

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي و البحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكر بلقايد- تلمسان

Université Aboubakr Belkaïd-Tlemcen

كلية التكنولوجيا

Faculté de Technologie

Département de Génie Electrique et Electronique (GEE)

Filière : Electronique



MASTER INSTRUMENTATION
PROJET DE FIN D'ETUDES

Présenté par : Benomari Houari & Khelifa Sidi Mohamed.

Intitulé du Sujet

Réalisation d'un parking intelligent (smart parking) en
utilisant la technologie des objets connectés-IOT

Soutenu en 04/07/2021, devant le jury composé de :

M^r MASSOUM Nouredine

MCB

Univ. Tlemcen

Président

M^r NEMMICHE Ahmed

MCB

Univ. Tlemcen

Examineur

M^r BOUANATI Sidi Mohammed

M.A.A

Univ. Tlemcen

Encadrant

Année Universitaire 2020-2021

Remerciements

Avant tout, nous remercions Allah de nous avoir gardé en bonne santé et donné le courage pour mener à bien ce projet de fin d'études. Nous remercions également nos familles pour les sacrifices qu'elles ont consentis afin que nous puissions terminer nos études.

*Nous remercions cordialement Monsieur **Bouanati Sidi Mohammed** maitre d'assistant de classe « A » à l'Université Abou bekɾBelkaid Tlemcen pour nous avoir encadrés.*

*De même, nous remercions Monsieur **Massoum Noureddine** qui nous a fait l'honneur de présider le jury pour la soutenance de notre PFE.*

*Nous remercions également Monsieur **Nemmich Ahmed** l'examineur, qui a eu l'honneur d'accepter nos sympathies pour siéger dans notre jury.*

Nos remerciements vont également à tous nos professeurs, pour nous avoir formés durant nos études.

Nous tenons à remercier en particulier tous ceux qui ont contribué à finaliser ce modeste travail qui, nous l'espérons, sera à la hauteur de leur engagement.

Liste des abréviations

IoT: Internet of Things.

RFID: Radio-Frequency Identification.

IP: Internet Protocol.

GPS: Global Positioning System.

Wi-Fi: Wireless Fidelity.

WSN: Wireless Sensor Network.

M2M: Machine to Machine.

IDE: Integrated Development Environment.

NodeMCU: Node Microcontroller Unit.

TCP: Transmission Control Protocol.

USB: Universal Serial BUS.

IR: Infrared.

SPI: Serial Périphérique Interface.

I2C: Inter-Integrated Circuit.

SCL: Serial Clock Line.

SDA: Serial Data Line.

UV : Ultraviolet.

CAO : Conception Assistée par Ordinateur.

ISIS: Intelligent Schematic Input System.

IC : Integrated Circuit.

Table des matières

<i>Remerciements</i>	I
<i>Liste des abréviations</i>	II
<i>Table des matières</i>	III
<i>Liste des figures</i>	VI
<i>Liste des tableaux</i>	VIII
INTRODUCTION GENERALE	1
CHAPITRE I	2
I.1. Introduction	2
I.2. Historique	2
I.3. Définition	3
I.4. Fonctionnement de l'IoT	3
I.4.1 RFID (Radio Frequency Identification)	4
I.4.2 WSN (Wireless Sensor Network)	4
I.4.3 Machine to Machine (M2M)	5
I.5. Domaines d'application de l'IoT	5
I.5.1 Internet industriel	6
I.5.2 Maison intelligente	6
I.5.3 Hôpital intelligent (Santé connectée)	7
I.5.4 Voiture intelligente	7
I.5.5 L'agriculture intelligente	8
I.5.6 Parking intelligent	9
I.5.7 Ville intelligente	9
I.6. L'avenir de l'internet des objets	10
I.7. Les avantages et les inconvénients de IoT	11
I. 7.1. Les avantages	11
I. 7.2. Les inconvénients	11
I.8. Conclusion	12
CHAPITRE II	13
II.1 Introduction	13
II.2 Les composants électroniques utilisés	13
II.3 La carte Arduino	13
II. 3.1 Les différentes cartes Arduino	13
II. 3.2 Aperçu sur Arduino Méga 2560	14
II. 3.3 Caractéristiques de l'Arduino Mega2560	15
II. 3.4 Configuration du brochage de l'Arduino Méga 2560	16

II. 3.4.1	Alimentation	16
II. 3.4.3	Broches d'Entrées et sortie analogiques	17
II. 3.4.4	Autres broches	17
II. 3.5	Microcontrôleur ATmega2560	18
II. 3.6	L'environnement de développement (IDE) de l'Arduino	19
II. 4	La carte WI-FI « NodeMCU »	20
II. 4.1	Description de la carte NodeMCU / ESP8266	20
II. 4.2	Caractéristiques de NodeMCU	21
II. 4.3	Configuration du brochage de NodeMCU	22
II. 4.4	Plateformes de développement de l'ESP8266	23
II. 4.5	BLYNK	24
II. 4.5.1	Description du plateforme BLYNK	24
II. 4.5.2	Comment fonctionne BLYNK ?	24
II. 5	Servomoteur	25
II. 5.1	Description de servomoteur	25
II. 5.2	Programmation de servomoteur	26
II. 6	Capteur IR (FC-15)	26
II. 6.1	Description de capteur IR	28
II. 6.2	Principe de fonctionnement	28
II. 6.3	Émetteur IR et récepteur IR	29
II. 6.4	Schéma bloc d'un capteur IR	30
II. 7	Capteur de flamme	31
II. 7.1	Présentation de capteur de flamme IR	31
II. 7.2	Les différents types de capteur de flammes	31
II. 7.3	Principe de fonctionnement de capteur de flamme	32
II. 8	Écran LCD I2C 16X2 avec rétro-éclairage bleu	33
II. 8.1	Présentation du matériel de l'écran LCD I2C	34
a.	Écran LCD	34
b.	Adaptateur I2C pour LCD	34
II. 8.2	Adresse I2C du LCD	34
II. 8.3	Connecter un Arduino Méga à un écran LCD I2C	34
II. 9	Système RFID	34
II. 9.1	Description de la carte RFID RC522	35
II. 9.2	Principe de fonctionnement d'un système RFID	36
II. 10	Buzzer	36
II. 10.1	Fonctionnalité	37
II. 11	CONCLUSION	37
CHAPITRE III		38
III.1	Introduction	38
III.2	La partie Simulation	39
III.2.1	Présentation du logiciel PROTEUS	39
III.2.2	Présentation du module ISIS	39
III.3	Schémas et méthodologie suivie	40
III.3.1	Description du prototype	40

III.3.1.1	Bloc d'accès au parking	41
III.3.1.2	Bloc de la gestion du parking	42
III.3.1.3	Bloc d'affichage des informations	43
III.4	La partie réalisation	44
III.5	Conclusion	56
CONCLUSION GENERALE		57
REFERENCES BIBLIOGRAPHIQUES		58
RESUME		61

Liste des figures

Figure I. 1 : Historique de l'IOT. -----	2
Figure I. 2 : Internet des objets. -----	3
Figure I. 3 : Technologie RFID. -----	4
Figure I. 4 : Le fonctionnement de la technologie WSN. -----	4
Figure I. 5 : Le fonctionnement de la technologie M2M. -----	5
Figure I. 6 : Les Domaines d'application de l'IoT. -----	5
Figure I. 7 : Internet industriel. -----	6
Figure I. 8 : Maison intelligente. -----	6
Figure I. 9 : Hôpitaux intelligente. -----	7
Figure I. 10 : Voiture intelligent. -----	8
Figure I. 11 : L'agriculture intelligente. -----	8
Figure I. 12 : Parking intelligent. -----	9
Figure I. 13 : Ville intelligente. -----	10
Figure I. 14 : L'avenir de l'internet des objets. -----	11
Figure II. 1 : Les différentes cartes Arduino. -----	14
Figure II. 2 : Une carte Arduino Méga2560. -----	15
Figure II. 3 : Alimentation de l'Arduino Méga2560. -----	16
Figure II. 4 : Broches d'E/S numériques de l'Arduino Méga2560. -----	17
Figure II. 5 : Broches d'E/S analogiques de l'Arduino Méga2560. -----	17
Figure II. 6 : Broches RESET / AREF de l'Arduino Méga2560. -----	18
Figure II. 7 : Microcontrôleur ATmega2560. -----	18
Figure II. 8 : Fonctionnement d'un programme Arduino. -----	19
Figure II. 9 : La carte WI-FI « NodeMCU-ESP8266/ESP32 ». -----	20
Figure II. 10 : Une carte WI-FI ESP8266. -----	21
Figure II. 11 : L'essentiel sur ESP8266. -----	23
Figure II. 12 : Connecter ESP8266 avec téléphone mobile. -----	23
Figure II. 13 : Fonctionne Blynk. -----	25
Figure II. 14 : Servomoteur "9 grammes". -----	25
Figure II. 15 : Structure de Servomoteur "9 grammes". -----	26
Figure II. 16 : Capteur IR et Capteurs de flamme. -----	27
Figure II. 17 : Structure d'un Capteur IR. -----	28
Figure II. 18 : Principe de fonctionnement d'un capteur IR. -----	29
Figure II. 19 : Émetteur IR. -----	29
Figure II. 20 : Récepteur IR. -----	30
Figure II. 21 : Schéma bloc d'un capteur IR. -----	30
Figure II. 22 : Structure d'un Capteur de flamme. -----	31
Figure II. 23 : Schéma du circuit du module de détection de flamme à infrarouge IR. -----	33
Figure II. 24 : Structure d'un écran LCD-I2C (16*2). -----	33
Figure II. 25 : Schéma de brochage du RC522. -----	35
Figure II. 26 : Schéma de principe de système RFID. -----	36
Figure II. 27 : Buzzer (Avertisseur sonore). -----	36
Figure III. 1 : Schéma général de communication du système prototype. -----	38
Figure III. 2 : Représentation schématique du prototype. -----	40
Figure III. 3 : Le branchement du capteur IR et servomoteur avec Arduino Méga sous ISIS. -----	41
Figure III. 4 : Représentation du système RFID sous Fritzing. -----	42
Figure III. 5 : Branchement des capteurs IR avec la carte ESP8266. -----	42

Figure III. 6 : Bloc d'affichage sous ISIS.-----	43
Figure III. 7 : Affiche un message d'accueil « Bienvenu Smart parking ».-----	44
Figure III. 8 : Nombre de places occupées et libres.-----	44
Figure III. 9 : Présenter sa carte RFID valide à l'entrée.-----	45
Figure III. 10 : Programme pour le système RFID.-----	46
Figure III. 11 : Détection d'une voiture en cas d'entrée.-----	47
Figure III. 12 : L'afficheur LCD en cas d'entrée d'une voiture.-----	47
Figure III. 13 : Code de l'entrée de parking sur l'Arduino IDE -----	48
Figure III. 14 : Les données de connexion entre BLYNK et ESP8266 via Wifi.-----	49
Figure III. 15 : Ajoute le tableau et les LEDs dans l'application BLYNK.-----	49
Figure III. 16 : Configuration de l'objet « tableau » dans l'application.-----	50
Figure III. 17 : Code de création du tableau dans l'Arduino IDE.-----	50
Figure III. 18 : Code d'affiche le nombre de place et le prix.-----	51
Figure III. 19 : Code pour affiche l'emplacement et le prix.-----	51
Figure III. 20 : Affichage de l'emplacement sur l'écran du smart phone.-----	52
Figure III. 21 : Affiche le prix sur smart phone.-----	52
Figure III. 22 : Détection d'une voiture en cas de sortie.-----	53
Figure III. 23 : L'afficheur LCD en cas de sortie trio voiture.-----	53
Figure III. 24 : La détection d'incendie.-----	54
Figure III. 25 : Affichage du message d'alerte en cas d'incendie.-----	55
Figure III. 26 : représentation globale du smart parking.-----	55
Figure III. 27 : représentation globale du smart parking.-----	56

Liste des tableaux

Tableau II. 1 : Composants électroniques utilisés.-----	13
Tableau II. 2 : Caractéristiques de l'Arduino Mega2560. -----	15
Tableau II. 3 : Caractéristiques de carte WI-FI ESP8266.-----	21
Tableau II. 4 : Configuration du brochage de carte WI-FI ESP8266. -----	22
Tableau II. 5 : Sortie et entrée d'un Capteur IR. -----	28
Tableau II. 6 : Les spécifications de la carte RFID RC522. -----	35
Tableau III. 1 : Le brochage du module RC522 avec ESP8266. -----	45
Tableau III. 2 : Branchement des capteurs IR avec la carte ESP8266. -----	48
Tableau III. 3 : Représentation le pin virtuel de cheque spot.-----	50

Introduction générale

Introduction générale

Aujourd'hui, le transport représente un secteur important pour le développement économique des pays. On effectue plusieurs moyens de transport tels que (avion, voiture, train, bateau). Cependant, le moyen de transport le plus utilisé par les individus est la voiture.

Par conséquent, nous constatons une augmentation du nombre de véhicules. Ceci nécessite la réalisation des infrastructures pour gérer et réduire le trafic des véhicules (routes, parkings, etc...).

En effet, notre projet consiste à réaliser un parking intelligent en utilisant la technologie IOT. Cette technologie nous offre la possibilité de développer des applications et d'installer des systèmes connectés en utilisant des capteurs, des actionneurs et d'autres objets physiques, connectés via des protocoles de communication, ce qui permet une interconnexion des données et l'échange des informations.

Notre projet de fin d'études est organisé autour de trois chapitres :

Dans le premier chapitre, nous avons donné un aperçu sur la technologie IoT, ces domaines d'application, et ces avantages et inconvénients.

Le deuxième chapitre sera consacré à la description des différents éléments utilisés pour la réalisation du prototype.

Le dernier chapitre est divisé en deux parties, la première partie consiste à simuler notre système sous l'environnement PROTEUS. Par la suite, nous allons passer à la réalisation du circuit global.

Chapitre I

Internet des objets (IoT)

I.1. Introduction

Ces dernières années, l'internet des objets (IoT) est devenu l'un des phénomènes les plus discutés, car l'internet des objets a suscité l'intérêt de nombreux chercheurs, ingénieurs et industriels de toutes disciplines. L'internet des objets fait partie de l'internet du futur et comprendra des milliards de "choses" dotées d'une connectivité intelligente.

Cette nouvelle technologie rend le tissu du monde qui nous entoure plus intelligent et plus réactif dans tous les domaines de la surveillance, de la sécurité, de la santé, des maisons et des villes intelligentes.

Notre objectif dans cette partie est d'abord de définir les principaux concepts, Historique, fonctionnement, domaines d'application, l'avenir, les avantages et les inconvénients de L'internet des objets.

I.2. Historique

Le terme « Internet des objets » a été inventé par Kevin Ashton en 1999, initialement pour promouvoir la technologie RFID. Les premiers objets communicants apparaissent dans les années 2000. Le réfrigérateur relié à internet du coréen LG, puis en 2003 la lampe DAL qui s'éclaire de différentes couleurs selon ce qui se passe dans son environnement. Ensuite, la popularité du terme « Internet des objets » s'est accélérée après l'émergence des Smartphones en 2007 et la création d'un groupe de recherche appelé l'Alliance IPSo, spécialisée dans la promotion de l'utilisation du protocole Internet (IP) pour les réseaux de micro-organismes intelligents en 2008, pour atteindre le marché de masse début 2014. [1]

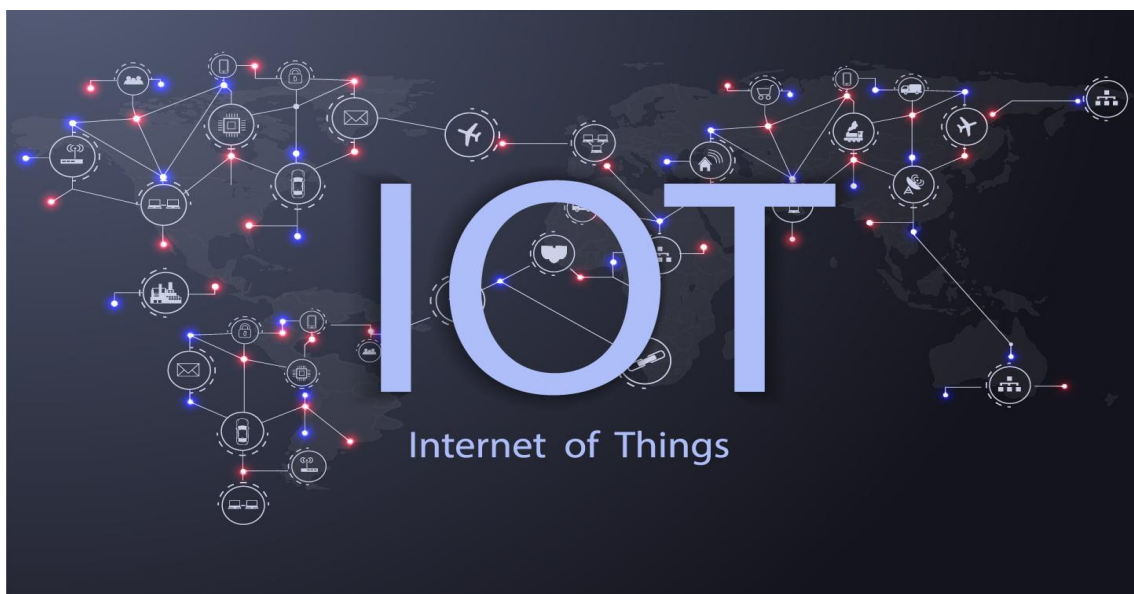


Figure I. 1 : Historique de l'IOT. [2]

I.4.1 RFID (Radio Frequency Identification)

Le terme RFID comprend toutes les technologies qui utilisent des fréquences radio pour identifier automatiquement des objets ou des personnes. Il s'agit d'une technologie qui permet de stocker et de récupérer des informations à distance grâce à une étiquette qui émet des fréquences radio. [6]



Figure I. 3 : Technologie RFID. [7]

I.4.2 WSN (Wireless Sensor Network)

Il s'agit d'un groupe de nœuds qui communiquent sans fil et sont organisés en un réseau collaboratif. Chaque nœud a une capacité de traitement et peut contenir différents types des capteurs dédiés. Les WSN peuvent mesurer les conditions environnementales telles que la température, le son, les niveaux de pollution, l'humidité et le vent. Comme son nom l'indique, le WSN est un réseau de capteurs sans fil qui peut être la technologie nécessaire pour alimenter l'Internet des objets. [8]

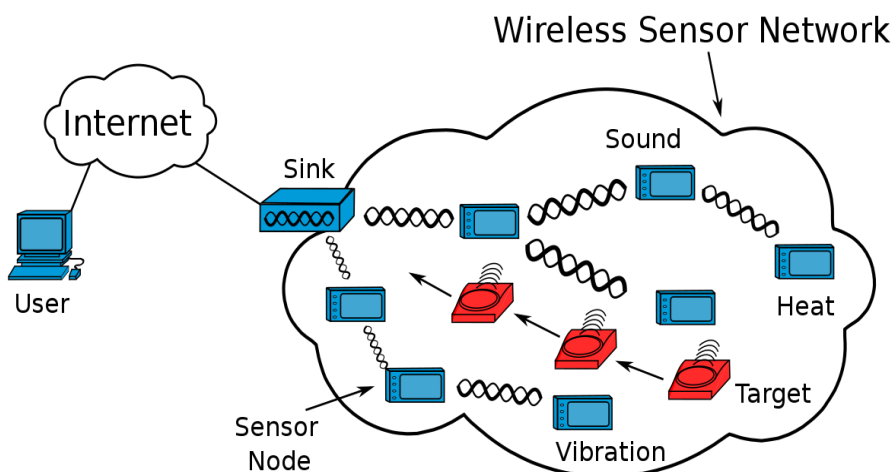


Figure I. 4 : Le fonctionnement de la technologie WSN. [9]

I.4.3 Machine to Machine (M2M)

Il permet d'associer des technologies de l'information et de la communication avec des objets intelligents pour qu'ils puissent interagir sans intervention humaine dans le système d'information. [10]

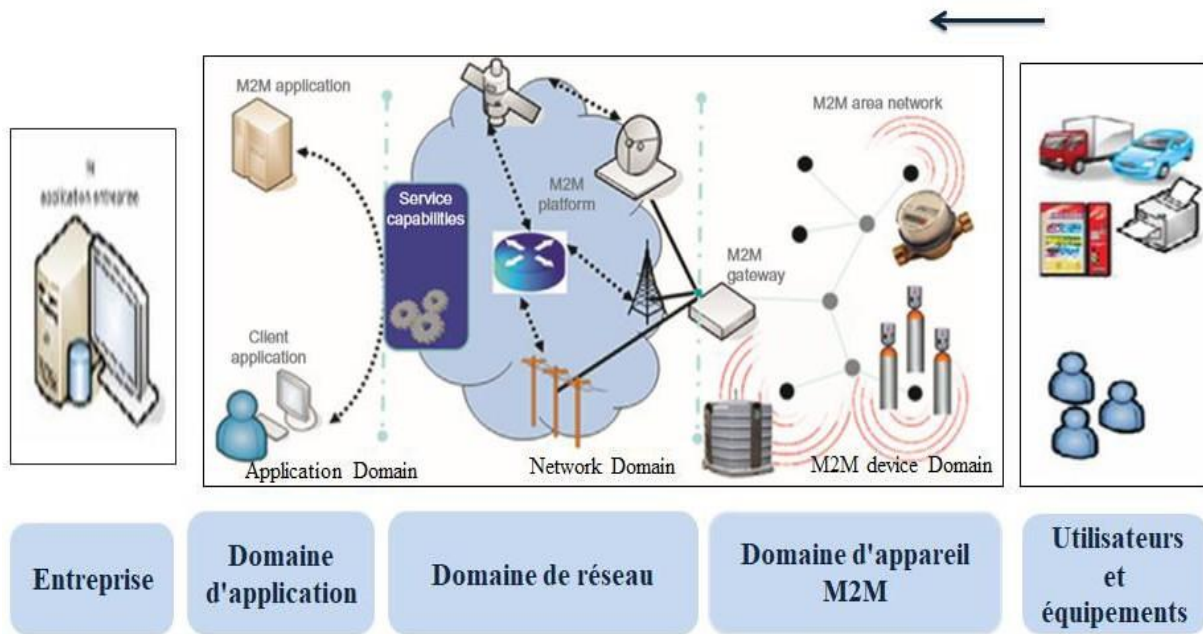


Figure I. 5 : Le fonctionnement de la technologie M2M.

I.5. Domaines d'application de l'IoT

Il existe plusieurs applications de l'Internet des objets. Ces applications couvrent de nombreux secteurs, notamment l'automobile, les télécommunications, l'énergie, etc...

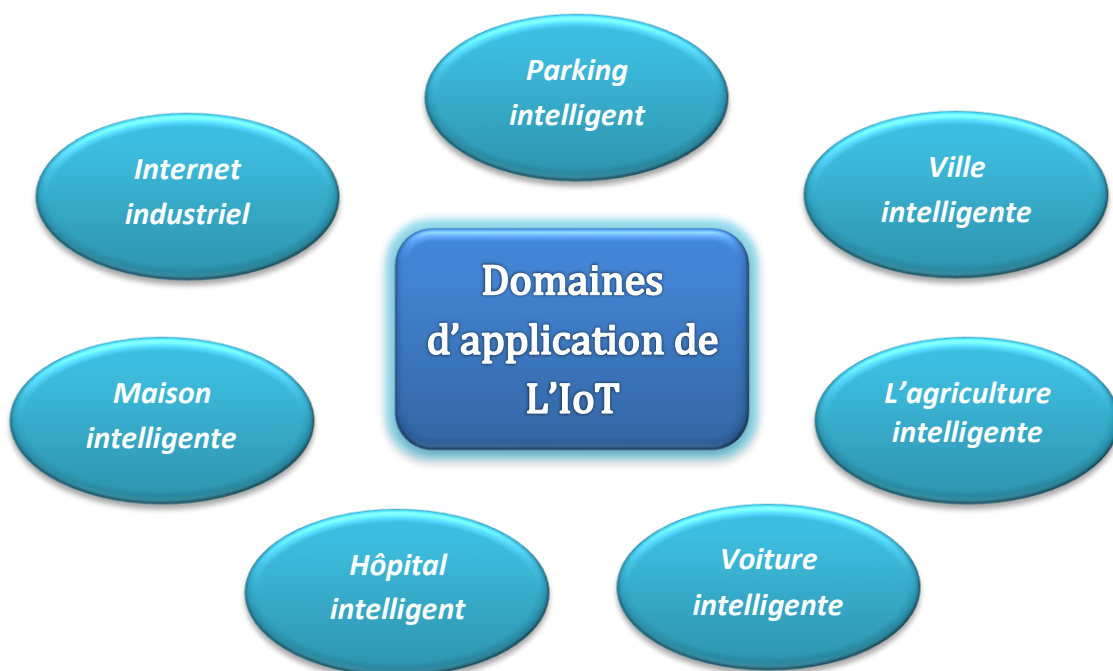


Figure I. 6 : Les Domaines d'application de l'IoT.

I.5.3 Hôpital intelligent (Santé connectée)

L'Internet des objets offre aux hôpitaux de grandes opportunités pour améliorer leurs services. L'objectif principal est d'améliorer la sécurité et le confort des patients, de créer un bon environnement de travail pour les employés et de gérer le stock de médicaments et de dispositifs médicaux tels que des ceintures de fitness pour surveiller le cœur. [13]

L'IoT a diverses applications dans le domaine des soins de santé, qui vont des équipements de surveillance à distance aux capteurs avancés et intelligents, en passant par l'intégration des équipements. Elle a le potentiel d'améliorer la manière dont les médecins dispensent les soins et de préserver la sécurité et la santé des patients. [11]



Figure I. 9 : Hôpitaux intelligente. [15]

I.5.4 Voiture intelligente

La technologie de la voiture connectée est un réseau vaste et étendu de multiples capteurs, antennes, logiciels intégrés et technologies qui aident à la communication pour naviguer dans notre monde complexe. Elle a la responsabilité de prendre des décisions avec cohérence, précision et rapidité. [11]

Il doit également être fiable. Ces exigences deviendront encore plus critiques lorsque les humains abandonneront entièrement le contrôle du volant et des freins aux véhicules autonomes ou automatisés qui sont actuellement testés avec succès sur nos autoroutes. [11]



Figure I. 10 : Voiture intelligent. [16]

I.5.5 L'agriculture intelligente

L'agriculture intelligente est une application de l'IoT souvent négligée. Pourtant, en raison du nombre d'opérations agricoles généralement éloignées et du grand nombre de bétail sur lequel les agriculteurs travaillent, tout cela peut être surveillé, et contrôlé par l'Internet des objets et peut également améliorer la façon dont les agriculteurs travaillent. [11]

Mais cette idée n'a pas encore retenu l'attention à grande échelle. Néanmoins, elle reste l'une des applications de l'IoT a développé. [11]



Figure I. 11 : L'agriculture intelligente. [17]

I.5.6 Parking intelligent

Le parking intelligent est une solution de stationnement qui peut comprendre des capteurs intégrés, des caméras et des capteurs de comptage. Ces dispositifs sont généralement intégrés aux places de stationnement ou positionnés à côté d'elles pour détecter si les places de stationnement sont libres ou occupées. Cela se fait par la collecte d'information en temps réel. Ces derniers sont ensuite transmis à une application mobile ou un site web de stationnement intelligent, qui communique la disponibilité aux utilisateurs. Dans ce contexte plusieurs entreprises proposent également d'autres options, comme les prix et les emplacements de stationnement. [18]

Ces systèmes à base de la technologie IoT représentent l'une des atouts des villes intelligentes. [18]



Figure I. 12 : Parking intelligent. [19]

I.5.7 Ville intelligente

Une ville intelligente est une ville qui utilise les technologies de l'information et de la communication (TIC) pour améliorer la qualité des services publics ou réduire leurs frais. [20]

Comme son nom l'indique, une ville intelligente est une innovation très vaste, qui couvre un large éventail de cas d'utilisation, de la distribution d'eau à la gestion du trafic, en passant par la gestion des déchets, la surveillance de l'environnement et la sécurité urbaine. [11]

La raison pour laquelle elle est si populaire est qu'elle tente d'éliminer l'inconfort et les problèmes des personnes vivant dans les villes. Les solutions IoT fournies par le domaine des villes intelligentes permettent de résoudre divers problèmes liés aux villes, notamment la circulation, de réduire la pollution atmosphérique et sonore et de rendre les villes plus sûres.

[11]



Figure I. 13 : Ville intelligente. [21]

I.6. L'avenir de l'internet des objets

Les analystes prévoient que, dans les prochaines années, l'internet des objets s'étendra pour conquérir presque tous les domaines de la vie quotidienne, du travail aux loisirs. De nombreuses applications de l'IoT sont actuellement à l'étude dans les domaines les plus disparates, de l'élevage à la médecine. L'IoT se développe également en parallèle avec l'utilisation de l'intelligence artificielle. Par exemple, on étudie de nouveaux systèmes qui intègrent des machines pour les examens cliniques capables d'envoyer automatiquement les résultats des analyses à des bases de données où l'intelligence artificielle, en analysant de grandes masses de données, est capable d'extrapoler un diagnostic et de recommander un traitement. L'intelligence artificielle, le big data et l'IoT sont connectés les uns aux autres à un niveau profond et le seront de plus en plus dans les applications futures. Des milliards de dispositifs et d'objets divers connectés à l'internet transmettent une immense masse de données qu'un esprit humain ne saurait analyser ou gérer. L'utilisation de l'intelligence artificielle pourrait au contraire trouver de nombreux nouveaux domaines d'application, en

déTECTANT par exemple les similitudes et les différences en analysant des millions de dossiers médicaux, à l'instar de ce qui se fait dans le domaine médical, de patients présentant la même pathologie. [22]



Figure I. 14 : L'avenir de l'internet des objets. [23]

I.7. Les avantages et les inconvénients de IoT

La connexion et l'échange de données avec d'autres appareils et des systèmes par Internet, ou simplement l'Internet des objets, possède des avantages et des inconvénients, par exemple :

I. 7.1. Les avantages

- Connexion à un réseau de dispositifs qui communiquent entre eux sans intervention humaine.
- Réduire le temps perdu dans les transactions administratives et améliorer les services publics.
- Proposer des solutions efficaces aux problèmes de surveillance et de protection des habitations, des entreprises et des autres bâtiments.

I. 7.2. Les inconvénients

Bien que l'Internet des objets présente de multiples avantages qui facilitent grandement le quotidien de beaucoup de personnes, que ce soit dans un cadre privé ou professionnel. Cependant, l'utilisation excessive de cette technologie révolutionnaire comporte de nombreux risques. Car avec l'Iot, de grandes quantités de données sont stockées et transmises, et donc que les risques liés à la protection des données demeurent.

La protection des données et la sécurité de l'utilisateur et de l'entreprise doivent donc être primordiales.

I.8. Conclusion

L'internet des Objets « Internet of Things » est devenu l'un des phénomènes les plus discutés aujourd'hui. L'IoT est un domaine d'investissement rentable car il est la technologie fait partie de l'internet du futur. L'internet des objets rend le tissu du monde qui nous entoure plus intelligent ce qui engendre autant de craintes qu'elle est porteuse de promesses.

Chapitre II

**Description des différents composants
utilisés pour la réalisation du projet**

II.1 Introduction

Les composants électroniques sont aujourd'hui une partie intégrante de notre vie. En fait, le but de ces composants est de connecter à d'autres appareils pour réaliser une fonction électronique afin de rendre notre vie plus facile et plus pratique.

Dans ce chapitre, nous présentons les différents composants électroniques programmables nécessaires pour la réalisation de notre projet.

II.2 Les composants électroniques utilisés

Pour réaliser notre prototype, nous avons utilisés plusieurs composants.

Ces dispositifs sont indiqués dans le tableau II. 1 :

Articles	Quantité
Arduino Méga2560	1
NodeMCU/ESP8266	1
Servomoteur	3
Capteur IR	8
Capteur de flamme	1
Afficheur LCD 16*2 - I2c	1
Buzzer	1
RFID-RC522	1

Tableau II. 1 : Composants électroniques utilisés.

II.3 La carte Arduino

II.3.1 Les différentes cartes Arduino

Il existe plusieurs types de carte Arduino (DUE, Méga, UNO, Ethernet, Leonardo, Robot Micro, YUN, NANO, MINI et etc...)

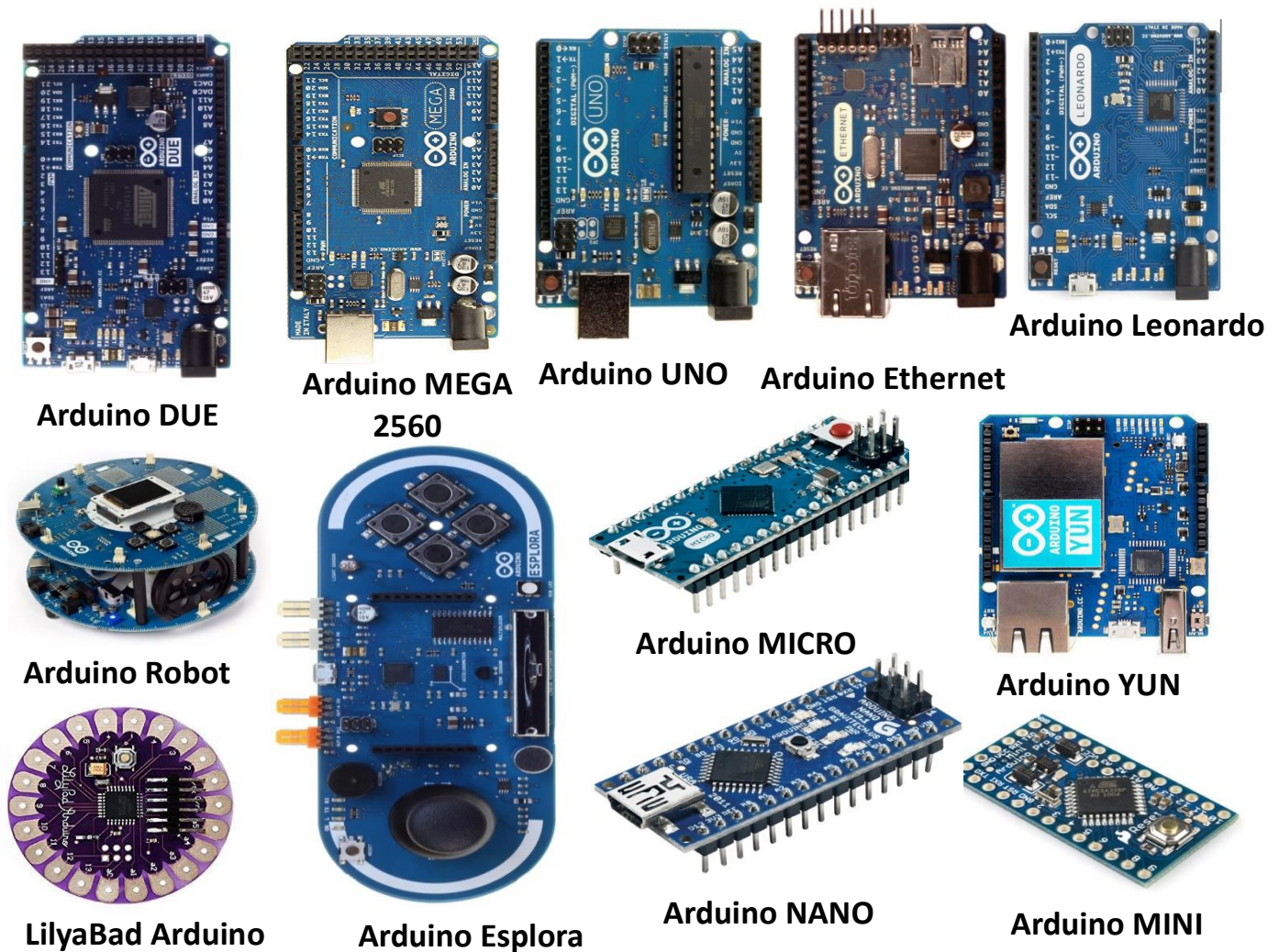


Figure II. 1 : Les différentes cartes Arduino.

II. 3.2 Aperçu sur Arduino Méga 2560

Arduino Méga 2560 est une carte électronique programmable basée sur microcontrôleur ATmega2560, avec des broches d'entrée / sortie numériques et analogiques, un oscillateur 16 MHz, une Connexion USB, un port ICSP et un bouton de réinitialisation. [24]

Afin d'alimenter la carte Arduino Méga, il suffit d'utiliser un câble USB pour connecter le module avec l'ordinateur. En outre, il est possible d'utiliser une source de tension externe telles que (une batterie ou un adaptateur AC-DC « 12V »). [24]

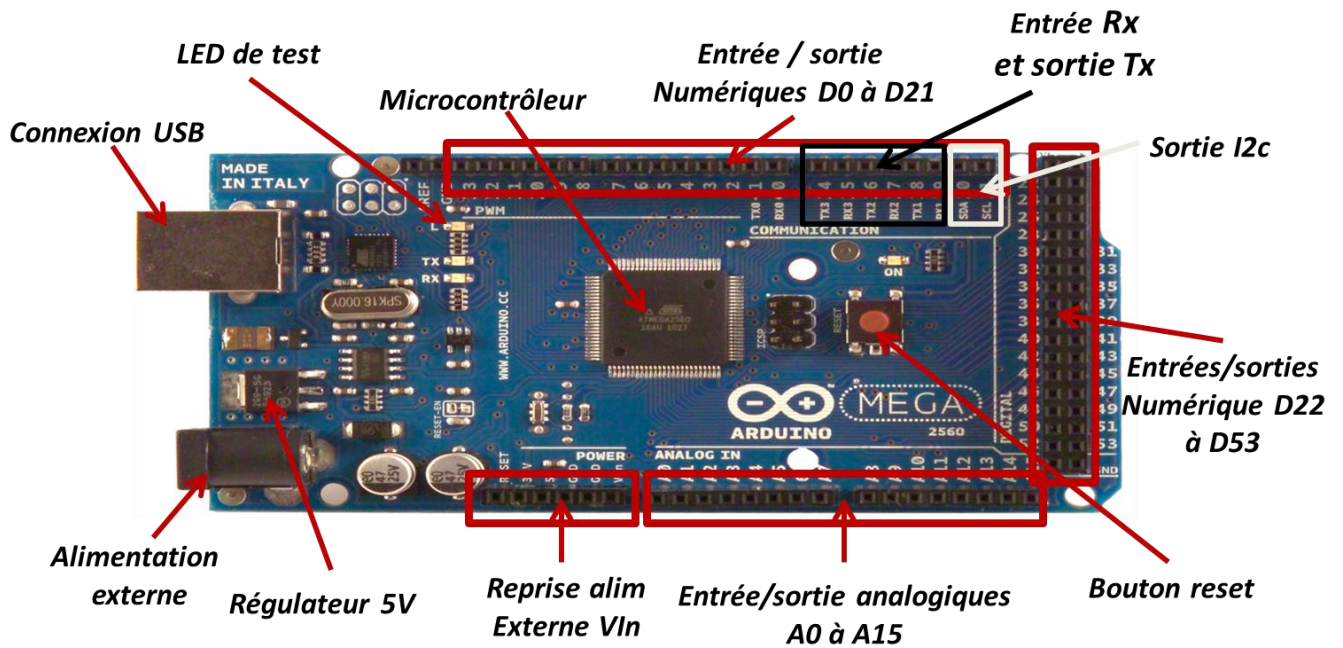


Figure II. 2 : Une carte Arduino Méga2560.

II. 3.3 Caractéristiques de l'Arduino Mega2560

La carte Arduino Méga est omniprésente dans de nombreux projets. Ceci est dû à ses caractéristiques intéressantes.

Les caractéristiques de la carte Arduino Méga sont représentées dans le tableau II.2 :

Spécifications techniques	
Microcontrôleur	ATmega2560
Tension de fonctionnement	5 V
Tension d'entrée (recommandé)	7 à 12 V
Tension d'entrée (limites)	6 à 20 V
Broches d'E/S numériques	54 (dont 14 fournissent une sortie PWM)
Broches d'entrée analogique	16
Courant continu par broche E/S	40 mA
Courant continu pour la broche 3.3V	50 mA
I2Cs	1
Mémoire Flash	256 KB (8 KB utilisés par le boot loader)
Vitesse d'horloge	16 MHz
EEPROM	4 KB
SRAM	8 KB

Tableau II. 2 : Caractéristiques de l'Arduino Mega2560. [24]

II. 3.4 Configuration du brochage de l'Arduino Méga 2560

II. 3.4.1 Alimentation

- **Vin** : La tension d'entrée positive lorsque la carte Arduino est utilisée avec une source de tension externe (qui se distingue de la tension 5V connectée à l'USB ou à d'autres alimentations régulées 5V). [24]
- **5V** : L'alimentation régulée utilisée pour alimenter les composants de la carte. Elle peut provenir soit du VIN, soit être fournie par USB. [24]
- **3V3** : Une tension régulée de 3.3 V générée par le régulateur embarqué. L'appel de courant maximal est de 50 mA. [24]
- **GND** : Broche de masse (ou 0V). [24]

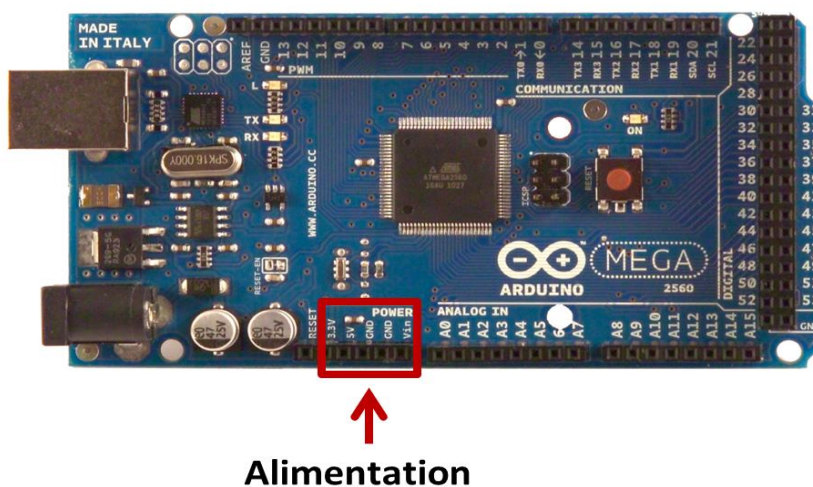


Figure II. 3 : Alimentation de l'Arduino Méga2560.

II. 3.4.2 Broches d'E/S numériques

- **Série** : (TX) et (RX). Utilisé pour transmettre (TX) et recevoir (RX) des données série TTL. [24]
- **Interruptions externes** : Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, un front montant ou descendant, ou un changement de valeur. [24]
- **PWM** : 0 à 13 Offre une sortie PWM de 8 bits. [24]
- **SPI** : (MISO), (MOSI), (SCK), (SS). Ces broches disponibles pour la communication SPI. [24]
- **LED** : LED de tests connectés à la broche numérique 13. [24]
- **I2C** : (SDA) et (SCL). [24]

Broches d'E/S numériques

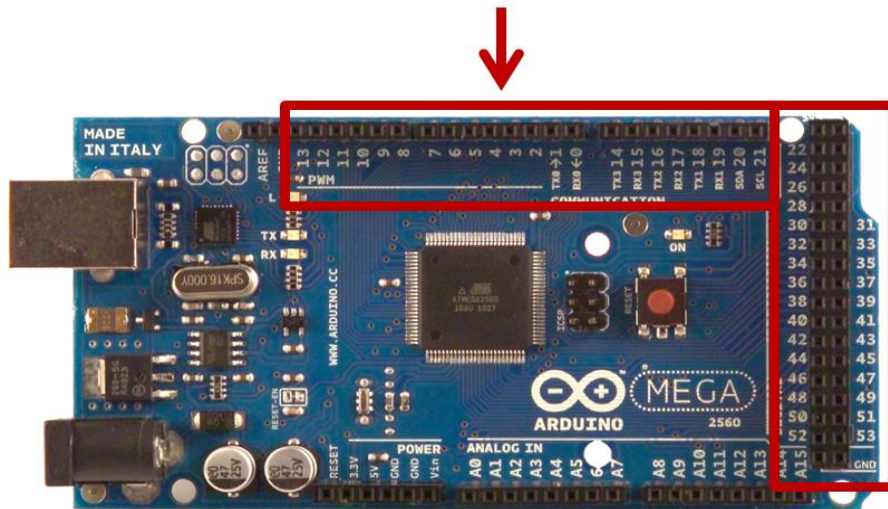
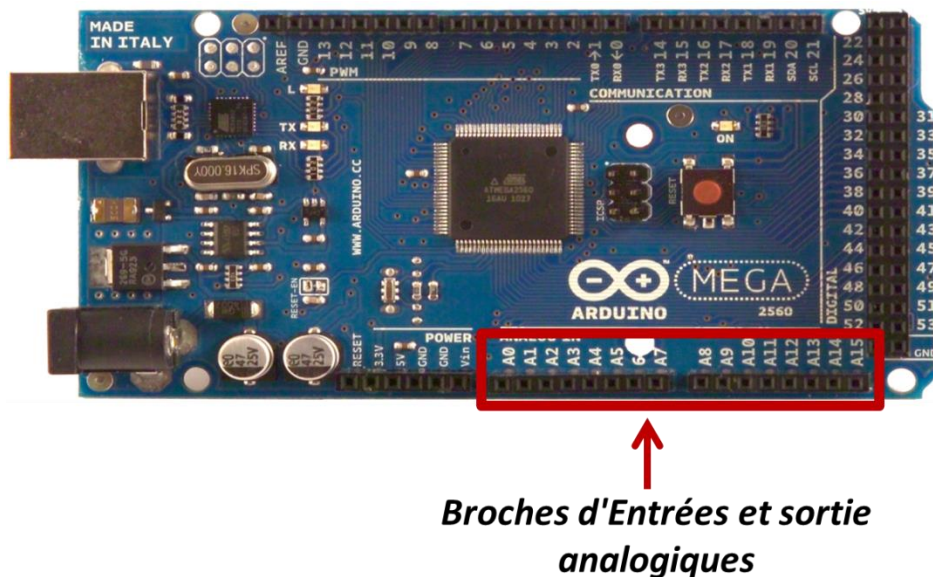


Figure II. 4 : Broches d'E/S numériques de l'Arduino Méga2560.

II. 3.4.3 Broches d'Entrées et sortie analogiques

La carte dispose de 16 entrées et sortie analogiques pour cela, étiquetées A0 à A15.



Broches d'Entrées et sortie analogiques

Figure II. 5 : Broches d'E/S analogiques de l'Arduino Méga2560.

II. 3.4.4 Autres broches

- **AREF** : Tension de référence pour les entrées analogiques (si différente de 5V). [12]
- **Reset** : En plaçant cette broche au niveau BAS, le microcontrôleur est réinitialisé (redémarré). [24]

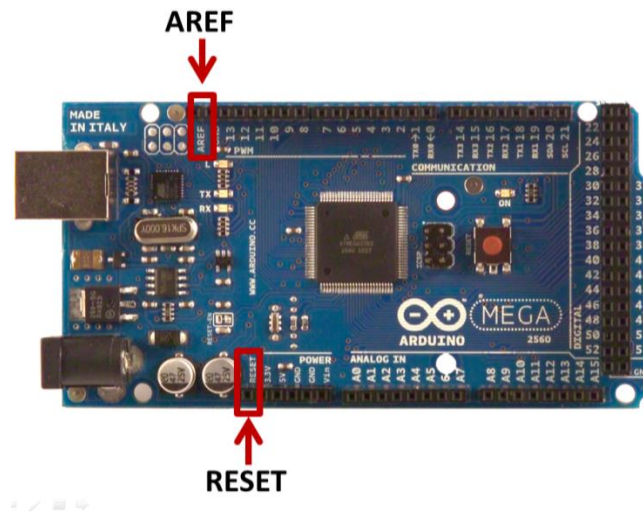


Figure II. 6 : Broches RESET / AREF de l'Arduino Méga2560.

II. 3.5 Microcontrôleur ATmega2560

Le microcontrôleur est une composante très importante dans la carte Arduino. La puce qui équipe la carte Arduino Méga est une « microcontrôleur ATmega2560 ». C'est un microcontrôleur d'une famille AVR® RISC de Microchip 8 bits, dont la programmation peut être réalisée en langage C/C++. [25]

Le microcontrôleur ATmega2560 embarque un processeur central pour l'exécution du programme stocker dans une mémoire Flash ISP de 256 Ko, une SRAM de 8 Ko, une EEPROM de 4 Ko pour enregistrer les informations et conserver les données. Ainsi qu'il possède des 86 lignes d'E/S générales, 32 registres de travail généraux, un Le dispositif atteint un débit de 16 MIPS à 16 MHz et fonctionne entre 4,5 et 5,5 volts. [25]

Ces broches sont utilisées pour connecter le microcontrôleur avec différents composants électroniques. [25]

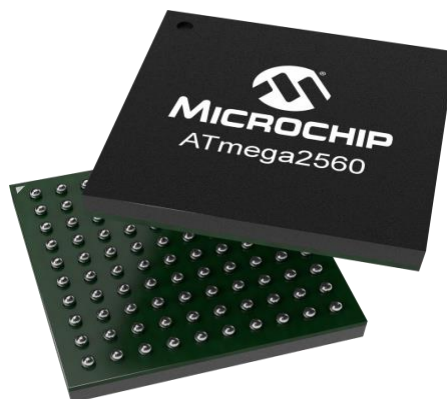


Figure II. 7 : Microcontrôleur ATmega2560. [25]

II. 3.6 L'environnement de développement (IDE) de l'Arduino

Arduino (IDE) est un logiciel pour écrire des programmes et de les télécharger sur votre carte. Pour commencer à utiliser l'Arduino, vous trouverez deux options :

- Si vous disposez d'une connexion Internet fiable, vous devez utiliser l'IDE en ligne (Arduino Web Editor). [26]
- Si vous souhaitez travailler hors ligne, vous devez utiliser la dernière version de l'EDI de bureau. [26]

Arduino (IDE) est un environnement de développement intégré de Microsoft. Même si c'est un IDE Microsoft, vous pouvez l'utiliser sur les ordinateurs Windows, Mac et Linux.

Ce logiciel, Il prend en charge différents langages de programmation, dont C, C++, etc. Ainsi que, Le code que vous écrivez ne requiert que deux fonctions de base :

- La fonction setup () : fonction exécutée à l'initialisation de la carte.
- La fonction loop () : fonction exécutée en boucle infinie.

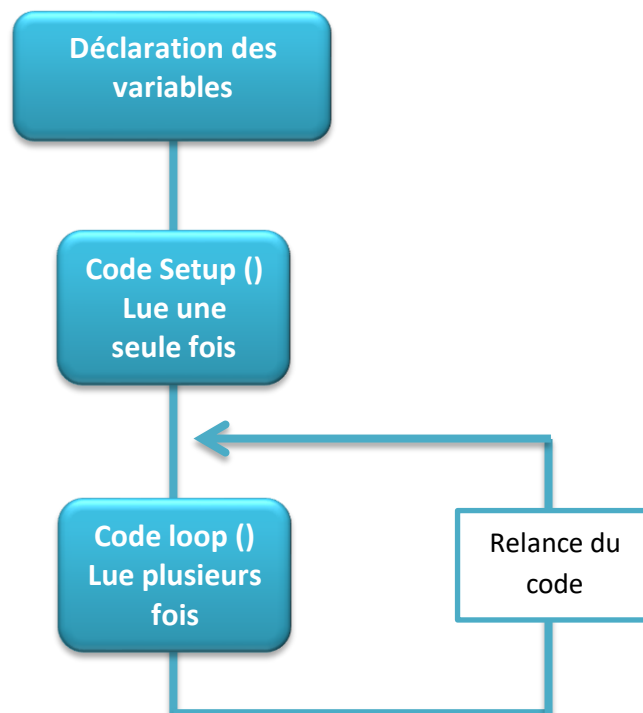


Figure II. 8 : Fonctionnement d'un programme Arduino.

II.4 La carte WI-FI « NodeMCU »

Il existe deux différents de carte WI-FI « NodeMCU » :

- NodeMCU-ESP32
- NodeMCU-ESP8266

Dans notre projet une utilisée NodeMCU-ESP8266.

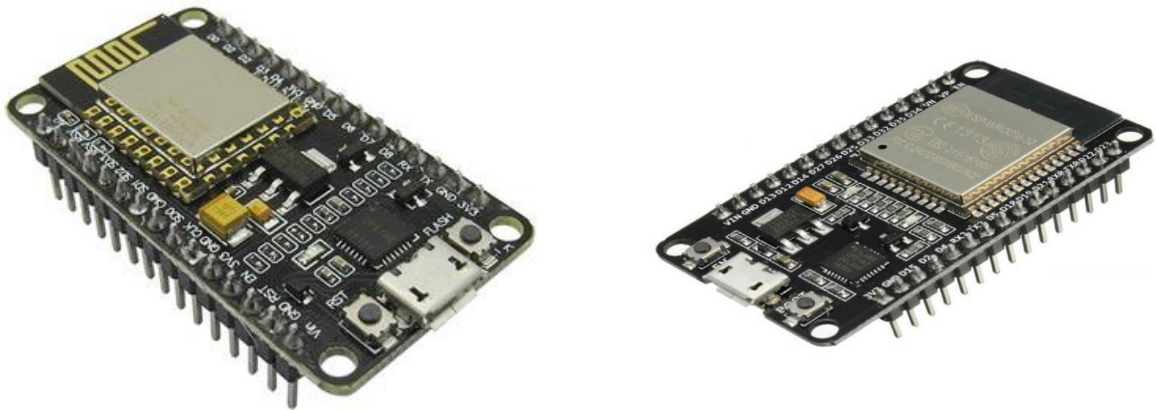


Figure II. 9 : La carte WI-FI « NodeMCU-ESP8266/ESP32 ».

II.4.1 Description de la carte NodeMCU / ESP8266

Le NodeMCU est un environnement de développement logiciel et matériel open-source construit autour d'un système sur puce (SoC) peu coûteux appelé ESP8266, développée par Expressif Systèmes avec le protocole TCP/IP. [27]

L'ESP8266 intègre un émetteur-récepteur Wi-Fi 802.11b/g/n HT40, ce qui lui permet non seulement de se connecter à un réseau Wi-Fi et d'interagir avec Internet, mais aussi de créer son propre réseau. Par conséquent, L'ESP8266 fournit une fonctionnalité d'Internet des objets (IOT). [27]

Cette carte est composée de tous les éléments importants d'un ordinateur moderne : CPU, RAM, réseau (sans fil), et même un système d'exploitation moderne et un SDK. Ainsi elle dispose de 17 broches GPIO réparties sur les têtes de broche des deux côtés de la carte de développement. [27]

Pour fonctionner, la carte esp8266 a besoin d'une alimentation électrique. Elle peut être alimentée soit par un ordinateur via une connexion USB (3.3V), soit par une alimentation externe qui fournit une tension continue comprise entre 3.3V et 10V. Ensuite, un régulateur se charge d'abaisser la tension à 3.3V pour le bon fonctionnement de la carte. [27]

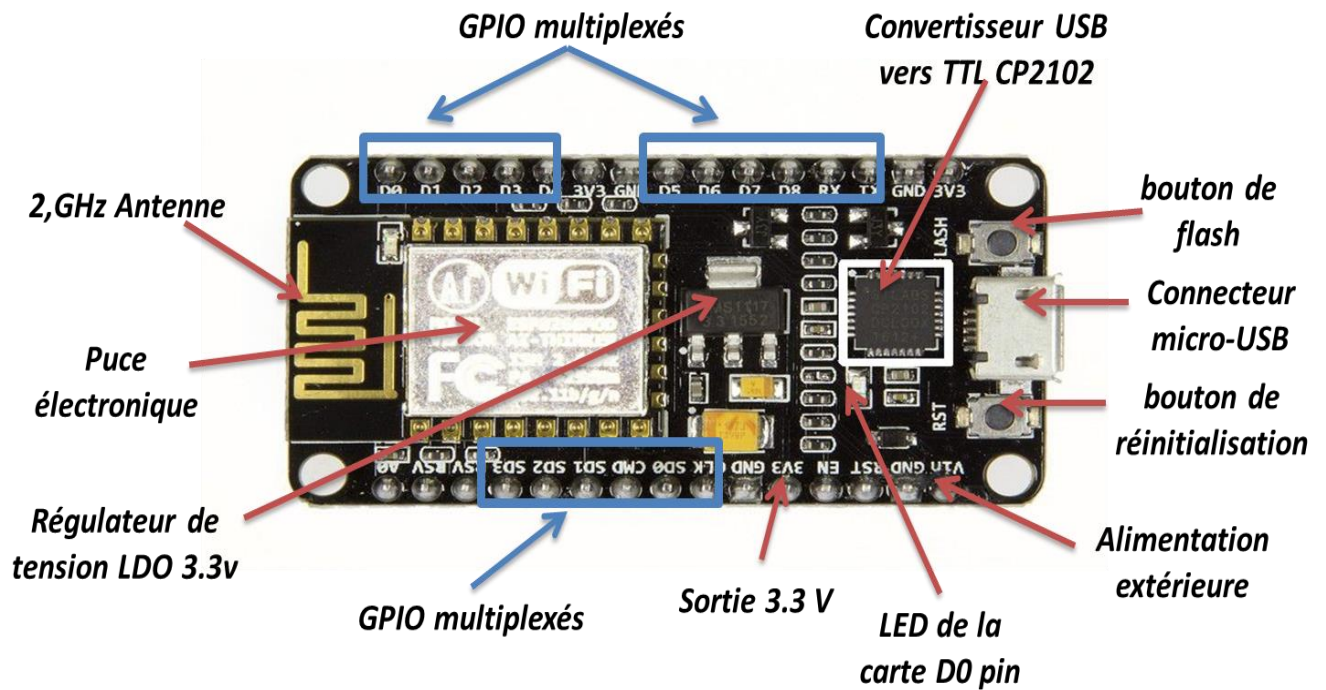


Figure II. 10 : Une carte WI-FI ESP8266.

II .4.2 Caractéristiques de NodeMCU

NodeMCU est une carte de développement à la base de l'ESP8266. Cette carte se distingue par ses caractéristiques techniques.

Ces caractéristiques sont présentées dans le tableau ci-dessous.

Spécifications techniques	
Microcontrôleur	Tensilica Xtensa® 32-bit LX106
Wi-Fi	802.11b/g/n
Tension de fonctionnement	3,3 V
GPIO Pins	Broches d'E/S numériques : 11 Broches pour connecter la puce de mémoire flash : 6
Broches d'entrée analogique	1
I2Cs	1
Mémoire Flash	4MB
Vitesse d'horloge	80 à 160 MHz
SRAM	128 KB
Plage de températures	-40C / 125C

Tableau II. 3 : Caractéristiques de carte WI-FI ESP8266. [27]

II .4.3 Configuration du brochage de NodeMCU

Le NodeMCU ESP8266 dispose d'un total de 30 broches qui l'interface avec le monde extérieur. La connexion de ces broches est expliquée dans le tableau II. 4 :

Catégorie de broche	Nom	Description
Alimentation	Micro-USB, 3.3V, GND, Vin	<p>Micro-USB : NodeMCU peut être alimenté par le port USB.</p> <p>3.3V : Une tension régulée de 3.3V peut être fournie à cette broche pour alimenter la carte.</p> <p>GND : Broche de masse (ou 0V).</p> <p>Vin : Alimentation externe</p>
Broches de contrôle	EN, RST	La broche et le bouton réinitialise le microcontrôleur
Broche analogique	A0	Utilisé pour mesurer la tension analogique dans la gamme de 0-3.3V
Broches GPIO	GPIO 0 à GPIO 16	NodeMCU a 17 broches d'entrée-sortie d'usage général sur sa carte. Et (GPIO 6 - 11) sont utilisées pour connecter la puce de mémoire flash.
Broches SPI	SD1, CMD, SD0, CLK	NodeMCU a quatre broches disponibles pour la communication SPI.
Broches UART	TXD0, RXD0, TXD2, RXD2	NodeMCU possède deux interfaces UART, UART0 (RXD0 & TXD0) et UART1 (RXD1 & TXD1). UART1 est utilisé pour télécharger le firmware/programme.
Broches I2C		Le NodeMCU supporte la fonctionnalité I2C mais en raison de la fonctionnalité interne de ces broches, vous devez trouver quelle broche est I2C.

Tableau II. 4 : Configuration du brochage de carte WI-FI ESP8266. [28]

II .4.5 BLYNK

Blynk est plateforme spécialement conçue pour l'Internet des objets. Il peut accomplir plusieurs tâches telles que le contrôler à distance du matériel, affichage des données du capteur et stockages des informations. [30]

II .4.5.1 Description du plateforme BLYNK

Blynk est une plate-forme avec des applications IOS et Android qui peuvent contrôler Arduino, ESP8266 et Raspberry Pi et d'autres applications sur Internet. [30]

Il s'agit d'un tableau de bord numérique dans lequel vous pouvez créer une interface graphique pour le projet. Pour ce faire, il faut passer par quelques étapes :

La première étape est de télécharger l'application Blynk. Une fois cette application téléchargée, il faut créer un nouveau compte Blynk. Pour cela, utilisez votre adresse e-mail personnelle. [30]

Par la suite, commencez par créer un nouveau projet. Pour faire, il faut sélectionner le modèle de matériel que vous allez utiliser. [30]

Après la création d'un projet, vous recevrez automatiquement un jeton d'authentification. À l'étape suivante, ajoutez et configurez des Widgets. [30]

Une fois terminé, appuyez sur le bouton PLAY. [30]

II .4.5.2 Comment fonctionne BLYNK ?

Il y a trois composants principaux dans la plateforme :

- **Application Blynk** : vous permet de créer une excellente interface pour votre projet en utilisant les différents widgets que nous fournissons. [30]
- **Serveur Blynk** : responsable de toutes les communications entre le Smartphone et le hardware. Vous pouvez utiliser notre Blynk Cloud ou exécuter un serveur privé Blynk localement. Il est open source, peut facilement gérer des milliers d'appareils et peut même être lancé sur un Raspberry Pi. [30]
- **Bibliothèque Blynk** : adaptée à toutes les plateformes hardware populaires, prend en charge la communication avec le serveur et traite toutes les commandes entrantes et sortantes. [30]

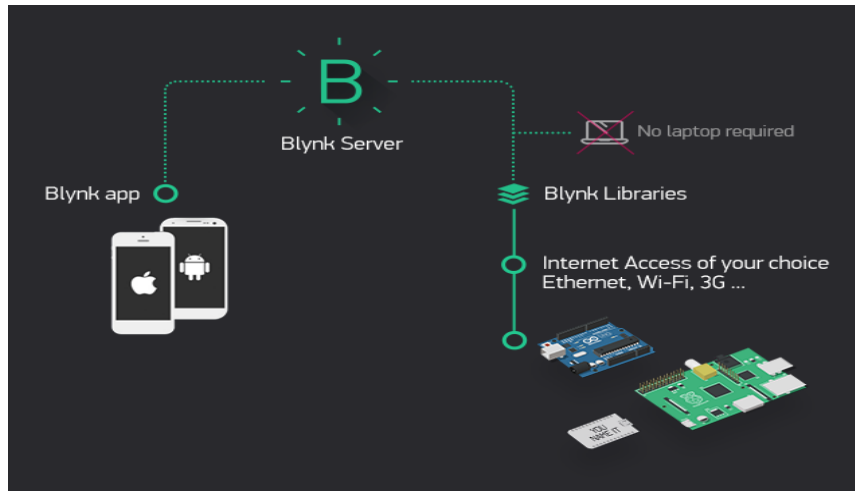


Figure II. 13 : Fonctionne Blynk. [30]

II.5 Servomoteur

Un servomoteur (vient du latin servus qui signifie « esclave »)



Figure II. 14 : Servomoteur "9 grammes".

II.5.1 Description de servomoteur

Le servomoteur est un système relativement complexe qui intègre des composants électroniques et électromécaniques.

Ce dispositif est composé d'un moteur à courant continu à deux fils, d'un train d'engrenages, d'un potentiomètre, d'un circuit intégré et d'un arbre de sortie. De plus, ils sont connectés trois fils à ce dispositif : un fil de terre (0V), un fil de tension d'entrée (généralement 4,5V à 6V) et un fil de commande. [31]

À l'aide du fil de commande envoie un signal à ce servomoteur, qui ordonne à l'arbre de tourner dans une position angulaire spécifique (0°-180°).

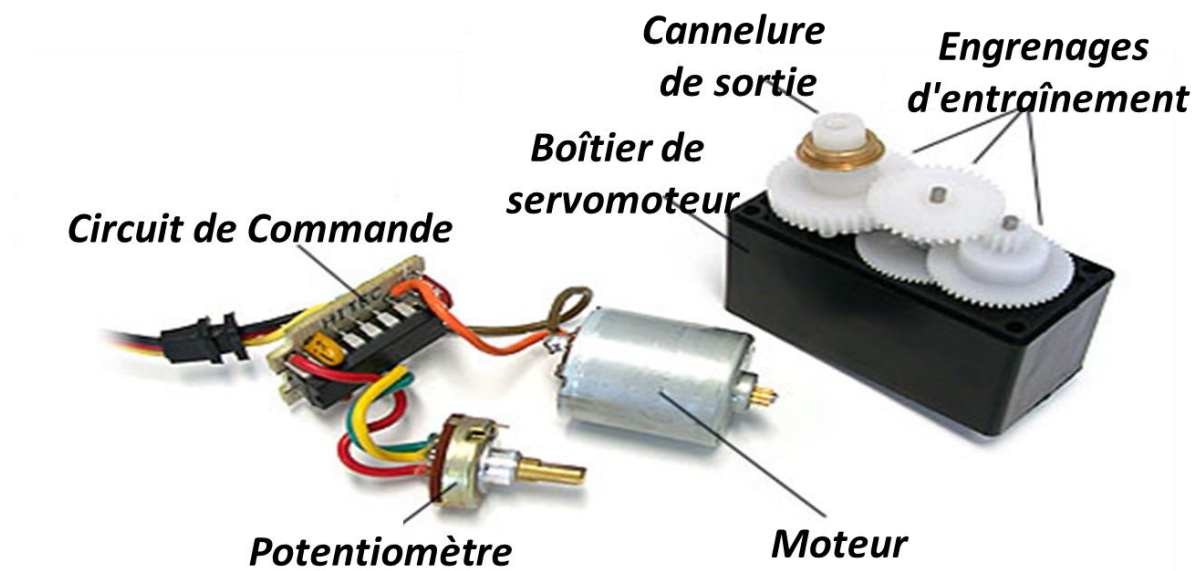


Figure II. 15 : Structure de Servomoteur "9 grammes". [32]

II.5.2 Programmation de servomoteur

Afin de faire bouger le servomoteur, nous devons utiliser une bibliothèque de codes appelée "Servo". Il s'agit de la configuration standard de l'environnement de développement Arduino (IDE). [33]

La bibliothèque de servomoteur peut contrôler jusqu'à 48 servomoteurs en même temps via la carte Arduino Méga, Il suffit d'importer la bibliothèque en ajoutant cette ligne en début de programme pour l'utiliser : « #include <Servo.h> ». [33]

En effet, afin de créer un objet Servo, il faut simplement déclarer une variable (globale) de type " Servo ", exemple : " monServo ", Si vous souhaitez utiliser plusieurs servomoteurs en même temps, vous devez créer plusieurs variables, une pour chaque servomoteur. [33]

II.6 Capteur IR (FC-15)

Les capteurs sont un composant important. Ces derniers peuvent être utilisés pour détecter différents attributs du monde réel, de la distance à la chaleur en passant par la pression. Par conséquent, Les produits d'aujourd'hui utilisent des capteurs pour détecter tout ce qui les entoure.

En générale, Le capteur est un composant qui transforme une grandeur physique (température, pression, lumière, vitesse, etc.) en un signal de mesure électrique. De plus, il est très précis et consomme moins d'énergie.

Il existe plusieurs types de capteurs qui sont fréquemment utilisés dans la plupart des applications électroniques, tels que :

- Capteur de température
- Capteur IR
- Capteurs de proximité
- Capteurs de pression
- Capteurs de niveau
- Capteurs de fumée et de gaz
- Capteurs de flamme

Dans notre projet une utilisée deux modale de capteur : Capteur IR et Capteurs de flamme.

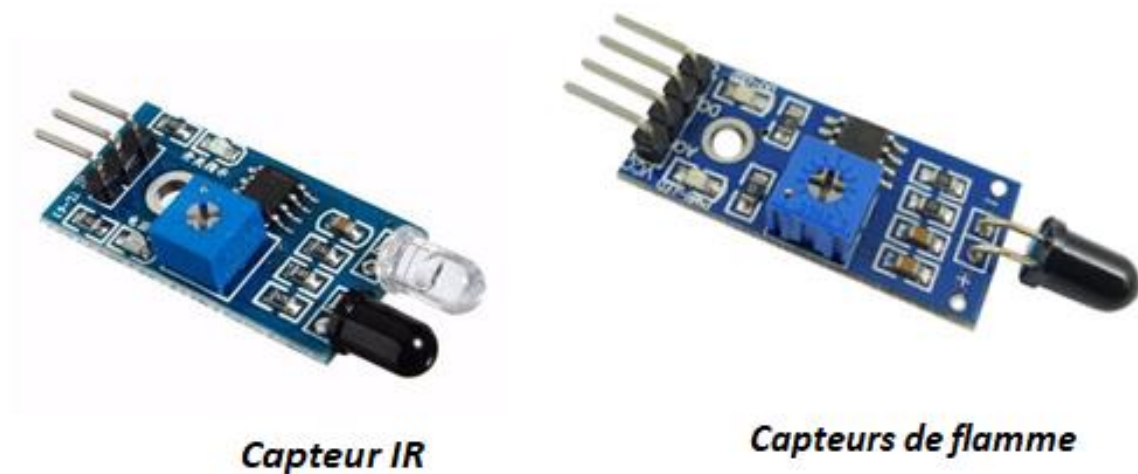


Figure II. 16 : Capteur IR et Capteurs de flamme.

Les capteurs infrarouges peuvent être passifs ou actifs.

Les capteurs infrarouges passifs sont principalement des détecteurs infrarouges qui n'utilisent pas de source infrarouge. [34]

Les types de capteurs infrarouges actifs sont composés de deux éléments : une source infrarouge et un détecteur infrarouge. [34]

Ce capteur classé dans le deuxième type, est composé d'une diode infrarouge (émetteur) et d'une photodiode (récepteur). En fonction de la distance de l'obstacle, le récepteur recevra plus ou moins de lumière infrarouge réfléchi. [34]

II .6.1 Description de capteur IR

Le capteur IR peut être alimenté par une tension entre 3 et 5 V DC, la présence de tension est indiquée par sa LED rouge. [35]

Ce module utilise une paire de diodes infrarouges (un émetteur et un récepteur) qui fonctionnent par réflexion. Il peut détecter un objet à une distance comprise entre 2 et 30 cm avec un angle de détection de 35°, cette distance peut être réglée par la résistance variable.

Ainsi la couleur influe également l'opération de détection. En effet, la distance de détection sera minimale si l'objet est noir, et maximale s'il est blanc. [35]

En fin, La stabilité du signal est garantie par l'utilisation d'un comparateur LM393. [35]

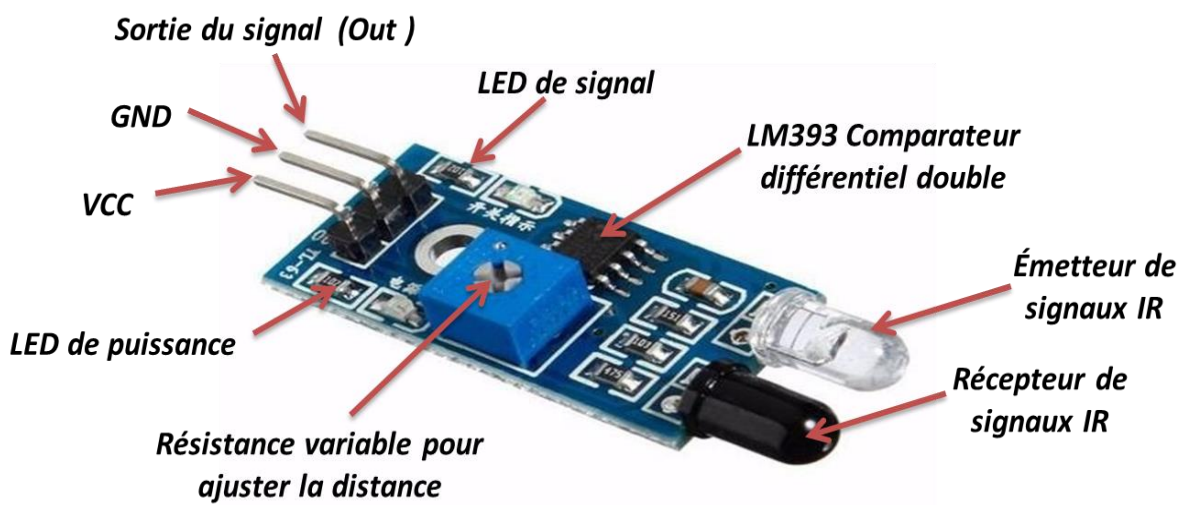


Figure II. 17 : Structure d'un Capteur IR.

Les broches de ce capteur sont présentées dans le tableau suivant :

Sortie et entrée du PIN	
Entrée du PIN	VCC : Entrée d'alimentation 3.3V-5V DC. GND : 0V Broche d'alimentation.
Sortie du PIN	OUT : Broche de sortie numérique

Tableau II. 5 : Sortie et entrée d'un Capteur IR.

II .6.2 Principe de fonctionnement

Lorsque Le capteur infrarouge envoie un rayonnement qui atteint le corps, il est réfléchi vers le récepteur infrarouge. [36]

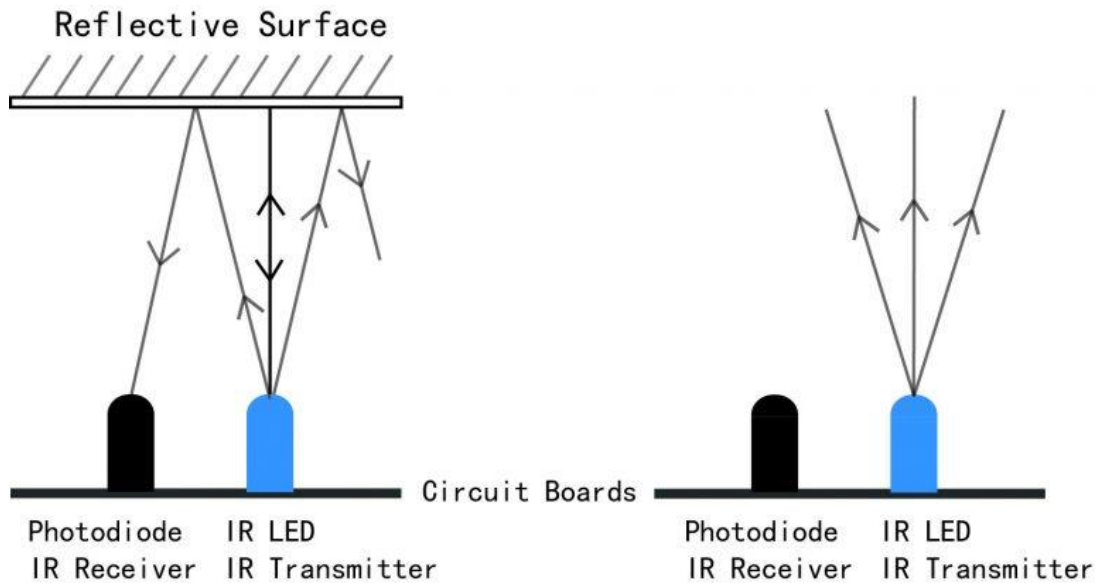


Figure II. 18 : Principe de fonctionnement d'un capteur IR. [35]

II .6.3 Émetteur IR et récepteur IR

a. Émetteur IR

Un émetteur infrarouge est une diode électroluminescente (LED) qui émet des radiations infrarouges. C'est pourquoi on les appelle des LED IR. Même si une LED IR ressemble à une LED normale, le rayonnement qu'elle émet est invisible pour l'œil humain. [34]



Figure II. 19 : Émetteur IR.

b. Récepteur IR

Les récepteurs infrarouges sont également appelés capteurs infrarouges car ils détectent le rayonnement d'un émetteur IR. Les récepteurs IR se présentent sous la forme de photodiodes et de phototransistors. Les photodiodes infrarouges sont différentes des photodiodes normales car elles ne détectent que le rayonnement infrarouge. L'image d'un récepteur IR typique ou d'une photodiode est présentée dans la figure II.21. [34]



Figure II. 20 : Récepteur IR.

II .6.4 Schéma bloc d'un capteur IR

Lorsque l'émetteur IR émet un rayonnement, ce dernier atteint l'objet et une partie du rayonnement est réfléchi vers le récepteur IR. Par conséquent, Le comparateur LM393 détecte la faible valeur de résistance de la photodiode et envoie un signal logique haut (5V) à la sortie de module (out). [36]

Dans le cas contraire, Le comparateur LM393 détecte la valeur de résistance haute de la photodiode et envoie un signal logique bas (0V) à la sortie du capteur (out). [36]

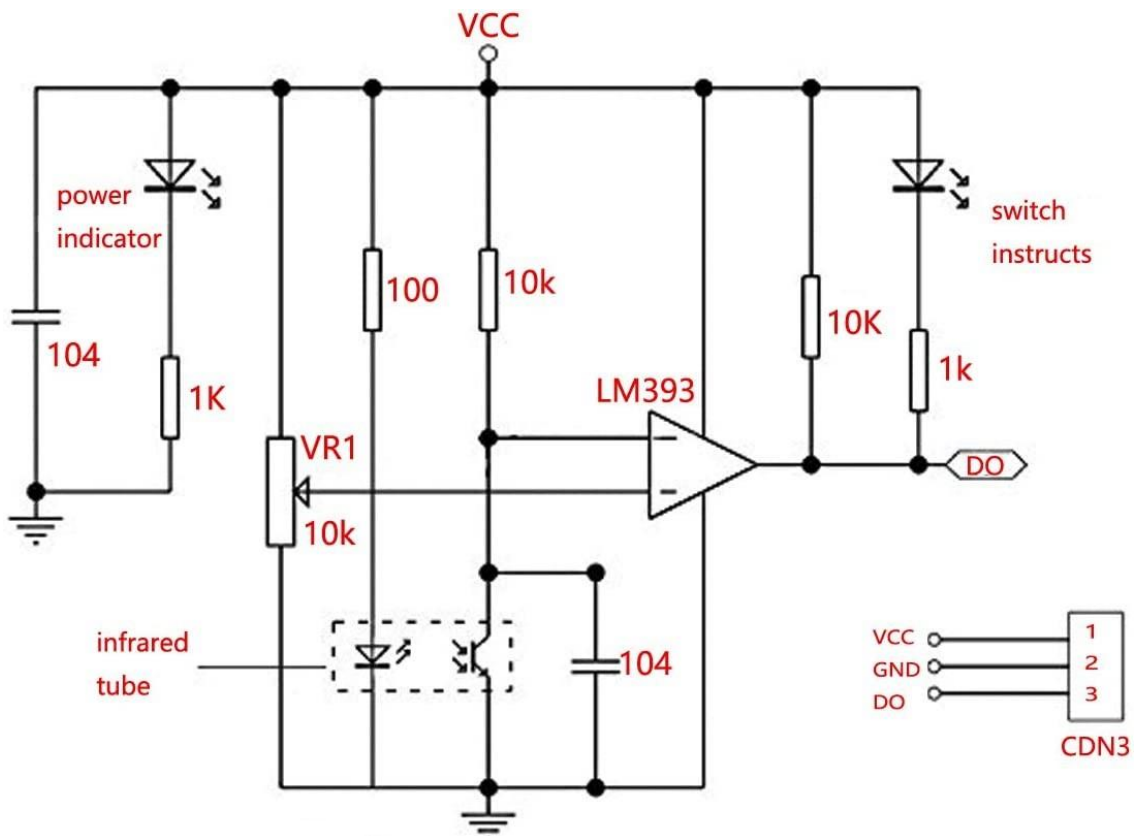


Figure II. 21 : Schéma bloc d'un capteur IR. [36]

II.7 Capteur de flamme

Les capteurs de flamme optiques réagissent au rayonnement émis par tous les types de flammes. Ils sont élaborés pour prévenir très rapidement (en quelques dizaines de millisecondes) via des signaux numériques.

II.7.1 Présentation de capteur de flamme IR

Le capteur de détection de flamme est le capteur le plus sensible aux longueurs d'onde infrarouges de la flamme. Ce capteur est composé de 3 ou 4 broches. La broche Vcc pour l'alimentation, GND de masse et deux sorties (Analogique A0 et Numérique D0).

Les capteurs à 4 broches et à 3 broches sont constitués des mêmes composants. Les principaux composants de ces capteurs sont un phototransistor NPN YG1006 (sensible aux sources de rayonnement dans la gamme de longueur d'onde de 760 nm à 1100 nm), un circuit intégré comparateur LM393, une résistance variable (pot de réglage), une LED d'alimentation et une LED de sortie. [37]

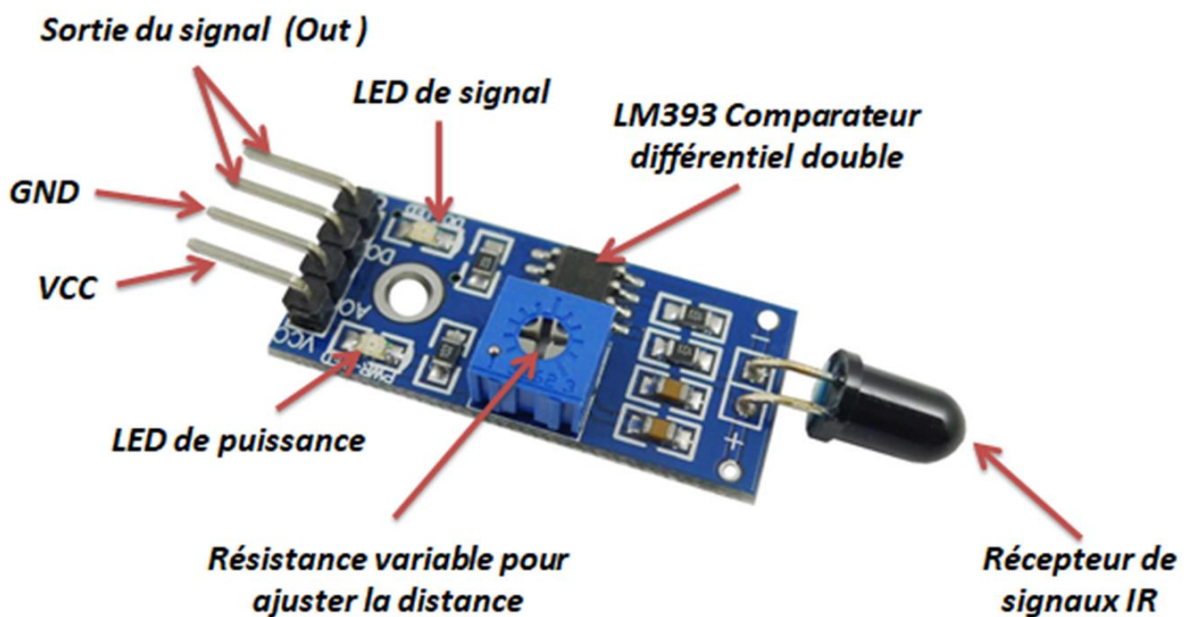


Figure II. 22 : Structure d'un Capteur de flamme.

II.7.2 Les différents types de capteur de flammes

Capteur de flamme UV : Il se compose d'un capteur sensible au rayonnement ultraviolet. Par rapport à une référence soigneusement calculée, un signal de sortie est généré, qui est fonction du rayonnement. [38]

Capteur de flamme IR : Il est composé d'un capteur sensible au rayonnement infrarouge produit par la combustion du dioxyde de carbone dans la flamme. [38]

Capteur de flamme UV-IR : Ce détecteur combine un capteur UV et un capteur IR pour la détection de feux d'hydrocarbures, d'hydrogène et de métaux avec temps de réponse très rapide.

Capteur de flamme IR3 : trois capteurs pyroélectriques détectent le rayonnement IR lié à différents pics d'émission. Le capteur IR3 utilisé pour détecter les flammes d'hydrocarbures n'est pas sensible au rayonnement solaire. [38]

Capteur de flamme IR4 : Combine 4 capteurs infrarouges pour détecter les flammes d'hydrocarbures et d'hydrogène dans une portée maximale de 5 à 65 mètres, tout en assurant un haut degré de résistance aux alarmes de harcèlement. [38]

II.7.3 Principe de fonctionnement de capteur de flamme

Dans un premier temps, nous devons connecter le capteur à l'alimentation 5v. Ensuite, il faut régler la tension de seuil à l'entrée du comparateur LM393 (3) doit réglée à l'aide d'une résistance variable. [39]

Lorsque ce capteur détecte une flamme, la résistance du phototransistor diminue. Ainsi, une faible tension provenant du phototransistor est transmise à l'entrée du circuit intégré (2). [39]

Par la suite, le Circuit intégré LM393 (IC) compare cette tension avec la tension de seuil. Dans cette condition, la tension d'entrée est plus faible par rapport à la tension de seuil. Par conséquent, la sortie du comparateur LM393 génère une tension basse (0V) dans la sortie du capteur. [39]

Par contre, si le module ne détecte pas de flamme, la résistance du phototransistor est élevée. Ainsi, une haute tension provenant du phototransistor est transmise à l'entrée du circuit intégré LM393 (IC). [39]

Ce dernier compare cette tension avec la tension de seuil. Dans ce cas, la tension d'entrée est supérieure à la tension de seuil, La sortie du capteur passe donc à l'état haut (5V). [39]

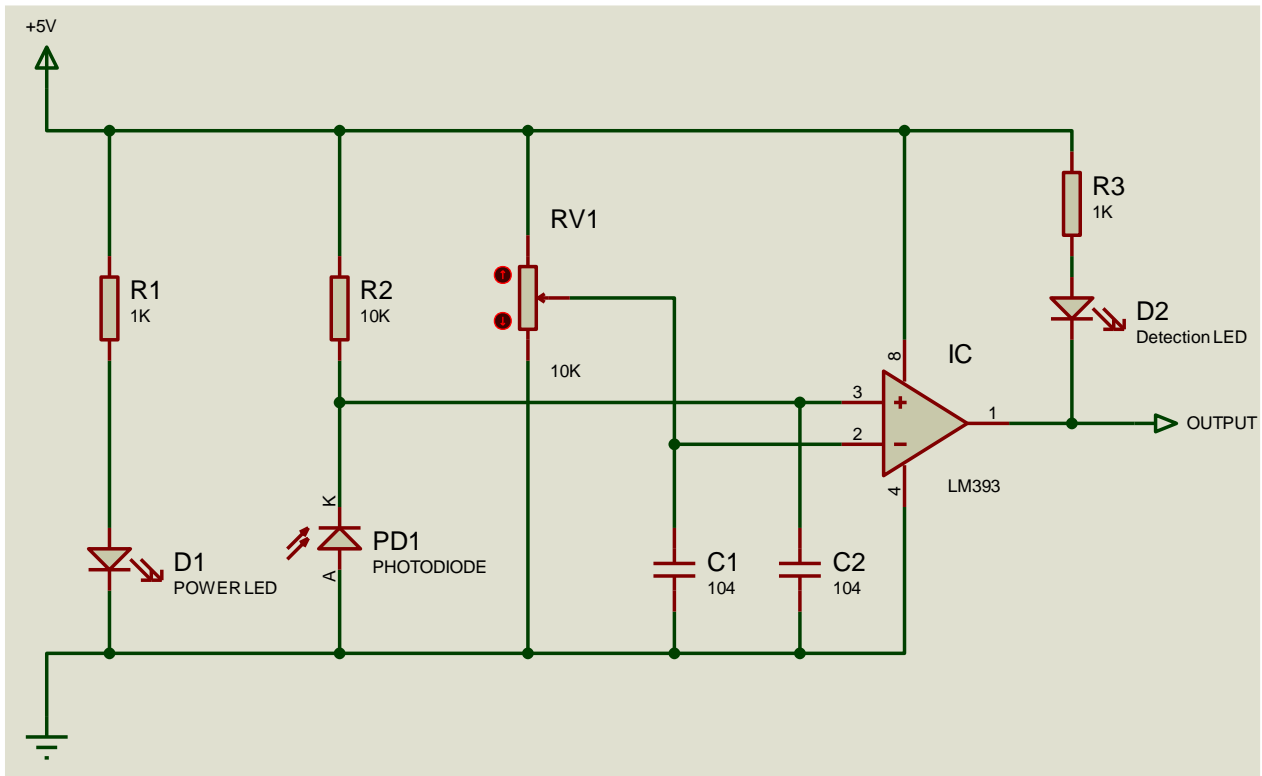


Figure II. 23 : Schéma du circuit du module de détection de flamme à infrarouge IR. [39]

II.8 Écran LCD I2C 16X2 avec rétro-éclairage bleu

Il s'agit d'un écran LCD 16X2 avec rétro éclairage bleu équipé d'un convertisseur I2C qui permet la connexion à Arduino en utilisant seulement deux broches par connexion, plus les deux câbles d'alimentation.

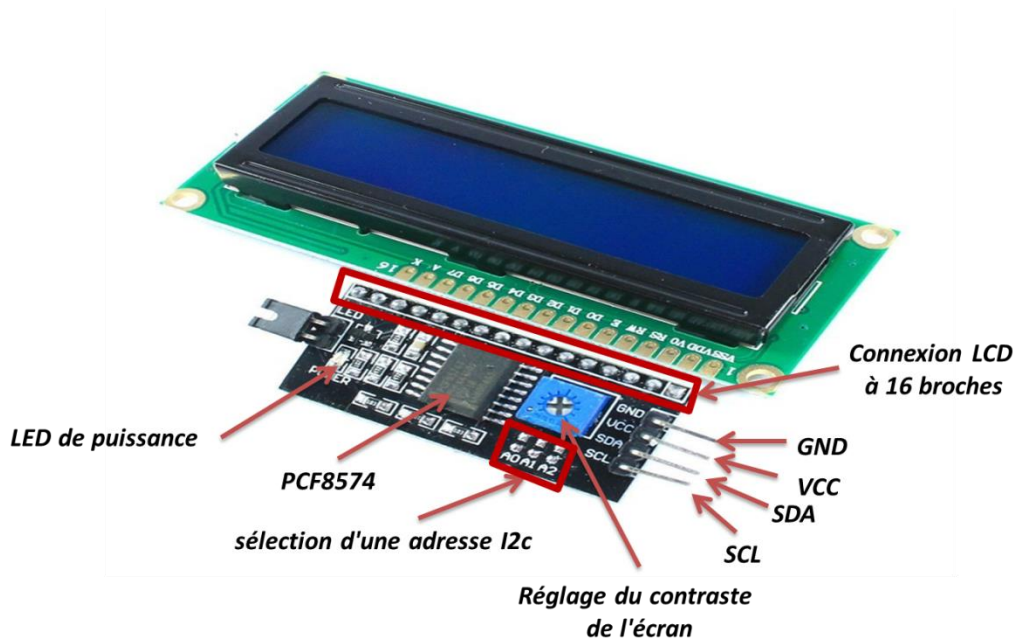


Figure II. 24 : Structure d'un écran LCD-I2C (16*2).

II .8.1 Présentation du matériel de l'écran LCD I2C

Un écran LCD I2C typique se compose d'un écran LCD à caractères basé sur le HD44780 et d'un adaptateur LCD I2C. [40]

a. Écran LCD

Ces écrans LCD sont idéaux pour l'affichage de texte/caractères uniquement. Par exemple, un LCD 16×2 caractères, doté d'un rétro éclairage par LED et peut afficher 32, avec 16 caractères sur chaque rangée. [40]

b. Adaptateur I2C pour LCD

Le cœur de l'adaptateur est une puce d'extension d'E / S 8 bits-PCF8574. Cette puce convertit les données I2C de l'Arduino en données parallèles requises par l'écran LCD. [40]

II .8.2 Adresse I2C du LCD

Si vous utilisez plusieurs périphériques sur le même bus I2C, vous devrez peut-être définir une adresse I2C différente pour la carte, afin qu'elle n'entre pas en conflit avec un autre périphérique I2C. [40]

Pour ce faire, il y a trois connexions nommées A0, A1 et A2 pour définir (3 bits) l'adresse I2C entre 0x20 et 0x27. L'adresse par défaut est 0x27 (A0, A1 et A2 laissés ouverts). Une paire est fermée en connectant les deux plots respectifs ensemble. [40]

II .8.3 Connecter un Arduino Méga à un écran LCD I2C

La connexion d'un LCD I2C est beaucoup plus facile que celle d'un LCD standard. En effet, vous n'avez besoin de connecter que 4 broches au lieu de 12. [40]

Par conséquent, connectez la broche VIN à la sortie 5V de l'Arduino, GND à la masse et les 2 autres (SDA, SCL) à la sortie I2C BUS de l'Arduino qui sont les broches numériques 20 et 21. [40]

II .9 Système RFID

Le terme RFID « Radio-Frequency Identification » englobe toutes les technologies qui utilisent les ondes radio pour identifier automatiquement des objets ou des personnes. [41]

Le système RFID ou système d'identification par radiofréquence se compose de deux éléments principaux :

- L'étiquette RFID (carte ou tag) est elle-même équipée d'une puce reliée à une antenne.
- Lecteur RFID équipée d'une antenne intégrée ou externe. [41] [42]

II.9.1 Description de la carte RFID RC522

Le module RFID RC522 basé sur le circuit intégré MFRC522. Ce module est conçu pour créer un champ électromagnétique de 13,56 MHz pour communiquer avec les étiquettes RFID. Ainsi que, Le lecteur peut communiquer avec un microcontrôleur via une interface périphérique série (SPI) à 4 broches avec un débit maximal de 10 Mbps. Il permet aussi de communiquer via les protocoles I2C et UART. [42]

La tension de fonctionnement du module est de 2,5-3,3V. De plus, les broches logiques sont tolérantes à 5V, donc nous pouvons facilement le connecter à un Arduino ou tout microcontrôleur avec une logique 5V sans utiliser un convertisseur de niveau logique. [42]

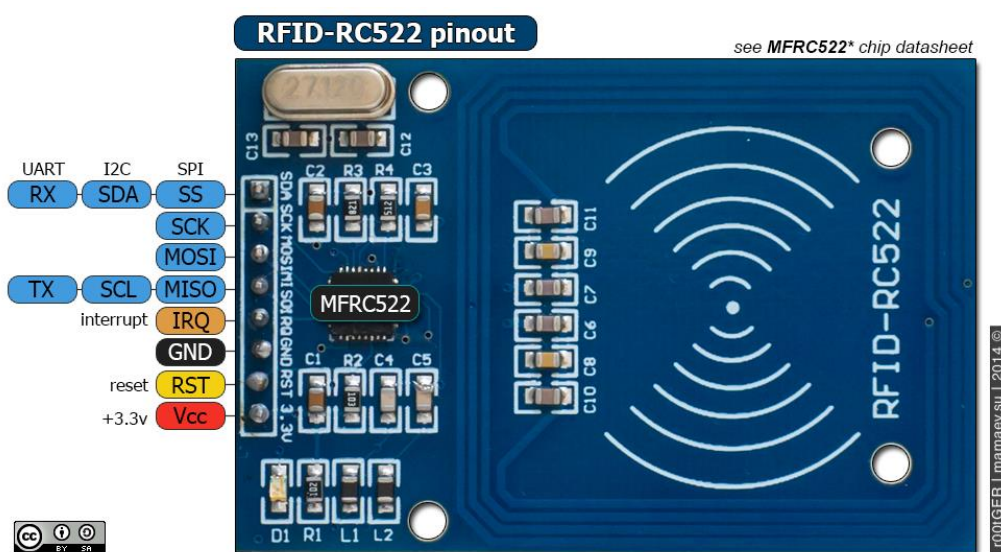


Figure II. 25 : Schéma de brochage du RC522.

Le tableau suivant indique les spécifications techniques du lecteur RFID :

Spécifications techniques	
Gamme de fréquences	Bande ISM 13,56 MHz
Interface hôte	SPI / I2C / UART
Tension d'alimentation	2,5 V à 3,3
Max. Courant de fonctionnement	V13-26mA
Min. Courant (hors tension)	10µA
Entrées logiques	Tolérance de 5V

Tableau II. 6 : Les spécifications de la carte RFID RC522. [42]

II .9.2 Principe de fonctionnement d'un système RFID

L'étiquette RFID (carte ou tag) est elle-même équipée d'une puce reliée à une antenne, l'antenne permet à la puce de transmettre les informations (numéro de série) qui sont lues par un lecteur. [42]

Une fois les informations transmises au lecteur RFID équipée d'une antenne intégrée ou externe, celui-ci n'a plus qu'à convertir les ondes-radio en données et ceux-ci pourront être lus par un logiciel RFID. [42]

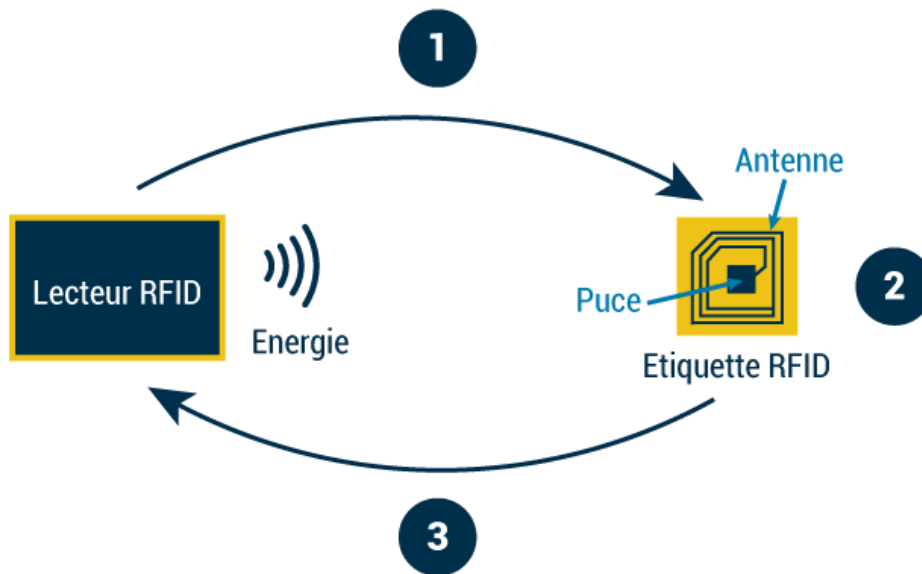


Figure II. 26 : Schéma de principe de système RFID. [43]

II .10 Buzzer

Un avertisseur sonore est un composant électronique qui produit un son lorsqu'une tension lui est appliquée.



Figure II. 27 : Buzzer (Avertisseur sonore).

II .10.1 Fonctionnalité

Le buzzer possède deux petites pattes de fixation. La puissance sonore de ce composant est d'environ 85 dB / cm et il nécessite une tension continue pour fonctionner. La tension continue doit généralement être comprise entre 3 V et 28 V. Un buzzer conçu pour 6 V peut généralement bien fonctionner à n'importe quelle tension d'alimentation de 4 V à 8 V, tandis qu'un buzzer conçu pour 12 V peut fonctionner parfaitement à une tension de 6 V à 28 V (veuillez-vous référer aux spécifications données) fabricant pour éviter les erreurs). [44]

Les buzzers électriques sont utilisés pour émettre des bips.

II .11 CONCLUSION

Dans ce chapitre, nous avons donné une description générale des différents éléments qui constituent notre prototype. Les principaux dispositifs sont la carte Arduino méga et la carte ESP8266 et le module IR.

Le capteur IR est utilisé pour la détection tandis que la carte Arduino méga a pour mission de commander les différents périphériques (capteur IR, servomoteur, capteur de flamme et l'écran LCD).

La carte ESP8266 permet la gestion efficace du parking intelligent et la communication de différentes informations (l'emplacement des véhicules, tarification, etc...).

Chapitre III

Simulation et réalisation du circuit

III.1 Introduction

Notre projet consiste à réaliser un système du stationnement intelligent.

Dans ce chapitre nous allons expliquer le principe de fonctionnement de notre parking intelligent. Ainsi que les différents blocs qui constituent notre projet.

Ces différents étapes sont divisées en deux parties :

La première partie concerne l'accès au parking tandis que figure 1.

La deuxième partie est exploitée pour la gestion de parking.

Pour faire, on a fait appel à la technologie IoT, Cette nouvelle technologie représente une combinaison entre les différents domaines tels que : Electroniques, automatiques, informatiques et télécommunications.

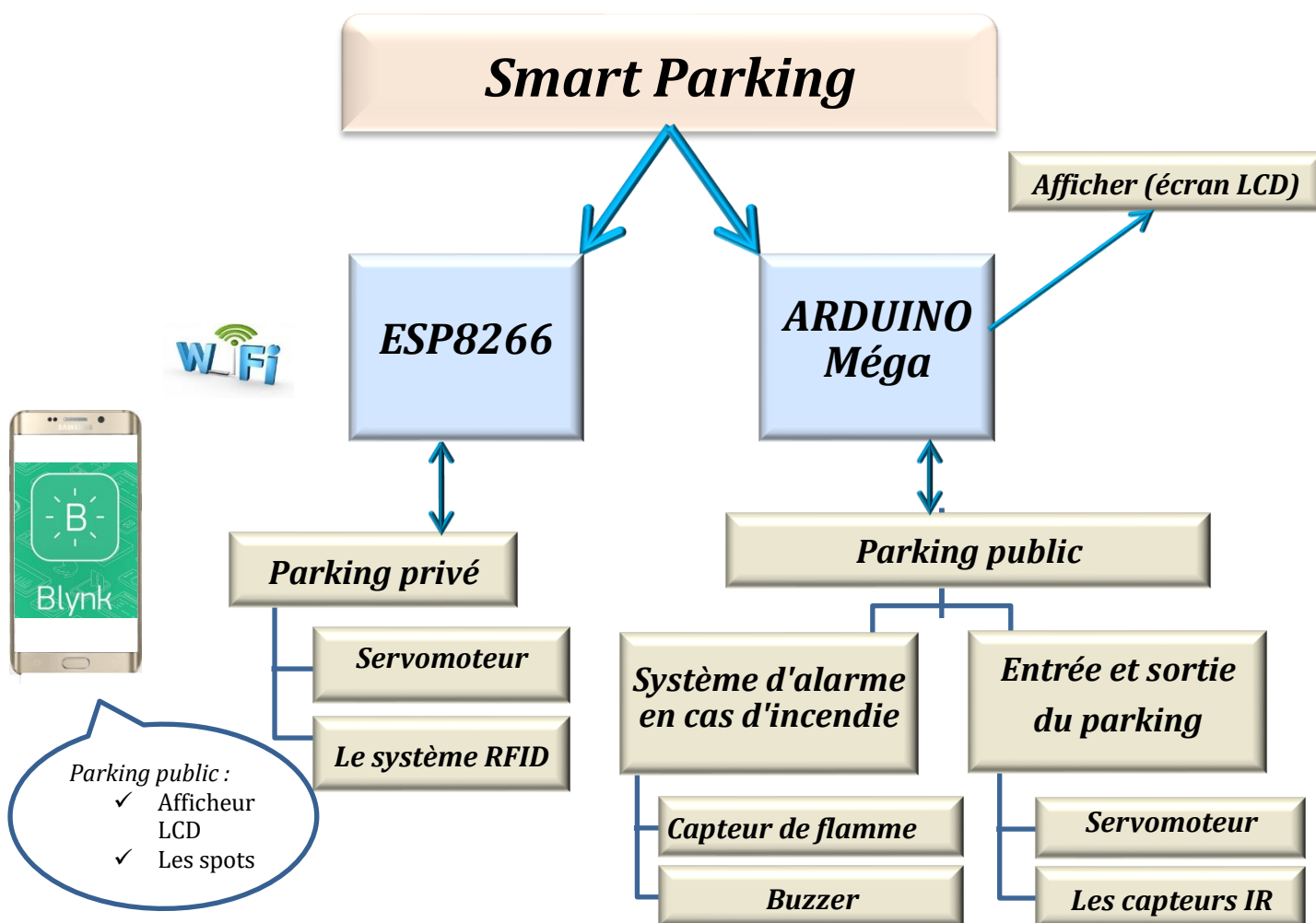


Figure III. 1 : Schéma général de communication du système prototype.

Cependant, Avant de passer à la réalisation de notre circuit, est très intéressant de simuler notre prototype.

III.2 La partie Simulation

La simulation joue un rôle important pour le développement scientifique et technologique. Elle nous offre la possibilité d'étudier et de modéliser les différents systèmes avant de passer à la réalisation. [45]

Ainsi, elle permet aux fabricants de réduire le coût et les dépenses engendrés par les essais expérimentaux. [45]

On peut trouver plusieurs logiciels de simulation tels que Multisim, PDN Analyzer...etc. Cependant le logiciel le plus utilisé est le logiciel « PROTEUS ».

III.2.1 Présentation du logiciel PROTEUS

PROTEUS est une suite logicielle qui permet la CAO électronique. Développé par la société « Labcenter Electronics ».

En générale, le logiciel PROTEUS est principalement composé de deux modules : ISIS et ARES. [46]

Cette suite logicielle est très connue dans le domaine de l'électronique. À part la popularité de l'outil, de nombreuses institutions dont les lycées et les universités utilisent PROTEUS.

ISIS : Ce logiciel de PROTEUS est principalement connu pour la création de schémas et la simulation électrique. [46]

ARES : Ce logiciel de PROTEUS est principalement utilisé pour créer des schémas de circuits imprimés. [46]

III.2.2 Présentation du module ISIS

Le logiciel PROTEUS ISIS est principalement utilisé pour l'édition de schémas électriques. En plus, le logiciel nous permet également de simuler des schémas précédemment créés, ce qui nous permet de détecter certaines erreurs au stade de la conception. Grâce à lui, nous pouvons également contrôler la plupart des aspects graphiques du circuit. [46]

III.3 Schémas et méthodologie suivie

III.3.1 Description du prototype

Notre projet consiste à réaliser d'un parking intelligent (smart parking) en utilisant la technologie des objets connectés-IoT.

Pour mieux comprendre le principe de fonctionnement de notre prototype nous avons divisé notre projet en trois parties :

- Accès au parking.
- Gestion du parking.
- Affichage des informations.

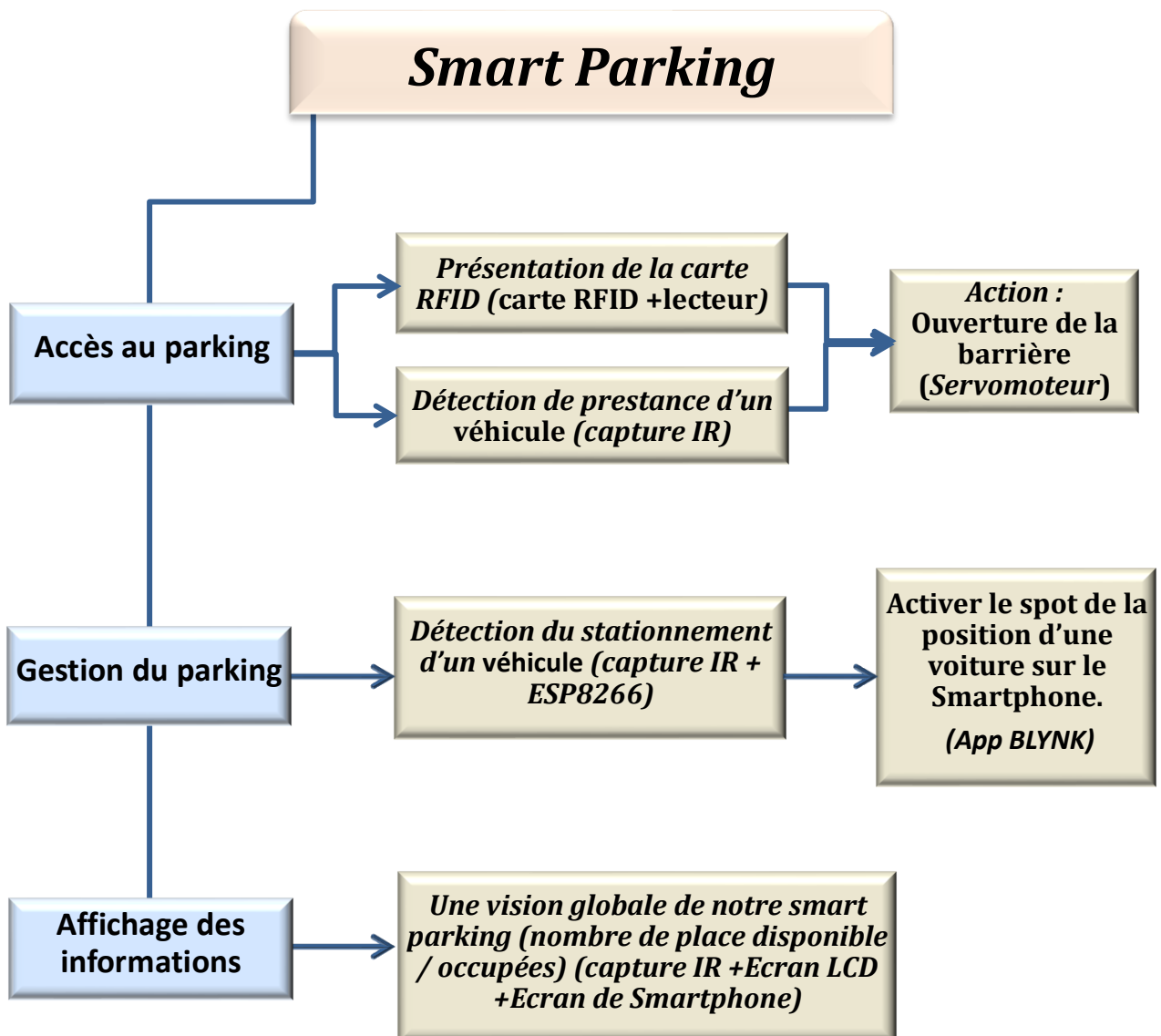


Figure III. 2 : Représentation schématique du prototype.

III.3.1.1 Bloc d'accès au parking

Ce bloc est divisé en deux catégories :

- Catégorie des clients non abonnés
- Catégorie des clients abonnés

Pour la première catégorie (parking publique) l'accès est assuré en utilisant un capteur IR et un servomoteur.

Ces deux éléments sont connectés à une carte Arduino Méga dont la sortie du capteur IR est liée avec une entrée analogique de la carte Arduino tandis que le servomoteur est branché avec une sortie numérique de l'Arduino Méga.

En effet, la présence d'un véhicule à l'entrée du parking génère une tension à la sortie du module IR. Ceci permet à l'Arduino Méga de commander (faire tourner) le servomoteur et de contrôler la position de la barrière.

La figure III.3 illustre le branchement de ce module avec la carte Arduino sous ISIS :

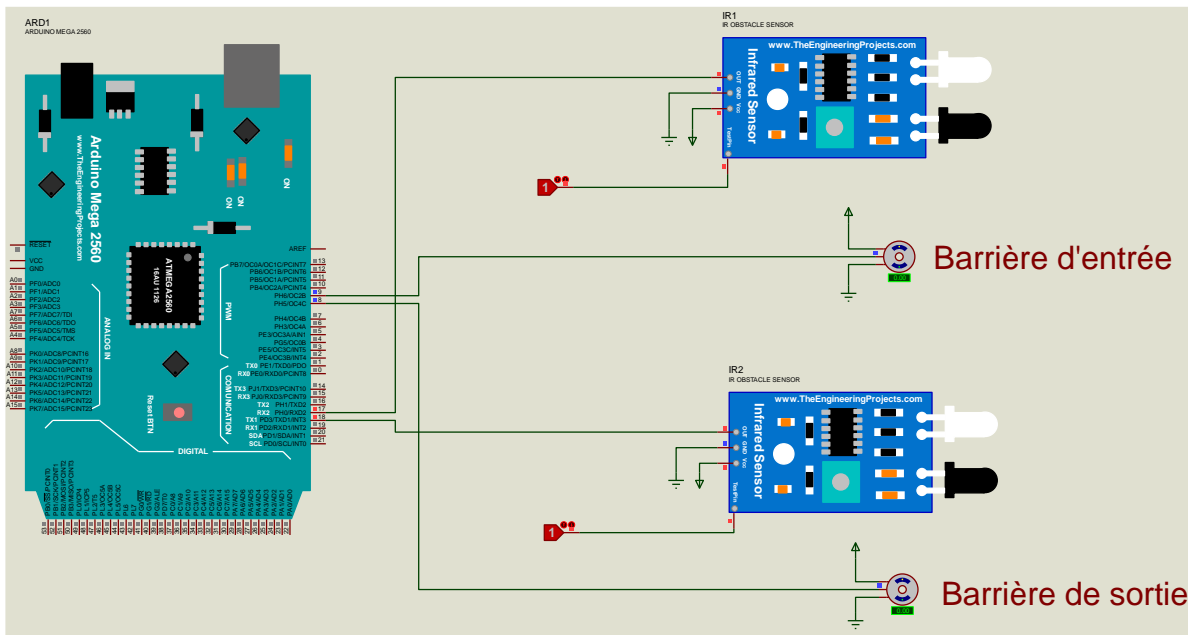


Figure III. 3 : Le branchement du capteur IR et servomoteur avec Arduino Méga sous ISIS.

Pour la deuxième catégorie (parking privé) l'accès est effectué via un système RFID (carte RFID +lecteur). Ce dernier est connecté à la carte ESP8266 en utilisant des câbles de liaison (voir figure III.4).

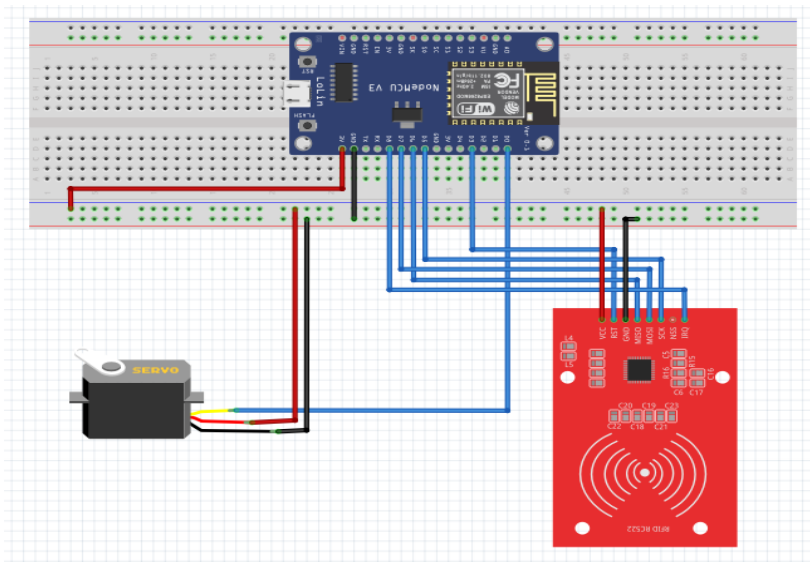


Figure III. 4 : Représentation du système RFID sous Fritzing.

Dans ce cas, pour que le client abonné puisse accéder au parking il suffit de présenter sa carte RFID.

III.3.1.2 Bloc de la gestion du parking

Le bon fonctionnement de ce bloc nécessite l'intervention de deux parties.

La première partie représente la partie matérielle. Elle est constituée d'une carte ESP8266 et des capteurs IR.

La figure III.5 illustre le branchement de ces éléments

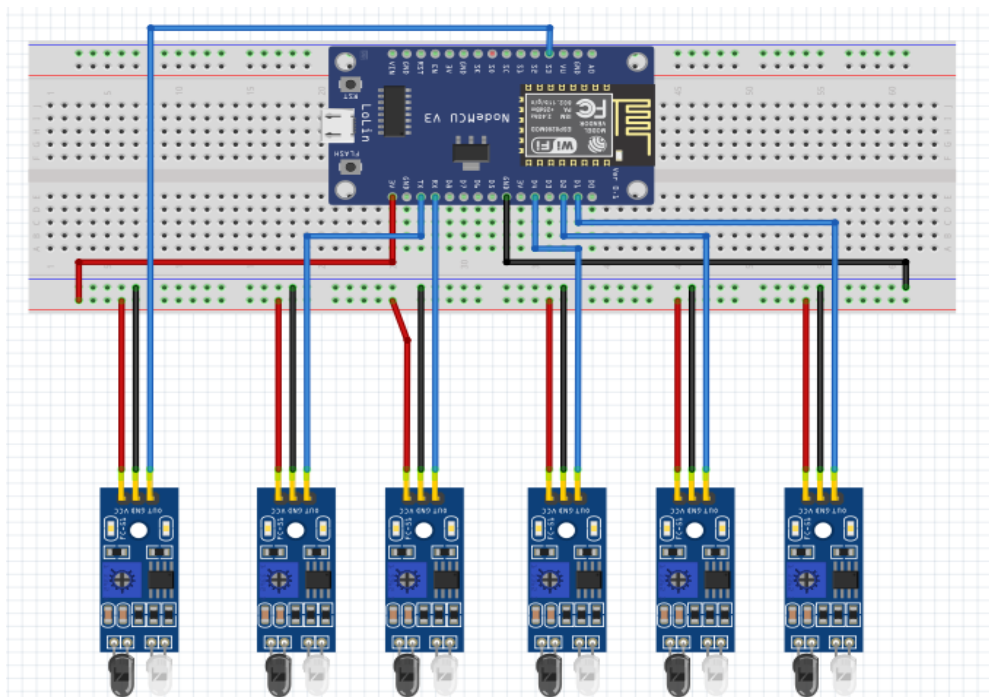


Figure III. 5 : Branchement des capteurs IR avec la carte ESP8266.

La deuxième partie nécessite le développement d'une application sur le Smartphone en utilisant l'application BLYNK.

En effet, une fois que le véhicule est accédé au parking, il suffit de choisir un emplacement pour stationner.

Après le stationnement, le capteur IR (déjà installé) détecte la présence de cette voiture. Il délivre un signal électrique à la carte ESP8266.

Cette dernière est connectée via une connexion wifi avec l'application déjà installé sur le Smartphone. Ceci nous permet de visualiser l'emplacement occupé par le véhicule.

III.3.1.3 Bloc d'affichage des informations

Dans ce dernier bloc on trouve deux types d'affichages.

Le premier type est réservé pour l'affichage de nombres de places libres et occupées. Pour cela il faut utiliser un écran LCD connecté avec le module I2C (voir figure III. 6).

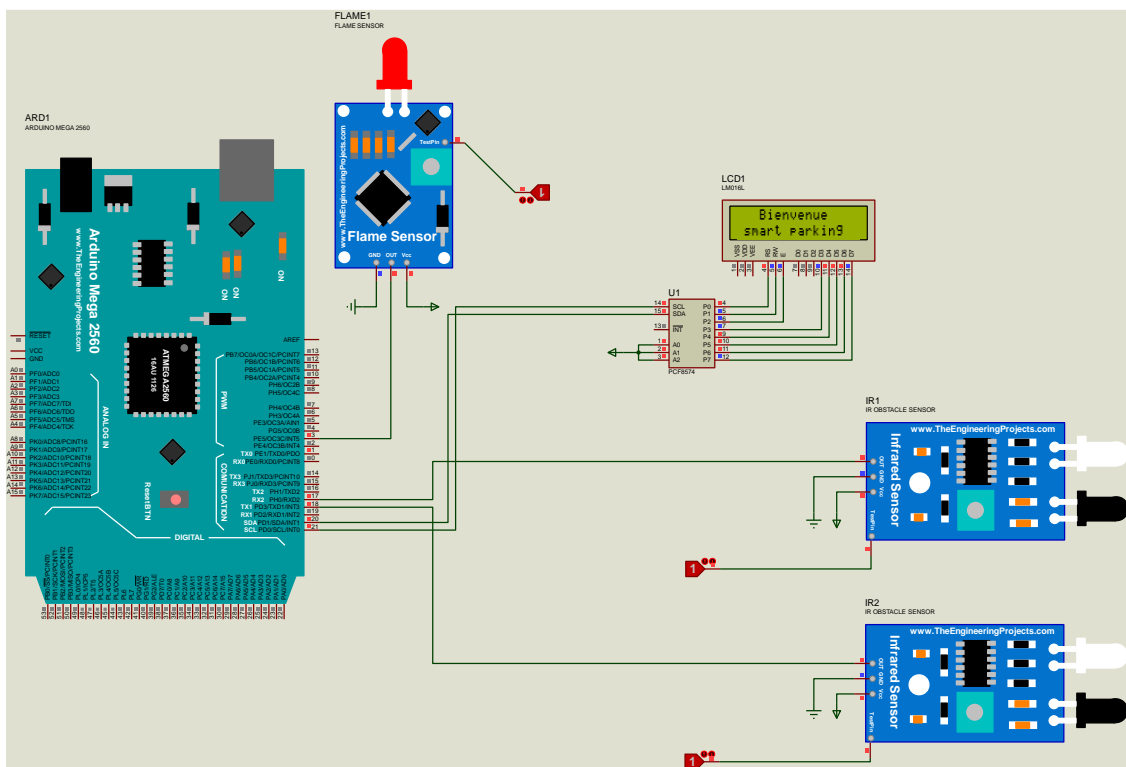


Figure III. 6 : Bloc d'affichage sous ISIS.

Le deuxième type d'affichage nous donne une vue globale sur l'emplacement de tous les véhicules en utilisant l'écran du smart phone.

III.4 La partie réalisation

Lors de la mise en marche de notre circuit, la carte ESP8266 s'est connectée au smart parking. Et l'écran LCD affiche un message d'accueil « Bienvenue Smart parking »



Figure III. 7 : Affiche un message d'accueil « Bienvenue Smart parking ».

Après quelque temps (3 second) L'écran LCD affiche le nombre de places vacants ainsi que nombre de places occupées.



Figure III. 8 : Nombre de places occupées et libres.

Dans Le premier cas où le véhicule représente un client abonne, l'accès au parking nécessite l'utilisation d'un système RFID. Ce system est constitué d'une carte RFID (ou tag) et d'un lecteur RFID (module RC522).

Ce dernier est équipé d'une puce intégrée. Cette puce permet de transmettre l'information (numéro de série) vers le lecteur RFID sous la forme d'ondes RADIO via une antenne intégrée.

Ce module utilise le protocole SPI (Serial Périphérique Interface) pour communiquer avec l'ESP8862. La communication SPI utilise des boches spécifiques des microcontrôleurs ESP8266. Le brochage se fait comme suit :

RC522	ESP8266
VCC	3V3
GND (Masse)	GND
RST (Reset)	D3 (GPIO0)
(SPI)MISO (Master Input Slave Output)	D6 (GPIO12)
(SPI)MOSI (Master Output Slave Input)	D7 (GPIO13)
(SPI)SCK (Serial Clock)	D5 (GPIO14)
(SPI)SS/SDA (Slave select)	D8 (GPIO15)

Tableau III. 1 : Le brochage du module RC522 avec ESP8266.

Dans ce cas, pour que l'abonné puisse accéder au parking, il lui suffit de présenter sa carte RFID (carte ou tag).



Figure III. 9 : Présenter sa carte RFID valide à l'entrée.

Une fois que les informations de la carte RFID sont réceptionnées puis convertis par lecteur RFID, il faut comparer ces données avec les codes déjà enregistrés dans L'ARDUINO IDE.

Si le code de la carte RFID du client correspond bien au code déjà enregistrer dans la L'ARDUINO IDE, la sortie de carte Esp8266 envoie un signal électrique. Par conséquent, le servomoteur est excité et change de position. Ceci permet l'ouverture de la barrière et autorise le client abonné d'accéder au parking privé.

Le programme de cette partie est présenté dans la figure III.10 suivantes :

```
if ((content.substring(1)=="B9 87 EF A5") || (content.substring(1)=="95 9A 07 6D"))
{
    monServomoteur.write(30);
    delay(3000);
    monServomoteur.write(90);
}
```

Figure III. 10 : Programme pour le système RFID.

Dans le deuxième cas où le client n'est pas un abonné, le véhicule n'a pas le droit d'accéder au parking prive mais au parking public.

En effet, lors de la présence d'un véhicule, la lumière IR émise par l'émetteur est réfléchi vers la photodiode.

De ce fait, la sortie du comparateur LM393 génère une tension de 5V. Cette tension est transmise à l'entrés analogique (Pin A0) de la carte ARDUINO Méga. Par la suite, la sortie numérique de l'ARDUINO se trouve à l'état haut. Par conséquent, le servomoteur est excité et change sa position initiale de 0° à 90°. Ceci permet l'ouverture de la barrière. Dans ces conditions le véhicule puisse accéder au parking public.

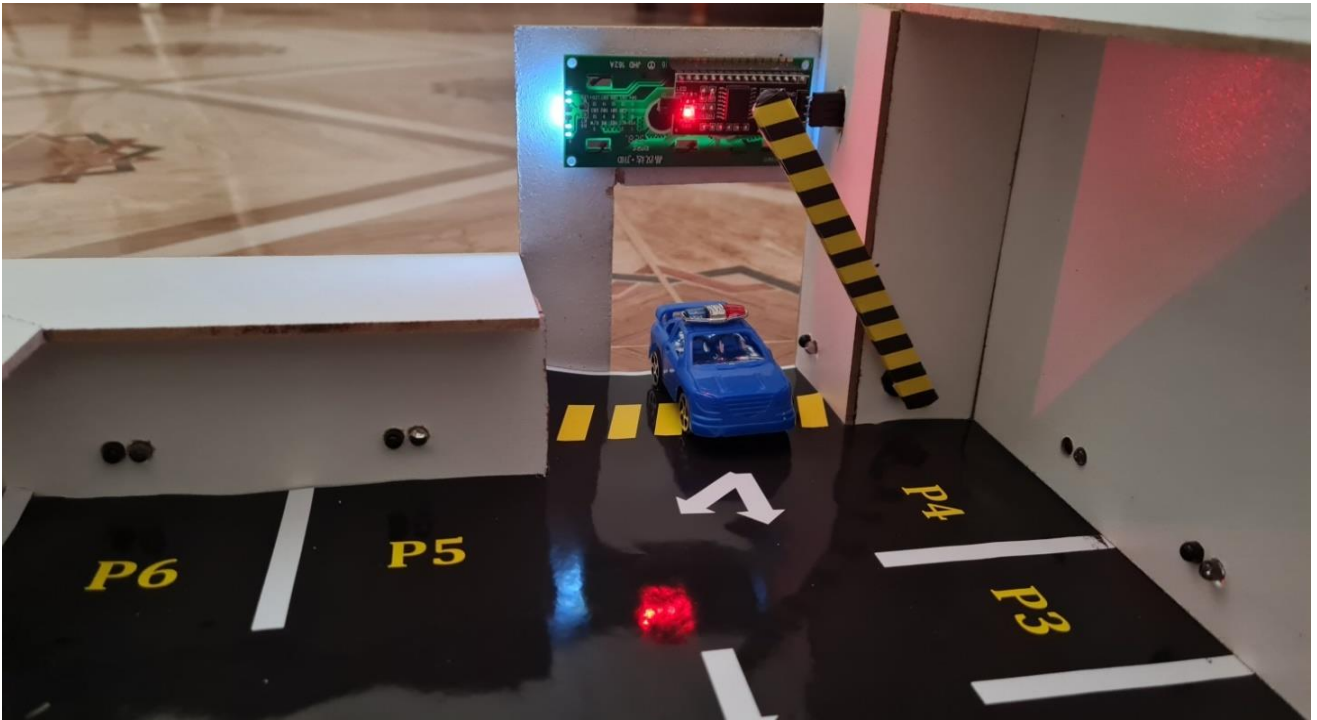


Figure III. 11 : Détection d'une voiture en cas d'entrée.

Par conséquent, le nombre de places disponibles décrémente et le nombre des places occupées s'incrémentes. Ces informations sont visualisées sur l'écran LCD.



Figure III. 12 : L'afficheur LCD en cas d'entrée d'une voiture.

Pour réaliser cette partie, nous rajoutons le programme suivant à l'Arduino IDE.

```

if (digitalRead(capteur1)==LOW && espace !=0)
{
  if (espace<=valMax && espace>=1)espace--;
  do{
    //lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Espace occupe: ");
    lcd.print(6-espace);
    lcd.setCursor(0,1);
    lcd.print("Espace Libre : ");
    lcd.print(espace);
    monServomoteur2.write(40);

  }while((digitalRead(capteur1)==LOW));
  delay(3000);
  monServomoteur2.write(90);
}
    
```

Figure III. 13 : Code de l'entrée de parking sur l'Arduino IDE

Une fois accédé au parking, le conducteur devra choisir un emplacement afin de stationner. Cet emplacement est équipé par des capteurs IR. Ces capteur IR sont connectes à une carte esp8266.

Numéro du capteur IR	ESP8266
Capteur 1 (place 1)	D1 (GPIO5)
Capteur 2 (place 2)	D2 (GPIO4)
Capteur 3 (place 3)	D4 (GPIO2)
Capteur 4 (place 4)	RX (GPIO3)
Capteur 5 (place 5)	TX (GPO1)
Capteur 6 (place 6)	SD3 (GPIO10)

Tableau III. 2 : Branchement des capteurs IR avec la carte ESP8266.

Cependant, avant d'utiliser la carte ESP8266 il est indispensable de la configurée.

Pour cela il faut passer par plusieurs étapes.

La premier étape consiste à télécharges la Package ESP8266. Une fois téléchargé, il suffit d'effectuer une recherche afin d'installer cette carte dans la plate-forme Arduino IDE.

L'étape suivante nécessite le téléchargement et l'installation des Bibliothèques « ESP8266 » ainsi que « BLYNK ».

Une fois terminé, l'étape suivante consiste à réaliser une application ANDROID en utilisant l'application BLYNK (déjà installée sur notre Smartphone).

Pour faire, il faut tout d'abord créer un compte en utilisant un email personnel.

Par la suite, il suffit de créer un nouveau projet en utilisant le code d'authentification envoyé par l'application BLYNK dans la boîte Email.

L'étape suivante nécessite d'établir une connexion entre ESP8266 et l'application BLYNK (via la connexion wifi).

Pour accomplir cette tâche, il faut utiliser les données : Nom d'utilisateur et le mot de passe du wifi dans le programme l'ESP8266 :

```
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

char auth[] = "CQ4ouKCqilWRW2fkd0cwbbPSVJn7aH3I";

char ssid[] = "D-Link.DZ";
char pass[] = "benomari@122";
```

Figure III. 14 : Les données de connexion entre BLYNK et ESP8266 via Wifi.

Ensuite, il faut créer les objets « LED ou spot » et « tableau ». Pour faire il faut utiliser l'icône (+) de l'application BLYNK (voir la figure III.15)

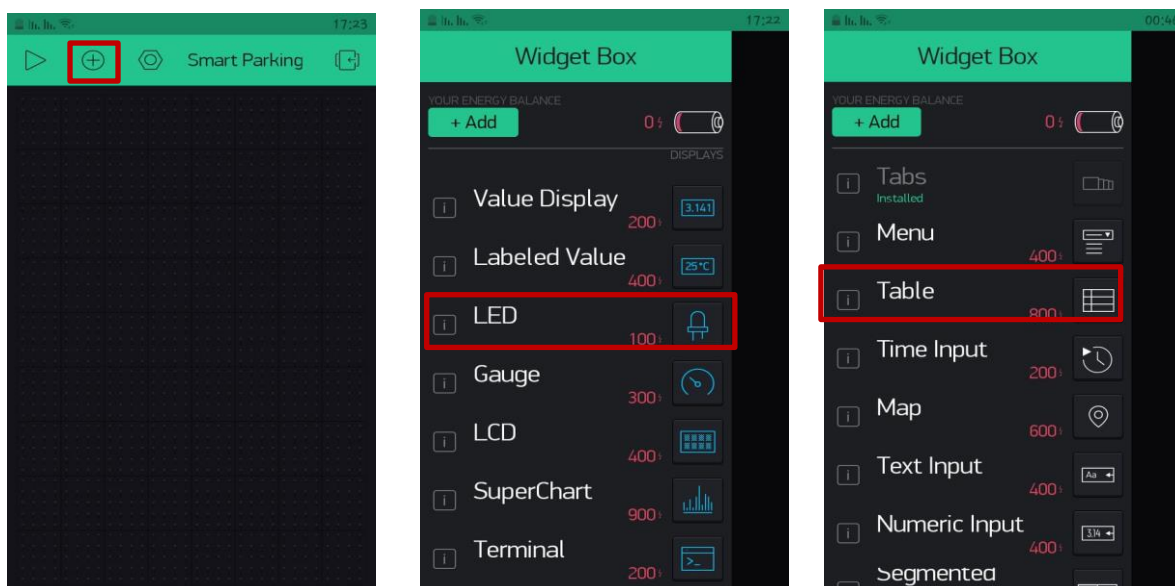


Figure III. 15 : Ajoute le tableau et les LEDs dans l'application BLYNK.

Après la création des spots, il est important de les configurer en affectant pour chaque spot un port virtuel.

Les ports virtuels nous permettent d'établir une liaison entre la carte ESP8266 et l'application BLYNK.

Le tableau III. 3 nous indique les ports virtuels utilisés pour chaque spot :

Numéro du spot (LED)	Numéro du capteur IR (place)	Pin Virtuel
Spot 1	Place (P1)	V10
Spot 2	Place (P2)	V11
Spot 3	Place (P3)	V12
Spot 4	Place (P4)	V13
Spot 5	Place (P5)	V14
Spot 6	Place (P6)	V15

Tableau III. 3 : Représentation le pin virtuel de cheque spot.

Après la création et la configuration des « spots » on passe à la configuration de l'objet « tableau ».

Pour faire, il faut tout d'abord affecter à ce tableau un port virtuel (V0).



Figure III. 16 : Configuration de l'objet « tableau » dans l'application.

L'étape suivant consiste à créer les lignes et les colonnes du tableau dont les lignes représentent le nombre de capteurs IR et les colonnes représentent le numéro de chaque place ainsi que le prix.

```

table.addRow(1, "P1", 0);
table.addRow(2, "P2", 0);
table.addRow(3, "P3", 0);
table.addRow(4, "P4", 0);
table.addRow(5, "P5", 0);
table.addRow(6, "P6", 0);
    
```

Figure III. 17 : Code de création du tableau dans l'Arduino IDE.

Pour cela, il suffit d'utiliser le programme suivant :

```
String pr1=String(prix1)+" DA";  
Blynk.virtualWrite(V0, "update", 1, pr1, "P1");  
table.pickRow(prix1);
```

Figure III. 18 : Code d'affiche le nombre de place et le prix.

En effet, lors de la détection (stationnement véhicule), le module IR activé, envoie un signal électrique vers l'entrée numérique de la carte ESP8266 (Ex Pin D1).

Par la suite la carte NODE MCU transmet cette information à l'application ANDROID (BLYNK) via une connexion Wifi.

Ceci permet d'afficher l'emplacement occupé par le véhicule sur l'écran de notre téléphone portable ainsi que le prix de chaque place de stationnement

Pour réaliser ces étapes, il est nécessaire d'ajouter ce code dans le programme de l'arduino.

```
if (digitalRead(capteur1)==0) {  
    Blynk.virtualWrite(V10, 1000);  
    cmp1++;  
    delay(1000);  
    if(cmp1%10==0) prix1=prix1+10;  
    String pr1=String(prix1)+" DA";  
    Blynk.virtualWrite(V0, "update", 1, pr1, "P1");  
    table.pickRow(prix1);  
}  
else{  
    Blynk.virtualWrite(V10, 0);  
    cmp1=0;  
    prix1=30;  
}
```

Figure III. 19 : Code pour afficher l'emplacement et le prix.



Figure III. 20 : Affichage de l'emplacement sur l'écran du smart phone.

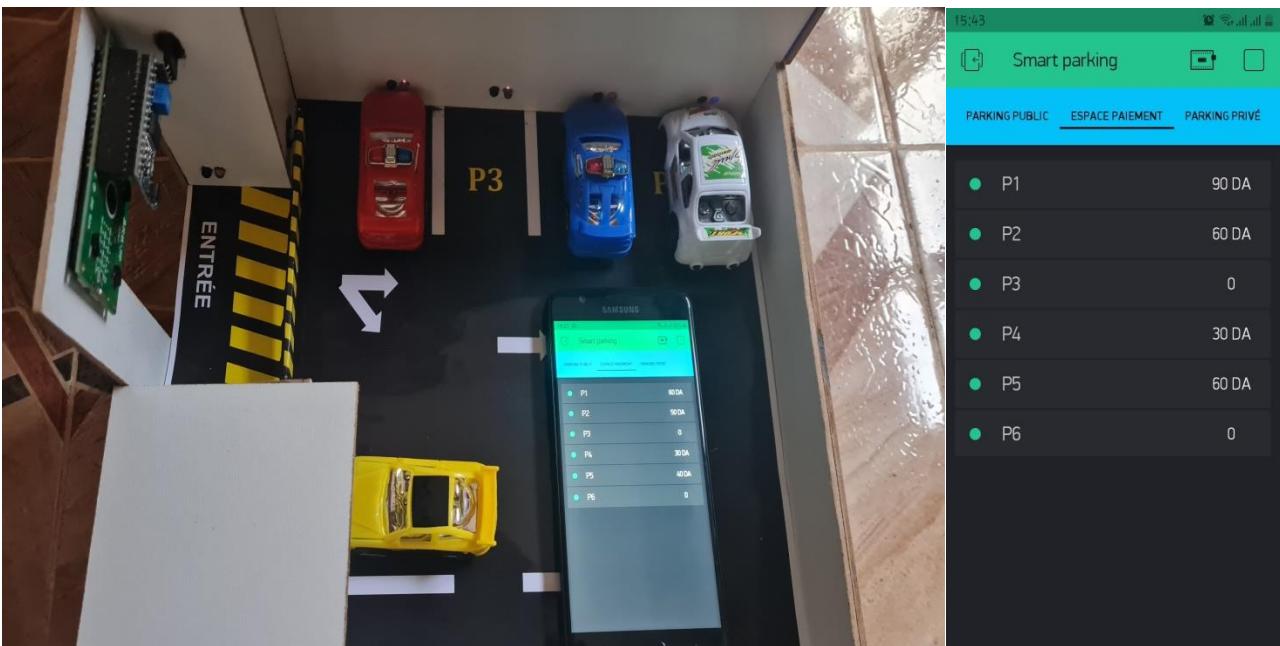


Figure III. 21 : Affiche le prix sur smart phone.

En fin, lorsque la voiture quitte son emplacement, l'écran du Smartphone nous indique le nombre de place libre (spot éteint).

Pour que le véhicule puisse quitté le parking, il suffit de se dirigé vers la sortie de parking.

Cette sortie est équipée par un capteur IR et un servomoteur commande par carte Arduino Méga (voir la figure III.22).

Une fois que la voiture quitte définitivement le parking, le nombre des places libres s'incrémentes et le nombre de places occupées décrémente.

Ces données sont affichées sur l'écran LCD (voir la figure III.23).



Figure III. 22 : Détection d'une voiture en cas de sortie.



Figure III. 23 : L'afficheur LCD en cas de sortie trio voiture.

Afin de renforcer les mesures de sécurité nous avons installé un détecteur de flamme.

Ce système est constitué d'un capteur de flamme et d'un BUZZER. Ces deux dispositifs sont commandés par la carte Arduino Méga.

En effet, lors d'un incendie, le capteur de flamme IR détecte la flamme via le phototransistor et génère une tension à la sortie de l'amplificateur opérationnel LM393.



Figure III. 24 : La détection d'incendie.

Par la suite, ce signal électrique est transmis à l'entrée numérique (pin 3) de la carte Arduino Méga. De ce fait, le système d'alerte s'active et le BUZZER se déclenche (un effet sonore).

Une fois déclencher, la carte Arduino transmet un signal numérique à l'afficheur LCD. Ce dernier convertit ce signal et affiche le message « alerte d'incendie ».



Figure III. 25 : Affichage du message d'alerte en cas d'incendie.

Les figures III.26 et III.27 représentant une vue globale de notre prototype.

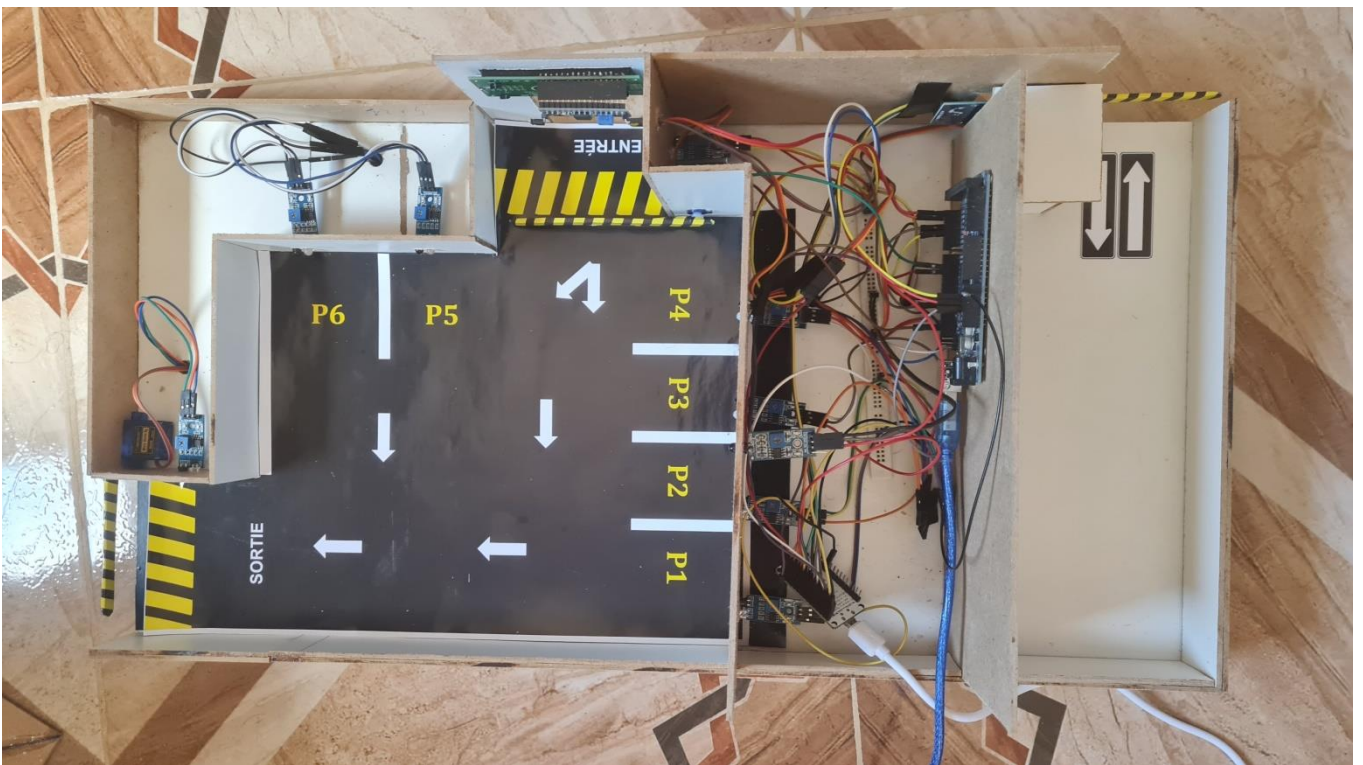


Figure III. 26 : représentation globale du smart parking.



Figure III. 27 : représentation globale du smart parking.

III.5 Conclusion

Dans ce chapitre, nous avons présenté les différents étages qui constituent notre projet.

Pour une gestion optimale et efficace, il était intéressant pour nous de faire une simulation en utilisant le logiciel PROTEUS.

Ensuite, nous avons procédé à la réalisation des trois blocs : le bloc d'accès au parking, le bloc de gestion du parking et le bloc d'affichage des informations.

Une fois réalisé, plusieurs essais ont été effectués. Ces tests nous a permis de valider le bon fonctionnement de notre système.

Conclusion générale

Conclusion générale

Ces dernières années, le parc automobile a connu une expansion très importante, cette augmentation a créé lui des principaux problèmes de la ville intelligent qui est la congestion. C'est dans ce contexte que notre projet s'intervient.

En effet, l'objectif de notre mémoire est de réaliser un parking intelligent en utilisant la technologie IoT.

Notre prototype se compose de plusieurs blocs, à savoir le bloc d'accès au parking, le bloc de gestion de parking et le bloc d'affichage.

La première étape consistant à simuler notre système en utilisant le logiciel PROTEUS.

Ceci nous a permis d'une part d'optimiser la conception de notre prototype et d'autre part de développer le programme sous la plateforme ARDUINO IDE.

Après l'élaboration de la partie simulation, nous avons procédé à la réalisation du circuit.

Une fois achevée, nous avons effectué plusieurs tests. Ces tests nous ont permis de vérifier la qualité du fonctionnement de notre système.

En effet, les résultats obtenus valident le bon fonctionnement de notre prototype.

La perspective qui découle de ce projet porte sur l'amélioration des aspects de sécurité du parking intelligent en installation des caméras de surveillance et d'ajouter d'autre option comme la possibilité de réserver un emplacement à distance pour une durée bien déterminée.

En raison de l'évolution technologique dans le domaine de l'automobile et de l'émergence des voitures autonomes, il est possible d'ajouter de nouveaux moyens pour accéder aux parkings privés tels que l'accès au parking via d'identification du matricule de la voiture.

Références bibliographiques

- [1] : <https://www.cigref.fr/archives/histoire-cigref/blog/histoire-des-objets-inanimés-aux-objets-connectés/>
- [2]:<https://itchronicles.com/wp-content/uploads/2019/06/bigstock-Internet-Of-Things-iot-Devi-245519329-Converted-01.jpg>
- [3]: M. Han and H. Zhang, "Business intelligence architecture based on internet of things " Journal of Theoretical & Applied Information Technology, vol. 50, no. 1, pp. 90-95, 2013.
<http://www.jatit.org/volumes/Vol50No1/13Vol50No1.pdf>
- [4]: <https://daspal.com/wp-content/uploads/2021/03/iot.jpeg>
- [5]: <https://www.intelligenzaartificiale.it/internet-of-things/>
- [6] : les experts Ooreka, "Système RFID : définition et fonctionnement d'un système RFID ", 22 déc. 2015; <https://rfid.ooreka.fr/comprendre/systeme-rfid>
- [7] : <https://www.asiarfid.com/wp-content/uploads/2020/06/rfid%E6%8A%80%E6%9C%AF.jpg>
- [8] : https://en.wikipedia.org/wiki/Wireless_sensor_network
- [9] https://upload.wikimedia.org/wikipedia/commons/thumb/3/32/Wireless_Sensor_Network_General_Structure.svg/1200px-Wireless_Sensor_Network_General_Structure.svg.png
- [10] : N. Daniel, R. Marcel, and K. Daniel, Livre blanc Machine To Machine enjeux et perspectives: Orange Business Services, Syntec informatique, Fing, 2006, 40 p.
http://img.icbdr.com/images/LJ/cms/pdf/LB_M2M_vf.pdf
- [11] : <https://data-flair.training/blogs/iot-applications/>
- [12] : https://blog.integral-system.fr/wp-content/uploads/2019/02/industry-2496192_960_720.jpg
- [13] : <https://actualiteinformatique.fr/internet-of-things-iot/quest-ce-que-iot-internet-of-things-internet-des-objets>
- [14] : <https://fado.vn/blog/wp-content/uploads/2020/10/thiet-b%E1%BB%8B-smart-home-la-gi.jpg>
- [15] : <https://www.newquest.fr/wp-content/uploads/2020/09/services-clinique-hopital-connecte-1280x720.jpg>
- [16] : https://img.lovepik.com/photo/40019/8829.jpg_wh860.jpg

- [17] : <https://p8v9g2h2.rocketcdn.me/wp-content/uploads/2021/02/IoT-in-Agriculture-1.jpg>
- [18] : <https://www.parkeagle.com/2018/05/12/what-is-smart-parking/>
- [19] : <https://3.imimg.com/data3/OR/LN/MY-9473625/secure-parking-1-500x500.jpg>
- [20] : https://fr.wikipedia.org/wiki/Ville_intelligente
- [21] : https://which-50.com/wp-content/uploads/2016/11/iStock_96327189_MEDIUM1.jpg
- [22] : <https://www.intelligenzaartificiale.it/internet-of-things/>
- [23] : <https://d1fmx1rbmqrrr.cloudfront.net/zdnet/i/edit/ne/2017/04/iotent.jpg>
- [24] : <https://www.robotshop.com/media/files/PDF/ArduinoMega2560Datasheet.pdf>
- [25] : <https://www.microchip.com/wwwproducts/en/ATmega2560>
- [26] : <https://www.arduino.cc/en/Guide>
- [27] : <https://lastminuteengineers.com/esp8266-nodemcu-arduino-tutorial/>
- [28] : <https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet>
- [29] : <https://randomnerdtutorials.com/getting-started-with-esp8266-wifi-transceiver-review/>
- [30] : <https://booteille.github.io/blynk-docs-fr/>
- [31] : <https://components101.com/motors/servo-motor-basics-pinout-datasheet>
- [32] : <https://www.servocity.com/servo-faqs/>
- [33] : <https://lastminuteengineers.com/servo-motor-arduino-tutorial/>
- [34] : <https://www.electronicshub.org/ir-sensor/>
- [35] : <https://osoyoo.com/2018/12/21/arduino-lesson-ir-obstacle-avoidance-module/>
- [36] : <https://www.playembedded.org/blog/detecting-obstacle-with-ir-sensor-and-arduino/>
- [37] : <https://www.electroduino.com/ir-infrared-flame-sensor-module/>
- [39] : <https://www.electroduino.com/ir-infrared-flame-sensor-module/>
- [38] : <https://www.gazdetect.com/installations-fixes/detecteurs-optiques-de-flammes/>
- [40] : <https://lastminuteengineers.com/i2c-lcd-arduino-tutorial/>
- [41] : <https://rfid.ooreka.fr/comprendre/systeme-rfid>

[42] : <https://lastminuteengineers.com/how-rfid-works-rc522-arduino-tutorial/>

[43] : <https://www.timcod.fr/solutions/tracabilite-technologies/rfid/>

[44] : https://www.sonelec-musique.com/electronique_theorie_buzzers.html

[45] : AFTIS Fatah, AKKOUCHE Merzouk. Conception et simulation d'un multi-afficheur pour une ligne de production. Bejaïa : Université A/MIRA Bejaïa, Faculté de Technologie, 2014-2015.

[46] : http://blewando.fr/elv/Terminale/proteus/chap1/chap_1.htm

Résumé

Ces dernières années, le parc automobile a connu une expansion très importante, cette augmentation a créé lui des principaux problèmes de la ville intelligent qui est la congestion.

Dans ce mémoire, nous avons réalisé un parking intelligent (smart parking) en utilisant la technologie des objets connectés-IOT. Ce système permet de faciliter la gestion des parkings et d'aider les conducteurs de trouve un emplacement de stationnement.

Pour concevoir notre prototype nous avons utilisé une carte Arduino méga afin de contrôler les différents périphériques tels que le servomoteur, l'écran LCD et le système d'alarme (Buzzer).

Ainsi, pour une gestion optimale et efficace nous avons réalisé un circuit électronique en utilisant la carte ESP8266. Ce système nous permet de communiquer les données et de visualiser les places libres/occupées sur l'écran d'un smart phone via l'application BLYNK.

Mots clés

Automobile, ville intelligent, parking intelligent, technologie des objets, Arduino méga, ESP8266, l'application BLYNK.

Abstract

Recently, one of the most important issues in a smart city is congestion problems.

In this project, we have realized a smart parking using the Internet of things-IOT. This system allows to facilitate the management of parking spaces and to help drivers to find a parking place.

To design our prototype we used an Arduino mega board to control the different peripherals such as the servo motor, the LCD screen and the alarm system (Buzzer).

Also, for an optimal and efficient management we have made an electronic circuit using the ESP8266 board. This system allows us to communicate the data and to visualize the free/occupied places on the screen of a smart phone via the BLYNK application.

Keywords

Smart city, smart parking, Internet of things, Arduino mega, ESP8266, the BLYNK application.

ملخص

في السنوات الأخيرة، زاد استخدام السيارات بشكل كبير جدًا، وقد أدت هذه الزيادة الى مشكلة كبيرة في المدينة الذكية وهي الازدحام.

في هذه الأطروحة، اقترحنا مرآب سيارات ذكي باستخدام تقنية إنترنت الأشياء (IOT). يساعد هذا النظام على تسهيل إدارة مواقف السيارات ومساعدة السائقين للوصول إلى أماكن وقوف السيارات بسهولة من خلال معرفة المكان الشاغر.

استخدمنا لوحة Arduino Mega لفتح وإغلاق الحاجز وكذلك عرض عدد المساحات الشاغرة والفارغة. وبالإضافة إلى ذلك، فإنه ينشط نظام الإنذار في حالة الحريق.

يمكن الكشف عن مكان وقوف السيارات باستخدام مستشعر الأشعة تحت الحمراء الذي يتصل بلوحة ESP8266 (NodeMCU) التي تمت برمجتها من قبل Arduino IDE. بالإضافة إلى ذلك، يمكن للمستخدمين الوصول إلى معلومات أماكن وقوف السيارات باستخدام الهاتف الذكي عبر تطبيق BLYNK.

الكلمات المفتاحية

المدينة الذكية، مواقف السيارات الذكية، إنترنت الأشياء، Arduino mega، ESP8266، تطبيق BLYNK.