وزارة البحث العلمي والتعليم العالي
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET
DE LA RECHERCHE SCIENTIFIQUE
جامعة عبد الحميد بن باديس مستغانم
Université Abdelhamid Ibn Badis Mostaganem
كلية العلوم و التكنولوجيا
Faculté des Sciences et de la Technologie
DEPARTEMENT DE GENIE ELECTRIQUE

MEMOIRE

Mémoire de fin d'études en vue d'obtention du diplôme De Master

Spécialité : Électronique des Systèmes Embarqués

Thème :

Intitule du subjet

Conception et réalisation d'un système IoT de surveillance de la température basé sur ESP32

Présenté par : *Lakehal Abderrahmane*
Damen Abdelmalek

Devant le jury composé de :

Président : *M. BENAOUALI Mohamed*	*MAA*	*Université de Mostaganem.*
Examinateur : *M. SOLTANE BENALOU A.*	*MAA*	*Université de Mostaganem*
Encadreur: *M. BENTOUMI Mohamed*	*MCA*	*Université de Mostaganem*
Co-encadreur: *M. BENOUALI Abdelhak*	*MAA*	*Université de Mostaganem*

Soutenu le : 13/10/2025

REMERCIEMENTS

*Nous tenons avant tout à remercier **Allah** le Tout-Puissant, source de force, de persévérance et de patience, pour nous avoir permis d'atteindre ce moment avec foi et détermination.*

*Nos remerciements vont ensuite à Monsieur **BENTOUMI Mohamed**, notre encadreur, pour sa disponibilité, sa bienveillance et ses conseils toujours pertinents. Son accompagnement constant, alliant rigueur scientifique et écoute humaine, a donné à ce travail toute sa consistance.*

*Nous souhaitons également exprimer notre profonde gratitude à Monsieur **BENOUALI Abdelhak, co-encadreur**, pour son suivi attentif, ses orientations précieuses et son appui technique tout au long de la réalisation de ce projet.*

*Nous remercions sincèrement Monsieur **BENAOUALI Mohamed**, président du jury, ainsi que Monsieur **SOLTANE BENALOU Abdelkader,** examinateur, pour l'attention qu'ils ont accordée à notre travail et pour leurs remarques constructives qui ont contribué à l'enrichir.*

*Nos remerciements s'adressent aussi à l'ensemble du corps enseignant et administratif de la Faculté des Sciences et Technologies de l'Université de Mostaganem, en particulier au **Département de Génie Électrique**, pour la qualité des enseignements reçus et l'environnement académique stimulant qui a façonné nos compétences.*

Nos remerciements vont également aux laboratoires et services techniques qui ont facilité la réalisation pratique de ce projet, en partageant leurs ressources et leurs conseils.

Enfin, une pensée empreinte d'amour et de reconnaissance à nos familles, piliers de notre réussite, pour leur patience, leurs sacrifices silencieux et leur soutien constant.

*Et à nous-mêmes, **Lakehal Abderrahmane** et **Damen Abdelmalek** , pour chaque effort, chaque persévérance et chaque moment de doute surmonté. Ce mémoire est le reflet de notre engagement, de notre passion et de notre détermination à aller jusqu'au bout.*

À toutes et à tous, trouvez dans ces mots l'expression sincère de notre reconnaissance infinie.

DÉDICACE

Je dédie ce travail, fruit de plusieurs mois d'efforts, de persévérance et de passion,

à toutes les personnes qui ont cru en moi et m'ont soutenu tout au long de ce parcours.

À mes chers parents, pour leur amour inconditionnel, leurs prières sincères et leur confiance sans limite. Vous êtes ma plus grande source de force et de motivation.

À ma famille, pour leur soutien, leur compréhension et leur présence constante dans les moments de doute comme dans les moments de réussite.

À mes chers amis, pour leur amitié sincère, leurs encouragements et leur bienveillance qui ont rendu ce chemin plus léger et plus agréable.

À, ***Damen Abdelmalek*** *pour son soutien, ses conseils et sa présence qui ont profondément marqué cette période de ma vie.*

Enfin, à toutes celles et ceux qui, de près ou de loin, ont contribué à la réalisation de ce mémoire,

je vous adresse ma gratitude la plus sincère.

Abderrahmane LAKEHAL

DÉDICACE

Je dédie ce travail, fruit de plusieurs mois d'efforts, de persévérance et de passion,

à toutes les personnes qui ont cru en moi et m'ont soutenu tout au long de ce parcours.

À mes chers parents, pour leur amour inconditionnel, leurs prières sincères et leur confiance sans limite. Vous êtes ma plus grande source de force et de motivation.

À ma famille, pour leur soutien, leur compréhension et leur présence constante dans les moments de doute comme dans les moments de réussite.

À mes chers amis, pour leur amitié sincère, leurs encouragements et leur bienveillance qui ont rendu ce chemin plus léger et plus agréable.

À, ***Abderrahmane LAKEHAL*** *pour son soutien, ses conseils et sa présence qui ont profondément marqué cette période de ma vie.*

Enfin, à toutes celles et ceux qui, de près ou de loin, ont contribué à la réalisation de ce mémoire,

je vous adresse ma gratitude la plus sincère.

Damen Abdelmalek

TABLE DES MATIERES

REMERCIEMENTS.....2
DÉDICACE.....3
DÉDICACE.....4
TABLE DES MATIERES.....5
LISTE DES ABREVIATION.....9
LISTE DES FIGURES.....10
LISTE DES FIGURES.....11
LISTE DES TABLEAUX.....12
INTRODUCTION GENERALE.....13

Introduction 14
1. Problématique.....14
2. Les hypothèses.....18
3. Objectifs de recherche.....18
4. Méthodologie.....19
5. Choix du thème de mémoire.....20
6. Structure et présentation des chapitres.....20
Conclusion générale

CHAPITRE I : l'internet des objets et les systèmes embarques – présentation, Intégration et perspectives futures.....17
Introduction.....18
1. Définitions et Concepts.....18
2. Architecture de l'IoT et des Systèmes Embarqués.....19
3. Les Technologies Fondamentales qui Soutiennent l'IoT et les Système Embarqués 24
4. Domaines d'Applications Étendus…….....……32
5. Défis Majeurs de l'Internet des Objets et des Systèmes Embarqués…….....……39
6. Orientations Futures…….....…..42
Conclusion…….....…….46

CHAPITRE II : ÉTUDE MATERIELLE DU SYSTEME…….....……..47
1. Introduction…….....……..48
2. Description détaillée des composants électroniques…….....…….48
2.1 Microcontrôleur ESP32…….....……48
2.2 Capteur de température (D18B20S)…….....……51
2.3 Afficheur OLED (SSD1306, I2C)…….....…….53
2.4 Le Boutons poussoirs…….....……54
4. Conclusion…….....…58
CHAPITRE III : conception, réalisation et validation du système….....59
Introduction …….....……60
1. Méthodologie de travail et montage du système…….....……60
2. Programmation et description logicielle…….....……62
3. Résultats et interprétation…….....……65
Conclusion …….....……66
Conclusion Générale…….....……67

Résume

Ce mémoire porte sur le concept de l'Internet des Objets (IoT) et des systèmes embarqués, mettant en évidence leur rôle essentiel dans la collecte et le traitement en temps réel des données via des réseaux connectés. Il présente les définitions clés, les architectures en couches de l'IoT, ainsi que les technologies fondamentales telles que les capteurs, les microcontrôleurs et les protocoles de communication. Les applications dans des domaines variés tels que les villes intelligentes, l'industrie 4.0, la santé, l'agriculture intelligente et les véhicules connectés sont abordées, tout en soulignant les défis majeurs liés à la sécurité, la consommation énergétique, la scalabilité, l'interopérabilité et la réglementation. Enfin, le mémoire explore les perspectives d'avenir impliquant l'intelligence artificielle, le edge computing et les réseaux avancés.

Mots-clés : Internet des Objets, Systèmes Embarqués, informatique en périphérie, Sécurité et Confidentialité, Intelligence Artificielle

Abstract:

This thesis focuses on the concept of the Internet of Things (IoT) and embedded systems, highlighting their critical role in real-time data collection and processing through connected networks. It covers key definitions, the layered architectures of IoT, and fundamental technologies such as sensors, microcontrollers, and communication protocols. The document discusses applications across diverse fields such as smart cities, Industry 4.0, healthcare, smart agriculture, and connected vehicles, emphasizing major challenges including security, energy consumption, scalability, interoperability, and regulation. The thesis also anticipates future developments involving artificial intelligence integration, edge computing, and advanced communication networks.

Keywords: Internet of Things, Embedded Systems, Edge Computing, Security and Privacy, Artificial Intelligence

الملخص

تركز هذه المذكرة على مفهوم الإنترنت للأشياء وأنظمة الحوسبة المدمجة، موضحةً كيف يشكلان ثنائياً تكنولوجياً فريداً يسهل مراقبة البيانات وتحليلها في الوقت الفعلي عبر اتصال الشبكات. تستعرض المذكرة التعريفات الأساسية، البنية الطبقية لنظم الإنترنت للأشياء، والتقنيات الداعمة مثل الحساسات والمتحكمات والبروتوكولات المختلفة. كما تتناول تطبيقاتها في مجالات متنوعة مثل المدن الذكية، الصناعة 4.0، الرعاية الصحية، الزراعة الذكية، والمركبات المتصلة، مع إبراز التحديات المتعلقة بالأمان الطاقة، التوسع، التوافقية واللوائح. كما تستشرف المذكرة آفاق التطور المستقبلي مع ولوج تقنيات الذكاء الاصطناعي، الحوسبة الطرفية، وتقنيات الاتصالات المتقدمة.

الكلمات المفتاحية الإنترنت للأشياء، الأنظمة المدمجة، الحوسبة الطرفية، الأمان والخصوصية، الذكاء الاصطناعي

LISTE DES ABREVIATION

Sigle	Signification	Traduction en arabe
IoT	Internet of Things	الإنترنت للأشياء
IdO	Internet des Objets	انترنت الاشياء
M2M	Machine to Machine	الاتصال بين الأجهزة
IIoT	Industrial Internet of Things	الإنترنت الصناعي للأشياء
LTE	Long-Term Evolution	تطور طويل الأمد – تقنية اتصالات
SSO	Single Sign-On	تسجيل دخول موحد
RFID	Radio Frequency Identification	التعرف بترددات الراديو
API	Application Programming Interface / Interface de Programmation d'Applications	واجهة برمجة التطبيقات
MQTT	Message Queuing Telemetry Transport	بروتوكول نقل رسائل لأجهزة إنترنت الأشياء
LAN	Local Area Network	الشبكة المحلية
AI	Artificial Intelligence	الذكاء الاصطناعي
USB	Universal Serial Bus	ناقل تسلسلي عام
GPS	Global Positioning System	نظام تحديد المواقع العالمي
Wi-Fi	Wireless Fidelity	الشبكات اللاسلكية
BLE	Bluetooth Low Energy	بلوتوث منخفض الطاقة
HTTP	HyperText Transfer Protocol	بروتوكول نقل النص الفائق
HTTPS	HyperText Transfer Protocol Secure	بروتوكول نقل النص الفائق الآمن

LISTE DES FIGURES

Figure 1: Schéma illustrant à trois couches (Three-Layer Architecture)……..…20

Figure 2 : Schéma illustrant l'architecture à cinq couches (Five-Layer Architecture.....21

Figure 3 : Schéma illustrant l'intégration des systèmes embarqués dans ces couches……23

Figure 4: Schéma fonctionnel simplifié du système de mesure…….……24

Figure 5: Architecture générale du système à base d'ESP32……26

Figure 6: Classification des technologies de communication sans fil selon la portée et le débit de données (Data Rate)……..27

Figure 7 : Diagramme comparatif entre le Cloud Computing et l'Edge Computing……29

Figure 8: Architecture fonctionnelle et composants de l'Intelligence Artificielle (IA) et du Machine Learning…….……30

Figure 9 : Illustration d'une usine intelligente (Industrie 4.0) avec robots, machines connectées et écrans de données en temps réel…….……34

Figure 10: Illustration de l'intégration des Drones et de l'irrigation intelligente dans l'Agriculture de Précision…….……36

Figure 11: application LoT dans le commerce de détail et la logistique….……38

Figure 12: Schéma de brochage typique de l'ESP32 (ESP-WROOM-32 Pinout…….50

Figure 13: Capteur de température numérique étanche (sonde) de type DS18B207……..……51

Figure 14 : "Figure 2.2 – Connexion DS18B20 à l'ESP32"……...…….52

Figure 15 : Module d'affichage OLED ET brochage de connexion…….……53

Figure 16: Structure interne et principe de connexion des bornes d'un interrupteur tactile (Tactile Switch)……54

Figure 17:schéma Fonctionnel du système IoT ……57

Figure 18: Schéma de câblage et d'interfaçage de l'ESP32 avec le capteur de température DS18B20 et l'afficheur OLED…….……79

LISTE DES FIGURE

Figure 1 : Assemblage complet du système de surveillance de température (Partie matérielle)……..62

Figure 2: Interfaces d'affichage locales sur l'écran OLED……..64

Figure 3: Vues des données sur la page d'accueil du serveur Web……65

LISTE DES TABLEAUX

Tableau 1: Comparaison proposée entre les modèles d'architecture de l'IoT (3 couches vs 5 couches)……..…….22

Tableau 2: Résumé des technologies fondamentales de l'Internet des Objets et des systèmes embarqués avec exemples d'utilisation…….……..31

Tableau 3: Exemples d'applications de l'IoT et des systèmes embarqués dans différents secteurs avec principaux avantages……38

Tableau 4: Caractéristiques techniques principales de l'ESP32-WROOM-32……..…….49

Tableau 5: Caractéristiques techniques de l'afficheur OLED (SSD1306)……...54

Tableau 6: Schéma de câblage et affectation des broches de l'ESP3……61

INTRODUCTION GENERAL

Introduction générale

Au cours des dernières décennies, le monde a connu un développement sans précédent dans le domaine des technologies de l'information et de la communication, ce qui a entraîné une transformation qualitative des modes de gestion, de contrôle et de supervision dans divers secteurs vitaux. Au cœur de cette transformation, le concept de l'Internet des Objets (IoT – Internet of Things) s'est imposé comme l'un des piliers fondamentaux de la construction des villes intelligentes et du développement de solutions innovantes dans plusieurs domaines, allant de l'industrie, la santé, les transports, l'agriculture et l'environnement, jusqu'à la vie quotidienne dans les foyers.

L'Internet des Objets repose sur le principe de la connexion des objets physiques – tels que les appareils, capteurs et équipements – aux réseaux de communication numériques, permettant la collecte, le traitement et l'analyse de grandes quantités de données à distance et en temps réel. Cette intégration entre le monde matériel et le monde numérique a permis une amélioration notable de la qualité des services, une augmentation de l'efficacité opérationnelle et une réduction des coûts, faisant de cette technologie une pierre angulaire des stratégies de transformation numérique et de transition vers une économie intelligente et verte.

Parmi les applications pratiques les plus importantes de l'Internet des Objets figurent les systèmes de mesure et de surveillance à distance, en particulier la surveillance de la température, considérée comme une variable physique vitale et essentielle dans de nombreux domaines. Les industries agroalimentaire et pharmaceutique, les systèmes de réfrigération et de stockage, ainsi que la surveillance environnementale et les applications domestiques intelligentes, dépendent toutes d'une mesure et d'un contrôle précis de la température. La capacité de transférer ces données vers des plateformes cloud ou des applications mobiles en temps réel ouvre la voie au développement de systèmes prédictifs d'alerte précoce, contribuant ainsi à réduire les risques et à garantir la qualité et la sécurité des opérations.

1. Problématique

Malgré les vastes opportunités offertes par l'Internet des Objets, le développement d'un système efficace de surveillance de la température à distance reste confronté à un ensemble de défis. Sur le plan matériel, se pose la question du choix des composants électroniques adaptés, tels que les microcontrôleurs, capteurs et modules de communication. Sur le plan technique, le défi majeur reste d'assurer la précision des mesures et la stabilité de la transmission des données via les réseaux sans fil. Sur le plan logiciel, la difficulté réside dans le développement d'une interface intuitive permettant à l'utilisateur de lire et d'analyser facilement les données. À cela s'ajoute le défi applicatif consistant à rendre le système adaptable à divers scénarios incluant les contextes industriels, environnementaux et domestiques.

À partir de ces constats, la problématique centrale de cette recherche peut être formulée comme suit :

Comment concevoir et réaliser un système intelligent basé sur l'Internet des Objets, permettant de mesurer et de surveiller la température à distance et en temps réel, tout en garantissant précision, fiabilité, facilité d'utilisation et évolutivité ?

Afin de traiter cette problématique, celle-ci peut être déclinée en questions suivantes :

- Quels sont les critères techniques pour le choix du microcontrôleur et des capteurs adaptés à ce système ?

- Comment parvenir à un équilibre entre la précision de la mesure et la fiabilité de la transmission sans fil des données ?
- Quelles sont les plateformes et logiciels les plus appropriés pour l'affichage et l'analyse des données en temps réel ?
- Dans quelle mesure ce prototype peut-il être étendu à d'autres applications dans les domaines industriel et environnemental ?

2. Les hypothèses

Cette recherche repose sur un ensemble d'hypothèses, parmi lesquelles :

- L'adoption d'un microcontrôleur de type ESP32 permet d'intégrer efficacement la connectivité sans fil (Wi-Fi) et le traitement des données.
- L'utilisation d'un capteur de température de type One Wire peut garantir la précision et la fiabilité des mesures.
- Le développement d'une interface cloud ou d'une application mobile facilite l'accès de l'utilisateur aux données en temps réel et de manière dynamique.
- Ce système peut constituer un prototype extensible et évolutif, adaptable à divers besoins industriels et environnementaux.

3. Objectifs de recherche

Ce projet vise à atteindre un ensemble d'objectifs, dont les principaux sont :

- Concevoir et réaliser un dispositif basé sur l'Internet des Objets pour mesurer et surveiller la température à distance.
- Transmettre les données vers une plateforme cloud ou une application mobile pour les afficher et les analyser.
- Développer une interface intuitive permettant de consulter les mesures et de recevoir des alertes en cas de besoin.
- Tester le système en pratique afin de vérifier son efficacité et sa fiabilité.
- Proposer des pistes de développement futur permettant d'exploiter ce système dans divers domaines.

4. Méthodologie

Phase théorique : revue de la littérature scientifique relative à l'Internet des Objets et à ses applications dans les systèmes de surveillance.

Phase de conception : choix des composants électroniques appropriés (ESP32, capteur de température, modules de communication) et élaboration des schémas.

Phase de réalisation : programmation du microcontrôleur, intégration avec la plateforme cloud et développement d'une interface de visualisation.

Phase expérimentale : tests de précision des mesures, évaluation de la fiabilité de la transmission des données et analyse des résultats pour valider les hypothèses.

5. Choix du thème de mémoire

Le choix du thème intitulé « Conception et réalisation d'un système intelligent de mesure et de surveillance de la température à distance basé sur l'Internet des Objets (IoT) » s'inscrit dans le cadre du développement des technologies intelligentes appliquées à la supervision et au contrôle en temps réel.

Ce sujet a été retenu pour plusieurs raisons :

- Il répond aux enjeux actuels liés à la digitalisation et à l'automatisation des systèmes de suivi dans les secteurs industriel, environnemental et domestique.
- Il permet de mettre en œuvre concrètement les principes de l'Internet des Objets, à travers la conception d'un prototype fonctionnel combinant capteurs, microcontrôleurs et plateformes cloud.
- Il offre une dimension scientifique et pratique, conciliant théorie, modélisation et expérimentation.
- Enfin, il s'inscrit dans une dynamique d'innovation technologique visant la création de solutions durables, intelligentes et accessibles, contribuant ainsi à la transformation numérique du territoire.

6. Structure et présentation des chapitres

La structure du mémoire est conçue pour progresser du général au particulier, en passant de la théorie à l'application :

Introduction générale

Chapitre I : Ce chapitre met l'accent sur le cadre théorique et méthodologique, en présentant les concepts fondamentaux de l'Internet des objets, ses différentes applications, ainsi que les études antérieures pertinentes, tout en précisant la méthodologie scientifique adoptée.

Chapitre II : Ce chapitre traite des aspects techniques et conceptuels du système proposé, allant du choix des composants à l'élaboration des schémas et des logiciels nécessaires.

Chapitre III : Ce chapitre est consacré à l'application pratique, avec la présentation des étapes de réalisation du prototype, le test de ses performances et la comparaison de ses résultats avec des systèmes similaires, pour conclure par une conclusion générale comprenant les principaux résultats, recommandations et perspectives futures.

CHAPITRE I : L’internet des objets et les systèmes embarqués : concepts, fondements, applications, défis et orientations futures

Introduction

Le monde connaît aujourd'hui une transformation technologique sans précédent, portée par l'émergence de nouveaux concepts et technologies qui redéfinissent la manière dont nous interagissons avec notre environnement. Au cœur de cette transformation se trouvent deux technologies clés : l'Internet des Objets (IoT) et les Systèmes Embarqués. Ces deux technologies, qui peuvent sembler distinctes à première vue, s'intègrent en réalité de manière symbiotique pour permettre une nouvelle génération d'appareils intelligents et de systèmes automatisés promettant de révolutionner divers secteurs.

1. Définitions et Concepts

1.1. Définition de l'Internet des Objets (IoT) et son Importance Croissante

L'Internet des Objets (IoT) décrit un réseau d'objets physiques, ou "choses", qui sont intégrés à des capteurs, des logiciels et d'autres technologies dans le but de se connecter et d'échanger des données avec d'autres appareils et systèmes via Internet. Ces appareils vont des outils ménagers quotidiens comme les thermostats intelligents et les moniteurs pour bébés, aux équipements industriels complexes tels que les machines industrielles et les dispositifs de surveillance des patients.

L'importance fondamentale de l'IoT réside dans sa capacité à permettre une connexion transparente entre les personnes, les processus et les objets. Grâce à l'informatique à faible coût, aux plateformes cloud, à l'analyse de mégadonnées et aux technologies mobiles, les objets physiques peuvent désormais partager et collecter des données avec une intervention humaine minimale. Cela crée un environnement hyperconnecté où les systèmes numériques peuvent enregistrer, surveiller et ajuster chaque interaction entre les objets connectés. Ce concept va au-delà de la simple connexion d'appareils ; la véritable transformation réside dans la création d'un environnement "intelligent" intégré. L'objectif n'est plus seulement de collecter des données, mais de les utiliser pour permettre une communication fluide entre différentes entités, et pour contrôler, surveiller et gérer diverses fonctions sans nécessiter d'intervention humaine directe. Cette évolution indique que les appareils prennent des décisions autonomes et s'adaptent aux conditions changeantes, ce qui représente un changement qualitatif dans l'interaction entre l'homme et la technologie.

1.2. Définition des Systèmes Embarqués et leur Rôle Essentiel

Un système embarqué est un système informatique spécialisé basé sur un microprocesseur ou un microcontrôleur, composé de composants matériels et logiciels conçus pour exécuter des fonctions dédiées au sein d'un système mécanique ou électrique plus vaste. Ces systèmes sont intégrés dans les appareils pour contrôler, surveiller et gérer leurs diverses fonctions sans nécessiter d'intervention humaine directe.

Les systèmes embarqués varient en complexité, des systèmes extrêmement simples aux systèmes très complexes, et peuvent inclure un seul microcontrôleur ou un ensemble de processeurs connectés à des périphériques et des réseaux. Les instructions de programmation de ces systèmes sont généralement stockées dans des mémoires ROM (Read-Only Memory) ou des puces de mémoire flash. L'importance de la conception des systèmes embarqués réside dans leur large diffusion ; des milliards de dispositifs embarqués sont utilisés aujourd'hui dans de nombreuses industries, y compris les équipements médicaux et industriels, les systèmes de transport et les

équipements militaires, ainsi que les appareils de consommation courante tels que les montres numériques, les appareils de cuisine et les automobiles. Les systèmes embarqués sont appréciés pour leur petite taille, leur vitesse, leur puissance et leur conception pour des fonctions très spécifiques. Alors que les systèmes à usage général peuvent exécuter plusieurs fonctions, ils sont souvent trop coûteux pour de nombreuses applications et peuvent ne pas atteindre le niveau de fiabilité, la faible consommation d'énergie, la taille minimale et les autres caractéristiques fonctionnelles et de performance offertes par les systèmes embarqués.

Les systèmes embarqués sont le "cerveau" des dispositifs IoT, fournissant la base matérielle et logique qui leur permet de fonctionner intelligemment et efficacement. Ces systèmes sont conçus pour exécuter des fonctions spécifiques, contrôler et surveiller. Cela correspond parfaitement aux exigences des dispositifs IoT qui nécessitent des capacités de traitement en temps réel et une faible consommation d'énergie. Étant donné que les systèmes embarqués sont ceux qui alimentent les capteurs, les processeurs et les actionneurs qui collectent les données, traitent les informations et exécutent les actions, ils constituent la fonction essentielle des dispositifs IoT. Cela signifie que tout dispositif IoT est, par essence, un système embarqué connecté au réseau, faisant des systèmes embarqués le composant crucial qui confère aux dispositifs IoT l'intelligence et la capacité de fonctionner de manière autonome.

2. Architecture de l'IoT et des Systèmes Embarqués

L'architecture de l'IoT est un cadre qui définit la manière dont les dispositifs, les applications et les réseaux IoT interagissent pour fournir les fonctionnalités requises. Cette architecture vise à assurer un traitement efficace de l'information, de la couche physique à la couche application.

2.1. Les Couches Architecturales Fondamentales de l'IoT (Modèle à 3 Couches et 5 Couches)

L'architecture de l'IoT est généralement présentée en deux modèles principaux : le modèle à trois couches et le modèle à cinq couches. L'évolution de l'architecture de 3 à 5 couches reflète la maturité du domaine et la complexité croissante des applications, nécessitant un traitement plus détaillé des données et des opérations. Le modèle à trois couches était généralement accepté aux premiers stades de développement, car il définissait l'idée principale de l'IoT. Cependant, il s'est avéré insuffisant pour la recherche IoT axée sur des aspects plus précis. Cela indique que la nécessité du modèle à cinq couches est apparue avec la complexité et l'expansion croissantes des systèmes IoT, nécessitant des couches supplémentaires pour gérer le transport efficace des données (couche de transport) et les aspects économiques et stratégiques (couche métier). Cette évolution reflète une transition d'une focalisation purement technique à l'intégration des aspects commerciaux et stratégiques pour réaliser la pleine valeur de l'IoT.

2.1.1 Modèle à trois Couches (Five-Layer Architecture)

Ce modèle est le plus fondamental et se compose de trois couches principales : la couche de perception, la couche réseau et la couche d'applications.

- **Couche de Perception (Perception Layer) :**

Cette couche est également connue sous le nom de couche de capteurs ou couche physique. Sa fonction principale est de reconnaître les objets et de collecter des informations de

l'environnement physique à l'aide d'une variété de capteurs tels que les capteurs de température, de mouvement, de pression, d'humidité, de lumière, de GPS, de caméras et d'étiquettes RFID, ainsi que des actionneurs. Cette couche est responsable de la collecte des données brutes et de leur conversion en un format numérique pouvant être traité par les couches supérieures.

- **Couche Réseau (Network Layer) :**

Cette couche est également connue sous le nom de couche de transport ou de communication. Elle agit comme un pont entre la couche de perception et les couches supérieures, et est responsable du transfert de données et de la communication entre les dispositifs IoT et l'unité de traitement centrale. Cette couche utilise une variété de protocoles de communication tels que le Wi-Fi, le Bluetooth, le Zigbee, le LoRaWAN, les réseaux 4G et 5G, et l'Ethernet.

- **Couche applications (Application Layer) :**

C'est la couche supérieure de ce modèle, et elle interagit directement avec l'utilisateur final. Cette couche fournit des interfaces conviviales et des fonctionnalités qui permettent aux utilisateurs d'accéder aux dispositifs IoT et de les contrôler. Cette couche définit également les applications qui utilisent la technologie IoT, telles que les systèmes de maisons intelligentes et de villes intelligentes.

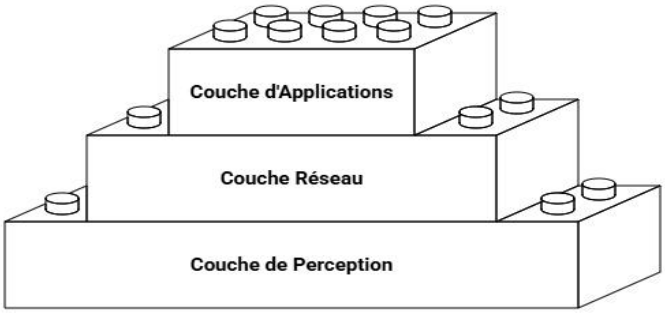


Figure 1.1 : Schéma illustrant le modèle IoT à trois couches (Three-Layer Architecture)

2.1.2 Modèle à Cinq Couches (Five-Layer Architecture)

Ce modèle étend le modèle à trois couches en ajoutant les couches de transport et métier, et est considéré comme définissant mieux tous les aspects de l'IoT.

- **Couche de Perception (Perception Layer) :**

Remplit la même fonction que dans le modèle à trois couches.

- **Couche de Transport (Transport Layer) :**

Cette couche est responsable du transfert des données des capteurs de la couche de perception vers la couche de traitement. Elle utilise divers réseaux et technologies tels que le RFID, le Bluetooth et la 3G pour assurer un transfert efficace des données.

- **Couche de Traitement (Processing Layer) :**

Cette couche est également connue sous le nom de couche de middleware. Sa fonction principale est de stocker, d'analyser et de traiter les données qu'elle reçoit. Cette couche comprend des systèmes de gestion de données, des plateformes d'analyse et des algorithmes d'apprentissage automatique qui sont utilisés pour extraire des informations utiles et prendre des décisions.

- **Couche applications (Application Layer) :**

  Remplit la même fonction que dans le modèle à trois couches.

- **Couche Métier (Business Layer) :**

C'est la couche supérieure du modèle à cinq couches. Elle est responsable de la gestion complète des activités IoT, y compris les modèles commerciaux, les analyses stratégiques, la recherche et le développement, le marketing et la gestion des utilisateurs. Cette couche vise à transformer les données et les informations en valeur commerciale tangible.



Figure 2.1 : Schéma illustrant l'architecture à cinq couches (Five-Layer Architecture)

2.2. Comparaison entre les Modèles d'Architecture de l'IoT (3 Couches vs 5 Couches)

Le tableau ci-dessous présente une comparaison synthétique entre le modèle à trois couches et le modèle à cinq couches de l'architecture IoT, en mettant en évidence les fonctions principales de chaque couche.

Tableau 1: Comparaison proposée entre les modèles d'architecture de l'IoT (3 couches vs 5 couches).

Couche / Layer	Module 3 Couches	Module 5 Couches	Function principal
Perception	Presenter	Presenter	Collecte des données des capteurs et actionneurs, conversion en format numérique
Roseau / Network	Presenter	Divisée en Transport et Traitement	Communication entre les dispositifs et transfert des données
Transport	Non present	Presenter	Transfert efficace des données entre la couche Perception et la couche Traitement
Treatment / Processing / Middleware	Partie de la couche Réseau	Presenter	Analyse et traitement des données, incluant Edge Computing
Applications	Presenter	Presenter	Interface utilisateur et contrôle des dispositifs IoT
Business / Métier	Non presenter	Presenter	Gestion des activités IoT, analyses stratégiques, transformation des données en valeur commerciale

2.3. Intégration des Systèmes Embarqués dans ces Couches

Les systèmes embarqués jouent un rôle crucial dans l'IoT en fournissant l'intelligence fondamentale qui permet aux appareils de communiquer entre eux et avec le cloud. Ils ne sont pas de simples composants passifs de l'IoT, mais des moteurs actifs d'intelligence et d'efficacité au niveau de l'appareil, permettant un traitement en temps réel et réduisant la dépendance au cloud centralisé à chaque étape.

•Dans la couche de perception :

Les systèmes embarqués constituent le cœur de cette couche, où ils alimentent et gèrent les capteurs et les actionneurs. Les microcontrôleurs tels qu'Arduino, ESP32 et Raspberry Pi sont des composants clés de cette couche, collectant les données des capteurs et les convertissant en un format numérique pouvant être traité.

•Dans la couche réseau :

Les systèmes embarqués sont responsables de la communication entre les appareils, prenant en charge divers protocoles de communication tels que le Wi-Fi, le Bluetooth et le Zigbee, et gérant le routage des données pour assurer une communication efficace.

•Dans la couche de traitement/middleware :

Les systèmes embarqués peuvent effectuer un traitement initial des données au niveau de l'appareil, ce que l'on appelle le calcul en périphérie (Edge Computing), avant de les envoyer au cloud. Cela réduit considérablement le temps de latence et améliore l'efficacité globale du système. Les systèmes embarqués sont conçus pour contrôler et surveiller et traiter les données localement. Cela signifie que leur fonction ne se limite pas à la collecte de données (couche de perception) ou à leur transmission (couche réseau), mais s'étend au traitement rapide des données au niveau de l'appareil. Cette capacité à traiter les données en périphérie du réseau réduit le temps de latence et permet de prendre des décisions et des actions immédiates, réduisant ainsi la nécessité de transférer toutes les données vers le cloud pour traitement. Cela renforce l'autonomie des dispositifs IoT et leur capacité à prendre des décisions immédiates sans dépendre constamment d'une connexion Internet centrale.

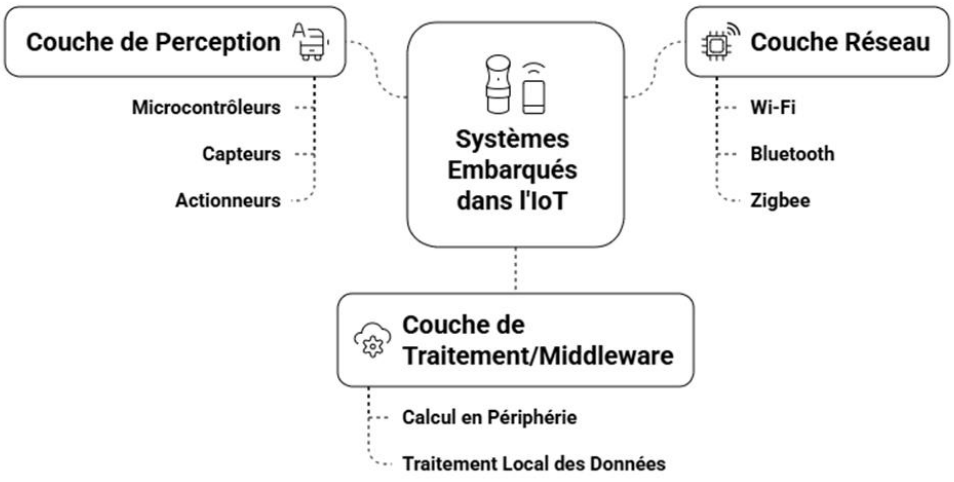


Figure 3 : Schéma illustrant l'intégration des systèmes embarqués dans ces couches

Source : Réalisée par les étudiants, via Napkin AI (https://app.napkin.ai)

3. Les Technologies Fondamentales qui Soutiennent l'IoT et les Systèmes Embarqués

La construction et le développement des systèmes IoT et des systèmes embarqués nécessitent un large éventail de technologies fondamentales qui s'intègrent pour permettre la collecte, le traitement et la prise de décision des données. Ces technologies sont la colonne vertébrale qui soutient les fonctions complexes et les applications diverses de l'IoT.

3.1. Capteurs et Actionneurs (Sensors and Actuators) : Types et Fonctions

Les capteurs et les actionneurs constituent la pierre angulaire de l'Internet des Objets Industriel (IoT), agissant comme les "sens" et les "muscles" du système. Ils sont l'interface physique vitale de l'IoT, représentant le point de rencontre entre le monde physique et le monde numérique, permettant l'interaction bidirectionnelle essentielle à l'automatisation intelligente. Sans capteurs, les dispositifs IoT ne peuvent pas percevoir leur environnement, et sans actionneurs, ils ne peuvent pas agir sur les données. Cette intégration est ce qui transforme les données brutes en actions significatives, incarnant l'essence de l'IoT dans la connexion des mondes physique et numérique.

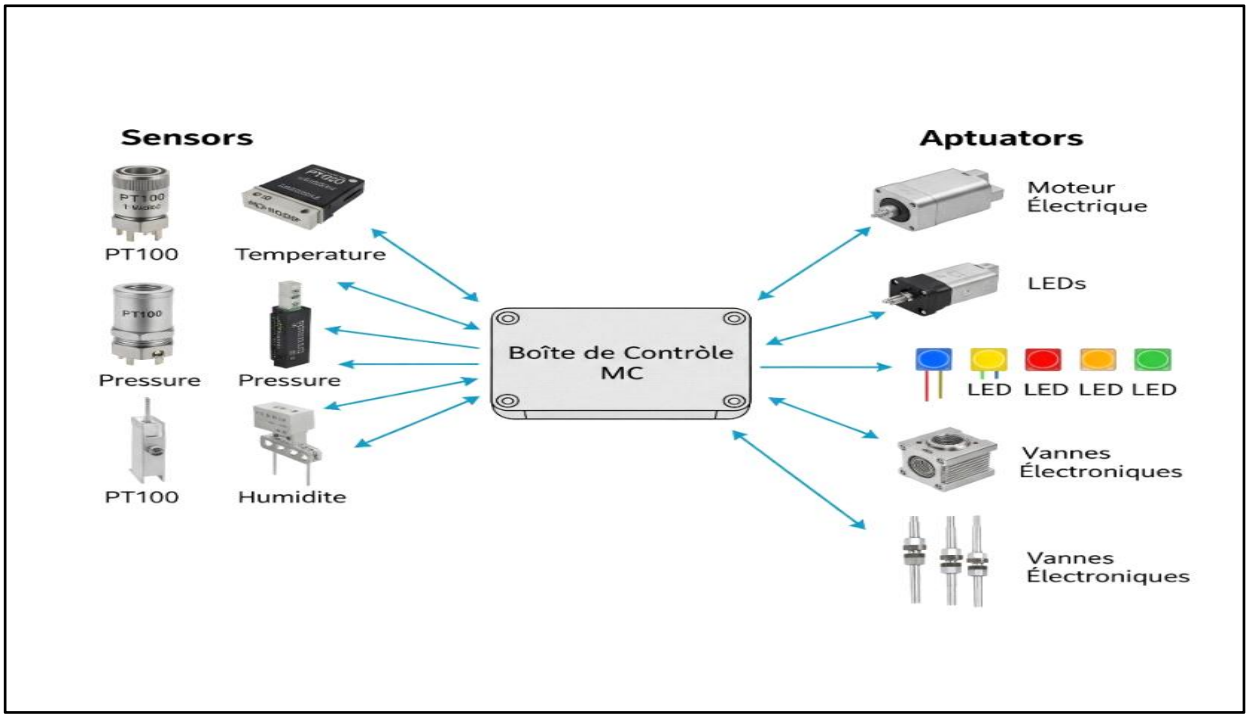


Figure 4: Schéma fonctionnel simplifié du système de mesure

3.1.1. Capteurs:

Ce sont des dispositifs qui détectent les changements dans les propriétés physiques, électriques, chimiques ou les variables environnementales, telles que la température, la lumière, la pression et le mouvement, et les convertissent en un signal électrique traitable. Les capteurs collectent des données vitales de l'environnement physique, telles que la température, la pression, l'humidité et la localisation, et fournissent des informations en temps réel. Les types de capteurs courants incluent les capteurs de température, les capteurs de pression, les capteurs optiques, les

capteurs de mouvement, les capteurs de niveau, les capteurs chimiques et les capteurs d'humidité. Les capteurs intelligents sont une évolution de ces dispositifs, combinant les capacités de détection avec des ressources informatiques (comme un microprocesseur ou un microcontrôleur) pour collecter et analyser des valeurs de haute précision.

3.1.2 Actionneurs :

Ce sont des composants responsables de la conversion des signaux électriques du système en énergie cinétique, permettant un mouvement ou une modification contrôlée des systèmes physiques dans les environnements industriels. Les actionneurs répondent aux signaux entrants du système IoT et initient des actions spécifiques basées sur les données reçues. Parmi les types d'actionneurs courants, on trouve les actionneurs linéaires qui permettent un mouvement de va-et-vient sur une trajectoire rectiligne (par exemple, pousser, tirer, soulever), les actionneurs rotatifs qui contrôlent les vannes, et les actionneurs pneumatiques qui convertissent l'air comprimé en mouvement mécanique. Les actionneurs intelligents sont des dispositifs électromécaniques avancés qui intègrent des capteurs, des composants électroniques de contrôle et des interfaces de communication pour réagir aux propriétés physiques détectées telles que la lumière, la chaleur, l'humidité ou la pression.

3.1.3 Integration:

Les nœuds de capteurs et d'actionneurs travaillent ensemble pour percevoir leur environnement, prendre des décisions et exécuter des tâches. Ce processus implique l'acquisition de données analogiques à partir des capteurs, leur traitement et leur conversion en signaux numériques, puis l'envoi de signaux de contrôle aux actionneurs pour exécuter les actions spécifiées.

3.2. Microcontrôleurs et Processeurs (Microcontrollers and Processors) : Leur Rôle dans les Dispositifs Périphériques

De petites puces informatiques appelées microcontrôleurs sont fréquemment utilisées dans les systèmes IoT pour contrôler et surveiller divers systèmes et appareils. Ces microcontrôleurs se trouvent généralement dans les capteurs, les actionneurs et d'autres composants électroniques, et aident à gérer le fonctionnement et le comportement de l'appareil tout en traitant et en envoyant des données.

Les microcontrôleurs sont les "mini-cerveaux" qui permettent l'intelligence distribuée en périphérie du réseau, réduisant ainsi la latence et renforçant l'autonomie des dispositifs IoT. Cela signifie que leur rôle ne se limite pas à la collecte de données, mais s'étend à leur traitement local. La capacité de traiter les données en périphérie du réseau (Edge Processing) réduit la nécessité d'envoyer toutes les données vers le cloud, ce qui diminue la latence et augmente l'efficacité énergétique. Cela renforce l'autonomie des dispositifs IoT et leur capacité à prendre des décisions immédiates sans dépendre constamment d'une connexion Internet centrale.

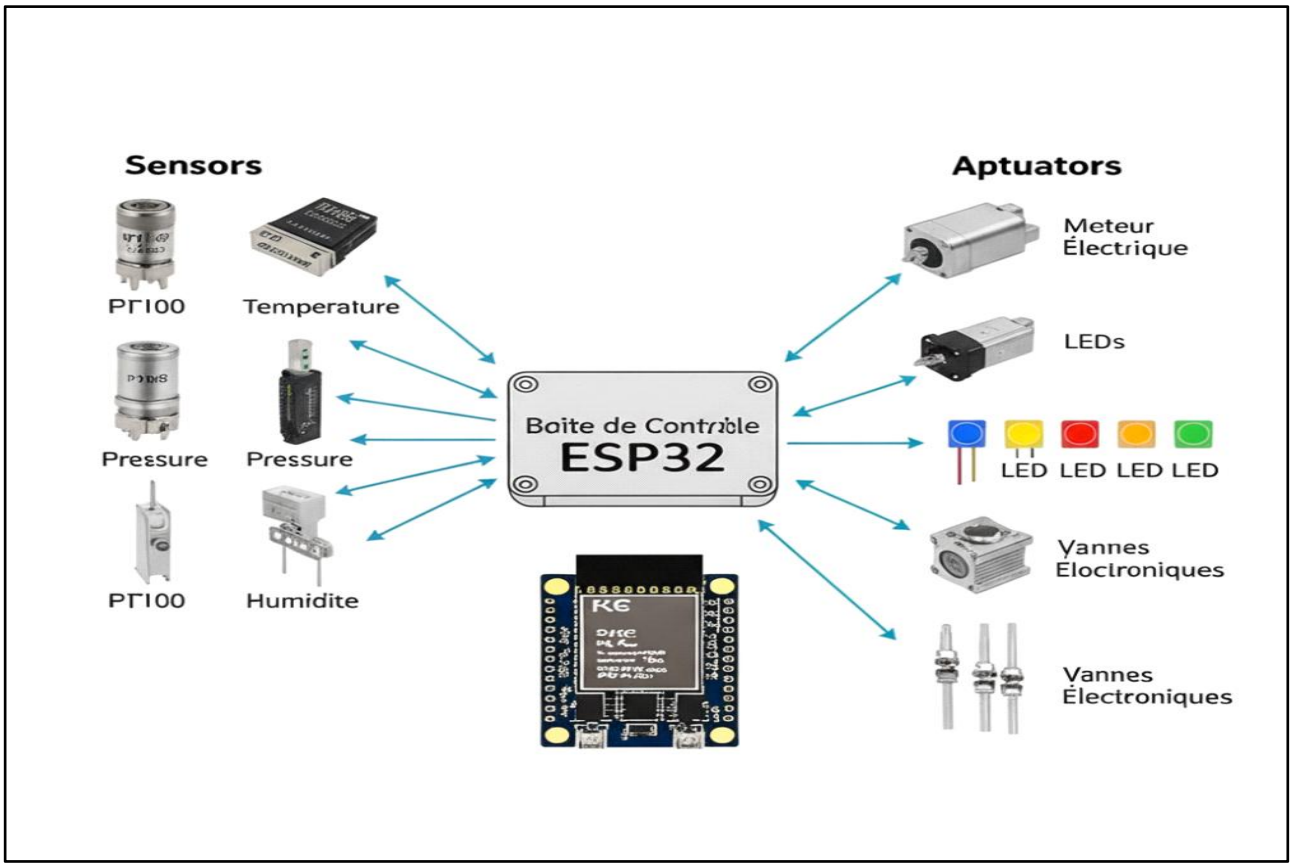


Figure 5: Architecture générale du système à base d'ESP32

Source : Image extraite du moteur de recherche Google

3.3. Protocoles de Communication et Réseaux (Communication Protocols and Networks) : Wi-Fi, Bluetooth, Zigbee, LoRaWAN, MQTT, CoAP, AMQP, 6LoWPAN, 5G/6G

Les protocoles de communication dans l'IoT garantissent que les appareils de différents fabricants peuvent fonctionner ensemble de manière transparente. La diversité des protocoles de communication et des réseaux dans l'IoT reflète l'équilibre entre les exigences en matière de consommation d'énergie, de bande passante, de latence et d'évolutivité, ce qui conduit à un écosystème complexe nécessitant des solutions multi-protocoles. Chaque protocole a ses propres spécifications et ses avantages et inconvénients. Cette diversité n'est pas aléatoire, mais est une réponse aux exigences spécifiques de l'application. Par exemple, BLE et LoRaWAN ont été conçus pour une faible consommation d'énergie , tandis que la 5G/6G vise des débits de données plus élevés et une latence plus faible. Cette diversité crée des défis en matière d'interopérabilité mais est essentielle pour répondre aux besoins variés des dispositifs IoT, des capteurs simples aux véhicules connectés complexes.

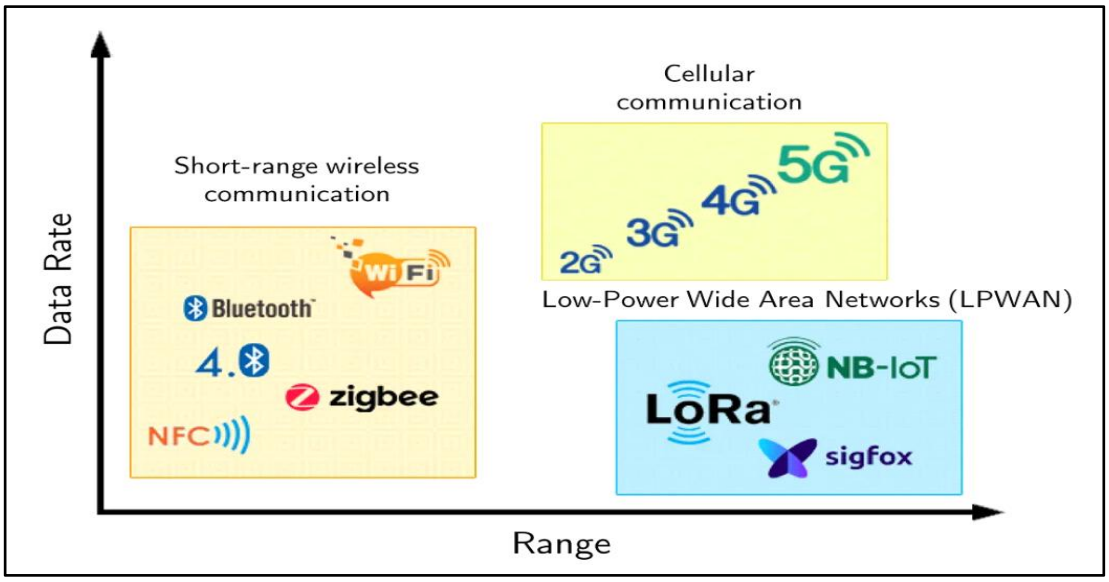


Figure 6: Classification des technologies de communication sans fil selon la portée et le débit de données (Data Rate)

Protocoles de réseau à courte portée :

- **Wi-Fi (802.11) :** Une option à faible coût, courante dans les maisons et les bureaux, mais qui consomme de l'énergie en continu et a une portée limitée.
- **Bluetooth Low Energy (BLE) :** Réduit considérablement la consommation d'énergie et le coût tout en maintenant une portée de connexion similaire au Bluetooth classique, adapté aux appareils électroniques grand public.
- **Zigbee (IEEE 802.15.4) :** Spécification pour les protocoles de communication de haut niveau utilisés pour créer des réseaux personnels avec de petites radios numériques à faible consommation d'énergie.
- **Near Field Communication (NFC) :** Un ensemble de protocoles de communication à courte portée (4 cm ou moins) pour l'échange de données entre deux appareils électroniques, avec une configuration simple et une connexion à faible vitesse.
- Protocoles de réseau étendu à faible consommation (LPWAN) :
- **LoRaWAN :** Convient aux appareils aux ressources limitées qui nécessitent une connexion à longue portée.
- **6LoWPAN :** Permet la connectivité IP standard (IPv6) sur des réseaux sans fil à faible consommation d'énergie (IEEE802.15.4), et prend en charge un nombre énorme d'adresses IP.
- Réseaux cellulaires :
- **LTE (Long-Term Evolution) :** Norme de communication sans fil à large bande pour les appareils mobiles, qui augmente la capacité et la vitesse des réseaux sans fil.
- **5G :** Offre des capacités de cinquième génération améliorées et étendues en termes de débits de données et de latence. Il est idéal pour connecter les dispositifs IoT.

- **6G :** Vise à répondre aux exigences IoT plus strictes par rapport à la 5G, y compris la connectivité massive, les communications IoT ultra-fiables et à faible latence, les protocoles de communication améliorés et la couverture réseau IoT étendue, ainsi que les dispositifs IoT intelligents.
- Protocoles de couche application/messagerie :
- **MQTT (Message Queuing Telemetry Transport) :** Protocole de messagerie léger pour la communication de machine à machine, principalement utilisé pour les communications à faible bande passante vers des emplacements distants. Il utilise un modèle de publication-abonnement.
- **CoAP (Constrained Application Protocol) :** Protocole conçu pour les appareils à bande passante limitée et les réseaux à capacité limitée.
- **AMQP (Advanced Message Queuing Protocol) :** Couche logicielle qui crée l'interopérabilité entre les middlewares de messagerie, aidant un ensemble de systèmes et d'applications à travailler ensemble.
- **DDS (Data Distribution Service) :** Protocole de communication peer-to-peer polyvalent, adapté aux systèmes en temps réel.

3.4. Cloud Computing et Edge Computing (Cloud and Edge Computing) : Avantages et Défis

La transition de la dépendance totale au cloud vers des modèles de calcul hybrides (périphérie-cloud) représente un changement stratégique dans l'architecture IoT, motivé par le besoin urgent de réduire la latence et d'améliorer la confidentialité et l'efficacité. Au début, le cloud était la solution idéale pour stocker et traiter d'énormes quantités de données. Cependant, avec l'augmentation du nombre de dispositifs IoT et des exigences de traitement en temps réel , les défis de latence, de consommation de bande passante et de confidentialité sont devenus évidents. Cela a conduit à l'émergence du calcul en périphérie comme solution pour traiter les données plus près géographiquement de l'utilisateur. Les modèles hybrides sont le résultat de cette évolution, l'industrie réalisant que la solution optimale réside dans la répartition des charges de travail entre la périphérie et le cloud, ce qui équilibre la puissance centrale et l'efficacité locale.

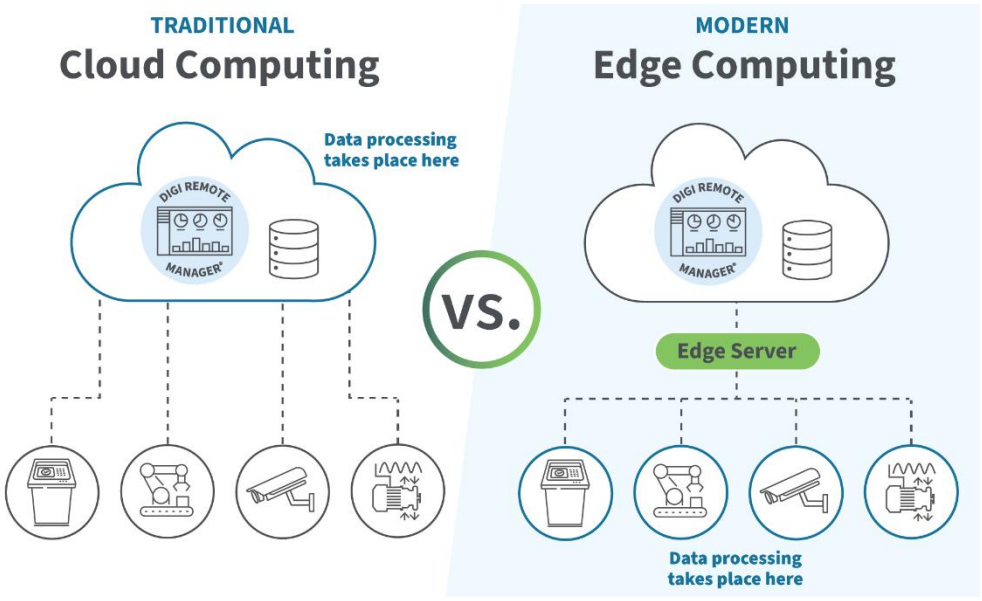


Figure 7 : Diagramme comparatif entre le Cloud Computing et l'Edge Computing

Source: Digi International (2024)

- **Cloud Computing :** Offre une grande flexibilité en termes de stockage de données et de puissance de calcul. Il permet aux utilisateurs d'accéder à l'infrastructure nécessaire pour évoluer sans avoir à la gérer eux-mêmes.
- **Avantages :** Évolutivité supérieure, flexibilité en matière de stockage et de puissance de traitement, et réduction des investissements initiaux.
- **Défis :** Latence élevée (résultant du transfert de données vers des serveurs distants), consommation de bande passante et préoccupations en matière de confidentialité et de sécurité des données sensibles.
- **Edge Computing :** Rapproche le traitement géographiquement de l'utilisateur, où les données sont traitées localement sur les dispositifs périphériques.
- **Avantages :** Réduction significative de la latence (de 2 secondes à 200 millisecondes dans certains cas) , amélioration de la confidentialité des données (les informations sensibles sont traitées localement) , et réduction de la consommation de bande passante et des coûts d'exploitation.
- **Défis :** Capacités de traitement et de mémoire limitées sur les dispositifs périphériques, et complexité de la gestion des ressources distribuées.
- **Modèles Hybrides :** Combinent les forces du calcul en périphérie et du cloud, tels que le Fog Computing et le Mist Computing. Ces modèles améliorent l'utilisation de la bande passante et prennent en charge les applications à faible latence et sensibles à la confidentialité.

3.5. Intelligence Artificielle et Machine Learning (AI and Machine Learning) : Analyse des Données et Prise de Décision

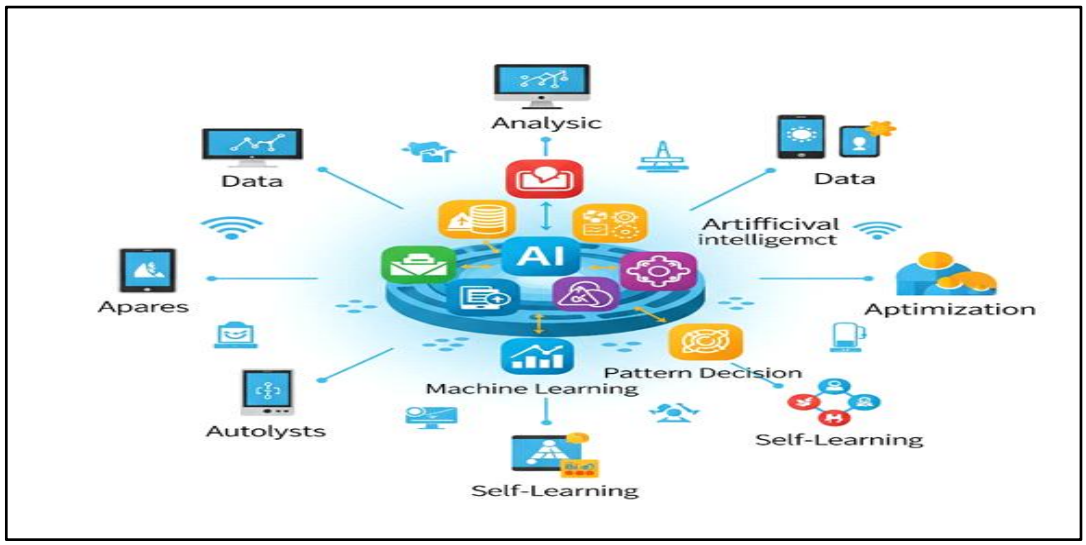


Figure 8: Architecture fonctionnelle et composants de l'Intelligence Artificielle (IA) et du Machine Learning

L'Intelligence Artificielle (IA) et l'apprentissage automatique (ML) améliorent considérablement les capacités de l'IoT en permettant aux appareils d'analyser les données localement, de prendre des décisions éclairées en temps réel et d'apprendre des modèles pour améliorer les performances. L'intégration de l'IA dans les systèmes embarqués représente un bond qualitatif dans les capacités de l'IoT, passant de la simple collecte de données à la compréhension intelligente et à la prise de décision autonome au niveau de l'appareil, ouvrant de nouvelles perspectives pour l'automatisation et l'optimisation. L'IoT collecte les données , et l'IA les traite et les analyse. Lorsque l'IA est directement intégrée aux systèmes embarqués , l'appareil lui-même devient capable d'analyser les données localement et de prendre des décisions éclairées en temps réel. Cela réduit considérablement la latence et rend les systèmes plus autonomes, ce qui conduit à une automatisation plus intelligente et à des décisions plus rapides. Cette transition de simples "objets connectés" à des "objets intelligents" est au cœur de la révolution de l'IoT et de l'IA.

- **IA Embarquée (Embedded AI) :** C'est l'intégration de l'IA directement dans les systèmes embarqués, permettant aux appareils de traiter les données et de prendre des décisions de manière autonome en périphérie du réseau sans dépendre des services cloud. Cela conduit à des opérations plus rapides et plus efficaces.
- **Technologies :** Des technologies telles que TinyML (apprentissage automatique miniaturisé) sont spécialement conçues pour les appareils à faible consommation d'énergie. La compression de modèles (comme l'élagage et la quantification) est également utilisée pour réduire la taille des modèles d'IA sans sacrifier la précision.
- Applications :
- **Soins de santé :** Les appareils portables utilisent l'IA pour surveiller la fréquence cardiaque et détecter les anomalies en temps réel.
- **Industrie 4.0 :** Les machines industrielles équipées de systèmes d'IA embarqués peuvent surveiller les équipements, traiter les données des capteurs en temps réel et effectuer des autodiagnostics, réduisant ainsi les temps d'arrêt et améliorant l'efficacité opérationnelle.

- **Agriculture intelligente :** Utilisée pour analyser les données des capteurs en temps réel (par exemple, les niveaux d'humidité du sol) afin de permettre aux systèmes d'irrigation robotisés d'arroser automatiquement les cultures.
- **Avantages :** Prise de décision plus rapide , efficacité et automatisation accrues , et personnalisation et sécurité améliorées.

3.6. Les Technologies Fondamentales avec Exemples d'Utilisation

Ce tableau offre une vue d'ensemble complète et concise des technologies fondamentales qui constituent la colonne vertébrale de l'IoT et des systèmes embarqués. Il relie chaque technologie à une brève description, puis fournit des exemples pratiques et spécifiques de ses utilisations dans des contextes IoT, ce qui renforce la compréhension pratique. Il identifie également les sources pour chaque technologie, ce qui permet au lecteur de se référer facilement aux informations originales pour plus de détails et de vérification. Bien qu'il se concentre sur chaque technologie individuellement, il illustre implicitement comment ces technologies s'intègrent pour créer des solutions IoT complexes, et constitue un excellent outil pédagogique pour les étudiants de troisième cycle, car il résume les informations essentielles de manière organisée et facile à comprendre.

Tableau 2: Résumé des technologies fondamentales de l'Internet des Objets et des systèmes embarqués avec exemples d'utilisation

Technologies	Description	Examples d'Utilisation dans l'IoT
Captures et Actioners	Dispositifs qui détectent les changements environnementaux (capteurs) et convertissent les signaux en mouvement physique (actionneurs).	Capteurs de température dans les thermostats intelligents, capteurs de mouvement dans les systèmes de sécurité, actionneurs de contrôle de vannes dans l'irrigation intelligente.
Microcontrollers et Processors	Petites puces informatiques qui contrôlent et surveillent les systèmes et les appareils, et fournissent la puissance de traitement en périphérie du réseau.	Microcontrôleurs dans les appareils portables, Raspberry Pi dans les passerelles IoT.
Protocoles de Communication et Réseaux	Normes et règles permettant l'échange de données entre les appareils et les	Wi-Fi pour les maisons intelligentes, Bluetooth pour les appareils portables, LoRaWAN pour l'agriculture intelligente à longue portée, MQTT pour les messages de capteurs,

	systèmes.	5G/6G pour les véhicules connectés.
Cloud Computing et Edge Computing	**Cloud :** Traitement et stockage des données sur des serveurs distants. **Edge :** Traitement des données près de leur source.	**Cloud :** Analyse de méga données, stockage de données historiques. **Edge :** Traitement de données en temps réel dans les usines intelligentes, prise de décisions immédiates dans les véhicules autonomes.
Intelligence Artificielle et Machine Learning	Algorithmes permettant aux systèmes d'apprendre des données et de prendre des décisions.	Maintenance prédictive des machines, reconnaissance de formes dans les soins de santé, optimisation de la consommation d'énergie dans les villes intelligentes.

4. Domaines d'Applications Étendus

Les technologies de l'Internet des Objets et des systèmes embarqués transforment radicalement un large éventail de secteurs, offrant des solutions innovantes qui augmentent l'efficacité, améliorent la qualité de vie et soutiennent la durabilité. Ces applications vont au-delà de la simple connexion d'appareils, pour former des écosystèmes intégrés qui répondent intelligemment aux besoins des utilisateurs et de l'environnement.

4.1. Villes Intelligentes (Smart Cities) : Gestion du Trafic, Énergie, Déchets, Sécurité Publique

Les villes intelligentes tirent parti des technologies IoT pour améliorer l'efficacité, la durabilité et la qualité de vie urbaine globale. Les villes intelligentes ne sont pas seulement un ensemble d'appareils connectés, mais un écosystème intégré où les technologies IoT et les systèmes embarqués transforment l'infrastructure traditionnelle en entités intelligentes qui répondent dynamiquement aux besoins des habitants et de l'environnement, renforçant ainsi la durabilité et la résilience. L'IoT dans les villes intelligentes va au-delà de la simple surveillance. Il permet le contrôle à distance et l'activation automatique de l'infrastructure. Cela signifie que les données collectées par les capteurs ne sont pas seulement utilisées pour l'analyse, mais pour actionner les actionneurs qui modifient la réalité physique, comme les feux de circulation, l'éclairage et la gestion des déchets. Cette transition de la surveillance à la réponse dynamique et à l'optimisation globale est ce qui rend les villes véritablement intelligentes, et met en évidence le rôle des systèmes embarqués en tant qu'élément exécutif essentiel.

- **Gestion du trafic :** Les capteurs de trafic surveillent le flux des véhicules, les embouteillages et la vitesse en temps réel pour optimiser le timing des feux de circulation, réduire les retards et améliorer la sécurité routière. Les systèmes de transport intelligents basés sur l'IoT révolutionnent la mobilité urbaine en fournissant des données en temps réel pour une gestion dynamique du trafic et des solutions de stationnement intelligentes.
- **Gestion de l'énergie :** Les réseaux intelligents améliorent l'efficacité énergétique. Les capteurs d'énergie intelligents surveillent la consommation d'énergie en temps réel,

permettant aux services publics d'optimiser la distribution, de détecter les pannes et d'améliorer l'efficacité énergétique. Les villes visent à réduire la consommation de ressources, à diminuer l'empreinte carbone et à améliorer les opérations urbaines.

- **Gestion des déchets :** Les solutions IoT améliorent la gestion des déchets grâce à la surveillance en temps réel et à l'analyse des données. Les capteurs sans fil dans les poubelles identifient les niveaux de déchets, ce qui optimise les itinéraires de collecte et réduit les coûts d'exploitation.
- **Sécurité publique :** L'IA peut améliorer la sécurité publique en analysant les données des caméras de surveillance, des microphones et d'autres capteurs pour détecter et prévenir les crimes et autres menaces de sécurité.

4.2. Industrie 4.0 (Industry 4.0) : Maintenance Prédictive, Contrôle Qualité, Gestion des Chaînes d'Approvisionnement

Les systèmes IoT embarqués transforment l'industrie manufacturière, connue sous le nom d'Industrie 4.0, en intégrant les technologies de l'information et de la communication au cadre industriel pour renforcer l'automatisation et la numérisation. L'Industrie 4.0 est l'incarnation du concept d'« usine intelligente » où les données sont transformées en valeur économique directe grâce à l'automatisation avancée et à l'amélioration continue, reflétant une transformation radicale des modèles de production. L'IoT dans l'Industrie 4.0 ne se limite pas à la collecte de données, mais les utilise pour l'analyse en temps réel, la maintenance prédictive et l'amélioration de la qualité. Cela indique que les données sont utilisées pour prendre des décisions automatiques ou semi-automatiques qui affectent directement l'efficacité et les coûts de production. Cette transition des opérations manuelles vers des systèmes intelligents autonomes réduit les temps d'arrêt et le gaspillage et augmente la productivité , illustrant comment l'IoT crée une valeur économique tangible dans l'industrie.

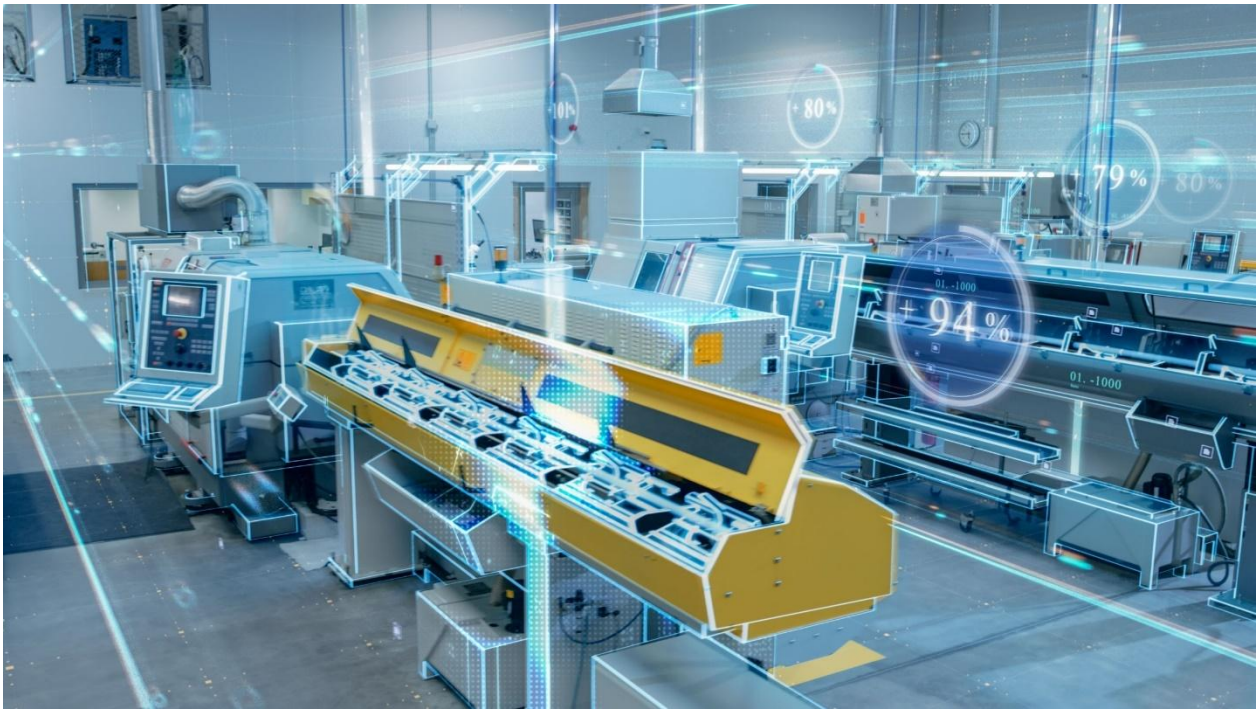


Figure 9 : Illustration d'une usine intelligente (Industrie 4.0) avec robots, machines connectées et écrans de données en temps réel

- **Maintenance prédictive :** Ces systèmes collectent des données en temps réel, telles que la température, la pression ou les vibrations, des machines d'usine, puis les analysent à l'aide d'algorithmes d'apprentissage automatique pour prédire le moment où une maintenance est nécessaire. Cela prévient les arrêts imprévus et réduit les coûts.
- **Contrôle qualité :** Les systèmes IoT embarqués surveillent les lignes de production en temps réel pour détecter les défauts potentiels dans les produits fabriqués. Cela permet d'identifier et de corriger rapidement les défauts, ce qui réduit le gaspillage et améliore la qualité des produits.
- **Gestion intelligente des chaînes d'approvisionnement :** Les systèmes IoT suivent les niveaux de stock et automatisent le processus de commande, ce qui réduit les risques de rupture de stock ou de surstockage. Ils peuvent également suivre la livraison des matériaux au sein de la chaîne d'approvisionnement, ce qui améliore l'efficacité de la chaîne d'approvisionnement et réduit les retards.
- **Surveillance et gestion à distance :** Les systèmes IoT embarqués peuvent surveiller à distance les performances et l'état des machines et des équipements. Cela aide à identifier les problèmes potentiels et réduit les risques de défaillance grâce à des actions préventives.

4.3. Soins de Santé (Healthcare) : Surveillance à Distance des Patients, Dispositifs Médicaux Intelligents

L'IoT transforme les soins de santé en offrant des méthodes innovantes pour surveiller, diagnostiquer et traiter les patients, ce qui améliore considérablement l'expérience des patients et réduit les coûts. L'IoT dans les soins de santé représente une transition vers un modèle de soins de

santé plus proactif et personnalisé, où les patients sont mieux à même de gérer leur santé et de réduire le fardeau sur l'infrastructure de santé traditionnelle. Les applications mentionnées, telles que la surveillance du glucose, de la fréquence cardiaque et les dispositifs ingérables , sont toutes axées sur la surveillance continue et la collecte de données objectives. Cela permet des interventions plus faciles et plus efficaces et réduit la paperasserie et la bureaucratie. Cette transition des soins réactifs (lorsqu'une personne tombe malade) aux soins préventifs et proactifs (surveillance continue) réduit la nécessité d'un séjour dans un établissement de soins de santé et diminue les coûts des soins de santé , modifiant le modèle de soins de santé de centralisé à décentralisé.

- **Surveillance à distance des patients (RPM) :** Les appareils portables, tels que les montres intelligentes, les téléphones et les capteurs, collectent automatiquement les mesures de santé comme la température corporelle, la tension artérielle et le pourcentage de graisse corporelle. Ces appareils peuvent recommander un traitement médical ou générer des alertes basées sur les informations collectées.
- **Surveillance du glucose et de la fréquence cardiaque :** Les dispositifs IoT équipés de capteurs intégrés ou portables surveillent automatiquement les niveaux de glucose ou la fréquence cardiaque, enregistrent les données et alertent les professionnels de la santé. Certains dispositifs embarqués peuvent même fournir de l'insuline automatiquement si nécessaire.
- **Surveillance de l'hygiène des mains :** Les dispositifs IoT sont utilisés pour surveiller l'hygiène dans les établissements de soins de santé afin de garantir des normes de propreté appropriées.
- **Capteurs ingérables :** De petits capteurs en forme de pilule sont ingérés pour collecter diverses données biométriques, telles que la détermination du niveau de pH de l'estomac ou l'identification d'hémorragies internes.
- **Chirurgie robotique :** Les chirurgiens peuvent injecter de petits robots connectés à Internet dans le corps humain pour effectuer des opérations chirurgicales, réduisant ainsi les erreurs humaines lorsque la précision est requise.

4.4. Agriculture Intelligente (Smart Agriculture) : Surveillance des Cultures, Irrigation Intelligente, Lutte Antiparasitaire

- Les systèmes IoT embarqués améliorent considérablement le niveau d'intelligence de la production agricole grâce aux innovations matérielles et à l'évolution de l'architecture. L'agriculture intelligente est un exemple clé de la manière dont l'IoT et les systèmes embarqués peuvent augmenter l'efficacité des ressources et atténuer les défis environnementaux, conduisant à des pratiques agricoles plus durables et productives. Les données indiquent des améliorations significatives de la valeur de production par unité de surface (augmentation de 81%), une réduction de la demande de main-d'œuvre (62,5%) et une diminution des émissions de carbone (38,6%). Ces chiffres ne montrent pas seulement l'efficacité économique, mais aussi les avantages environnementaux. L'utilisation de capteurs et d'actionneurs permet une irrigation de précision et une lutte antiparasitaire intégrée, ce qui conduit à une meilleure utilisation de l'eau et à une réduction de l'utilisation des pesticides. Cela démontre que l'IoT et les systèmes embarqués ne visent pas seulement à augmenter la productivité, mais à réaliser une agriculture durable grâce à l'utilisation optimale des ressources.



Figure 10: Illustration de l'intégration des Drones et de l'irrigation intelligente dans l'Agriculture de Précision

- **Surveillance des cultures :** Les capteurs de sol détectent les niveaux d'azote, de phosphore et de potassium (NPK), la teneur en humidité, le pH et la conductivité électrique, ce qui permet une application précise des engrais et une irrigation contrôlée.
- **Irrigation intelligente :** Les systèmes d'irrigation automatisés utilisent des capteurs d'humidité du sol et des données météorologiques pour optimiser la gestion des cultures, augmenter les rendements et conserver les ressources en eau. Les microcontrôleurs embarqués, tels que l'ESP32, peuvent faire fonctionner des contrôleurs PID pour ajuster dynamiquement les ouvertures des vannes à impulsion en fonction des seuils d'humidité du sol.
- **Lutte antiparasitaire et contre les maladies :** Des modèles légers, tels que YOLOv3-Tiny, sont déployés sur des dispositifs périphériques pour identifier les types de ravageurs avec une précision de 90%, ce qui réduit l'utilisation de pesticides de 35%. Les capteurs de maladies végétales identifient les infections à un stade précoce, ce qui aide à prévenir les pertes de récoltes.

4.5. Véhicules Connectés et Transport (Connected Vehicles and Transportation) : Systèmes d'Aide à la Conduite, Gestion des Flottes

Les véhicules connectés sont intégrés dans les systèmes IoT en leur permettant de communiquer et d'interagir avec d'autres appareils et systèmes au sein d'un réseau IoT. Les véhicules agissent comme des nœuds qui collectent, traitent et transmettent des données. Les véhicules connectés sont plus que de simples dispositifs IoT mobiles ; ce sont des plateformes de données mobiles qui transforment l'infrastructure de transport en un écosystème interactif et intelligent, améliorant la sécurité, l'efficacité et réduisant les coûts environnementaux. Les véhicules connectés ne se contentent pas de collecter des données de leurs propres capteurs, mais peuvent accéder à des informations essentielles qui dépassent leur environnement immédiat. Cela signifie qu'ils interagissent avec l'infrastructure (V2I) et avec d'autres véhicules (V2V). Cette capacité à échanger des données à grande échelle permet une gestion dynamique du trafic et une réduction des coûts d'infrastructure, les véhicules eux-mêmes agissant comme des capteurs mobiles. Cela réduit le besoin d'équipements d'infrastructure coûteux et rend le système de transport plus durable sur le plan environnemental, illustrant comment l'IoT crée de la valeur au-delà des simples fonctions du véhicule individuel.

- **Amélioration de la gestion du trafic :** La communication en temps réel entre les véhicules et l'infrastructure facilite les algorithmes de feux de circulation dynamiques, ce qui contribue à réduire les embouteillages et les émissions.
- **Systèmes avancés d'aide à la conduite (ADAS) :** Ces systèmes fournissent des avertissements de sécurité, évitent les accidents et aident à la conduite.
- **Gestion des flottes :** Permet la surveillance et le contrôle à distance des véhicules, y compris le niveau de carburant, l'état de la batterie, le suivi de la localisation, la pression des pneus, l'état de chargement et l'état d'alarme.
- **Sécurité routière :** Les applications incluent l'avertissement post-collision, l'avertissement pré-collision, l'avertissement de changement de voie et l'assistance à la circulation aux intersections.

4.6. Commerce de Détail et Logistique (Retail and Logistics) : Gestion des Stocks, Expérience Client, Suivi des Expéditions

L'IoT modifie considérablement l'industrie mondiale du commerce de détail en introduisant des innovations qui répondent aux habitudes de consommation en évolution et permettent aux détaillants d'offrir des expériences d'achat plus personnalisées. Dans les secteurs du commerce de détail et de la logistique, l'IoT transforme les opérations de réactives en prédictives, ce qui se traduit par une efficacité accrue, une réduction du gaspillage et une amélioration radicale de l'expérience client. Les applications mentionnées (gestion des stocks, analyse des ventes, prévention des pertes, optimisation de la chaîne d'approvisionnement) bénéficient toutes des données en temps réel. Ces données ne sont pas seulement utilisées pour la surveillance, mais pour la prédiction (comme la maintenance prédictive , l'expédition prédictive) et l'automatisation (comme la commande automatique , les magasins Amazon Go sans caisse). Cette transition des opérations manuelles vers des systèmes intelligents autonomes réduit les erreurs humaines et le gaspillage et augmente la satisfaction client , illustrant comment l'IoT crée de la valeur au-delà de la simple efficacité opérationnelle.

- **Gestion des stocks :** Les technologies IoT, telles que les capteurs RFID et les étiquettes intelligentes, permettent le suivi et la gestion des stocks en temps réel. Cette application réduit les erreurs humaines et le gaspillage et garantit que les bons produits sont toujours en stock.
- **Analyse des ventes et expérience client :** Les dispositifs IoT collectent des informations sur le comportement des clients, leurs préférences et leurs achats passés. Ces données sont utilisées pour améliorer les stratégies de vente et de marketing et personnaliser les offres. Les capteurs de proximité et les étiquettes intelligentes permettent de fournir un contenu personnalisé aux acheteurs dans des endroits spécifiques du magasin.
- **Prévention des pertes :** Des caméras et des capteurs IoT sont déployés pour surveiller l'activité du magasin, ce qui aide à dissuader le vol et la fraude.
- **Optimisation de la chaîne d'approvisionnement et de la logistique :** Les capteurs IoT surveillent les marchandises du point de production à la livraison, garantissant que les produits sont livrés à temps et en bon état. Dans la logistique, l'IoT offre des capacités de suivi et de traçabilité plus rapides, plus précises, plus prédictives et plus sécurisées.

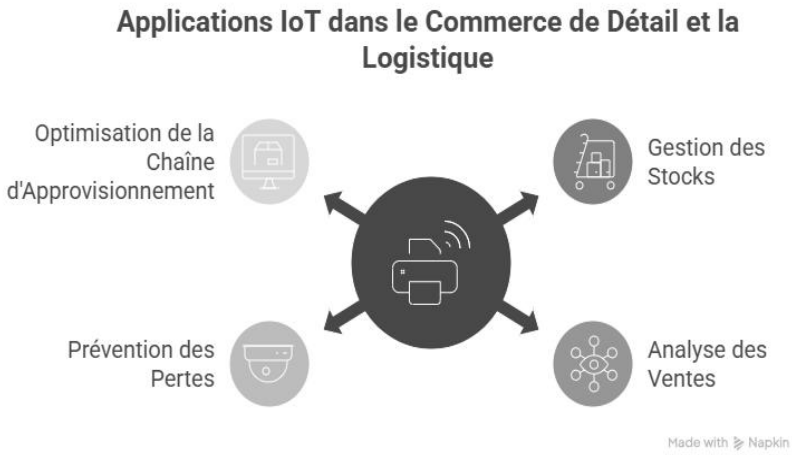


Figure 11: application LoT dans le commerce de détail et la logistique

4.7. Exemples d'Applications de l'IoT et des Systèmes Embarqués dans Divers Secteurs

Ce tableau offre un aperçu rapide et facile à comprendre de la vaste gamme d'applications de l'IoT et des systèmes embarqués dans divers secteurs. Il relie chaque application à ses principaux avantages, illustrant la valeur ajoutée que ces technologies apportent. Il aide également l'étudiant en thèse à démontrer une compréhension globale de l'impact de l'IoT sur le monde réel, ce qui est crucial dans un chapitre dédié aux applications. De plus, il renvoie le lecteur aux sources originales pour une exploration plus approfondie de chaque secteur, et permet une comparaison rapide entre les types d'applications et les avantages à travers les différents secteurs, ce qui peut révéler des modèles communs ou des différences uniques.

Tableau 3: Exemples d'applications de l'IoT et des systèmes embarqués dans différents secteurs avec principaux avantages

Sector	Examples applications	Advantages Principaux
Villes Intelligentes	Gestion intelligente du trafic, éclairage public écoénergétique, gestion intelligente des déchets, surveillance de la qualité de l'air.	Réduction des embouteillages, diminution de la consommation d'énergie, amélioration de la propreté, renforcement de la sécurité publique.
Industrie 4.0	Maintenance prédictive des machines, surveillance de la qualité de production en temps réel, gestion intelligente des stocks, suivi de la chaîne	Réduction des temps d'arrêt, amélioration de la qualité des produits, augmentation de l'efficacité opérationnelle, réduction des coûts.

	d'approvisionnement.	
So ins de Santé	Surveillance à distance des patients (cœur, glucose), appareils portables, chirurgie robotique, capteurs d'hygiène.	Soins de santé proactifs, réduction du besoin d'hospitalisation, amélioration de la précision des traitements, réduction des erreurs humaines.
Agriculture Intelligent	Irrigation intelligente, surveillance de la santé des cultures, lutte antiparasitaire et contre les maladies, suivi du bétail.	Augmentation des rendements, économie d'eau, réduction de l'utilisation de pesticides, amélioration de la durabilité environnementale.
Véhicules Connects et Transport	Systèmes d'aide à la conduite, gestion des flottes, gestion intelligente du trafic, avertissements de sécurité.	Amélioration de la sécurité routière, efficacité énergétique, réduction des embouteouteillages, amélioration de l'expérience de conduite.
Commerce de Détail et Logistique	Gestion des stocks en temps réel, expériences d'achat personnalisées, prévention du vol, suivi des expéditions.	Réduction du gaspillage, augmentation des ventes, amélioration de la satisfaction client, efficacité de la chaîne d'approvisionnement.

5. Défis Majeurs de l'Internet des Objets et des Systèmes Embarqués

Malgré le potentiel immense offert par l'Internet des Objets et les systèmes embarqués, leur adoption et leur déploiement à grande échelle sont confrontés à un ensemble de défis fondamentaux. Ces défis nécessitent des solutions innovantes et une approche multifacette pour assurer la croissance durable et sécurisée de ces technologies.

5.1. Sécurité et Confidentialité (Security and Privacy) : Points Faibles, Protection des Données

La sécurité et la confidentialité représentent des défis considérables pour les applications IoT, soulevant un tout nouveau niveau de préoccupations concernant la confidentialité des utilisateurs en ligne. La sécurité et la confidentialité ne sont pas seulement des défis techniques, mais des obstacles majeurs à l'adoption généralisée de l'IoT, nécessitant une approche globale combinant l'innovation technique, la confiance réglementaire et la sensibilisation des utilisateurs. Les données indiquent que la majorité des gens ne croient pas que les produits sont sécurisés, et moins d'un tiers des utilisateurs font confiance aux fabricants, aux fournisseurs de services et aux gouvernements pour préserver leur vie privée. Cela entrave considérablement l'adoption de l'IoT. Ce n'est pas seulement un problème technique, mais un problème de confiance publique. Les solutions proposées, telles que le chiffrement fort et les mises à jour régulières , sont nécessaires, mais doivent être accompagnées de normes internationales, de certifications et de réglementations pour instaurer la confiance. Cela

indique que surmonter les défis de sécurité et de confidentialité nécessite un effort multifacette qui va au-delà des solutions techniques pour inclure les dimensions juridiques, réglementaires et sociales.

- **Points faibles :** Les dispositifs IoT augmentent considérablement la surface d'attaque. De nombreux dispositifs IoT sont conçus avec des mesures de sécurité minimales, ce qui les rend vulnérables aux piratages et autres cyberattaques. Les menaces courantes incluent l'accès non autorisé, les violations de données, les attaques par déni de service distribué (DDoS) et les logiciels malveillants.
- **Préoccupations en matière de confidentialité :** Les dispositifs IoT peuvent surveiller les comportements des utilisateurs et collecter des détails personnels sensibles. Les gens craignent particulièrement les violations et les abus, car la majorité des gens ne croient pas que les produits sont sécurisés.
- **Défis de mise en œuvre :** Les dispositifs IoT ont souvent de petits modules de sortie, ce qui rend impossible d'y adapter les méthodes de protection actuelles, nécessitant des algorithmes légers.
- **Solutions :** Mettre en œuvre des contrôles d'accès appropriés, utiliser des techniques de chiffrement robustes et mettre à jour régulièrement les logiciels et les micrologiciels. Une approche de sécurité multicouche est recommandée, y compris la sécurisation du matériel de l'appareil, des communications réseau et du stockage des données.

5.2. Consommation d'Énergie (Power Consumption) : Contraintes de Batterie, Conception de Systèmes Eco énergétiques

L'efficacité énergétique est l'un des défis les plus importants des systèmes numériques modernes, en particulier en ce qui concerne les systèmes embarqués autonomes. Le défi de la consommation d'énergie dans l'IoT est un paradoxe ; alors que les appareils visent à être omniprésents, les contraintes énergétiques limitent leur diffusion et leur durabilité, poussant l'innovation vers des solutions "zéro énergie" et "récolte d'énergie autonome". L'IoT vise une connectivité massive et le déploiement de milliards d'appareils. Cependant, les contraintes énergétiques et la durée de vie limitée des batteries sont des défis majeurs. Les solutions traditionnelles comme les câbles ou le remplacement des batteries ne sont pas pratiques à cette échelle. Cela conduit à la nécessité d'appareils qui récoltent l'énergie de leur environnement et de conceptions à très faible consommation d'énergie. Ce défi stimule l'innovation non seulement dans l'amélioration de l'efficacité énergétique, mais aussi dans la modification du modèle d'alimentation des dispositifs IoT vers des solutions plus durables.

- **Contraintes :** De nombreux dispositifs IoT, en particulier ceux situés dans des endroits éloignés, fonctionnent sur batterie. La consommation d'énergie a un impact significatif sur la durée de vie de la batterie, ce qui rend la conception de systèmes écoénergétiques cruciale.
- **Défis :** La consommation d'énergie est influencée par la tension d'alimentation et la fréquence d'horloge. Les applications qui impliquent des capteurs fonctionnant dans des systèmes autonomes, comme la surveillance à distance, nécessitent une grande efficacité énergétique.
- **Solutions :** Utiliser des techniques telles que le Dynamic Voltage and Frequency Scaling (DVFS) pour ajuster dynamiquement la tension d'alimentation et la fréquence d'horloge. Concevoir des protocoles de communication à faible consommation d'énergie, tels que Bluetooth LE, Zigbee et LoRaWAN, pour réduire la consommation d'énergie tout en

maintenant la connectivité. Intégrer les tests d'efficacité énergétique dans le processus de développement.

5.3. Évolutivité (Scalability) : Gestion d'un Nombre Colossal d'Appareils et de Données

Le nombre d'appareils connectés à Internet augmente considérablement, à un rythme de 8% par an, et devrait dépasser les 30 milliards d'appareils d'ici 2025. Cette croissance massive impose des défis importants à l'infrastructure cloud. L'évolutivité dans l'IoT ne concerne pas seulement l'augmentation du nombre d'appareils, mais le développement d'une infrastructure flexible et distribuée capable de traiter d'énormes quantités de données de manière efficace, ce qui pousse vers des modèles de calcul décentralisés. La croissance massive du nombre de dispositifs IoT exerce une pression considérable sur l'infrastructure centralisée basée sur le cloud. Cela entraîne une charge importante et une latence inacceptable. La solution ne réside pas seulement dans l'augmentation de la capacité du cloud, mais dans la répartition des charges de travail via le calcul en périphérie. Cela indique que la véritable évolutivité de l'IoT réside dans la capacité à traiter les données là où elles sont générées, ce qui réduit la pression sur le réseau central et garantit une réponse rapide, permettant ainsi une croissance sans précédent du nombre d'appareils et d'applications.

- **Défis :** Gérer ce nombre colossal d'appareils et les données qu'ils génèrent tout en maintenant la qualité de service (QoS). Les architectures basées sur le cloud peuvent entraîner une charge de travail importante sur le cloud et une latence inacceptable.
- **Solutions :** Les nouveaux modèles de calcul tels que le calcul en périphérie, qui traitent les données sur des appareils proches de la périphérie plutôt que sur des serveurs cloud distants, sont prometteurs pour la construction de systèmes IoT plus évolutifs et à faible latence.

5.4. Interopérabilité et Normes (Interoperability and Standardization) : Compatibilité entre Différents Appareils et Plateformes

Malgré la croissance rapide de l'IoT, assurer une communication fluide entre les appareils reste un défi majeur. Le manque de normes unifiées dans l'IoT est un obstacle majeur à l'adoption à grande échelle, ce qui conduit à un écosystème fragmenté qui augmente les coûts et limite l'innovation, et nécessite des efforts de collaboration à l'échelle de l'industrie. Les appareils de différents fournisseurs peuvent ne pas prendre en charge les mêmes normes de communication et cela entraîne des difficultés d'intégration. Cette fragmentation oblige les entreprises à s'appuyer sur des solutions propriétaires , ce qui augmente les coûts et limite l'évolutivité. Les solutions proposées, telles que les protocoles unifiés et les passerelles IoT , sont des tentatives pour surmonter ce défi. Cela indique que l'interopérabilité n'est pas seulement une caractéristique technique, mais une condition préalable essentielle pour réaliser le plein potentiel de l'IoT, ce qui nécessite une large collaboration entre les acteurs de l'industrie pour établir des normes communes.

- **Défis :** Les dispositifs IoT utilisent différents protocoles de communication (MQTT, CoAP, HTTP, Zigbee, LoRaWAN), ce qui entraîne des problèmes de compatibilité. Le manque de normes universellement acceptées dans l'industrie IoT entraîne des difficultés d'intégration.
- **Solutions :** Adopter des protocoles standardisés, mettre en œuvre des intégrations basées sur des API, utiliser le calcul en périphérie et assurer la conformité en matière de sécurité. Les passerelles IoT (IoT Gateways) servent d'intermédiaires pour traduire les données entre des protocoles incompatibles.

5.5. Latence (Latency) : Importance du Traitement en Temps Réel

La latence est le délai entre l'envoi et la réception ou le traitement des données. Dans de nombreuses applications IoT, en particulier celles qui nécessitent des réponses immédiates, comme les véhicules autonomes ou les systèmes industriels, la latence doit être très faible. La latence n'est pas seulement une mesure de performance, mais un facteur crucial pour déterminer la faisabilité et la sécurité des applications IoT critiques, ce qui pousse vers des architectures décentralisées et un traitement intelligent en périphérie. Des applications telles que les véhicules autonomes, les soins de santé et l'automatisation industrielle nécessitent des décisions immédiates et un traitement en temps réel. Tout délai (latence) peut avoir des conséquences désastreuses, comme des accidents de voiture ou des erreurs d'usine. Cela montre que la latence est un défi crucial pour la sécurité et les fonctionnalités. Les solutions telles que le calcul en périphérie ne sont pas de simples améliorations, mais sont essentielles pour permettre ces applications critiques, ce qui souligne que l'architecture IoT doit évoluer pour répondre à ces exigences strictes.

- **Défis :** Le transfert de grandes quantités de données IoT vers le cloud pour traitement peut entraîner une latence inacceptable.
- **Solutions :** Le calcul en périphérie réduit la latence en traitant les données localement, près de leur source.

5.6. Défis Réglementaires et Juridiques (Regulatory and Legal Challenges)

L'IoT est confronté à des défis réglementaires et juridiques en raison de sa nature complexe et évolutive. Les défis réglementaires et juridiques sont un facteur crucial dans la formation de l'avenir de l'IoT, car ils peuvent entraver ou accélérer l'innovation en fonction de la capacité des gouvernements et des organismes de réglementation à suivre le rythme des développements technologiques. Avec l'augmentation du nombre de dispositifs IoT et leur impact sur la vie quotidienne, la protection des données et de la vie privée devient de plus en plus importante. Le manque de réglementation peut entraîner une méfiance et des menaces de sécurité. Cela indique que l'innovation technique seule ne suffit pas ; elle doit être accompagnée d'un cadre juridique et réglementaire solide qui garantit la protection des utilisateurs et renforce la confiance dans la technologie. Ce défi nécessite un dialogue continu entre les développeurs et les organismes de réglementation pour créer un environnement qui permette une croissance responsable de l'IoT.

- **Défis :** L'absence d'un cadre réglementaire et juridique unifié pour l'IoT peut entraîner confusion et incertitude. Les problèmes de confidentialité et de sécurité nécessitent des réglementations strictes.

6. Orientations Futures

L'Internet des Objets et les systèmes embarqués se dirigent vers un avenir caractérisé par une intégration, une intelligence et une durabilité accrues, stimulés par les innovations dans les domaines de l'informatique et des communications. Ces orientations promettent d'apporter des transformations profondes dans la façon dont nous interagissons avec le monde qui nous entoure.

6.1. Intégration Croissante de l'Intelligence Artificielle et de l'Edge Computing

L'intégration de l'Intelligence Artificielle avec l'Internet des Objets (AIoT) est en augmentation, créant des systèmes intelligents capables de prendre des décisions autonomes et d'effectuer des analyses améliorées. L'intégration profonde entre l'IA et le calcul en périphérie dans l'IoT conduira à l'émergence d'une nouvelle génération d'appareils intelligents autonomes qui ne se

contentent pas de collecter des données, mais les comprennent et agissent en conséquence de manière indépendante, ouvrant la voie à des applications plus complexes et autonomes. L'IoT collecte les données, l'IA les analyse, et le calcul en périphérie les traite localement. L'intégration croissante signifie que ces composants ne fonctionnent plus séparément, mais s'intègrent pour former des systèmes intelligents capables de prendre des décisions autonomes. Cette transition d'appareils connectés à des appareils intelligents autonomes conduira à des applications plus complexes et autonomes, telles que des machines auto-diagnostiques et des réseaux intelligents respectueux de l'environnement. Cela représente un saut qualitatif dans l'automatisation et le contrôle, où l'appareil lui-même devient une source d'intelligence.

- **IA Embarquée :** Permettra aux appareils d'analyser les données localement, de prendre des décisions éclairées en temps réel et d'apprendre des modèles pour améliorer les performances.
- **Edge Computing :** Deviendra plus intelligent et plus économe en énergie, avec des capacités de traitement basées sur l'IA.
- **Avantages :** Automatisation plus intelligente, décisions plus rapides, efficacité accrue, personnalisation et sécurité améliorées.

6.2. Évolution des Réseaux 6G et leur Impact sur l'IoT

La 6G vise à répondre aux exigences IoT plus strictes par rapport à la 5G. La 6G n'est pas seulement une mise à niveau de la 5G, mais une révolution dans la connectivité qui éliminera les limitations actuelles de l'IoT, permettant la réalisation d'une vision d'un monde cyber-physique transparent où toutes les choses sont connectées intelligemment et avec une fiabilité ultra-élevée. La 5G a apporté des améliorations significatives, mais la 6G vise à répondre à des exigences plus strictes pour l'IoT. Cela signifie que la 6G ira au-delà de la simple vitesse et de la latence pour offrir une connectivité massive, une fiabilité ultra-élevée et une couverture étendue. Cette évolution permettra des applications IoT qui ne sont actuellement pas réalisables, telles que les dispositifs implantés ou les communications sous-marines. Cela indique que la 6G est un catalyseur essentiel pour la prochaine génération d'IoT, ouvrant des perspectives sans précédent d'interconnexion et d'intelligence.

- **Objectifs :** Débits de données plus élevés, latence plus faible, durabilité accrue avec une consommation d'énergie réduite, et connectivité plus sécurisée et fiable.
- **Exigences que la 6G satisfera :** Connectivité massive (prise en charge d'un plus grand nombre de dispositifs IoT), communications IoT ultra-fiables et à faible latence, protocoles de communication améliorés, couverture réseau IoT étendue et dispositifs IoT intelligents.
- **Orientations de recherche :** Intelligence distribuée (comme l'apprentissage fédéré et l'apprentissage par renforcement multi-agents) et IoT par satellite.

6.3. Rôle de l'Informatique Quantique (Quantum Computing) dans le Renforcement de la Sécurité et du Traitement

L'informatique quantique recèle un potentiel considérable pour améliorer les processus de développement logiciel, renforcer les performances du système et permettre des applications innovantes. Bien que l'informatique quantique en soit encore à ses débuts pour les systèmes embarqués, son potentiel à révolutionner la sécurité et le traitement suggère un avenir où les dispositifs IoT pourront relever des défis complexes de cybersécurité et analyser d'énormes quantités

de données avec une efficacité sans précédent. L'IoT est confronté à de graves défis de sécurité et nécessite le traitement d'énormes quantités de données. L'informatique quantique promet une sécurité renforcée et un traitement parallèle d'énormes quantités de données. Cela indique que l'informatique quantique pourrait être la solution ultime à ces défis. Bien que l'intégration directe soit lointaine , la recherche sur les algorithmes inspirés du quantique pourrait conduire à des améliorations immédiates de l'efficacité des systèmes embarqués existants, ouvrant la voie à un avenir plus sûr et plus intelligent pour l'IoT.

- Avantages potentiels :
- **Amélioration des algorithmes d'optimisation :** Résoudre des problèmes complexes plus rapidement.
- **Simulation et tests plus rapides :** Réduire les cycles de développement et le temps de mise sur le marché.
- **Sécurité renforcée :** Améliorer les techniques de chiffrement, offrant des niveaux de sécurité plus élevés pour les systèmes embarqués contre les menaces cybernétiques potentielles. Les algorithmes de chiffrement quantique peuvent offrir un chiffrement plus fort.
- **Traitement parallèle :** Traiter d'énormes quantités de données en parallèle, permettant l'analyse et la prise de décision en temps réel.
- **Amélioration de l'apprentissage automatique :** Renforcer les performances des algorithmes d'apprentissage automatique utilisés dans les systèmes embarqués.
- **Défis et perspectives :** Malgré le potentiel, l'intégration directe d'ordinateurs quantiques complets dans les systèmes embarqués quotidiens est encore lointaine de plusieurs décennies. Cependant, des algorithmes classiques inspirés des principes quantiques pourraient apparaître et fonctionner plus efficacement sur le matériel embarqué traditionnel.

6.4. Jumeaux Numériques (Digital Twins) et Maintenance Prédictive

Les jumeaux numériques sont des répliques virtuelles d'entités ou de processus physiques, mises à jour en permanence avec des données en temps réel provenant de capteurs embarqués. Les jumeaux numériques, alimentés par les données IoT et les analyses d'IA, transformeront la maintenance de réactive en proactive, ce qui entraînera des améliorations radicales de l'efficacité opérationnelle et de la durée de vie des actifs. La maintenance prédictive est une application clé de l'IoT dans l'industrie. Elle implique la collecte et l'analyse de données en temps réel pour prédire les pannes. Les jumeaux numériques vont plus loin en créant un modèle virtuel vivant de l'actif physique. Ce modèle virtuel, alimenté par les données IoT, peut simuler le comportement de l'actif et prédire les pannes avec une plus grande précision, permettant une maintenance avant que le problème ne survienne et augmentant la durée de vie de l'appareil. Cela réduit les temps d'arrêt et les coûts et améliore considérablement l'efficacité.

- **Maintenance prédictive :** C'est l'une des applications principales des jumeaux numériques dans l'Industrie 4.0. Les systèmes IoT embarqués collectent des données en temps réel des machines d'usine, puis les analysent à l'aide d'algorithmes d'apprentissage automatique pour prédire le moment où une maintenance est nécessaire. Cela prévient les arrêts imprévus et réduit les coûts.

6.5. Durabilité et Solutions Écologiques dans la Conception et le Déploiement des Systèmes IoT

Les villes intelligentes recherchent de plus en plus de nouvelles méthodes efficaces pour réduire la consommation de ressources, diminuer l'empreinte carbone et améliorer les opérations urbaines. La durabilité n'est pas seulement une caractéristique supplémentaire de l'IoT, mais un moteur essentiel de l'innovation dans la conception et le déploiement des systèmes, où les solutions futures visent à atteindre une efficacité environnementale et économique intégrée. Au début, l'accent était mis sur l'efficacité et la réduction des coûts. Avec la prise de conscience environnementale croissante, la durabilité et la réduction de l'empreinte carbone sont devenues des objectifs majeurs. L'IoT y contribue en améliorant l'utilisation des ressources , en réduisant le gaspillage et en intégrant les énergies renouvelables. Cela indique que la conception des futurs systèmes IoT ne se concentrera pas seulement sur les performances fonctionnelles, mais inclura également des considérations environnementales profondes, faisant de la durabilité une partie intégrante du cycle de vie du produit et de l'application.

- **Objectifs :** Réduire l'utilisation des ressources, diminuer les émissions de carbone , encourager l'utilisation des énergies renouvelables , améliorer la qualité de l'air.
- Applications :
- **Éclairage public intelligent :** Utilise des capteurs et des modules sans fil pour contrôler l'utilisation des ressources, ce qui réduit considérablement la consommation d'énergie.
- **Gestion des déchets :** Les solutions IoT améliorent la gestion des déchets grâce à la surveillance en temps réel et à l'analyse des données, ce qui réduit les trajets inutiles des camions poubelles.
- **Agriculture intelligente :** Les systèmes IoT embarqués réduisent les émissions de carbone de 38,6% par rapport à l'agriculture traditionnelle.
- **Véhicules connectés :** Aident à réduire la quantité d'équipements installés dans l'infrastructure, la consommation d'énergie et les opérations de maintenance, contribuant ainsi à un système de transport routier plus respectueux de l'environnement.

7. Conclusion

En conclusion, le premier chapitre met en lumière l'importance cruciale de l'intégration entre l'Internet des Objets (IoT) et les systèmes embarqués, qui ensemble constituent une avancée technologique majeure transformant notre manière d'interagir avec notre environnement. Les systèmes embarqués fournissent l'intelligence et les capacités opérationnelles locales des dispositifs connectés, tandis que l'IoT offre l'infrastructure réseau permettant la communication et l'échange massif de données. Ce chapitre a présenté les définitions clés, les architectures multicouches caractéristiques, ainsi que les technologies fondamentales supports à ces systèmes. Ensemble, ces éléments forment un écosystème intelligent capable d'influencer de nombreux secteurs d'activité et de préparer le terrain pour des innovations et applications futures significatives.

De plus, ce chapitre souligne que malgré les avancées prometteuses des technologies IoT et des systèmes embarqués, leur déploiement à grande échelle est confronté à des défis majeurs. Ceux-ci incluent la sécurisation des données et la protection de la vie privée, la gestion efficace de la consommation énergétique, la nécessité d'assurer l'évolutivité pour un nombre croissant d'appareils et de données, ainsi que l'interopérabilité entre divers systèmes. Par ailleurs, la réduction de la latence pour un traitement en temps réel et la mise en place de cadres réglementaires et juridiques clairs sont essentielles pour garantir une adoption durable et sécurisée. Ces défis constituent autant d'opportunités pour enrichir la recherche et développer des solutions innovantes afin de maximiser l'impact positif de ces technologies sur la société moderne.

Chapitre 2 : Étude des composants matériels du système IoT réalisé

2. Introduction

Ce chapitre est consacré à l'étude des composants matériels constituant le système proposé. L'architecture matérielle repose principalement sur le microcontrôleur **ESP32**, choisi pour ses performances élevées, sa faible consommation énergétique et ses capacités de communication intégrées. Ces caractéristiques en font une plateforme particulièrement adaptée aux systèmes embarqués modernes, notamment dans les applications de mesure, de contrôle et de communication.

L'objectif de ce chapitre est de présenter de manière détaillée les différents éléments matériels du système, ainsi que leur rôle respectif dans son fonctionnement global. Une attention particulière est portée au microcontrôleur ESP32, qui assure le traitement des données, la gestion des entrées/sorties et la communication avec les périphériques externes. Les autres composants matériels, tels que les capteurs, les modules de communication et les dispositifs d'alimentation, sont également décrits afin de mettre en évidence leur intégration au sein de l'architecture globale.

Cette étude permet ainsi de justifier les choix technologiques adoptés et de fournir une compréhension claire du fonctionnement matériel du système, constituant une base essentielle pour les chapitres suivants consacrés au développement logiciel et aux expérimentations.

2. Description détaillée des composants électroniques

2.1. Microcontrôleur ESP32

2.1.1. Fonction principale

L'ESP32 est un module MCU générique puissant intégrant le Wi-Fi, le Bluetooth et le Bluetooth Low Energy (BLE). Il est conçu pour une large gamme d'applications, allant des réseaux de capteurs à faible consommation aux tâches plus exigeantes comme l'encodage vocal ou le streaming musical. Dans le cadre de ce projet, l'ESP32 sert de cerveau central au système, gérant la lecture des capteurs, le traitement des données, la communication réseau et le contrôle des actionneurs. Sa polyvalence en fait un choix idéal pour les dispositifs IoT nécessitant une connectivité flexible et une capacité de traitement embarquée.

2.1.2. Caractéristiques techniques importantes

Au cœur du module ESP32-WROOM-32 se trouve la puce ESP32-D0WDQ6, dotée de deux microprocesseurs Xtensa® 32 bits LX6. La fréquence d'horloge du CPU est ajustable de 80 MHz à 240 MHz, offrant une flexibilité pour optimiser la performance ou la consommation d'énergie. Un coprocesseur basse consommation permet de surveiller les périphériques en continu même lorsque le CPU principal est éteint, une caractéristique essentielle pour les applications alimentées par batterie.

En termes de mémoire, l'ESP32 intègre 520 KB de SRAM sur puce, 448 KB de ROM pour le démarrage et les fonctions de base, et 4 MB de flash SPI intégrée pour le stockage du programme. Pour la connectivité sans fil, il supporte le Wi-Fi 802.11 b/g/n (avec des débits allant jusqu'à 150 Mbps) et le Bluetooth v4.2 BR/EDR et BLE. La portée physique est assurée par une puissance de sortie de 20 dBm à l'antenne.

L'ESP32 intègre un riche ensemble de périphériques, y compris des capteurs tactiles capacitifs, des capteurs Hall, des interfaces SD card, UART, SPI, I2S, I2C, PWM (LED et moteur), IR, un compteur d'impulsions, 34 GPIO programmables, des ADC et des DAC. La tension de fonctionnement recommandée est de 3.0 V à 3.6 V, avec un courant de fonctionnement moyen de 80

mA. Le courant de veille profonde est inférieur à 5 µA, ce qui le rend particulièrement adapté aux applications alimentées par batterie et aux appareils électroniques portables. La température de fonctionnement recommandée s'étend de –40 °C à +85 °C. Le système d'exploitation choisi pour l'ESP32 est FreeRTOS avec LwIP, et il intègre des fonctionnalités de sécurité comme TLS 1.2 avec accélération matérielle, le démarrage sécurisé et le chiffrement flash.

Le tableau récapitule les principales caractéristiques techniques de l'ESP32-WROOM-32. Ce tableau condense les informations techniques cruciales de l'ESP32 provenant de plusieurs sources. Il permet une vue d'ensemble rapide et comparative des capacités du microcontrôleur, essentielle pour les ingénieurs et chercheurs. Sa présence facilite la référence rapide aux spécifications clés sans avoir à parcourir de longs textes, améliorant la lisibilité et l'accessibilité des données techniques.

Tableau 4: Caractéristiques techniques principales de l'ESP32-WROOM-32

Caractéristique	Spécification
CPU	Dual-core Xtensa® 32-bit LX6
Fréquence CPU	80 MHz à 240 MHz (ajustable)
SRAM	520 KB (sur puce)
Flash	4 MB (intégrée)
Wi-Fi	802.11 b/g/n (jusqu'à 150 Mbps)
Bluetooth	v4.2 BR/EDR et BLE
Tension (Op.)	3.0 V ~ 3.6 V
Courant (Moy.)	80 mA
Courant (Veille)	< 5 µA
Périphériques	SD, UART, SPI, I2C, PWM, GPIO, ADC, DAC, etc.

2.1.3. Schéma de câblage ou d'utilisation typique

Afin de mieux comprendre les capacités de connexion et les fonctions des différentes broches du module ESP32, la figure suivante (Figure 12) présente le schéma de brochage typique du module. Ce schéma est essentiel pour le câblage dans divers projets.

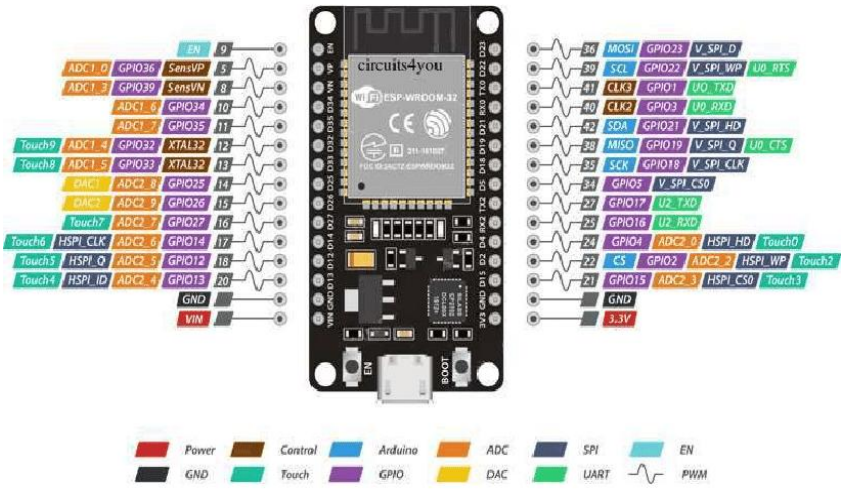


Figure 12: Schéma de brochage typique de l'ESP32 (ESP-WROOM-32 Pinout)

Ce schéma illustrera les connexions d'alimentation (3.3V, GND), les broches de communication série (TX, RX pour le flashage et le débogage), et quelques broches GPIO typiquement utilisées pour les entrées/sorties numériques. Il mettra en évidence la facilité d'utilisation du module ESP32-WROOM-32 sur une platine d'expérimentation, soulignant sa conception conviviale pour le prototypage rapide.

2.1.4. Justification du choix

Le choix de l'ESP32 est impératif en raison de sa double capacité de communication intégrée, Wi-Fi et Bluetooth Low Energy (BLE), qui est fondamentale pour la conception d'un dispositif IoT de mesure de température à distance. Le Wi-Fi permet d'établir une connexion directe et robuste au cloud (ou à un serveur central) via le réseau local, assurant ainsi la transmission continue des données de température capturées vers l'utilisateur final ou une base de données pour l'analyse à longue portée. Parallèlement, le BLE est crucial pour la configuration initiale du capteur, les diagnostics locaux, ou l'interaction de proximité avec une application mobile, tout en minimisant la consommation énergétique de la batterie dans le contexte d'une application alimentée par batterie.

Ses modes de faible consommation, notamment le courant de veille inférieur à 5 µA, sont cruciaux pour les applications alimentées par batterie, garantissant une autonomie prolongée du capteur de température. La présence de deux cœurs CPU et d'une fréquence d'horloge ajustable de 80 MHz à 240 MHz offre une puissance de traitement suffisante pour la lecture des capteurs, la gestion de l'affichage, la communication réseau et d'éventuelles logiques de contrôle plus complexes.

Le riche ensemble de périphériques (GPIO, ADC, I2C, SPI, UART) assure une grande flexibilité pour l'interfaçage avec tous les composants prévus (capteur D20B18, OLED, boutons, LEDs, buzzer) sans nécessiter de composants additionnels complexes. La maturité de son écosystème de développement (Arduino IDE, ESP-IDF, MicroPython) et la disponibilité de nombreuses ressources et bibliothèques simplifient grandement le prototypage et le développement logiciel.

La flexibilité des GPIO de l'ESP32, y compris les broches d'entrée uniquement (GPIO34, 35, 36, 39) et les broches PWM , permet une optimisation de l'affectation des broches pour les différents capteurs et actionneurs, réduisant le besoin de multiplexage externe et simplifiant la conception du

circuit. Pour un projet avec plusieurs entrées (boutons, capteur D20B18) et sorties (LEDs, OLED, buzzer), la disponibilité de nombreuses broches GPIO (34 programmables) sur l'ESP32 est un avantage majeur. Cela permet d'attribuer une broche dédiée à chaque composant, simplifiant le câblage et la programmation. L'identification des broches d'entrée uniquement est cruciale pour éviter les erreurs de configuration logicielle qui pourraient endommager les composants ou entraîner un comportement inattendu. La capacité PWM est essentielle pour le buzzer passif et potentiellement pour le contrôle de la luminosité des LEDs.

La sélection de l'ESP32, avec ses capacités avancées de gestion de l'énergie et son coprocesseur basse consommation , n'est pas seulement un choix technique, mais aussi une décision stratégique pour la viabilité à long terme d'un produit IoT. Pour un dispositif de mesure de température à distance, l'autonomie est souvent une contrainte majeure. Le coprocesseur basse consommation permet de surveiller les capteurs sans réveiller le CPU principal, réduisant drastiquement la consommation en mode veille profonde (<5µA). Cela se traduit par des cycles de vie de batterie plus longs, réduisant les coûts de maintenance et les perturbations pour l'utilisateur, ce qui est un avantage concurrentiel significatif dans le marché de l'IoT. Le support de FreeRTOS et LwIP facilite l'implémentation de ces modes de gestion de l'énergie.

2.2. Capteur de température (D18B20S)

2.2.1. Fonction principale :

Le capteur DS18B20 est le composant principal du dispositif, dédié au calcul et au suivi précis de la température ambiante. Sa fonction est de convertir une grandeur physique (la température) en un signal numérique. Il ne convertit pas la température en une valeur de tension, mais plutôt en une valeur numérique directement exploitable par le microcontrôleur. Cette conversion interne et la transmission numérique fiable via l'interface 1-Wire en font un choix optimal pour des applications nécessitant une mesure de température directe et précise.

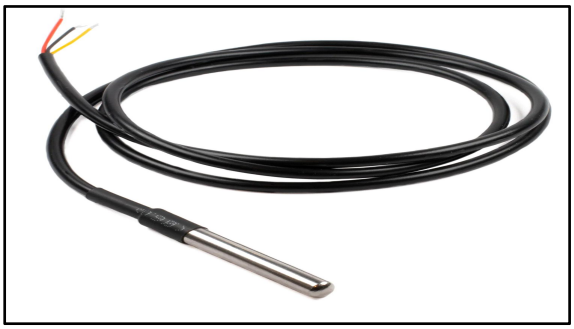


Figure 13: Capteur de température numérique étanche (sonde) de type DS18B20

2.2.2. Caractéristiques techniques importantes

Le capteur DS18B20 fonctionne avec une tension de 3 V à 5,5 V. Pour la mesure de température, il offre une résolution de conversion programmable de 9 à 12 bits. Cette résolution influence directement la granularité de la mesure et le temps de conversion :

- **Résolution 12 bits** : 0,0625 °C (temps de conversion maximal : 750 ms).
- **Résolution 11 bits** : 0,125 °C (375 ms).

- **Résolution 10 bits** : 0,25 °C (187,5 ms).
- **Résolution 9 bits** : 0,5 °C (93,75 ms).

Le capteur offre une précision de ±0,5 °C sur sa plage de fonctionnement principale, comprise entre −10 °C et +85 °C. Sa plage de mesure est étendue et s'étend de −55 °C à +125 °C.

L'interface de communication du capteur est de type **numérique à fil unique (1-Wire)**. Cette technologie présente un avantage significatif pour l'application envisagée, car elle permet de connecter plusieurs capteurs sur un même bus de données, réduisant ainsi le câblage. De plus, le capteur prend en charge l'alimentation parasitaire (*parasite power*), ce qui contribue à une économie supplémentaire de fils.

Par ailleurs, le DS18B20 consomme un courant très faible en mode veille, un aspect essentiel pour une gestion optimisée de l'énergie. Il nécessite l'utilisation d'une résistance de *pull-up* de **4,7 kΩ** sur la ligne de données afin d'assurer une communication fiable et stable.

Le **Tableau 5** récapitule les principales spécifications du capteur DS18B20, permettant une évaluation rapide de ses performances dans le cadre du projet. Ce tableau facilite également la comparaison avec d'autres capteurs potentiels et justifie son adéquation aux exigences de précision du système de mesure de la température. La mention de la tension de fonctionnement et de la faible consommation en veille est enfin déterminante pour le dimensionnement et la gestion de l'alimentation du système.

Schéma de câblage ou d'utilisation typique

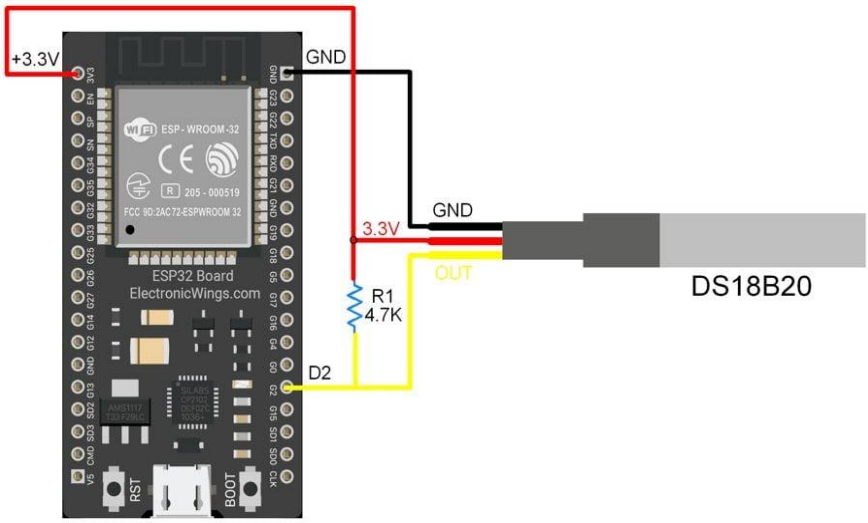


Figure 14 : – Connexion DS18B20 à l'ESP32"

2.3. Afficheur OLED (SSD1306, I2C)

2.3.1. Fonction principale

L'afficheur OLED permet de visualiser des informations textuelles et graphiques. Dans ce projet, il sera utilisé pour afficher les valeurs de température et d'humidité mesurées par le capteur D20B18, ainsi que l'état du système (par exemple, connexion Wi-Fi, état de la batterie). Sa capacité à afficher des informations claires et concises est essentielle pour l'interaction utilisateur locale.

2.3.2. Caractéristiques techniques importantes

L'afficheur OLED sélectionné offre une haute résolution de 128x64 pixels. Il utilise la technologie OLED (Organic Light Emitting Diode), où chaque pixel émet sa propre lumière, offrant un contraste élevé et un angle de vision large de 160 degrés. Contrairement aux écrans LCD, il ne nécessite pas de rétroéclairage, ce qui contribue à sa finesse et à son efficacité.

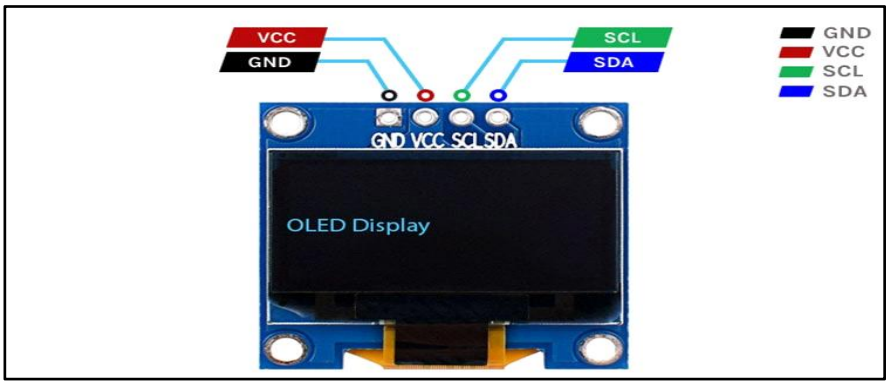


Figure 15 : Module d'affichage OLED ET brochage de connexion

Source : https://passionelectronique.fr/

Sa consommation d'énergie est très faible, seulement 0.06W en utilisation normale ou environ 20 mA (selon les pixels actifs). Il est compatible avec une tension d'alimentation de 3V à 5V , fonctionnant ainsi très bien avec l'ESP32 qui opère à 3.3V. L'afficheur peut fonctionner dans une large plage de températures, de -30°C à 70°C. Ses dimensions compactes sont de L27.8 x W27.3 x H4.3 MM. Le circuit intégré de pilotage est le SSD1306, et l'interface de communication est I2C , avec une adresse I2C typique de 0x3C.

Le tableau 2.3 met en évidence les avantages clés de l'OLED pour les applications IoT, notamment sa faible consommation et son contraste élevé. Il fournit les informations nécessaires pour l'intégration matérielle et logicielle, en particulier l'interface I2C et la tension de fonctionnement, qui sont compatibles avec l'ESP32.

Tableau 5: Caractéristiques techniques de l'afficheur OLED (SSD1306)

Caractéristique	Spécification
Résolution	128x64 pixels
Technologie	OLED
Angle de vision	160 degrés
Consommation	0.06W (normal) / ~20mA
Tension (Op.)	3V - 5V
Driver IC	SSD1306
Interface	I2C (adresse typ. 0x3C)

2.3.3. Présente Schéma de câblage ou d'utilisation typique

- **Schéma de câblage de l'afficheur OLED avec ESP32 (figure 15) :**

Ce schéma montrera les connexions I2C : la broche SDA de l'OLED à la broche SDA de l'ESP32 (GPIO21 par défaut), la broche SCL de l'OLED à la broche SCL de l'ESP32 (GPIO22 par défaut). Les broches VCC et GND de l'OLED seront connectées respectivement au 3.3V et au GND de l'ESP32. Ces connexions simples soulignent l'efficacité de l'interface I2C pour les systèmes embarqués.

2.4. Boutons poussoirs

2.4.1. Fonction principale

Les boutons poussoirs sont des composants d'entrée simples qui permettent à l'utilisateur d'interagir avec le système. Dans ce projet, ils pourraient être utilisés pour naviguer dans les menus de l'afficheur OLED, déclencher une mesure manuelle, ou réinitialiser le système, offrant une interface utilisateur directe et tangible.



Figure 16: Structure interne et principe de connexion des bornes d'un interrupteur tactile (Tactile Switch)

2.4.2. Interfaçage des boutons poussoirs avec l'ESP32 : Configuration logicielle et gestion du rebond.

Les boutons poussoirs sont généralement des interrupteurs momentanés, fonctionnant en mode normalement ouvert (NO) ou normalement fermé (NC). Ils sont interfacés en étant connectés à des broches GPIO de l'ESP32. Pour éviter les états flottants des broches d'entrée, les boutons nécessitent une résistance de pull-up ou pull-down. L'ESP32 est avantageux car il dispose de résistances de pull-up internes qui peuvent être configurées par logiciel, simplifiant ainsi le câblage externe. Un problème courant avec les contacts mécaniques des boutons est le rebond (chattering), qui peut générer des signaux multiples pour une seule pression, nécessitant une solution de dé-rebond matérielle ou logicielle.

2.5. LEDs (rouge, verte, jaune)

2.5.1. Fonction principale

Les LEDs (Light Emitting Diodes) sont des indicateurs visuels simples qui fournissent un retour d'information sur l'état du système. Dans ce projet, elles pourraient indiquer l'état de la connexion Wi-Fi (verte), une erreur (rouge), ou un état d'attente/avertissement (jaune), offrant une signalisation rapide et intuitive.

La tension directe (Vf) des LEDs varie selon leur couleur. Pour les LEDs rouges et jaunes, elle est typiquement de 1.8V à 2.2V. Pour les LEDs vertes, elle se situe généralement entre 2.0V et 3.0V. Le courant direct (If) recommandé est généralement de 20mA pour une luminosité optimale sans endommager la LED.

Chaque LED doit être connectée en série avec une résistance pour limiter le courant et éviter de l'endommager. La valeur de la résistance est calculée par la loi d'Ohm : R = (Vs - Vf) / If, où Vs est la tension d'alimentation (3.3V pour les GPIO de l'ESP32). La luminosité de la LED est directement proportionnelle au courant qui la traverse. Les LEDs sont interfacées en étant connectées à des broches GPIO de l'ESP32.

2.6. Module d’alimentation (batterie ou alimentation USB)

Le choix du module d’alimentation est une étape essentielle dans la conception d’un système embarqué, en particulier pour les dispositifs IoT qui nécessitent une autonomie énergétique ou une mobilité.

Dans notre projet, deux options principales d'alimentation sont envisagées :

- Batterie Li-ion 3.7V

Nous avons prévu l’utilisation d’une **batterie lithium-ion 3.7V** comme source d’énergie autonome. Ce type de batterie est léger, rechargeable, et présente une bonne capacité énergétique, ce qui en fait un excellent choix pour les applications embarquées mobiles.

Cette batterie sera connectée à un module de charge adapté (par exemple, un module TP4056) afin de permettre une **recharge sécurisée via une entrée 5V** (USB). Cela permet d’assurer une utilisation continue sans devoir remplacer la batterie manuellement.

- Alimentation directe via un chargeur 5V (USB)

Une deuxième option consiste à utiliser un **chargeur 5V** connecté directement au module de charge, avec ou sans batterie.
Deux cas sont possibles :

- Soit la batterie est présente et se recharge automatiquement.
- Soit la batterie est absente, et le système est directement alimenté par le chargeur 5V (mode secteur).

Choix retenu pour les tests

Lors des phases de programmation, de test et de mise au point, nous utiliserons la connexion USB de l'ESP32. Elle servira à la fois :

- pour l'alimentation électrique de la carte,
- et pour la programmation du microcontrôleur via l'environnement Arduino IDE.

Cette méthode est simple, pratique et adaptée à l'étape de développement.

3 Schéma fonctionnel du système:

3.1. Explication du flux de données entre les composants

Le schéma fonctionnel décrira les interactions logiques et le flux d'informations entre les principaux blocs du système. Le **Capteur D20B18** collecte les données de température et d'humidité de l'environnement. Ces données sont numérisées et transmises via une interface single-wire au **Microcontrôleur ESP32**.

L'**ESP32**, en tant qu'unité de traitement centrale , reçoit et traite ces données. Il est également responsable de la gestion des

Boutons poussoirs pour les interactions utilisateur (par exemple, déclenchement de mesures, navigation). Les données traitées (température, humidité, état du système) sont ensuite envoyées à l'

Afficheur OLED via l'interface I2C pour une visualisation locale.

L'ESP32 utilise sa connectivité **Wi-Fi** pour transmettre les données collectées à une plateforme cloud ou un serveur distant, permettant la surveillance à distance. Les

LEDs sont contrôlées par l'ESP32 pour fournir des indicateurs visuels d'état (par exemple, connexion Wi-Fi, alerte). Le

Buzzer est activé par l'ESP32 pour générer des alertes sonores en cas d'événements spécifiques (par exemple, dépassement de seuil de température). Le

Module d'alimentation (batterie ou USB) fournit l'énergie nécessaire à l'ensemble des composants du système.

Le flux de données met en évidence la centralisation du traitement sur l'ESP32, ce qui est caractéristique des systèmes IoT basés sur des microcontrôleurs. L'ESP32 ne se contente pas de collecter des données brutes ; il les traite (par exemple, conversion des lectures du D20B18 en valeurs significatives, formatage pour l'affichage ou la transmission). Cette capacité de "edge computing" réduit la dépendance au cloud pour les décisions immédiates et peut améliorer la réactivité du système, par exemple en déclenchant une alerte locale via le buzzer avant même que les données n'atteignent le serveur.

La conception du flux de données doit anticiper les modes de faible consommation de l'ESP32 pour optimiser l'autonomie. Pour un système alimenté par batterie, l'ESP32 passera probablement une grande partie de son temps en mode veille profonde ou légère. Le flux de données doit être conçu pour que les capteurs puissent être lus périodiquement (par exemple, réveil de l'ESP32 par un minuteur RTC) , les données traitées et transmises, puis le système retourne en veille. Cela implique une planification minutieuse des cycles de mesure et de transmission pour minimiser le temps d'activité du microcontrôleur et maximiser la durée de vie de la batterie.

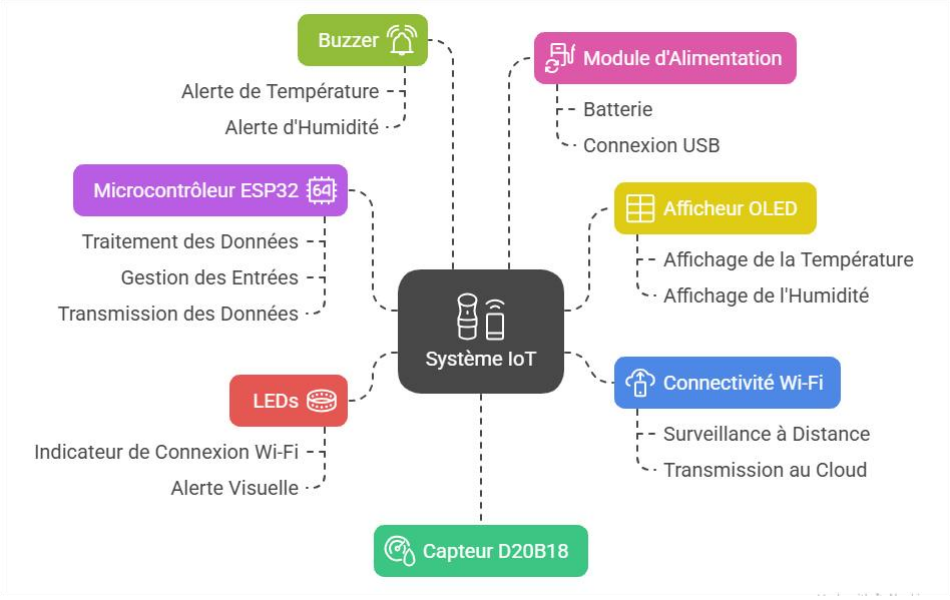


Figure 17:schéma Fonctionnel du système IoT (*Réalisée par les étudiants, via Napkin AI (https://app.napkin.ai)*)

Le système fonctionnel d'un dispositif de mesure de température IoT se compose de plusieurs éléments principaux. Le processus commence par le **capteur de température (DS18B20)**, qui mesure la grandeur physique et la convertit en données numériques. Ces données sont ensuite envoyées à l'**unité de traitement (ESP32)**, qui sert de cerveau au système en lisant, traitant et gérant les informations. Par la suite, le système utilise le **module de communication (Wi-Fi)** intégré à l'ESP32 pour transférer les données sur Internet vers une **plateforme cloud**, où elles sont stockées et analysées. Enfin, les données sont affichées pour l'utilisateur via une **interface utilisateur** (telle qu'une application mobile ou un tableau de bord web), permettant de surveiller la température en temps réel et d'accéder aux données historiques.

3.2 Schema électrique:

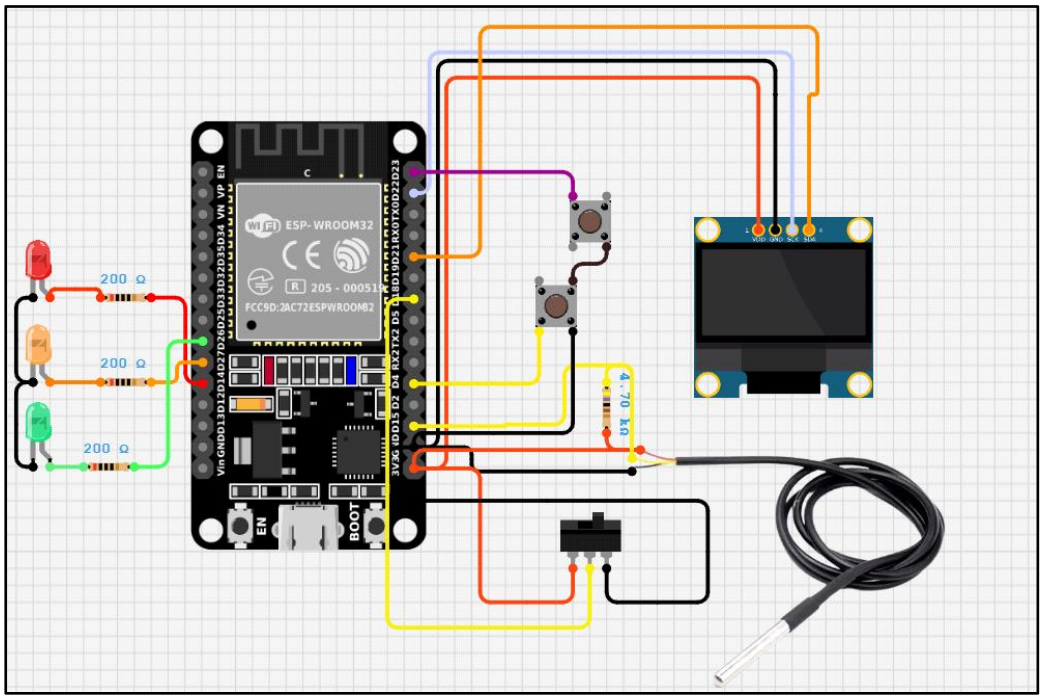


Figure 18: **Schéma de câblage et d'interfaçage de l'ESP32 avec le capteur de température DS18B20 et l'afficheur OLED**

Ce schéma illustre un système électronique complet de surveillance de la température, basé sur une carte de contrôle **ESP32**. Le système utilise un **capteur de température numérique DS18B20** pour mesurer la température ambiante, et les relevés sont affichés sur un **écran OLED** pour fournir des informations visuelles précises. De plus, trois **indicateurs lumineux (LED)** de différentes couleurs (rouge, jaune et vert) sont utilisés pour signaler visuellement l'état du système ou la plage de température. Le schéma intègre également des **boutons de commande et un interrupteur** pour permettre à l'utilisateur d'interagir avec le dispositif ou de contrôler ses fonctions.

Conclusion

En conclusion, cette étude matérielle a permis d'identifier et de justifier le choix des composants essentiels du système IoT dédié à la mesure et à la surveillance intelligente, notamment la carte ESP32, le capteur DHT22 et l'écran OLED, dont la complémentarité garantit fiabilité, précision et faible consommation énergétique. L'architecture matérielle ainsi proposée constitue une base solide pour la phase suivante consacrée à l'étude logicielle et au développement du programme embarqué, assurant la collecte, la transmission et l'exploitation des données environnementales.

CHAPITRE III : REALISATION ET VALIDATION DU SYSTEME

Introduction du chapitre

Ce chapitre représente la phase pratique de notre projet, marquant le passage de l'étude théorique et conceptuelle à la mise en œuvre matérielle et logicielle du système de mesure et de surveillance de la température basé sur les technologies de l'Internet des Objets (IoT).

Il a pour objectif de présenter de manière détaillée les différentes étapes ayant conduit à la réalisation complète du système, depuis le choix des composants électroniques appropriés, tels que le microcontrôleur ESP32 et le capteur DHT22, jusqu'à la conception logicielle, l'assemblage physique et la validation des performances afin de garantir la précision et la stabilité du dispositif.

Dans ce cadre, nous décrirons la méthodologie de travail adoptée, incluant la préparation des circuits et des connexions électriques, la programmation du microcontrôleur, ainsi que la gestion de la communication entre les différents composants. Enfin, nous présenterons les résultats obtenus et leur interprétation, mettant en évidence l'efficacité et la fiabilité du système réalisé.

1. Méthodologie de travail et montage du système

La réalisation du système a suivi une approche systématique, débutant par l'assemblage physique des composants avant de passer à la programmation. L'objectif était de créer une structure modulaire et robuste, facile à mettre en œuvre et à maintenir.

1.1. Role des composant électroniques

- **Microcontrôleur ESP32 :** C'est le cerveau du système. Il lit les données du capteur, gère l'affichage sur l'écran OLED, contrôle les LED de signalisation et, surtout, établit une connexion Wi-Fi pour héberger le serveur web et transmettre les données à distance.
- **Capteur de température DS18B20 :** Ce capteur numérique utilise le protocole One-Wire pour communiquer avec l'ESP32. Il fournit des mesures de température précises et fiables, essentielles à la fonctionnalité principale du système.
- **Afficheur OLED SH1106 (128x64) :** Il permet d'afficher localement les informations vitales en temps réel, telles que la température actuelle, les valeurs minimales et maximales, et un graphique historique, sans nécessiter de connexion à un appareil externe.
- **LED (Rouge, Jaune, Verte) :** Elles servent d'indicateurs visuels pour signaler rapidement l'état de la température : vert pour la température normale, jaune pour un avertissement et rouge pour une température critique.
- **Boutons et interrupteur :** Ils permettent à l'utilisateur d'interagir directement avec le système, en changeant l'écran d'affichage sur l'OLED ou en réinitialisant les statistiques, ainsi que d'activer/désactiver la transmission des données via l'interrupteur.

1.2. Schéma des connexions (Pinout)

L'interfaçage des différents modules avec l'ESP32 a été réalisé selon le schéma de câblage suivant. Ce tableau récapitule les connexions pour chaque composant afin d'assurer une bonne communication et un bon fonctionnement.

Tableau 6: Schéma de câblage et affectation des broches de l'ESP32

Composant	Pin sur l'ESP32	Fonction
Capteur DS18B20	**GPIO 15**	Données de température (bus One-Wire)
LED Rouge	**GPIO 14**	Indicateur de température critique
LED Jaune	**GPIO 27**	Indicateur d'avertissement de température
LED Verte	**GPIO 26**	Indicateur de température normale
Interrupteur	**GPIO 18**	Activation/désactivation de la transmission Wi-Fi
Bouton Suivant	**GPIO 12**	Changement de page sur l'écran OLED
Bouton Reset	**GPIO 13**	Réinitialisation des statistiques de température
Afficheur OLED	**SDA & SCL**	I²C pour l'affichage (Pins par défaut)

1.3. É'tapes de montage

1. **Préparation :** Souder les pins de l'ESP32 et des modules si nécessaire.
2. **Câblage :** Connecter les composants à l'ESP32 en utilisant des fils de connexion (jumpers), en suivant scrupuleusement le schéma de pinout ci-dessus. Le capteur DS18B20 nécessite également une résistance de pull-up de 4.7kΩ sur la ligne de données.
3. **Montage :** Fixer tous les composants sur une plaque d'essai (breadboard) pour le prototypage ou sur un circuit imprimé (PCB) pour la version finale, en s'assurant que les connexions sont stables et sécurisées.
4. **Alimentation :** Connecter l'ensemble à une source d'alimentation stable (via le port USB de l'ESP32 ou une alimentation externe).

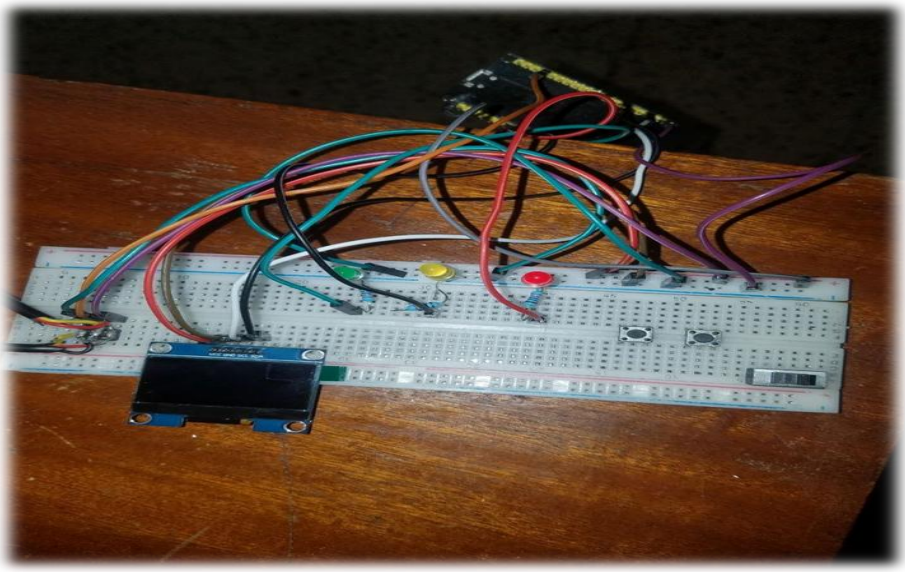


Photo 1: Assemblage complet du système de surveillance de température (Partie matérielle).

Source : Conception et réalisation par les étudiants, Projet de surveillance de température, juillet-2025.

2. Programmation et description logicielle

La logique de notre système a été développée en utilisant l'environnement de développement intégré **Arduino IDE**. Ce choix a été motivé par sa facilité d'utilisation, la richesse de ses bibliothèques et la compatibilité avec l'ESP32.

2.1. Description du code

Le code source du programme est [Voir l'annexe] programme C++ pour l'ESP32, il est conçu pour être à la fois fonctionnel sur l'appareil et interactif via une interface web. Voici une analyse des sections principales du code**.**

2.2. Analyse du code source

Le programme se décompose en plusieurs parties clés : les bibliothèques et les variables, la fonction setup() et la fonction loop().

- **Déclaration des bibliothèques et des variables :**
  - #include <WiFi.h> : Gère les fonctionnalités Wi-Fi de l'ESP32, en mode point d'accès (AP) et station (STA).
  - #include <OneWire.h> et <DallasTemperature.h> : Indispensables pour la communication avec le capteur DS18B20.
  - #include <SH1106Wire.h> : Permet de piloter l'écran OLED pour l'affichage local.
  - Les variables ap_ssid, ap_password, wifi_ssid, wifi_password définissent les paramètres de connexion Wi-Fi.
  - Les constantes pour les broches des composants (ONE_WIRE_BUS, ledRed, etc.) facilitent la gestion du câblage.

- Les variables temperatureC, minTemp, maxTemp, etc., stockent les valeurs de température et les statistiques.

- **La fonction setup() :**
  - Serial.begin(115200); : Initialise la communication série pour le débogage.
  - sensors.begin(); : Démarre la communication avec le capteur DS18B20.
  - display.init(); : Initialise l'écran OLED.
  - pinMode(...) : Configure les broches de l'ESP32 en entrée (INPUT) ou en sortie (OUTPUT) pour les LEDs, les boutons et l'interrupteur.
  - WiFi.mode(WIFI_AP_STA); et WiFi.softAP(...) : Configurent l'ESP32 pour qu'il agisse à la fois comme point d'accès Wi-Fi et comme station connectée à un routeur existant.
  - server.begin(); : Démarre le serveur web sur le port 80.
- **La fonction loop() :**
  - Cette boucle s'exécute continuellement.
  - sensors.requestTemperatures(); et temperatureC = sensors.getTempCByIndex(0); : Ces lignes demandent et lisent la température du capteur.
  - Les instructions if et digitalWrite contrôlent les **LEDs** en fonction de la plage de température mesurée.
  - if (digitalRead(buttonNext) == LOW) : Détecte l'appui sur le bouton Suivant pour passer à la page d'affichage suivante sur l'écran OLED.
  - if (digitalRead(buttonReset) == LOW) : Détecte l'appui sur le bouton Reset pour réinitialiser les statistiques de température.
  - **Les fonctions drawPage0, drawPage1, drawPage2** : Ces fonctions sont responsables de l'affichage sur l'écran OLED.
    - drawPage0 affiche la température actuelle et son état (COOL, WARM, HOT).
    - drawPage1 affiche les statistiques min/max et la moyenne.
    - drawPage2 génère un graphique des 30 dernières lectures de température.
  - **Gestion du serveur web :** Le code utilise server.available() pour écouter les requêtes des clients. Selon la requête, il envoie soit les données de température brutes (GET /temp) soit la page web complète (html). Le contenu de la page HTML est stocké dans la variable html.

2.3. Entrées et Sorties du Système (Inputs/Outputs)

- **Entrées :**
  - **Données du capteur DS18B20 :** La température mesurée en temps réel.
  - **Interrupteur switchPin (GPIO 18) :** L'état de l'interrupteur détermine si les données sont envoyées ou non au serveur web.
  - **Bouton buttonNext (GPIO 12) :** L'appui sur ce bouton est une entrée qui change l'état d'affichage.
  - **Bouton buttonReset (GPIO 13) :** L'appui sur ce bouton est une entrée qui déclenche la réinitialisation des statistiques.
- **Sorties :**
  - **LEDs (GPIO 14, 27, 26) :** L'état de ces LEDs (ON/OFF) est une sortie visuelle.
  - **Afficheur OLED (I²C) :** L'écran affiche les données et les graphiques.
  - **Serveur Web (Port 80) :** L'ESP32 envoie des pages HTML et des données de température (/temp) aux clients connectés.

3. Résultats et interprétation

Cette section présente les résultats des tests et des expériences menés sur notre prototype, ainsi qu'une analyse de son efficacité.

3.1. Conditions de l'expérimentation

Le système a été testé dans un environnement de laboratoire contrôlé. Le capteur DS18B20 a été placé dans une zone à température ambiante, puis exposé à des sources de chaleur pour simuler des variations de température. Les tests de connectivité Wi-Fi ont également été effectués en variant la distance par rapport au routeur pour évaluer la stabilité de la connexion.

3.2. Affichage des résultats

Les données sont présentées en temps réel de deux manières : sur l'écran OLED localement et via une page web accessible à distance.

Affichage sur l'écran OLED : Le système dispose de trois pages d'affichage, qui peuvent être parcourues à l'aide du bouton "Suivant" :

- **Page 1 :** Affiche la température actuelle et l'état du système (Cool, Warm, Hot).
- **Page 2 :** Affiche les valeurs minimales, maximales et la moyenne des lectures.
- **Page 3 :** Affiche un graphique des 30 dernières lectures pour une visualisation de l'historique.

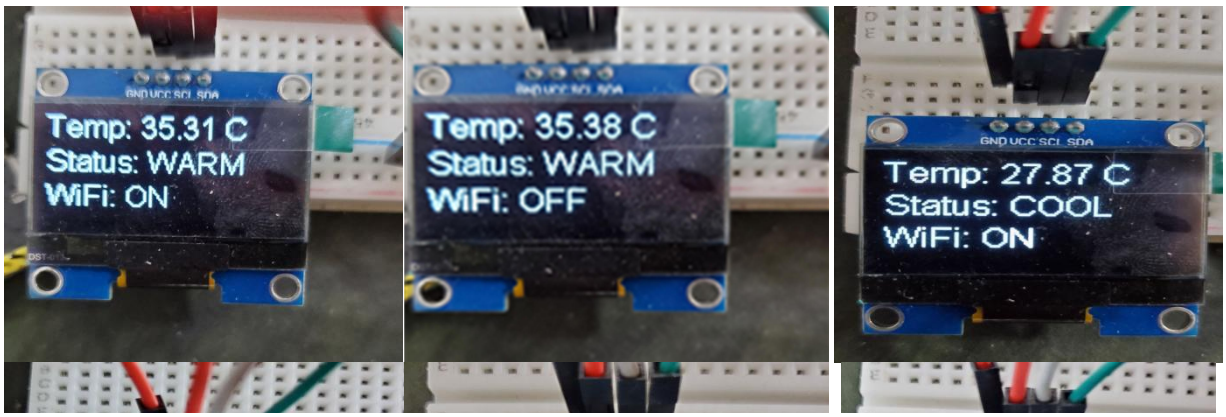
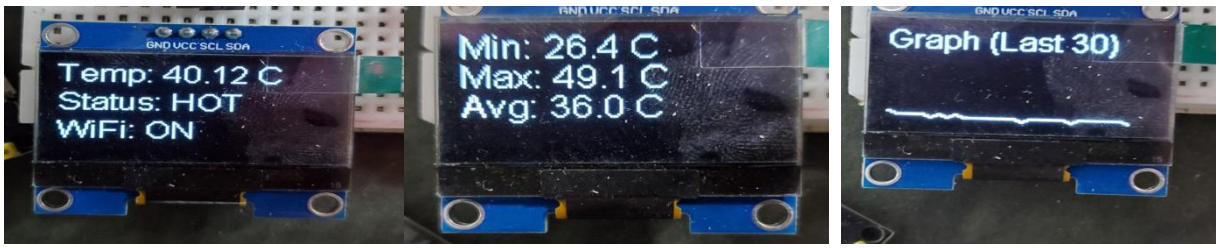


Photo 2 : Interfaces d'affichage locales sur l'écran OLED



Affichage sur la page HTML : En accédant à l'adresse IP de l'ESP32 via un navigateur, une interface utilisateur simple permet de consulter la température en direct. La page web affiche la température actuelle, un indicateur d'état (Cool, Warm, Hot), un indicateur de progression visuel et les valeurs minimales et maximales.

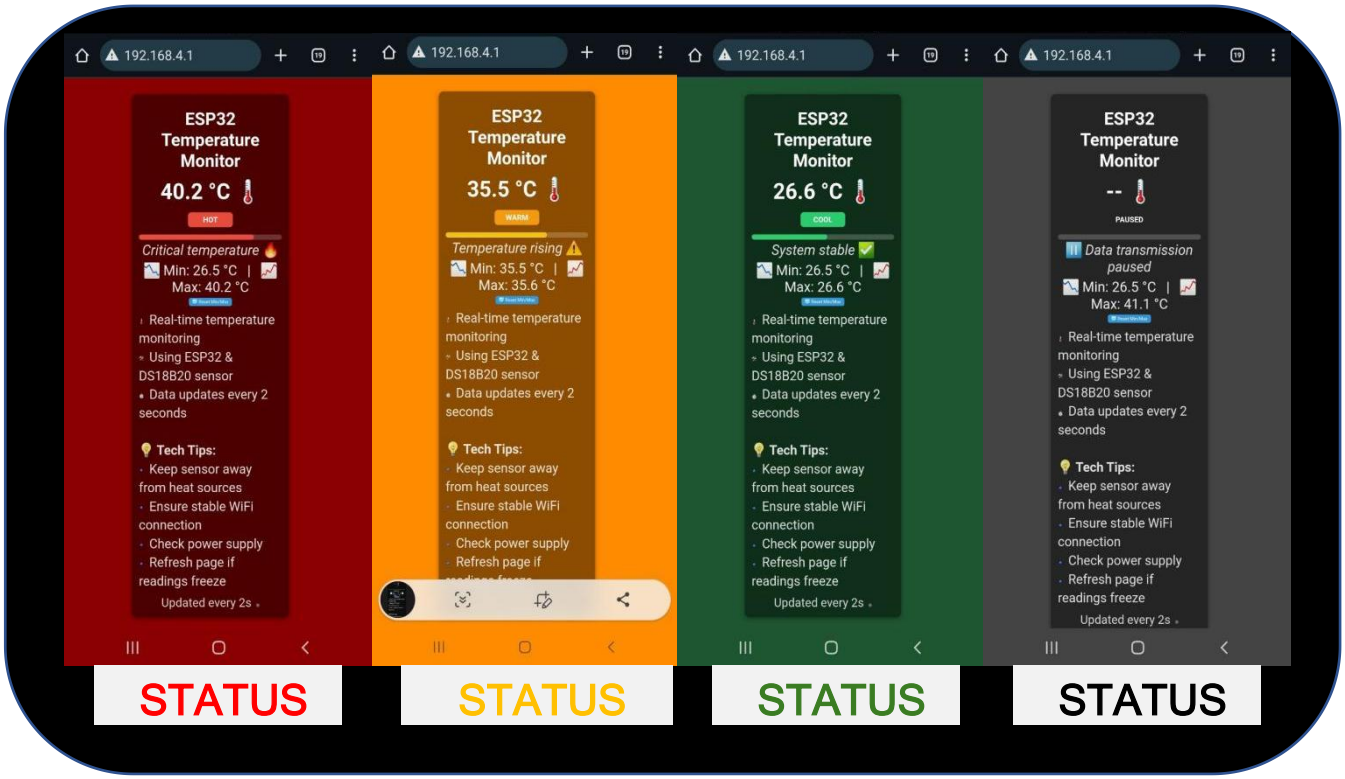


photo 3 Vues des données sur la page d'accueil du serveur Web

Source : Réalisation par les étudiants juillet (2025).

3.3. Analyse des résultats et efficacité

Les tests ont confirmé que notre système fonctionne de manière fiable :

- **Précision :** Les lectures du capteur DS18B20 se sont révélées précises et cohérentes.
- **Connectivité :** Le module Wi-Fi de l'ESP32 a maintenu une connexion stable, permettant la transmission des données en temps réel vers le serveur web intégré.
- **Réactivité :** Les LEDs de signalisation s'allument instantanément lors du franchissement des seuils de température, et la page web est mise à jour toutes les deux secondes, assurant une surveillance efficace.
- **Interface utilisateur :** L'écran OLED et la page web offrent des interfaces claires et conviviales, permettant à l'utilisateur de surveiller les conditions de température sans difficulté.

3.4. Problèmes rencontrés et solutions

Lors des premières phases de montage, un problème a été observé avec un affichage LCD. Une erreur de branchement sur les broches de tension (VCC) et de terre (GND) a provoqué un court-circuit et l'endommagement du module. Ce problème, bien que mineur, souligne l'importance d'une

vérification rigoureuse des connexions avant la mise sous tension. La solution a été simple : remplacer le module défectueux et s'assurer que le câblage respecte strictement le schéma électrique.

Conclusion

Ce chapitre a démontré la faisabilité et la robustesse de notre solution de surveillance de température à distance. En combinant la méthodologie de travail, la programmation soignée et la validation des résultats, nous avons pu construire un prototype fonctionnel qui répond à la problématique initiale. Les performances du système, à la fois en termes de mesure précise et de transmission fiable des données, valident notre approche. Nous pouvons ainsi affirmer que notre solution est efficace pour la surveillance de la température. Le prochain et dernier chapitre, la conclusion générale, résumera l'ensemble de notre travail, fera le bilan des objectifs atteints et proposera des perspectives d'amélioration pour l'avenir.

Conclusion Générale

Notre projet, qui a porté sur la **conception et la réalisation d'un système IoT de mesure et de surveillance de température à distance**, représente un parcours complet, de l'idée théorique à la mise en œuvre pratique. Nous avons réussi à construire une solution technologique tangible qui répond à la problématique de la surveillance environnementale en temps réel, en nous appuyant sur les principes fondamentaux de l'Internet des Objets étudiés au début de ce travail.

Dans le premier chapitre, nous avons posé les bases théoriques en fournissant une vue d'ensemble du concept de l'IoT, de ses composants clés et de ses diverses applications, ce qui nous a permis de comprendre le contexte technologique de notre projet. Ensuite, dans le deuxième chapitre, nous avons abordé l'aspect technique en sélectionnant les composants matériels appropriés, de la **carte de développement ESP32** au **capteur DS18B20**, et en présentant les schémas électroniques qui ont guidé le processus d'assemblage.

Le troisième chapitre a transformé les idées en une réalité concrète. Nous avons détaillé les étapes de l'assemblage physique et de la programmation à l'aide de l'environnement **Arduino IDE**, et nous avons montré comment le code a réussi à lier le matériel et le logiciel. Nous avons également présenté les résultats obtenus, que ce soit via l'écran **OLED** ou le **serveur web intégré**, ce qui a confirmé l'efficacité du système et sa capacité à collecter, transmettre et afficher des données de manière fiable.

Ce projet a atteint tous les objectifs fixés au préalable, de la capacité à mesurer précisément la température, à la transmission sans fil des données, jusqu'à la fourniture d'une interface utilisateur simple et directe. Cependant, le champ d'amélioration est toujours ouvert.

Perspectives Futures

Il est possible de faire évoluer ce projet pour y inclure :

- L'ajout d'un **module de stockage de données externe** pour conserver les lectures à long terme.
- L'intégration d'un **actionneur supplémentaire** (comme un ventilateur ou un chauffage) pour un contrôle automatique de la température.
- Le développement d'une **application mobile dédiée** avec une interface plus avancée.
- L'utilisation d'autres **protocoles de communication** tels que MQTT pour une transmission de données plus efficace.

En conclusion, ce travail représente une contribution modeste aux applications de l'Internet des Objets et ouvre la voie à de futures recherches et innovations dans le domaine des systèmes connectés.

L’annexe

Code c++

```
#include <WiFi.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <SH1106Wire.h>
#include <Wire.h>
// ---------------- Paramètres WiFi ----------------
const char* ap_ssid = "SmartTemp Monitor";
const char* ap_password = "12345678";
const char* wifi_ssid = "TP-LINK_DBF1DB_EXT";
const char* wifi_password = "@46728769@.";
// ---------------- Écran OLED ----------------
SH1106Wire display(0x3C, SDA, SCL);
// ---------------- Capteur de température DS18B20 ----------------
#define ONE_WIRE_BUS 15
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
// ---------------- LEDs ----------------
const int ledRed    = 14;
const int ledYellow = 27;
const int ledGreen  = 26;
// ---------------- Switch de contrôle ----------------
const int switchPin = 18;
bool sendEnabled = true;
// ---------------- Boutons ----------------
const int buttonNext  = 12;
const int buttonReset = 13;
// ---------------- Variables de température ----------------
float temperatureC = 0.0;
float minTemp = 1000.0;
```

```
float maxTemp = -1000.0;
float sumTemp = 0.0;
unsigned long countTemp = 0;
// ---------------- Gestion des pages ----------------
int currentPage = 0;
const int totalPages = 3;
// ---------------- Stockage des 30 dernières mesures ----------------
const int historySize = 30;
float tempHistory[historySize];
int historyIndex = 0;
// ---------------- Serveur Web ----------------
WiFiServer server(80);
// ---------------- Page HTML ----------------
String html = R"rawliteral(
<!DOCTYPE html>
<html>
<head><title>ESP32 Temperature Monitor</title></head>
<body>
<h2>    Smart Temperature Monitor</h2>
<p>Loading...</p>
<p>    Min: -- °C   |       Max: -- °C</p>
<p>    Reset Min/Max</p>
<p>    Surveillance en temps réel avec ESP32 & DS18B20</p>
<p>   Mise à jour toutes les 2 secondes</p>
<p>    Conseils techniques:
<ul>
<li>Éloigner le capteur des sources de chaleur</li>
<li>Assurer une connexion WiFi stable</li>
<li>Vérifier l'alimentation</li>
<li>Actualiser la page si les mesures se bloquent</li>
</ul>
</p>
<p>Mise à jour toutes les 2s</p>
```

```
</body>
</html>
)rawliteral";
// ---------------- Fonctions utilitaires ----------------
void resetStats() {
  minTemp = 1000.0;
  maxTemp = -1000.0;
  sumTemp = 0.0;
  countTemp = 0;
  for (int i = 0; i < historySize; i++)
    tempHistory[i] = 0;
}
void drawPage0() {
  display.drawString(0, 0, "Temp: " + String(temperatureC) + " C");
  String statusText = "Status: ";
  statusText += (temperatureC < 30) ? "COOL" : (temperatureC <= 40) ? "WARM" : "HOT";
  display.drawString(0, 20, statusText);
  display.drawString(0, 40, "WiFi: " + String(sendEnabled ? "ON" : "OFF"));
}
void drawPage1() {
  display.drawString(0, 0, "Min: " + String(minTemp, 1) + " C");
  display.drawString(0, 16, "Max: " + String(maxTemp, 1) + " C");
  float avg = (countTemp > 0) ? (sumTemp / countTemp) : 0;
  display.drawString(0, 32, "Avg: " + String(avg, 1) + " C");
}
void drawPage2() {
  display.drawString(0, 0, "Graph (Last 30)");
  int baseY = 63;           // Bas de l'écran
  int graphHeight = 30;
  float localMin = 1000, localMax = -1000;
  // Trouver les valeurs min et max dans l'historique
  for (int i = 0; i < historySize; i++) {
    if (tempHistory[i] < localMin) localMin = tempHistory[i];
```

```
        if (tempHistory[i] > localMax) localMax = tempHistory[i];
    }
    float range = (localMax - localMin);
    if (range < 1) range = 1; // Éviter la division par zéro
    // Dessiner le graphique
    for (int i = 0; i < historySize - 1; i++) {
        int index1 = (historyIndex + i) % historySize;
        int index2 = (historyIndex + i + 1) % historySize;
        int x1 = i * 4;
        int x2 = (i + 1) * 4;
        int y1 = baseY - ((tempHistory[index1] - localMin) / range) * graphHeight;
        int y2 = baseY - ((tempHistory[index2] - localMin) / range) * graphHeight;
        display.drawLine(x1, y1, x2, y2);
    }
}
// ---------------- Setup initial ----------------
void setup() {
    Serial.begin(115200);
    sensors.begin();
    display.init();
    display.flipScreenVertically();
    display.setFont(ArialMT_Plain_16);
    pinMode(ledRed, OUTPUT);
    pinMode(ledYellow, OUTPUT);
    pinMode(ledGreen, OUTPUT);
    pinMode(switchPin, INPUT);
    pinMode(buttonNext, INPUT_PULLUP);
    pinMode(buttonReset, INPUT_PULLUP);
    WiFi.mode(WIFI_AP_STA);
    WiFi.softAP(ap_ssid, ap_password);
    Serial.print("IP du point d'accès: "); Serial.println(WiFi.softAPIP());
    WiFi.begin(wifi_ssid, wifi_password);
    Serial.print("Connexion au WiFi");
```

```
    int attempts = 0;
    while (WiFi.status() != WL_CONNECTED && attempts < 20) {
        delay(500);
        Serial.print(".");
        attempts++;
    }
    if (WiFi.status() == WL_CONNECTED) {
        Serial.println("\nConnecté au routeur!");
        Serial.print("IP locale: "); Serial.println(WiFi.localIP());
    } else {
        Serial.println("\nÉchec de connexion au routeur. Mode AP uniquement.");
    }
    server.begin();
    resetStats();
}
// ---------------- Boucle principale ----------------
void loop() {
    sendEnabled = digitalRead(switchPin);
    sensors.requestTemperatures();
    temperatureC = sensors.getTempCByIndex(0);
    // Mise à jour des valeurs
    if (temperatureC != DEVICE_DISCONNECTED_C) {
        if (temperatureC < minTemp) minTemp = temperatureC;
        if (temperatureC > maxTemp) maxTemp = temperatureC;
        sumTemp += temperatureC;
        countTemp++;
        tempHistory[historyIndex] = temperatureC;
        historyIndex = (historyIndex + 1) % historySize;
    }
    // Contrôle des LEDs
    digitalWrite(ledGreen, temperatureC < 30 ? HIGH : LOW);
    digitalWrite(ledYellow, (temperatureC >= 30 && temperatureC <= 40) ? HIGH : LOW);
    digitalWrite(ledRed, temperatureC > 40 ? HIGH : LOW);
```

```
// Navigation entre les pages
if (digitalRead(buttonNext) == LOW) {
   currentPage = (currentPage + 1) % totalPages;
   delay(200);
}
// Réinitialisation des valeurs
if (digitalRead(buttonReset) == LOW) {
   resetStats();
   currentPage = 0;
   delay(200);
}
// Affichage sur OLED
display.clear();
display.setTextAlignment(TEXT_ALIGN_LEFT);
display.setFont(ArialMT_Plain_16);
if (currentPage == 0) drawPage0();
else if (currentPage == 1) drawPage1();
else if (currentPage == 2) drawPage2();
display.display();
// Gestion des clients Web
WiFiClient client = server.available();
if (client) {
   String request = client.readStringUntil('\r');
   client.flush();
   if (sendEnabled) {
      if (request.indexOf("GET /temp") >= 0) {
         client.println("HTTP/1.1 200 OK");
         client.println("Content-Type: text/plain");
         client.println();
         client.print(temperatureC);
      } else {
         client.println("HTTP/1.1 200 OK");
         client.println("Content-type:text/html");
```

```
            client.println();
            client.print(html);
          }
        } else {
          client.println("HTTP/1.1 503 Service Unavailable");
          client.println("Content-Type: text/plain");
          client.println();
          client.print("OFF");
        }
        client.stop();
    }
    delay(500);
}
```

Bibliothèque

1. *Hands-on ESP32 with Arduino IDE : Unleash the power of IoT with ESP32.*
   Disponible sur : https://sciendo.com/fr/chapter/9781837637713
2. *Vedat Ozan Oner – Developing IoT Projects with ESP32 (2e édition).*
   Disponible sur : https://www.fnac.com/livre-numerique/a17527855/Vedat-Ozan-Oner-Developing-IoT-Projects-with-ESP32-Second-Edition
3. https://fr.scribd.com/document/551264192/Cours-Architecture-IOT-3
4. https://app.cirkitdesigner.com/project/745b1bea-ea43-401c-aa76-5700d76f8bbe
5. https://www.amazon.fr/Pack-d%C3%A9marrage-IoT-projets-Apprendre/dp/B0FKH1K8R7
6. *A Low-Cost IoT System Based on the ESP32 Microcontroller for Efficient Monitoring of a Pilot Anaerobic Biogas Reactor.*
   https://www.researchgate.net/publication/387376411_A_Low-Cost_IoT_System_Based_on_the_ESP32_Microcontroller_for_Efficient_Monitoring_of_a_Pilot_Anaerobic_Biogas_Reactor
7. *Monitoring of Temperature in Retail Refrigerated Cabinets Applying IoT Over Open-Source Hardware and Software.*
   Disponible sur : https://www.mdpi.com/1424-8220/20/3/846
8. *IoT-based Real-time Temperature Monitoring in Critical Systems.*
   Disponible sur : https://www.igminresearch.com/articles/a-pdf/igmin303.pdf
9. *Un système de surveillance environnementale basé sur l'IoT utilisant un microcontrôleur ESP32.*
   Disponible sur : https://www.researchgate.net/publication/388940302_AN_IOT-BASED_ENVIRONMENTAL_MONITORING_SYSTEM_USING_TEMPERATURE_HUMIDITY_AIR_QUALITY_LIGHT_INTENSITY_AND_RAIN_DETECTION_SENSORS_WITH_ESP_32_MICROCONTROLLER
10. *Devfolio – Guide de projets ESP32 et IoT en français.*
    Disponible sur : https://devfolio.co/projects/livre-esp-francais-pdf-a807
11. *Introduction à l'ESP32 et à l'Internet des Objets (IoT).*
    Disponible sur : https://fr.scribd.com/document/732567481/esp32final
12. *MicroPython – Programmation de microcontrôleurs (page Wikipédia française).*
    Disponible sur : https://fr.wikipedia.org/wiki/MicroPython
13. **Digi International.** (2024). *Edge Computing vs. Cloud Computing: Differences and Relationship.*
    Disponible sur : https://www.digi.com/blog/post/edge-computing-vs-cloud-computing
14. **Metrology and Quality News**. *Industrial Tech 4.0: Emerging Trends Shaping the Future of Smart Factories.* 2024.
    Disponible sur : https://metrology.news/industry-4-0-reviewing-the-future-of-manufacturing/