



Ministère de l'enseignement supérieur
et de la Recherche Scientifique



Sagemcom

Département Télécommunications

MEMOIRE DE PROJET DE FIN D'ETUDES

*Présenté pour l'obtention du
Licence informatique embarqué et IOT*

Sujet : conception et réalisation d'un testeur de haut-parleur

Réalisé par
Elbez slaheddine

Encadré par :

Mme. Houda Jouini Chaabouni (encadreur universitaire)
MS. Fouzi Handani (encadreur de la société)

Résumé

Ce travail s'inscrit dans le cadre de réalisation du projet de fin d'études en vue de l'obtention du diplôme de licence fondamentale en science informatique, délivré par université central de Tunisie

Notre projet s'intitule « Testeur de haut-parleur», il concerne la conception et la réalisation d'un système autonome pour le teste des différent model de haut-parleur.

Ce projet nous a été très bénéfique parce qu'il a été pour nous l'occasion d'appliquer les connaissances théoriques acquises tout au long de notre cursus universitaire dans un cadre professionnel réel et certainement une initiation fructueuse à la vie professionnelle en entreprise.

Dédicace

Je dédie ce travail

A mes parents pour leurs affections et leurs prières quotidiennes en vue
d'assurer mon éducation ;

A mes proches et amis pour leurs conseils et encouragements.

A tous personnels de SAGEMCOM qui m'ont aidé à réaliser mon travail
A vous qui nous suivons

Remerciements

A DIEU Tout Puissant, pour m'avoir donné la force, la santé et l'intelligence nécessaire pour accomplir ce travail ;

A mes parents et tous mes frères et sœurs pour leurs volontés, leurs énormes sacrifices et pour l'amour qu'ils ont su me donner tout au long de mon étude ;

A mes encadreurs pour m'avoir soutenu, guidé, encouragé et orienté dans ce travail, je leur suis très reconnaissant ;

A tout le personnel de SAGEMCOM, qui m'a bien aidé à atteindre mon objectif ;

A mes oncles, tantes, cousins et cousines pour la fraternité et pour m'avoir accepté de vous manquer durant l'élaboration du présent travail ;

A tous mes amis, connaissances et collaborateurs pour m'avoir toujours encouragé même pendant les moments les plus difficiles et surtout pour la confiance que vous détenez en moi ;

Enfin, tous ceux que je n'ai pas pu citer.

Table des matières

Introduction générale	1
Chapitre I : contexte générale et étude Préalable	2
I. Introduction	3
II. Présentation de « SAGEMCOM Tunisie »	3
1. Statut et organisation générale	3
2. Domaine d'activités.....	4
3. Différents services de SAGEMCOM.....	4
4. Processus de fabrication standard des HAUT-PARLEUR	5
5. Missions du service d'industrialisation	6
III. Problématique	6
IV. Solution proposée	6
V. Générâtes sur le haut-parleur.....	7
1. Définition.....	7
2. Principe générale de fonctionnement d'un hautparleur	7
3. Différents types de haut-parleur.....	8
VI. Exemple de différent teste haut-parleur	9
1. Test de fonctionnalité	10
2. Teste de qualité de sons.....	11
VII. Conclusion.....	12
Chapitre II : Conception et simulation.....	13
I. Introduction	14
II. Composition et choix des matériels du testeur de haut-parleur.....	14
1. Partie 1	14
2. Partie 2	20
III. Conception du testeur de haut-parleur	22
1. Choix de logiciel	22
2. simulation de la partie transitoire et diode à jonction	24
3. Conception de la partie transitoire et diode a jonction.....	24
4. Conception de la partie de traitement et de commande	26
IV. Conclusion.....	35

Chapitre III : Réalisation.....	36
I. Introduction	37
II. Differnt cas possible	37
III. Premier cas	38
IV. Deuxième cas.....	40
V. Conclusion.....	44
Conclusion générale et Perspectives.....	45
Webographie	46

Liste des figures

Figure.1 : IMPLANTATION DES SITES SAGEMCOM	3
Figure.2 : PRODUITS DE SAGEMCOM	4
Figure.3 : PROCESSUS DE FABRICATION D'UN DECODEUR.....	5
Figure.4 : HAUT-PARLEUR	7
Figure.5 : COMPOSITION GENERALE D'UN HAUT-PARLEUR.....	8
Figure.6 : HAUT-PARLEUR GRAVE.....	8
Figure.7 : HAUT-PARLEUR MEDIUM.....	9
Figure.8 : HAUT-PARLEUR TWEETER.....	9
Figure.9 : TESTE AVEC UNE PILE	10
Figure.10 : TESTE AVEC MULTIMETRE	11
Figure.11 : CAPTEUR DE SONS	11
Figure.12 : RIGHTMARK AUDIO ANALYZER	12
Figure.13 : CHEMIN D'UNITE.....	14
Figure.14 : DOUCHETTE USB	15
Figure.15 : CARTES STM32401	16
Figure.16 : STM32 CUBE IDE	17
Figure.17 : FIREBASE	18
Figure.18 : ESP01	18
Figure.19 : CHEMIN STM32+ESP8266+BASE DE DONNE	19
Figure.20 : CAPTEURS DE SONS	19
Figure.21 : AFFICHEUR LCD 16*2.....	20
Figure.22 : CHEMIN EXPLICATIVE DU BOX	20
Figure.23 : TRANSISTORS 2N3904	21
Figure.24 : DIODE A JONCTION	21
Figure.25 : COMPOSITIONS GENERALES	22
Figure.26 : PROTEUS 8	23
Figure 27 : ALTIUM	23
Figure.28 : SIMULATION DE TRANSISTORS ET DIODE A JONCTION	24
Figure.29 : SCHEMA PCB.....	25
Figure.30 : CONCEPTIONS 2D.....	25
Figure 31 : CONCEPTION 3D.....	26
Figure.32 : LES BROCHES D'UN CABLE USB.....	26

Figure.33 : BRANCHEMENTS STM32 AVEC PORT USB	27
Figure.34 : CONFIGURATIONS STM32.....	28
Figure.35 : PROGRAMMATION DE LA CARTE STM32	28
Figure.36 : BRANCHEMENT DE STM32 AVEC ESP8266	29
Figure.37 : BRANCHEMENTS DE TX RX	29
Figure.38 : CONFIGURATION STM32 AVEC ESP8266	30
Figure.39 : TRANSFERTS DE DONNEES A ESP8266	30
Figure.40 : LA PROGRAMMATION DE LA CARTE ESP8266	31
Figure.41 : CHEMIN DU CARTE STM32 AVEC AFFICHEUR 16*2	31
Figure.42 : CONFIGURATION I2C DE LA CARTE STM32	32
Figure.43 : PROGRAMMATIONS DE LA CARTE STM32 POUR AFFICHER UN MSG RECUPERE DE LA CARTE ESP01	32
Figure.44 : CHEMIN DE LA CARTE STM32 + KY038	33
Figure.45 : ACTIVATION DES TRANSISTORS ET DU CAPTEUR, LIRE VALEUR DU CAPTEUR	34
Figure.46 : LA PROGRAMMATION DU ESP8266 POUR ENREGISTRES LES DONNEES SUR FIREBASE	34
Figure.47 : LA BASE DE DONNEES AVANT ET APRE LE TESTE	35
Figure.48 : CAS POSSIBLE.....	37
Figure.49 : ETAT DU TESTE 1 DU PRODUIT 61940024XY	38
Figure.50 : AFFICHER L'ETAT DU TESTE1 61940024XY	38
Figure.51 : ETAT DU TESTE 1 DU PRODUIT 420093465629XYXY	39
Figure.52 : AFFICHER L'ETAT DU TESTE1 6194019607382XYXY	39
Figure.53 : CODE A BARRE N'EXISTE PAS SUR LA BASE DE DONNES.....	39
Figure.54 : CODE À BARRE D'UN BOX	40
Figure.55 :AFFICHER LE CODE A BARRE ENTRE.....	40
Figure.56 : BASE DE DONNER FIRE BASE.....	40
Figure.57 : TESTE PRECEDENT DE HAUT-PARLEUR NON VALIDE	41
Figure.58 : CODE A BARRE N'EXISTE PAS SUR BASE DE DONNES	41
Figure.59 : TESTE PRECEDENT DE HAUT-PARLEUR VALIDE	41
Figure.60 :ACTIVATION DU CAPTEUR DE SONS.....	42
Figure.61 : ACTIVATION DE LED	42
Figure.62 : TESTE VALIDE	42
Figure.63 : LA MISE A JOUR DE LA BASE DE DONNES.....	43

Figure.64 : TESTE NON VALIDE	43
Figure.65 : LA MISE A JOUR DE LA BASE DE DONNES.....	44
Figure.66 :TESTEUR HAUT-PARLEUR.....	Erreur ! Signet non défini.

Acronymes

RAM	Random-access memory
ROM	Read-only memory
E/S	Port entre sortie
NoSQL	Not Only Structured Quercy Language
SQL	Structured Query Language
wifi	Wireless Fidelity
SOC	system-on-a-chip
TCP/IP	Transmission Control Protocol/Internet Protocol
USB	universal serial bus
PNP	Positive Negative Positive Transistor
I2C	Inter-Integrated Circuit
PCB	printed circuit board
LED	light-emitting diode

Introduction générale

Le développement économique mondial actuel a créé un environnement très dynamique, où les technologies dans tous les domaines sont devenues de plus en plus avancées, par rapport à quelques années seulement auparavant. Cet environnement impose, donc, à toutes les industries, toutes activités confondues, d'être toujours à la page de ce développement, et de suivre toute nouveauté technologique pour stimuler la concurrence entre les industriels du même secteur.

SAGEMCOM (Société d'applications générales d'électricité et de Mécanique) est une société orientée dans le secteur des télécommunications et de l'électronique de défense et de sécurité. Ce secteur est l'un des secteurs les plus sollicité par les contraintes de la concurrence. La société SAGEMCOM, étant l'une des industriels les plus importants dans ce secteur, cherche à être toujours à la hauteur de sa réputation.

En effet, l'un des facteurs déterminant de la capacité concurrentielle d'une entreprise est le procès de production utilisé dans le site.

Consciente de ces orientations et soucieuse de la compétitivité de son organisation, la direction de SAGEMCOM a décidé de chercher une nouvelle solution pour un testeur plus autonome qui assure le bon fonctionnement des haut-parleurs.

Dans le cadre de notre projet de fin d'études, et en relation avec l'environnement industriel, nous sommes amenés à réaliser un testeur qui a d'objectif l'automatisation est l'optimisation des tests de fonctionnement des haut-parleurs.

Nous présentons à travers les différents chapitres de ce rapport la démarche que nous avons suivie pour élaborer le travail demandé :

- Le premier chapitre est consacré à la présentation du contexte générale et étude Préalable.
- Le deuxième chapitre est dédié à la phase de conception et de simulation de notre testeur.
- Le troisième chapitre est destiné à la phase de réalisation de notre projet.

Nous clôturons le rapport avec une conclusion et des perspectives.

Chapitre I : contexte générale et étude Préalable

I. Introduction

Dans ce chapitre nous présentons une étude préalable nécessaire à la définition et à la caractérisation de notre projet. Pour cela, nous commençons par la délimitation du cadre du projet, suivie d'une présentation succincte de l'organisme d'accueil, puis la citation des problèmes à résoudre suivie d'une étude de l'existant, d'une description de la solution proposée et nous terminons par la présentation de la méthodologie adoptée pour le développement et la mise en œuvre la solution proposée.

II. Présentation de « SAGEMCOM Tunisie »

SAGEMCOM est l'une des grandes entreprises Françaises présentes dans le secteur des télécommunications et de l'électronique de défense et de sécurité. Elle est implantée dans plus de 30 pays dont la Tunisie et emploie plus de 72000 personnes.

La figure 1 nous présente les sites industriels, les bureaux d'études et les représentations commerciales mondiaux de SAGEMCOM.

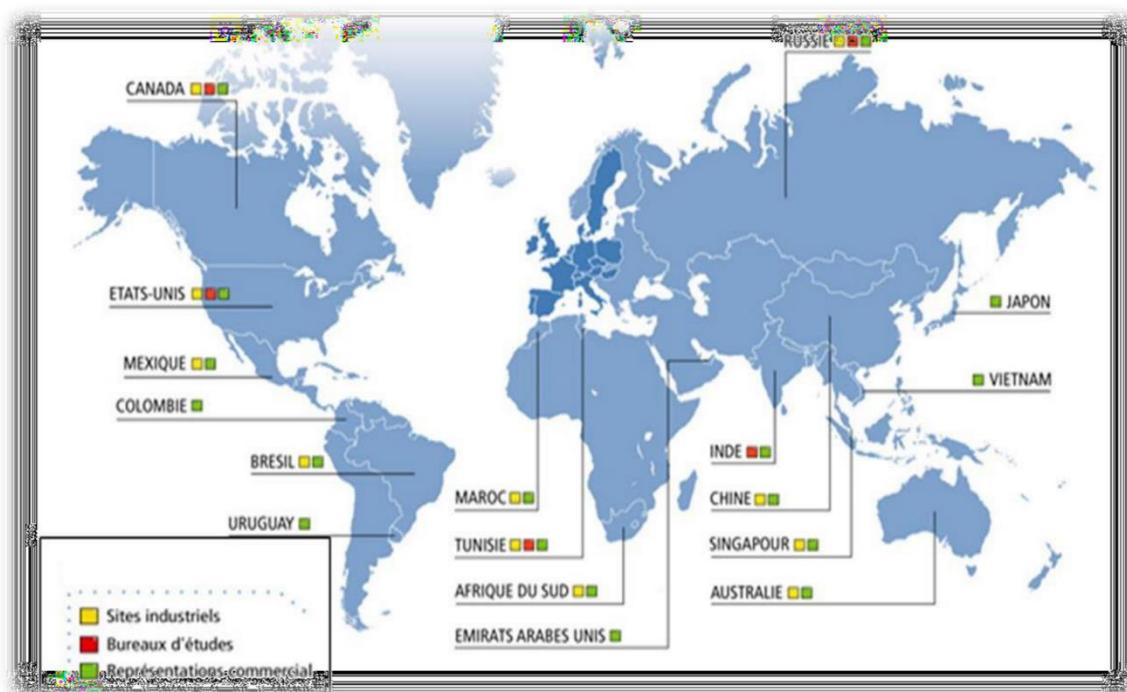


Figure.1 : IMPLANTATION DES SITES SAGEMCOM

1. Statut et organisation générale

SAGEMCOM Tunisie est une Filiale de SAGEMCOM créée en décembre 2002. C'est une société anonyme à responsabilité limitée (SARL).

2. Domaine d'activités

SAGEMCOM est une société totalement exportatrice qui opère dans le secteur de la communication, du partenariat industriel, de l'énergie, du traitement et de la transmission numérique de l'information. Elle fabrique une large gamme de produits en grande et moyenne série tels que :

Les équipements pour la maîtrise de l'énergie électriques : Compteurs d'énergie ;

Les modems et Routeurs ADSL ;

-Les décodeurs TV.

SAGEMCOM Tunisie réalise principalement cinq activités correspondant à 7 unités de fabrication (UF). Elle est passée à 8 à partir du mois de septembre 2006 :

UF Energie : compteurs électriques ;

UF ATR (Activité Terminale Résidentielle) : Les modems et Routeurs ADSL ;

UF STB (Set Top Box): décodeurs numériques.

Dans la figure 2 on présente les différents produits de SAGEMCOM



Figure.2 : PRODUITS DE SAGEMCOM

3. Différents services de SAGEMCOM

Les différents services de SAGEMCOM Tunisie sont les suivants :

Le service qualité et environnement : le management de la qualité et de l'environnement.

Le service d'achat : Il assure les achats des divers besoins.

Le service maintenance : Il est chargé de maîtriser le taux de disponibilité des moyens et minimiser le coût d'exploitation des équipements.

Le service méthode et procédés : Il a pour rôle la maîtrise des procédés spéciaux de fabrication et la contribution à l'amélioration de l'industrialisation des produits.

Le service ressources humaines : Il s'occupe des recrutements, de la paie, de la gestion administrative et de la santé des travailleurs.

Le service formation : il assure l'intégration des nouveaux embauchés ainsi que la planification.

Le service industrialisation : Il a comme mission le développement des moyens de test notamment les testeurs fonctionnels, assurer le passage de la phase prototypage à la phase de mise en production des séries de produits.

4. Processus de fabrication standard des HAUT-PARLEUR

Le processus de fabrication est réparti sur trois zones

Zone CMS : Monter les composants sur la surface du cuivre imprimée et Usinée « CIU » d'où L'obtention du cuivre imprimé usiné « CIE ».

Zone Test : Tester les CIE obtenus « Test in-situ : la présence des composants et le bon fonctionnement du groupe des composants / Test fonctionnel / Test wifi ».

Zone Intégration : Obtention de produits finis caractérisés par les soft clients.

Les différentes phases de fabrication des décodeurs sont illustrées par la figure 3 :

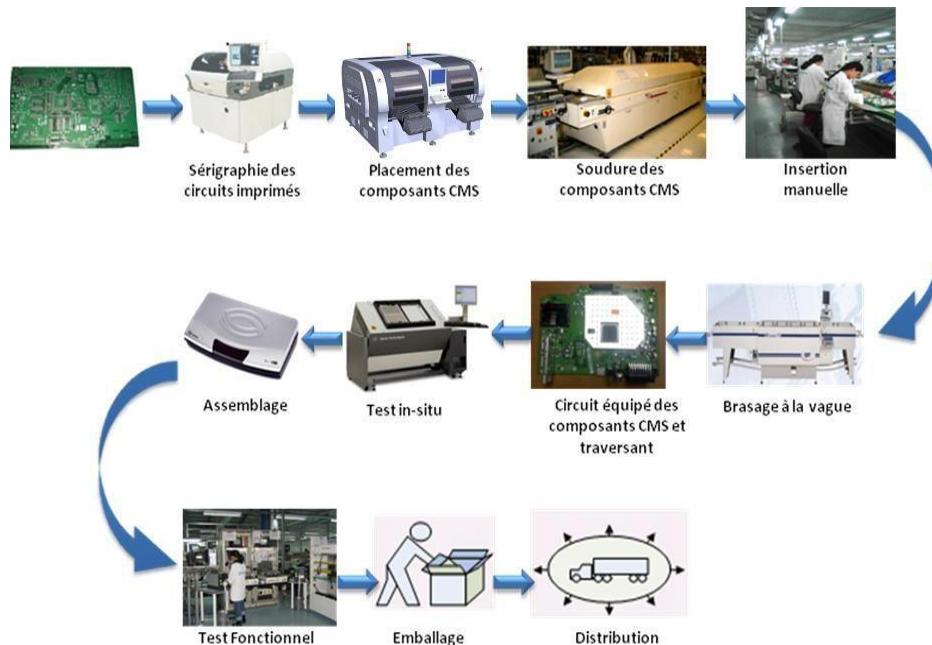


Figure.3 : PROCESSUS DE FABRICATION D'UN DECODEUR

5. Missions du service d'industrialisation

Ce projet de fin d'étude se déroule au sein du service d'industrialisation. Les missions principales de l'industrialisation d'un nouveau produit sont :

- L'installation et la qualification du processus et des nouveaux flux de production ;
- L'installation et la qualification des moyens de fabrication et de test ;
- La fabrication des prototypes et des préséries ;
- La facilitation du démarrage des produits en masse production ;
- La participation aux choix technologiques et la validation des nouvelles sources.

III. Problématique

Afin d'assurer que tous produit sortant de l'entreprise soit tester pour assurer leur qualité, l'entreprise doit être capable de tester ses produits fabriqués (Haut-parleurs et microphones ...) en visualisant leurs fonctionnalité.

Dans ce contexte, le service d'industrialisation est responsable de gérer tous les problèmes et les anomalies qui peuvent influencer la rentabilité de la ligne dont l'efficacité des produits acoustiques que nous allons entamer au cours de ce projet fin d'études.

La problématique à traiter se matérialise lors du déroulement de la phase de test qui vient après la phase d'industrialisation du projet. Le test des produits de Sagemcom se faisait manuellement par l'ingénieur de Sagemcom Tunisie et cela prendra un temps de test trop long. De plus, le nombre de produit testé par jour est très élevé donc il y a un risque de faire confondre les numéros de séries des produits.

C'est dans ce cadre que notre projet de fin d'études s'est inscrit, intitulé « Réalisation d'un testeur de haut-parleur : du testeur du bon fonctionnement des haut-parleurs ».

De l'évolutivité et la maintenabilité de la solution.

IV. Solution proposée

Pour l'automatisation de la procédure de teste des haut-parleurs nous avons proposé de développer un système automatique de teste pour cela on doit faire :

- Vérifier l'état du teste précédent pour chaque modèle de haut-parleur.
- Exécute le teste de fonctionnalité d'une manier automatique.
- Enregistre le résultat de teste sur notre base de données

V. Générâtes sur le haut-parleur

1. Définition

Un haut-parleur est un transducteur électromécanique destiné à produire des sons à partir d'un signal électrique. On emploie parfois aussi ce terme pour désigner l'appareil destiné à l'émission sonore dont il est un des constituants principaux (dans ce cas, se reporter à l'article "enceinte acoustique"). [1]



Figure.4 : HAUT-PARLEUR

2. Principe générale de fonctionnement d'un hautparleur

Le haut-parleur électrodynamique est constitué principalement :

- d'un aimant circulaire ;
- d'une bobine circulaire mobile placée autour d'un des pôles de l'aimant ;
- d'une membrane reliée à la bobine ;
- d'un « saladier » ou support qui contient l'aimant, la bobine et la membrane.

Les fils de la bobine sont connectés à la sortie du haut-parleur. [2]

Comme montre la figure 2

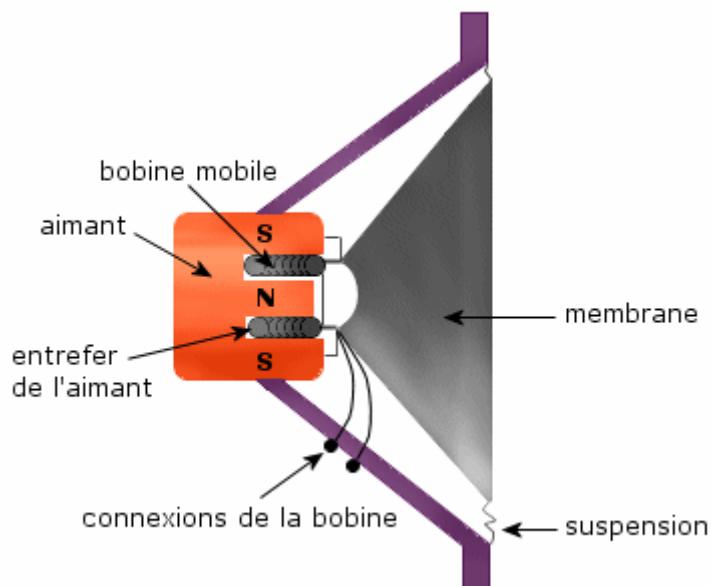


Figure.5 : COMPOSITION GENERALE D'UN HAUT-PARLEUR

Il fonctionne suivant le principe suivant :

- Un moteur transforme l'énergie électrique en énergie mécanique.
- ce moteur transmet cette énergie mécanique à la membrane.
- la membrane transmet l'énergie mécanique à l'air ambiant – d'où le son.

3. Différents types de haut-parleur

Il existe trois types de haut-parleurs :

Un haut-parleur grave ou haut-parleur de grave est un haut-parleur ayant pour fonction de reproduire les basses fréquences (sons graves situés entre 20 et 500 Hz approximativement).[3]



Figure.6 : HAUT-PARLEUR GRAVE

Médium, médium fonctionne dans une plage de fréquences comprise entre 150 Hz et 10 kHz, souvent entre 500 Hz et 5 kHz. Le choix dépend de la conception de l'enceinte acoustique[3]



Figure.7 : HAUT-PARLEUR MEDIUM

Tweeter, est un haut-parleur destiné à reproduire les fréquences sonores élevées, soit les sons aigus². Généralement, les fréquences reproduites sont supérieures à 2 000 Hz et peuvent s'étendre jusqu'aux ultrasons [3] ;

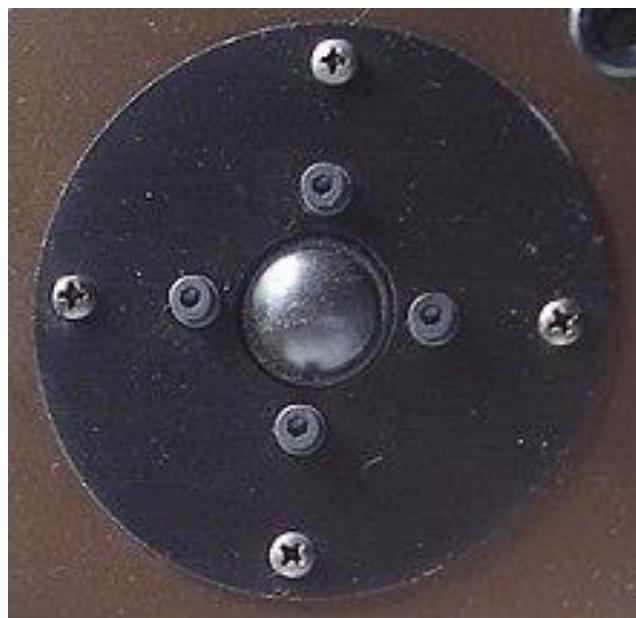


Figure.8 : HAUT-PARLEUR TWEETER

VI. Exemple de différent teste haut-parleur

Il est nécessaire pour un fabricant de haut-parleur de tester ces produits pour assurer leur bon fonctionnement et leur qualité.

Il y a plusieurs types et manières afin de tester leurs produits.

1. Test de fonctionnalité

Pour tester le bon fonctionnement de notre haut-parleur il y a plusieurs options :

Option 1 : assurer que tous les composants d'un haut-parleur sont bien assemblés et qu'il n'y a pas d'anomalies sur chaque composant exemple :

- Il n'y a pas de rupture de cuivre dans la bobine.
- La membrane est bien fixée sur la suspension.

Option 2 : brancher une pile 9V aux bornes du haut-parleur, avec la borne positive sur la borne positive de la pile, et la borne négative sur la borne négative de la pile. Comme le montre la figure 9

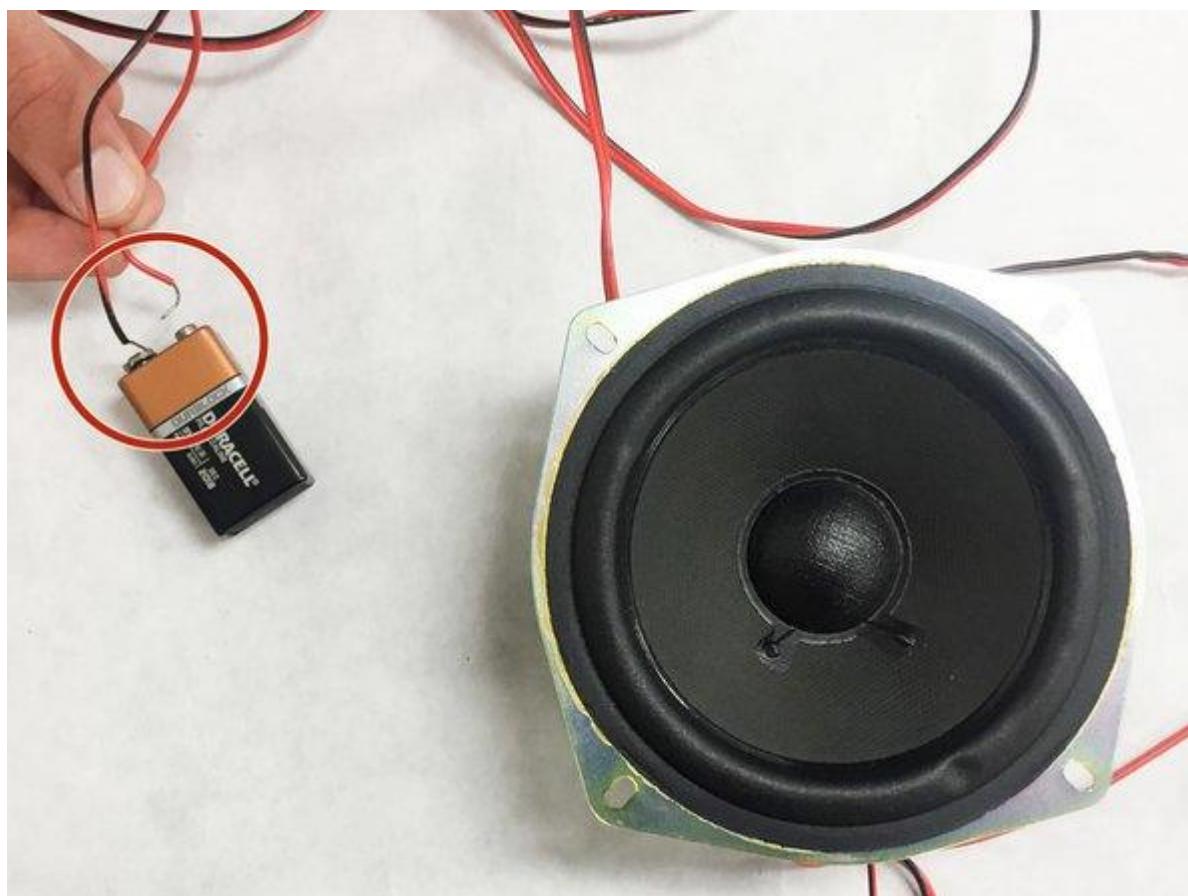


Figure.9 : TESTE AVEC UNE PILE

Si le haut-parleur vibre, alors il fonctionne correctement.

Option 3 : utiliser un multimètre numérique, mettre le paramètre sur résistance Connecter les deux extrémités de l'enceinte aux fils positif et négatif du multimètre comme le montre la figure 10

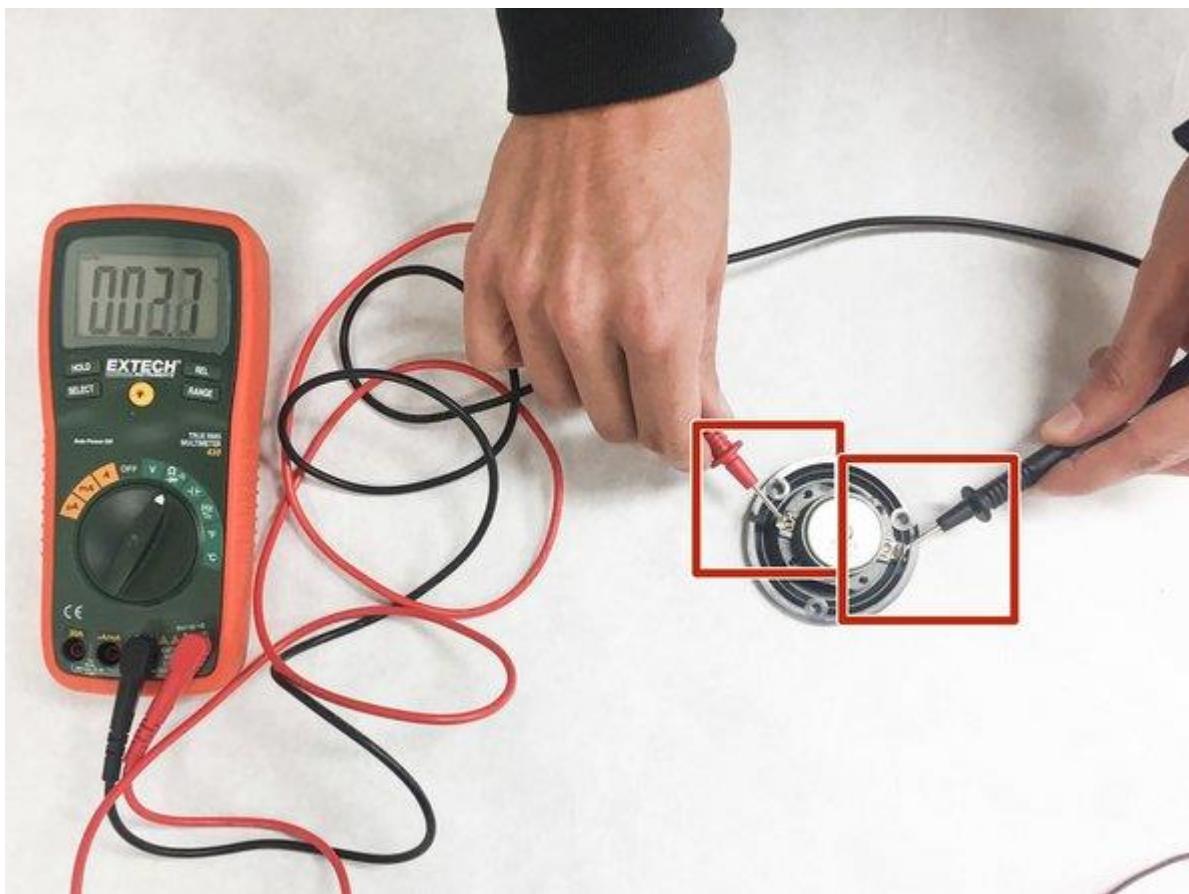


Figure.10 : TESTE AVEC MULTIMETRE

Si le multimètre sonne d'une tonalité cela indique que notre circuit est continu à travers le haut-parleur, donc il est fonctionnel.

Option 4 : brancher le haut-parleur avec une source audio.

2. Teste de qualité de sons

Pour tester la qualité de sons il nous nésite généralement de 2 composants-Un capteur pour capte les ondes sonores produites par un haut-parleur comme le montre la figure 11

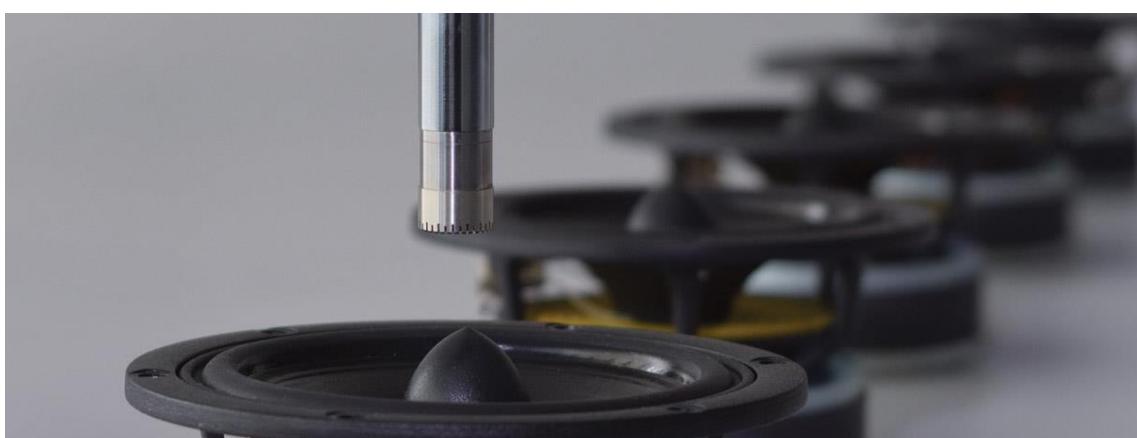


Figure.11 : CAPTEUR DE SONS

Et pour traiter les données récupérées par le capteur il nous faut un logiciel de traitement sonore qui peut nous fournir des informations comme tonalité, qualité, sonie acuité, ...

Exemple de logiciel de traitement sonore RIGHTMARK AUDIO ANALYZER c'est un logiciel d'analyse de la qualité des signaux audiois envoyés par toutes les sources analogiques et numériques.

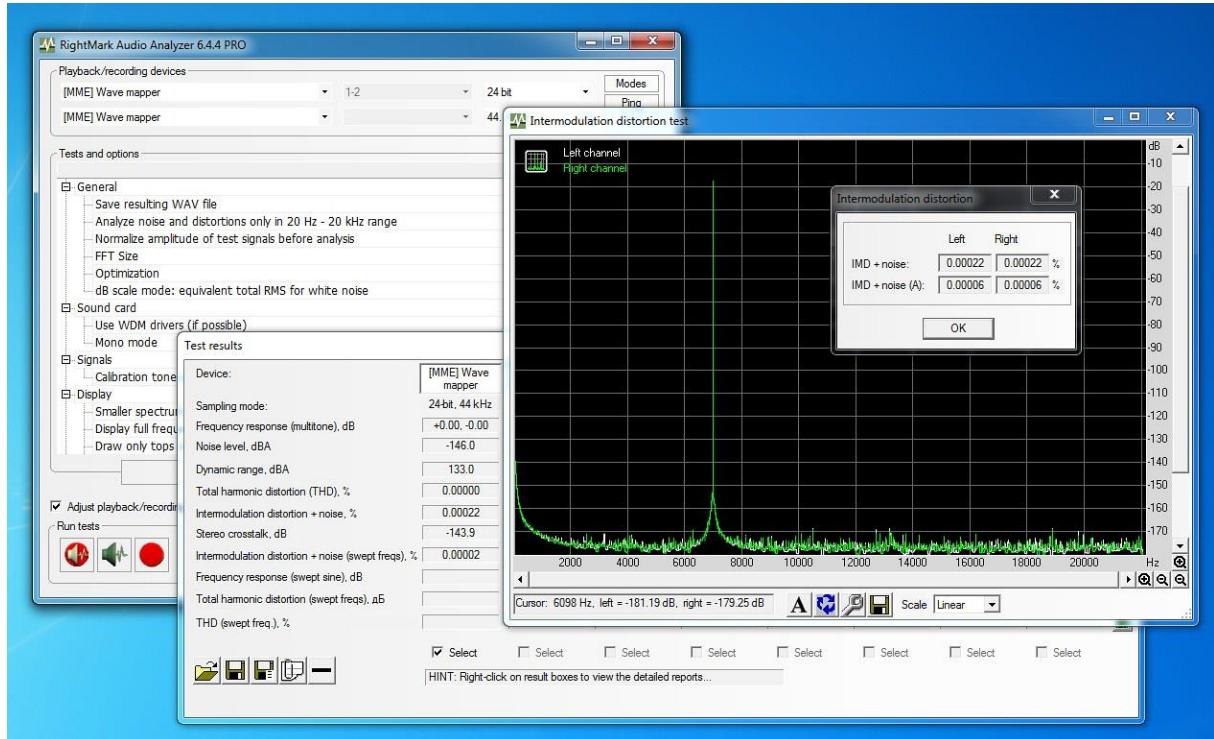


Figure.12 : RIGHTMARK AUDIO ANALYZER

VII. Conclusion

Tout au long de ce chapitre, nous avons évoqué le cadre général du projet. Nous avons commencé tout d'abord par une présentation de l'organisme d'accueil, l'idée maitresse, la problématique et les différent types de haut-parleur ainsi que leur composant et les plusieurs manieur de les teste. Ceci nous a permis de comprendre les besoins et d'envisager la solution la plus adéquate pour la réalisation de notre projet.

Chapitre II : Conception et simulation

I. Introduction

Cette partie a pour but de présenter les différentes solutions technologiques avec notre conception du système afin d'appliquer des choix raisonnables et technique réalisables.

II. Composition et choix des matériels du testeur de haut-parleur

Nous avons organisé notre projet tout en adoptant 2 parties, la première est celle du traitement de données et de commande et la seconde est celle de la propagation de commande comme le montre la figure 13.

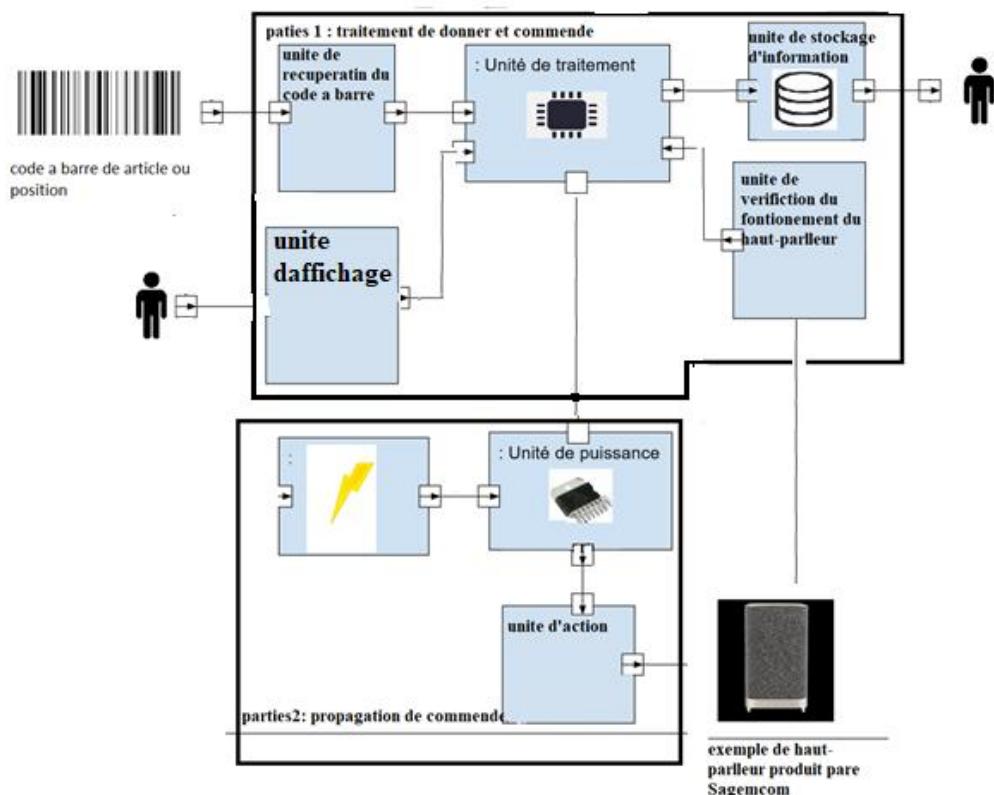


Figure.13 : CHEMIN D'UNITE

1. Partie 1

Dans cette partie nous nous intéressons à la récupération des données et des commandes.

Unité de récupération de code à barre :

Nous avons choisi d'utiliser une douchelette usb pour récupérer le code à barre de l'article ou du box de teste comme le montre la figure 14.



Figure.14 : DOUCHETTE USB

Nous avons choisi ce modèle de lecteur de code à barre parce qu'il est le moins cher sur le marché.

Unité de traitement :

Nous avons le choix d'utiliser dans cette partie un microcontrôleur (arduino, esp32, stm32...) ou un microprocesseur (raspberry pi, beaglebone).

Dans notre cas nous avons favorisé l'utilisation d'un microcontrôleur pour les raisons suivants :

- Faible temps requis pour effectuer l'opération.
- Le microcontrôleur est simple à interfaçer avec leur **RAM**, **ROM** et leur port **d'E/S** supplémentaires.
- Le coût et la taille du système sont inférieurs par rapport au microprocesseur.

Entre les différents types de microcontrôleurs nous avons choisi d'utiliser la carte stm32f401

Comme le montre la figure 15.

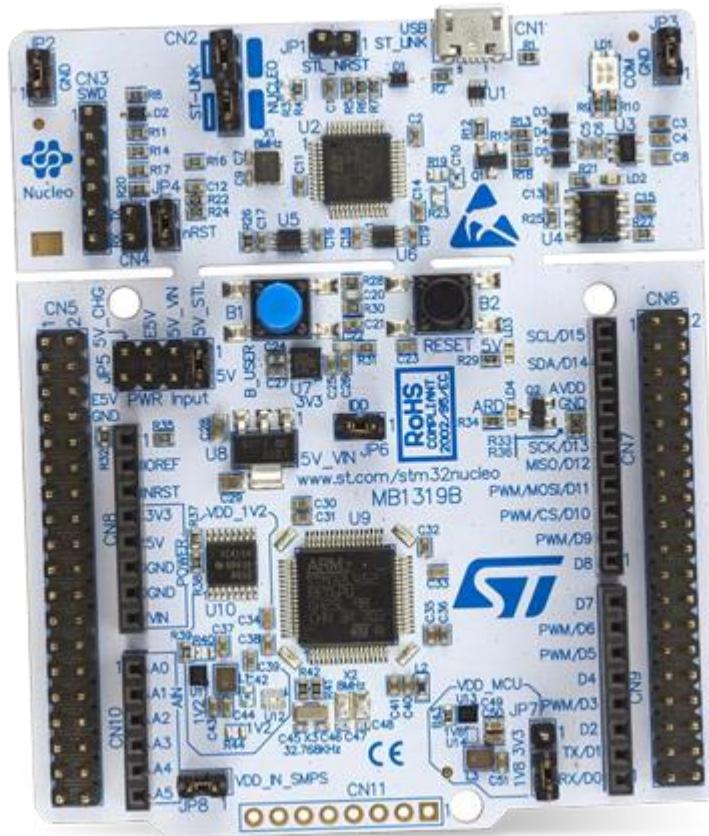


Figure.15 : CARTES STM32401

Le choix de cette carte a été fait pour les raisons suivant :

- Les microcontrôleurs STM32 offrent un grand nombre de périphériques de communication série et parallèle que les cartes arduino ou esp32.
- STM32 a plus de mémoire flash qui est le double par rapport à l'Arduino, alors que la **RAM** est 10 fois plus grande que celle d'Arduino.
- Les microcontrôleurs STM32 sont plus orientés pour l'industrie que les cartes l'Arduino ou esp32.

Pour développer la carte stm32 nous avons choisi d'utiliser le logiciel stm32cubeide comme le montre la figure 16.

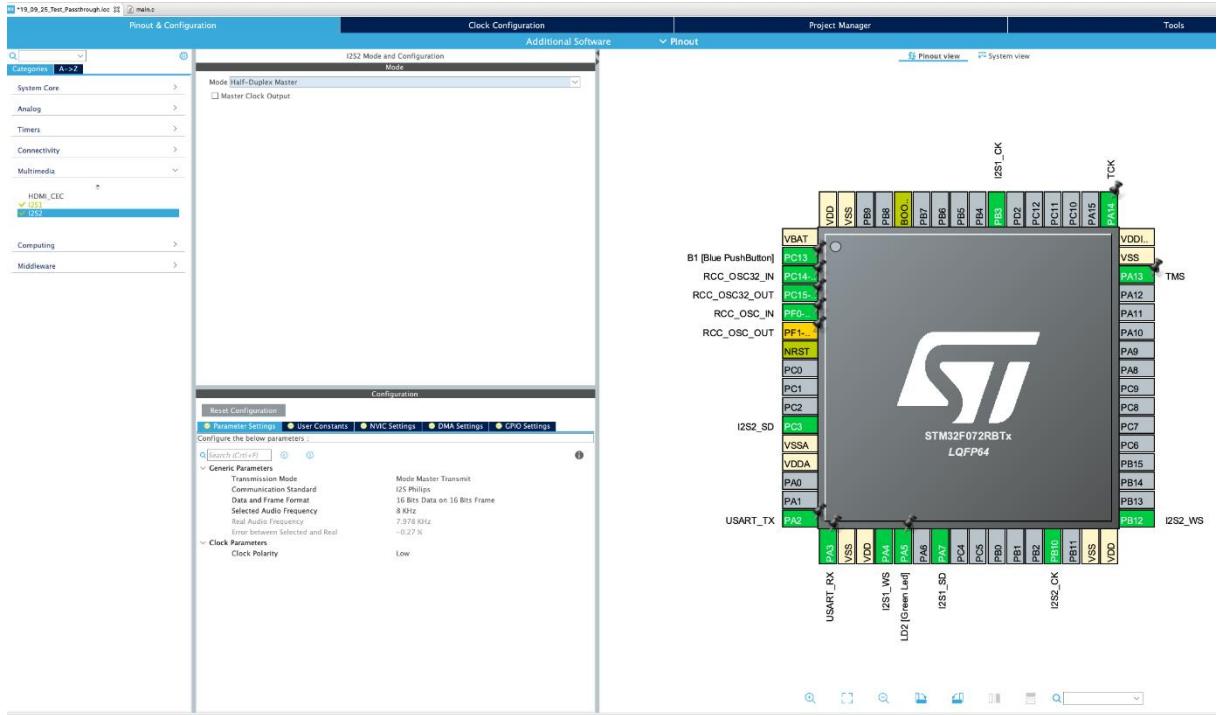


Figure.16 : STM32 CUBE IDE

Ce logiciel nous permet de configurer graphiquement et d'une manière plus facile que d'autres logiciels de développement.

Unité de stockage d'information :

Notre projet nécessite d'avoir une base de données pour stocker les informations et les détails de chaque produit, nous nous intéressons qu'aux haut-parleurs pour la sécurité et la confidentialité de la base de données de Sagemcom ils ne nous ont pas donné l'accès à leur base de données, alors pour le prototype du projet testeur nous allons utiliser une base données de notre choix.

Nous avons choisi d'utiliser Firebase qui est développé par google qui est une base de données **NoSQL** non relationnelle cela nous permet d'accéder et modifier nos données dans un temps réel est d'une manière plus facile qu'un tableau **SQL** classique.

La figure 17 montre la base de données Firebase.

The screenshot shows the Firebase Realtime Database interface. On the left, there's a sidebar with navigation links like 'Créer', 'Authentification', 'Realtime Database', 'Storage', 'Hosting', 'Functions', 'Machine Learning', 'Publier et surveiller', 'Analytics', and 'Engager'. The main area is titled 'Realtime Database' and has tabs for 'Données', 'Règles', 'Sauvegardes', and 'Utilisation'. A message at the top says 'Protégez vos ressources Realtime Database des utilisations abusives telles que la fraude à la facturation et le hameçonnage' with a link to 'Configurer App Check'. Below this is a code editor-like interface showing a JSON structure. The URL in the browser bar is 'https://esp01-8afe9-default-rtdb.firebaseio.com/'. The JSON structure under 'a' contains several nodes with random IDs: 4260093465551XY, 4260093465629XY, 61924300003929XY, 61948024XY, 6194819607382XY, and 61942530XY. At the bottom, it says 'Emplacement de la base de données : Belgique (europe-west1)'.

Figure.17 : FIREBASE

Pour assurer la connexion entre le microcontrôleur est la base de données il nous faut un modèle de connexion à internet pour accéder à Firebase.

Nous avons choisi d'utiliser un modèle esp01 comme le montre la figure 18.

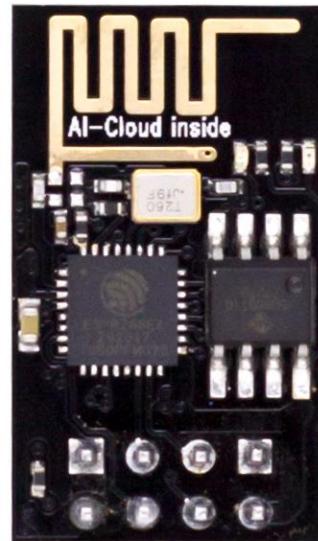


Figure.18 : ESP01

Ce module est un émetteur-récepteur sans fil **WIFI** série ESP 01 ESP8266 est un **SOC** autonome avec une pile de protocoles **TCP/IP** intégrée qui peut donner à n'importe quel microcontrôleur un accès à un réseau **Wi-Fi** , est capable d'héberger une application ou de décharger toutes les fonctions de réseau **Wi-Fi** d'un autre processeur d'application.

Donc pour relier la carte stm32 à la base données nous avons utilisé un intermédiaire dans notre cas c'est le module esp8266 comme le montre figure 19.

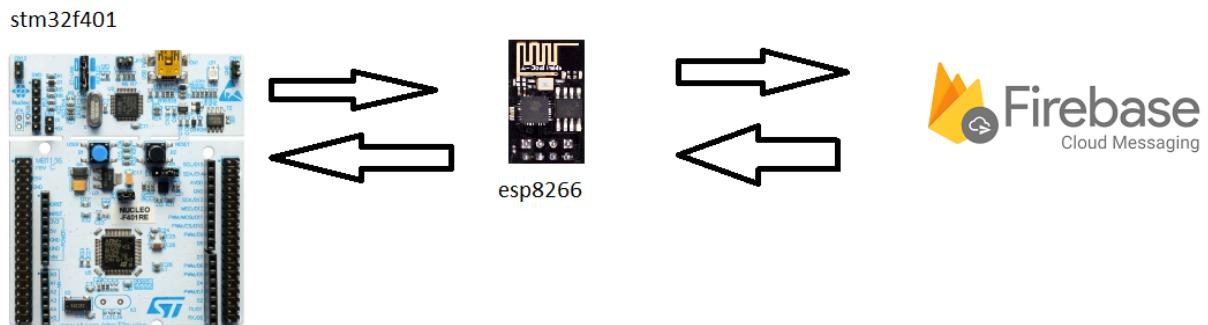


Figure.19 : CHEMIN STM32+ESP8266+BASE DE DONNE

Unité de vérification de fonctionnement de teste :

Pour assurer l'autonomie du système nous avons utilisé un capteur de sons.

Nous avons choisi d'utiliser le modèle ky-038 comme la figure 20 le montre.



Figure.20 : CAPTEURS DE SONS

Ce capteur émet un signal si le microphone du capteur reçoit un son de Fréquence bien définie sella indique que notre haut-parleur est bien fonctionnel si non le haut-parleur a des problèmes.

Unité d'affichage :

Afin d'afficher à l'opérateur les différents opérations ou les traitements du microcontrôleur nous avons choisi d'installer un afficher LCD (16*2) comme le montre la figure 21.

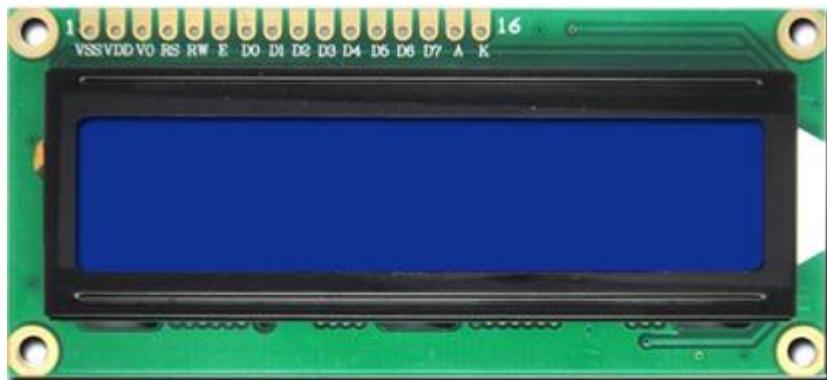


Figure.21 : AFFICHEUR LCD 16*2

Nous avons choisi ce modèle car il est le moins cher dans le marché.

2. Partie 2

Dans cette partie on s'intéresse à la propagation de commande.

Unité d'action sur le haut-parleur :

Pour automatiser le teste, on doit brancher les différents câbles (alimentation, jack ...) d'une manière autonome donc, Sagemcom a exigé d'utiliser des vérins électriques comme le montre la figure 22.

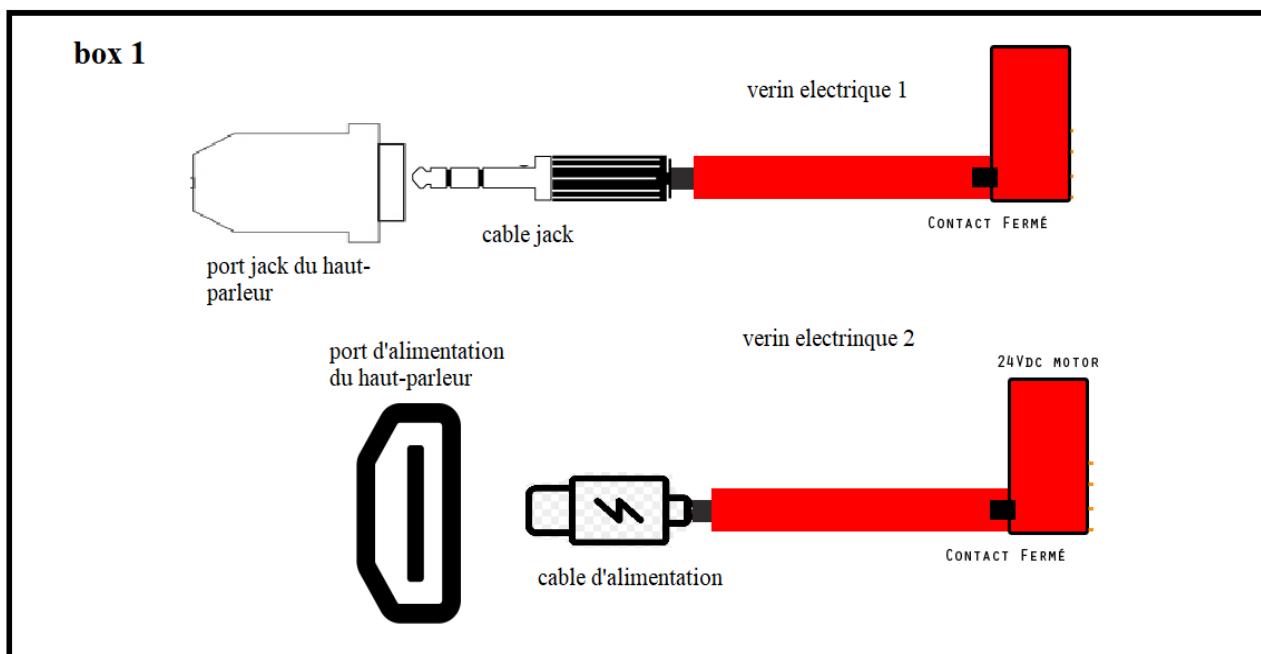


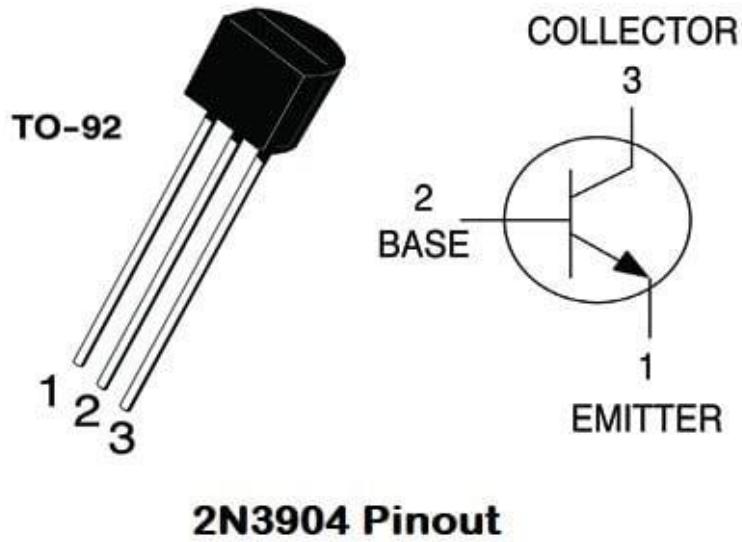
Figure.22 : CHEMIN EXPLICATIVE DU BOX

Ce type de vérin fonctionne avec une tension de 12v.

Unité de puissance :

Pour contrôler les vérins électriques nous avons choisi d'utiliser un transistor avec une diode à jonction pour cela nous avons utilisé le transitoire NPN 2n3904 comme le montre la figure 23.

Ce type de transistor nous permet de commander une entrée de tension jusqu'à 12v avec 5v



2N3904 Pinout

Figure.23 : TRANSISTORS 2N3904

l'entrée.

Nous avons utilisé les diodes à jonction BZX85C12-TAP cela nous permet d'assurer le passage du courant dans un seul sens pour éviter un retour de courant.



Figure.24 : DIODE A JONCTION

Nous avons choisi cette méthode plutôt que les relais pour les raisons suivantes :

- les transistors sont plus pétries et moins cher que les relais.
- les relais consomme plus d'énergies pour leur fermeture de courant.
- le courant consommé par les vérins n'est pas très élevé pour utiliser un relais.

La figure suivante nous montre tous les composants physiques et virtuels du projet testeur.

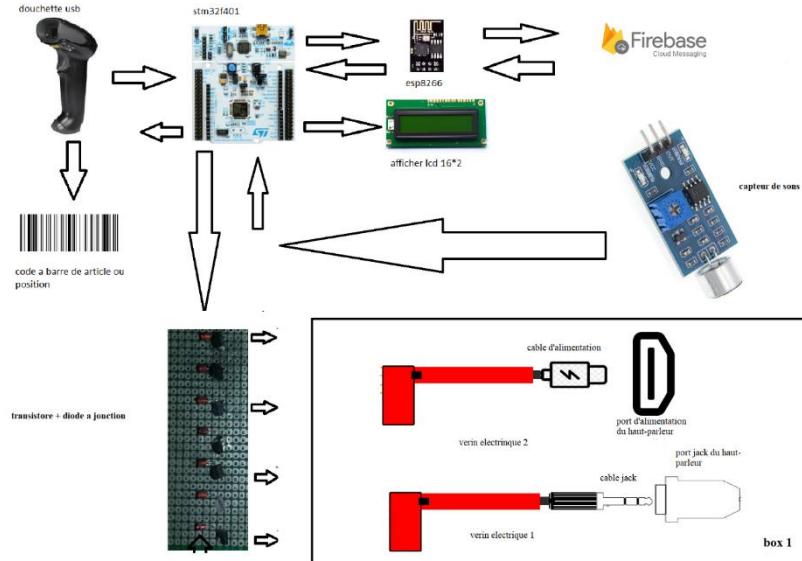


Figure.25 : COMPOSITIONS GENERALES

III. Conception du testeur de haut-parleur

1. Choix de logiciel

Dans la partie de conception on va utiliser 2 logiciels de conception.

-Proteus 8 Professional est un logiciel qui peut être utilisé pour dessiner des schémas, la disposition des circuits imprimés, coder et même simuler le schéma nous allons l'utiliser pour la simulation comme le montre la figure 26.

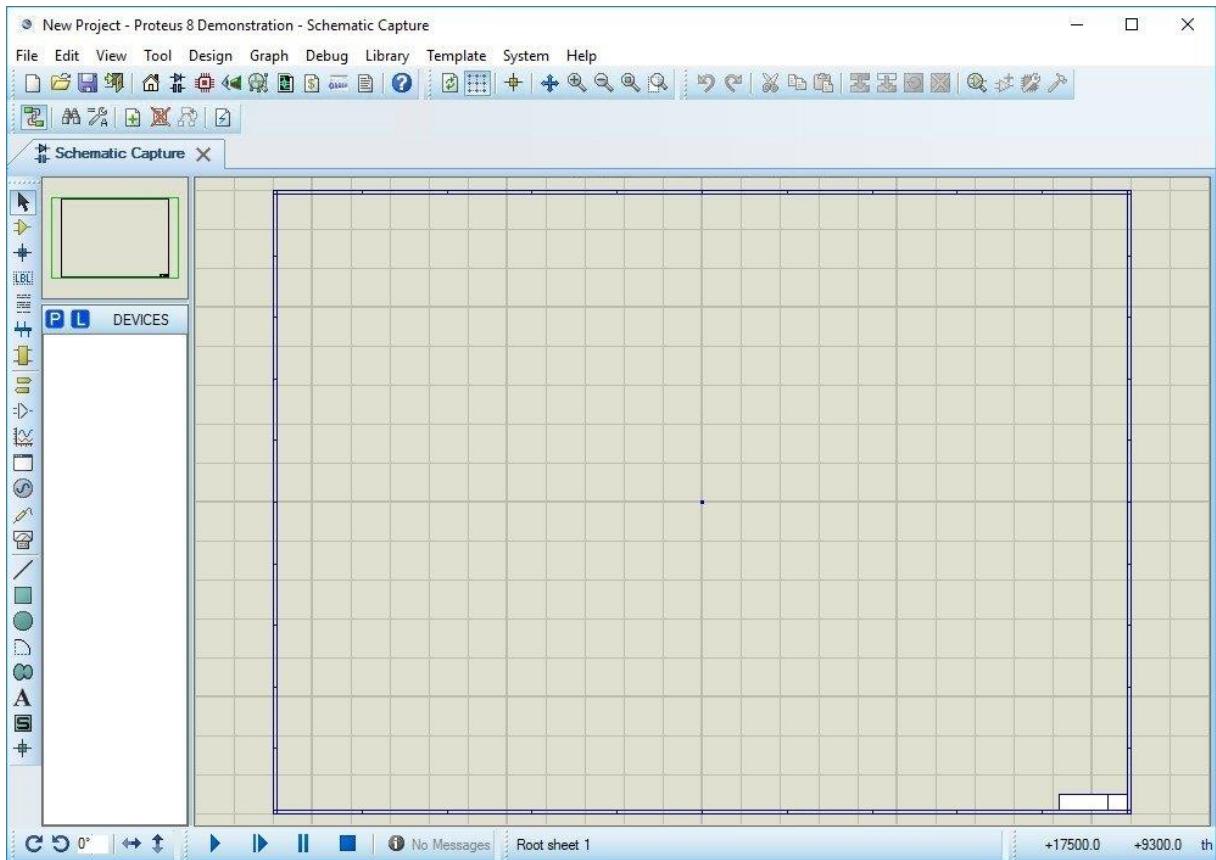


Figure.26 : PROTEUS 8

- Altium Designer System Engineering (SE) est un éditeur de schémas complet qui comprend de puissantes capacités de collaboration et un ensemble complet d'outils de capture de schémas pour créer, modifier, simuler et documenter rapidement des schémas nous l'avons utilisé pour la conception des carte PCB comme le montre la figure 27

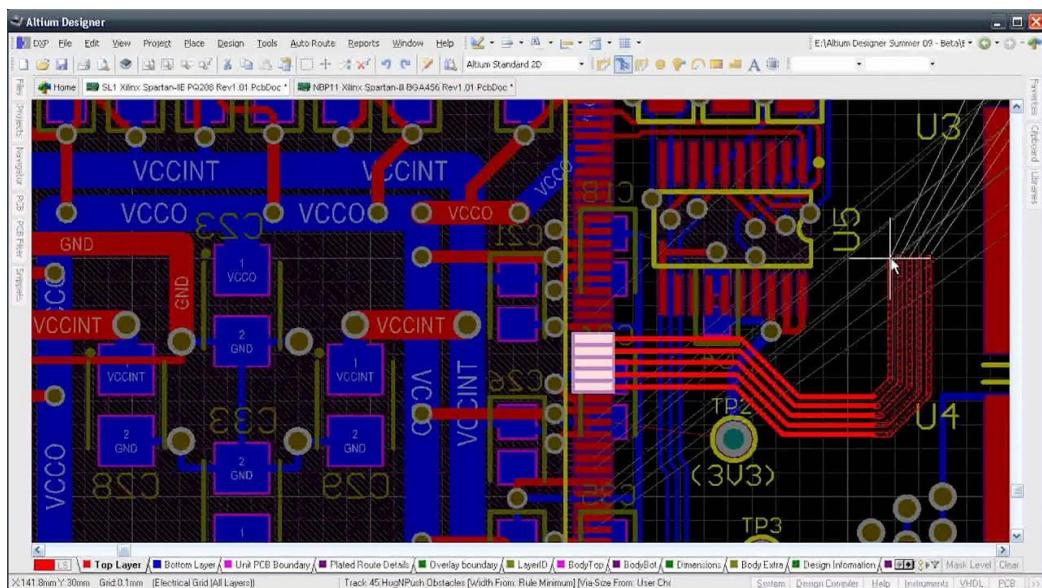


Figure 27 : ALTIUM

2. simulation de la partie transitoire et diode à jonction

Dans cette partie nous avons utilisé le logiciel Proteus pour la simulation et pour les vérins électriques nous avons utilisé des lampes de 12 v.

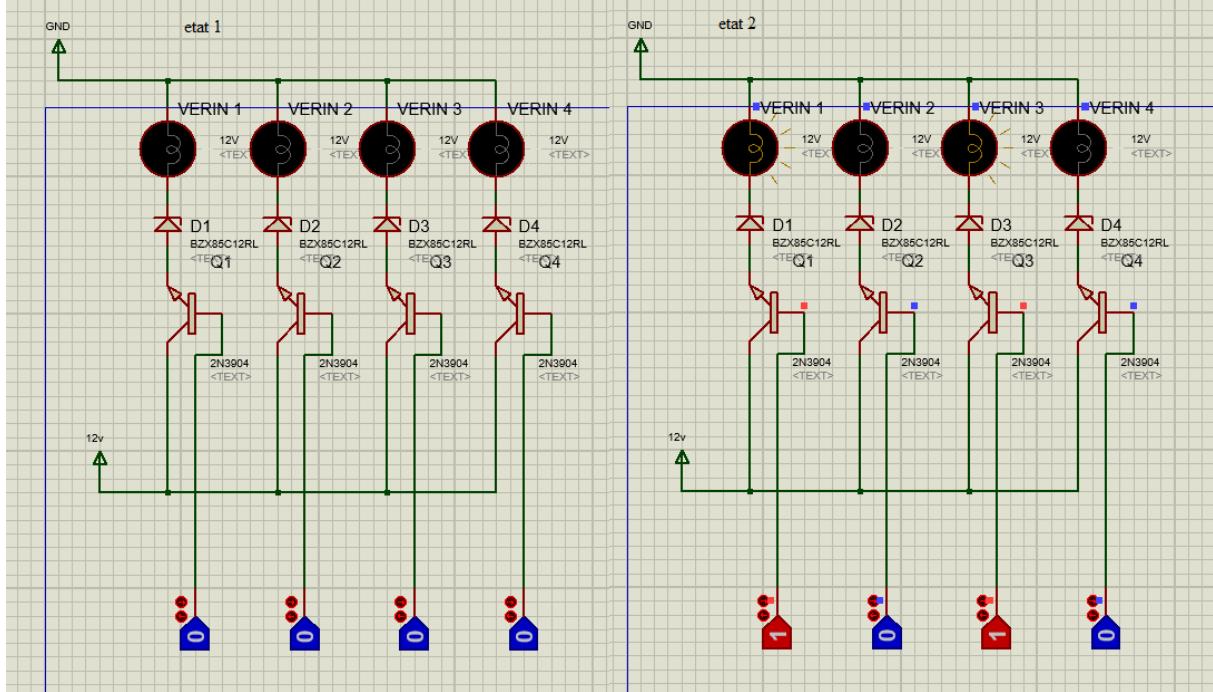


Figure.28 : SIMULATION DE TRANSISTORS ET DIODE A JONCTION

Comme le montre la figure 28 nous avons 2 états :

- état 1 les 4 transistors ne sont pas existés donc, les vérins électriques ne fonctionnent pas.
- état 2 le transistor Q1 et Q3 sont existés donc, le vérin 1 et vérin 3 sont en marche.

Dans cette partie nous avons utilisé le logiciel altium pour la conception d'une carte PCB

3. Conception de la partie transitoire et diode a jonction

La figure 29 représente la conception schématique de la carte.

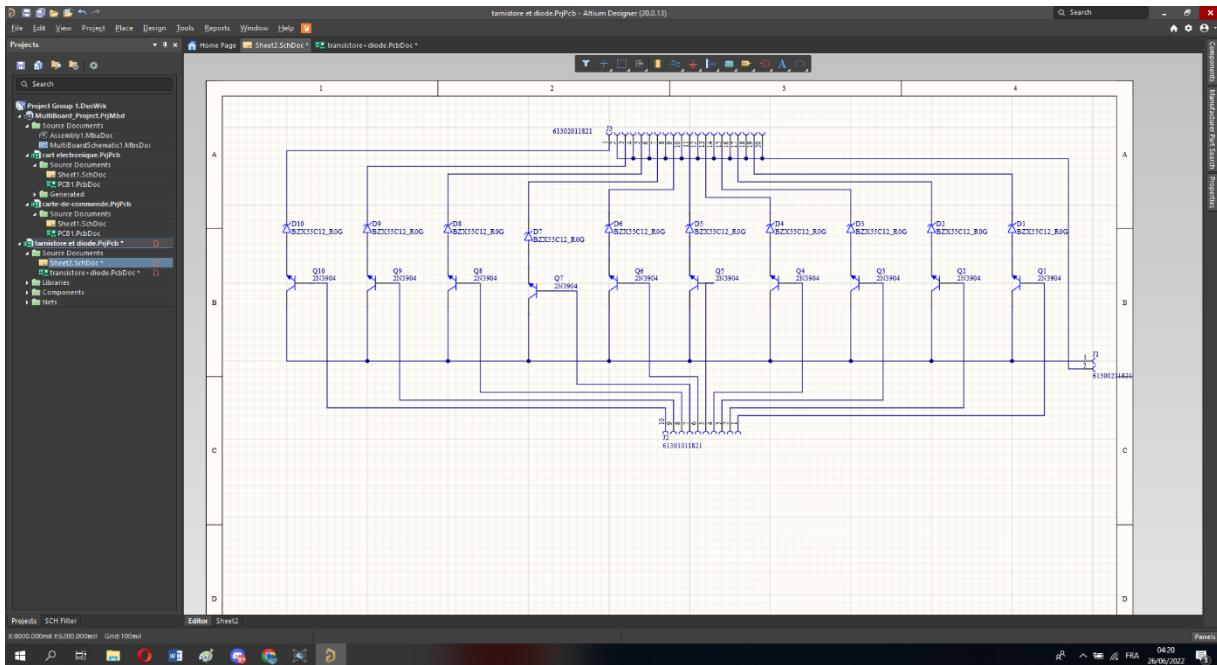


Figure.29 : SCHEMA PCB

La figure 30 représente la conception PCB en 2d de la carte

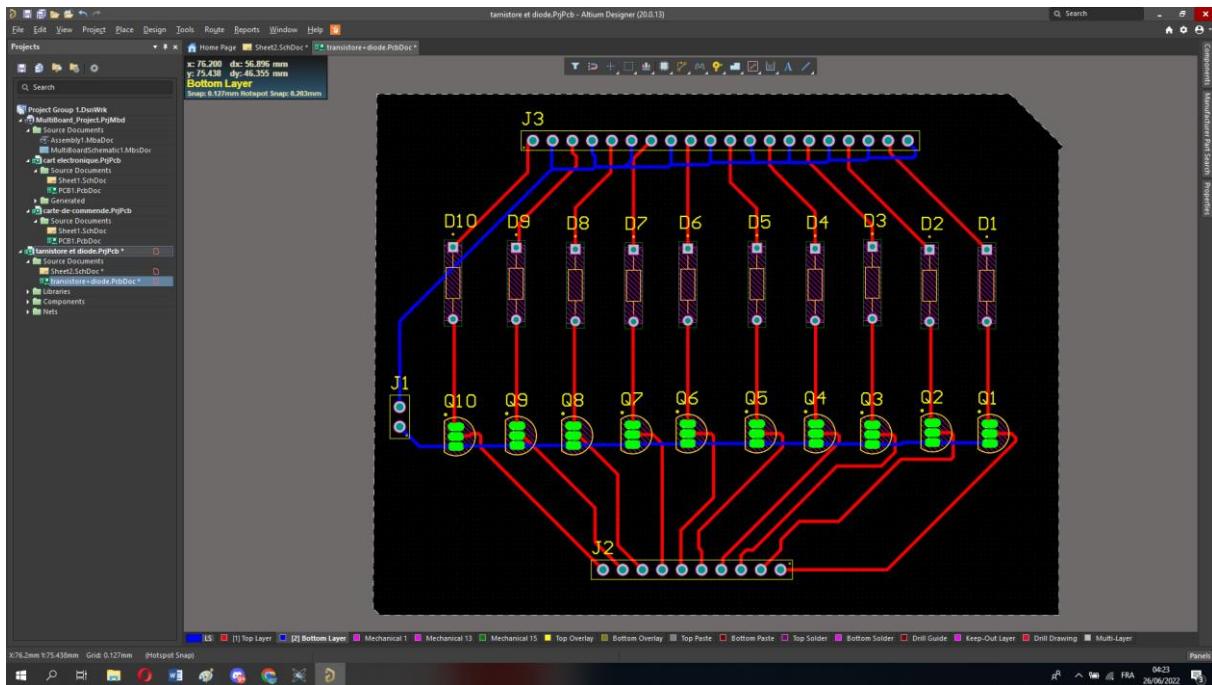


Figure.30 : CONCEPTIONS 2D

La figure 31 représente la conception PCB en 3d de la carte

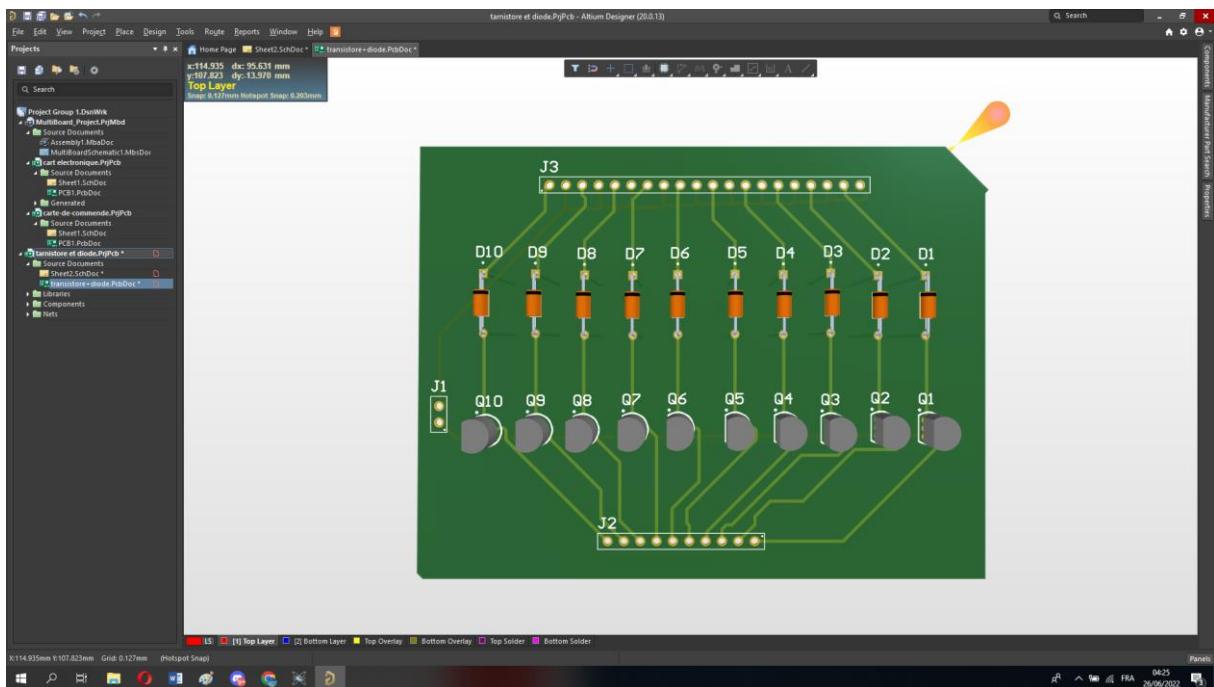


Figure 31 : CONCEPTION 3D

4. Conception de la partie de traitement et de commande

Dans cette partie on s'intéresse à la conception et à la programmation des différentes cartes électroniques et les protocoles de communication entre les différents composants.

Première étape consiste à lire le code à barre introduit par la douchette usb à la carte stm32 pour le traitement et l'identification.

La douchette usb comme plusieurs périphérique usb (claviers, souris ...) communique avec le protocole USB HID (Human Interface Device protocol) celui-ci se compose d'un "host" et d'un "devis" dans notre cas le "host" est la carte stm32 et le "devis" est la douchette usb.

Tout produit utilise le Protocol USB HID a les mêmes sorties de data comme le montre la figure 32.



Figure.32 : LES BROCHES D'UN CABLE USB

Les différentes broches :

- Les broches 1 et 4 sont pour l'alimentation du périphérique.
- La broche 2 est la borne de données négative notée D- (DM), tandis que la broche 3 est la borne de données positive notée D+ (DP).

Le branchement de la carte stm32 avec la douchede usb est schématisé dans la figure 33.

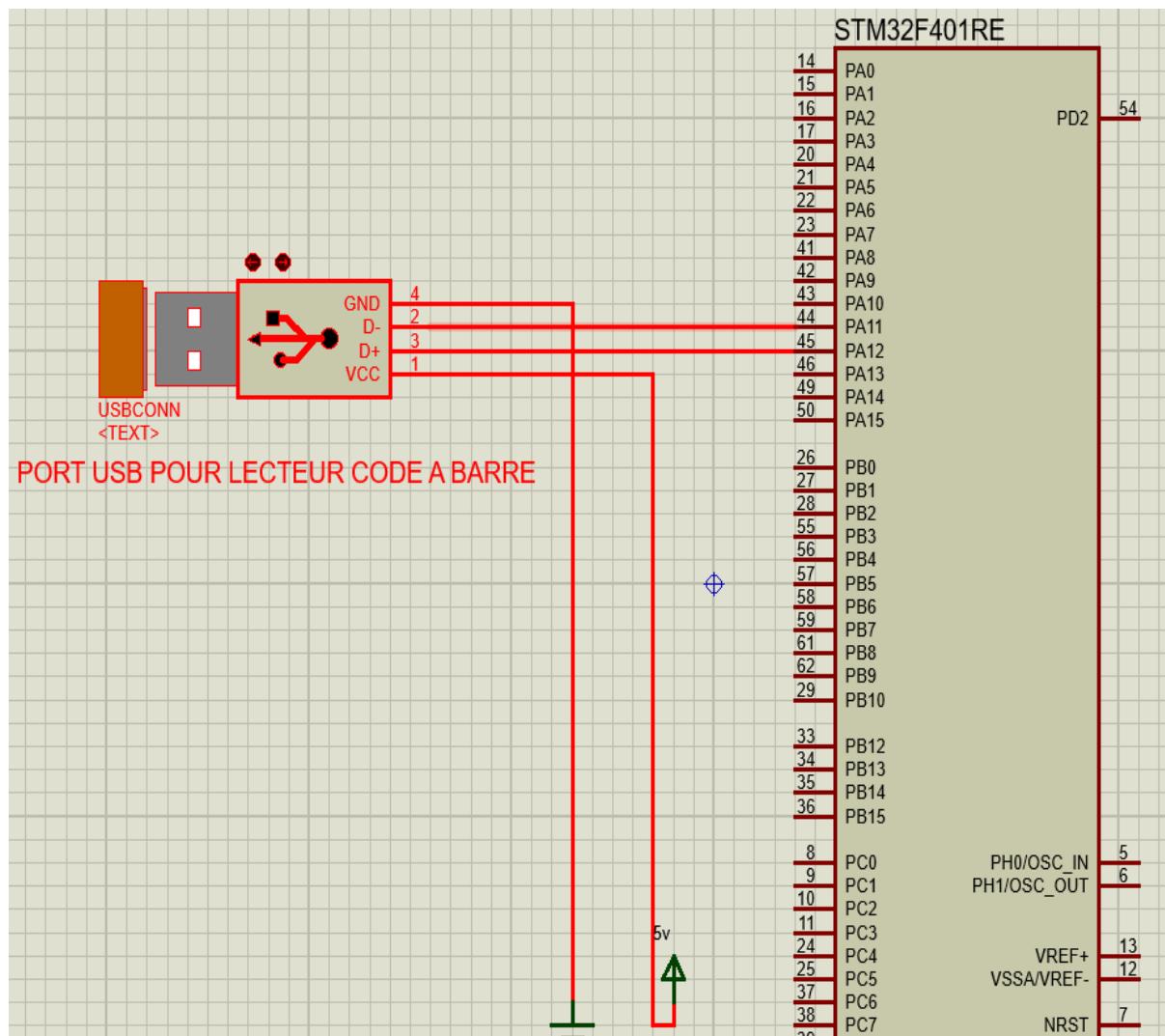


Figure.33 : BRANCHEMENTS STM32 AVEC PORT USB

La configuration et la programmation de la carte stm32 avec le Protocol "USB_OTG_FS» pour lire les données de la douchette **USB** comme le montre la figure 34 et 35.

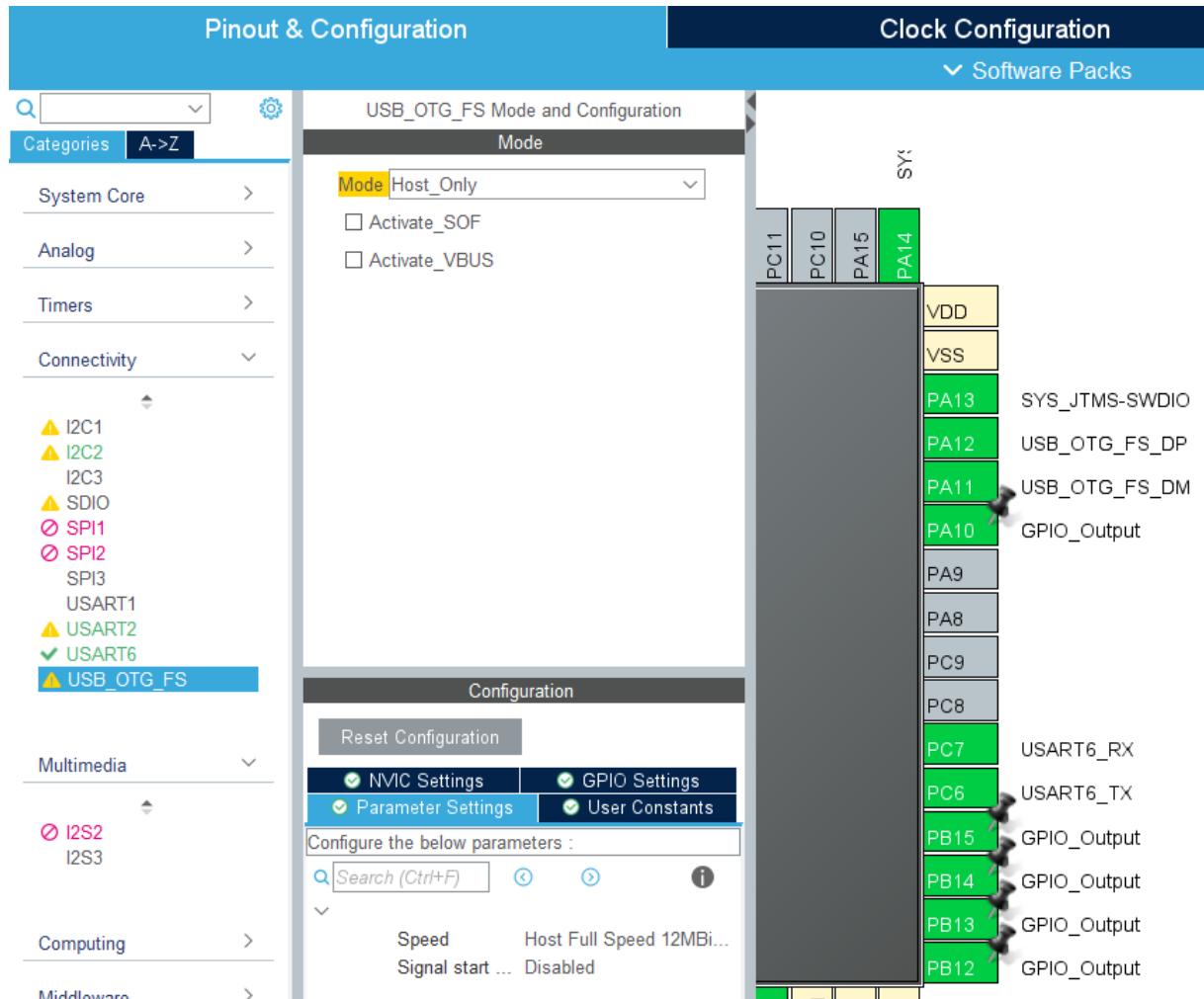


Figure.35 : CONFIGURATIONS STM32

```

149
150
151 void USBH_HID_EventCallback(USBH_HandleTypeDef *phost)
152 {
153     HID_KEYBD_Info_TypeDef *keybd_info;
154
155     HID_HandleTypeDef *HID_Handle = (HID_HandleTypeDef *) phost->pActiveClass->pData;
156     if(HID_Handle->Init == USBH_HID_KeyboardInit){
157         keybd_info = USBH_HID_GetKeybdInfo(phost);
158         keycode = USBH_HID_GetASCIICode(keybd_info);
159         uart_length=sprintf(uart_tx_buffer, "%c",keycode);
160         strcat(buffer1,uart_tx_buffer);
161     }
162 }
163

```

Figure.34 : PROGRAMMATION DE LA CARTE STM32

Nous avons développé une procédure pour récupérer les données d'entrées par les branches P11 P12 et les Transférer en un String pour faciliter la manipulation et l'identification du code à barre entrée s'il est un code à barre d'un haut-parleur ou d'un box.

Pour vérifier que le teste précédent du haut-parleur est validé il nous faut un accès à notre base de données.

Pour cela nous avons branché une carte esp8266 avec stm32 pour l'accès à la base de données comme le montre la figure 36

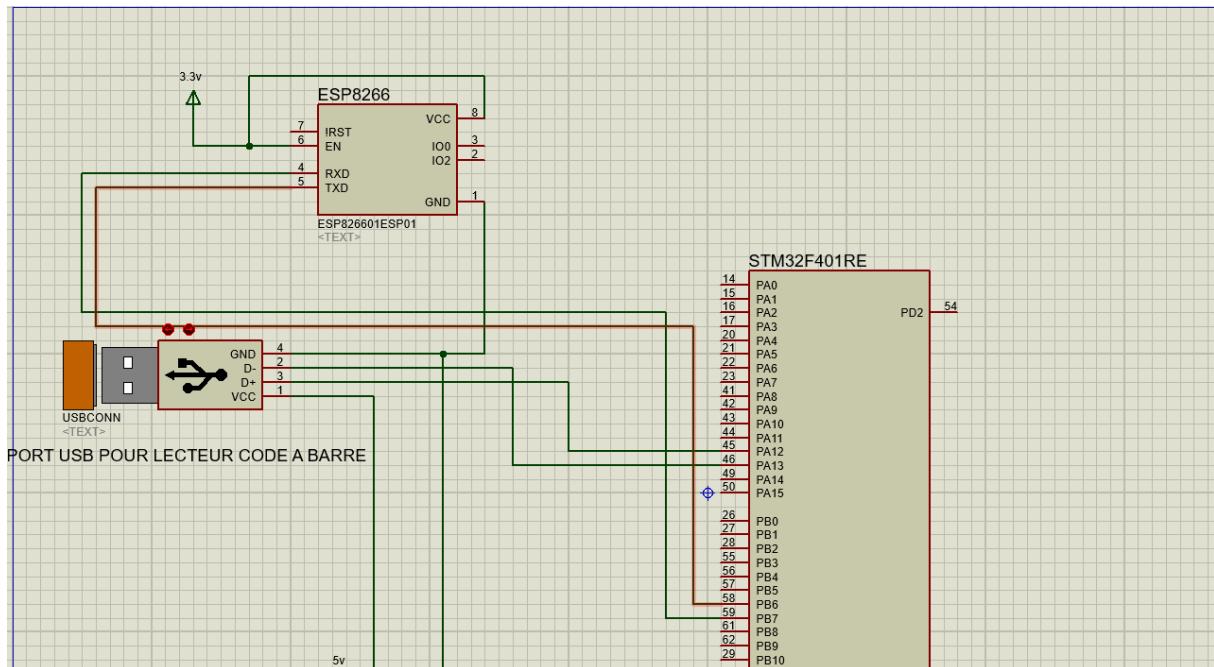


Figure.36 : BRANCHEMENT DE STM32 AVEC ESP8266

Nous avons branché le tx (Transiter) de la carte esp8266 à RX (receveur) de la carte stm32 et RX de la carte esp8266 à TX de carte stm32 comme le montre la figure 37.

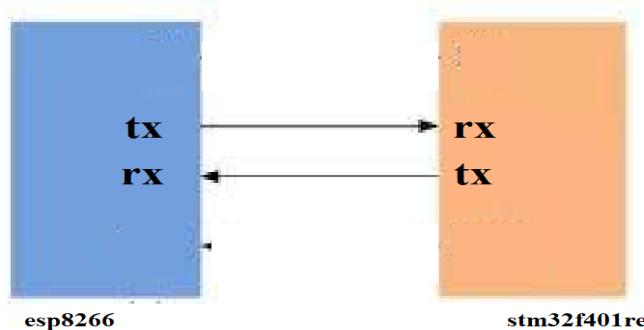


Figure.37 : BRANCHEMENTS DE TX RX

La configuration de la broche PB7 pour être RX (receveur) et PB6 pour être TX comme le montre la figure 38

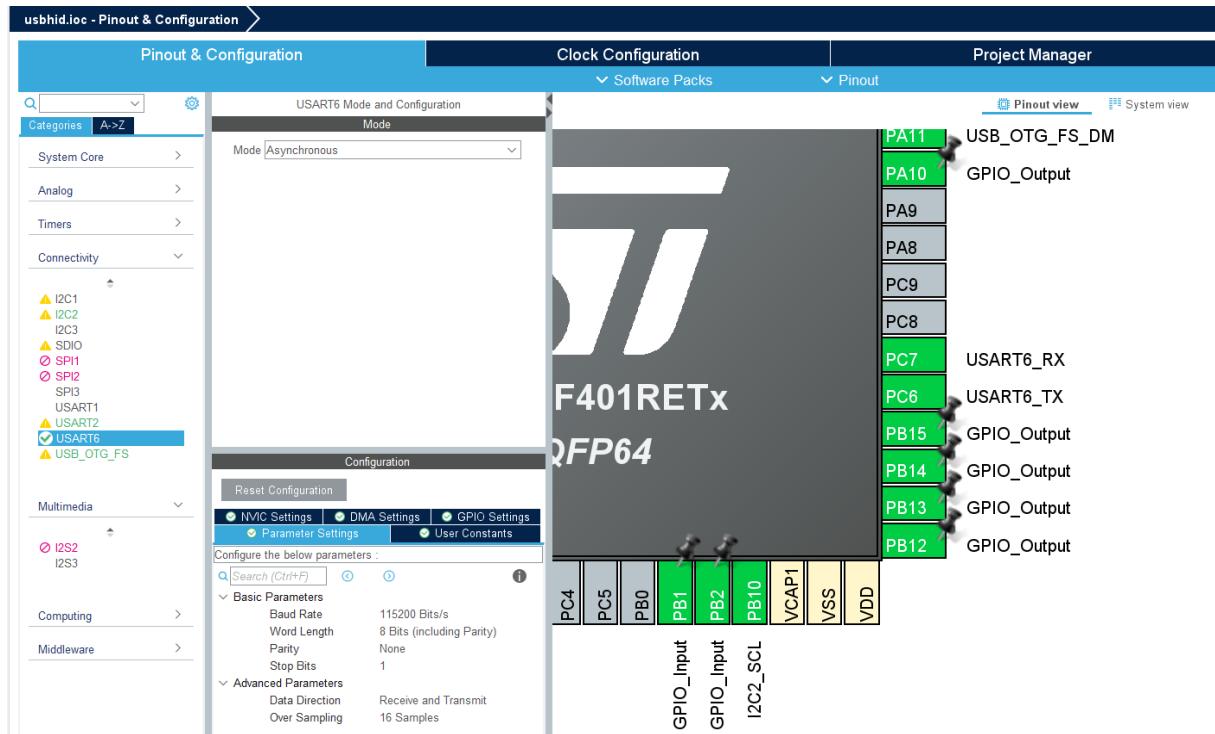


Figure.38 : CONFIGURATION STM32 AVEC ESP8266

Nous avons programmé la carte stm32 pour transférer les requêtes et récupérer les données comme la montre la figure 39.

```

456
457
458     HAL_UART_Transmit(&huart6,(uint8_t*)buffer8,sizeof(buffer8),3000); // transfer les donnees a la carte esp8266
459     HAL_UART_Receive(&huart6,(uint8_t*)rx_buffer4,3,3000); // recuperation des donnees
460
461
462

```

Figure.39 : TRANSFERTS DE DONNEES A ESP8266

Et même chose pour la carte esp8266 nous l'avons programmée pour récupérer les requêtes et les vérifier sur la base de données et les transmettre à la carte stm32 comme le montre la figure 40.

```

if(Serial.available()!=0)
{

sendDataPrevMillis = millis();
String nam = Serial.readString(); // recuperation de donnees de la carte stm32
delay(10);

Serial.printf("%s",n= Firebase.getBool(fbdo,nam) ? fbdo.to<bool>() ? "true" : "false" : "error"); // verifie le code a barre sur la basse de donner
}

```

Figure.40 : LA PROGRAMMATION DE LA CARTE ESP8266

Après que la carte stm32 récupère les données elle les affiche sur l'écran lcd pour cela nous avons utilisé un afficher lcd 16*2 **i2c** comme le montre la figure 41

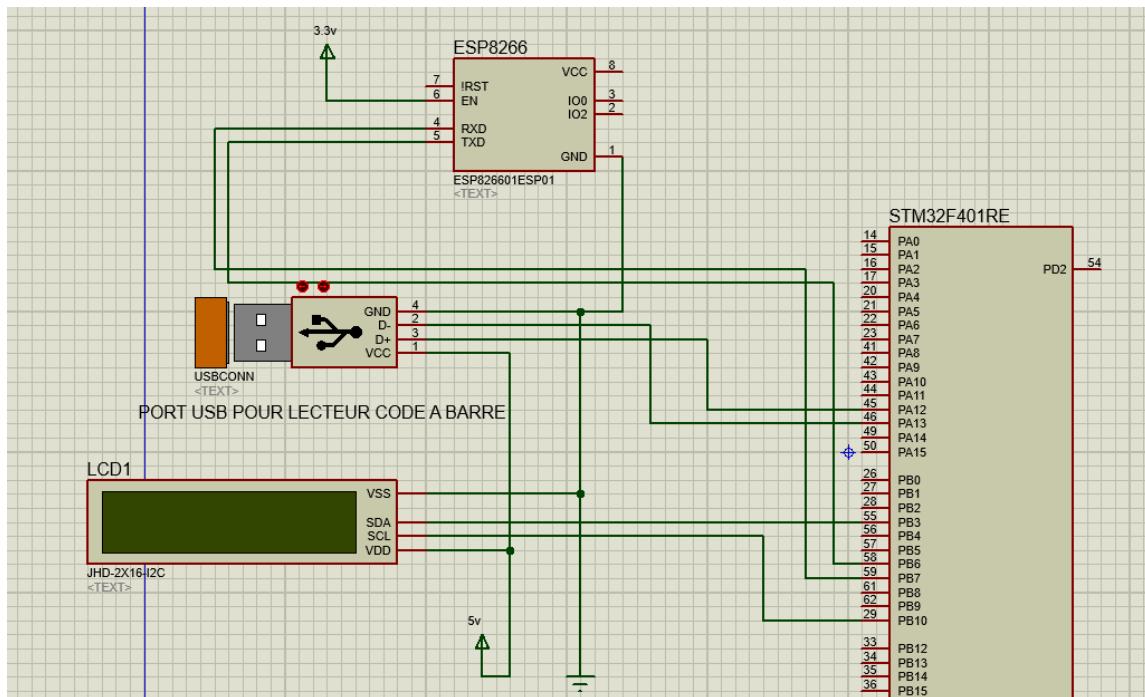


Figure.41 : CHEMIN DU CARTE STM32 AVEC AFFICHEUR 16*2

Nous avons configuré le port PB3 et PB10 pour être des sorties i2c comme le montre la figure 42.

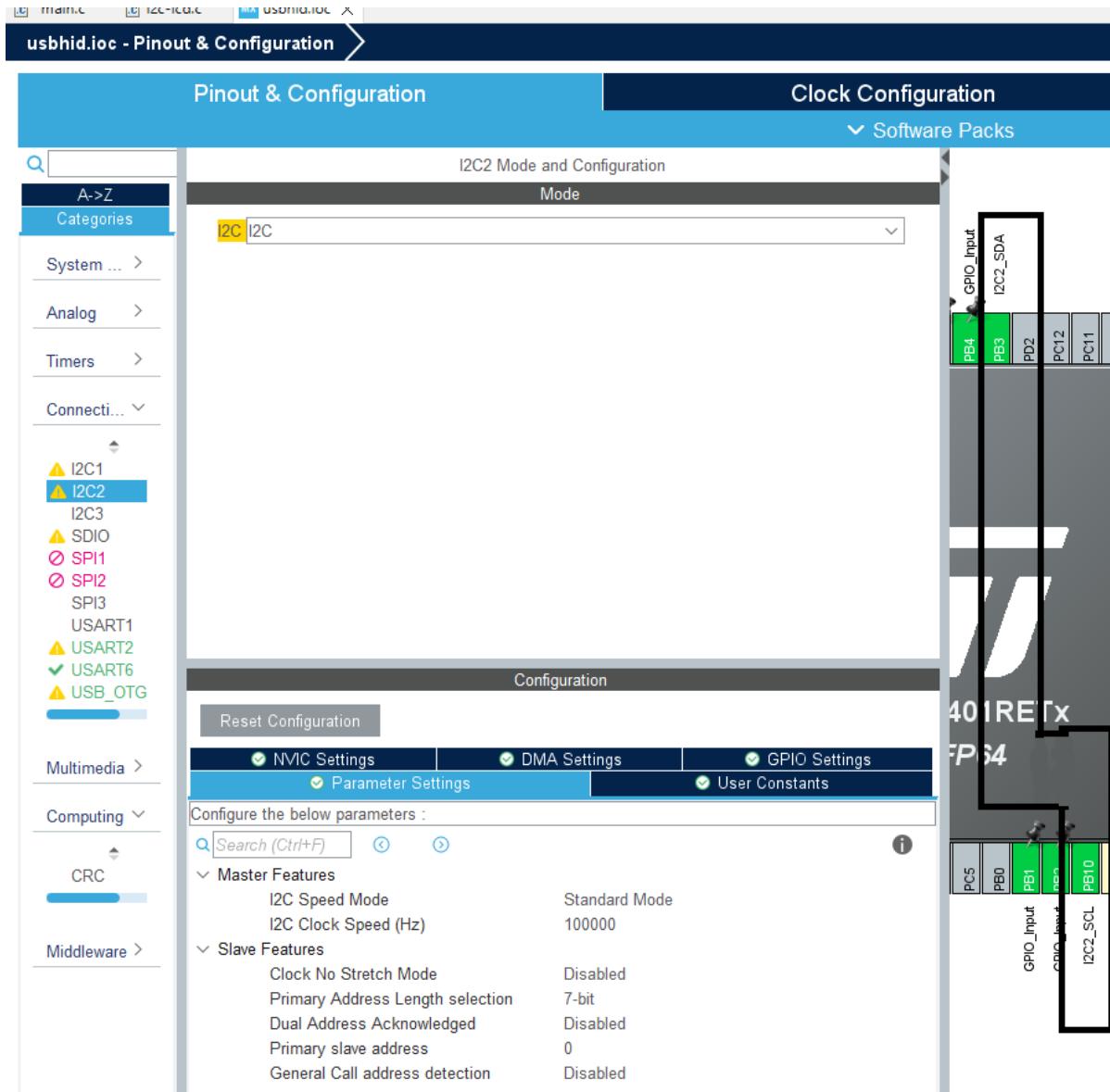


Figure.42 : CONFIGURATION I2C DE LA CARTE STM32

Nous avons programmé la carte stm32 pour afficher l'état du teste précédent comme le montre figure 43.

```

HAL_UART_Transmit(&huart6,(uint8_t*)buffer8,sizeof(buffer8),3000); // transfer les donnees a la carte esp8266
HAL_UART_Receive(&huart6,(uint8_t*)rx_buffer4,3,3000); // recuperation des donnees
lcd_clear (); // supprime le msg precedent
lcd_put_cur(0, 0); // le curseur est a la position 0.0
lcd_send_string(rx_buffer4); // afficher le msg resuver de la carte esp8266
void lcd_send_string (char *str)
{
    while (*str) lcd_send_data (*str++);
}
    ,5,3,3000);

```

Figure.43 : PROGRAMMATIONS DE LA CARTE STM32 POUR AFFICHER UN MSG RECUPERE DE LA CARTE ESP01

Si le teste précédent est validé les vérins électriques et le capteur seront activés comme le montre la figure 44.

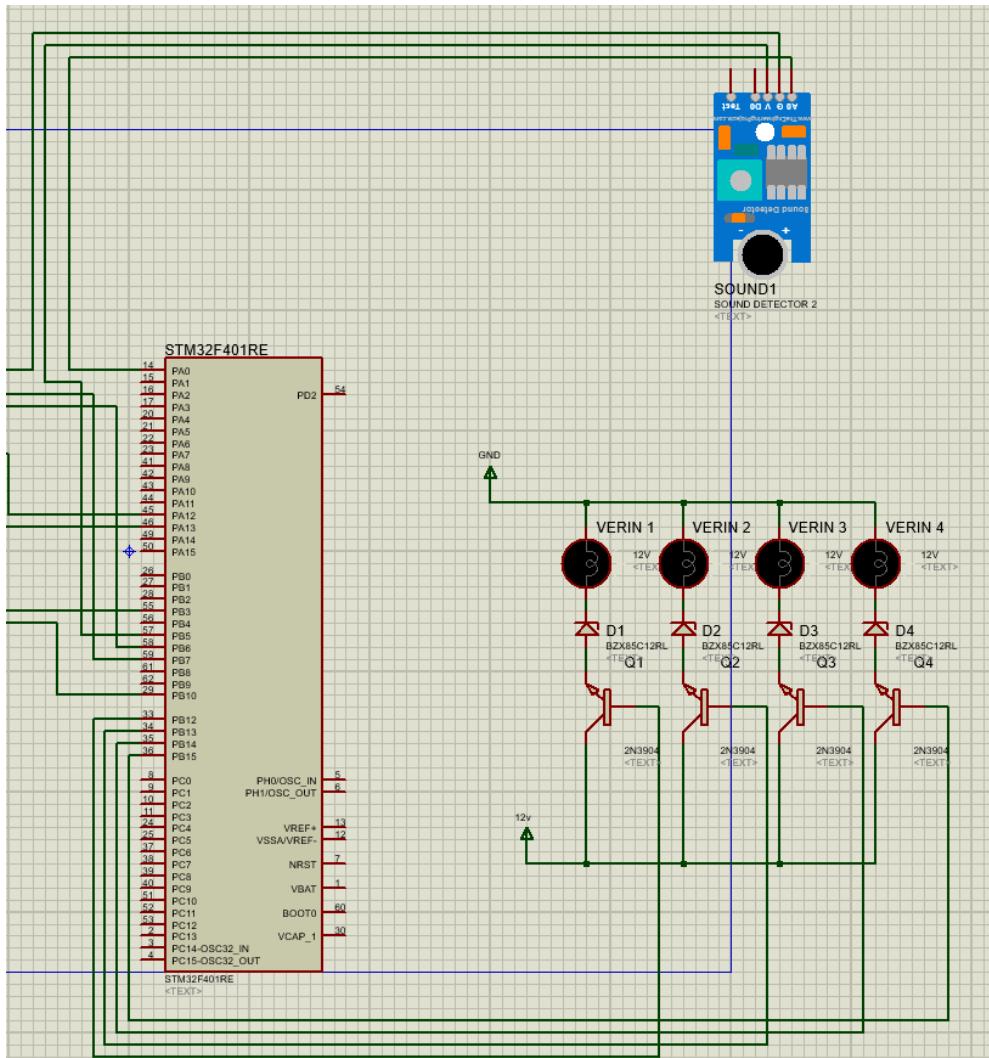


Figure.44 : CHEMIN DE LA CARTE STM32 + KY038

Pour cela nous avons programmé la carte stm32 pour activer les transistors puis le capteur de sons en suite, de lire la valeur analogique sortante du capteur de sons comme le montre la figure 45.

```

35/
358 while (((HAL_GetTick() - startMillis) < sampleWindow)) // |
359 {
360 HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, 1); // active verin electrique 1
361 HAL_GPIO_WritePin(GPIOB, GPIO_PIN_13, 1); // active verin electrique 2
362 HAL_ADC_Start(&hadc1);
363 HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY);
364 sample = HAL_ADC_GetValue(&hadc1) ; // lecture de valeur analogique de la sortie du capteur
365 HAL_ADC_Stop(&hadc1);
366
367
368 if (sample < 1024) // enlever tout les valeur qui ne ssont pas logic logic
369 {
370 if (sample > signalMax)
371 {
372 signalMax = sample; // enregistre la valeur maximalle
373 }
374 else if (sample < signalMin)
375 {
376 signalMin = sample; // enregistre la valeur minimall
377 }
378 }
379 }
380
381
382 peakToPeak = signalMax - signalMin; // max - min = peak-peak amplitude
383 db = map(peakToPeak,20,900,49.5,90); // transforme la tension entre en DB
384 lcd_clear ();
385 lcd_put_cur(0,0);
386 lcd_send_string ("bd = ");
387 lcd_put_cur(0,4);
388 k++;
389
390 if (db <= 80)
391 {
392 sprintf ( msg , "%hu\r\n" , db ) ;
393 lcd_put_cur(0,4);
394 lcd_send_string (msg); // affiche le valeur DB
395

```

Figure.45 : ACTIVATION DES TRANSISTORS ET DU CAPTEUR, LIRE VALEUR DU CAPTEUR

Après cette étape la carte stm32 envoie une requête à la carte esp8266 pour enregistrer sur la base de données l'état du teste de fonctionnement ainsi que numéro de box du teste comme le montre la figure 46.

```

z = Serial.readString(); // lire de carte stm32 le path pour l'enregistrement des donnees |
if (z !=0)
break ;
}
Serial.print("ook");
Serial.printf("%s", Firebase.setBool(fbdo,z , true) ? "ook" : "err"); // envoie le resultat du test fonctionnel
z="";
c="";
while (1){
a = Serial.readString(); // lire de carte stm32 le path pour l'enregistrement des donnees
if (a !=0)
break ;
}
Serial.print("ook");
delay(10);
while (1){
b = Serial.readString(); // lire de la carte stm32 le numero de box de teste
if (b !=0)
break ;
}
Serial.print("ook");

Serial.printf("%s", Firebase.setString(fbdo, a , b) ? "ook" : "err"); // envoyer a la base de donnees le numero de box

```

Figure.46 : LA PROGRAMMATION DU ESP8266 POUR ENREGISTRES LES DONNEES SUR FIREBASE

La carte esp8266 va enregistrer les nouvelles informations sur notre base de données.

Exemple de la base de données avant et après le test Comme le montre la figure 47



Figure.47 : LA BASE DE DONNEES AVANT ET APRE LE TESTE

IV. Conclusion

Dans ce chapitre, nous avons cité toute démarche et étape réalisée faite pour notre étude pour avoir une conception convenable pour notre nécessité.

Chapitre III : Réalisation

I. Introduction

La réalisation est une partie très importante dans notre projet donc, nous avons essayé d'accumuler la plus part des cas que nous pouvons rencontrer dans notre système afin de visualiser tous les scénarios possibles.

II. Différent cas possible

Dans notre système il y a plusieurs scénarios et cas possible comme la montre figure 48

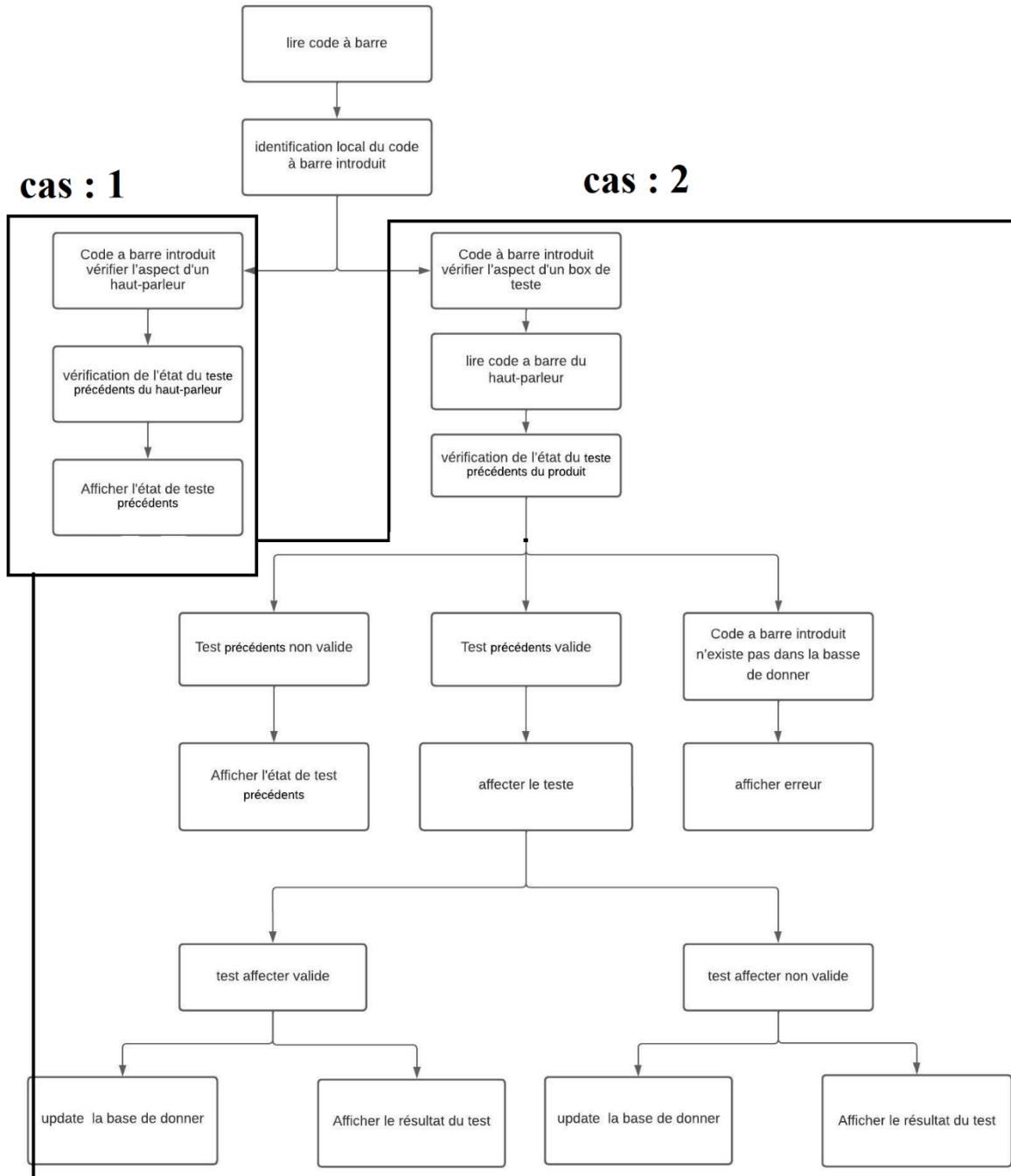


Figure.48 : CAS POSSIBLE

III. Premier cas

Dans le cas du 1^{er} cas le premier code à barre entrée est d'un haut-parleur le système ce contente juste de savoir l'état du teste précédent de l'article.

Exemple 1 :

Le haut-parleur qui a le code à barre 61940024 est validé sur le teste 1 dans notre base de données. Comme la montre figure 49

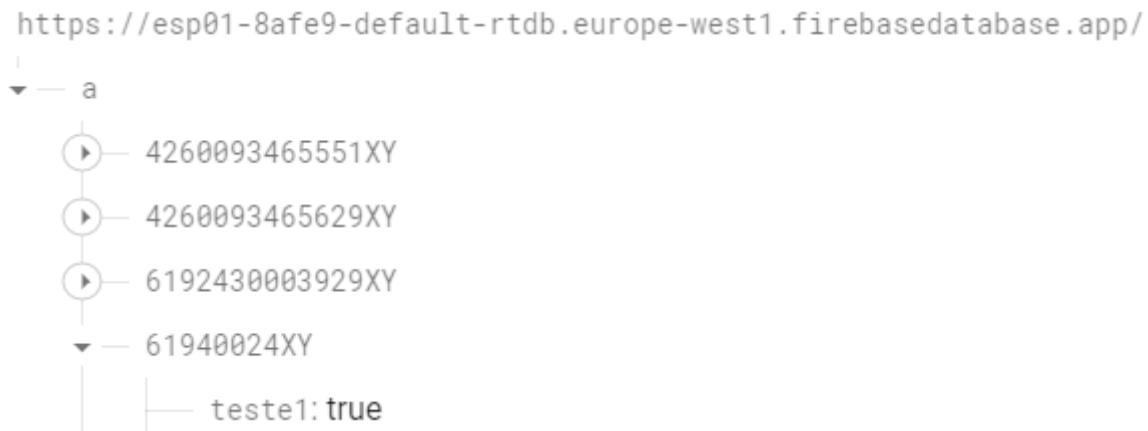


Figure.49 : ETAT DU TESTE 1 DU PRODUIT 61940024XY

Donc, le système nous affiche l'état du teste précédent. Comme la montre figure 50



**Figure.50 : AFFICHER L'ETAT DU TESTE1
61940024XY**

Exemple 2 :

Le haut-parleur qui a le code à barre 61940196607382 n'est pas validé sur le teste 1 dans notre base de données. Comme la montre figure 51



Figure.51 : ETAT DU TESTE 1 DU PRODUIT 420093465629XYXY

Donc, le système nous affiché l'état du teste précédent. Comme la montre figure 52



**Figure.52 : AFFICHER L'ETAT DU TESTE1
6194019607382XYXY**

Est si le code à barre du haut-parleur n'existe pas sur la base de données. Comme la montre figure 53



**Figure.53 : CODE A BARRE N'EXISTE PAS
SUR LA BASE DE DONNES**

Donc, le système nous affiche une erreur avec le numéro du code à barre entrée.

IV. Deuxième cas

Dans le 2^{eme} cas le premier code à barre entrée est du box ou le teste va être admis sur l'article.

Exemple d'un code à barre entrée. Comme la montre figure 54



Figure.54 : CODE À BARRE D'UN BOX

Le système affiche sur l'écran LCD le numéro box que nous avons choisi pour tester notre haut-parleur spécifique. . Comme la montre figure 55

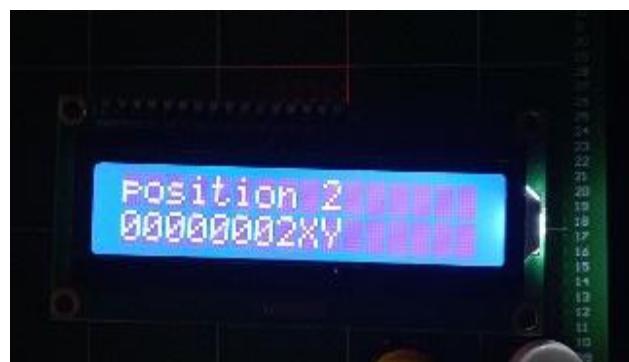


Figure.55 : AFFICHER LE CODE A BARRE ENTRE

Par la suite, nous entrons le code à barre de l'article cela fait que la carte stm32 envoie une requête à la carte esp8266 qui est connectée à notre base de données pour vérifier l'état du teste précédent s'il est validé ou non. Comme la montre figure 56



Figure.56 : BASE DE DONNER FIRE BASE

Donc dans cette étape il y a 3 possibilités :

1^{er} possibilité : Si le teste précédent de l'article n'est pas validé donc le système affiche false avec le numéro de code à barre est bien sur les vérins et le capteur de sons ne seront pas activés. Comme la montre figure 57



Figure.57 : TESTE PRECEDENT DE HAUT-PARLEUR NON
VALIDE

2^{eme} possibilité : Si le code à barre n'est pas dans la base de données le système affiche error est bien sur les vérins et le capteur de sons ne seront pas activés. Comme la montre figure 58



Figure.58 : CODE A BARRE N'EXISTE PAS SUR BASE
DE DONNES

3^{eme} possibilité : Si le teste précédent de l'article est validé donc, le système affiche truee avec le numéro de code à barre et les vérins avec le capteur de sons seront activés. . Comme la montre figure 59



Figure.59 : TESTE PRECEDENT DE HAUT-PARLEUR
VALIDE

Les vérins et le capteur de sons seront actif simultanément, pour la simulation des vérins électriques nous avons choisi d'utiliser des **LED**. . Comme la montre figure 60 et 61

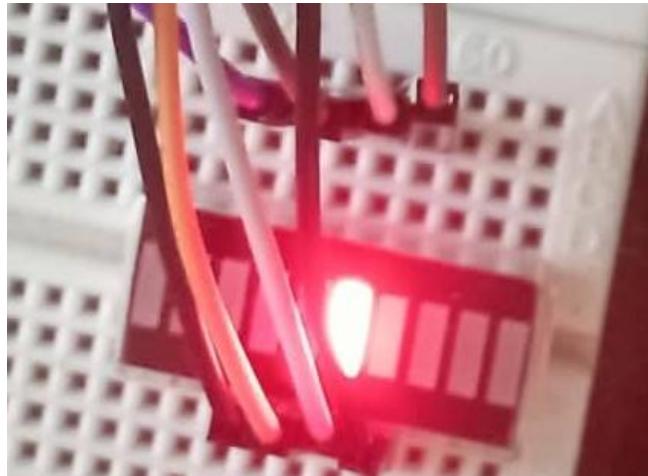


Figure.61 : ACTIVATION DE LED

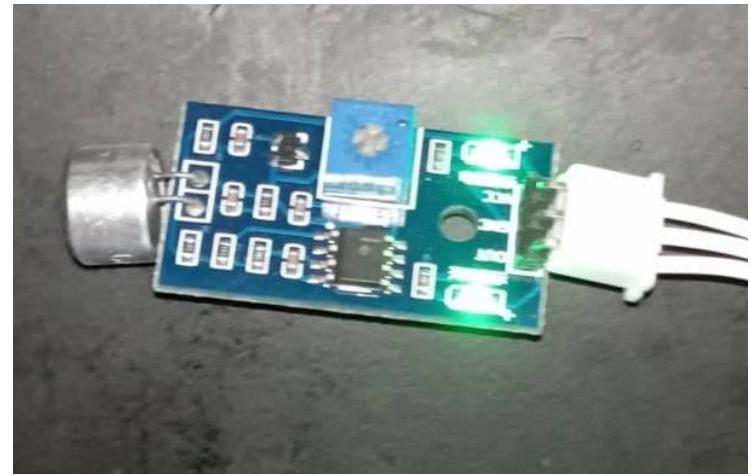


Figure.60 : ACTIVATION DU CAPTEUR DE SONS

Le capteur intercepte les sons dans une durée de temps bien définie.

Si le capteur de sons reçoit 85 dB ou plus donc le teste est validé le système nous affiche test validé. Comme la montre figure 62



Figure.62 : TESTE VALIDE

La base de données subit une mise à jour pour ajouter les détails de notre test. Comme la montre figure 63



Figure.63 : LA MISE A JOUR DE LA BASE DE DONNES

Si le capteur de sons ne reçoit pas au moins 85 dB donc, le teste n'est pas validé le système nous affiche teste non valide. Comme la montre figure 64

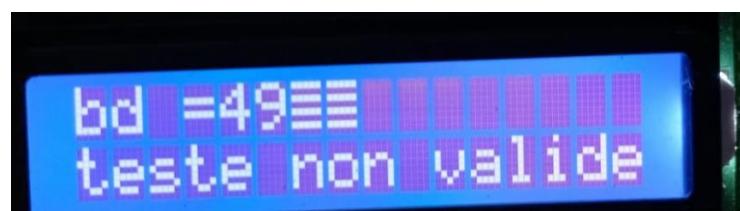


Figure.64 : TESTE NON VALIDE

La base de données subit une mise à jour pour ajouter les détails de notre test. . Comme la montre figure 65



Figure.65 : LA MISE A JOUR DE LA BASE DE DONNES

V. Conclusion

Dans ce chapitre, nous avons présenté les différents cas et scenario possible qu'on peut rencontrer lors du test des différents haut-parleurs avec leurs entrées sorties.

Conclusion générale

Dans notre projet nous avons mis en œuvre un testeur de haut-parleur qui a pour but d'assurer plus d'autonomie et de précision que la démarche manuelle .

Pour aboutir à ce résultat, nous avons commencé par le contexte général et étude Préalable.

Ensuite, nous avons analysé et spécifié les besoins nécessaires pour le développement de notre projet. Nous avons pu dégager les fonctionnalités principales que le système doit intégrer ainsi que les fonctionnalités secondaires qui répondent aux besoins non fonctionnels qu'elle doit satisfaire.

Par la suite, nous avons procédé à l'étape de conception et simulation, nous avons présenté les environnements matériels et logiciels de conception et de simulation utilisés afin de pouvoir choisir les différents matériel le plus adapte pour notre projet avec leur programmation.

Puis, nous avons cité tous les différents cas de scenario avec leurs paramètres.

Malgré que nous avons rencontré plusieurs problèmes durant la conception nous avons pu les dépasser et ce grâce à ce que nous avons appris tout au long de nos études académiques et l'assistance de nos encadreurs

En perspective, nous pouvons améliorer notre étude dans le but de mieux automatiser le fonctionnement de système par une interface graphique qui permet au utilisateur de mieux accéder à les différents paramètres et résultat du test sur la base de donner.

Webographie

- [1] <https://www.techno-science.net/definition/1263.html>.[consulté le 14-05-2022]
- [2] <https://www.maxicours.com/se/cours/application-le-haut-parleur/>.[consulté le 14-05-2022]
- [3] [https://fr.wikipedia.org/wiki/Boomer_\(haut-parleur\)](https://fr.wikipedia.org/wiki/Boomer_(haut-parleur)) .[consulté le 14-05-2022]