

ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Rapport du projet Smart Parking

Remerciement

Notre gratitude et nos Remerciements vont à notre fidèle professeurde sa complaisance à répondre à toutes les questions liées à la compréhension du cours et à la réalisation de certaines fonctionnalités.

Nous vous remercions d'avoir partagé avec nous votre passion pour l'enseignement. Nous avons grandement apprécié votre soutien tout au long du semestre.

Table des matières

Remerciement	1
Table des matières	2
Introduction.....	4
Introduction.....	5
1. La carte de Arduino	5
a. Le type de la carte Arduino : MEGA	5
b. Composantes de la carte Arduino Mega	6
2. Servo-moteur:.....	6
a. Principe de fonctionnement d'un servo-moteur.....	6
3. Le module LED	6
4. Cristal liquide d'affichage (LCD).....	7
a. Description de l'écran LCD.....	7
5. Capteur Infrarouge:.....	7
a. Principe de fonctionnement:.....	7
6. Capteur de gaz.....	8
a. Fonctionnement	8
7. La carte MCU NODE ESP8266	8
Conclusion	8
Introduction.....	9
1. Simulation des différents circuits:	9
a. Simulateur Isis Proteus:	9
b. Langage Arduino.....	9
2. Schéma de parking intelligent:	10
a. Premier pas : utiliser un servomoteur.....	10
b. Deuxième pas : utiliser un capteur IR SENSO	11
c. Troisième pas : simulation du parking intelligent	12
Conclusion	13
1. Réalisation du parking intelligent:	14
a. Parking intelligent sur Isis Proteus	14
b. Réalisation de parking intelligent:.....	15
Conclusion	15
Conclusion	16

Table des figures

Figure 1: la carte Arduino Mega	5
Figure 2: Servomoteur "9 grammes"	6
figure 3:Module LED RGB	6
figure 4:Ecran LCD AVEC 16*2	7
figure 5:Capteur infrarouge.....	7
figure 6:Simulation des différent circuits	9
figure 7:Simulation des différent circuits:	10
figure 8:montage reliant un servo-moteur et un Arduino.....	11
figure 9:montage reliant un capteur IR SENSOR et un Arduino	13
figure 10: montage reliant les défèrent composantes du parking intelligent.....	14
figure 11:smart parking	15

Introduction

Aujourd'hui, beaucoup d'états comme la Chine et les États-Unis investissent des milliards de dollars pour le convoi du développement exponentiel des machines intelligentes comme dans le domaine militaire où on entend parler des drones qui s'avèrent être très efficaces, ou dans le domaine industriel où on utilise de nos jours les robots de soudure et de peinture notamment dans l'automobile.

Dans ce sens le smart parking est l'aménagement des parkings grâce aux nouvelles technologies, la question que l'on peut se poser est : pourquoi le smart parking?

Le smart parking a été développé pour diverses raisons. Depuis plusieurs années, les villes ont remarqué que leurs conducteurs avaient de réels problèmes pour trouver une place de parking facilement. Il y a peu de place et le délai de stationnement est souvent mal adapté. Cela entraîne un bouchonnement des villes, d'après l'étude du professeur Donald Shoup, 8% à 74% des zones de bouchons en villes sont provoquées par les automobilistes cherchant un stationnement. Un problème de pollution se pose également, les automobilistes qui tournent dans la ville pour chercher une place de parking polluent la ville sans s'en rendre compte. Le temps de recherche entraîne aussi l'énervement des usagers ce qui n'aide pas à trouver une place, en effet 1 milliard d'heures sont perdues tous les jours pour chercher une place de parking dans le monde. Il a été calculé que dans les grandes villes comme Sydney, New York ou Londres, le temps moyen pour trouver une place se situe entre 3.5 et 14 minutes ¹ ce qui peut être largement diminué avec le smart parking. De plus, les moyens de paiement ne sont pas toujours pratiques pour les utilisateurs. On rencontre aussi d'autres problèmes plus spécifiques comme par exemple retrouver sa place dans un parking souterrain (parking sombre, voitures identiques, espace très vaste...).

Chapitre I : Matériaux et composants

Introduction

Le choix des matériaux est une tâche fondamentale et très complexe, et pour réussir à bien l'aborder, on a posé un groupe de critères qu'on devait les satisfaire pour bien réussir notre choix. Les critères imposés sont les suivants :

- Au niveau de matériau : caractéristiques mécaniques, esthétiques, thermiques, électriques, économiques, environnementales, physiques.
- Au niveau du procédé : volume, masse, géométrie, taille de la série, caractéristiques économiques et environnementales.

1. La carte de Arduino :

Une carte Arduino est une petite carte électronique équipée d'un microcontrôleur. Le microcontrôleur permet, à partir d'événements détectés par des capteurs, de programmer et commander des actionneurs ; la carte Arduino est donc une interface programmable. Elle peut être programmée pour analyser et produire des signaux électriques, de manière à effectuer des tâches très diverses comme le contrôle des appareils domestiques, éclairage, chauffage, le pilotage d'un robot, de l'informatique embarquée, etc.

a. Le type de la carte Arduino : MEGA

C'est la carte idéale pour découvrir l'environnement ARDUINO. Elle permet à tout débutant de se lancer dans tous ses premiers petits projets. Comme c'est la carte la plus utilisée, il est très facile de se référer aux tutoriels très nombreux sur le net et ainsi de ne pas rester seul dans son exploration. Sa simplicité devient par contre un handicap lorsqu'il s'agit de multiplier les périphériques, de manipuler des algorithmes lourds ou d'interagir avec les OS / Android pour lesquels d'autres cartes Arduino sont plus adaptées.



Figure 1: la carte Arduino Mega

b. Composantes de la carte Arduino Mega :

- Prise jack : permet de brancher une alimentation (pile, batterie, adaptateur au centre Vin 7 à 12 V)
- Microcontrôleur : stocke le programme et l'exécute.
- Entrées analogiques : permet de brancher des capteurs et des détecteurs analogiques.
- Connexion USB (Universal Serial Bus) : Permet d'alimenter la carte en énergie électrique (5V), et de téléverser le programme dans la carte.
- Entrées et sorties numériques (Digital) : Permet de brancher des actionneurs.

2. Servo-moteur:

Un servo-moteur est un type de moteur électrique. C'est un dispositif typiquement utilisé en modélisme pour, par exemple, contrôler la d'une voiture télécommandée. Sur un servo-moteur, l'angle de l'axe reste fixé dans une position et peut varier entre 0 et 180° en fonction du signal envoyé.



Figure 2: Servomoteur "9 grammes"

a. Principe de fonctionnement d'un servo-moteur:

La plupart des servomoteurs sont commandés par l'intermédiaire d'un câble électrique à trois fils qui permet d'alimenter le moteur et de lui transmettre des consignes de position sous forme d'un signal codé en largeur d'impulsion plus communément appelé PWM. Cela signifie que c'est la durée des impulsions qui détermine l'angle absolu de l'axe de sortie et donc la position du bras de commande du servomoteur. Le signal est répété périodiquement, en général toutes les 20 millisecondes, ce qui permet à l'électronique de contrôler et de corriger continuellement la position angulaire de l'axe de sortie, cette dernière étant mesurée par le potentiomètre.

3. Le module LED:

Une LED RGB est une LED qui peut s'allumer dans toutes les couleurs en fonction de trois couleurs: le rouge le vert et le bleu. La LED RGB a 4 broches: une commune à l'ensemble des LEDs et une pour chaque couleur de la LED. La broche commune pourra, selon les modèles, être le + (anode commune) ou le - (cathode commune). Ici la LED RGB a pour broche commune la cathode (borne-). Certaines LED RGB intègrent déjà les résistances de limitation de courant dans des modules.

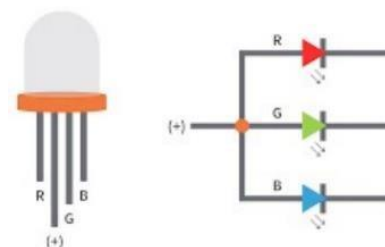


figure 3:Module LED RGB

4. Cristal liquide d'affichage (LCD):

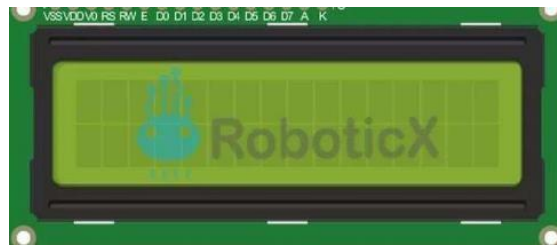


figure 4: Ecran LCD AVEC 16*2

a. Description de l'écran LCD:

Ce type d'écran est le plus commun parmi les passionnés d'électronique et les projets électroniques, car ils sont disponibles à bas prix et faciles à programmer, et ce type est disponible en différentes tailles et couleurs, avec 16x2 et les mois entre eux. 20 × 2, 20 × 4 et autres. Nous entendons par 16 × 2, c'est-à-dire, il y a deux lignes et chaque ligne composées de 16 caractères. Dans notre cas on a utilisé un écran à cristaux liquides 16*2 qui est caractérisé par l'utilisation du system I2C opération et contrôle à travers deux fils seulement, ce qui est une énorme provision pour les prises du panneau de contrôle utilise dans notre projet et facilite le processus de connexion de fonctionnement.

5. Capteur Infrarouge:

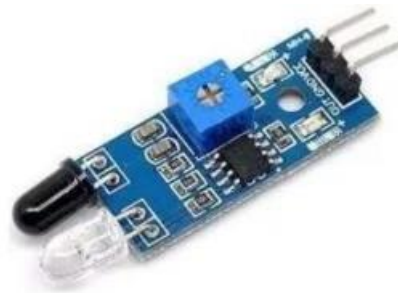


figure 5: Capteur infrarouge

Le capteur d'évitement d'obstacles à capteur infrarouge IR est une solution peu coûteuse pour la détection d'évitement pour la robotique, la voiture intelligente et d'autres utilisations électroniques. Dans cette leçon, nous allons vous montrer comment fonctionne le capteur d'évitement d'obstacles et comment l'utiliser avec la carte ARDUINO MEGA. Ce dernier contient un émetteur IR et récepteur IR. La responsabilité de l'émetteur IR ou de l'émetteur infrarouge est la transmission des ondes infrarouges tandis que le travail du récepteur IR est de recevoir ces ondes infrarouges. Le récepteur IR envoie constamment des données numériques sous la forme de 0 (LOW) ou 1 (HIGH) à la broche Vout du capteur.

a. Principe de fonctionnement:

L'émetteur infrarouge envoie un signal infrarouge qui, en cas de surface réfléchissante (par exemple de couleur blanche), rebondit dans certaines directions, y compris celle du récepteur infrarouge qui capte le signal détectant l'objet. Lorsque la surface est absorbant, le signal IR n'est pas réfléchi et l'objet ne peut pas être détecté par le capteur. Ce résultat se produirait même si l'objet est absent.

6. Capteur de gaz:

Un capteur de gaz est défini comme un composant dont au moins une de ses propriétés physiques change quand il est soumis à un changement d'environnement gazeux. D'une manière générale, un capteur est composé de deux éléments principaux : l'élément sensible et le transducteur :

- L'élément sensible est le cœur du capteur, sur lequel se passe la réaction avec l'espèce gazeuse.
- Le dispositif permettant la conversion du résultat de la réaction entre le gaz et l'élément sensible en un signal facilement mesurable (signal électrique, optique ou acoustique).

a. Fonctionnement

Le capteur de fumée MQ-2 est sensible à la fumée et aux gaz inflammables suivants:

- LPG, Butane, Propane, Méthane, L'alcool, Hydrogène...

La tension émise par le capteur change en fonction du niveau de fumée / gaz qui existe dans l'atmosphère. Le capteur délivre une tension proportionnelle à la concentration de fumée / gaz. En d'autres termes, la relation entre la tension et la concentration de gaz est la suivante:

- Plus la concentration de gaz est élevée, plus la tension de sortie est élevée
- Plus la concentration de gaz est faible, plus la tension de sortie est basse

7. La carte MCU NODE ESP8266 :

Module basé sur un ESP8266 cadencé à 80 MHz et exécutant le firmware open source NodeMCU. Cette carte se programme via l'IDE Arduino et est compatible avec les scripts LUA. Ce microcontrôleur dispose d'une interface WiFi idéale pour les objets connectés. Des connecteurs latéraux mâles et femelles permettent d'enficher le module sur une plaque de montage rapide. L'interface sans fil Wifi permet la création de point d'accès sans fil, l'hébergement d'un serveur, la connexion à internet et le partage des données par exemple. Le module se programme directement à partir de l'IDE Arduino (installation d'une extension nécessaire) et nécessite un cordon. Son implantation le rend compatible avec les plaques de connexions rapides.

Conclusion :

Le choix des matériaux était la phase la plus délicate dans notre projet. En effet, il ne portait pas seulement sur un aspect purement technique répondant à des exigences fonctionnelles, mais aussi à des attentes relevant des préférences des utilisateurs dans le cadre d'un marché spécifique. En gros, le choix des matériaux a été analysé sous l'angle de l'ingénierie des matériaux, du temps de réponse, sans oublier enfin le côté esthétique (le design industriel).

Chapitre 2: simulation

Introduction:

Notre simulation de parking intelligent était avant l'avant-dernière tâche que nous devions faire, les études étaient déjà terminées et les matériaux étaient déjà en place, il ne manquait plus qu'une structure pour accueillir ces appareils puis vérifier si les mesures que nous avons faites concilieraient les côtés techniques avec le côté esthétique.

1. Simulation des différents circuits:

a. Simulateur Isis Proteus:

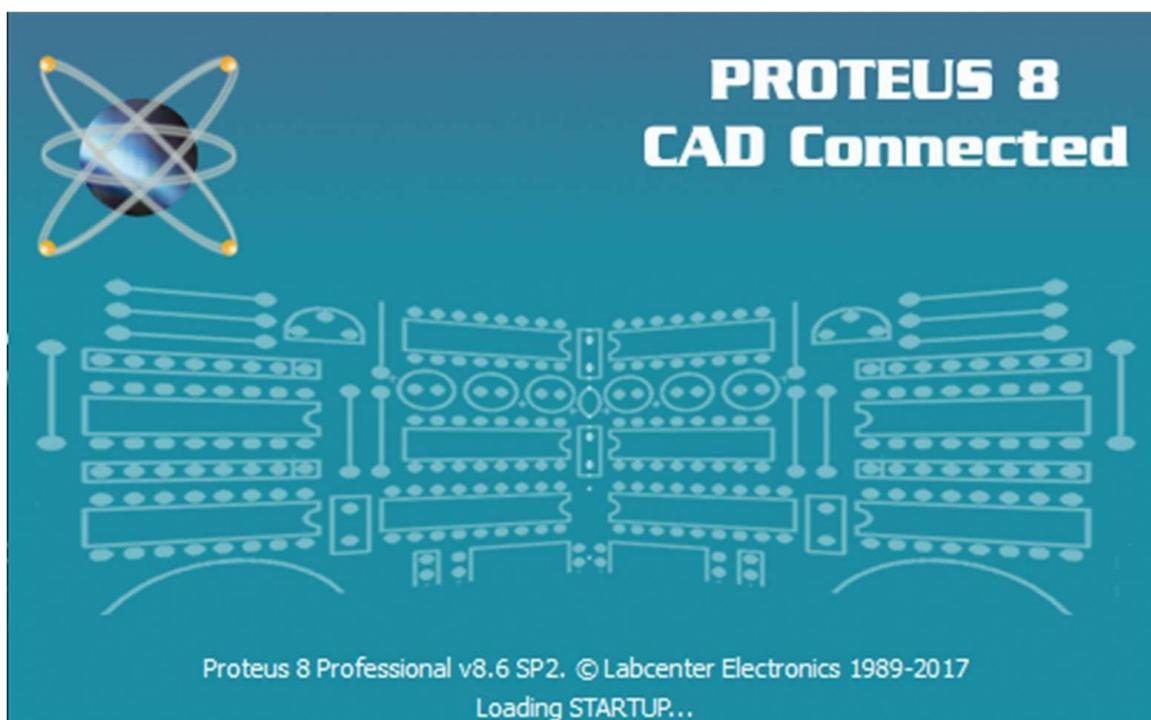


figure 6: Simulation des différents circuits

Isis Proteus est une suite logicielle destinée à l'électronique. Développé par la société Labcenter Electronics, elle est principalement connue pour éditer des schémas électriques. Par ailleurs le logiciel permet également de simuler ces schémas.

Les circuits électriques conçus grâce à ce logiciel peuvent être utilisés dans des documentations car le logiciel permet de contrôler la majorité de l'aspect graphique des circuits.

b. Langage Arduino:

Définition:

Le langage Arduino est inspiré de plusieurs langages. On retrouve notamment des similarités avec le C, le C++. Le langage impose une structure particulière typique de l'informatique embarquée. La programmation se fait de manière séquentielle c'est-à-dire une suite d'instructions élémentaires.

Interface d'utilisateur:

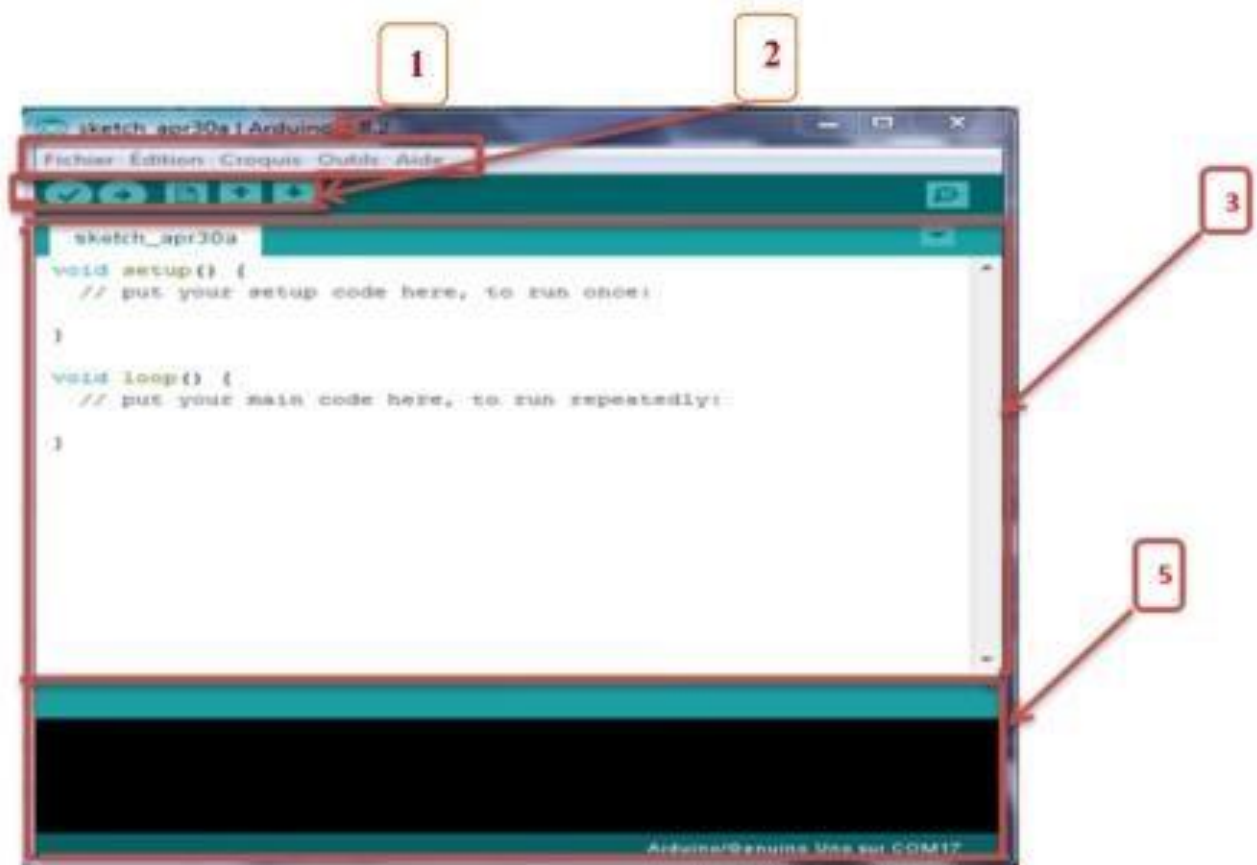


figure 7:Simulation des différent circuits:

- Le cadre numéro 1 : ce sont les options de configuration du logiciel
- Le cadre numéro 2 : il contient les boutons qui vont nous servir lorsqu'on va programmer nos cartes
- Le cadre numéro 3 : ce bloc va contenir le programme que nous allons créer
- Le cadre numéro 4 : celui-ci est important, car il va nous aider à corriger les fautes dans notre programme. C'est le débogueur.

Un programme (ou "sketch") Arduino est constitué de 2 fonctions distinctes :

- La fonction de configuration "void setup" exécutée une seule fois au lancement du Programme. On appelle aussi cette fonction : fonction d'initialisation.
- La fonction "void loop" qui est ensuite exécutée indéfiniment en boucle.

2. Schéma de parking intelligent:

Voici le schéma de notre robot: photo de projet

a. Premier pas : utiliser un servomoteur:

Afin de bien maîtriser la programmation et l'utilisation de la carte Arduino, on va tout d'abord commencer par des programmes simples dont le but est de réaliser un programme optimal pour

notre parking intelligent. Commençons au début par un programme assez simple, dans lequel on cherche à sert à Faire tourner le bras du servo-moteur.

Le code Arduino utilisé :

```
//mettre l'angle du bras de votre servomoteur à 90°

#include <Servo.h> Servo monServo; // création d'un nouveau servomoteur

void setup() {

    monServo.attach(2, 1000, 2000);

    monServo.write(90);

}
```

Commentaires :

- On a utilisé la fonction `monServo.attach(2, 1000, 2000)` pour Attacher le servo-moteur au "pin" 2 et fixer l'angle minimal et maximal du tournage respectivement sur les angles correspondants aux valeurs 1000 et 2000.
- La fonction `monServo.write(90)` sert à Faire tourner le bras du servo moteur pour un angle de 90.

b. Deuxième pas : utiliser un capteur IR SENSO:

Le schéma réaliser sur ISIS:

Afin de mettre en évidence le fonctionnement du capteur IR SENSOR, on réalise la simulation suivante sur Proteus, on relie le capteur a la carte Arduino et une LED. On va lire les résultats indiqués par le capteur avec un LED.

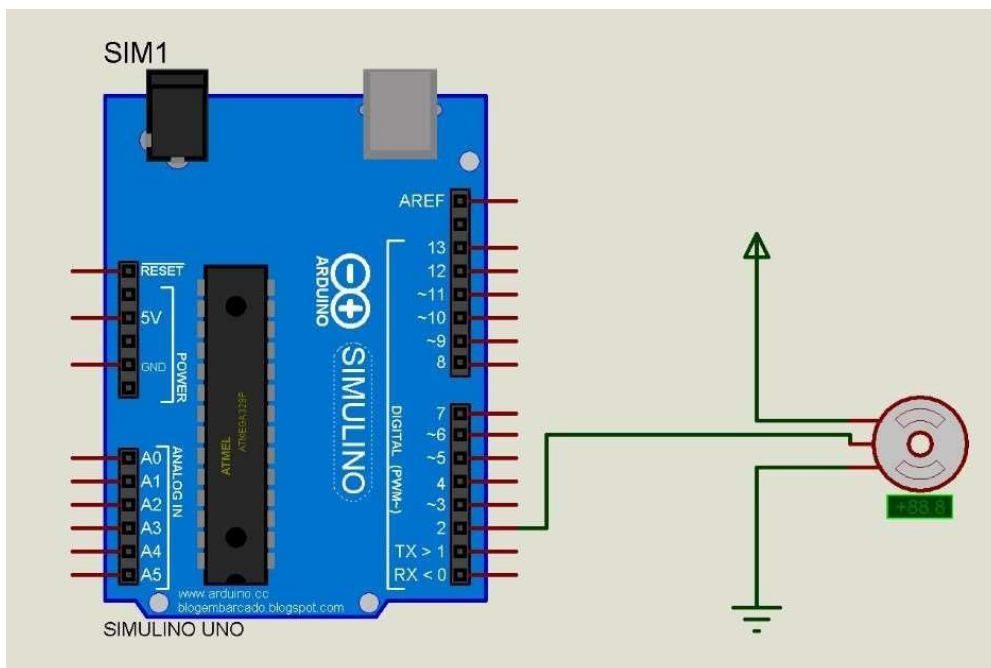


figure 8:montage reliant un servo-moteur et un Arduino

Le code Arduino utilisé

```
int LED = 13;
int IR = 2;
int IRETAT;
void setup () {
    // put your setup code here, to run once:
    pinMode(LED,OUTPUT);
    pinMode(IR,INPUT);
}
void loop() {
    // put your main code here, to run repeatedly:
    IRETAT=digitalRead(IR);
    if(IRETAT==0){
        digitalWrite(LED,HIGH);
    }
    else {
        digitalWrite(LED,LOW);
    }
}
```

Commentaires :

- L'instruction const int LED = 13 sert à attacher la constante LED au port 3 de la carte Arduino.
- L'instruction const int IR à attacher la constante IR au port 2 de la carte Arduino.
- L'instruction pinMode(LED , OUTPUT) sert à fixer le port 13 pour être une sortie.
- L'instruction pinMode(IR , INPUT) sert à fixer le port 2 pour être une entrée.
- L'instruction digitalWrite(LED , LOW) sert à mettre le port 13 (LED) au niveau bas (0V)
- L'instruction digitalWrite(LED , HIGH) sert à mettre le port 13 au niveau haut (5V)

c. Troisième pas : simulation du parking intelligent

Après avoir étudié séparément et choisi minutieusement les composants contribuant dans la construction de notre parking intelligent, on les a assimilés tous ensemble et voilà ce qu'on a obtenu :

Schéma électrique du parking intelligent avant l'intégration de la carte mcu node:

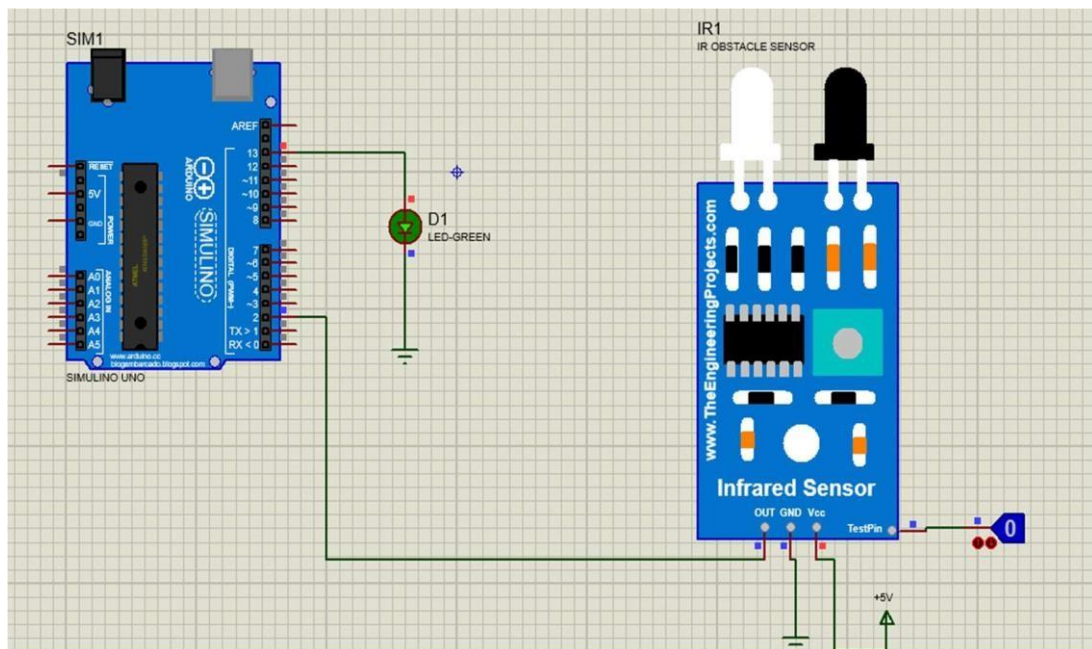


figure 9:montage reliant un capteur IR SENSOR et un Arduino

Conclusion :

Dans ce chapitre, nous avons cité en détail les différentes étapes d'un schéma bloc du circuit de Maquette d'un parking avec barrière automatique.

Notre objectif dans cette partie de ce travail était de réaliser tous les composants de notre projet c'est-à-dire le servomoteur avec le lisse, l'écran LCD et les différents capteurs. Ensuite, la connexion et programmation les composants principaux avec le microcontrôleur Arduino; dans laquelle nous avons tester l'entrée et sorti des véhicules compter et décompter les nombres des véhicules avec afficher le tous les séquences dans l'écran.

Chapitre 3: Réalisation

1. Réalisation du parking intelligent:

a. Parking intelligent sur Isis Proteus:

Avant de réaliser ce projet il suffit de réaliser sur le simulateur le montage suivant :

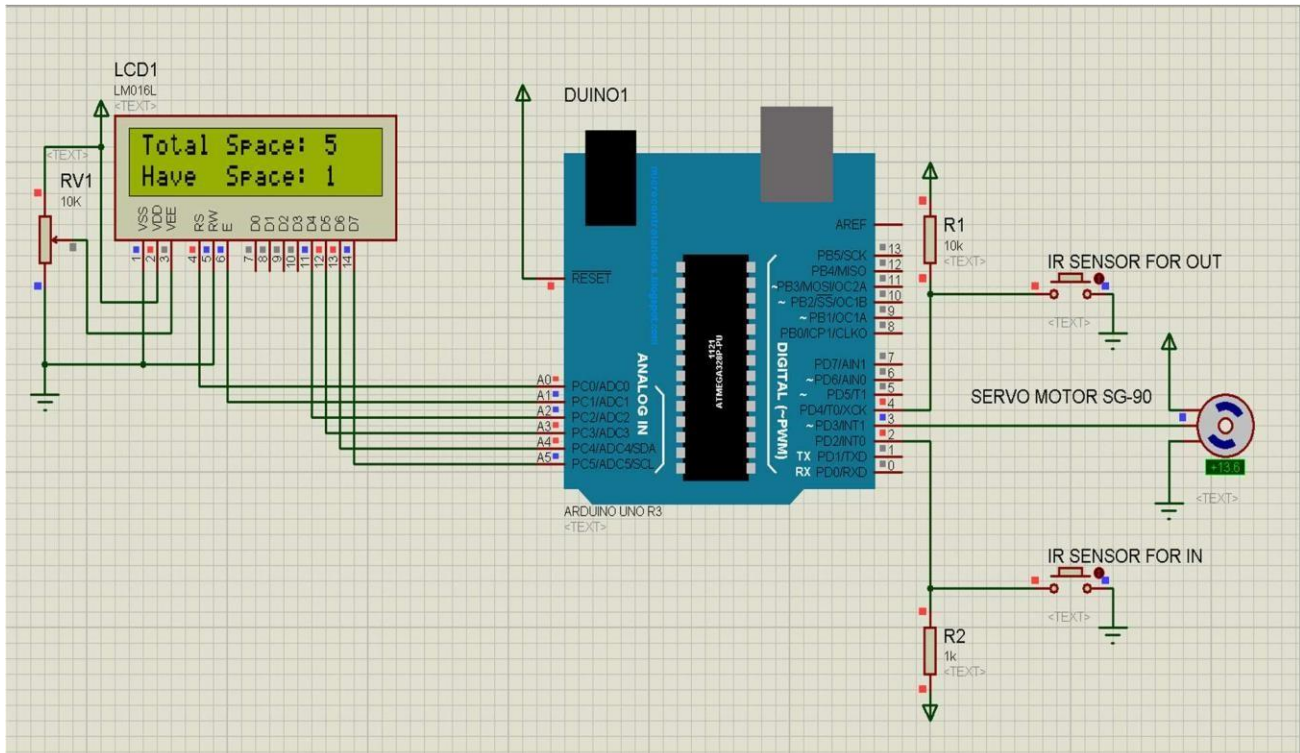


figure 10: montage reliant les différent composantes du parking intelligent

b. Réalisation de parking intelligent:

Voici une images de notre projet final :



figure 11:smart parking

Conclusion:

Après avoir intégré la carte mcu node on peut maintenant naviguer dans le site web et savoir l'état de notre parking et même réserver.

Conclusion

Pour répondre aux problèmes de stationnement, le parking intelligent représente une alternative au garage classique, il tire un grand profil de l'espace disponible. Le système de parking constitue la solution idéale pour automatiser le processus d'entrée et de sortie. La réalisation d'un parking intelligent est présentée dans notre travail.

Le premier chapitre: traite des généralités sur l'ARDUINO et les servomoteurs. Ces deux composants sont très importants dans le domaine de la microélectronique et les systèmes embarqués et automatiques. L'ARDUINO et les servomoteurs peuvent être utilisés dans des dizaines d'applications liées à cette technologie. Pour cette raison, Nous avons représenté les différents types de ces composants ainsi que le mécanisme de fonctionnement de chacun d'eux.

Aux deuxièmes chapitres: Méthode de connexion et programmation des composants principaux avec le Arduino ; dans laquelle nous avons réalisé le système global (Tester l'entrée et sorti des véhicules, monte et descend la lisse, compter et décompter les nombres des véhicules avec afficher le tous les séquences dans L'écran).

Aux troisièmes chapitre : nous avons réalisé une application de parking intelligent qui utilise ces composants électroniques, sans oublier de donner le câblage et codage utilisés dans cette manipulation.