

UNIVERSITE KASDI MERBAH OUARGLA  
Faculté des Sciences Appliquées  
Département de Génie Electrique



Mémoire  
MASTER ACADEMIQUE  
Domaine : Sciences et technologies  
Filière : Génie électrique  
Spécialité : Electrotechnique Industrielle  
Présenté par :

Amir Djebiri

Hadjer Bakhaled

**Thème:**

# Etude et réalisation d'un compteur d'énergie intelligent

Soumis au jury composé de :

M<sup>r</sup> Benmakhlouf Abdeslam

MAA

Président

UKM Ouargla

M<sup>r</sup> Taibi Djamel

MAA

Encadreur/rapporteur

UKM Ouargla

M<sup>r</sup> Djarah Djalal

MAA

Examineur

UKM Ouargla

Année universitaire 2019/2020

# *Remerciements*

Avant tout, on remercie **ALLAH** le Tout-puissant de nous avoir donné le courage, la volonté et la patience de mener à terme ce présent travail dans des meilleures conditions.

On tient tout d'abord à remercier notre encadreur **Mr Taibi Djemal**, pour avoir accepté de nous encadrer, et pour le thème qu'il nous a proposé. Ainsi que toutes ces remarques constructives qui nous ont permis d'approfondir les connaissances scientifiques.

Nous remercierons particulièrement **Mr Razoug.R**, **Mr Benmakhlouf.A** et **Mr Djarah.D** pour ses aide précieuse, ses conseils et pour le temps qu'ils nous sont consacrés tout au long de ce travail.

Sans oublier on remercie **Charaf Eddine Cheraa** pour sa contribution et son aide qui nous ont été indispensable pour la réalisation de ce travail.

Merci à nos parents d'avoir su nous écouter et nous motiver.  
Merci à l'ensemble de nos amis(e), qui ont été là pendant les périodes de doute et de stress.

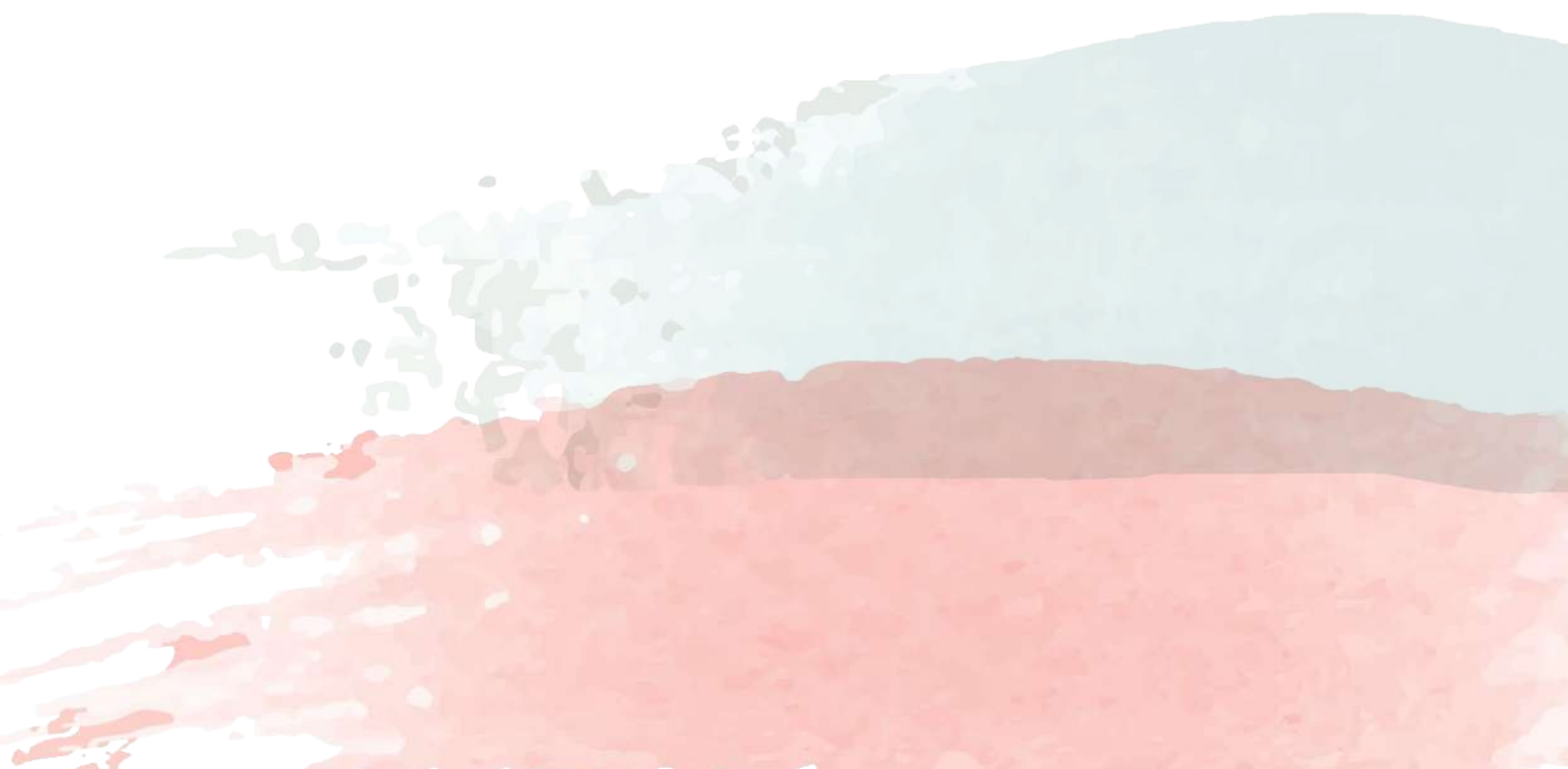
*Merci encore à tous...*

# *Dédicaces*

*Nos parents.*

*Nos frères et sœurs.*

*Et tout (es) nos amis.*



## Résumé

Dans ce projet, on s'est intéressé à la conception et à la réalisation d'un compteur intelligent d'énergie électrique.

Tout d'abord, on a étudié les compteurs d'énergie classique et électronique. Ensuite, on a développé la conception interne de ce compteur en utilisant des capteurs de tension et de courant et un microcontrôleur ESP32.

Enfin, on a réalisé une communication qui est basée sur une base de données et une application de smartphone qui permet de connecter le client au compteur.

## Abstract

In this project, we were interested in the design and realization of a smart energy meter.

First, we studied the conventional and the electronical energy meters. Then, we developed the internal design of this meter using voltage and current sensors and an ESP32 microcontroller.

Finally, we realized a communication based on a data base and a smartphone application so the meter and the client can be connected.

## المخلص

في هذا المشروع، قمنا بتصميم وإنجاز عداد طاقة ذكي. أولاً، درسنا عداد الطاقة الكلاسيكي و الإلكتروني. ثم قمنا بتطوير التصميم الداخلي للعداد باستعمال حساسات التيار و الجهد الكهربائي و المتحكم ESP32. و أخيراً أنشأنا وسيلة تواصل بين العداد و المستهلك و ذلك عبر قاعدة بيانات و تطبيق يعمل على الهواتف الذكية.



# Sommaire

Introduction Générale .....	2
<b>Chapitre I : Généralités sur les compteurs d'énergie électrique</b>	
I.1 Introduction .....	4
I.2 Le développement des compteurs d'énergie électrique .....	4
I.2.1 Compteur électromécanique .....	4
I.2.1-1 Définition .....	4
I.2.1-2 Le principe de fonctionnement et la conception .....	5
I.2.1-3 Les avantages et les inconvénients .....	6
a- Les avantages .....	6
b- Les inconvénients .....	7
I.2.2 Compteur électronique .....	7
I.2.2-1 Définition .....	7
I.2.2-2 Le principe de fonctionnement et la conception .....	8
I.2.2-3 Les avantages et les inconvénients .....	9
a- Les avantages .....	9
b- Les inconvénients .....	10
I.2.3 Compteur intelligent .....	10
I.2.3-1 Définition .....	10
I.2.3-2 Description générale des compteurs intelligents .....	10
I.2.3-3 Les avantages et les inconvénients .....	12
a- Les avantages .....	12
b- Les inconvénients .....	13
I.3 Conclusion .....	13
<b>Chapitre II : Conception matérielle et logicielle du projet</b>	
II.1 Introduction .....	15
II.2 ESP32 .....	15
II.2.1 Espressif et l'ESP32 .....	15
II.2.2 La définition de ESP32 .....	16
II.2.3 Les modules et les cartes de développement ESP32 .....	16
II.2.3-1 Les modules ESP32 .....	16
II.2.3-2 Les cartes de développement ESP32 .....	17
II.2.4 ESP32 WROVER-B T8 V1.8 .....	18
II.2.4-1 Les caractéristiques .....	18



II.2.4-2 Les entrées et les sorties .....	18
II.3 Les composants utilisés .....	20
II.3.1 Le capteur de courant ACS712 .....	20
II.3.2 Le capteur de tension ZMPT101B .....	21
II.3.3 Afficheur à cristaux liquides LCD 16x2 avec module I2C .....	22
II.3.4 Module relais 5V à 4 canaux .....	22
II.4 La programmation .....	23
II.4.1 Arduino IDE .....	23
II.4.2 Arduino IDE et l'ESP32 .....	23
II.4.3 Logiciel d'application .....	26
II.4.3-1 Flutter .....	26
II.4.3-2 Visual Studio Code .....	26
II.5 Conclusion .....	27
<b>Chapitre III : Réalisation pratique du compteur intelligent d'énergie</b>	
III.1 Introduction .....	29
III.2 Schéma synoptique du compteur intelligent d'énergie .....	29
III.3 Calcul de l'énergie consommée .....	31
III.4 Le Protocole de communication .....	31
III.4.1 Description du Protocole de communication .....	31
III.4.2 Conception et réalisation de l'application de smartphone .....	32
III.4.3 La base des données Firebase .....	34
III.5 La réalisation et les résultats pratiques du compteur intelligent d'énergie .....	34
III.5.1 Le schéma globale de compteur intelligent d'énergie .....	34
III.5.2 Réalisation pratique .....	36
III.5.3 Les résultats pratiques .....	37
III.6 Conclusion .....	39
Conclusion Générale .....	41
Bibliographie .....	42
Annexe A .....	43



### Abréviations

• <b>PDL</b>	Point de livraison
• <b>LCD</b>	Liquid Crystal Display
• <b>GSM</b>	The Global System for Mobile
• <b>SMS</b>	short message service
• <b>CPL</b>	Power-line communication
• <b>HF</b>	Haute fréquence
• <b>IOT</b>	internet of things
• <b>SOC</b>	A system on a chip
• <b>GPIO</b>	A general-purpose input/output
• <b>USB</b>	Universal Serial Bus
• <b>SD</b>	Secure Digital
• <b>UART</b>	A universal asynchronous receiver-transmitter
• <b>SPI</b>	The Serial Peripheral Interface
• <b>SDIO</b>	Secure Digital Input Output
• <b>LED</b>	A light-emitting diode
• <b>PWM</b>	Pulse-width modulation
• <b>I2S</b>	Inter-IC Sound
• <b>I2C</b>	Inter-Integrated Circuit
• <b>ADC</b>	analog-to-digital converter
• <b>BLE</b>	Bluetooth Low Energy
• <b>GND</b>	Ground
• <b>AC</b>	Alternating current
• <b>DC</b>	Direct current
• <b>IDE</b>	integrated development environment
• <b>SDK</b>	Software Development Kit
• <b>UI</b>	User interface
• <b>IOS</b>	iPhone Operating system
• <b>apk</b>	Android application package



### Symboles

• $u(t)$	Tension instantanée présente par le réseau	V
• $i(t)$	Courant instantané parcourant par le réseau	A
• $E$	L'énergie consommée	KWh
• $t$	Le temps	ms



## Liste des figures

<b>Figure I.1</b> : Compteur électromécanique .....	4
<b>Figure I.2</b> : Compteur électromécanique .....	5
<b>Figure I.3</b> : Les index Compteur électromécanique .....	5
<b>Figure I.4</b> : Compteur électromécanique .....	6
<b>Figure I.5</b> : Compteur électronique .....	7
<b>Figure I.6</b> : Compteur d'énergie électronique .....	9
<b>Figure I.7</b> : Compteur intelligent .....	10
<b>Figure I.8</b> : Module GSM de compteur intelligent .....	11
<b>Figure I.9</b> : Illustration du principe général du CPL .....	11
<b>Figure I.10</b> : Illustration du principe général du IOT .....	12
<b>Figure II.1</b> : Le logo de Espressif .....	15
<b>Figure II.2</b> : Module ESP32 .....	16
<b>Figure II.3</b> : Les modules ESP32 .....	16
<b>Figure II.4</b> : Les cartes ESP32 .....	17
<b>Figure II.5</b> : ESP32 WROVER-B T8 V1.8 .....	18
<b>Figure II.6</b> : La carte de ESP32 WROVER-B T8 .....	19
<b>Figure II.7</b> : Brochages du ESP32 WROVER-B T8 .....	19
<b>Figure II.8</b> : Capteur de courant ACS712 .....	20
<b>Figure II.9</b> : Effet Hall .....	20
<b>Figure II.10</b> : Capteur de tension ZMPT101B .....	21
<b>Figure II.11</b> : Signal de sortie de capteur de tension .....	21
<b>Figure II.12</b> : Écran LCD 16x2 avec module I2C .....	22
<b>Figure II.13</b> : Module relais 5V à 4 canaux .....	22
<b>Figure II.14</b> : Arduino IDE .....	23
<b>Figure II.15</b> : Flutter .....	26
<b>Figure II.16</b> : Dart .....	26
<b>Figure II.17</b> : VScode .....	26
<b>Figure III.1</b> : Schéma synoptique du compteur d'énergie .....	29
<b>Figure III.2</b> : Organigramme du fonctionnement de notre système .....	30
<b>Figure III.3</b> : Le Protocole de communication .....	31
<b>Figure III.4</b> : Interface principale de l'application (Data page) .....	32
<b>Figure III.5</b> : La deuxième interface de l'application (Control page) .....	33

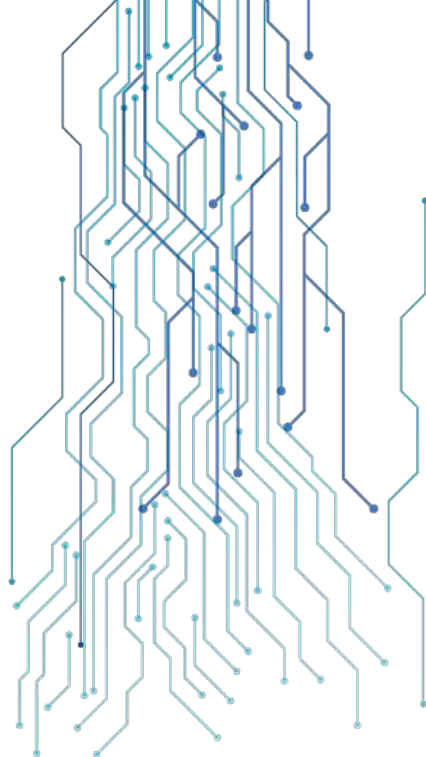


<b>Figure III.6</b> : La troisième interface de l'application (About page) .....	33
<b>Figure III.7</b> : Les extensions utilisées dans Visual Studio Code .....	34
<b>Figure III.8</b> : L'interface de la base des données Firebase .....	34
<b>Figure III.9</b> : Le schéma global du compteur intelligent d'énergie .....	35
<b>Figure III.10</b> : Réalisation pratique d'un compteur intelligent de d'énergie.	36
<b>Figure III.11</b> : Structure du compteur intelligent d'énergie .....	36
<b>Figure III.12</b> : Mesure de courant .....	37
<b>Figure III.13</b> : Mesure de tension .....	37
<b>Figure III.14</b> : Transfert de données avec l'application de smartphone.....	38
<b>Figure III.15</b> : L'affichage des mêmes données dans la base des données Firebase .....	38

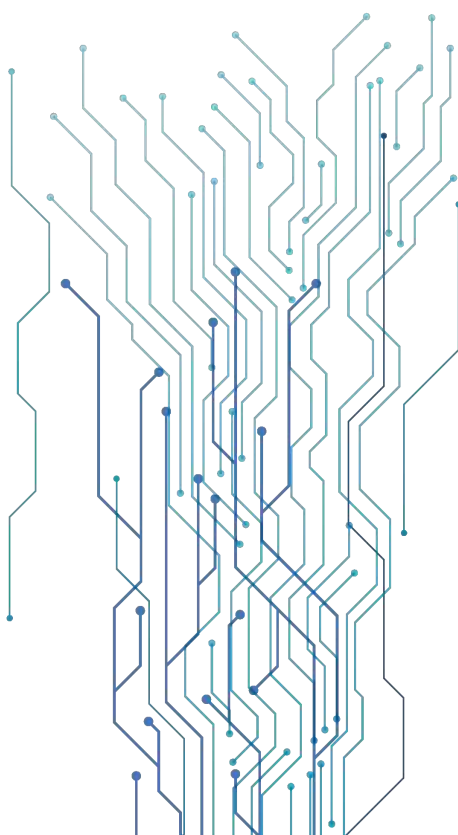


## Liste des tableaux

<b>Tableau II.1</b> : Les caractéristiques de la carte ESP32 WROVER-B .....	18
<b>Tableau II.2</b> : Les caractéristiques du capteur de tension .....	21



# Introduction générale





Après avoir découvert l'électricité et sa diffusion parmi les pays, les compteurs électriques sont apparus pour calculer l'énergie consommée. Mais à cause du grand nombre des compteurs dans chaque cité, la facturation et la collection des informations et des données de consommation est devenue un souci pour les fournisseurs. D'autre part la consommation quotidienne du client augmente d'une façon rapide. Alors pour résoudre ces problèmes, les inventeurs et les ingénieurs décident d'utiliser les nouvelles technologies informatiques et économiques pour mettre en œuvre ses compteurs électriques qui permettent de faciliter la facturation et contrôler la consommation du client.

Le compteur d'énergie qui a pour rôle le calcul de la consommation ainsi que l'établissement de la facture, est un élément essentiel pour les sociétés de l'électricité. Au départ, seuls les compteurs électromécaniques ont été utilisés. Après, il y a eu recours aux compteurs électroniques.

Vue le progrès technologique dans les domaines informatiques et télécommunications, l'électricien est amené à créer des nouvelles catégories des compteurs d'énergie. Dans ce contexte, l'idée de remplacer le compteur classique par un compteur intelligent ou communiquant est apparue. C'est un compteur programmable permettant d'envoyer et de recevoir des données tels que la consommation. Par conséquent, le client va être capable de suivre sa consommation en termes d'énergie et de coût.

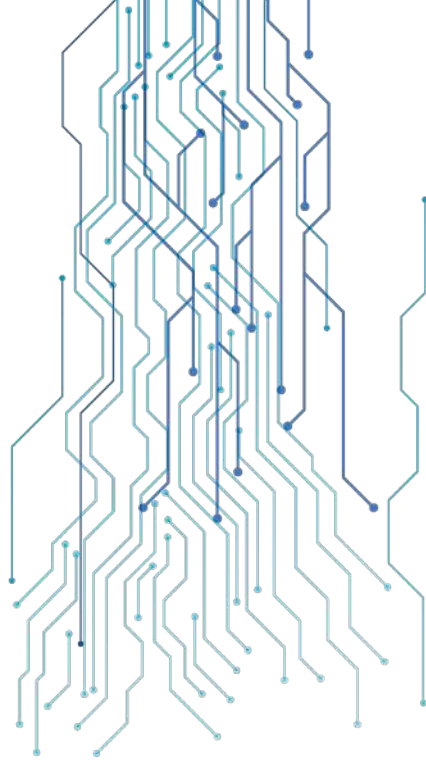
Dans ce projet, on va utiliser une carte ESP32 Wrover B, un capteur de courant, un capteur de tension, un afficheur LCD 16x2 et une application de smartphone afin de réaliser un compteur d'énergie communiquant.

Ce mémoire est subdivisé en trois chapitres, et il est organisé comme suit :

Dans le premier chapitre, on va faire une étude générale sur les compteurs d'énergie électrique et leurs développements.

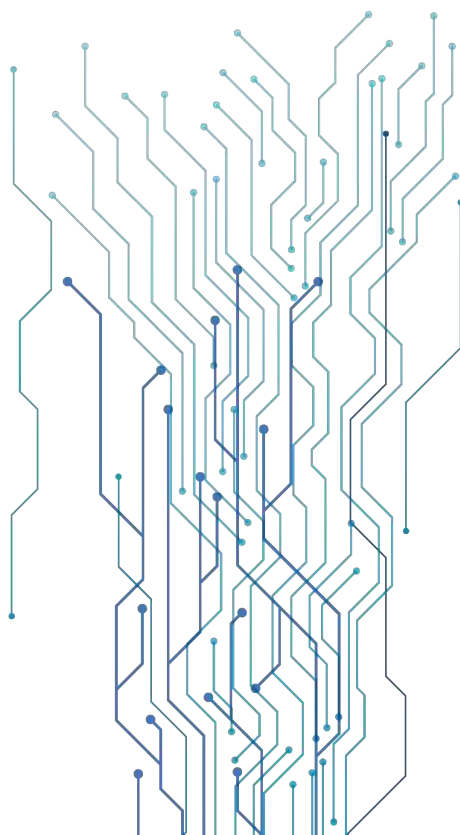
Le deuxième chapitre est consacré à la conception matérielle et logicielle du projet, soit software et hardware.

Dans le dernier chapitre, on va s'intéresser à la communication entre le compteur d'énergie et les clients à base de ESP32 et smartphone ainsi que la réalisation pratique du compteur proposé.



# Chapitre I

## *Généralités sur les compteurs d'énergie électrique*



## I.1 Introduction

Tout logement, parc ou une entreprise alimentée en électricité est doté d'un compteur électrique, pour mesurer la consommation de ce local.

Avec le temps, le développement des compteurs d'énergie électrique a passé de trois générations, le compteur électromécanique, le compteur électronique et le compteur intelligent.

Dans ce chapitre, on s'intéresse particulièrement au développement des compteurs d'énergie, leur conception, leurs principes de fonctionnement, et les avantages et les inconvénients de chaque type de compteurs.

## I.2 Le développement des compteurs d'énergie électrique

Un compteur électrique est un organe électrotechnique **indispensable pour toute installation électrique** servant à mesurer et quantifier la quantité d'énergie électrique consommée dans un lieu comme un habitation, industrie, une école, une administration, une entreprise ...etc. Il est utilisé par les fournisseurs d'électricité afin de facturer la consommation d'énergie au client. Cette énergie est calculée en fonction de la puissance instantanée et du temps d'utilisation. Le compteur électrique donne des informations en kilowattheure (kWh) et **1 kWh** est équivalent à  **$3,6 \times 10^6$  J**.

A l'origine ces appareils étaient de conception électromécanique, ils sont remplacés dorénavant par des compteurs électroniques. Les nouvelles versions de compteurs électriques sont des compteurs communiquant appelés parfois compteurs intelligents. [1] [2]

### I.2.1 Compteur électromécanique

#### I.2.1-1 Définition :

Les compteurs électromécaniques sont les compteurs les plus anciens et les plus traditionnels installés grâce à sa grande qualité est sa robustesse et sa simplicité d'utilisation. Il s'agit de la première génération des compteurs installés et ils utilisent un dispositif mécanique de comptage pour afficher un seul indice de consommation qui est l'énergie. Ce compteur est généralement situé dans le logement lorsqu'il s'agit d'un logement individuel. Dans le cas d'un immeuble comportant plusieurs logements, il est situé à l'extérieur, souvent dans un compartiment ou local technique qui lui est réservé.



Figure I.1 : Compteur électromécanique

### I.2.1-2 Le principe de fonctionnement et la conception :

Les éléments essentiels et principaux du compteur sont visibles à travers le capot transparent à savoir :

(A) L'inducteur « Intensité » constitué par quelques spires de gros fil.

(B) L'inducteur « Tension » constitué par une bobine comportant un grand nombre de spires de fil fin.

(C) Le disque en aluminium constituant le rotor.

(D) L'aimant de freinage.

(E) Le totaliseur d'énergie constitué d'un ensemble d'engrenages qui actionne un dispositif d'affichage. [3]

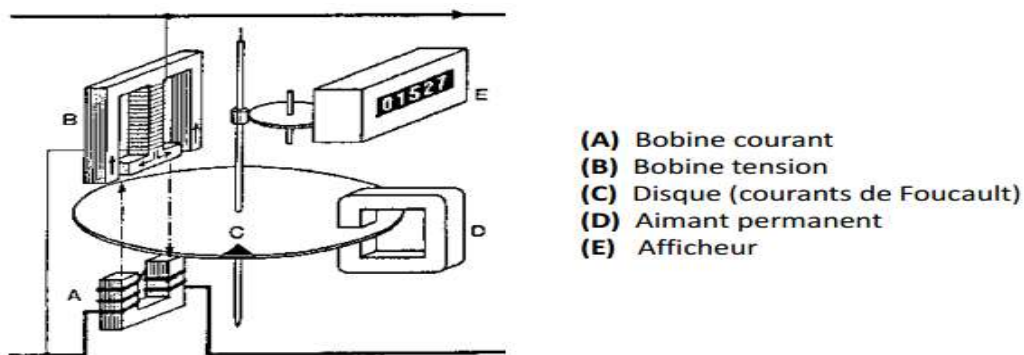


Figure I.2 : Compteur électromécanique [2]

Le compteur électromécanique a pour fonction de comptabiliser la quantité d'énergie électrique consommée dans un lieu alimenté par cette énergie. Il s'agit d'un boîtier repérable par sa forme carrée. Il se fixe à la paroi au moyen de trois points d'attache. Il est équipé d'un index à défilement mécanique. Cet index comporte :

- 5 chiffres.
- Le numéro de matricule ou PDL (point de livraison) à fournir pour souscrire un abonnement correspond aux trois derniers chiffres, ce numéro permet de distinguer le compteur lorsqu'il se situe au même endroit que ceux de voisins.
- Un disque rotatif en aluminium matérialise la consommation du logement en temps réel. [4]



Figure I.3 : Les index Compteur électromécanique



Le principe de fonctionnement des compteurs électromécaniques est basé sur l'électromagnétisme, cela dépend du principe du scientifique anglais Ferraris, qui a développé la théorie en 1885, et cette théorie dit (il est possible de générer un couple dans un élément rotatif en mouvement libre s'il y a deux flux magnétiques et deux écoulements qui coupent entre eux et s'il y a un angle entre eux et dans ce cas l'élément rotatif est le disque et le flux résultant des enroulements de courant et de tension.



**Figure I.4 :**  
**Compteur électromécanique**

Le comptage de l'énergie s'effectue en comptant le nombre de rotations d'un disque mobile qu'est monté sur un axe tournant avec un mécanisme de comptage mécanique. Ce disque est soumis aux champs magnétiques alternatifs produits grâce à deux électroaimants disposés au-dessus et en dessous du disque.

Chaque aimant est traversé par un courant électrique, l'un est parcouru par le courant circulant dans le fil de phase et l'autre par un courant proportionnel à la tension du réseau. Le disque opère un mouvement de rotation dont la vitesse est proportionnelle à la consommation d'électricité et il indique la consommation d'électricité depuis sa mise en service. En plus de cela pour les compteurs triphasés, ils sont constitués des mêmes éléments que les compteurs monophasés, mais à la différence de ceux-ci, ils comportent deux ou trois bobines tension et intensité. Leur équipement mobile est constitué soit d'un disque, soit de deux ou trois disques solidaires.

### **I.2.1-3 Les avantages et les inconvénients d'un compteur électromécanique (Classique) :**

#### **a- Les avantages :**

Ce type de compteur est le plus ancien des affichages mécaniques, il présente l'avantage de :

- Haute fiabilité lors de la coupure de courant et même en cas de court-circuit, ils peuvent continuer à fonctionner après avoir éliminé toutes les conséquences.
- Convient aux réseaux de faible qualité.
- Le prix d'achat est inférieur aux prix des appareils électroniques.
- Pas besoin de configuration supplémentaire. Tout ce qui est nécessaire est le câblage correct.

- De plus en cas de détérioration ou de vandalisme, la dernière formation enregistrée reste affichée ce qui rend sa lecture toujours possible.
- Longue durée de vie grâce à sa grande robustesse et sa grande qualité.
- Sa facilité d'installation et d'utilisation.

**b- Les inconvénients :**

Tandis que ce compteur électromécanique procure à leurs utilisateurs une foule d'avantages, il leur pose des inconvénients et des problèmes comme :

- Précision réduite avec consommation réduite. Cela est dû à une diminution du champ électromagnétique.
- Petite classe de précision. Il ne dépasse pas une valeur de 2 unités.
- Défaillances de suivi avec une forte variation des courants de charge.
- Coefficient élevé de consommation intérieure. C'est-à-dire le compteur lui-même est capable d'influencer partiellement la quantité dans la réception d'électricité.
- Dimensions relativement grandes.

C'est pour ça la technologie a évolué et sont apparus les compteurs numériques. Cependant, il est très aisé de les programmer pour compter l'électricité qui a été prélevée ou injectée dans le réseau.

**I.2.2 Compteur électronique****I.2.2-1 Définition**

C'est un compteur d'énergie électrique numérique qui fonctionne avec un système de comptage électronique, il s'agit de la deuxième génération des compteurs installés (après le compteur électromécanique), sur lequel on peut lire les informations qui sont affichées sur un dispositif d'affichage à cristaux liquides (LCD). Ce type de compteur est plus sensible aux surintensités et surtensions, et tout particulièrement à la foudre et pour mesurer les fortes intensités. [3]



**Figure I.5 : Compteur électronique**

### I.2.2-2 Le principe de fonctionnement et la conception

Maintenant, place aux compteurs électroniques. Ils sont facilement reconnaissables grâce à leur conception basée sur les circuits et les composants électroniques modernes, ils se composent d'un cadran d'affichage digital de la consommation d'électricité et de deux boutons poussoirs S et D pour faciliter la lecture des index et des informations du compteur.

- **La touche D (Défilement)** : elle permet d'accéder à l'index du compteur.
- **La touche S (Sélection)** : elle permet de faire défiler les informations qui sont relatives à l'appareil telles que le numéro de série du compteur ; l'option tarifaire ; la puissance instantanée ; le réglage du disjoncteur choisi ; la puissance que vous avez souscrite, le contrôle du fonctionnement de l'écran.

Pour un compteur d'électricité numérique, la formule fondamentale de la mesure de la puissance active est toujours représentée par l'intégral :

$$P_a = \frac{1}{T} \int_0^t u(t)i(t)dt$$

Avec  $P_a$  : la puissance active (w).

$u(t)$  : tension instantanée présente par le réseau.

$i(t)$  : courant instantané parcourant par le réseau.

Et pour calculer l'énergie consommée, on a l'équation suivante :

$$E = P_a . t$$

Avec  $E$  : l'énergie consommée (Kwh).

$P_a$  : la puissance active (Kw).

$t$  : le temps (h).

Alors le compteur électronique détecte le signal de tension et de courant par un capteur de tension et un capteur de courant, puis le signal de tension et de courant provenant des capteurs sont multipliés pour obtenir une capacité instantanée et par l'intégration et la multiplication avec le temps de ce dernier, nous obtenons l'énergie consommée en kilowattheures, après ces mesures sont stockées dans des enregistrements. Le compteur est caractérisé par un voyant lumineux sur la façade du compteur indique qu'il fonctionne bien. Ce voyant clignote plus ou moins vite selon la quantité d'électricité consommée.

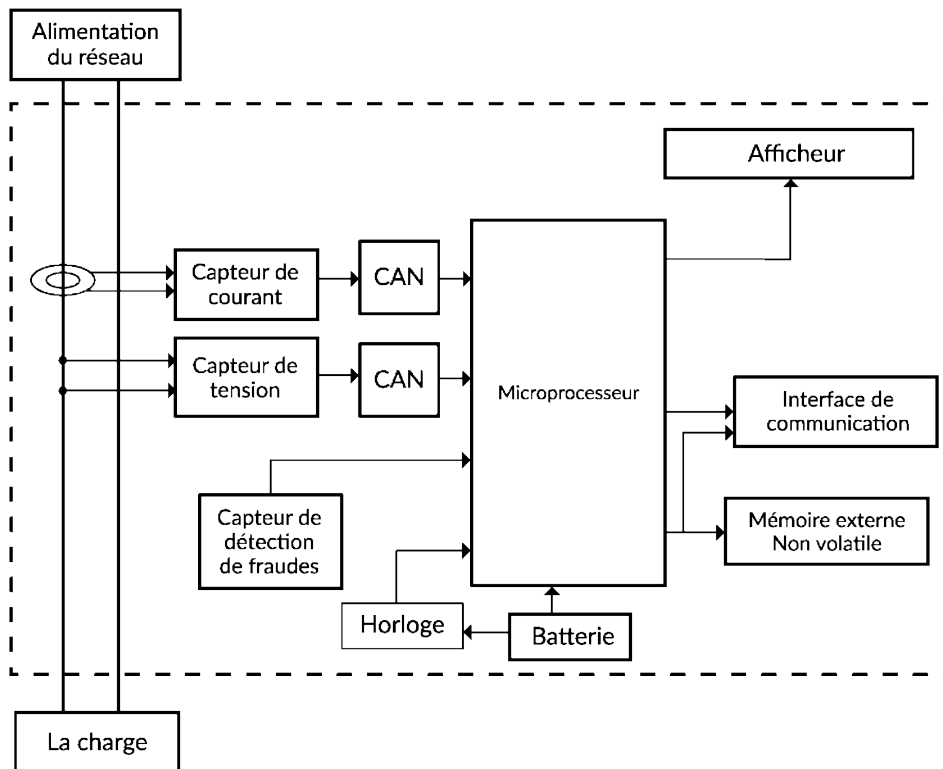


Figure I.6 : Compteur d'énergie électronique [2]

Outre ses fonctionnalités de comptage, le compteur électronique est mieux armé que son homologue à disque pour protéger l'installations en cas de foudre.

### I.2.2-3 Les avantages et les inconvénients d'un compteur électronique (numérique)

#### a- Les avantages

- Le compteur numérique (affichage LCD) présente l'avantage de pouvoir afficher différentes informations : (tarifs du KWh, Watt, tension).
- Les données calculées ainsi que les paramètres programmés sont enregistrées sur une mémoire non-volatile permettant leur sauvegarde en cas de coupure d'alimentation.
- Ces compteurs sont plus sensibles aux surintensités et surtensions, et tout particulièrement la foudre et les pannes de courant pourront être détectées plus rapidement.
- Ce type de compteur est caractérisé par la précision, fiabilité, modularité, souplesse...etc.
- Erreur minimale.
- Dimensions compactes par rapport aux appareils à induction.
- Manque de pièces mécaniques les plus susceptibles de s'user.

**b- Les inconvénients :**

Les compteurs électroniques présentent certains inconvénients tels que :

- Haute sensibilité et instabilité aux fortes modifications réseau.
- Prix élevé, surtout quand il s'agit d'appareils multi-tarifs.
- La réparation coûte très cher.
- Problème de détection des défauts.
- Difficulté à effectuer des réparations.
- Sensibilité à un changement brusque de température.

**I.2.3 Compteur intelligent****I.2.3-1 Définition**

Un smart meter, ou un compteur intelligent en français, est un compteur d'énergie électrique communiquant présente une nouvelle génération de compteurs qui se caractérise par des technologies avancées de communication qui mesure la consommation électrique d'un bâtiment, d'une entreprise



**Figure I.7 : Compteur intelligent**

ou d'un foyer avec une manière détaillée et précise en temps réel. Ce compteur enregistre la quantité d'énergie consommée de façon électronique qui permet notamment de transmettre des données de consommation, comme les index, et qui peut recevoir des informations ou des ordres.

**I.2.3-2 Description générale sur les compteurs intelligents :**

Le « comptage intelligent » est un concept qui remonte à la fin des années 80, il désigne un système comprenant des compteurs communiquant de manière mono ou bidirectionnelle à travers un réseau de communication fixe avec un système informatique central situé chez le gestionnaire de réseau. Ce système peut communiquer avec l'extérieur ceci signifie qu'il peut recevoir des ordres et envoyer des données et des informations sans l'intervention physique d'un technicien. Il a été conçu pour faciliter la vie des clients et les consommateurs. Il est équipé d'un système informatisé reliés au système électrique du foyer, ce compteur calcule à 1 kwh près le débit de l'électricité consommée. Le chiffre précis en temps réel est fourni par le compteur afin d'informer le consommateur et le distributeur. De plus il transmet simplement les informations qu'il affiche et qui sont directement connectées aux bases de données des fournisseurs d'énergie. Ces informations facilitent les relevés à distance, le suivi précis des consommations et la fraude.

La conception et le principe du fonctionnement d'un compteur intelligent d'énergie varie d'un compteur à l'autre ça dépend du protocole de communication et du nombre des fonctionnalités, par exemple :

- **Le compteur d'énergie intelligent CPL** : ce compteur communiquant utilise une technologie connue depuis les années cinquante : le Courant Porteur en Ligne (CPL) qui permet d'envoyer et de recevoir les informations dans les câbles électriques existant dans l'habitation. Le principe de cette technologie consiste à coupler un signal haute fréquence (HF, un signal à plus haute fréquence et de faible énergie) au signal 50 Hz du réseau électrique (figure 9). Le signal HF se propage sur l'installation électrique et peut être reçu et décodé à distance. Ainsi le signal CPL est reçu par tout récepteur CPL de même catégorie se trouvant sur le même réseau électrique. [3]

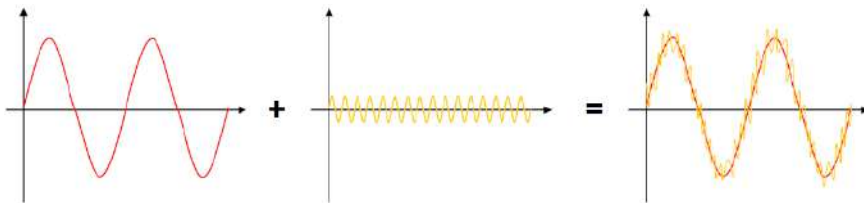


Figure I.9 : illustration du principe général du CPL [3]

- **Le compteur d'énergie intelligent avec module GSM** : ce type de compteur intelligent utilise un mode de communication basé sur un module GSM qui envoie les données et reçoit des ordres à travers des SMS entre les compteurs et agence commerciale de consommation qui collecte les informations dans la région et fait les factures.



Figure I.8 : Module GSM de compteur intelligent

- **Le compteur d'énergie intelligent IOT** : ce compteur utilise une technologie qui s'appelle IOT « internet of things ». Cette technologie permet d'envoyer les données et recevoir des ordres en temps réel entre plusieurs appareils ensemble, ce mode de communication nous permet d'avoir beaucoup de fonctionnalités comme : contrôler tous les appareils dans la maison avec un seule télécommande, réduire la consommation d'énergie, éviter des problèmes électriques comme la surcharge.

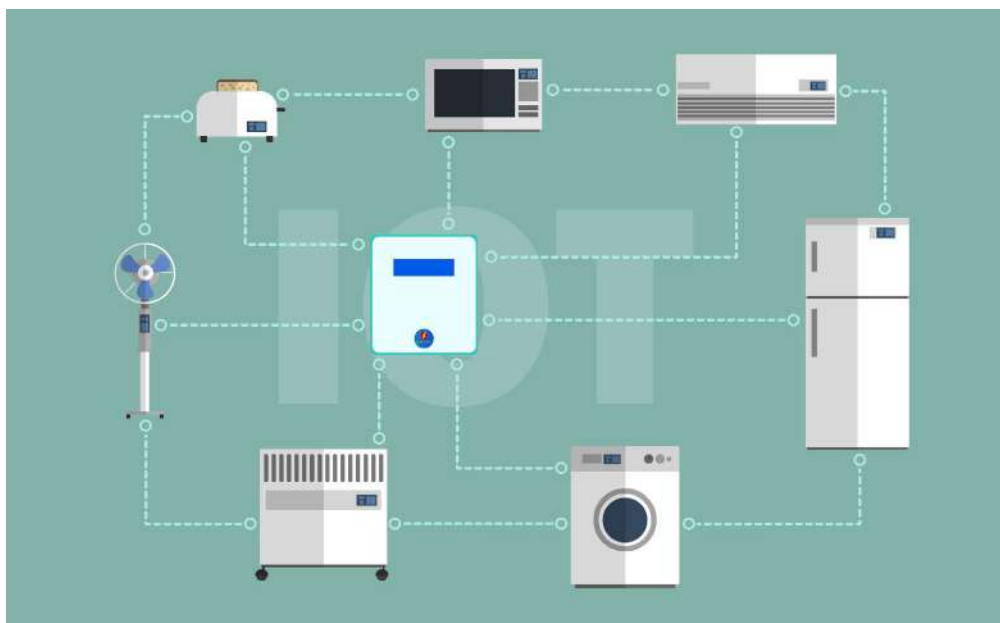


Figure I.10 : illustration du principe général du IOT

### I.2.3-3 Les avantages et les inconvénients d'un compteur

**intelligent :**

#### **a- Les avantages :**

- Le « smart meter » permet de relever et contrôler la consommation électrique à distance et permet d'éviter de déranger les clients par les agents que doivent faire la lecture sur place du compteur.
- Il sauvegarde les consommations, compteur par compteur, pour finalement rendre le rapport au gestionnaire.
- Ce type de compteur fait le calcul de la consommation d'énergie avec une manière détaillée et précise en temps réel.
- Une détection plus rapide et automatique des pannes.
- Contribue de réduire le budget et la facture de l'électricité puisque le consommateur dispose d'informations beaucoup plus détaillées relatives à sa consommation qui lui permettent de faire des économies d'énergie en adaptant ses habitudes.
- Faciliter la vie du consommateur dans la mesure où il :
  - Permet d'être facturé par rapport à sa consommation réelle, et non plus par rapport à une estimation de consommation.
  - Ne nécessite plus l'intervention d'un technicien pour la mise en service, ou la modification de la puissance.
  - Lui permet de suivre sa consommation exacte sur internet pour la mieux maîtriser et faire des économies d'énergie.



- Absence de dérangement pour les opérations liées au compteur.

**b- Les inconvénients:**

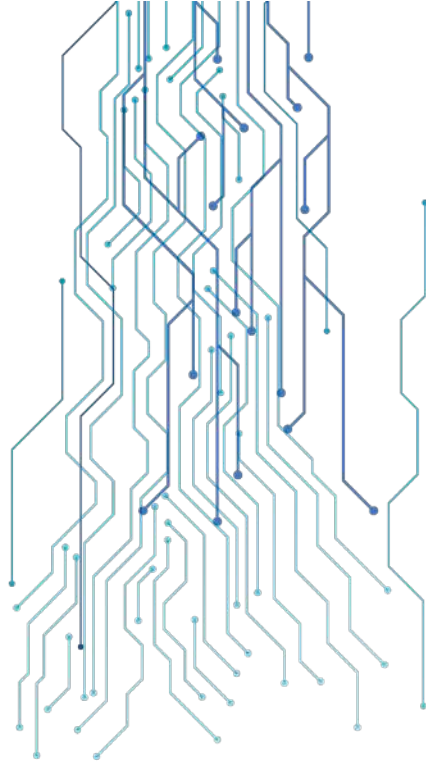
- Difficile à installer puisqu'il se caractérise par des technologies avancées et très sensibles.
- Prix élevé et la réparation coûte très cher.

### **I.3 Conclusion**

Dans ce chapitre, à partir d'étude générale sur l'évolution de la technologie utilisée dans la mesure de la consommation électrique, nous avons parlé sur les types des compteurs d'énergie électrique présents dans les installations électriques et nous avons aussi donné une vision globale et détaillé sur les compteurs et leur fonctionnement ainsi que leurs avantages et les problèmes.

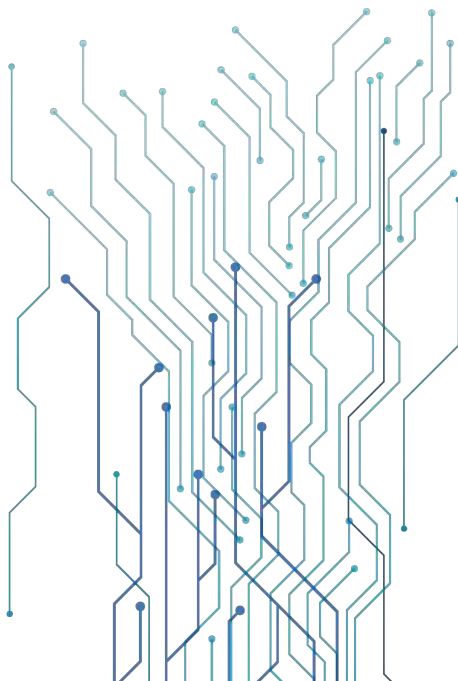
Alors , d'après tout ça on conclut que le compteur d'énergie électrique est un élément important et essentiel dans les réseaux et les installations électriques et il est entendu que plus les compteurs sont précis et intelligents dans la mesure on va trouver que l'équité sera atteinte entre le consommateur et les sociétés de distribution électrique, contrairement aux anciennes technologies des compteurs et à cet effet on va détailler dans le chapitre suivant sur les composants électroniques utilisés dans notre projet pour réaliser le circuit d'un compteur intelligent.





# Chapitre II

## *Conception matérielle et logicielle du projet*



## II.1 Introduction

Depuis longtemps, la création des circuits électroniques à partir des composants comme (résistance, capacité, inductance, transistor...) a été difficile, et sa modification est vraiment complexée parce que ses circuits sont fabriqués pour faire un travail spécifique et pour faire un petit changement il faut passer par plusieurs opérations comme (des calculs, Soudage, refaire les schémas...).

Mais après le développement de la technologie électronique, la création des systèmes à base d'un circuit électronique est devenue très simple et facile grâce aux cartes de développements et des modules programmables, qui permettent de réaliser plusieurs fonctions dans le même système.

Dans ce chapitre, nous allons parler spécialement de la carte ESP32 WROVER-B T8 V1.8 mentionnant ses caractéristiques et nous donnerons le schéma de principe de cette carte avec les composants que nous utiliserons dans ce système.

## II.2 ESP32

### II.2.1 Espressif et l'ESP32

Espressif Systems est une multinationale publique, société de semi-conducteurs établis en 2008, avec siège à Shanghai et des bureaux en Grande Chine, en Inde et en Europe. Ils ont une équipe passionnée d'ingénieurs et de scientifiques de partout dans le monde, axée sur le développement de pointe Wifi et Bluetooth, faible puissance IOT solutions. [5]



Figure II.1 : Le logo de Espressif

La puce ESP8266wifi, faite par Espressif. Elle est apparue sur certains sites chinois au milieu de 2014 et au début il a été utilisé comme « pont » pour connecter les microcontrôleurs (Arduino...) aux réseaux wifi grâce à son très faible coût.

En raison du fait que le micrologiciel original n'était pas bien documenté, qu'il a quelques bogues et qu'il n'offrait que des fonctionnalités « standard », la communauté des fabricants a mis au point des micrologiciels alternatifs (dont le plus célèbre est sûrement NodeMCU) pour exploiter pleinement la puissance de la puce et construire des systèmes complets, sans avoir besoin de microcontrôleurs externes.

En septembre 2016, après une phase de test bêta qui a duré quelques mois, Espressif a annoncé et mis à disposition le successeur de l'ESP8266, appelé ESP32. [6]

### II.2.2 La définition de ESP32

ESP32 est un système à faible coût et à faible consommation d'énergie sur une puce (SOC) série avec Wi-Fi et double mode Bluetooth capacités.



Figure II.2 : Module ESP32

ESP32 offre une plateforme robuste, qui aide à répondre aux exigences continues en matière d'utilisation efficace de la puissance, de conception compacte, de sécurité, de haute performance et de fiabilité.

Espressif fournit des ressources matérielles et logicielles de base pour aider les développeurs d'applications à réaliser leurs idées à l'aide du matériel de la série ESP32. Le cadre de développement de logiciels par Espressif est destiné au développement d'applications IOT avec Wi-Fi, Bluetooth, gestion de l'énergie et plusieurs autres fonctionnalités du système.

### II.2.3 Les modules et les cartes de développement ESP32

Espressif conçoit et fabrique différents modules et cartes de développement pour aider les utilisateurs à évaluer le potentiel de la famille de puces ESP32.

#### II.2.3-1 Les modules ESP32

Il s'agit d'une famille de modules basés sur ESP32 avec certains composants clés intégrés, y compris un oscillateur à cristaux et un circuit correspondant à l'antenne. Les modules constituent des solutions prêtes à l'emploi pour l'intégration dans les produits finaux. Combinés à quelques composants supplémentaires, tels qu'une interface de programmation, des résistances d'amorçage et des têtes de broche, ces modules peuvent également être utilisés pour évaluer les fonctionnalités de ESP32.



Module ESP32-WROOM-32  
(avant et arrière)



Module ESP32-SOLO-1  
(avant et arrière)



Module ESP32-WROVER  
(avant et arrière)

Figure II.3 : Les modules ESP32

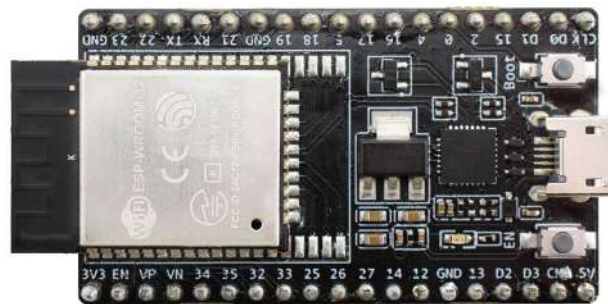
### II.2.3-1 Les cartes de développement ESP32

Les Caractéristiques des différentes cartes de développement sont :

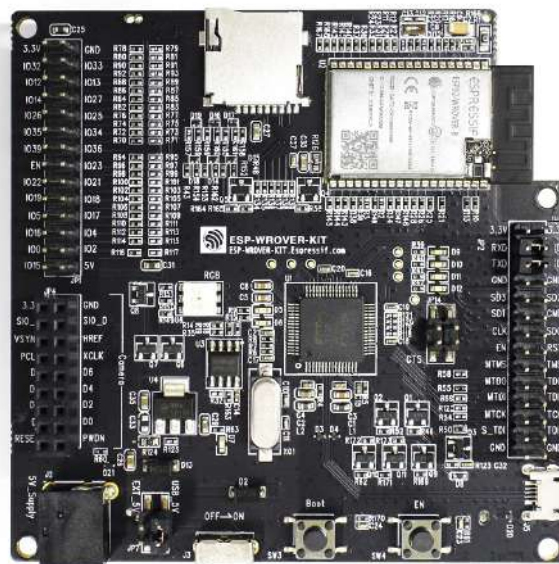
- Accès à différentes broches GPIO ESP32.
- Différentes interfaces : USB, JTAG.
- Différents périphériques : pavés tactiles, écrans LCD, emplacements pour cartes SD, en-têtes femelles pour modules de caméra, etc. [5]



**La carte  
ESP32-PICO-KIT V4.1**



**La carte  
ESP32 DevKitC V4**



**La carte  
ESP-WROVER-KIT**

**Figure II.4 : Les cartes ESP32**

## II.2.4 ESP32 WROVER-B T8 V1.8

### II.2.4-1 Les caractéristiques

Dans notre projet on a utilisé la carte ESP32 WROVER-B T8 V1.8, qui a les caractéristiques suivantes :



**Figure II.5 : ESP32 WROVER-B T8 V1.8**

Puce	ESPRESSIF-ESP32 WROVER-B
Microprocesseur	Microprocesseur Xtensa LX6 simple/double cœur 32 bits
Horloge embarquée	Oscillateur à cristal de 40 MHz
SRAM	520 KO
Flash	4 MO
Nombre des pins	38 pins
Interface modulaire	Carte SD, UART, SPI, SDIO, LED PWM, TV PWM, I2S, I2C, IRGPIO, capteur tactile à condensateur, ADC, préamplificateur DACLNA
Tension de fonctionnement	2.3 V-3.6 V
WI-FI	Plage de fréquence : 2.4 GHz ~ 2.5 GHz
Bluetooth	Protocole : Conforme à la norme Bluetooth v4.2 BR/EDR et BLE

**Tableau II.1 : Les caractéristiques de la carte ESP32 WROVER-B**

### II.2.4-2 Les entrées et les sorties

- **Micro USB** : utilisée pour alimenter et connecter ESP32 à l'ordinateur.
- **L'entrée d'alimentation** : par laquelle ESP32 est connecté à une source d'alimentation externe en tant que batterie si le USB n'est pas utilisé pour le fonctionnement, pour le ESP32 WROVER-B fonctionnant à 2.3 V-3.6 V.
  - **Porte SD** : il contient 4 pins (CS, MOSI, SCLK, MISO), ou On peut lire la carte SD.
  - **Les pins d'énergie** : Comprend des pins d'alimentation de 5V et 3,3V, des ports de mise à la terre GND et un port Vin par lequel l'énergie peut être tirée de l'entrée d'alimentation.
  - **Les pins de multifonction** : c'est les pins qui peuvent être pin numérique ou analogique en même temps.
  - **Bouton reset** : Pour redémarrer le code chargé sur la carte. [7]

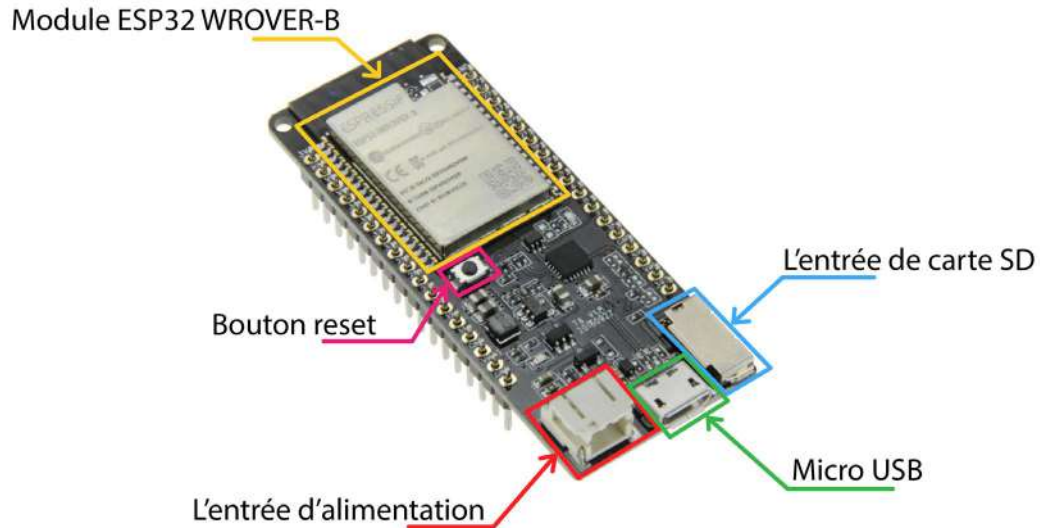
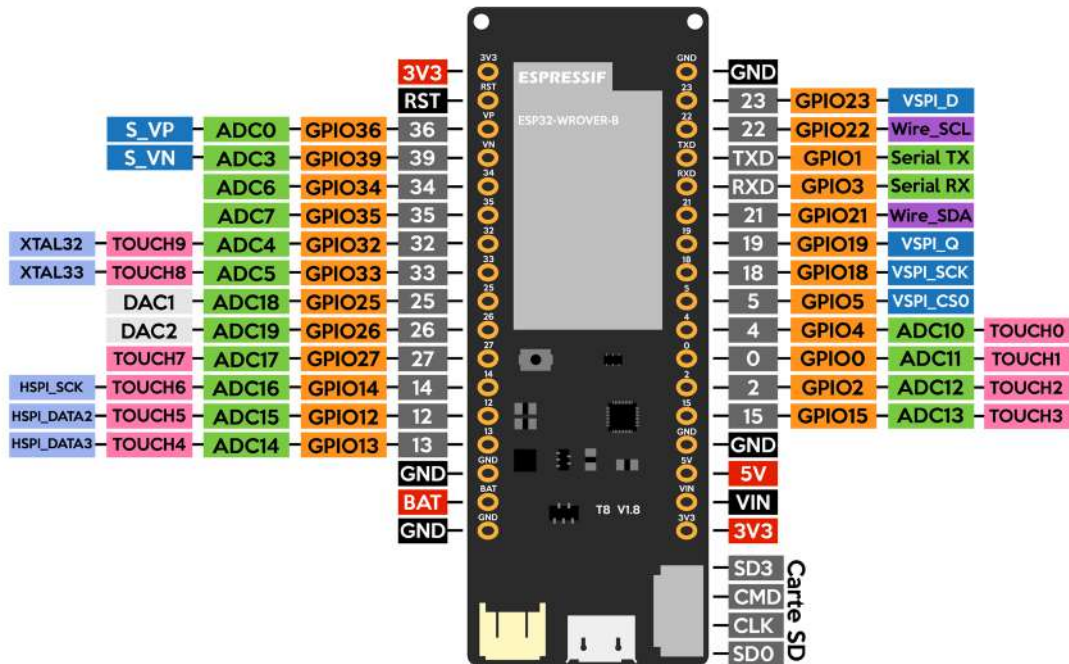


Figure II.6 : La carte ESP32 WROVER-B T8 V1.8



## II.3 Les composants utilisés

### II.3.1 Le capteur de courant ACS712

Dans ce travail, on a choisi un capteur de courant à effet Hall. C'est un dispositif qui détecte le courant électrique dans un fil, et génère un signal proportionnel à celui-ci. Le signal généré pourrait être une tension analogique ou numérique. Il peut ensuite être stocké pour une analyse plus approfondie dans un système d'acquisition de données. Le capteur de courant utilisé est un capteur à effet Hall, de modèle ACS712 20A, connecté en série avec la ligne dont le courant doit être mesuré.



Figure II.8: Capteur de courant ACS712

Le capteur de courant à effet Hall exploite l'effet Hall pour produire une tension image exacte du courant à mesurer. L'apparition d'un champ électrique transversal et d'une différence de potentiel dans un métal ou un semi-conducteur parcouru par un courant électrique lorsqu'on l'introduit dans un champ d'induction magnétique perpendiculaire à la direction du courant, est appelée effet Hall. [8]

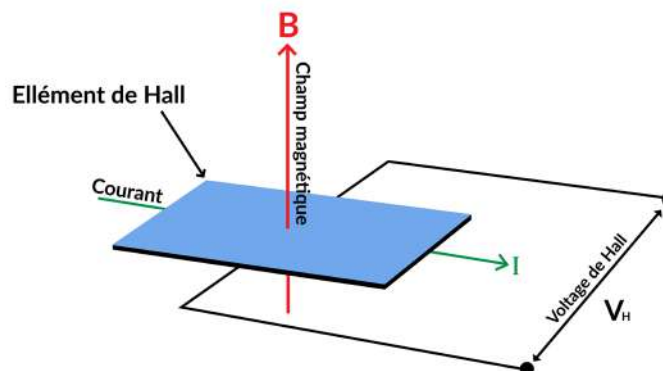


Figure II.9 : Effet Hall

Le capteur ACS712 dispose d'une isolation électrique entre le circuit de commande et le circuit auquel nous voulons mesurer le courant.

- Une large gamme de courants qu'ils peuvent mesurer jusqu'à 30 A.
- La résistance d'entrée est faible, ce qui rend l'énergie dissipée à l'intérieur est négligeable.
- La valeur de la résistance de sortie est exprimée en pourcentage de la tension d'alimentation, ce qui facilite le comptage dans ce capteur.

### II.3.2 Le capteur de tension ZMPT101B

ZMPT101B est un capteur de tension de type AC basé sur un transformateur de tension intégré. Et il est caractérisé par :

- Une haute précision.
- Une bonne fiabilité.
- Une large gamme de mesure (0-250 AC).
- Très facile à utiliser et à construire.
- Un potentiomètre pour régler la sortie ADC.



Figure II.10: Capteur de tension ZMPT101B

Le transformateur de tension intégré l'a réduit à 2,5 V (pic). Puis un décalage de 2,5 V a été inséré pour déplacer le signal vers un niveau positif où le ESP32 peut être détecté, donc le pic va être décaler a 5V. [9]

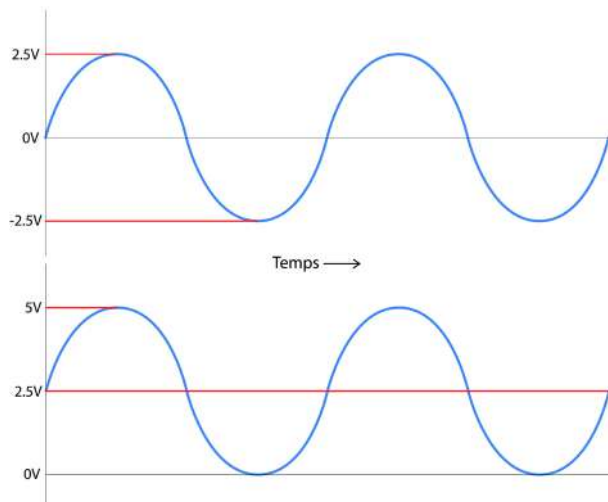


Figure II.11 : signal de sortie de capteur de tension

Courant d'entrée nominal	2mA
Courant de sortie nominal	2mA
Rapport de transformation	1000:1000
Tension d'isolation	4000V
Température de fonctionnement	-40°C ~ + 60°C

Tableau II.2 : les caractéristiques de capteur de tension

On peut utiliser ce capteur de tension dans plusieurs applications :

- Comptage (compteurs d'énergie électrique).
- Détection de défaut de masse.
- Equipement électrique domestique.
- Equipement de contrôle électrique et protection des relais.



### II.3.3 Afficheur à cristaux liquides LCD 16x2 avec module I2C

Ce type d'écran en cristal est le plus populaire en raison de son faible prix et sa programmation facile.

Ce que nous allons trouver dans notre projet est le LCD1602 Crystal Display, qui a deux lignes et 16 colonnes, et qui est alimenté par le pilote HD44780 Hitachi.



Figure II.12 : Écran LCD 16x2 avec module I2C

Les broches de cet afficheur sont :

- **RS** : cette broche nous détermine dans quelle mémoire nous écrivons des données, l'enregistreur de données ou l'enregistreur d'instructions.
- **R / W** : pour lire ou écrire, ce port spécifie soit sélectionner le mode lecture ou écriture.
- **Enable pin** : cette broche, si elle est activée, permet d'écrire sur l'écran LCD.
- **(D0-D7)** : ce sont les broches par lesquelles nous transmettons des données aux enregistreurs.

### II.3.4 Module relais 5V à 4 canaux

C'est une carte d'interface de relais 5V à 4 canaux de bas niveau, et chaque canal a besoin d'un courant de commande de 15-20mA.

Il peut être utilisé pour contrôler divers appareils et équipements avec un grand courant.

Il est équipé de relais à courant élevé qui fonctionnent sous AC 250V 10A ou DC 30V

10A. Il a une interface standard qui peut être contrôlée directement par microcontrôleur.

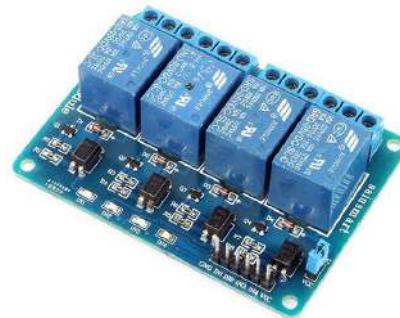


Figure II.13 : Module relais 5V à 4 canaux

Ce module est optiquement isolé du côté haute tension pour des exigences de sécurité et aussi empêcher la boucle de terre lors de l'interface avec le microcontrôleur. [11]

## II.4 La programmation

### II.4.1 Arduino IDE

L'environnement de développement Arduino IDE est l'outil utilisé pour écrire des codes avec le langage Arduino C, il est ensuite converti en une forme exécutive qui peut être placée sur le microcontrôleur.



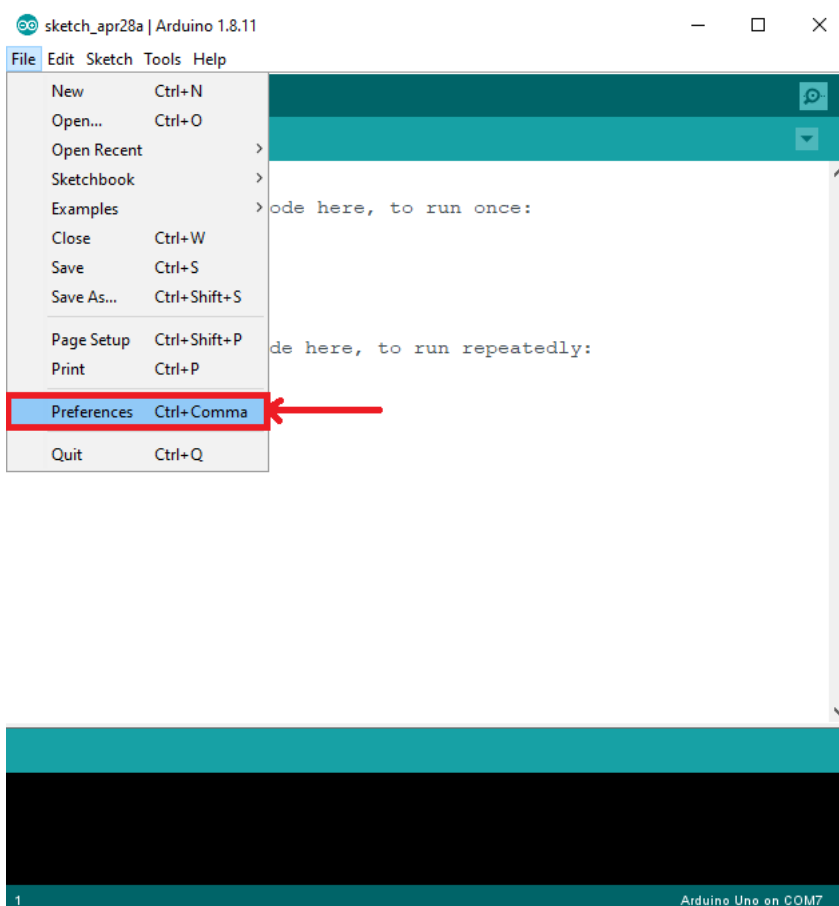
Figure II.14 : Arduino IDE

Cet environnement de développement est simple et facile à manipuler, et il est presque sans complications dans l'aspect général, il contient seulement ce que le programmeur a besoin pour commencer à développer et en même temps pour charger le code directement au microcontrôleur.

### II.4.2 Arduino IDE et l'ESP32

Dans notre projet, on a programmé l'ESP32 WROVER avec l'Arduino IDE, et pour faire ça il faut suivre les étapes suivantes :

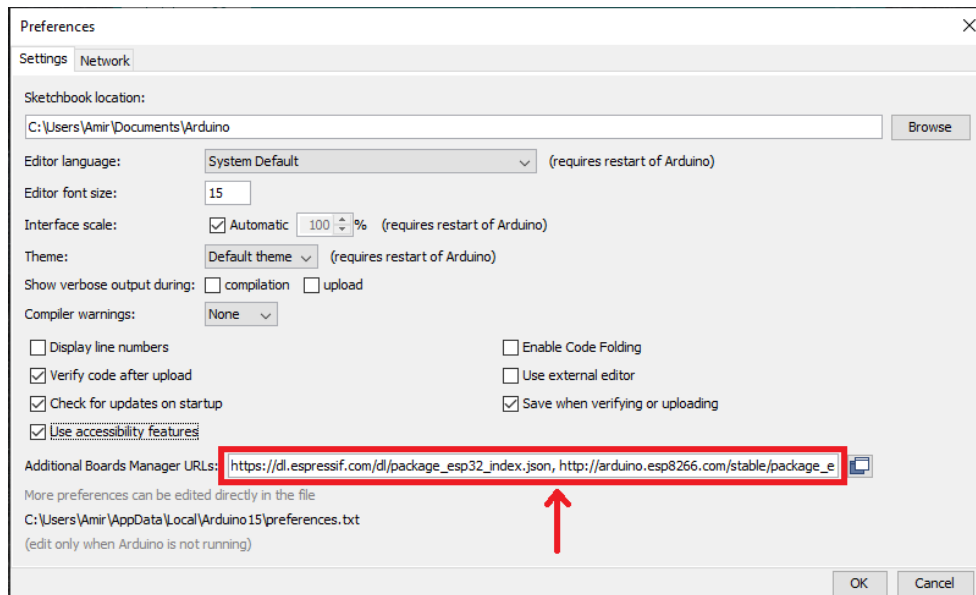
- En ouvre le programme Arduino IDE est dans la barre des menus on choisit "File" après "Preferences".



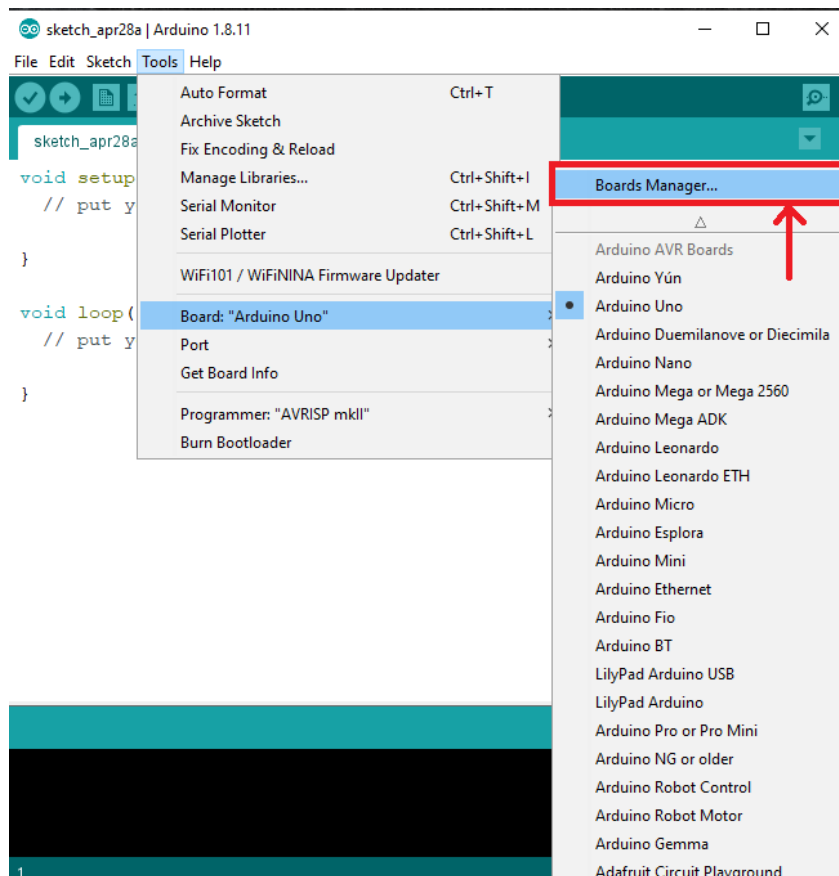
- Après la fenêtre de "Preferences" s'ouvre, et dans l'espace de "Additional Boards Manager URLs" on copie ce lien et on click "OK" :

[https://dl.espressif.com/dl/package\\_esp32\\_index.json](https://dl.espressif.com/dl/package_esp32_index.json),

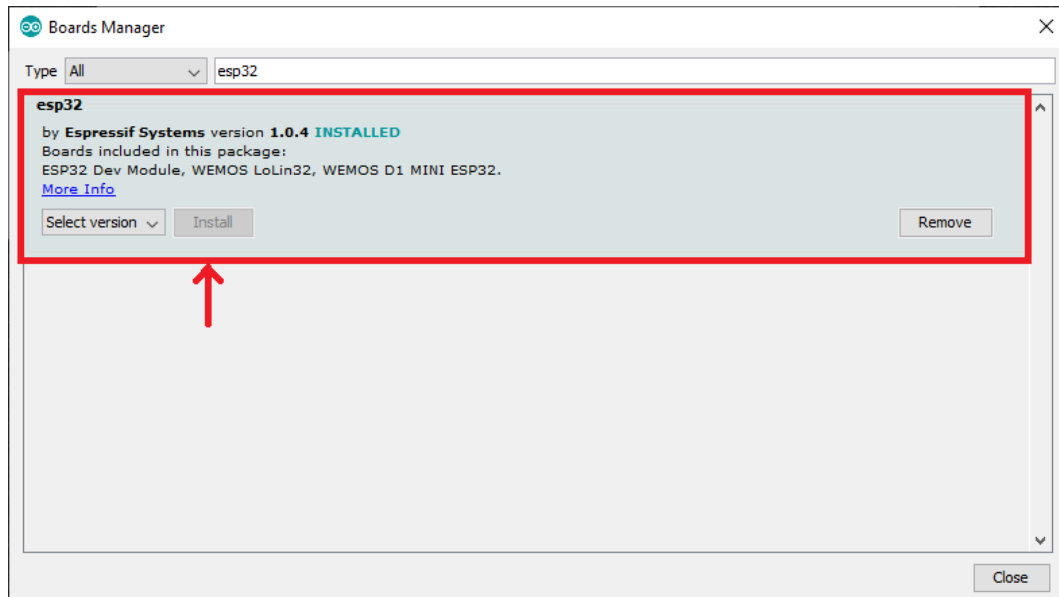
[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)



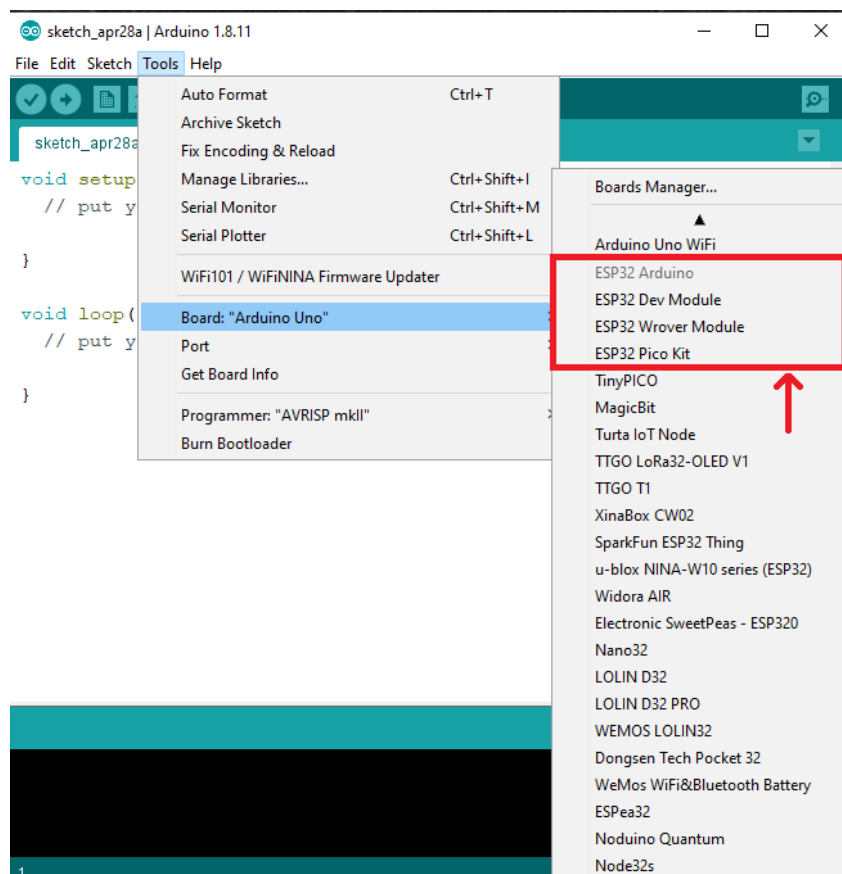
- Dans la deuxième étape on ouvre la liste de "Tools" et choisit "Board" après "Boards Manager..." :



- Dans la fenêtre de "Boards Manager" on recherche le mot "ESP32", un résultat est affiché de "Espressif Systems" qu'il contient le software des cartes ESP32, on click "Install" et on attend quelques minutes.



- À la fin d'installation on retour à la liste de "Tools" après "Board" et maintenant on peut programmer n'importe quel carte ESP32.



### II.4.3 Logiciel d'application :

#### II.4.3-1 Flutter :

Les applications mobiles sont devenues très populaires. Heureusement, il existe de nombreux outils de programmation disponibles pour les développeurs qui veulent les créer. Parmi ces outils, il y a Flutter, qui s'est distingué ces derniers temps.



Figure II.15 :  
Flutter

Flutter est un framework d'interface utilisateur mobile gratuit et open-source créé par Google et sorti en mai 2017. En quelques mots, il vous permet de créer une application mobile native avec une seule base de code. Cela signifie que vous pouvez utiliser un langage de programmation et une base de code pour créer deux applications différentes (pour iOS et Android).

Le flutter se compose de deux parties importantes :

- Un SDK (Software Development Kit) : une collection d'outils qui vont vous aider à développer vos applications. Cela inclut des outils pour compiler votre code en code machine natif (code pour iOS et Android).
- Un Framework (UI Library baser sur des widgets) : Une collection d'éléments d'interface réutilisables (boutons, entrées de texte, curseurs, etc.) que vous pouvez personnaliser pour vos propres besoins.

Pour développer avec Flutter, vous utiliserez un langage de programmation appelé Dart. La langue a été créé par Google en Octobre 2011, mais il a amélioré au cours de ces dernières années. Dart se concentre sur le développement frontal, et vous pouvez l'utiliser pour créer des applications mobiles et web. Si vous connaissez un peu de programmation, Dart est un langage de programmation d'objets typés. Vous pouvez comparer la syntaxe de Dart à JavaScript.



Figure II.16 :  
Dart

#### II.4.3-2 Visual Studio Code :

Visual Studio Code (Vscodé) est un éditeur de code source développé par Microsoft qui peut être exécuté sur Windows, macOS et Linux. Il est gratuit, open-source, et fournit le support pour le débogage ainsi que le contrôle de version Git intégrer, les points forts de la syntaxe, les extraits, et ainsi de suite.



Figure II.17 :  
VScodé

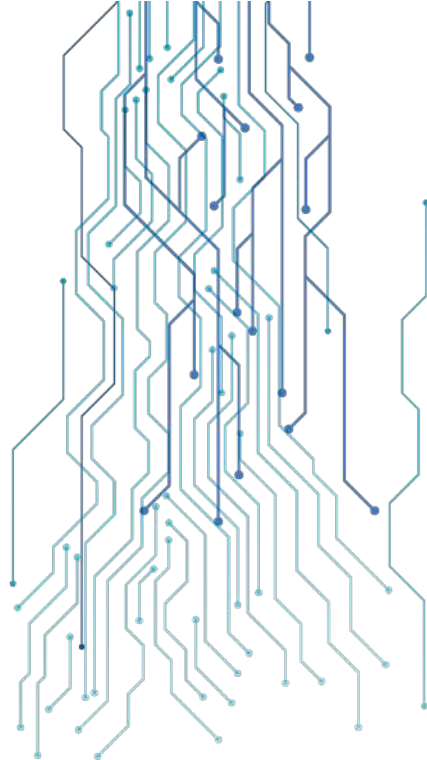


L'interface utilisateur de Vscode est hautement personnalisable, car les utilisateurs peuvent passer à différents thèmes, raccourcis clavier et préférences.

Vscode a été initialement annoncé en 2015 comme un projet open-source hébergé sur Github avant de sortir sur le web un an plus tard. Depuis lors, l'éditeur de code de Microsoft a gagné en popularité parmi les développeurs.

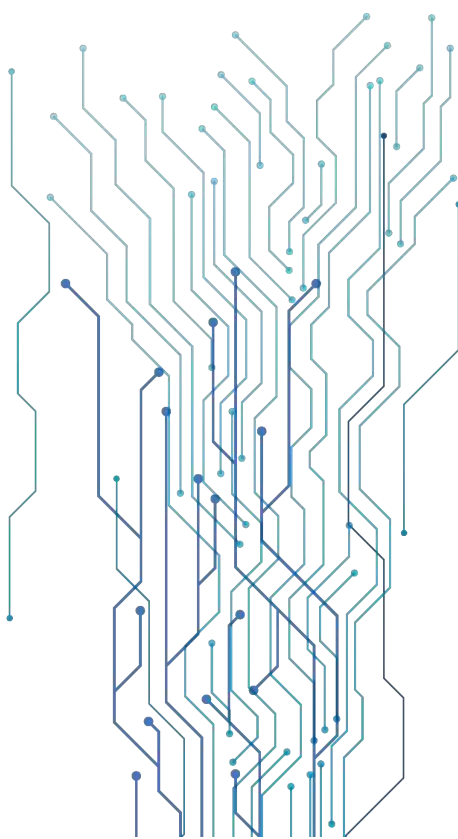
### **II.5 Conclusion**

Lorsque la carte ESP32 a été conçue, l'un des principaux objectifs de la conception était d'être facile à utiliser pour toutes les catégories de personnes et de rendre n'importe qui capable de faire de nombreux types de projets électroniques sans avoir besoin de connaître les grandes technologies. A cet égard, nous avons vu l'utilisation de cette carte, ainsi que quelques capteurs, pour réaliser un compteur intelligent d'énergie électrique qui lit les données à distance, que nous allons discuter dans le côté pratique du projet.



## Chapitre III

*Réalisation pratique du  
compteur intelligent  
d'énergie*



### III.1 Introduction

L'objectif de ce dernier chapitre est d'utiliser la technologie de communication afin de rendre le compteur d'énergie intelligent.

Dans la première partie, le schéma synoptique du compteur intelligent d'énergie est proposé. La deuxième partie est consacrée au protocole de communication utilisé.

La réalisation et les résultats pratiques du compteur d'énergie communicant sont présentés à la fin de ce chapitre.

### III.2 Schéma synoptique du compteur intelligent d'énergie

Le compteur intelligent d'énergie est constitué des éléments de bases du compteur d'énergie ainsi que l'ESP32 qui est le microprocesseur du projet et cette carte est nécessaire pour la communication entre le compteur standard et le client.

Le schéma du compteur intelligent d'énergie proposé est illustré par la figure III.1 suivante.

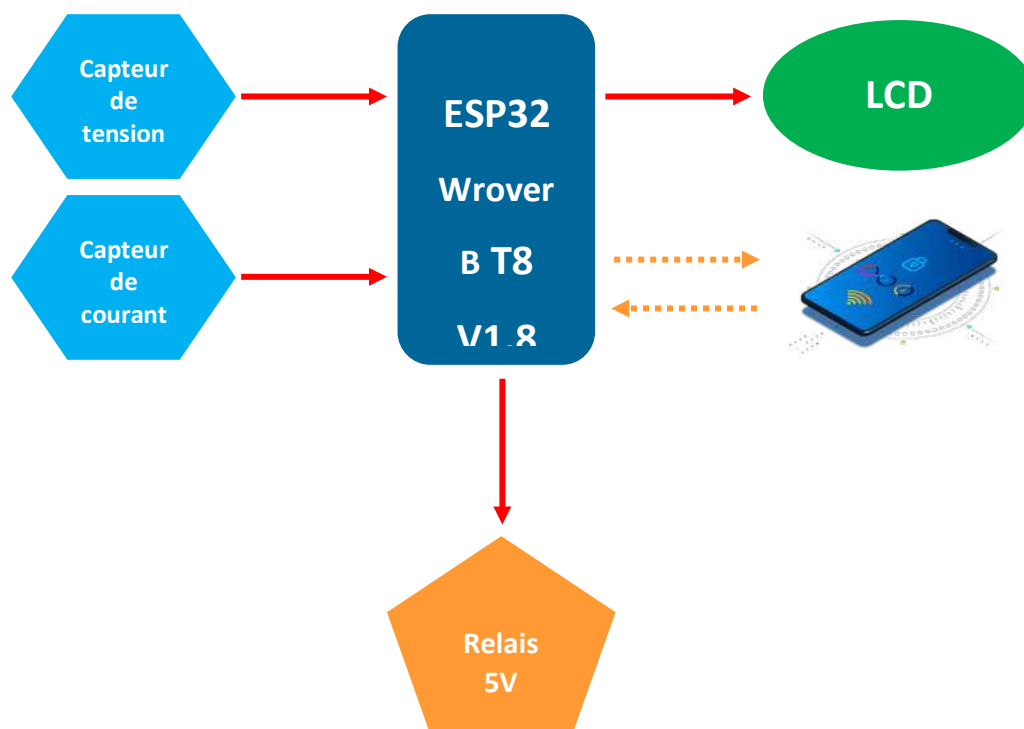


Figure III.1 : Schéma synoptique du compteur d'énergie



L'organigramme de la figure III.2 explique les étapes du fonctionnement de notre système afin de synchroniser les données avec l'ESP32 d'un côté et le smart phone d'autre côté.

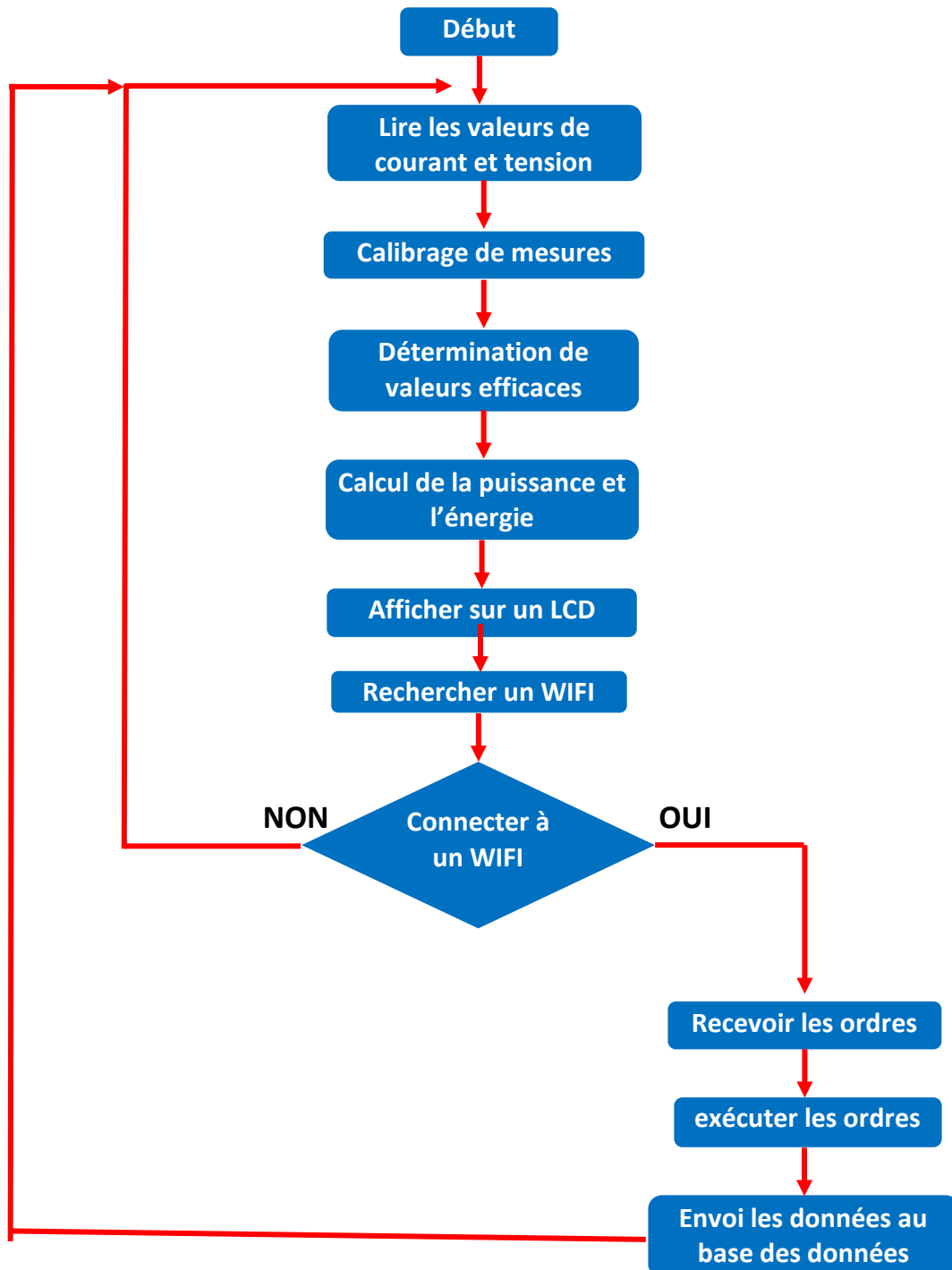


Figure III.2 : Organigramme du fonctionnement de notre système

### III.3 Calcul de l'énergie consommée

L'ESP32 utilise une bibliothèque (Library) Emonlib qui lit les données des capteurs de tension et de courant et calcule les valeurs suivantes :

- La tension efficace.
- La puissance active.
- Le courant efficace.
- La puissance réactive.
- Le facteur de puissance.
- La puissance apparente.

Et pour calculer l'énergie consommée :

$$E = \frac{P_a \cdot t}{3600000000}$$

$E$  : L'énergie consommée (KWh).

$P_a$  : La puissance active (W).

$t$  : Le temps (ms).

### III.4 Le Protocole de communication

#### III.4.1 Description du Protocole de communication

Le compteur intelligent d'énergie électrique aide les clients à suivre leur consommation de n'importe où, plus sa capacité à commander certains équipements à la maison. C'est pour ça que nous avons créé le Protocole de communication suivant :

- Le ESP32 qui est connecté avec un WIFI envoie les données de consommation via internet à la base des données (Firebase).
- L'application lit les données de la base des données et les affiche sur l'onglet data de l'interface utilisateur.
- Si nous avons entré des commandes pour les équipements reliés au compteur, l'application enregistre des caractères spécifiques dans la base des données.
- L'ESP vérifie d'une manière périodique si de nouveaux caractères spéciaux ont été enregistrés dans la base des données. Dépendant du caractère lu, l'ESP va commander (on/off) l'équipement concerné.



Figure III.3 : Le Protocole de communication

### III.4.2 Conception et réalisation de l'application de smartphone

L'Android ou iOS sont des systèmes d'exploitation pour appareils mobiles. Ces systèmes présentent plusieurs fonctionnalités et avantages. Parmi eux, on peut citer :

- La facilité d'accès à l'information.
- La facilité et la simplicité de développement.
- La rapidité d'exécution.

Pour ces raisons, on a intégré l'utilisation des applications de smartphone dans le compteur d'énergie. Cette intégration permet de rendre le compteur d'énergie communicant. En effet, le compteur d'énergie devient intelligent lorsqu'on assure une communication informatique entre le compteur d'énergie standard et le client.

Cette communication peut se faire à l'aide d'une application Android ou iOS. Le compteur envoie des données (des informations sur la consommation du client à la base des données virtuelle "Firebase"). Alors que l'application à l'accès de l'entrée à cette "Firebase" et de lire les nouvelles données.

L'interface principale de notre application est présentée par la figure III.5 suivante.

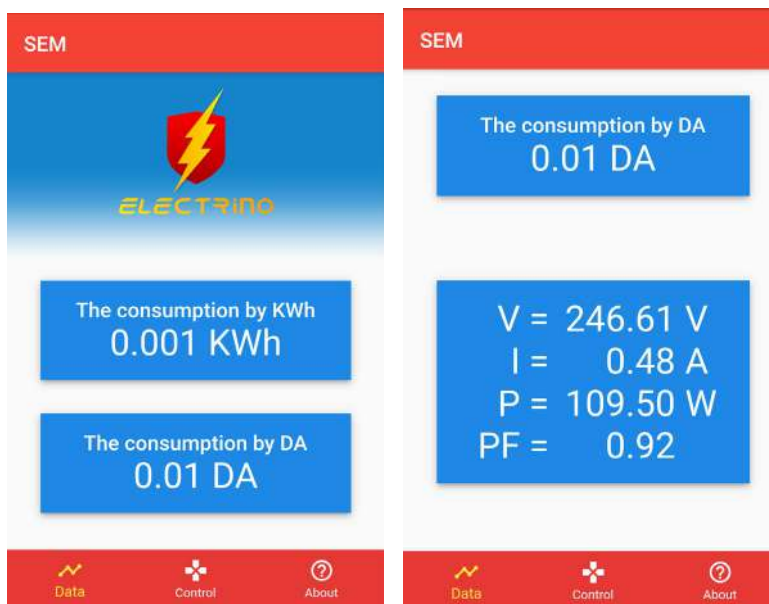
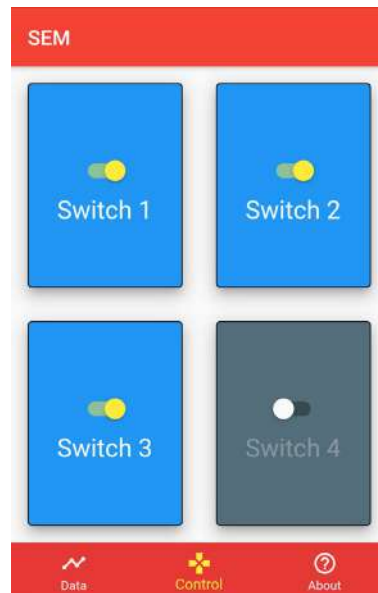


Figure III.4 : Interface principale de l'application (Data page)

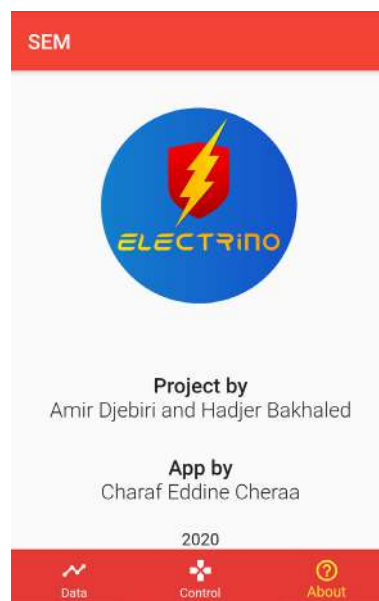
Dans cet interface (Data page) l'application affiche les données de la consommation avec le prix calculé.

La deuxième interface (Control page) permet de commander les relais qui sont reliés au compteur intelligent. (Figure III.6)



**Figure III.5 : La deuxième interface de l'application (Control page)**

La troisième interface (About page) contient des informations sur l'application.



**Figure III.6 : La troisième interface de l'application (About page)**

Pour créer une application avec Flutter sur Visual Studio Code, les deux étapes suivantes sont nécessaires :

- La première étape c'est l'installation de software du Flutter sur le PC, mais cette étape est vraiment compliquée donc il faut suivre le site officiel « [WWW.flutter.dev](http://WWW.flutter.dev) » avec tous les détails.

- La deuxième étape c'est l'installation de Visual Studio Code, mais après l'installation il faut télécharger deux extensions dans ce programme pour travailler avec flutter qui sont « Flutter et Dart ».

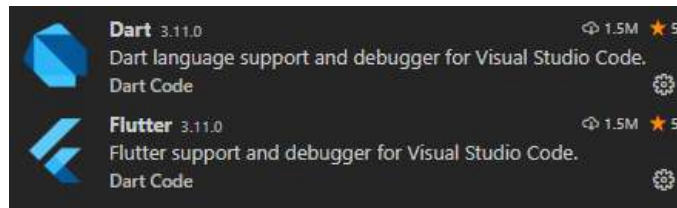


Figure III.7 : Les extensions utilisés dans Visual Studio Code

### III.4.3 La base des données Firebase

Dans ce projet, on a utilisé une base de données virtuelle : "Firebase". Firebase est une plateforme mobile qui permet de développer une base d'utilisateurs ainsi que des applications de haute qualité. Elle est constituée de fonctionnalités complémentaires. L'intégration de Firebase dans l'application s'effectue à l'aide de Flutter dans Visual Studio Code.

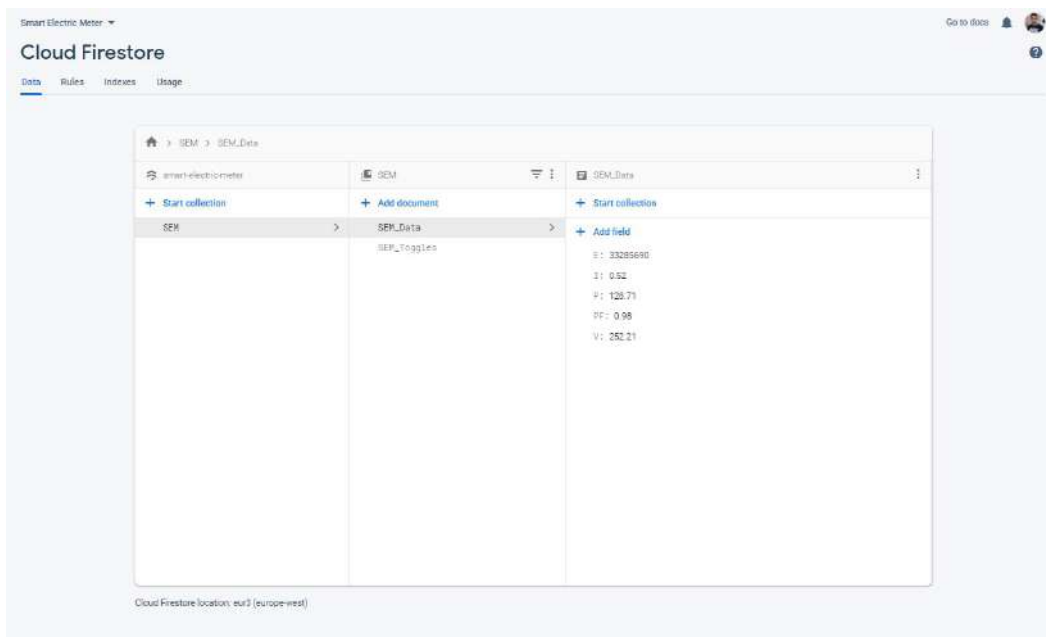


Figure III.8 : L'interface de la base des données Firebase

## III.5 La réalisation et les résultats pratiques du compteur intelligent d'énergie

### III.5.1 Le schéma global du compteur intelligent d'énergie

La figure III.9 suivante présente le circuit électronique de compteur ainsi que le câblage électrique.

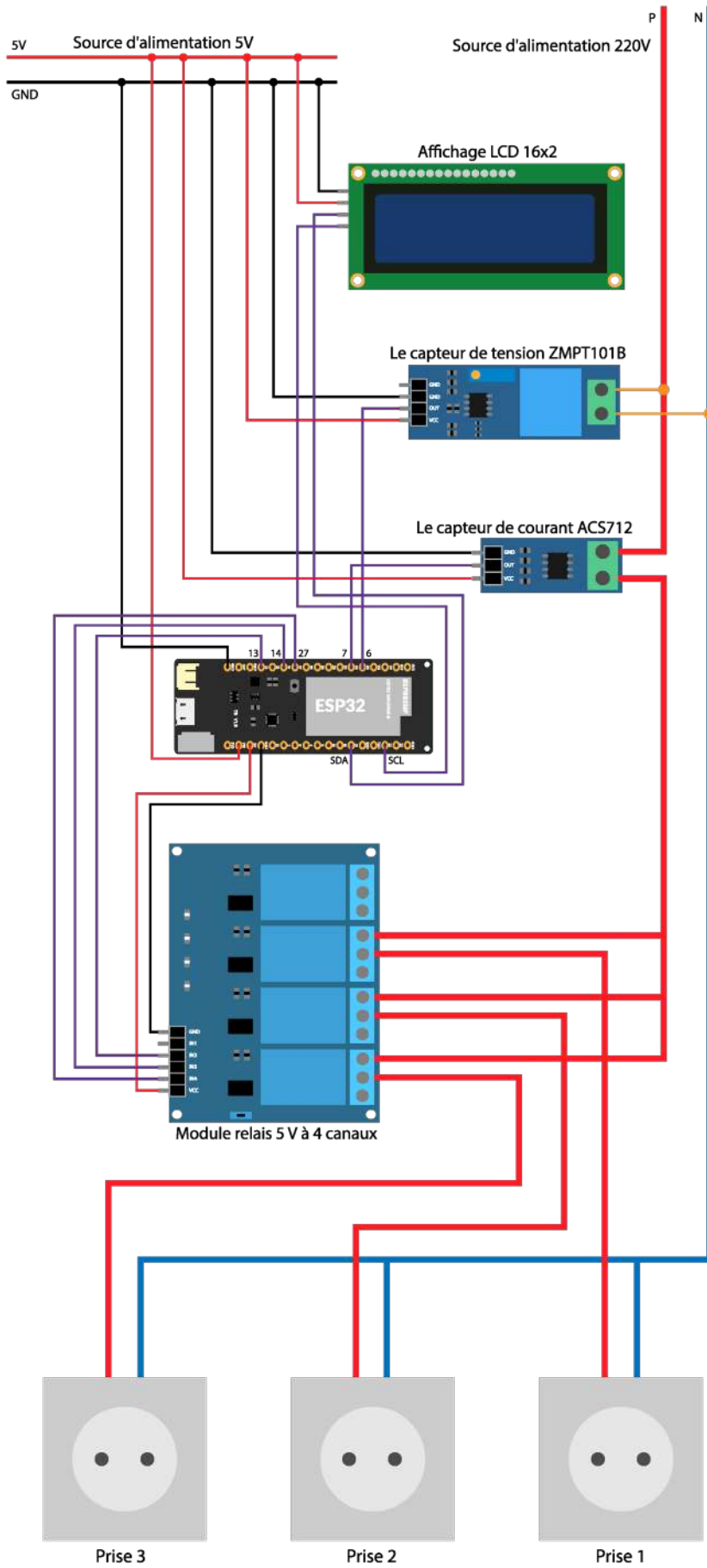


Figure III.9 : Le schéma global du compteur intelligent d'énergie

### III.5.2 Réalisation pratique

Les figures III.10 et III.11 illustrent la réalisation pratique du compteur intelligent d'énergie proposé.



Figure III.10 : Réalisation pratique d'un compteur intelligent de d'énergie

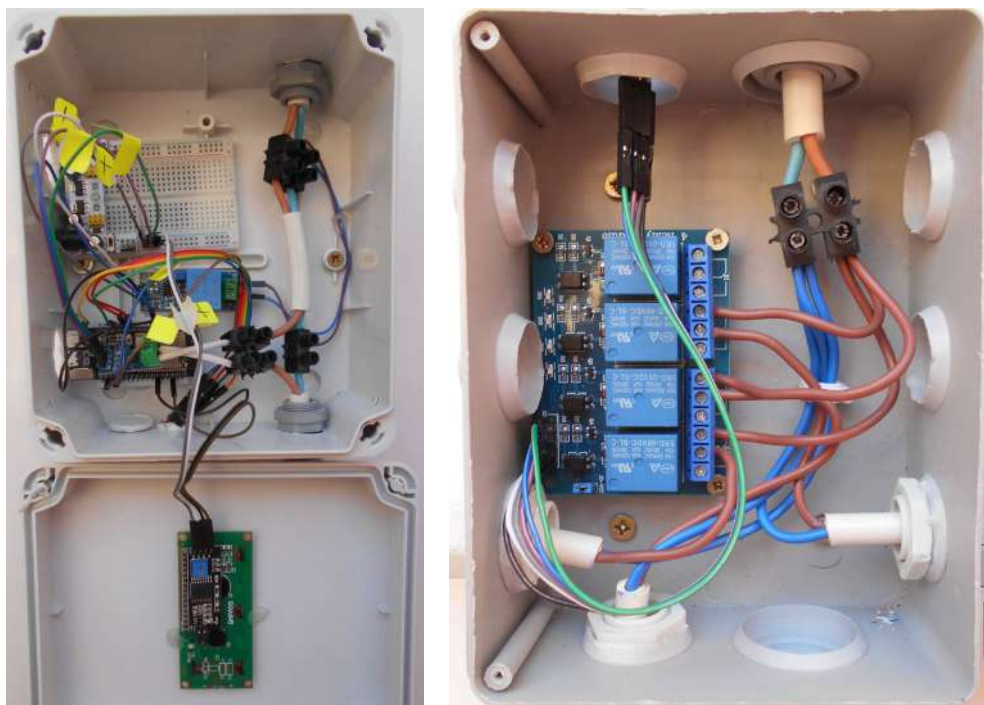


Figure III.11 : Structure du compteur intelligent d'énergie

### III.5.3 les résultats pratiques

Les figures suivantes illustrent les résultats pratiques du compteur intelligent d'énergie proposé, pour une charge résistive de type sèche-linge électrique.

La figure III.12 représente le résultat de la valeur du courant correcte.

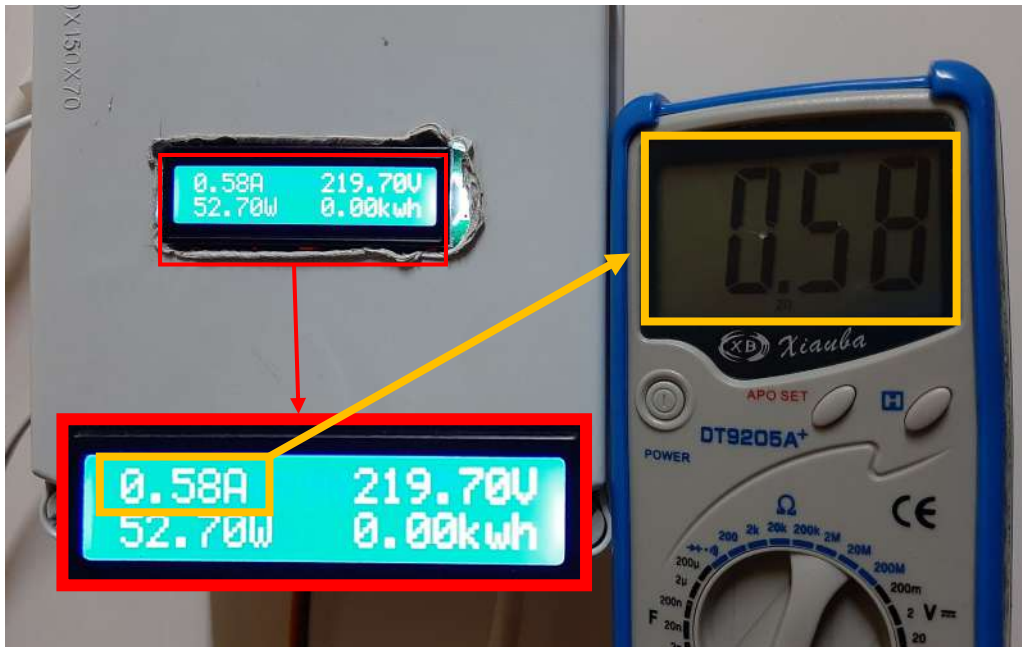


Figure III.12 : Mesure de courant

La figure III.13 représente le résultat de la valeur de tension correcte.

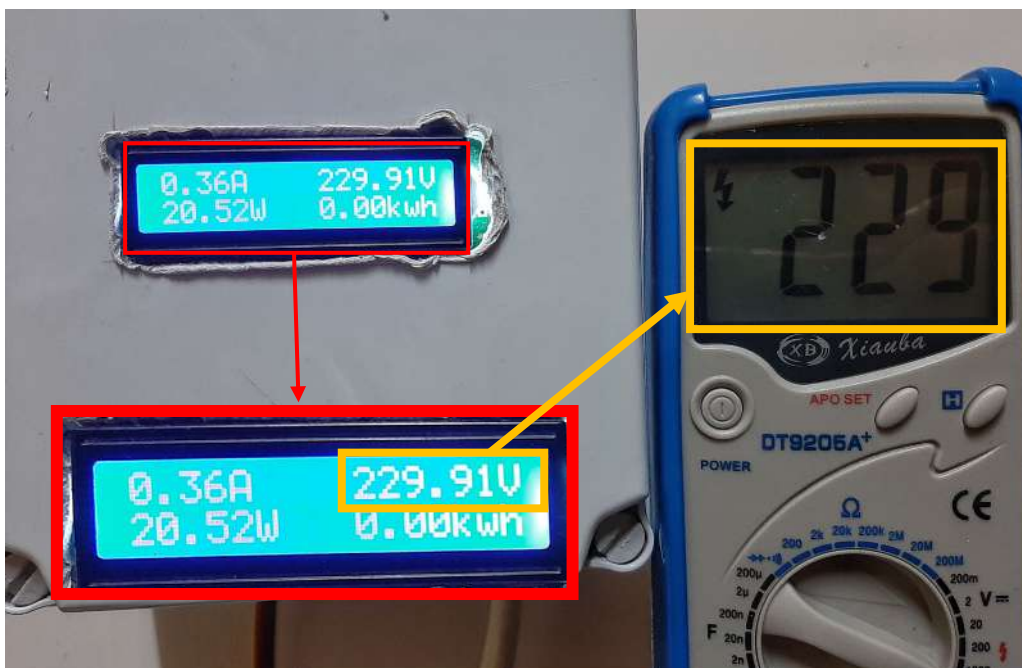


Figure III.13 : Mesure de tension



La figure III.14 représente le résultat du transfert des données avec l'application de smartphone.



Figure III.14 : Le résultat du transfert des données avec l'application de smartphone.

La figure III.15 représente l'affichage des mêmes données dans la base des données Firebase.

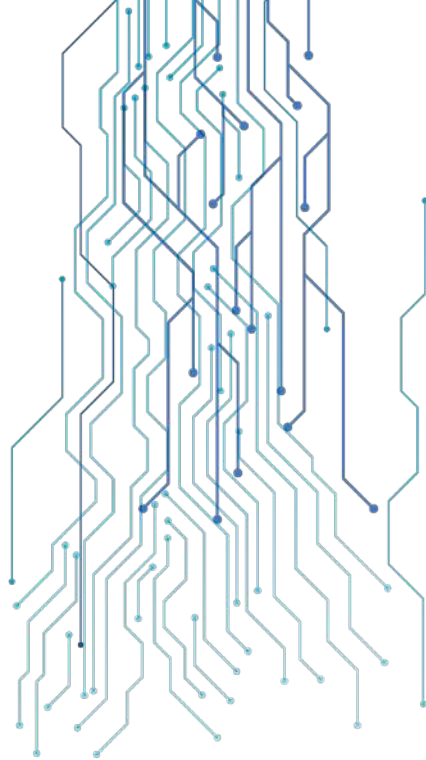


Figure III.15 : L'affichage des mêmes données dans la base des données Firebase

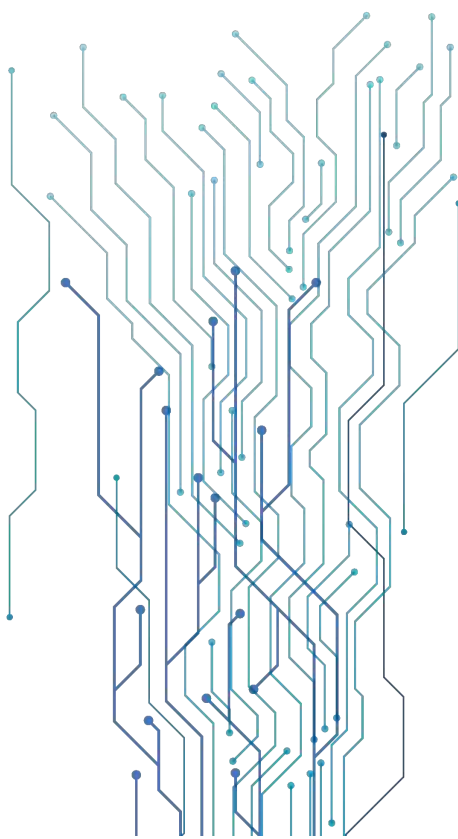


### III.6 Conclusion

Dans ce dernier chapitre, on a présenté l'architecture du circuit de mesure et communication utilisé pour rendre le compteur d'énergie intelligent. Avec l'architecture de la plateforme "Firebase" et en utilisant "Flutter", on a réalisé une application Android sous smartphone capable de synchroniser les données avec la carte ESP32 afin d'exécuter des ordres bien définis. La réalisation pratique est aussi présentée à la fin du chapitre ainsi que des tests qui prouvent le bon fonctionnement du compteur intelligent d'énergie.



# Conclusion générale





L'utilisation des outils de communication permet d'améliorer les systèmes de comptage d'énergie classique.

Dans ce cadre, ce projet de fin d'études a été proposé pour aider le consommateur de contrôler et vérifier sa consommation d'énergie électrique. On a étudié et réalisé un compteur intelligent d'énergie à base de la carte ESP32, d'une application Android et le système de IoT (Internet of Things).

Dans le premier chapitre, on a présenté les différents types des compteurs d'énergie, la conception, le principe de fonctionnement, et les avantages et les inconvénients de chaque type de compteurs.

Dans le deuxième chapitre, on a présenté les outils du notre projet que ce soit software ou hardware.

Le dernier chapitre a été consacré à la réalisation du compteur d'énergie intelligent par recours aux outils de communication Wi-Fi et smartphone.

Les résultats pratiques obtenus sont très satisfaisants montrant la communication directe entre le compteur d'énergie et le client.

Comme perspectives à ce travail, on peut citer :

- L'extension du compteur d'énergie communiquant pour les systèmes triphasés.
- L'enregistrement des données de consommation dans la base des données et dans un carte mémoire en cas l'absence de l'internet.
- Ajouter un système de protection intelligent en cas des problèmes électriques.

## Bibliographie

[1] <https://www.futura-sciences.com/maison/definitions/maison-compteur-electricite-10598/> .

[2] Dr. Mohamed ZELLAGUI. (Février 2018). Présentation de Comptage et Compteurs Électrique. Institut de Formation de l'Electricité et du Gaz Centre de Formation Ain M'Lila.

[3] Lamia HASNAOUI et Salma GABBADI. (Juin 2017). Etude des compteurs d'énergie électrique. Université Sidi Mohamed Ben Abdellah. Maroc.

[4] <https://www.expertise-energie.fr/compteurs/tout-savoir-sur-le-compteur-electromecanique/> .

[5] [www.espressif.com](http://www.espressif.com) .

[6] [www.esp32.net](http://www.esp32.net) .

[7] Espressif Systems. 2019. ESP32-Wrover-B Datasheet Version 1.3.

[8] MBARKI Imen. (Juillet 2018). Etude et réalisation d'un compteur d'énergie intelligent. Université de Gabès. Tunisie.

[9] Ahmed J Ali, Ahmed M T Ibraheem and Omar Talal Mahmood. (Janvier 2020). Design of a Smart Control and Protection System for Three-Phase Generator Using Arduino. Dep. of Electrical Power Engineering Technologies, Technical Engineering College/ Mosul, Northern Technical University, Iraq.

[10] LAIB Bachir et SAADANI Fathi. 2018. Commande d'une Barrière Automatique à Base d'Arduino. Université Echahid Hamma Lakhdar d'El-Oued.

[11] Handson Technology. 4 Channel 5V Optical Isolated Relay Module.

## Annexe A

```
#include "EmonLib.h"
EnergyMonitor emon1;

#include "ACS712.h"
ACS712 currentSensor(ACS712_20A, A7);

#include "ZMPT101B.h"
ZMPT101B voltageSensor(A6);

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);

#include <ssl_client.h>
#include <WiFiClientSecure.h>
WiFiClientSecure client;
WiFiClient timeClient;
#include <WiFi.h>

bool setupWifi(char *ssid, char *password);
void checkToggles();
void emonlib();
bool postData(float v, float i, float pf, float p, float e);

float V;
float I;
float P;
float FP;
double E;
unsigned long t;
unsigned long y;

void log(String message)
{
    if (Serial)
    {
        Serial.print(message);
    }
}
void logln(String message)
{
    if (Serial)
    {
        Serial.println(message);
    }
}
```



```
//=====
void setup()
{
  analogReadResolution(10);
  Serial.begin(115200);

  pinMode(13, OUTPUT);
  pinMode(14, OUTPUT);
  pinMode(27, OUTPUT);

  voltageSensor.calibrate();
  currentSensor.calibrate();

  emon1.voltage(A6, 230, 1.7);
  emon1.current(A7, 5);

  //lcd.begin(16,2);
  lcd.init();
  lcd.backlight();
  lcd.clear();

  if (setupWifi("My Wifi", "123456789"))
  {
    logln("done2");
  }
  else if (setupWifi("DC-WC", "wifipass"))
  {
    logln("done");
  }
  else if (setupWifi("M5", "123456789"))
  {
    logln("done1");
  }

  else if (setupWifi("DC-WC", "wifipass"))
  {
    logln("done3");
  }
  t = millis();
  E = 0;
}
```



```
//=====
void loop()
{
    checkToggles();
    emonlib();
    checkToggles();
    emonlib();
    checkToggles();
    emonlib();
    checkToggles();
    emonlib();

    if (!postData(V, I, FP, P, E))
    {
        logln("error!");
    }
}

bool setupWifi(char *ssid, char *password)
{
    WiFi.begin(ssid, password);
    int w = millis();
    while (WiFi.status() != WL_CONNECTED)
    {
        Serial.println(WiFi.status());
        delay(500);
        log(".");
        if (millis() > w + 10000)
        {
            logln("wifi failed");
            return false;
        }
    }
    log("WiFi connected");
    log("IP address: ");
    lcd.print("connected");
    logln(String(WiFi.localIP()));
    if (!client.connect("firestore.googleapis.com", 443))
    {
        logln("connection failed");
    }
    if (!timeClient.connect("worldtimeapi.org", 80))
    {
        logln("connection failed");
    }
    return true;
}
```





```
bool postData(float v, float i, float pf, float p, float e)
{
    bool res = false;
    String data = String("{\"fields\":{\"PF\":{\"doubleValue\":\")} + S
    String request = String("PATCH /v1/projects/smart-electric-meter/da
        "Host: firestore.googleapis.com\r\n" +
        "Content-Type: application/json\r\n" +
        "cache-control: no-cache\r\n" +
        "Connection: keep-alive\r\n" +
        "Content-Length: " + data.length() + "\r\n\r\n" +
        data + "\r\n\r\n";

    client.print(request);
    logln(request);
    unsigned long timeout = millis();
    while (client.available() == 0)
    {
        if (millis() - timeout > 5000)
        {
            logln(">>> Client Timeout ! (post data)");
            client.stop();
            return false;
        }
    }

    if (client.available())
    {
        String line = client.readStringUntil('\r').substring(9, 12);
        res = line == "200";
        logln(line);
    }
    while (client.available())
    {
        String line = client.readStringUntil('\r');
        logln(line);
    }

    logln("");
    logln("closing connection");
    return res;
}

String id()
{
    String request = String("GET /api/ip.txt HTTP/1.1\r\n") +
        "Host: worldtimeapi.org\r\n" +
        "Connection: keep-alive\r\n\r\n";

    timeClient.print(request);
    logln(request);
    unsigned long timeout = millis();
```



```
while (timeClient.available() == 0)
{
    if (millis() - timeout > 5000)
    {
        logln(">>> Client Timeout ! (id)");
        timeClient.stop();
        return "";
    }
}
while (timeClient.available())
{
    String line = timeClient.readStringUntil('\r\n');
    Serial.print(" ---- ");
    Serial.println(line);
    if (line.startsWith("unixtime: "))
    {
        return line.substring(10);
    }
}
Serial.print(" ----- ");
return "";
}

void emonlib()
{
    emon1.calcVI(20, 1000);
    emon1.serialprint();
    V = emon1.Vrms;
    I = (emon1.calcIrms(1480)-0.205)*2.4;
    FP = emon1.powerFactor;
    P = emon1.realPower;
    y = millis();
    E = E + (P) * ((y - t));
    t = y;
    lcd.clear();
    lcd.print(I);
    lcd.print("A");
    lcd.print(" ");
    lcd.print(V);
    lcd.print("V");
    lcd.setCursor(0, 1);
    lcd.print(P);
    lcd.print("W");
    lcd.print(" ");
    lcd.print(E / 3600000000);
    lcd.print("kwh");
}
```



```
void checkToggles()
{
    String res = "";
    String request = String("GET /v1/projects/smart-electric-meter/da
        Host: firestore.googleapis.com\r\n\r\n");
    client.print(request);
    unsigned long timeout = millis();
    while (client.available() == 0)
    {
        if (millis() - timeout > 5000)
        {
            logln(">>> Client Timeout ! (toggles)");
            client.stop();
            return;
        }
    }
    bool status = false;
    if (client.available())
    {
        String line = client.readStringUntil('\r').substring(9, 12);
        status = line == "200";
    }
    while (client.available())
    {
        String line = client.readStringUntil('\r');
        int pos = line.indexOf("stringValue");
        if (pos > -1) {
            res = line.substring(pos + 15, pos + 19);
        }
    }

    if (status) {
        digitalWrite(13, res.charAt(0) == '1' ? 1 : 0);
        digitalWrite(14, res.charAt(1) == '1' ? 1 : 0);
        digitalWrite(27, res.charAt(2) == '1' ? 1 : 0);
    }
}
```

Le code de ESP32