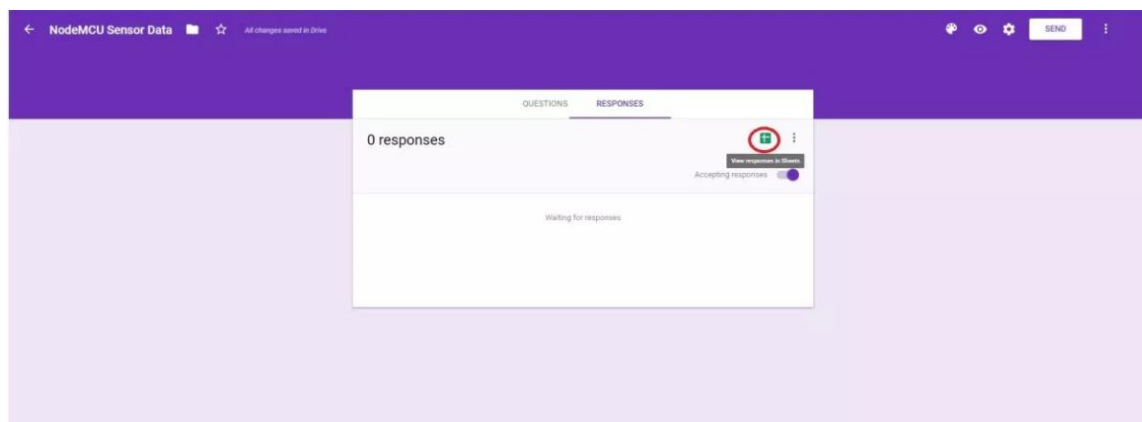


Figure III- 31 : Relier le formulaire avec la feuille 1.

Dans l'étape suivante, on va au site pushingbox.com afin de lier notre compte google avec cette API quand va l'utiliser pour l'envoi dans le code de la carte NodeMCU.

Maintenant, on va sur pushingbox.com et on lie notre compte en utilisant le même e-mail que nous avons utilisé pour Google Drive. Cliquons sur l'onglet "Mes services" et sur le bouton "Ajouter un service". Sélectionnons "CustomURL", ensuite, remplissons le formulaire. Nommons le service dans le champ URL racine, collons l'URL du formulaire Google, mais on va remplacer «viewform» par «formResponse». Exemple, en utilisant l'URL ci-dessus, ce serait : <https://docs.google.com/spreadsheets/d/1RCw7w2kVITGcaEX2T5I18fkJYPVAfWGRiesAwcj61Ng/formResponse>. Laissez le champ Méthode comme GET.

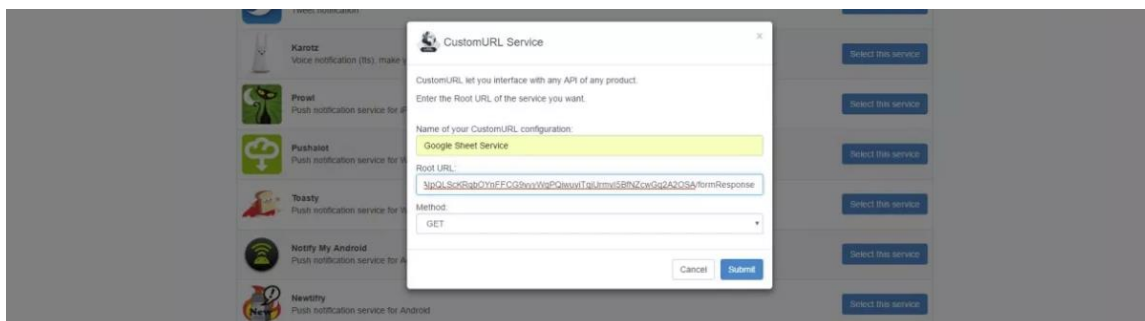


Figure III- 32 : Relier le formulaire avec la feuille 2.

Après cette étape, on clique sur 'My Scenarios' et on va copier le 'DeviceID' et on le colle à la ligne 89 dans notre projet.

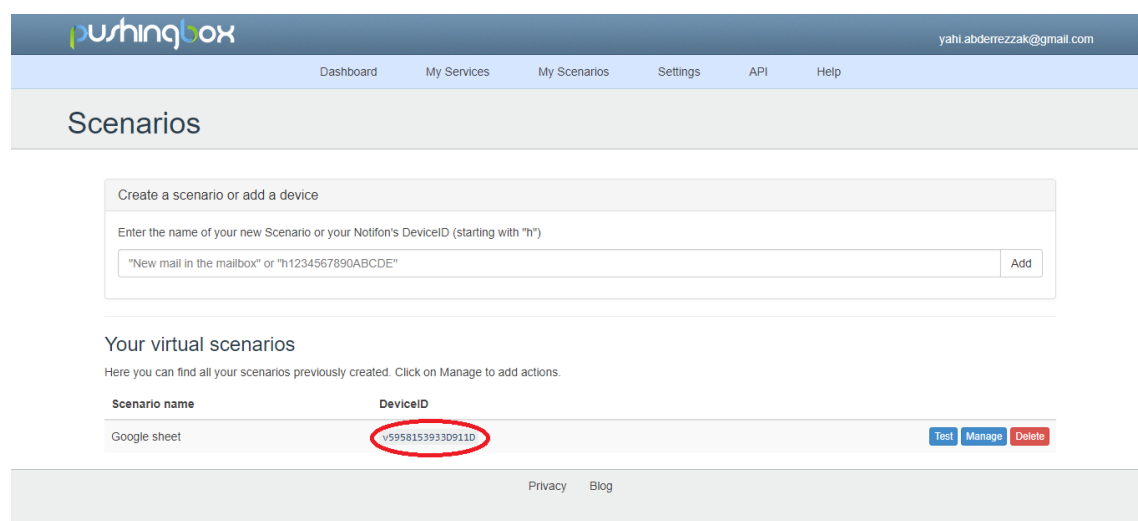


Figure III- 33: Obtenir le DEVID.

```

86 // We now create a URI for the request
87 String url = "/pushingbox?";
88 url += "devid=";
89 url += "v5958153933D911D";
90 url += "&data="+String(data)+"Kwh";

```

Figure III- 34 : Ajouter DEVID a notre code.

Finalement, on va alimenter nos modules et téléverser le code dans la carte arduino Mega, puis dans la carte NodeMCU. On va sur notre drive et attendre les résultats. Après avoir fermé le relais, on va recevoir les informations ou les données chaque 20 secondes.

Time	Data
17/05/2018 06:15:51	0.00 Kwh
17/05/2018 06:16:08	0.00 Kwh
17/05/2018 06:16:24	0.00 Kwh
17/05/2018 06:16:42	0.00 Kwh
17/05/2018 06:16:59	0.00 Kwh
17/05/2018 06:17:16	0.00 Kwh
17/05/2018 06:17:33	0.00 Kwh
17/05/2018 06:17:49	0.00 Kwh
17/05/2018 06:18:06	0.00 Kwh
17/05/2018 06:18:23	0.00 Kwh
17/05/2018 06:18:40	0.00 Kwh
17/05/2018 06:18:56	0.00 Kwh
17/05/2018 06:19:13	0.00 Kwh
17/05/2018 06:19:30	0.00 Kwh
17/05/2018 06:19:47	0.00 Kwh
17/05/2018 06:20:03	0.00 Kwh
17/05/2018 06:20:20	0.00 Kwh
17/05/2018 11:55:21	0.00 Kwh

Figure III- 35 : Etape sans fermeture de switch dans relais.

Time	Data
17/05/2018 01:07:43	37.74 kwh
17/05/2018 01:08:00	40.26 kwh
17/05/2018 01:08:18	42.78 kwh
17/05/2018 01:08:35	45.27 Kwh
17/05/2018 01:08:52	47.80 kwh
17/05/2018 01:09:09	50.28 kwh
17/05/2018 01:09:26	52.81 kwh

Figure III- 36 : Etape avec fermeture de switch dans relais.

III.5 Conclusion

L'informatique, l'électronique, et les télécommunications rendent notre vie automatisée. Ces disciplines nous facilitent l'accomplissement des tâches, même si elles sont complexes. Pour cela, il faut aller vers ces domaines pour un meilleur futur dans les NTIC.

Dans ce chapitre, on a présenté une étude sur l'électricité de point de vue théorique tout en expliquant la loi d'ohm, les concepts de résistance, de puissance ...

Une démonstration a été faite sur le matériel utilisé avec les explications nécessaires, citons le capteur de courant, relais, et le module WiFi qu'est NODEMCU/ESP8266. Après, on a testé la connexion de ces modules, et on a fini par vous présenter l'envoi de ces données au cloud (google docs).

Conclusion générale

Ce projet m'a permis de découvrir et de consolider mes connaissances en électronique et en informatique. Il m'a aussi permis de découvrir la technologie d'arduino qui, une fois maîtrisée, permet à son utilisateur de l'utiliser à des fins multiples, toutes plus intéressantes les unes que les autres. L'avantage est que je possède à présent un bagage de connaissances plus important sur le sujet, mes connaissances en informatique (la programmation c++, arduino, Cloud) ainsi qu'en électronique et en télécommunications ayant été fortement consolidées. Je peux donc dire que le travail réalisé a été grandement profitable. Le projet était vraiment très intéressant mais assez dur à réaliser étant donné que j'ai commencé à zéro et citons aussi la disponibilité du matériels. Mes connaissances en base de données en électronique n'étaient pas suffisantes au début du projet, et j'ai effectué beaucoup de recherches avant de pouvoir réaliser mon travail. Et cela a demandé du temps et un investissement personnel assez important dans le but de comprendre correctement le fonctionnement de ce projet. Enfin, ce projet était une occasion pour l'initiation au domaine de l'embarqué. Ce dernier qui est devenu un grand domaine de recherche aujourd'hui. En perspective, j'ai pensé à embarquer un système afin de trouver une solution pour l'envoi et le contrôle de l'énergie électrique pour minimiser le gaspillage de cette ressource. On peut aussi se focaliser sur les serveurs locaux et créer notre Cloud avec ses configurations, ensuite, créer des interfaces graphique à travers les plateformes en utilisant la technologie du web afin de voir les détails sur l'objet suivi, après on essaie d'ajouter la surveillance du l'eau et de gaz. Enfin, on peut assurer la sécurité de ces appareils et ces systèmes en utilisant des mécanismes de sécurité et des méthodes d'authentification plus élevés pour éviter le sabotage de ce projet

Bibliographie

- [1] <http://docplayer.fr/158873-Systemes-temps-reel-et-embarques-concepts-de-base-expression-des-contraintes-temporelles.html>
- [2]http://persoetis.ensea.fr/miramond/Enseignement/M2_ESA/ArchiEmbarquee/Cours/ArchitectureM2_ESA.pdf
- [3] BOULKROUNE Ramzi, Les systèmes embarqués, Université de Annaba-Ingénieur d'état en électronique option télécommunication 2009.
- [4]https://ent2d.ac-bordeaux.fr/disciplines/sti/wp-content/uploads/sites/41/2017/10/T.C.-3.1.4_3.docx
- [5] https://www.fun-mooc.fr/c4x/MinesTelecom/04013/asset/S4-5_-Objets-communicants.pdf
- [6] Smart home Automation with linux and raspberry Pi, 2nd edition (Goodwin, 2013), ISBN-10: 143025887X,En.
- [7] The Internet of Things,(Anupama C. Raman, Pethuru Raj,2017), ISBN-10: 1498761283,En
- [8] https://www.researchgate.net/publication/310490018_La_Seurite_des_Objets_Connectes
- [9] <http://www.fondation-stae.net/docs/bd/events/document-48.pdf>
- [10] https://www.newikis.com/fr/wiki/Big_data
- [11] https://www.territorioscuola.com/fr.php?title=Big_data
- [12] Internet of Things and Big Data Technologies for Next Generation Healthcare,(Amira S. Ashour, Chintan Bhatt, Nilanjan Dey,2017), ISBN-10:3319497359,En.
- [13] <http://www.lenaturaliste.net/forum/viewtopic.php?t=20934>
- [14] https://www.memoireonline.com/05/17/9963/m_La-qualite-de-service-au-niveau-iaas-vers-lutilisation-du-concept-software-defined-networking-2.html
- [15] https://www.makacra.com/L-Intelligence-Artificielle-au-service-de-l-humain-_a5307.html
- [16] <https://epnrelais59.files.wordpress.com/2015/10/arduino.pdf>
- [17]Le grande livre d'arduino, 2^{eme} edidtion (Erik bartmann ,2014) Edition EYROLLES, ISBN 978-3-95561-115-6 ,Fr.
- [18] A l'aventure avec arduino (John Wiley & Sons,2015) Edition EYROLLES,ISBN 978-1-118-94847-7,Fr.
- [19]Arduino pour les nuls poche, 2^{eme} edidtion(Wiley publishing, 2017)Edition First, ISBN numérique 978241202958
- [20][http://www.icpl.fr/images/PDF/technique/La%20photo%20panoramique%20a%20main%201evee%20\(Michel%20Evrat\).pdf](http://www.icpl.fr/images/PDF/technique/La%20photo%20panoramique%20a%20main%201evee%20(Michel%20Evrat).pdf)
- [21] <http://www.caminitodelrey.info/fr/5154/introduction-electricite-appliquee-industrie>
- [22]<http://www.capre06.eu/pages/environnement/energie-energies-renouvelables/energie-electrique.html>

[23] <http://docplayer.fr/46037691-Qu-est-ce-que-l-electricite.html>

[24] le grande livre de l'électricité, 4^{ème}(Thierry Gallauziaux et david Fedullo,2016) Edition EYROLLES, ISBN 978-2-212-14455-0,Fr

[25]<https://www.futura-sciences.com/planete/definitions/developpement-durable-compteur-intelligent-6952/>

[26] Internet of Things with the arduino YUN (Marco Schwartz, 2014) Park Publishing,ISBN V978-1-78328-800-7,En.

[27] <https://nodemcu.readthedocs.io/en/master/>.

[28] <https://openclassrooms.com/courses/les-requetes-http>.