

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكر بلقايد - تلمسان

Université Aboubakr Belkaïd – Tlemcen –

Faculté de TECHNOLOGIE



MEMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

En : ELECTRONIQUE

Spécialité : INSTRUMENTATION

Par : BENYOUB CHAIMAA ZINEB et BENADJEB A CHAIMA

Sujet

**Etude d'une station météo en vue d'anticiper le traitement
des cultures agricoles**

Soutenu publiquement, le 20 Juin 2023, devant le jury composé de :

Mme Bouazza née Guen Ahlam

M Zoughagh Nabil

M Brixi Nigassa M.E. A

Professeur

MCB

MCB

Université de Tlemcen

Université de Tlemcen

Université de Tlemcen

Présidente

Examineur

Encadrant

Année universitaire :2022 /2023



DEDICACE

— Je dédie cet humble travail :

En mémoire de **ma grand-mère et de mon grand-père** , que Dieu ait pitié d’eux dans son vaste paradis.

— A mon très chère père : **HAMZA**

Permettez-moi de vous exprimer mon grand amour. Je suis tellement fière d’être votre fille et de pouvoir enfin réaliser ce que vous vouliez et attendiez de moi. Elle n’a jamais cessé de répondre à nos besoins, de nous encourager et de nous aider à choisir la voie du succès. Votre patience, votre bienveillance, vos précieux conseils et votre confiance en moi ont été des facteurs décisifs dans ma réussite. Cher Père, soyez sûr de trouver des fruits dans cet humble travail de votre sacrifice ainsi qu’une profonde expression d’affection et de ma sincère gratitude, que Dieu vous protège et vous protège.

— A ma chère maman : **HAFIDA**

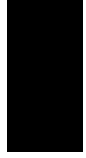
Si Dieu a fait le paradis sous les pieds des mères, ce n’est pas pour rien. Vous êtes pour moi le symbole de la bonté par excellence, la source de la tendresse et l’exemple du dévouement qui ne cesse de m’encourager et de prier pour moi. Vos prières et vos bénédictions m’ont beaucoup aidé dans mes études. Il n’y a pas de dévotion assez éloquente pour exprimer ce que vous méritez pour tous les sacrifices que vous avez faits pour moi. J’ai fait plus qu’une mère ne peut faire pour garder ses enfants sur le droit chemin de la vie et de l’éducation. Je vous dédie ce travail en témoignage de mon amour profond. Que Dieu vous protège et vous accorde santé, longue vie et bonheur.

— A Mes chers frères : **Islem et Mohammed**

que j’aime, que Dieu les protège et leur accorde la réussite. A Tous mes amis d’enfance et du long parcours scolaire et universitaire

— A Toute ma famille Tout ceux qui m’aiment et que j’aime.

— **BENYOUB CHAIMAA ZINEB**



DEDICACE

— Je dédie ce travail .

a ma mère :

Pour son amour ses encouragements et ses sacrifices .

a mon père :

Pour son soutien. Son affection et la confiance qu'il m'a accordé.

A mon frère **Abd Allah**

Pour ses soutiens moral et leur conseil.

A mes souers

Rania , Houda qui m'a aidé et supporté dans les moment difficile.

A tous les membres de ma famille .

a tous mes amis et tous ceux qui m'aiment.

Benhadjeba chaima



REMERCIEMENTS

ON remercie Allah qui nous a aidés et nous a donné la patience, le courage, la volonté et la force d'achever ce travail et durant toutes ces années d'études.

NOUS tenons à exprimer nos sincères remerciements à notre encadrant, **M. BRIXI-NIGASSA Mohammed El Amine**, pour ses précieux conseils, sa confiance, sa patience, ses encouragements et son regard critique pour donner le meilleur, ainsi que pour sa présence et son aide à toutes les étapes. de l'achèvement de ce travail. Merci pour tout ce que vous avez fait.

EN particulier, les membres du jury de notre travail, merci à tous. Pour avoir pris le temps de lire et d'évaluer ce travail.

— Nous tenons remercier **Mme Bouazza, née GUEN Ahlam**, professeur à la Faculté de Technologie, Université Abu Baker Belkaid, Tlemcen, pour son intérêt pour ce travail et son accord pour le présider et en discuter.

De même, nous tenons à remercier **M. zoughagh Nabil**, qui nous a fait l'honneur d'examiner notre travail et il a pris la peine de le lire.

NOUS tenons à remercier en cette occasion tout le corps professoral et administratif de la faculté de technologie département de génie électrique et électronique Université d'Abou-Bekr Belkaid Tlemcen pour la richesse et la qualité de leurs enseignements et qui déploient de grands efforts actualisés. Nous tenons à remercier aussi tous **les membres du laboratoire de recherche du Génie biomédical GBM**. Enfin, nous tenons à remercier tous personne qui a participé de près ou de loin pour rendre ce travail réalisable et présentable.



Résumé

Ce travail a pour objectif la réalisation d'une station météo permettant de récolter différents paramètres météorologiques tels que l'humidité, température, humidité du sol, intensité lumineuse et pluie en vue d'anticiper l'apparition de certaines maladies affectant les champs agricoles. Pour mener à bien ce projet, nous nous reposons sur trois composants de base qui sont une carte Wemos D1 mini intégrant un module WiFi ESP8266 sur laquelle sont branchés les différents capteurs, une base de données Firebase et une application Android « Weather and Farmer ».

Le but ici c'est de récolter les différentes données des capteurs branchés sur la carte Wemos D1 mini et les stocker sur une base de données Firebase que nous avons créé. Cette base de données assurera le lien entre la carte Wemos D1 mini et l'application Android. Cette dernière récupérera les données à partir de cette base de données et les affichera en temps réel sur l'interface de l'application.

L'application traite ces données reçues et les compare aux données de références relatives à chaque maladie. Si ces données sont en dehors de cette plage de référence, une notification en plus d'un sms sont envoyés directement à l'agriculteur signalant le risque d'apparition de maladies dans son champ. Grâce à ça, l'agriculture pourra anticiper le traitement de ses cultures et adopter la bonne conduite visant à préserver ses récoltes.

Les résultats obtenus dans ce travail montrent le bon fonctionnement de notre station météo et nous encourage à améliorer et à optimiser encore plus le prototype.

Mots clés : station météo, agriculture, Wemos miniD1, Firebase, MIT App Inventor, Android.



Abstract

The aim of this work is to create a weather station that can collect various meteorological parameters such as humidity, temperature, soil moisture, light intensity and rainfall, to allow a farmer to anticipate the onset of certain diseases affecting agricultural fields. To carry out this project, we are relying on three basic components : a Wemos D1 mini board incorporating an ESP8266 WiFi module to which the sensors are connected, a Firebase database and an Android "Weather and Farmer" application.

The aim here is to collect the various data from the sensors connected to the Wemos D1 mini card and store them in a Firebase database we've created. This database will provide the link between the Wemos D1 mini card and our Android application. The latter will retrieve the data from this database and display it in real time on the application's interface.

The application processes the data received and compares it with reference data for each disease. If these data fall outside this reference range, a notification and an SMS are sent directly to the farmer, indicating the risk of disease appearing in his field. This will enable farmers to anticipate the need to treat their crops and adopt the right approach to preserve their harvests.

The results obtained in this work show how well our weather station works, and encourage us to further improve and optimize the prototype.

Keywords : weather station, agriculture, Wemos miniD1, Firebase, MIT App Inventor, Android.

الملخص

الهدف من هذا العمل هو إنشاء محطة طقس قادرة على جمع مختلف العوامل المناخية مثل الرطوبة ودرجة الحرارة ورطوبة التربة وشدة الإضاءة وكمية الأمطار، لتمكين الفلاح من التنبؤ بظهور بعض الأمراض التي تؤثر على حقول الزراعة.

لتنفيذ هذا المشروع، نعتمد على ثلاثة مكونات أساسية: لوحة Wemos Mini D1 تحتوي على وحدة ESP8266 للواي فاي والتي يتم توصيل الحساسات بها، قاعدة بيانات firebase. وتطبيق Android "الطقس والفلاحة".

الهدف هنا هو جمع البيانات المختلفة من الحساسات المتصلة بلوحة Wemos mini d1. وتخزينها في قاعدة بيانات FIREBASE التي قمنا بإنشائها. ستوفر هذه القاعدة الرابط بين لوحة Wemos mini d1 وتطبيقنا على نظام Android سيقوم التطبيق بجمع البيانات من هذه القاعدة وعرضها في الوقت الحقيقي على واجهة التطبيق.

يقوم التطبيق بمعالجة البيانات المستلمة ومقارنتها مع بيانات المرجع لكل مرض. إذا تجاوزت هذه البيانات النطاق المرجعي، يتم إرسال إشعار notification ورسالة نصية SMS. مباشرة إلى الفلاح للإشارة إلى خطر ظهور المرض في حقله. هذا يمكن الفلاحين من التنبؤ بحاجتهم لمعالجة محاصيلهم واعتماد النهج الصحيح للحفاظ على محاصيلهم.

تظهر النتائج التي تم الحصول عليها في هذا العمل مدى جودة محطاتنا الجوية وتشجيعنا على تحسينها بشكل أفضل من النموذج الأولي.

كلمات مفتاحية:

Android، Inventor، App MIT، Wemos miniD1، firebase، Wi-Fi، capteurs، الفلاحة، محطة الارصاد الجوية



TABLE DES MATIÈRES

LISTE DES FIGURES	XII
LISTE DES TABLEAUX	XIII
LISTE DES ABRÉVIATIONS	XIV
INTRODUCTION GÉNÉRALE	1
1 Généralités sur l'agriculture et la station météo	4
1.1 INTRODUCTION	5
1.2 Historique	5
1.2.1 L'agriculture en Algérie	7
1.3 Quelques maladies des champs agricoles	8
1.3.1 la rouille brune sur les feuilles du blé	8
1.3.2 Pourriture grise ou botrytis sur les raisins	8
1.3.3 Pommes de terre tardives brillantes	9
1.3.4 Tavelure du Pommier	10
1.4 La Smart Agriculture et les stations météo	11
1.4.1 Smart Agriculture	11
1.4.2 La station météo	12
1.5 Les types des stations météo	13
1.5.1 Stations météo classiques	13
1.5.2 Stations météo professionnelles	14
1.5.3 Stations météo analogiques	15
1.5.4 Stations météo pour smartphone	15
1.6 Paramètres à mesurer dans une station météo	16
1.6.1 Température	17
1.6.2 L'intensité lumineuse	18
1.6.3 L'humidité	18
1.6.4 Humidité du sol	19

1.6.5	Précipitations	20
1.7	La problématique de notre projet	21
1.8	Conclusion	22
2	les outils matériel et logiciels pour la réalisation de notre station météo	23
2.1	Introduction	24
2.1.1	Concept de notre station météo :	24
2.2	La partie matérielle de la station météo	25
2.2.1	Carte Arduino	25
2.2.2	IDE (Integrated Développement Environnement) Arduino	26
2.2.3	Module Wifi ESP 8266	27
2.2.4	WeMos Lolin D1 mini :	28
2.2.5	Multiplexeur CD4051	32
2.2.6	Présentation des capteurs utilisés dans notre station météo	34
2.2.6.1	Photorésistance	34
2.2.6.2	Capteur de Température et d'Humidité	35
2.2.6.3	Capteur de pluie	36
2.2.6.4	Capteur d'humidité du sol	37
2.3	Partie logicielle de la station météo	38
2.3.1	Base de données Firebase	38
2.3.2	Applications Android	42
2.3.2.1	Présentation de l'outil MIT App inventor	42
	Conclusion	48
3	Conception et réalisation D'une station météo	49
3.1	Introduction	50
3.2	Description de notre station météo	50
3.3	Circuit électronique de notre station météo	51
3.3.1	Programmation de la carte WemosD1 mini avec l'IDE Arduino	52
3.4	Base de données	57
3.5	L'application Android « Weather and farming »	58
3.5.1	Partie Designer	59
3.5.2	Partie Blocks :	60
3.6	Tests pratique de notre station météo	65
3.7	Conclusion :	71
	CONCLUSION GÉNÉRALE	72

ANNEXES	77
A.1 Programme Arduino complet	77
A.2 Datasheet CD4051	79

LISTE DES FIGURES

1.1	Carte montrant l'évolution de l'agriculture dans l'ère néolithique [2].	6
1.2	Maladie de la rouille brune sur les feuilles du blé [6].	8
1.3	Pourriture grise ou botrytis sur les raisins [7].	9
1.4	Mildiou des pommes de terre. [7].	9
1.5	Tavelure du Pommier sur les feuilles et les fruits.	10
1.6	Schéma descriptif de la Smart Agriculture [9].	11
1.7	Exemple de station météo [8].	12
1.8	Exemple de station météo classique [11].	14
1.9	station météo professionnelle [11].	15
1.10	Exemple de station météo analogique.	15
1.11	Exemple de station météo connectée par Bluetooth [10].	16
1.12	Exemple de station météo connectée par WiFi [8].	16
1.13	Variables météorologiques [11].	17
1.14	Exemple de thermomètre électronique [10].	18
1.15	(a) Photorésistance - (b) photodiode [14, 15].	18
1.16	Exemple d'hygromètre à cheveux [10].	19
1.17	capteur d'humidité du sol utilisé avec Arduino.	20
1.18	Exemple de pluviomètre.	21
1.19	Schéma pour le principe de fonctionnement de notre système.	21
2.1	Schéma synoptique de la station météo proposée.	24
2.2	Quelques exemples de cartes Arduino [17].	26
2.3	Interface de l'IDE Arduino V2.10.	26
2.4	Exemple de modules Wifi ESP 8266 et cartes à base d'Arduino intégrant ce dernier.	28
2.5	Carte Wemos D1 mini [20].	29
2.6	Sélection de l'onglet préférences pour ajouter le lien d'installation des bibliothèques ESP8266.	31
2.7	Introduction du lien d'installation des bibliothèque ESP8266 dans l'IDE Arduino.	31

2.8	Sélection de la carte Wemos D1 mini depuis le gestionnaire de cartes.	32
2.9	Sélection de la carte Wemos D1 mini depuis une zone dédiée dans la nouvelle version de l'IDE Arduino.	32
2.10	Circuit multiplexeur CD4051 et ses broches [21].	33
2.11	Principe de fonctionnement d'un capteur.	34
2.12	Photorésistance avec son câblage sur Arduino [10].	35
2.13	Image réelle du capteur de Température et d'Humidité DHT22 avec son câblage avec Arduino [10].	36
2.14	Capteur de détection de pluie avec son câblage avec Arduino.	36
2.15	Capteur d'humidité du sol avec son câblage.	37
2.16	Composants de la partie logicielle de notre station météo.	38
2.17	Le logo Firebase de Google [26].	38
2.18	fenêtre d'accueil de Firebase.	39
2.19	Etape de création de projet dans Firebase.	39
2.20	Fenêtres du projet PFESTATION créé sur Firebase.	40
2.21	Fenêtre de création de base de données dans Firebase dans le projet PFESTATION.	40
2.22	Lien de la base de données.	41
2.23	Clé d'authentification de la base de données.	41
2.24	Page d'accueil de la plateforme MIT App Inventor.	42
2.25	Compte d'accès à la plateforme MIT App Inventor.	43
2.26	Fenêtre de la partie Designer dans MIT App Inventor.	43
2.27	Fenêtre de la partie Blocks dans MIT App Inventor.	44
2.28	Options de tests d'applications disponibles sur MIT App Inventor.	45
2.29	Emulateur de MIT App Inventor.	46
2.30	Code QR généré par MIT App Inventor pour se connecter à AI Companion.	46
2.31	Code QR généré par MIT App Inventor pour télécharger l'application installable en extension apk.	47
3.1	Schéma synoptique de notre station météo.	51
3.2	Circuit électronique de notre système station météo.	52
3.3	Déclaration des bibliothèques dans le programme Arduino.	53
3.4	Définition de la base de données Firebase pour pouvoir y accéder.	53
3.5	Stockage et synchronisation dans la base de données Firebase.	53
3.6	Déclaration des variables.	54
3.7	Première partie de la fonction void setup.	54

3.8	Lancement de la connexion WIFI avec la carte Wemos D1 mini ainsi que la connexion à la base de données Firebase.	55
3.9	Création de la fonction ChangeMux du CD4051.	55
3.10	Lecture et envoi des données des capteurs à la base de données Firebase.	56
3.11	Affichage des données capteurs sur le moniteur série.	57
3.12	Récupération du lien de notre base de données Firebase.	57
3.13	Données de notre base de données Firebase.	58
3.14	les 4 fenêtres (screens) principales de notre application Android.	59
3.15	Les quatre fenêtres principales d'accès à l'application et aux différents paramètres mesurés avec les maladies.	60
3.16	Bloc de l'écran d'accueil de l'application	60
3.17	Bloc gérant l'accès par mot de passe à l'application.	61
3.18	Bloc DB1, DB2 et DB3 qui ont pour fonction la vérification et l'affichage des paramètres pluie, intensité lumineuse et humidité avec notification et sms.	62
3.19	Bloc DB4 qui a pour fonction la vérification et l'affichage du paramètre humidité du sol avec notification et SMS.	63
3.20	Bloc DB5 qui a pour fonction la vérification et l'affichage du paramètre température avec notification et SMS.	64
3.21	Bloc de retour vers l'écran de connexion (login).	65
3.22	Bloc amenant l'utilisateur vers l'écran contenant les détails sur les maladies.	65
3.23	Bloc de retour à l'écran d'accueil à partir de l'écran contenant les détails sur les maladies.	65
3.24	Notre station météo.	66
3.25	Valeurs des capteurs sur le moniteur série	66
3.26	Données des capteurs stockés sur la base de données Firebase.	67
3.27	Changements de valeurs en temps réel du capteur de pluie au niveau de la base de données Firebase.	67
3.28	Lancement de l'application Android Weather and Farmer à partir du smartphone.	68
3.29	valeurs intermédiaires pas de maladie	68
3.30	Réception d'une notification en plus d'un SMS contenant la maladie à prendre en considération et affichage des paramètres en relation avec cette maladie sur fond rouge.	69
3.31	ecran des maladies	70
3.32	Réception du SMS sur un téléphone traditionnel.	70



LISTE DES TABLEAUX

2.1	Description d'interface de l'IDE Arduino.	27
2.2	Tableau représentant les différentes fonctions des broches de la carte Wemos D1 mini [20].	30
2.3	Les broches du multiplexeur CD4051.	33
2.4	Description de chaque composant de la partie Designer.	44
2.5	Description de chaque composant de la partie Blocks.	45



LISTE DES ABRÉVIATIONS

SPI,SCK Serial peripheral interface Serial clock, horloge

I2C,Data Inter integrated circuit donne numérique

MIT Massachusetts Institute of Technology.

NodeMCU Node MicroController Unit

SPI, SS Serial peripheral interface slave select

SPI,MISO Serial peripheral interface master input slave output

SPI,MOSI Serial peripheral interface master output slave input

IEEE Institute of Electrical and Electronics Engineers

IDE Integrated Development Environment

JSON JavaScript Object Notation

NoSQL Not only Structured Query Language

PWM pulse With modulation

UART universal asynchronous reciver

URL uniform Resource locato

SSID services set Identifier

BSD Berkeley software Distribution

GPIO General purpose input/outp

USB universal serial Bus

SOC System on a Chip

IOT Internet of Things

ESP electronic stability program

SDA Serial Data

SCL Serial Clock

WIFI Wirelesse Fidelity



INTRODUCTION GÉNÉRALE

L'AGRICULTURE a toujours été un domaine clé au fil des siècles pour toutes civilisations ou pays. Son importance est telle qu'elle est le pilier économique de plusieurs pays à travers le monde et devient le nerf de la guerre actuelle dans un monde où la population ne cesse de croître et où la capacité de pouvoir nourrir toutes ces personnes devient un enjeu majeur. Dans cette optique et en vue d'assurer la souveraineté et l'autosuffisance alimentaire dans ce domaine, l'Algérie mène une politique visant à augmenter et à diversifier sa production.

L'Algérie dispose d'une situation géographique stratégique offrant un climat favorable et des terres fertiles. Néanmoins, le pays doit faire face à quelques défis majeurs parmi lesquels l'optimisation de l'utilisation des ressources naturelles par exemple l'eau pour l'irrigation et surtout s'adapter aux changements climatiques qui affectent le monde entier. Ce dernier point est d'autant plus important car les conditions climatiques conditionnent l'apparition précoce ou tardive de certaines maladies affectant les champs de cultures et par conséquent impose à l'agriculteur ou l'ingénieur agronome d'avoir une vue d'ensemble sur les données météo afin d'anticiper le traitement des champs.

C'est dans cette optique que nous proposons de concevoir en collaboration avec un ingénieur agronome une station météo donnant accès en temps réel à des paramètres météorologiques tels que la température, l'humidité, l'ensoleillement, l'humidité du sol et la pluie. Pour se faire, nous nous basons sur une carte Wemos D1 mini qui peut être programmée en utilisant l'IDE traditionnelle Arduino et disposant d'un module WiFi ESP8266. Cette dernière est reliée à une base de données Firebase que nous avons créée et qui permet de stocker les données de ces capteurs. L'objectif de l'utilisation de cette base de données

Firestore est de faire le lien entre la carte Wemos D1 mini et par conséquent les capteurs et une application Android que nous avons créé. Cette application Android que nous avons nommé « Weather and Farmer » récupère les données des capteurs à partir de la base de données Firestore, les compare avec les données de références obtenues grâce à l'aide de l'ingénieur agronome et alerte l'agriculteur par le biais d'une notification et sms sur le risque de maladie des champs. La notification et le sms envoyés contiennent aussi l'information sur la maladie en fonction des paramètres favorisant son apparition.

Par soucis de clarté et de compréhension, nous avons scindé notre projet en trois chapitres :

- Dans le premier chapitre, nous donnons un bref aperçu de l'histoire de l'agriculture avec son importance dans la souveraineté et l'autosuffisance alimentaire nationale. Nous présentons quelques maladies répandues en Algérie sur lesquelles nous travaillerons. Nous exposons quelques types de stations météo déjà existantes sur le marché avant de terminer par poser le contexte et la problématique de notre travail.
- Le deuxième chapitre est consacré à l'étude de la carte Wemos D1 mini dont le choix est justifié par le fait que cette dernière intègre un module WiFi ESP8266. Nous présentons par la suite les différents capteurs utilisés dans notre travail, capteur de température, d'humidité, d'humidité sol, d'intensité lumineuse et de pluie. Nous donnerons aussi une brève description du circuit multiplexeur/démultiplexeur CD4051 nécessaire dans notre cas pour augmenter le nombre d'entrées/sorties analogiques de notre carte Wemos D1 mini. Nous ferons par la suite un rappel sur les étapes de création d'une base de données Firestore. Nous terminerons enfin par décrire les étapes de création d'une application Arduino en utilisant la plateforme MIT App Inventor.
- Dans le dernier chapitre, nous présenterons le principe de fonctionnement de notre station météo. Ensuite, Nous détaillerons et expliquerons en détails le programme Arduino, la base de données Firestore et l'application Android « Weather and Farmer » créés dans ce projet. Nous finirons par des tests pratiques montrant le bon fonctionnement de notre station météo.

Nous clôturerons ce projet par une conclusion générale et des perspectives visant à améliorer le travail réalisé.

Généralités sur l'agriculture et la station météo

Sommaire

1.1	INTRODUCTION	5
1.2	Historique	5
1.2.1	L'agriculture en Algérie	7
1.3	Quelques maladies des champs agricoles	8
1.3.1	la rouille brune sur les feuilles du blé	8
1.3.2	Pourriture grise ou botrytis sur les raisins	8
1.3.3	Pommes de terre tardives brillantes	9
1.3.4	Tavelure du Pommier	10
1.4	La Smart Agriculture et les stations météo	11
1.4.1	Smart Agriculture	11
1.4.2	La station météo	12
1.5	Les types des stations météo	13
1.5.1	Stations météo classiques	13
1.5.2	Stations météo professionnelles	14
1.5.3	Stations météo analogiques	15
1.5.4	Stations météo pour smartphone	15
1.6	Paramètres à mesurer dans une station météo	16
1.6.1	Température	17
1.6.2	L'intensité lumineuse	18
1.6.3	L'humidité	18
1.6.4	Humidité du sol	19
1.6.5	Précipitations	20
1.7	La problématique de notre projet	21
1.8	Conclusion	22

1.1 INTRODUCTION

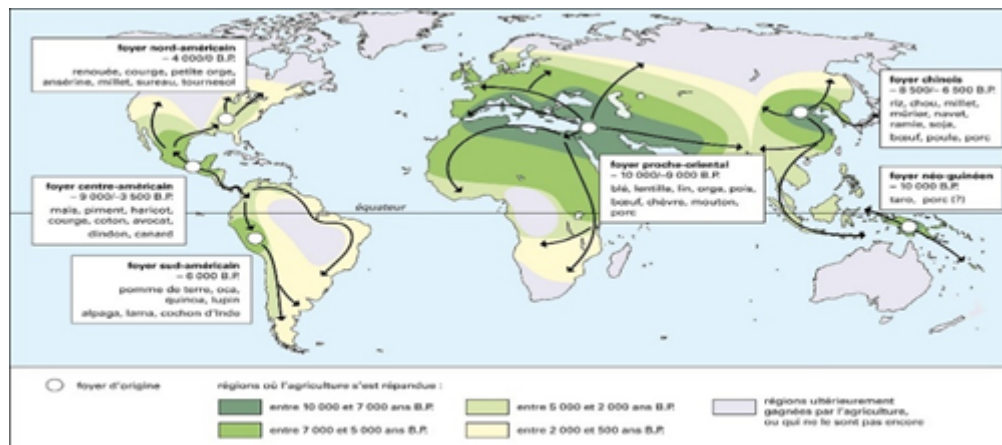
L'agriculture a toujours été et restera un domaine clé pour les différentes civilisations et pays. Son importance est telle qu'elle est le pilier économique de plusieurs pays à travers le monde. Elle permet de répondre aux besoins alimentaires d'un pays et des populations mondiales. Néanmoins, ce secteur stratégique a besoin d'attentions quant aux maladies qui peuvent l'affecter et ainsi faire baisser la production ou l'endommager. Les critères climatiques tels que la température, l'humidité, l'intensité du soleil, quantité de précipitations, vent, etc. . . sont les facteurs clé affectant ce secteur. Les systèmes dits « station météo » permettent justement de rassembler tous ces paramètres afin d'être utilisés pour prévenir une éventuelle maladie.

Dans ce chapitre, nous nous intéresserons justement aux stations météo. Mais avant, nous commencerons par donner un petit historique sur l'agriculture. Nous donnerons ensuite quelques maladies répandues ici en Algérie avant de présenter les différentes stations météo existantes en Algérie. Nous finirons par donner la problématique et l'objectif de notre travail.

1.2 Historique

Selon les historiens, les premières traces de l'agriculture remontent à plus de 10000 ans et ont été constatée d'abord au moyen orient, ensuite en chine, en Nouvelle Guinée, puis en Amérique centrale, avant de s'étendre à l'ensemble des continents [1]. A ces moments-là, les cinq millions d'êtres humains qui vivaient à cette époque, qui étaient généralement des nomades, se nourrissaient spécialement de la chasse, de la pêche et de la cueillette [2]. Néanmoins, et progressivement, l'homme a commencé à s'intéresser à la culture de blé et végétaux et à l'élevage d'animaux pour subvenir à ses besoins alimentaires. Les premières cultures ont été réalisées sur brûlis de forêts ou de savanes avec des outils très simples, en général de bois et de pierre taillée [1, 3]. En même temps, sans doute pour pouvoir être près des cultures et pouvoir surveiller leurs animaux, les populations d'agriculteurs se

sont sédentarisées. Le développement de l'agriculture a permis une augmentation sensible et une sécurisation des ressources alimentaires de ces petits groupes humains avec deux conséquences : une augmentation de la population agricole, puis rurale et enfin urbaine.



Pendant l'Antiquité, les techniques de culture sont restées assez rudimentaires, malgré quelques avancées mais toujours très lentes : labours, longues périodes de jachère, porté par des mules ou des chameaux lorsqu'ils ne sont pas à dos d'homme, meules manuels, . . . D'autre part, les techniques d'irrigation ont permis de cultiver des sols désertiques (notamment en Égypte) pour augmenter les rendements (comme le riz en Chine). En Europe, au cours du Moyen Âge, une série de petites innovations a permis des changements progressifs dans les méthodes agricoles [1]. La première révolution agricole des temps modernes a commencé en Italie, en Flandre et en Angleterre au XVIe siècle. Au Royaume-Uni, les règles d'enclos interdisent le pâturage communal pour les propriétaires qui deviennent maîtres chez eux. Les jachères disparaissent peu à peu au profit des racines et des pâturages artificiels. Au 19ème siècle, les machines agricoles à cheval ou à vapeur, en particulier les tondeuses à gazon, les batteuses, ont commencé à devenir populaires, et les premières moissonneuses-batteuses ont également été utilisées dans les plaines américaines. Avec le développement à long terme du commerce maritime, les plantes migrent fréquemment et colonisent de nouvelles terres lorsque les conditions climatiques leur sont favorables. Les exodes ruraux augmenteront, approvisionnant de nouvelles usines avec plus de travailleurs [1]. Depuis 1950, le progrès agricole fait un bond considérable

en avant. Il se base sur des semences améliorées et une utilisation intensive d'engrais chimiques, Multiplicateur de soins contre les plantes nocives. En même temps, la mécanisation agricole a explosé, entraînant des améliorations significatives des rendements et de la productivité du travail. Cela se traduira par un exode rural massif, mais nécessitera également plus de parcelles et d'exploitations. Dans le tiers-monde, la révolution verte utilise les mêmes facteurs de production, souvent ajoutés à l'irrigation, ce qui conduit également à des augmentations importantes de la production agricole [4].

1.2.1 L'agriculture en Algérie

Ce qu'il faut retenir, c'est que le secteur agricole est un secteur très important pour le pays représentant près de 12.3% du PIB et emploie plus de 10 pourcentage de la population active (données 2022) [5]. Les principales cultures agricoles du pays comprennent notamment les céréales, les arbres, les légumes et l'élevage de bétail. Cependant, le pays dispose d'un grand potentiel de développement, d'autant plus que de nombreuses zones agricoles sont actuellement sous-développées.

Pour assurer la souveraineté alimentaire et l'autosuffisance nationale, le pays doit constamment augmenter sa capacité de production tout en optimisant l'utilisation des ressources naturelles et s'adapter aux changements non seulement climatiques, mais aussi économiques et environnementaux [5]. Dans ce travail, l'une des données qui nous intéresse le plus ici est les changements climatiques qui mettent très souvent en difficultés l'agriculteur et le confronte à certaines maladies qui diminuent considérablement le rendement de ses champs agricoles ou endommagent ses cultures. Nous allons dans ce qui suit présenter quelques maladies répandues ici en Algérie et dont on peut anticiper les traitements en ayant accès à des données météo bien précises tels que les changements de température, humidité, humidité de la terre, précipitations, intensité lumineuse, etc. . . .

1.3 Quelques maladies des champs agricoles

1.3.1 la rouille brune sur les feuilles du blé

Cette maladie se présente sous forme des taches sur les feuilles, les gaines et les épis. Elle se caractérise par la présence de pustules allongées de couleur jaune à orange (figure 1.2). Les pustules sont disposées en rangées entre les nervures à la surface des feuilles. Les conditions propices à son développement sont une température entre [20 et 25] degrés Celsius [6].



Figure 1.2 – Maladie de la rouille brune sur les feuilles du blé [6].

1.3.2 Pourriture grise ou botrytis sur les raisins

Cette maladie est la conséquence de la présence de champignon nocif appelé Botrytis Cinerea dans les cultures de raisins. Une température relativement basse et une humidité élevée sont parmi les facteurs qui propagent la maladie. Si la température se situe entre 11 et 20 degré celsius, la moisissure se développera sur les cultures de raisins. Le facteur humidité est un facteur clé dans la propagation de la maladie en présence notamment d'une surpopulation, ce qui favorisera l'infection des noyaux sains par les noyaux infectés [7].



Figure 1.3 – Pourriture grise ou botrytis sur les raisins [7].

1.3.3 Pommes de terre tardives brillantes

Cette maladie, comme la pourriture grise est la conséquence de la présence de champignon nocif appelé Mildiou dans les cultures de pommes de terre. Cette maladie apparaît sous une forme légèrement pigmentée avec des taches noires brunes à violettes sur les côtés des feuilles et des tiges couleurs vives. Des taches violettes apparaissent aussi dans le sol et peuvent atteindre une profondeur de 8 à 95 mm sur les tubercules.[7]



Figure 1.4 – Mildiou des pommes de terre. [7].

1.3.4 Tavelure du Pommier

Apple scab en anglais : est une des maladies les plus importantes chez le pommier et est présente dans toutes les régions productrices du monde. Cette maladie est causée par le champignon *Venturia Inaequalis*. Ses symptômes apparaissent sous forme de lésions vert olive mais qui dans un stade avancé de la maladie tirent vers le noir. Les feuilles très infectées se dessèchent et tombent, tandis que les taches sur les fruits sont semblables à celle sur les feuilles mais développent un centre brun-noir et liégeuses avec le temps. Les conditions de son développement sont les précipitations fréquentes au printemps ; 25 heures de mouillure à 6 degré celsius déclenchent l'infection (9 heures pour 16-24 degré celsius). L'incubation de la tavelure dure 17 jours à 9 degré celsius (8 jours pour 17-24 degré celsius). Après 4 heures de sécheresse (inférieure à 84 pourcent d'humidité du sol) suivant la mouillure, l'infection est annulée.



Figure 1.5 – Tavelure du Pommier sur les feuilles et les fruits.

Toutes ces maladies citées précédemment peuvent être anticipé en ayant accès à certains paramètres météorologiques. Pour cette raison, le terme smart agriculture et station météo reviennent souvent de nos jours, car cela aide l'agriculteur à prendre une décision en fonction des données qu'il reçoit.

1.4 La Smart Agriculture et les stations météo

1.4.1 Smart Agriculture

Le mot "Smart Agriculture" est formé de "Smart", qui signifie "intelligent" en anglais, et est associé au mot "agriculture", qui désigne l'activité de production de plantes et d'animaux utiles à l'homme. Il est souvent utilisé pour désigner les technologies de l'automatisation, de l'électronique, de l'électricité, de l'informatique et de la communication dans les services agronomiques. Ainsi, l'agriculture intelligente permet l'amélioration des systèmes de contrôle et offre des fonctions de sécurité, de gestion de l'énergie et de communication (filaire ou sans fil via Wifi ou Bluetooth) que l'on retrouve dans les exploitations agricoles. Dans son application la plus avancée, Smart Farmer met en réseau et coordonne le fonctionnement de différents types d'équipements de mesure et de travail. Il peut prendre en charge les tâches les plus complexes ou les plus exigeantes tout en assurant la gestion de la ferme. D'autre part, il peut effectuer des opérations très basiques telles que la mesure de données importantes. Les applications possibles pour les fermes intelligentes impliquent la programmation, la surveillance et le contrôle à distance [8].

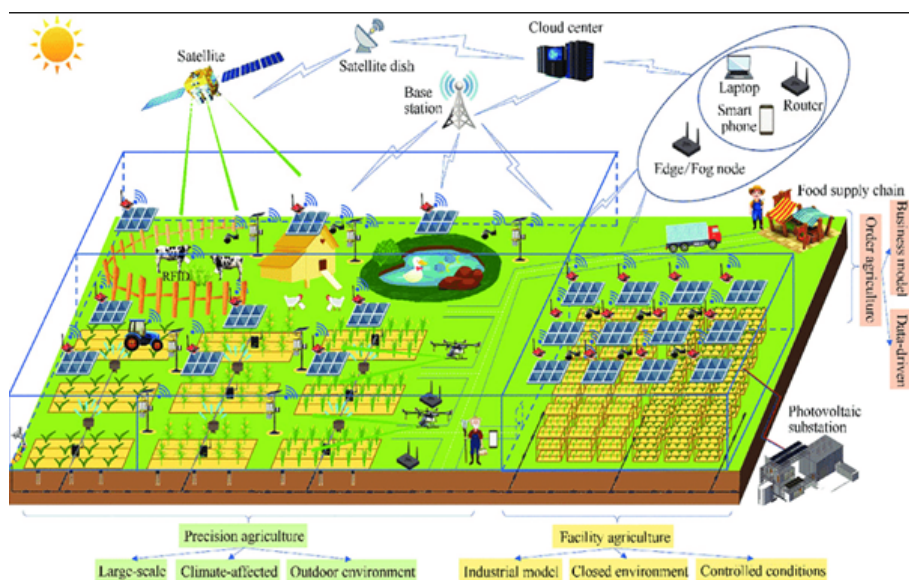


Figure 1.6 – Schéma descriptif de la Smart Agriculture [9].

1.4.2 La station météo

Une station météorologique est un appareil qui collecte et fournit des informations sur des grandeurs physiques liées au Changement climatique. Ces grandeurs physiques peuvent être la température, la vitesse du vent, les précipitations, l'humidité, la lumière le vent, etc... Les stations météo sont utilisées dans plusieurs domaines, notamment en agronomie, pour surveiller les conditions climatiques dans les serres, pour protéger les plantes et les légumes de certains facteurs de risques liés aux maladies pouvant les affecter ou affecter leur Rendement. La surveillance métrologique continue nous permet de prendre des dispositions et des décisions nécessaires afin de limiter la propagation de ces maladies. Une station météo est constituée généralement de plusieurs capteurs de mesures météorologiques. Ces derniers sont connectés à un boîtier qui enregistre, stocke et envoie généralement les mesures à une base de données via un réseau mobile par exemple ou réseau wifi [8].



Figure 1.7 – Exemple de station météo [8].

Au cours des années 1800 et au début des années 1900, il était fréquent que les stations métrologiques locales soient pourvues d'équipements de simulation dédiés. Les hygromètres mesurent les changements d'humidité dans l'air, tandis que les pluviomètres et les baromètres aident à déterminer les lectures précédentes et futures. Aujourd'hui, il existe des stations météorologiques numériques qui mesurent et rapportent mieux les informations pour une visualisation et une compréhension plus facile. Certaines stations météo peuvent même se connecter à des applications pour smartphone ou à des services en

ligne afin que les gens puissent accéder à leurs informations météorologiques de n'importe où. Les observateurs météorologiques modernes peuvent utiliser ces informations pour gérer leurs jardins et surveiller leurs fermes [8].

1.5 Les types des stations météo

1.5.1 Stations météo classiques

Ce sont de petites stations météo, généralement constituées de sondes de température, qui vous donnent une meilleure idée de ce que fait ou fera la météo. Il existe principalement deux types de stations météo classiques :

— Stations locales :

Elles sont utilisées pour connaître les conditions météorologiques à un seul endroit. Elles sont équipées d'un baromètre mécanique et d'un capteur de température. L'objectif de ces stations est de fournir une information sur la température. Cependant, il existe quelques variantes pour mieux optimiser leur utilisation. En effet, certains modèles sont équipés d'un hygromètre pour mesurer l'humidité de l'air. D'autres encore fournissent des informations sur les phases de la lune, jouant dans ce cas un rôle dans le jardinage. Enfin, ceux qui souhaitent connaître la température extérieure afin de détecter les orages ou les conditions givrantes opteront pour des stations de surveillance avec des capteurs externes.

— Stations météorologiques régionales :

Celles-ci fournissent des prévisions précises pour trois à quatre jours à des endroits sélectionnés, selon les conditions. Ces stations météo reçoivent les prévisions locales officielles. Certaines d'entre elles enverront même un message en cas d'alarme [10].



Figure 1.8 – Exemple de station météo classique [11].

1.5.2 Stations météo professionnelles

En général, les stations météo professionnelles fournissent plus d'informations que les stations météo conventionnelles. La variété de capteurs dont elles disposent permettent entre autres de mesurer la quantité de précipitations, la vitesse du vent, l'ensoleillement, la température et la pression atmosphérique. A cet effet, ces stations sont équipées non seulement de sondes de température et d'humidité, mais aussi de pluviomètres, d'anémomètres et de polarimètres. De plus, les stations sont connectées à une console qui centralise toutes les informations mesurées périodiquement via une connexion filaire ou sans fil selon la technologie utilisée. Les données sont ensuite exploitables sur ordinateur à l'aide de logiciels spécialisés. Il existe divers modèles de stations météo professionnelles selon le niveau de l'utilisateur, mais généralement, elles sont regroupées en trois catégories :

1. Pour utilisateurs débutants.
2. Pour utilisateurs intermédiaires.
3. Pour utilisateurs expérimentés.

Les débutants utiliseront des stations météo plus simples et faciles d'utilisation et dont les prix sont accessibles. Les intermédiaires choisiront des stations offrant plus de

fonctionnalités tandis que les utilisateurs expérimentés choisiront d'autres plus complexes [10].



Figure 1.9 – station météo professionnelle [11].

1.5.3 Stations météo analogiques

Les stations analogiques fournissent des mesures de différents paramètres météorologiques de manière analogique. Elles sont conçues généralement pour une utilisation dans la maison, le bureau, le jardin ou le bateau [8].



Figure 1.10 – Exemple de station météo analogique.

1.5.4 Stations météo pour smartphone

Ces stations météo intègrent un ou plusieurs capteurs connectés via Bluetooth ou Internet et accessibles via une application téléchargeable sur smartphone [8].



Figure 1.11 – Exemple de station météo connectée par Bluetooth [10].



Figure 1.12 – Exemple de station météo connectée par WiFi [8].

Quelque soit le type de station météo utilisé, l'objectif sera de relever des données météorologiques pour pouvoir les utiliser dans notre cas comme une aide à la décision pour l'agriculteur. Nous allons voir maintenant quels sont les principaux paramètres à mesurer en général dans une station météo.

1.6 Paramètres à mesurer dans une station météo

L'étude des phénomènes météorologiques implique l'étude de diverses variables météorologiques afin d'évaluer leur impact et leur évolution dans le temps et ainsi d'établir des prévisions. Les variables les plus importantes sont généralement [12] :

- Température.
- Humidité.
- Humidité du sol.

- L'intensité lumineuse.
- Précipitation.

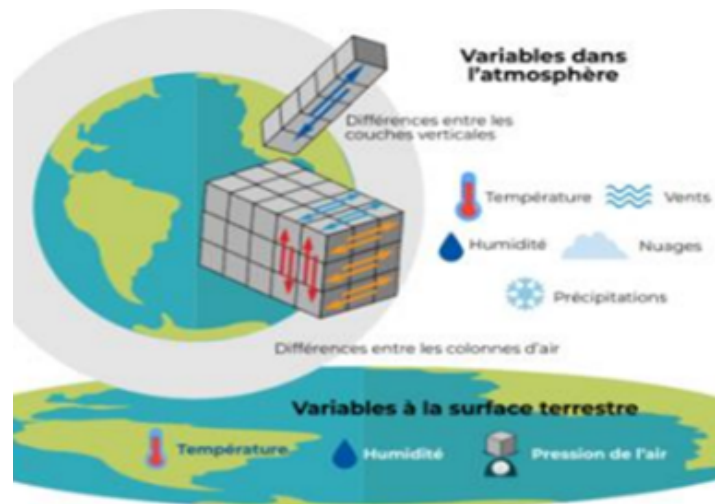


Figure 1.13 – Variables météorologiques [11].

1.6.1 Température

La température peut-être définie comme un ensemble de conditions atmosphériques qui sont variables et qui se traduisent subjectivement en sensations relatives de chaud et de froid [13]. L'appréciation exacte de la température est généralement fournie par un thermomètre ou un capteur de température. Les températures sont exprimées en degrés Celsius ou Kelvin. La température de l'air diminue avec l'altitude avec une baisse de 1C à 150 mm [11]. $T(\text{Kelvin}) = T(\text{Celsius}) + 273,15$.

- Thermomètre :

Le thermomètre a été utilisé pour la première fois pour mesurer une température à la fin du XVe siècle. Il est présent sous différents types et pour différents domaines tels que l'alimentation, la médecine, la métallurgie et la météo. Voici quelques types de thermomètres :

1. Gaz naturel.
2. Avec cadran et aiguilles.
3. Liquides (alcool ou mercure).

- 4. Électronique.
- 5. Magnétique.

Pour la météo, seuls les thermomètres de types liquides et électroniques sont utilisés. Les humains ont longtemps utilisé des thermomètres à liquide, mais ceux-ci sont progressivement abandonnés au détriments des thermomètres électroniques à cause notamment de leur grande précision.



Figure 1.14 – Exemple de thermomètre électronique [10].

1.6.2 L'intensité lumineuse

L'intensité lumineuse est une grandeur physique permettant de décrire la capacité d'éclairage d'une source lumineuse ponctuelle dans une direction particulière [11]. Elle se mesure généralement en candela "cd" en utilisant un capteur d'intensité lumineuse. Ce paramètre peut-être mesuré en utilisant une photodiode ou une photorésistance

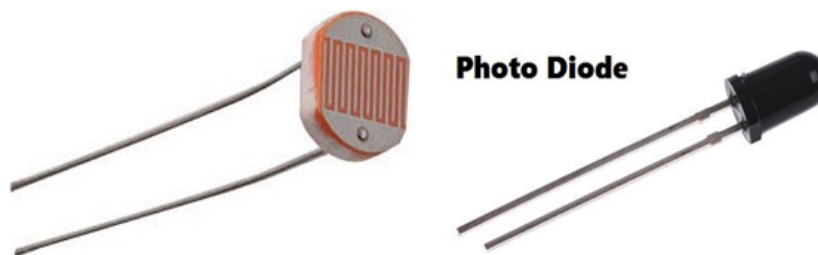


Figure 1.15 – (a) Photorésistance - (b) photodiode [14, 15].

1.6.3 L'humidité

L'humidité correspond d'un point de vue physique à la quantité de vapeur présente dans l'air. Généralement, la mesure de l'humidité fait référence à l'humidité relative exprimée

en pourcentage. L'humidité relative correspond au rapport entre la pression partielle de la vapeur d'eau et la pression de la vapeur saturante et ce pour un volume d'air et une température donnée [10]. On parle aussi d'humidité absolue comme étant le ratio masse d'eau et masse d'air. L'humidité peut être mesurée à l'aide d'un hygromètre, en voici quelques types [16] :

1. Hygromètre à cheveux.
2. Hygromètre à condensation.
3. Hygromètre capacitif.
4. Hygromètre à résistance



Figure 1.16 – Exemple d'hygromètre à cheveux [10] .

1.6.4 Humidité du sol

C'est la quantité d'eau dans le sol, exprimé en pourcentage. l'humidité du sol dépend de plusieurs facteurs représentés par la quantité de précipitations, la qualité du sol, la température de l'air et la quantité d'absorption par les plantes Le niveau d'eau dans le sol peut être mesuré par plusieurs moyens, le plus courant est le capteur d'humidité du sol.



Figure 1.17 – capteur d’humidité du sol utilisé avec Arduino.

1.6.5 Précipitations

Les précipitations qui peuvent prendre la forme liquide ou solide (pluie, brouillard, givre, grêle, neige, ...) sont le résultat de la condensation de la vapeur d’eau en altitude terre et qui finissent par tomber sur le sol par effet de gravité [10]. En général, lorsqu’on parle de précipitations, on parle de quantité de pluie tombée et cette dernière est mesurée par ce qu’on appelle un pluviomètre.

— Le pluviomètre :

Les mesures de précipitations peuvent être effectuées avec un pluviomètre, un instrument qui vous permet de comparer la quantité de précipitations qui tombe sur une zone particulière. Les pluviomètres actuels ont deux petits seaux qui contiennent généralement entre 0,2 ou 0,5 mm d’eau. Les précipitations sont mesurées par le nombre d’inclinaisons des pales et détectées par un système mécanique ou optique [11].

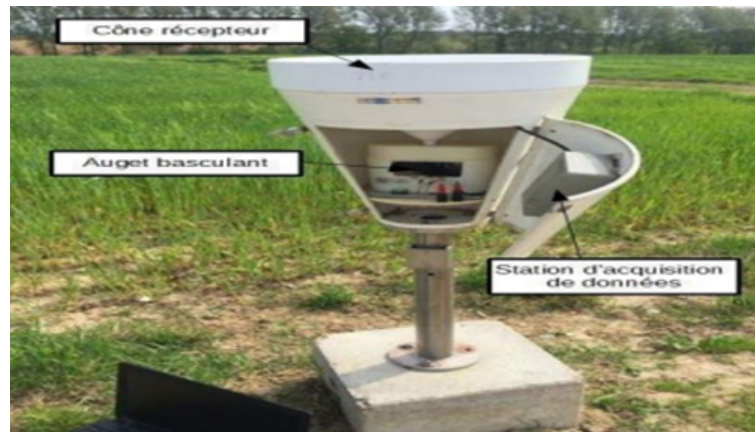


Figure 1.18 – Exemple de pluviomètre.

1.7 La problématique de notre projet

Les stations météo actuelles sur le marché, même si elles disposent de fonctions avancées, restent relativement chères, plus particulièrement dans le domaine de l'agriculture. L'objectif de ce travail est justement de fournir une solution simple, fiable et peu coûteuse permettant à n'importe quel agriculteur, même avec des moyens réduits d'acquérir ce dispositif. Pour réaliser ce projet, nous nous baserons sur une carte à base d'Arduino avec connexion internet qui permettra de récolter les données des capteurs, les stocker sur une base de données pour ensuite les afficher ensuite sur une application Android que nous aurons nous même créé.



Figure 1.19 – Schéma pour le principe de fonctionnement de notre système.

1.8 Conclusion

Dans ce chapitre nous avons présenté une station météo d'une manière générale, en commençant par une brève introduction avec un aperçu historique du secteur agricole et comment les conditions météorologiques affectent les rendements agricoles, puis nous avons parlé du développement de ces stations avec leurs différents types et variables mesurées par leurs appareils. Après cela, nous avons évoqué la problématique de notre projet qui est la réalisation d'une station météo de prévision des traitements des cultures. Dans le chapitre suivant, nous aborderons en détail les composants et les outils utilisés.

les outils matériel et logiciels pour la réalisation de notre station météo

Sommaire

2.1	Introduction	24
2.1.1	Concept de notre station météo :	24
2.2	La partie matérielle de la station météo	25
2.2.1	Carte Arduino	25
2.2.2	IDE (Integrated Développement Environnement) Arduino	26
2.2.3	Module Wifi ESP 8266	27
2.2.4	WeMos Lolin D1 mini :	28
2.2.5	Multiplexeur CD4051	32
2.2.6	Présentation des capteurs utilisés dans notre station météo	34
2.3	Partie logicielle de la station météo	38
2.3.1	Base de données Firebase	38
2.3.2	Applications Android	42
	Conclusion	48

2.1 Introduction

Après avoir exposé la problématique de notre travail dans le premier chapitre, nous allons présenter les différents composants constituant notre station météo et nécessaires à son bon fonctionnement. Nous commencerons ce chapitre par introduire la carte Arduino et le module Wifi ESP 8266 avant de détailler par la suite carte WeMos mini D1 utilisée dans notre projet. Les différents capteurs de température, humidité, humidité du sol, luminosité et pluie utilisés dans ce projet seront eux aussi exposés. En ce qui concerne la partie logicielle, nous présenterons l'outil de création d'application Android MIT App Inventor ainsi que l'outil de création et gestion de base de données Firebase de Google.

2.1.1 Concept de notre station météo :

Le concept que nous proposons pour la réalisation de cette station météo repose sur la base d'une carte Arduino disposant d'une connexion Wifi ESP8266, une base de données basée sur la plateforme Google Firebase et une application Android.

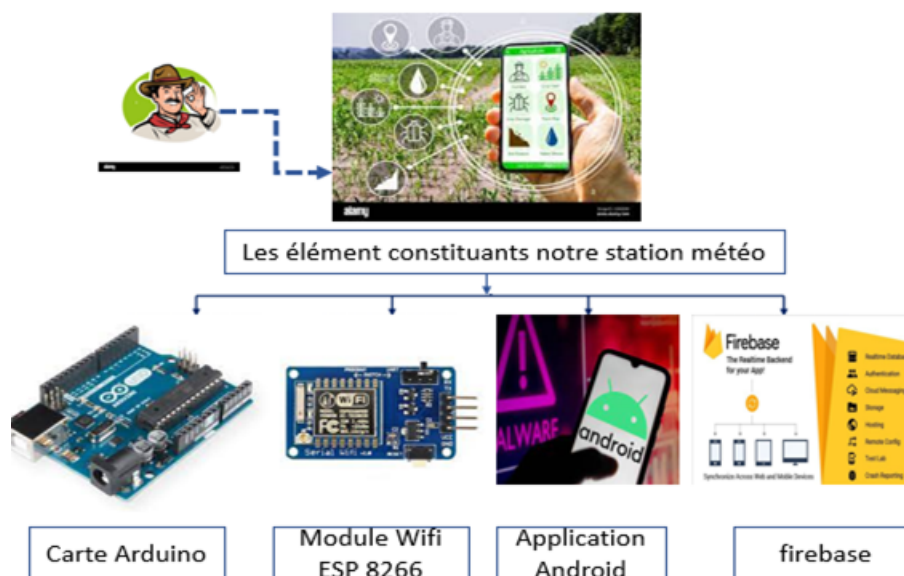


Figure 2.1 – Schéma synoptique de la station météo proposée.

L'objectif ici sera de brancher les différents capteurs de température, d'humidité, d'humidité sol, d'intensité lumineuse et de pluie sur la carte Arduino. Cette dernière

récoltera les informations provenant des capteurs avant de les envoyer vers une base de données Firebase que nous aurons créé. Cette base de données sert à synchroniser les données provenant de la carte Arduino avec l'application Android. En parlant d'application Android, nous utiliseront l'outil MIT App Inventor pour créer cette dernière.

Cette application devra récolter les informations reçues des différents capteurs, les analyser et notifier via une notification et un sms l'agriculteur sur les risques de maladies qui peuvent toucher son champ agricole, à partir de là il devra prendre une décision sur la conduite à adopter. Pour mieux comprendre le concept que nous proposons, nous avons scindé ce chapitre en deux parties : Matérielle et logicielle.

2.2 La partie matérielle de la station météo

Dans cette partie, nous présentons les différents composants électroniques constituant notre station météo.

2.2.1 Carte Arduino

Arduino est une entreprise qui conçoit, fabrique et soutient le développement des dispositifs électroniques et des logiciels, permettant à n'importe qui d'accéder facilement à des technologies avancées qui interagissent avec le monde physique [17]. Cette plateforme de développement est open source et intègre deux composants : La carte électronique et le logiciel IDE. Il existe une très grande variété de carte Arduino comme illustré sur la figure 2.2.

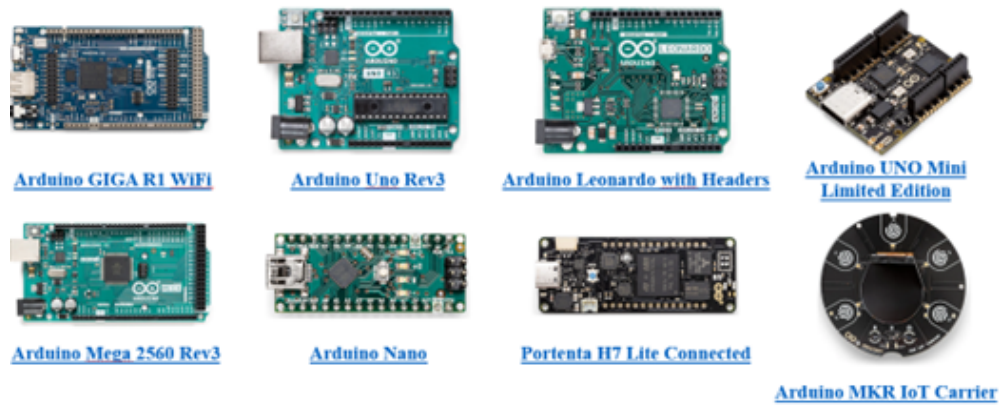


Figure 2.2 – Quelques exemples de cartes Arduino [17].

Dans notre cas, la présence d'une connexion Wifi pour faire fonctionner notre station météo nous a orienté vers le choix d'une carte Arduino intégrant un module Wifi ESP8266 comme nous le verrons par la suite.

2.2.2 IDE (Integrated Développement Environnement) Arduino

L'IDE d'Arduino est un logiciel open source et gratuit téléchargeable sur le site web d'Arduino. Il est utilisé pour écrire, compiler et télécharger un programme sur la carte physique. Ce logiciel comprend également d'autres fonctions comme le débogage et outils visuels. L'interface de l'IDE Arduino est illustrée par la figure 2.3.

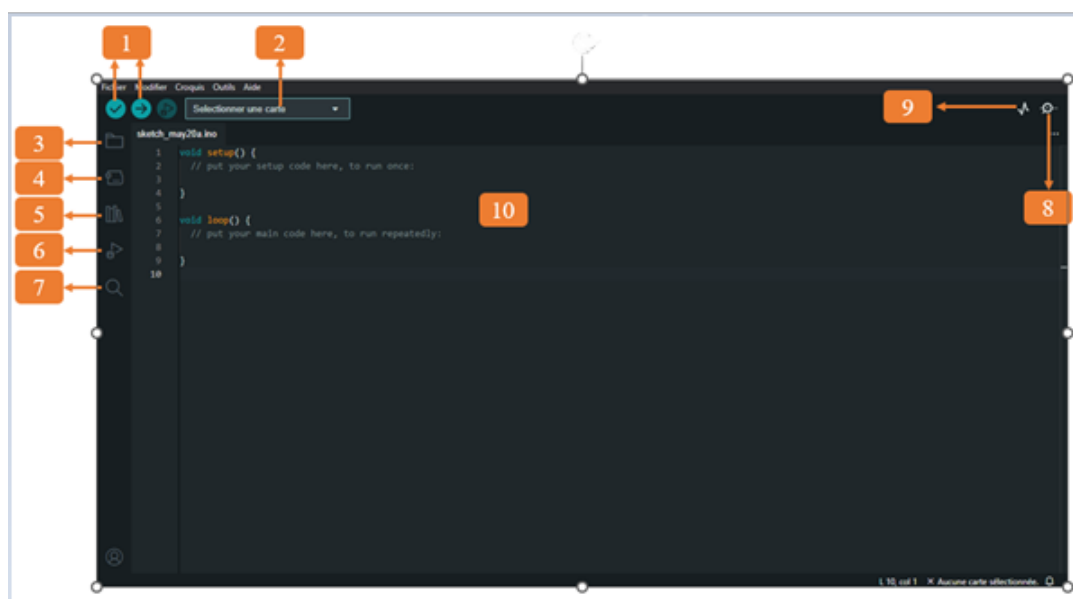


Figure 2.3 – Interface de l'IDE Arduino V2.10.

Description de l'interface	
1	Vérifier et téléverser le programme Arduino.
2	Accès rapide à la sélection de la carte Arduino utilisée.
3	Carnet de croquis (contient tous les programmes créés et sauvegardés).
4	Gestionnaire de cartes.
5	Gestionnaire de bibliothèque.
6	Débogueur.
7	Recherche dans le programme par mots clés.
8	Moniteur série.
9	Traceur de graphe série.
10	Zone d'écriture du programme Arduino

Table 2.1 – Description d'interface de l'IDE Arduino.

2.2.3 Module Wifi ESP 8266

Pour que deux appareils ou plus puissent communiquer entre eux, il faut qu'ils soient régis par les mêmes normes techniques. A cet effet, la norme IEEE 802.11 définit les protocoles qui permettent de mettre en communication un ou plusieurs appareils sans fil actuels compatibles Wi-Fi [18]. Les normes IEEE ont été certifiées par la Wifi Alliance, une organisation à but non lucratif créée pour certifier les produits IEEE 802.11 et les promouvoir en tant que normes de réseau local sans fil [19].

Dans notre travail, une connexion Wifi est nécessaire pour transmettre les informations des capteurs via la carte Arduino vers la base de données Firebase et l'application Android. Dans ce cas, plusieurs possibilités s'offrent à nous, soit partir sur une carte Arduino standard et lui rajouter un module Wifi, soit partir sur une solution à base d'Arduino intégrant un module WiFi.



Figure 2.4 – Exemple de modules Wifi ESP 8266 et cartes à base d’Arduino intégrant ce dernier.

Dans ce travail, nous avons choisi d’utiliser une carte électronique basée sur la plateforme Arduino et intégrant un module Wifi ESP 8266. Cette carte en question est WeMos Lolin D1 mini.

2.2.4 WeMos Lolin D1 mini :

Wemos est une plateforme de développement matérielle et logicielle et dont la programmation peut être faite en utilisant l’IDE Arduino. Wemos conçoit une variété de carte électroniques pour différentes utilisations. Dans notre travail, nous nous intéressons à la carte Wemos D1 mini qui intègre un module Wifi ESP8266. La figure 2.5 qui suit montre une description de cette carte.[20]

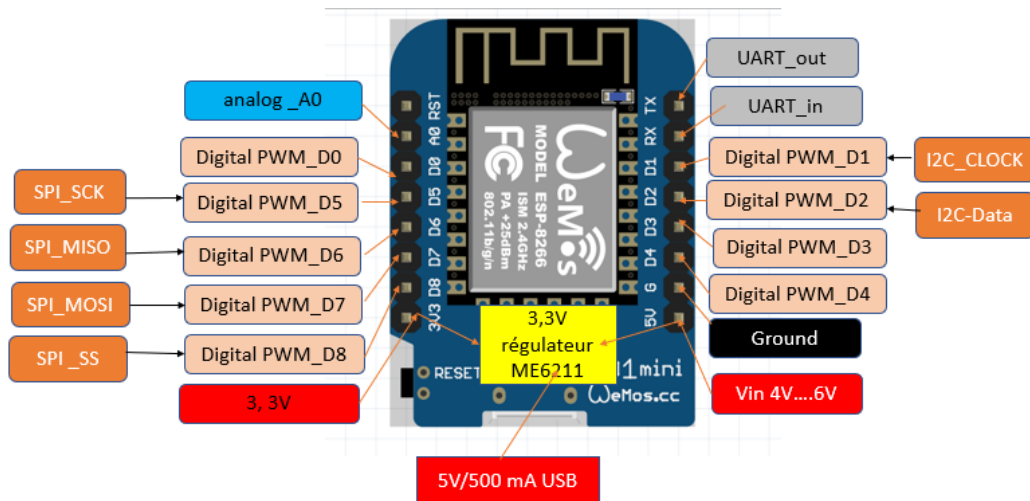


Figure 2.5 – Carte Wemos D1 mini [20].

La carte Wemos D1 mini comme le montre la figure 2.8 dispose de 16 broches réalisant chacune une fonction données. Le tableau qui suit résume les caractéristiques techniques de ces broches.

Broche	Fonction	Broche ESP-8266
TX	Données série TX, broche de transmission TXD	TXD
RX	Données série RX, broche de réception	RXD
A0	3,3V pour une entre analogique	A0
D0	IO (activer le boot loader)	GPIO16
D1	IO, SCL - signal d'horloge du bus I2C	GPIO5
D2	Signaux de données du bus IO, SDA - I1C	GPIO4
D3	IO, résistance de rappel GPIO0 10k	GPIO0
D4	E/S, résistance pullup 10k, LED interne (BUILTIN_LED), allumée au niveau bas	GPIO2
D5	E/S , SPI SCK	GPIO14
D6	E/S, SPI MISO	GPIO12
D7	E/S SPI MOSI	GPIO13
D8	IO, résistance pull-down 10K, SPI SS	GPIO15
G	Masse	GND
5V	Alimentation 5V connectée au régulateur 3.3V	(aucun)
3V3	3.3V	3V3
RST	réinitialisé. Allongez-vous sur le sol pour réinitialiser.	RST

Table 2.2 – Tableau représentant les différentes fonctions des broches de la carte Wemos D1 mini [20].

La carte Wemos D1 mini comme toutes les cartes intégrant un module ESP8266 nécessitent l'installation de bibliothèques afin qu'elles puissent être utilisées avec l'IDE Arduino et fonctionner correctement.

— Installation des bibliothèques ESP8266 dans l'IDE Arduino

Après avoir ouvert l'IDE Arduino, la première étape consiste à aller directement dans l'onglet Fichier Préférences (figure 2.6).

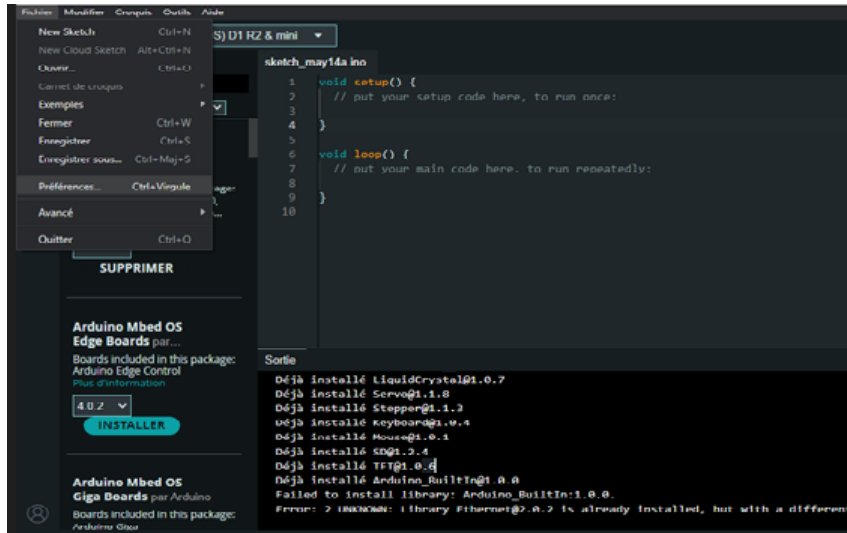


Figure 2.6 – Sélection de l'onglet préférences pour ajouter le lien d'installation des bibliothèques ESP8266.

Il suffit ensuite d'aller dans la zone URL de gestionnaire de cartes supplémentaires et coller le lien suivant et cliquer sur OK :

<http://arduino.esp8266.com/stable/packageesp8266comindex.json>

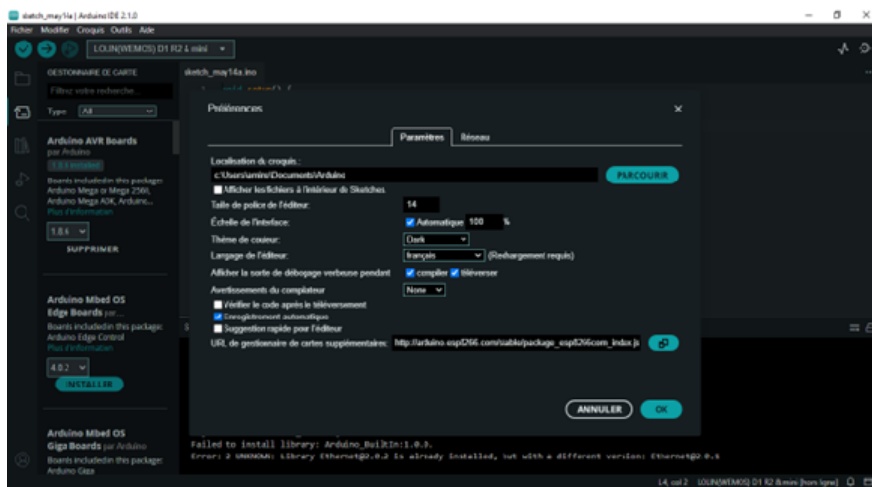


Figure 2.7 – Introduction du lien d'installation des bibliothèque ESP8266 dans l'IDE Arduino.

Cette étape terminée, alors nous n’aurons plus qu’à aller sélectionner la carte Wemos D1 mini dans le gestionnaire de cartes (figure 2.8) ou directement dans la zone de sélection de cartes (figure 2.9).

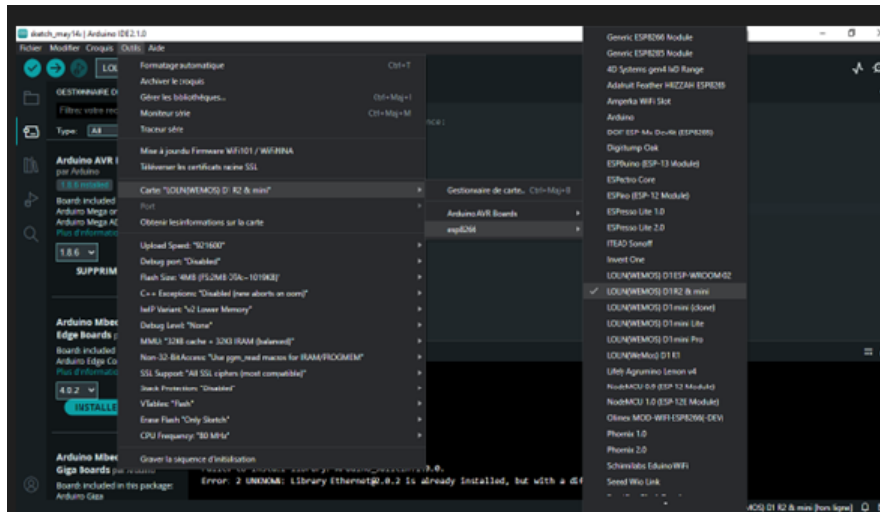


Figure 2.8 – Sélection de la carte Wemos D1 mini depuis le gestionnaire de cartes.

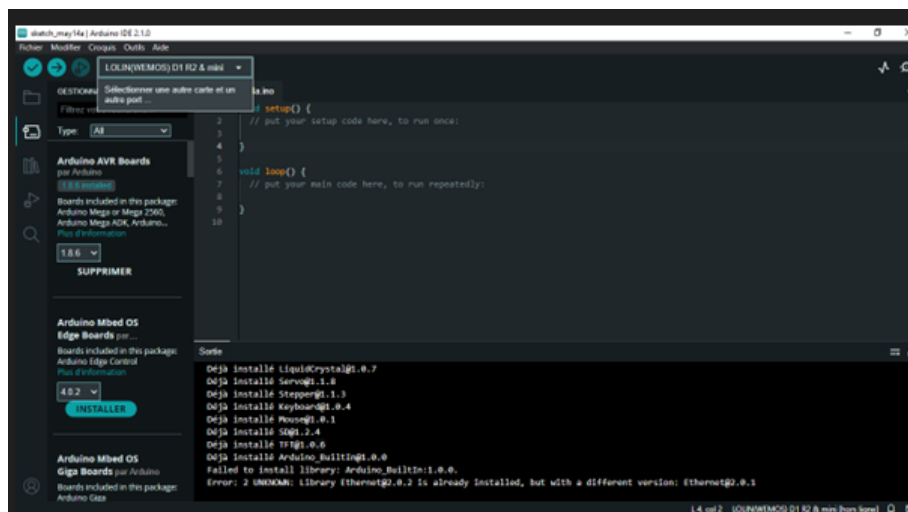


Figure 2.9 – Sélection de la carte Wemos D1 mini depuis une zone dédiée dans la nouvelle version de l’IDE Arduino.

2.2.5 Multiplexeur CD4051

Un multiplexeur est un circuit intégré permettant de démultiplier le nombre d’entrées / sorties analogiques ou numériques dans le cas où celles-ci sont limitées. Le CD4051 est multiplexeur / démultiplexeur à 8 canaux analogiques. Dans notre travail, c’est la fonction multiplexeur du CD4051 qui sera utilisée.

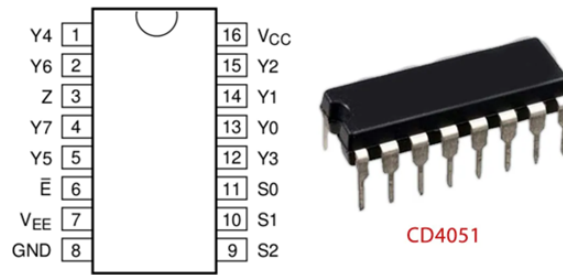


Figure 2.10 – Circuit multiplexeur CD4051 et ses broches [21].

Broche	Nom de la broche	Description
1	Y4	Sortie 4 du multiplexeur/démultiplexeur
2	Y6	Sortie 6 du multiplexeur/démultiplexeur
3	Z	Broche d'entrée/sortie commune à relier à l'Arduino par exemple
4	Y7	Sortie 7 du multiplexeur/démultiplexeur
5	Y5	Sortie 5 du multiplexeur/démultiplexeur
6	E'	Broche d'activation
7	VEE	Broche d'alimentation négative
8	GND	masse
9	A2	Broche d'adresse 2
10	A1	Broche d'adresse 1
11	A0	Broche d'adresse 0
12	Y3	Sortie 3 du multiplexeur/démultiplexeur
13	Y0	Sortie 0 du multiplexeur/démultiplexeur
14	Y1	Sortie 1 du multiplexeur/démultiplexeur
15	Y2	Sortie 2 du multiplexeur/démultiplexeur
16	VCC	Alimentation

Table 2.3 – Les broches du multiplexeur CD4051.

Dans notre cas, l'utilisation de ce circuit CD4051 est indispensable, car la carte Wemos D1 mini dispose d'une seule entrée analogique alors que nous avons besoin nous de trois entrées analogiques pour nos différents capteurs.

2.2.6 Présentation des capteurs utilisés dans notre station météo

Un capteur est un dispositif permettant de transformer une grandeur physique en une grandeur utilisable, le plus souvent électrique.

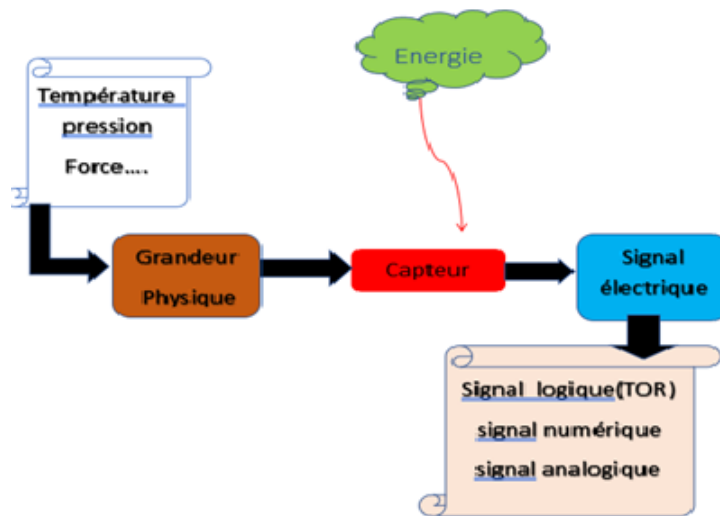


Figure 2.11 – Principe de fonctionnement d'un capteur.

Dans notre travail, nous aurons besoin de quatre capteurs pour cinq grandeurs physiques à analyser. C'est ce que nous allons présenter dans ce qui suit.

2.2.6.1 Photorésistance

Une photorésistance est le dispositif dont la particularité est que sa résistivité et par conséquent sa résistance varie en fonction de l'intensité de la lumière, plus celle-ci est élevée plus la résistivité décroît. Ses principales caractéristiques sont :

- Alimentation : 3 à 15 Vcc.
- Dimensions : Ø 5 x 5,3 mm.

- Nécessite une résistance supplémentaire de 1 K à 10 K (non incluse).

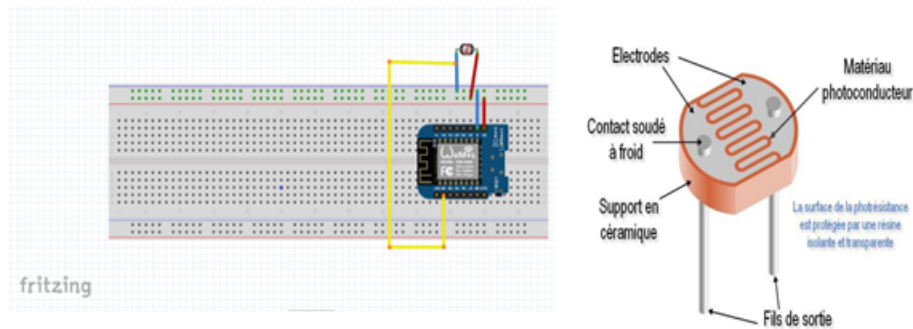


Figure 2.12 – Photorésistance avec son câblage sur Arduino [10].

2.2.6.2 Capteur de Température et d'Humidité

Il existe plusieurs types de capteurs de température et d'humidité pouvant être utilisés avec Arduino, tels que (le DHT22, DHT11 et DHT21). Nous avons dans notre travail opté pour le DHT22 (également connu sous la référence AM2302) car sa plage de mesure est la plus importante parmi les types que nous avons cités. Ce capteur peu coûteux peut détecter la température et l'humidité ambiantes de manière numérique l'aide d'un capteur d'humidité capacitif et d'une thermistance. Les principales caractéristiques du DHT22 sont :

- Faible coût.
- Tension d'alimentation : 3V à 5V (alimentation et données).
- Consommation : 2,5mA max lors de la conversion.
- Plage de mesure de l'humidité : de 0 à 100% avec une précision à 2-5% [10].
- Plage de mesure de la température : de -40°C à 80°C avec une précision $\pm 0.5^\circ\text{C}$.
- Échantillonnage à 0.5 Hz c'est à dire une mesure toute les 2 secondes.

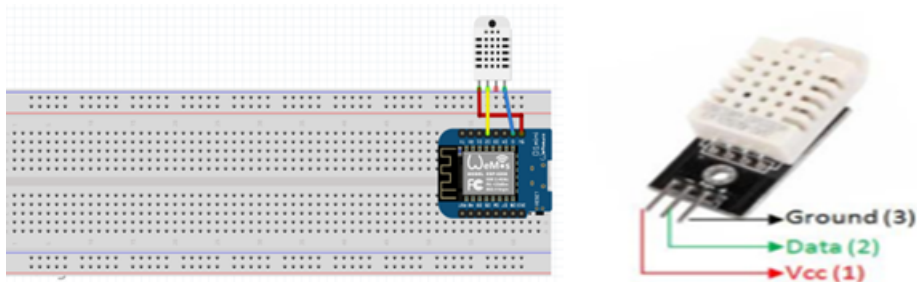


Figure 2.13 – Image réelle du capteur de Température et d’Humidité DHT22 avec son câblage avec Arduino [10].

2.2.6.3 Capteur de pluie

Ce capteur est idéal pour détecter s’il pleut, en particulier dans des projet de domotique pour les agriculteurs. Le capteur est placé avec un module d’interface disposant d’un potentiomètre pour le réglage de la sensibilité Utilisé pour choisir entre une sortie numérique et une sortie analogique. Le capteur de pluie détecte les gouttelettes d’eau lorsque la structure en forme de peigne est en court-circuit. Les principales caractéristiques des capteurs sont [22] :

- Taille du capteur : 39 mm x 54 mm.
- Tension : 3,3 V-5 V.
- LED de contrôle.

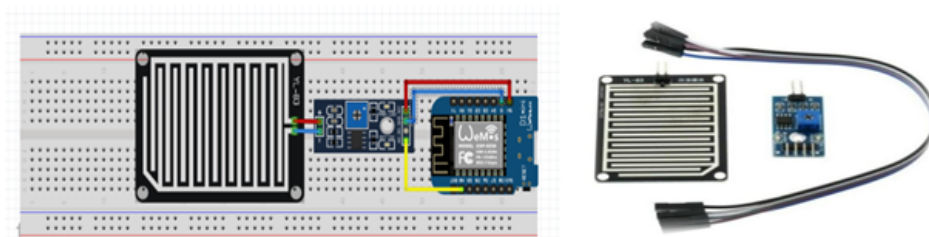


Figure 2.14 – Capteur de détection de pluie avec son câblage avec Arduino.

Ce capteur avec son module d’interface dispose de quatre broches :

- Vcc à connecter à la broche 3.3 ou 5V de l’Arduino.
- GND à connecter à la broche GND de l’Arduino.

- Broches D0 et A0 à connecter respectivement à une des broches numériques ou analogiques de l'Arduino.

2.2.6.4 Capteur d'humidité du sol

Le capteur d'humidité du sol se compose de deux sondes dont le rôle est de mesurer la teneur en eau du sol par le biais d'une résistance. Ces deux sondes permettent de faire passer plus ou moins de courant selon la quantité d'eau disponible dans le sol. Plus cette quantité est élevée, plus le sol conduit l'électricité et donc la résistance est faible. Plus cette quantité est faible, plus le sol conduit peu d'électricité et par conséquent la résistance est plus importante [23]. Les spécifications de ce capteur d'humidité du sol sont :

- Tension d'entrée : 3.3 à 5V.
- Tension de sortie : 0 à 4.2V.
- Courant d'entrée : 35mA.

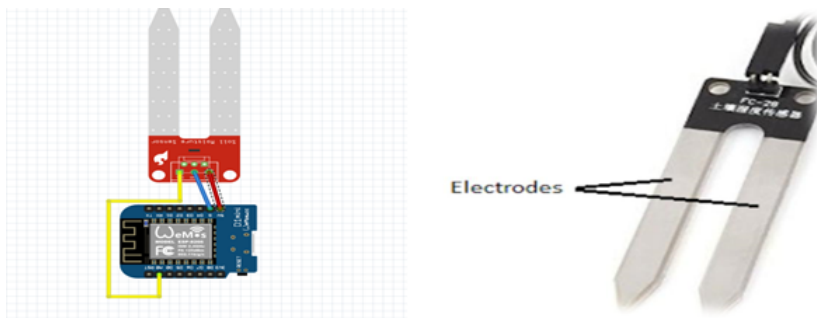


Figure 2.15 – Capteur d'humidité du sol avec son câblage.

Ce capteur dispose de trois broches :

- Vcc connectée avec la broche 3.3V ou 5V de l'Arduino.
- GND à connecter à la broche GND de l'Arduino.
- A0 à connecter la broche analogique A0 de l'Arduino.

2.3 Partie logicielle de la station météo

La partie logicielle de notre travail se compose en fait de deux parties : l'application Android et la base de données Firebase.

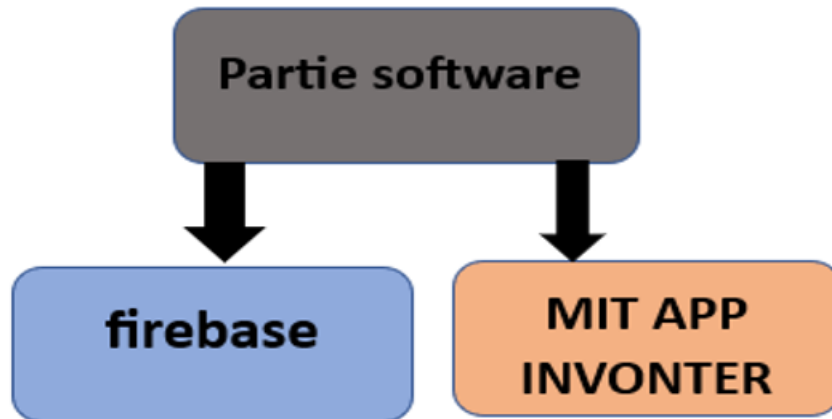


Figure 2.16 – Composants de la partie logicielle de notre station météo.

2.3.1 Base de données Firebase

Firebase est une plateforme mobile créée par google fournissant un certain nombre d'outils aidant la création et développement rapide d'applications web et mobiles [24]- [25]. Firebase est caractérisé comme un service BAAS (backend-as-a-service) et est classé dans la catégorie des bases de données programmes NoSQL qui ont la particularité de stocker les données sur des fichiers JSON pour les exploiter.



Figure 2.17 – Le logo Firebase de Google [26].

Pour pouvoir utiliser la base de données Firebase, il faut d'abord se rendre à l'adresse web suivante : <https://firebase.google.com/>. Ce lien nous emmène vers la page d'accueil de Firebase comme le montre la figure suivante.

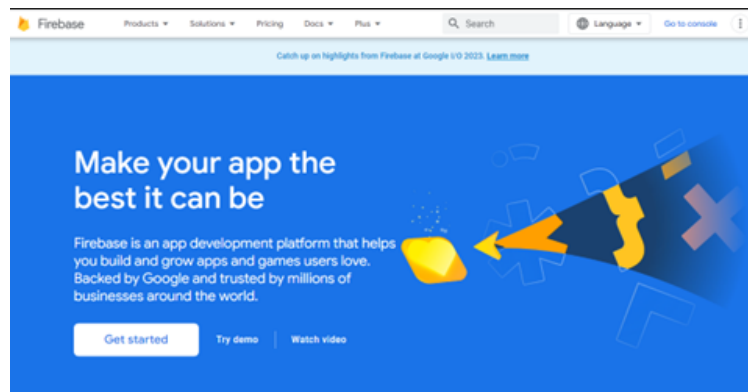


Figure 2.18 – fenêtre d'accueil de Firebase.

Pour pouvoir utiliser ce service, il faut impérativement avoir créé un compte dans Firebase en utilisant une adresse mail Gmail. Il suffira alors ensuite de cliquer sur Get Started, une deuxième fenêtre alors s'ouvre, il faudra ensuite cliquer sur ajouter un projet en spécifiant le nom du projet. En cliquant sur continuer, on peut choisir d'activer Google Analytics ou non et cliquer sur continuer. Enfin, nous choisissons notre compte et nous cliquons sur créer.

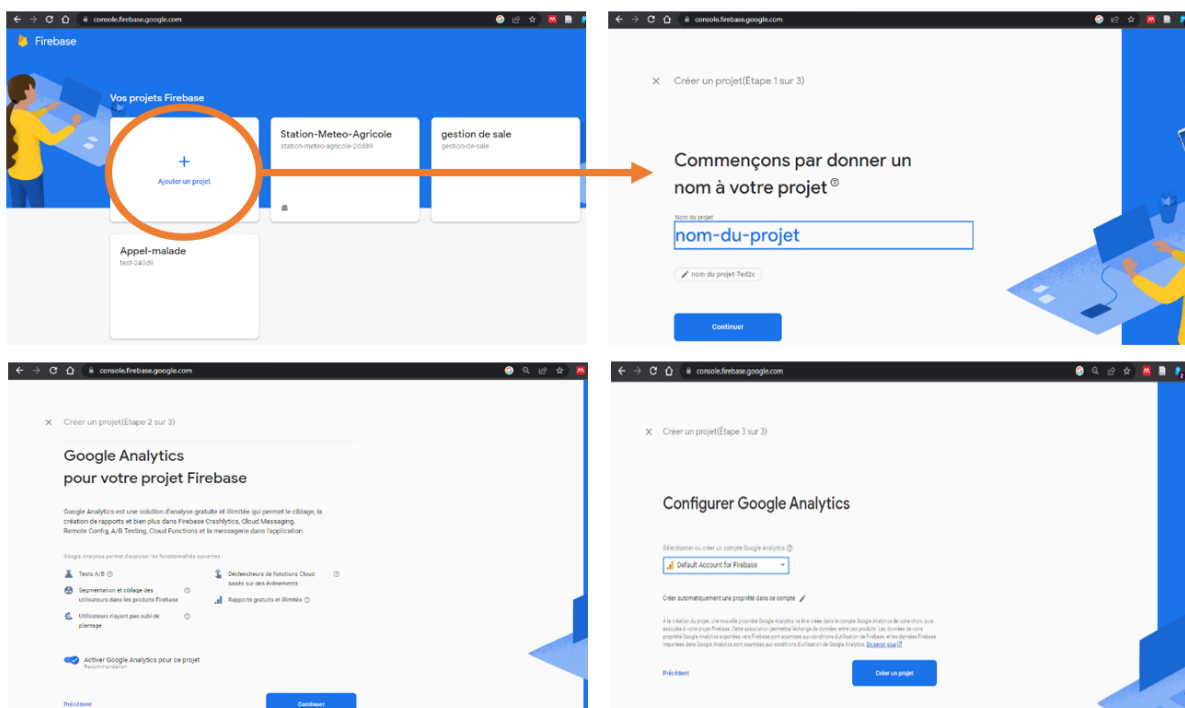


Figure 2.19 – Etape de création de projet dans Firebase.

La figure 2.20, qui suit montre le projet PFESTATION que nous avons créé dans Firebase pour stocker notre base de données.

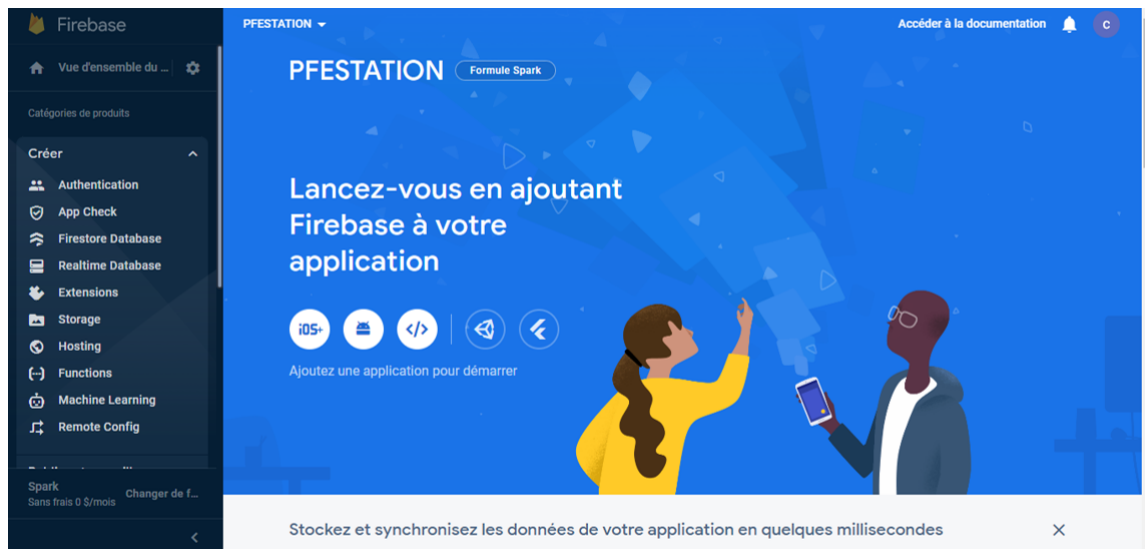


Figure 2.20 – Fenêtres du projet PFESTATION créé sur Firebase.

A partir de la fenêtre affichée (figure 2.20), nous cliquons sur Realtime database (base de données en temps réelle) et après nous cliquons sur créer une base de données.

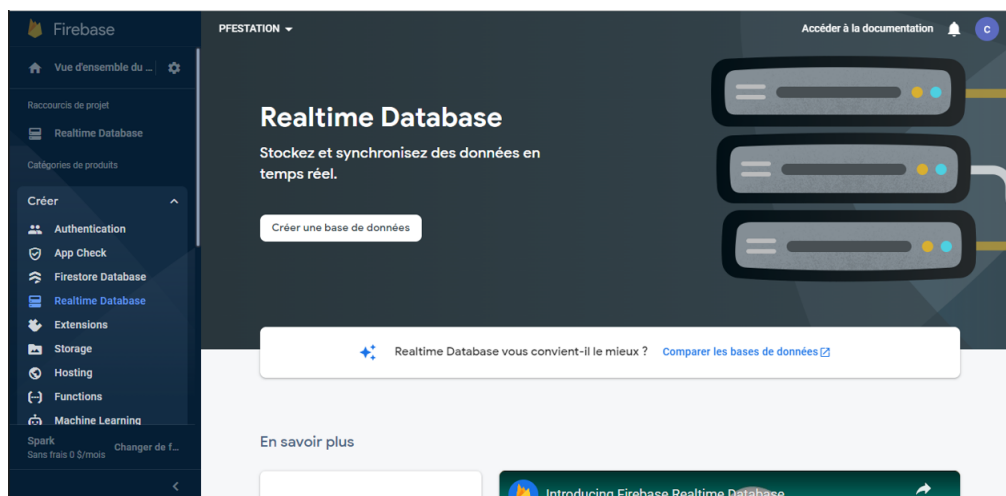


Figure 2.21 – Fenêtre de création de base de données dans Firebase dans le projet PFESTATION.

Cette dernière est une base de données stockées dans le cloud de Google. Les données sont sauvegardées dans un fichier JSON et sont synchronisées en permanence avec tous les utilisateurs connectés. Lorsqu'on crée par exemple une application mobile Android, tous les clients utilisant cette application partagent une instance de base de données et reçoivent en temps réel les données actualisés. La base de données créée contient le lien de la base et la clé d'authentification qui doivent être impérativement être récupérées

pour être utilisé non seulement dans le programme Arduino mais aussi dans l'application Android. Le lien est lui récupéré directement dans la fenêtre Realtime Database :

<https://pfestation-default-rtdb.firebaseio.com/>

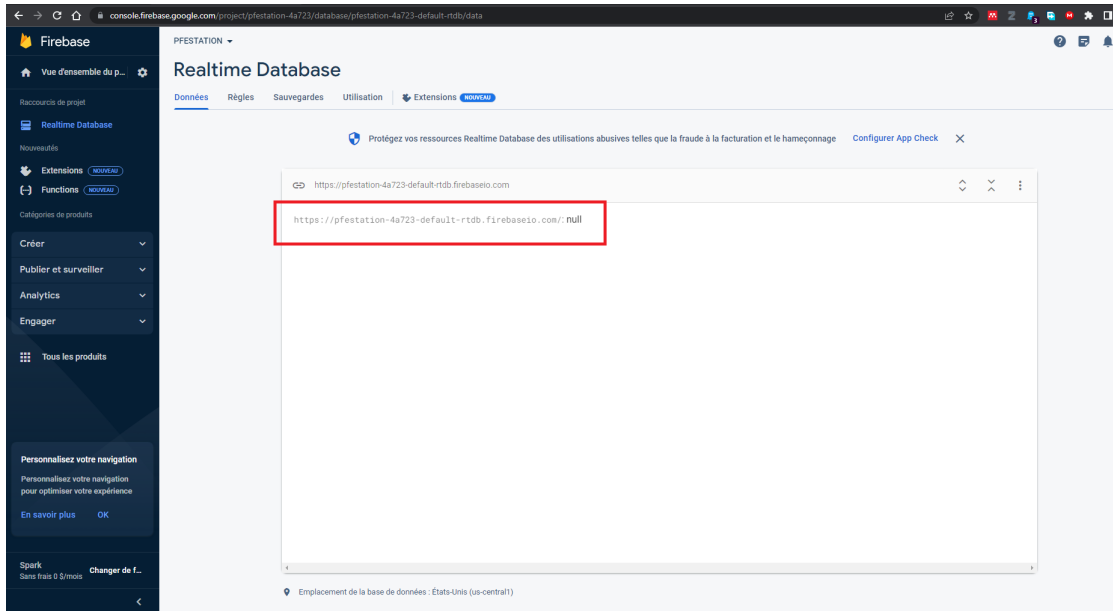


Figure 2.22 – Lien de la base de données.

Le code d'authentification quant à lui est récupéré en allant dans paramètre de projet, ensuite comptes de services et sélectionner codes secrets de la base de données.

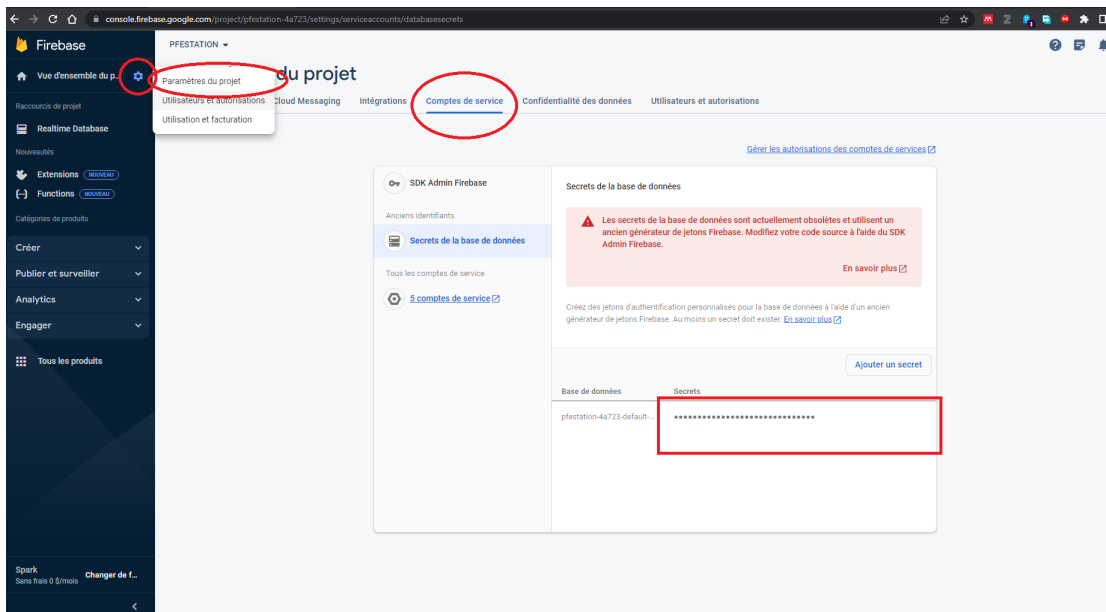


Figure 2.23 – Clé d'authentification de la base de données..

2.3.2 Applications Android

Android est un système d'exploitation mobile développé et maintenu jusqu'à aujourd'hui par Google. Il est principalement utilisé sur des appareils à écran tactile tels que les téléphones mobiles Et tablettes, téléviseurs, voitures, montres/bracelets connectés, etc... [27]. Les applications Android peuvent être créées en utilisant des outils divers tels que Andoid studio, MIT App inventor, etc... Dans ce travail, nous avons opté pour l'outils de MIT pour sa rapidité et simplicité de développement.

2.3.2.1 Présentation de l'outil MIT App inventor

App Inventor est un outil de développement d'applications Android en ligne gratuit pour téléphones et tablettes. Contrairement à d'autres outils de développement d'applications Android comme Android studio, il présente la particularité d'être simple, rapide, intuitif et basé sur une programmation graphique (par blocs) plutôt qu'en lignes d'instructions. Cette plateforme a été initialement développée par Google mais est aujourd'hui maintenue par le MIT [28].



Figure 2.24 – Page d'accueil de la plateforme MIT App Inventor.

Pour créer une application Android avec l'outil MIT App Inventor, nous allons cliquer sur <Create Apps>. A partir de là, la plateforme vous demande de vous connecter grâce

à votre compte. Si vous n'avez aucun compte alors il faudra en créer un avant de pouvoir utiliser la plateforme.

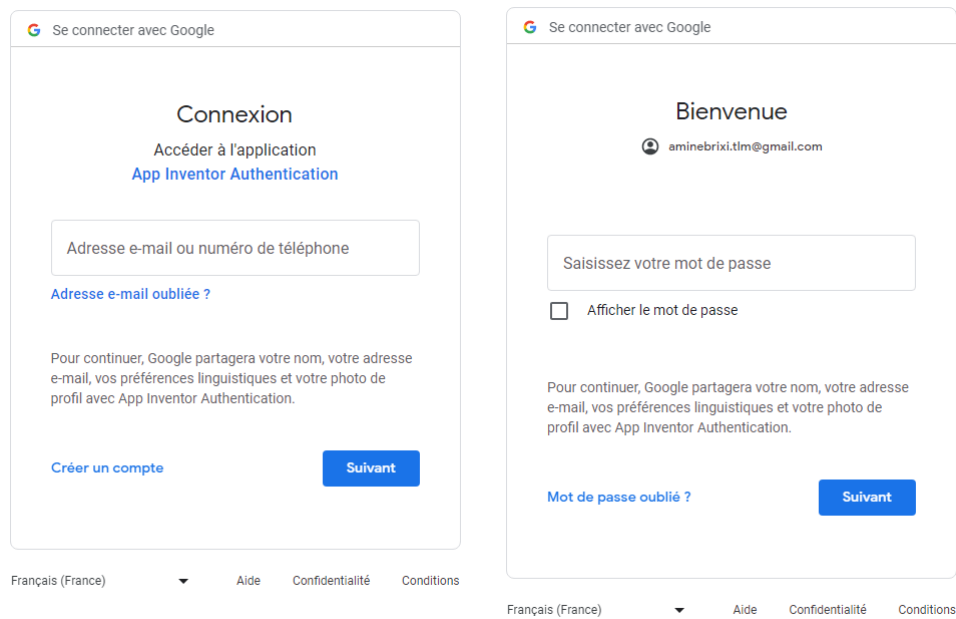


Figure 2.25 – Compte d'accès à la plateforme MIT App Inventor.

L'environnement de MIT App inventor est composé deux parties principales : designer et blocks.

1. Partie Designer : C'est dans cette partie qu'est programmée l'interface de l'application Android.

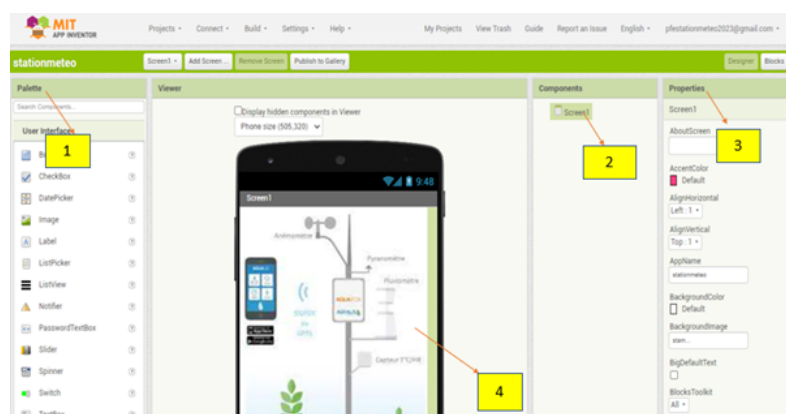


Figure 2.26 – Fenêtre de la partie Designer dans MIT App Inventor.

Cette partie se compose de quatre principaux volets :

Description	
1	Palette : C'est dans cette zone qu'on trouve tous les composants qu'on peut ajouter à notre application.
2	Composants : C'est dans cette partie que sont affichés tous les composants ajoutés à notre application.
3	Propriétés : représente les propriétés du composant sélectionné dans la zone composants.
4	Viewer : Permet de visualiser l'interface de l'application et les composants qui lui sont ajoutés.

Table 2.4 – Description de chaque composant de la partie Designer.

2. Partie Blocks : Cette partie nous permet de programmer les composants ajoutés dans la partie Designer.

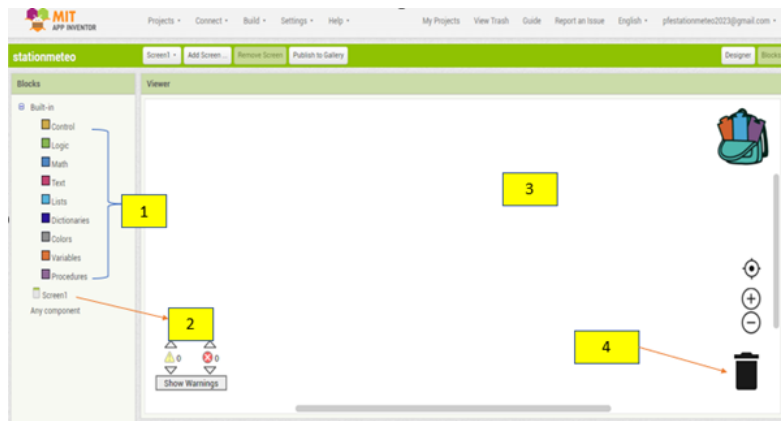


Figure 2.27 – Fenêtre de la partie Blocks dans MIT App Inventor.

Cette partie se compose de quatre principaux volets :

Description	
1	Blocs prédéfinis sur MIT App inventor (logique, mathématique, boucles, tests, variable, ...)
2	Blocs représentant les fonctions des composants utilisés dans l'interface de l'application.
3	L'espace de travail.
4	Corbeille.

Table 2.5 – Description de chaque composant de la partie Blocks.

Pour tester une application, nous disposons sur la plateforme MIT App inventor de plusieurs outils comme le montre la figure 2.6.

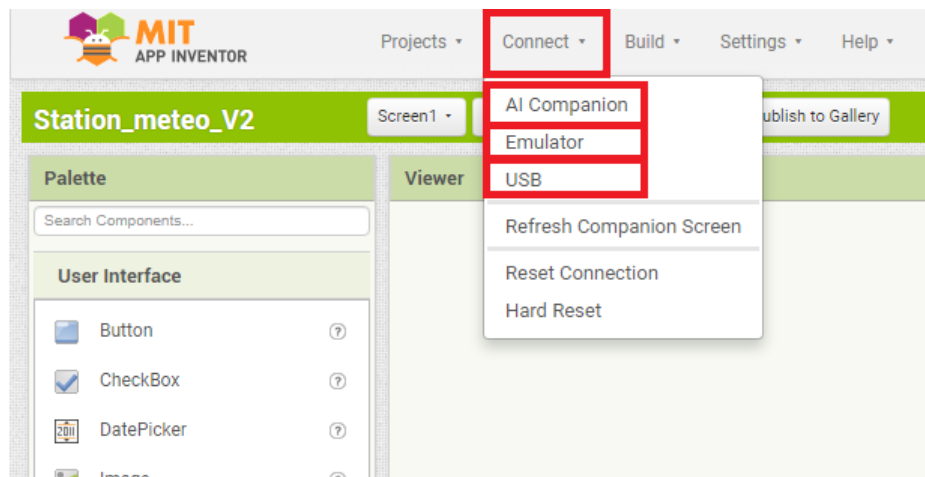


Figure 2.28 – Options de tests d'applications disponibles sur MIT App Inventor.

- Emulateur : Cet outil a pour rôle de rendre possible le test de l'application sur un émulateur nommé AiStarter installé sur ordinateur.

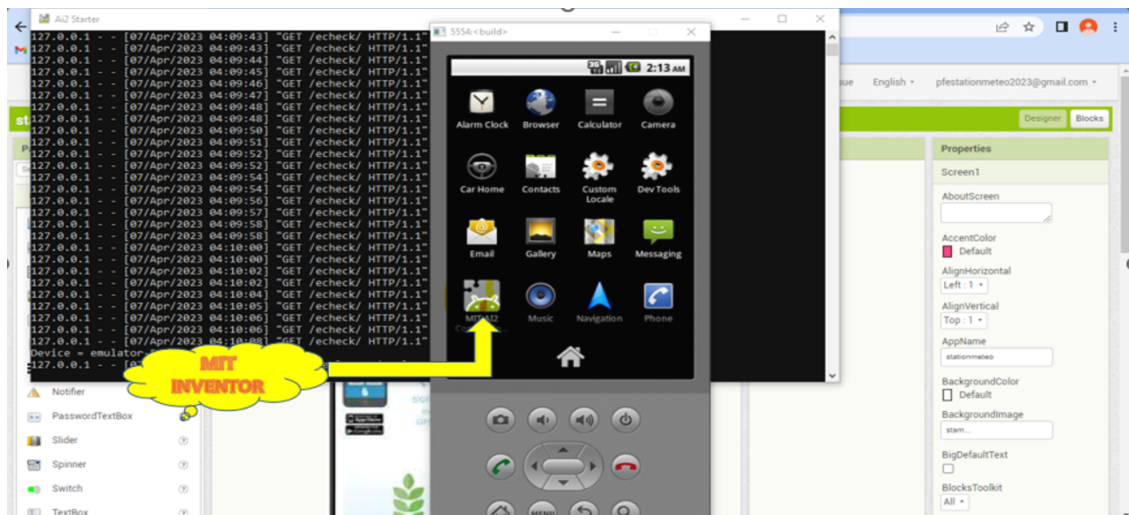


Figure 2.29 – Emulateur de MIT App Inventor.

- AI Companion : En utilisant cet outil, nous avons la possibilité de visualiser en temps réel les modifications apportées à notre application et ce sans installer l'application sur le smartphone.

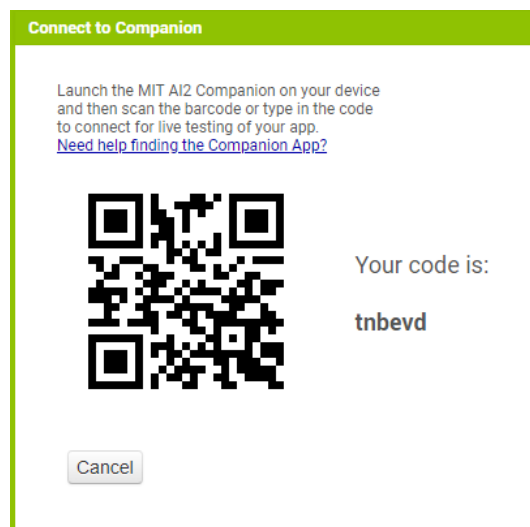


Figure 2.30 – Code QR généré par MIT App Inventor pour se connecter à AI Companion.

- USB : Dans ce cas, le smartphone sera connecté à l'ordinateur via câble USB et l'application pourra être testée en utilisant l'outil AiStarter.

Si l'application testée fonctionne correctement, alors nous pouvons passer à l'étape suivante qui est l'installation de cette dernière sur le smartphone. Pour cela, il existe deux possibilités :

- En scannant un code QR généré par la plateforme MIT App Inventor en utilisant l'application MIT AI2 Companion installée sur smartphone.
- En téléchargeant le fichier avec l'extension APK sur son ordinateur et le copier ensuite sur le smartphone avant de l'installer. Notons par ailleurs que cette opération peut nécessiter des autorisations lors de son installation selon la version Android du smartphone.

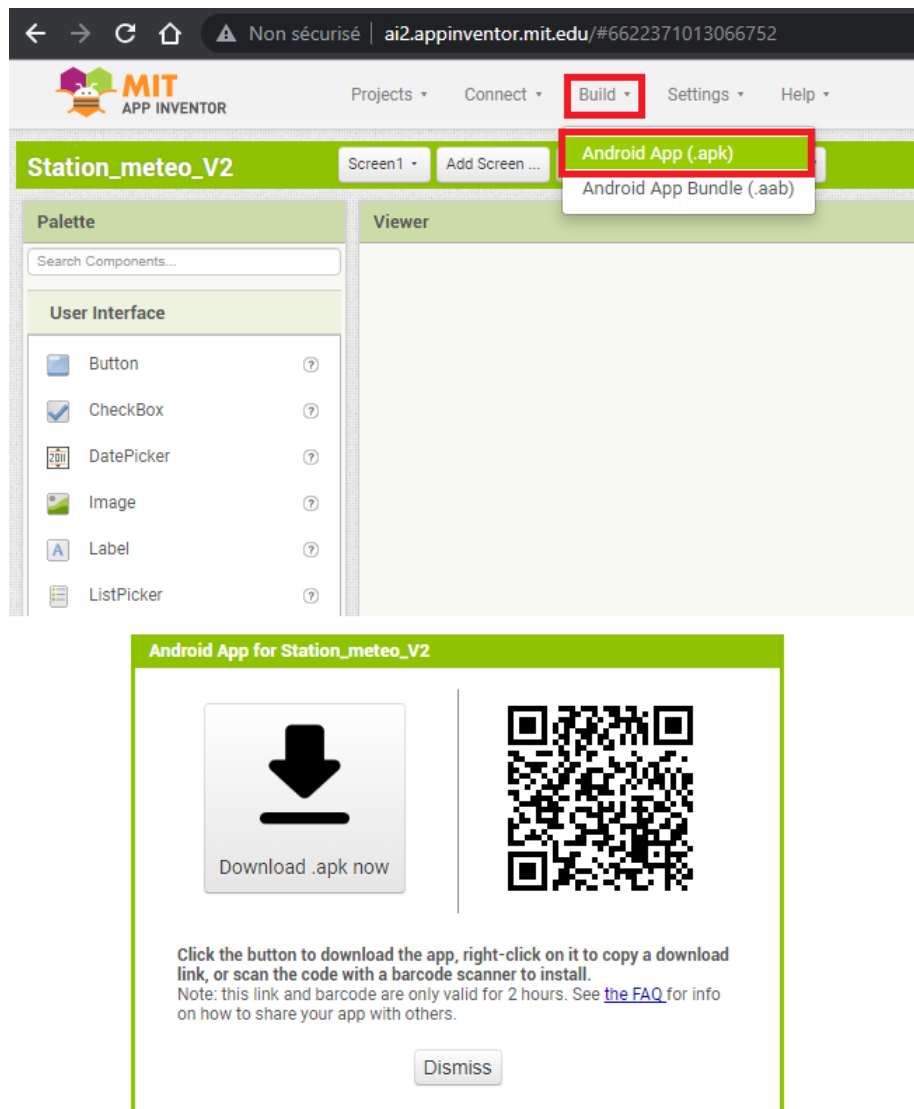


Figure 2.31 – Code QR généré par MIT App Inventor pour télécharger l'application installable en extension apk.

Conclusion

Dans ce chapitre, nous nous sommes focalisés sur la présentation des différents composants de notre station météo. Nous avons divisé ce chapitre en deux parties pour plus de clarté : une partie matérielle et une partie logicielle. Dans la partie matérielle, nous avons commencé par introduire Arduino et le module WiFi ESP8266 avant de se concentrer sur la carte Wemos D1 mini qui se base sur Arduino et dispose d'un module ESP8266 intégré. Nous avons ensuite présenté les capteurs et le multiplexeur CD4051 utilisés dans notre travail. Dans la seconde partie, nous nous sommes focalisés sur l'application Android et l'outil Firebase. Nous avons ainsi montré comment créer une application Android en utilisant l'outil MIT App Inventor. Nous avons justifié ce choix par le fait que ce dernier était simple, intuitif et rapide dans son utilisation. Nous avons aussi montré comment créer une base de données récoltant les données de nos capteurs en utilisant l'outil de Google Firebase.

Dans le chapitre suivant, nous allons exploiter toutes les notions présentées ici pour entamer la réalisation de notre station météo.

Conception et réalisation D'une station météo

Sommaire

3.1	Introduction	50
3.2	Description de notre station météo	50
3.3	Circuit électronique de notre station météo	51
3.3.1	Programmation de la carte WemosD1 mini avec l'IDE Arduino	52
3.4	Base de données	57
3.5	L'application Android « Weather and farming »	58
3.5.1	Partie Designer	59
3.5.2	Partie Blocks :	60
3.6	Tests pratique de notre station météo	65
3.7	Conclusion :	71

3.1 Introduction

Dans ce dernier chapitre, nous allons présenter la station météo réalisé dans ce travail. le circuit électronique mise en œuvre et l'application « Weather and farming » pour le traitement des données de la station météo, nous commencerons par le circuit électronique avant de présenter notre application, et nous finirons ce chapitre par des tests pratiques. Démonstration du principe de fonctionnement de notre station météo.

3.2 Description de notre station météo

La figure 3.1 montre le schéma synoptique décrivant le principe de fonctionnement de notre station météo. Cette dernière se base sur une carte Wemos D1 mini connectée en WiFi qui permet de récolter les données collectées par les différents capteurs (DHT22, LDR, capteur de pluie, Humidité du sol). Ces données en question sont instantanément synchronisées sur la base de données Firebase que nous avons créé puis sur l'application Android pour les afficher sur un smartphone. Lorsque les données récoltées par les capteurs ne dépassent pas un certain seuil qui est défini sur l'application (dépend de la maladie), l'application Android « Weather and farming » continue à afficher les données sans plus sur un fond vert. Dès qu'un seuil en relation avec une maladie est dépassé, par exemple température ou humidité trop élevés, l'application émet une notification à l'utilisateur et envoie un sms aux numéros de téléphones définis dans l'application, l'affichage des données dans ce cas bascule sur un fond rouge. Ceci a pour rôle d'alerter l'agriculteur ou l'ingénieur agronome sur le risque accru de développement d'une ou plusieurs maladies. Afin d'aider l'agriculteur ou l'ingénieur agronome dans son processus de traitement des cultures, il lui suffira de cliquer sur la notification, une fenêtre s'ouvre dans l'application Android montrant une liste de maladies et les mesures à adopter avec ces dernières.

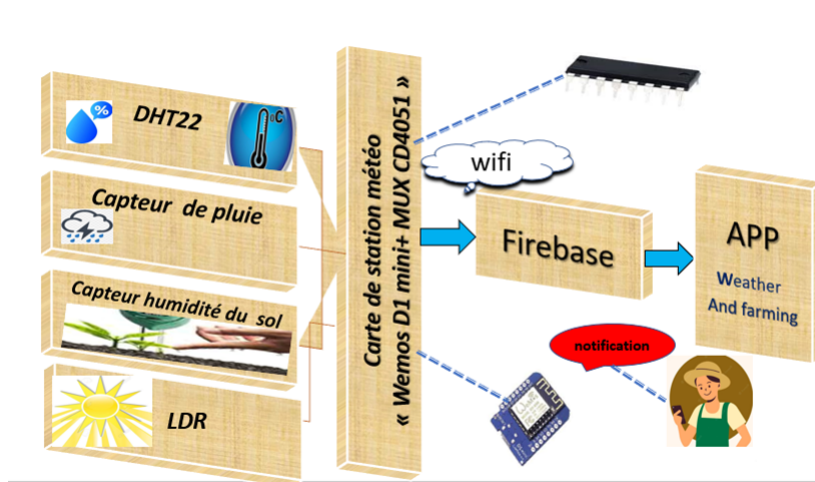


Figure 3.1 – Schéma synoptique de notre station météo.

Dans ce qui suit, nous allons expliquer le circuit électronique à base de carte Wemos D1 mini réalisé avec les différents brochages des capteurs, le détails du programme injecté dans la carte Wemos et comment synchroniser les données collectées par cette dernière sur la base de données Firebase

3.3 Circuit électronique de notre station météo

Comme mentionné le précédent chapitre, la partie électronique de notre station météo se compose d'une carte Wemos D1 mini qui intègre un module Wifi ESP8266 et , un multiplexeur CD4051, qui pour rappel a pour but l'augmentation du nombre les d'entrées/sorties analogiques de la carte Wemos (dispose uniquement d'une seule) et de 4 capteurs (DHT22, LDR, pluie et humidité du sol.

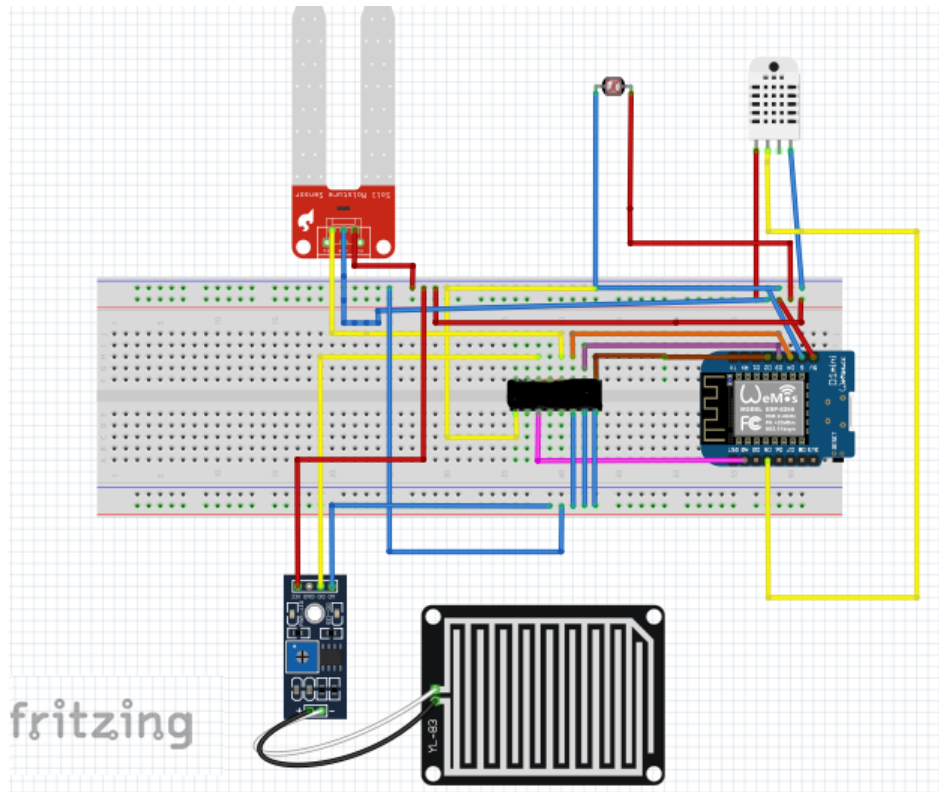


Figure 3.2 – Circuit électronique de notre système station météo.

3.3.1 Programmation de la carte WemosD1 mini avec l'IDE Arduino

La carte Wemos D1 mini comme les autres cartes Arduino nécessitent la présence d'un programme dans la carte pour pouvoir la faire fonctionner correctement et selon nos besoins. Afin de mieux comprendre le programme que nous avons élaboré pour notre station météo, nous avons scindé ce dernier en plusieurs parties qui seront présentées une par une dans ce qui suit.

— Partie 1 : déclaration des bibliothèques

Cette partie comme dans beaucoup de programmes Arduino a pour but la déclaration de toutes les bibliothèques indispensables à la bonne marche de notre station météo, telle que la bibliothèque ESP8266WiFi, DHT (capteurs de température/humidité), ...

```
#include <ESP8266WiFi.h>
#include <FirebaseESP8266.h>
#include <ArduinoJson.h>
#include <DHT.h>;
```

Figure 3.3 – Déclaration des bibliothèques dans le programme Arduino.

— **Partie 2 : Définition et initialisation**

Dans cette partie, il s'agit de définir le type de capteur DHT, sa broche, les broches d'adresses du circuit CD4051, les informations de connexion pour l'accès à la base de données Firebase, à savoir l'URL et le code d'authentification bloc et enfin le nom du réseau WiFi et son mot de passe pour l'accès à internet.

```
#define DHTPIN D5
#define DHTTYPE DHT22
DHT dht(DHTPIN,DHTTYPE);

#define FIREBASE_HOST "esp8266-e24ca-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "cqoKw1KIBGSKWEI905E4bYDeveI88qCxLcKUqRae"
#define WIFI_SSID "RedmiA9"
#define WIFI_PASSWORD "12345678"

#define MUX_A D4
#define MUX_B D3
#define MUX_C D2
```

Figure 3.4 – Définition de la base de données Firebase pour pouvoir y accéder.

— **Partie 3 : Synchronisation et stockage dans Firebase**

Ici, il s'agira d'initialiser la synchronisation et le stockage des données de nos capteurs en temps réel sous le format JSON.

```
FirebaseData firebaseData;
FirebaseJson json;
```

Figure 3.5 – Stockage et synchronisation dans la base de données Firebase.

— **Partie 4 : Déclaration et initialisation des différentes variables**

La 4ème partie de notre programme consiste à initialiser les différentes variables nécessaires au bon fonctionnement de notre station météo. Ces variables sont utilisées pour envoyer les données des capteurs sur la base de données Firebase.

```
int wait= 1000;  
float Sensor00;  
float Sensor11;  
float Sensor22;  
float Sensor33;  
float Sensor44;  
float Sensor55;  
float Sensor66;  
float Sensor77;
```

Figure 3.6 – Déclaration des variables.

— Partie 5 : Initialisation et démarrage du port série et du capteur DHT

Il s'agira dans cette partie d'initialiser le port série, le capteur DHT22 et la broche A0 comme entrée. Cette initialisation doit être contenue dans la section void setup qui vient au début de chaque programme et qui est exécutée une fois.

```
void setup() {  
  Serial.begin(9600);  
  dht.begin();  
  pinMode(A0, INPUT);  
}
```

Figure 3.7 – Première partie de la fonction void setup.

— Partie 6 : initialisation et démarrage du port série et du capteur DHT22

Après l'initialisation et le démarrage du port série et du capteur DHT22, il faudra lancer la connexion WiFi avec la carte Wemos D1 mini et tester si la carte a bien réussi à se connecter en affichant l'adresse IP et le message connected. Dans un deuxième temps, il s'agira de lancer la connexion avec la base de données Firebase en spécifiant les données d'accès définies dans la partie 2 (définition et initialisation) de notre programme.

```
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("connecting");
while (WiFi.status() != WL_CONNECTED) {
  Serial.print(".");
  delay(500);
}
Serial.println();
Serial.print("connected: ");

Serial.println(WiFi.localIP());

Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
delay(500);
```

Figure 3.8 – Lancement de la connexion WIFI avec la carte Wemos D1 mini ainsi que la connexion à la base de données Firebase.

— Partie 7 : Configuration des broches d'adresses du CD4051

Nous avons créé une fonction nommée `changeMux` dont le rôle est d'y définir les trois variables `a`, `b` et `c` et placer les valeurs de ces dernières dans les variables du CD4051 (MUX A, MUX B, MUX C).

```
void changeMux(int c, int b, int a) {
  digitalWrite(MUX_A, a);
  digitalWrite(MUX_B, b);
  digitalWrite(MUX_C, c);
}
```

Figure 3.9 – Création de la fonction `ChangeMux` du CD4051.

— Partie 8 : Lecture des capteurs et envoie des valeurs dans Firebase

Cette partie se concentre sur la lecture des données des capteurs DHT22, pluie, LDR et humidité du sol. En fonction de la combinaison des variables d'adresses définies précédemment, le CD4051 aiguillera la donnée à lire. Par exemple, si la combinaison définie est 000, cela veut dire que `a=0`, `b=0` et `c=0`, alors c'est le capteur de pluie qui sera lu. Elle est ensuite échelonnée sur une valeur allant par exemple de 0 à 100 en utilisant l'instruction `map` après quoi cette donnée est envoyée vers la base de données Firebase. Le même principe est repris pour le capteur LDR et le capteur d'humidité du sol. Notons ici que le délai d'envoi des données est de 1s mais que ce dernier peut être paramétrés

si besoin. En ce qui concerne le capteur DHT22, la lecture des données reprend le même concept cité précédemment, à une chose près que ce dernier n'est pas connecté au CD4051 mais à la carte Wemos D1 mini directement, car la lecture est faite de manière numérique sur ce capteur.

```

void loop(){
  changeMux(LOW, LOW,LOW ); //Data flow - X0
  int rain1 = analogRead(A0);
  Sensor00 = map(rain1, 0, 1023, 0, 100);
  Serial.println("RAIN1");
  Serial.println(Sensor00);
  Firebase.setFloat(firebaseData,"/data/pluie",Sensor00);
  delay(1000);
  changeMux(LOW, LOW,HIGH ); //Data flow - X1
  int moi1 = analogRead(A0);
  Sensor11 = map(moi1, 0, 1023, 0, 100);
  Serial.println("Moisture1");
  Serial.println(Sensor11);
  Firebase.setFloat(firebaseData,"/data2/humidité du sol",Sensor11);
  delay(1000);
  changeMux(HIGH, LOW,LOW);//////////X4
  int light=analogRead(A0);
  Sensor44 = map(light, 0, 1023, 0, 100);
  Serial.println("light");
  Serial.println(Sensor44);
  Firebase.setInt(firebaseData,"/data3/l'intensité lumineuse",Sensor44 );
  delay(1000);
  float hum=dht.readHumidity();
  Sensor66 = map(hum, 0, 1023, 0, 100);
  Serial.println("Humidité");
  Serial.println(hum);
  Firebase.setFloat(firebaseData,"/data4/ humidite",Sensor66);
  delay(1000);
  float tem=dht.readTemperature();
  Sensor66 = map(tem, 0, 1023, 0, 100);
  Serial.println("Temperature");
  Serial.println(tem);
  Firebase.setFloat(firebaseData,"/data5/ temperature",Sensor66);
  delay(1000);
}

```

Figure 3.10 – Lecture et envoi des données des capteurs à la base de données Firebase.

Voici un exemple de ce qu'on voit au niveau de notre moniteur série lors de la lecture des données des capteurs.

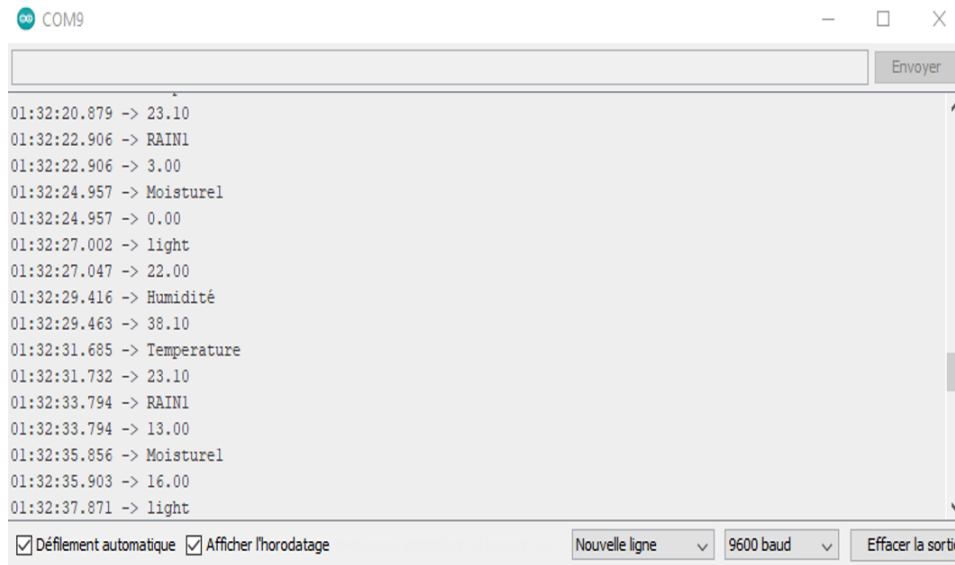


Figure 3.11 – Affichage des données capteurs sur le moniteur série.

3.4 Base de données

Dans notre travail, cette base de données joue un rôle primordial dans le bon fonctionnement de notre station météo. Elle permet de stocker les données émanant des capteurs et envoyés par la carte Wemos D1 mini mais plus encore, elle constitue le lien entre la partie électronique et la partie application Android de notre station météo.

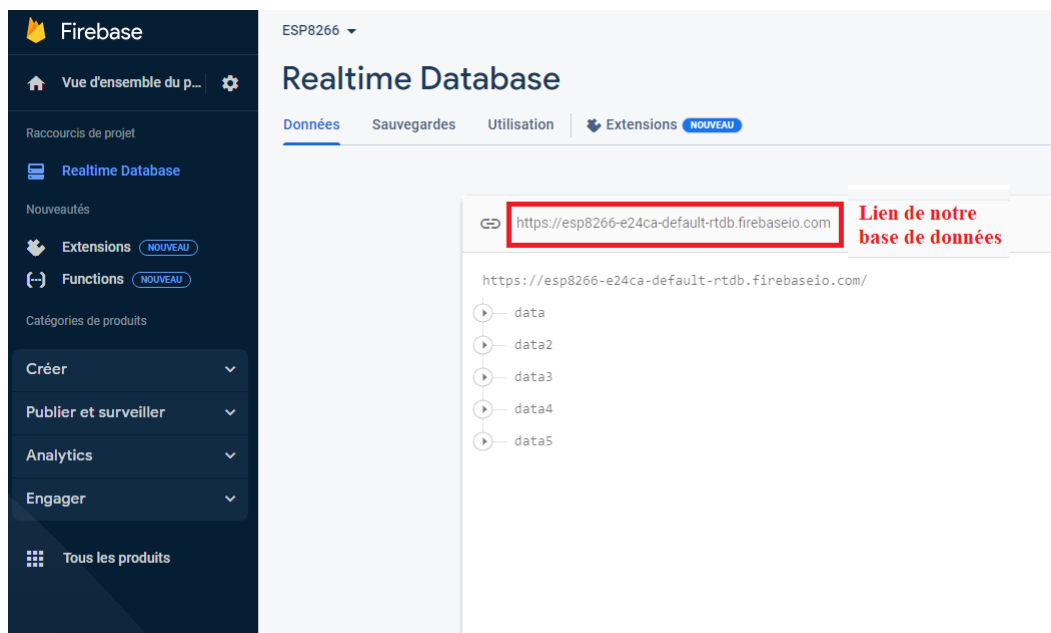


Figure 3.12 – Récupération du lien de notre base de données Firebase.

Comme nous pouvons le voir sur la figure 3.12, les données sont situées sous le lien de notre base de données. C'est ce que nous allons détailler dans ce qui suit.

— **Données de notre base de données Firebase**

Ces données comme nous l'avons évoqué à plusieurs reprises servent à stocker les données provenant des capteurs par le biais de la carte Wemos D1 mini.

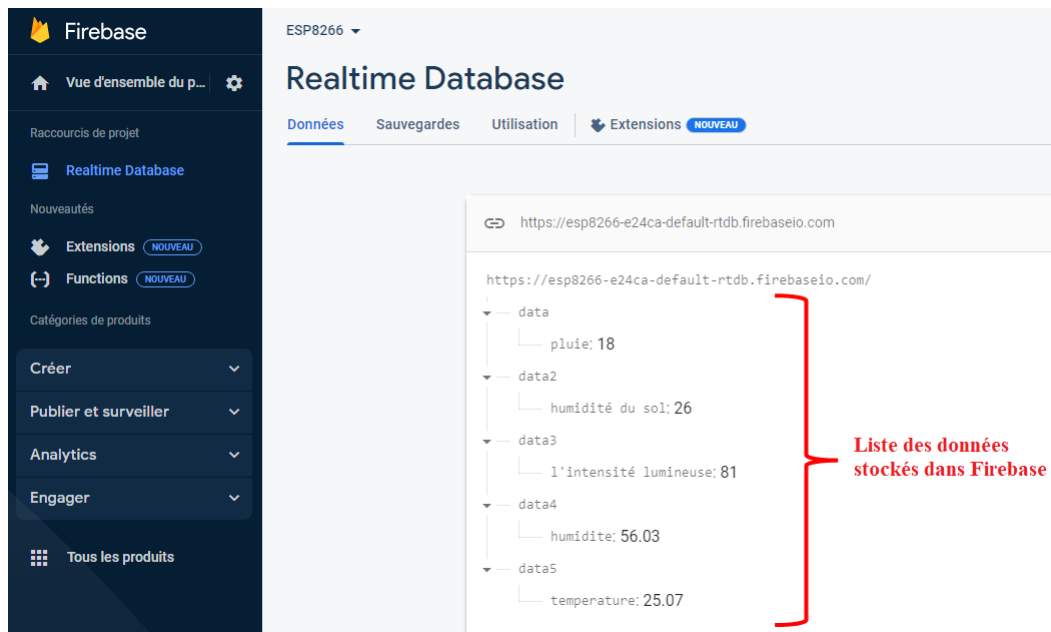


Figure 3.13 – Données de notre base de données Firebase.

Ces données peuvent être créées soit directement dans Firebase soit en utilisant la carte Wemos D1 mini, car quand la carte envoie les données à la base de données Firebase, si ces données existent (variables), alors seule la valeur est changée, mais si cette donnée n'existe pas, alors elle est automatiquement créée.

3.5 L'application Android « Weather and farming »

La conception de l'application Android en utilisant l'outil MIT App Inventor se divise en deux parties :

- Partie designer : partie dédiée au design de l'interface de notre application.

- Partie blocks : partie dédiée à la programmation des différents composants de notre application tels l'accès à la base de données Firebase, récolte des données des capteurs, notifications, envoi de sms, ...

3.5.1 Partie Designer

Notre application dispose de quatre fenêtres comme le montre la figure 3.14 qui suit

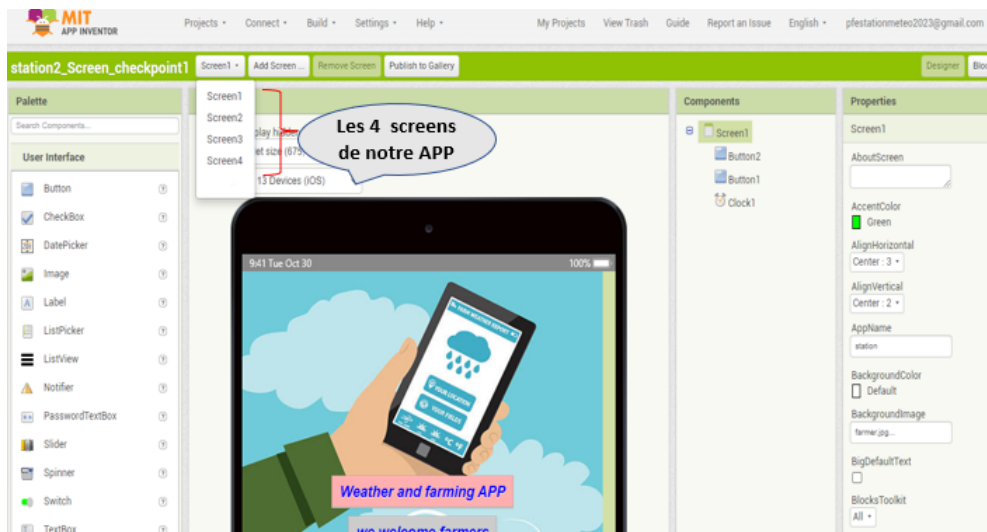


Figure 3.14 – les 4 fenêtres (screens) principales de notre application Android.

Lorsque l'application est lancée, un écran d'accueil s'affiche et reste affiché pendant quelques secondes. Un autre écran est ensuite ouvert, nous demandant d'introduire le code d'accès à l'application qui est obligatoire pour pouvoir consulter les données de la station météo. Si le code introduit est correct, une troisième fenêtre apparaît avec tout les paramètres des capteurs en temps réel affichés.

La figure 3.15 qui suit montre les quatre fenêtres que nous venons de citer.



Figure 3.15 – Les quatre fenêtres principales d'accès à l'application et et aux différents paramètres mesurés avec les maladies.

3.5.2 Partie Blocks :

— Block 1 :

Ce premier bloc est responsable de l'écran d'accueil lors du lancement de l'application depuis le smartphone.

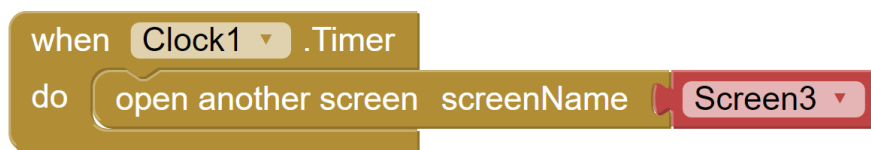


Figure 3.16 – Bloc de l'écran d'accueil de l'application .

— Block 2 :

Cette partie sert à vérifier si le mot de passe de connexion à l'application est correct ou non. Si c'est le cas, alors l'utilisateur pourra accéder aux données de la station météo, sinon l'utilisateur restera bloqué sur cet écran.

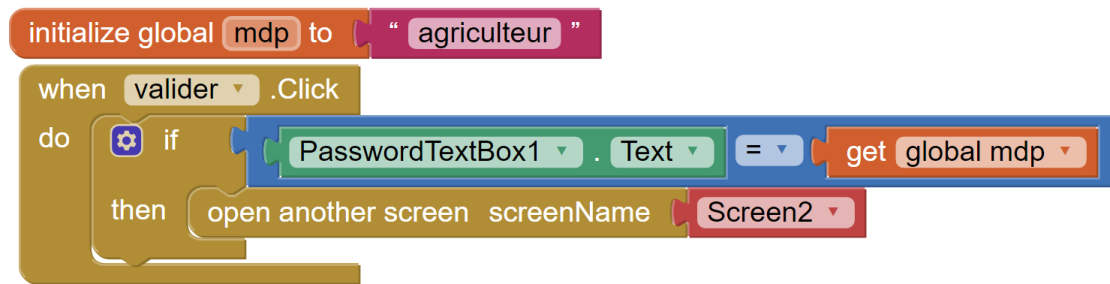


Figure 3.17 – Bloc gérant l'accès par mot de passe à l'application.

— Block 3 :

Cette partie qui est le coeur de notre application Android contient en fait cinq blocs Firebases (DB1, DB2, DB3, DB4, DB5) distincts qui ont la même fonction. Ainsi, chacun de ces blocs est responsable de la lecture d'un paramètre provenant de la base de données Firebase et correspondant au capteur que l'on veut lire, traite ensuite la donnée reçue en la comparant avec des valeurs références et envoie une notification en plus d'un sms si la valeur mesurée est en dehors de cette plage. De cette manière, l'agriculteur ou l'ingénieur agronome pourra prendre une décision quant à la conduite à adopter et au traitement adéquat.

Pour mieux comprendre ce que l'on vient de dire, nous allons détailler chaque bloc séparément.

- **Block Firebase DB1** : Ce dernier est responsable de la lecture du paramètre pluie. Si ce paramètre est $\geq 25\%$, alors une notification et un sms sont envoyés à l'agriculteur et le paramètre est affiché sur fond rouge. Par contre si les valeurs sont normales, alors dans ce cas les paramètres sont affichés sur fond vert et aucune notification ou sms ne sont envoyés.
- **Block Firebase DB2** : Même constat pour le deuxième bloc DB2, si le paramètre humidité est $\leq 30\%$, alors une notification et un sms sont envoyés à l'agriculteur, et sur l'application le paramètre passe sur un fond rouge, par contre si les valeurs sont normales, alors dans ce cas les paramètres sont affichés sur fond vert et aucune notification ou sms ne sont envoyés.

- **Block Firebase DB3 :** Le même principe est appliqué pour l'intensité lumineuse. Ici par exemple, si l'intensité lumineuse a pour valeur 50%, alors une notification et un sms sont envoyés à l'agriculteur et sur l'application le paramètre passe sur un fond rouge. Sinon, le paramètre reste sur un fond vert et aucune notification ou sms ne sont envoyés.



Figure 3.18 – Bloc DB1, DB2 et DB3 qui ont pour fonction la vérification et l’affichage des paramètres pluie, intensité lumineuse et humidité avec notification et sms.

- **Block Firebase DB4 :** Ce bloc est chargé de récupérer le paramètre humidité du sol de Firebase. Si cette valeur est $\geq 84\%$ une notification audio et un sms sont envoyés à l'agriculteur et le paramètre est affiché sur fond rouge. Dans ce cas, le risque de voir apparaître les maladies de pomme de terre tardive ainsi que la tavelure du pommier

citée dans le premier chapitre est important. Par ailleurs, notons au passage que la maladie de la tavelure du pommier dépend aussi du paramètre température.

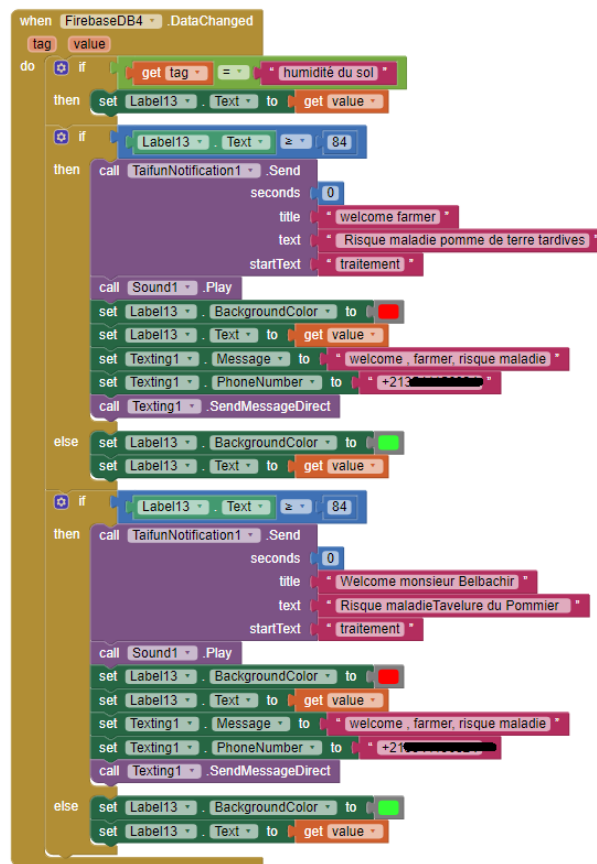


Figure 3.19 – Bloc DB4 qui a pour fonction la vérification et l’affichage du paramètre humidité du sol avec notification et SMS.

- **Block Firebase DB5** : Ce bloc est chargé de récupérer la paramètre température de notre base de données. Cette dernière est comparée à trois conditions :
 - température $t > 20^{\circ}\text{C}$ et $t \leq 25^{\circ}\text{C}$, dans ce cas on envoie une notification et sms à l’agriculteur en spécifiant que le risque de maladie de la rouille brune du blé est important avec toujours passage sur fond rouge du paramètre sur l’application.
 - température $t > 11^{\circ}\text{C}$ et $t \leq 20^{\circ}\text{C}$, dans ce cas on envoie une notification et sms à l’agriculteur en spécifiant que le risque de maladie de pourriture grise est important avec toujours passage sur fond rouge du paramètre sur l’application.
 - température $t > 17^{\circ}\text{C}$ et $t \leq 24^{\circ}\text{C}$, dans ce cas on envoie une notification et sms à l’agriculteur en spécifiant que le risque de maladie de la tavelure du

pommier est important avec toujours passage sur fond rouge du paramètre sur l'application. Pour cette maladie, il est nécessaire de prendre en considération le paramètre humidité du sol qui doit être supérieure à 84%.

```

when FirebaseDB5 .DataChanged
  tag value
do
  if get tag == "Temperature"
  then
    set Label19 .Text to get value
    if Label19 .Text > 20 and Label19 .Text ≤ 25
    then
      call TaifunNotification1 .Send
        seconds 0
        title "welcome farmer"
        text "Risque maladie de la rouille brune du blé"
        startText "traitement"
      call Sound1 .Play
      set Label19 .BackgroundColor to red
      set Label19 .Text to get value
      set Texting1 .Message to "welcome , farmer, risque maladie"
      set Texting1 .PhoneNumber to "+213"
      call Texting1 .SendMessageDirect
    else
      set Label19 .BackgroundColor to green
      set Label19 .Text to get value
    if Label19 .Text > 11 and Label19 .Text ≤ 20
    then
      call TaifunNotification1 .Send
        seconds 0
        title "welcome farmer"
        text "Risque Maladie porriture grise"
        startText "traitement"
      call Sound1 .Play
      set Label19 .BackgroundColor to red
      set Label19 .Text to get value
      set Texting1 .Message to "welcome , farmer, risque maladie"
      set Texting1 .PhoneNumber to "+213"
      call Texting1 .SendMessageDirect
    else
      set Label19 .BackgroundColor to green
      set Label19 .Text to get value
    if Label19 .Text ≥ 17 and Label19 .Text ≤ 24
    then
      call TaifunNotification1 .Send
        seconds 0
        title "Welcome monsieur Belbachir"
        text "Risque maladie Tavelure du Pommier"
        startText "traitement"
      call Sound1 .Play
      set Label19 .BackgroundColor to red
      set Label19 .Text to get value
      set Texting1 .Message to "Welcome monsieur belbachir, Risque maladie, tava..."
      set Texting1 .PhoneNumber to "+213"
      call Texting1 .SendMessageDirect
    else
      set Label19 .BackgroundColor to green
      set Label19 .Text to get value
  
```

Figure 3.20 – Bloc DB5 qui a pour fonction la vérification et l’affichage du paramètre température avec notification et SMS.

— Block 4 :

Cette partie sert à nous faire revenir à partir de l'écran ou sont affichés les paramètres à la fenêtre screen3 correspondant à la fenêtre de connexion.

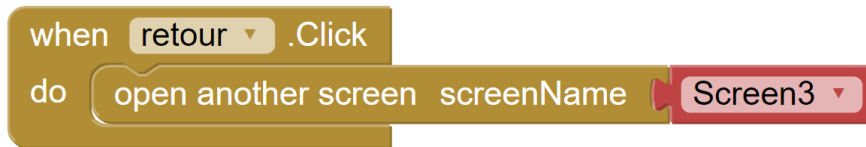


Figure 3.21 – Bloc de retour vers l'écran de connexion (login).

— Block 5 :

Cette partie permet en cliquant sur le bouton détails de nous amener sur l'écran contenant les détails sur quelques maladies.

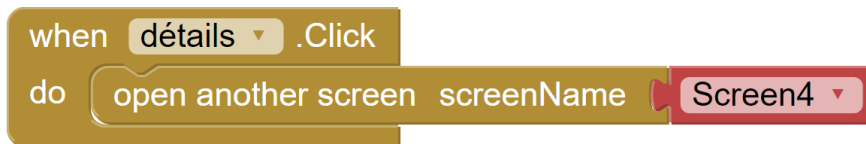


Figure 3.22 – Bloc amenant l'utilisateur vers l'écran contenant les détails sur les maladies.

— Block 6 :

Cette partie du programme sert à faire revenir l'utilisateur à l'écran d'accueil.

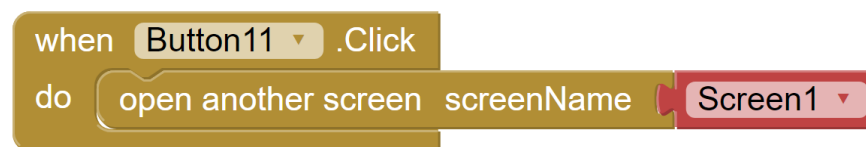


Figure 3.23 – Bloc de retour à l'écran d'accueil à partir de l'écran contenant les détails sur les maladies.

3.6 Tests pratique de notre station météo

Dans cette partie, nous faisons une démonstration montrant le bon fonctionnement de notre station météo.

Le câblage de la carte Wemos D1 mini avec les différents capteurs étant effectué, ces derniers sont placés dans un boîtier en prenant soin de laisser les capteurs apparents sauf le capteur d'humidité sol qui va être placé dans le pot de la plante pour pouvoir prendre les mesures.



Figure 3.24 – Notre station météo.

Avant de lancer le test complet avec l'application Android, nous allons procéder étape par étape pour vérifier que tout fonctionne correctement. Tout d'abord et après téléversement du programme dans la carte Arduino, nous ouvrons le port série de l'IDE Arduino afin de vérifier que les capteurs affichent bien des valeurs.

```

COM12
19:31:56.359 -> Humidité
19:31:56.359 -> 55.80
19:31:58.422 -> Temperature
19:31:58.422 -> 25.20
19:32:00.250 -> RAIN1
19:32:00.250 -> 44.00
19:32:08.265 -> Moisture1
19:32:08.265 -> 43.00
19:32:11.639 -> light
19:32:11.639 -> 41.00
19:32:14.171 -> Humidité
19:32:14.171 -> 56.70
19:32:16.756 -> Temperature
19:32:16.756 -> 25.30
19:32:19.590 -> RAIN1
19:32:19.590 -> 43.00
    
```

Figure 3.25 – Valeurs des capteurs sur le moniteur série .

A partir de là, nous accédons à notre base de données Firebase et nous regardons si les valeurs changent en fonction de ce qu'on lit sur le moniteur série.

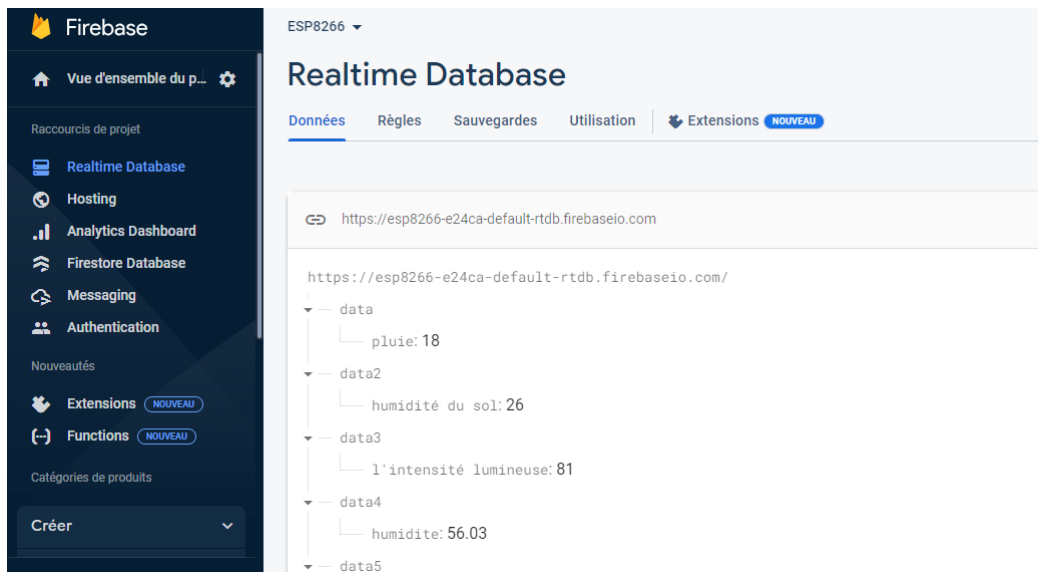


Figure 3.26 – Données des capteurs stockés sur la base de données Firebase.

Dès qu'un changement de valeur d'un capteur se produit, la valeur est instantanément modifiée sur la base de données Firebase comme le montre la figure 3.27.

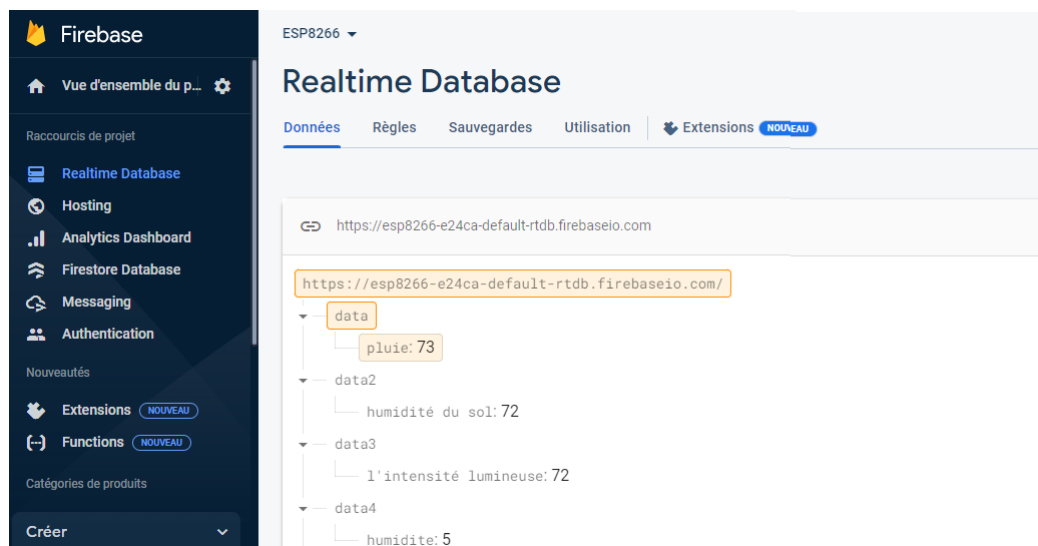


Figure 3.27 – Changements de valeurs en temps réel du capteur de pluie au niveau de la base de données Firebase.

Nous ouvrons ensuite l'application sur notre smartphone et introduisons notre code d'accès à l'application.

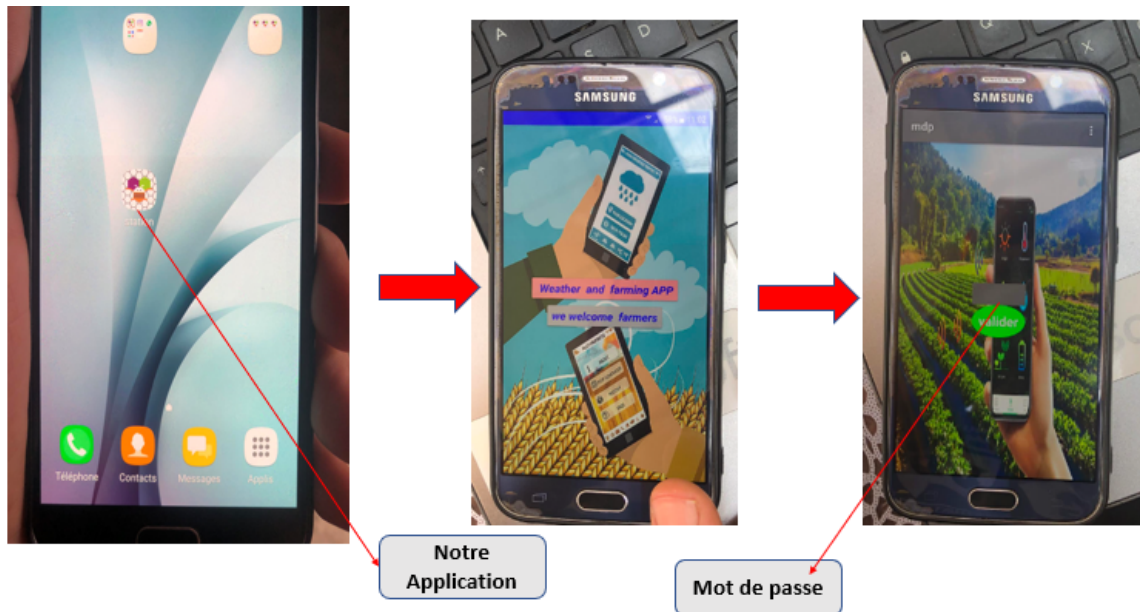


Figure 3.28 – Lancement de l' application Android Weather and Farmer à partir du smartphone.

Après ouverture de l'application et après avoir introduit notre code d'accès, on arrive sur la fenêtre contenant les informations des capteurs (figure 3.29).



Figure 3.29 – valeurs intermédiaires pas de maladie .

Nous pouvons remarquer ici que toutes les valeurs sont au vert, car dans ce cas tous les paramètres émis par les capteurs et testés sur l'application sont normales.

- En cas de risque de maladie : Par exemple, si la valeur d'humidité du sol est augmentée en versant de l'eau dans le pot de plante, et de même pour la température en mettant une source de chaleur à côté du capteur DHT22, nous constatons que les valeurs basculent sur un fond rouge dès que nous dépassons les seuils fixés dans l'application. Ceci implique qu'il subsiste un risque de maladie à prendre au sérieux. Dans ce cas, l'agriculteur ou l'ingénieur agronome reçoit une notification sonore en plus d'un SMS.

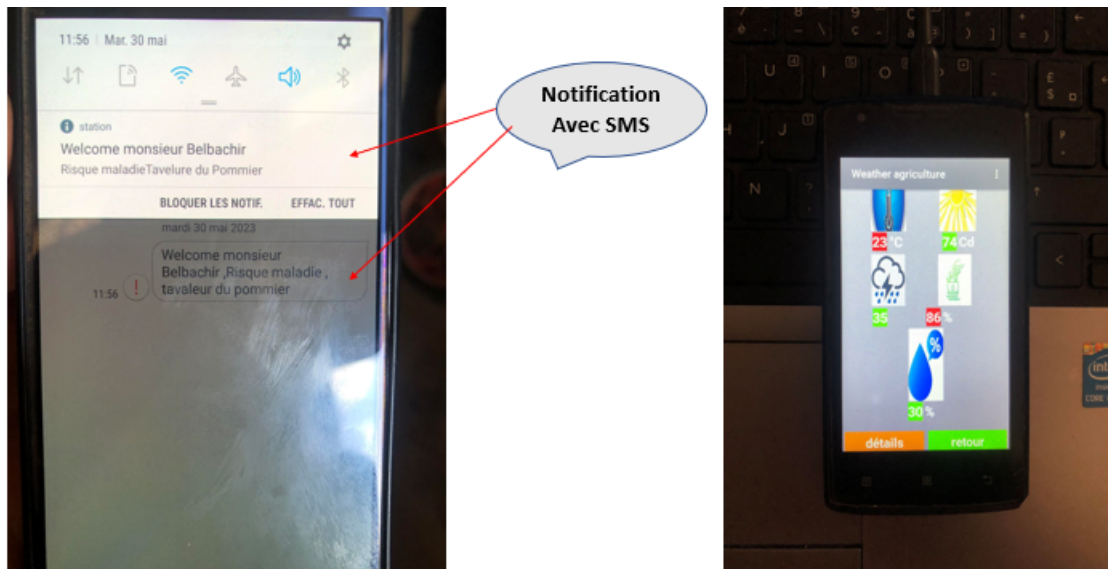


Figure 3.30 – Réception d'une notification en plus d'un SMS contenant la maladie à prendre en considération et affichage des paramètres en relation avec cette maladie sur fond rouge.

A partir de la notification et le SMS reçus, l'agriculteur peut cliquer en bas sur détails pour avoir plus détails sur les facteurs liés à la maladie (figure 3.31).



Figure 3.31 – ecran des maladies

Notons par ailleurs que la fonction SMS a été principalement ajoutée dans le cas où l'agriculteur ne dispose pas de smartphone, dans ce cas même avec un téléphone traditionnel, il recevra un SMS avec les détails de la maladie en question.

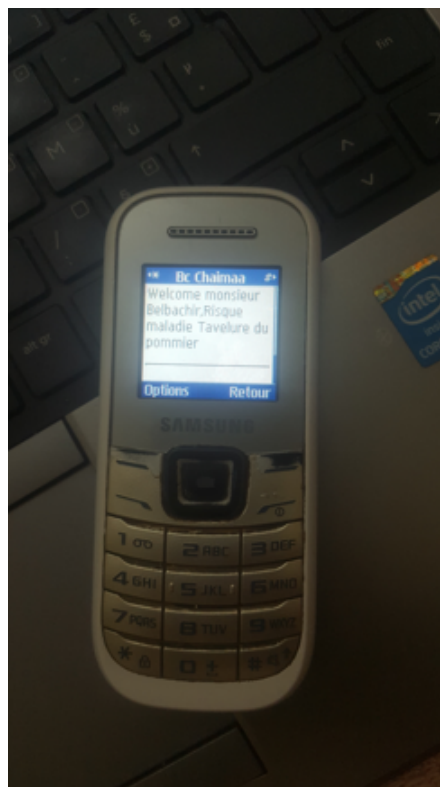


Figure 3.32 – Réception du SMS sur un téléphone traditionnel.

3.7 Conclusion :

Dans ce chapitre, nous avons présenté en détail les étapes de réalisation de notre station météo. Nous avons d'abord commencé par donner un schéma synoptique décrivant le principe de fonctionnement de notre station météo. Ensuite, nous avons réalisé le circuit électronique composé des différents capteurs et la carte Wemos D1 mini tout en détaillant le programme Arduino réalisé. Nous avons ensuite montré la base de données Firebase que nous avons créé avant de détailler l'application Android Weather and Farming que nous avons réalisé. Nous avons terminé ce chapitre par quelques tests pratiques montrant le bon fonctionnement de notre station météo.



Conclusion générale et perspectives

Conclusion générale

L'objectif de ce Projet était de répondre à un besoin d'anticipation pour le traitement des maladies dans les champs agricoles causées par les changements climatiques. A cet effet nous avons réalisé une station météorologique utilisant 4 capteurs permettant de récolter des informations sur la température, l'humidité, l'intensité Lumineuse, l'humidité du sol et la pluie. Pour atteindre cet objectif, nous nous sommes reposés sur trois composants essentiels qui sont : une carte Wemos D1 mini intégrant un module WiFi ESP8266 sur laquelle viennent se brancher les différents capteurs à travers un circuit multiplexeur/démultiplexeur CD4051, une application Android que nous avons nommé « Weather and Farmer » et une base de données Firebase. Le principe de fonctionnement que nous avons proposé reste simple. Les capteurs sont branchés sur la carte Wemos D1 mini à travers un circuit multiplexeur/démultiplexeur CD4051. Le rôle de ce dernier est juste d'augmenter le nombre d'entrées/sorties analogiques disponibles sur la carte Wemos D1 mini. Cette dernière récolte les informations provenant des capteurs et les envoie à la base de données Firebase. L'application Android « Weather and farmer » n'aura plus alors qu'à récupérer ces données pour les afficher. Sur cette même application, ces données sont comparés avec des valeurs de références correspondants à quelques maladies. Si les données sont dans cette plage de référence, alors les valeurs sont affichées dans l'application sur un fond vert signifiant que les paramètres sont bons et qu'aucun risque de maladie n'est présent. Si les données récoltées sont en en dehors de cette plage de références, les données sont affichées sur fond rouge et aussitôt une notification et un sms sont envoyés à l'agriculteur ou l'ingénieur agronome signalant un risque potentiel de maladie. Grâce à cette méthode,

nous pouvons détecter des maladies tels que la tavelure du pommier, la pourriture grise sur les raisins, pomme de terre tardives brillante et la maladie de la rouille brune. Les tests pratiques effectués dans le chapitre trois ont montré le bon fonctionnement du concept proposé. Nous avons ainsi pu constater qu'au moindre débordement sur la plage de référence défini dans l'application une notification et un sms sont envoyés pour alerter sur le risque d'éventuel maladies.

Perspectives

Malgré le bon fonctionnement de notre station météo, des améliorations et optimisations peuvent être apportés sur ce projet. Voici quelques suggestions :

- Valider ces résultats sur des champs agricoles en insistant sur une plage de temps propice au développement de ces maladies.
- Rajouter d'autres types de capteurs tels que l'anémomètre et le pluviomètre.
- Certaines maladies présentent d'autres facteurs en plus de la météo tels que le temps, par exemple pendant combien de temps est tombée la pluie ou pendant combien de temps la température a été au-dessus de 25°C par exemple.
- rajouter une charte graphique dans l'application montrant l'évolution sur une certaine période des paramètres météorologiques.
- Augmenter le nombre de maladies pouvant être détectées par ce systèmes.
- Certaines zones agricoles ou certains agriculteurs ne disposent pas toujours d'une connexion WiFi, il faudrait alors songer à modifier le type de connexion, par exemple rajouter un module à carte SIM pour se connecter en données cellulaires.



Bibliographie

- [1] « quels sont les grands moments de l’histoire de l’agriculture ? agriculture du maghreb ». <https://www.agri-mag.com/2020/04/26/quels-sont-les-grands-moments-de-l-histoire-de-l-agriculture> - (consulté le 11 mai 2023).
- [2] Agriculture - histoire des agricultures jusqu’au XIXe siècle - encyclopædia universalis. <https://www.universalis.fr/encyclopedie/agriculture-histoire-des-agricultures-jusqu-au-xixe-siecle/> - (consulté le 11 mai 2023).
- [3] « définition agriculture futura planète ». <https://www.futura-sciences.com/planete/definitions/developpement-durable-agriculture-11492> - (consulté le 11 mai 2023).
- [4] Agri algérie, « l’agriculture en algérie en 2022 – AgriAlgerie.com », 3 février 2022. <https://www.agrialgerie.com/l-agriculture-en-algerie-en-2022> - (consulté le 11 mai 2023).
- [5] Mary Nyasimi, D. Amwata, L. Hove, James Kinyangi, and George Wamukoya. L’agriculture intelligente face au climat : Quel impact pour l’afrique? Accepted : 2015-07-01T12 :24 :24Z.
- [6] Rouille brune. <https://profert.dz/fr/index.php/2017/11/02/rouille-brune> -(consulté le 11 mai 2023).
- [7] Mémoire de fin d’étude - etude des maladies fongiques affectant les cultures agricoles dans les deux régions de zyghoud youssef bani hamidan, constantine.pdf.
- [8] «réalisation d’une station météorologique pour une serre» mémoire de fin d’études master université mohamed khider biskra 2019.» .pdf.

- [9] Xing Yang, Lei Shu, Jianing Chen, Mohamed Amine Ferrag, Jun Wu, Edmond Nurellari, and Kai Huang. A survey on smart agriculture : Development modes, technologies, and security and privacy challenges. 8(2) :273–302.
- [10] FOUFOU NOUR ELHOUDA ET RADI NADJLAA «réalisation d’une station météo connectée memoire de master université aboubakr belkaïd– tlemcen, 2019 » .pdf.
- [11] « salmi nessrine et ould amar nessrine «conception et réalisation d’un module pour l’acquisition des données métrologique a base de l’arduino» université de saad dahlab blida 1- juin 2020. ».
- [12] Stations météo - <https://docplayer.fr/8378805-s-t-a-t-i-o-n-s-m-e-t-e-o.html>(consulté le 11 mai 2023).
- [13] É. larousse, « définitions : température - dictionnaire de français larousse ». <https://www.larousse.fr/dictionnaires/francais/temp%C3%A9rature/77211> - (consulté le 9 mai 2023).
- [14] « photoresistance-ldr-7mm.jpg (image JPEG, 458 × 458 pixels) ». https://www.distrionic.fr/9454-large_default/photoresistance-ldr-7mm.jpg - (consulté le 11 mai 2023).
- [15] https://cdn.instrumentationtools.com/wp-content/uploads/2016/07/instrumentation_tools.com_photo-diode.jpg – (consulté le 9 mai 2023).
- [16] Les différents types d’hygromètres, hygrometre.net - <https://www.hygrometre.net/prix-type-hygrometre/> - (consulté le 9 mai 2023).
- [17] « about arduino ». <https://www.arduino.cc/en/about> (consulté le 14 mai 2023).
- [18] « what is wi-fi? - definition and types », cisco. <https://www.cisco.com/c/en/us/products/wireless/what-is-wifi.html>-(consulté le 14 mai 2023).
- [19] « who we are | wi-fi alliance ». <https://www.wi-fi.org/who-we-are> - /(consulté le 17 mai 2023).
- [20] cdaviddav, « ESP8266 WeMos d1 mini tutorial », 1 avril 2021. <https://diyi0t.com/esp8266-wemos-d1-mini-tutorial/> -(consulté le 17 mai 2023).

- [21] Circuit intégré CD4051 multiplexeur / démultiplexeur à 8 canaux analogiques. <http://tiptopboards.com/366-cd4051-multiplexeur-démultiplexeur-8-canaux-analogiques.html> (consulté le 25 mai 2023).
- [22] <https://fluxworkshop.com/blogs/getting-started/lets-workshop-gy-30-light-intensity-sensor-module-1> - (consulté le 10 mai 2023).
- [23] <https://maker.pro/arduino/projects/arduino-soil-moisture-sensor> - (consulté le 17 mai 2023).
- [24] Etienne, « a quoi sert firebase, la plateforme mobile de google ? », junto, 25 avril 2019. <https://junto.fr/blog/firebase/> - (consulté le 25 mai 2023).
- [25] « what is firebase? », educative : Interactive courses for software developers. <https://www.educative.io/answers/what-is-firebas> - (consulté le 25 mai 2023).
- [26] « firebase », firebase. <https://firebase.google.com/> - (consulté le 25 mai 2023).
- [27] What is android? -androidauthority- here's everything you need to know. android -authority. - (consulté le 17 mai 2023).
- [28] Tribu.phm.education.gouv.fr -.

Annexes

A.1 Programme Arduino complet

Listing 1 – Stationmeteo.ino

```
1 //Programme PFE Instrumentation Electronique 2022-2023
2 //Benyoub + Benhadjba
3 #include <ESP8266WiFi.h>
4 #include <FirebaseESP8266.h>
5 #include <ArduinoJson.h>
6 #include <DHT.h>;
7 #define DHTPIN D5
8 #define DHTTYPE DHT22
9 DHT dht(DHTPIN, DHTTYPE);
10
11 #define FIREBASE_HOST "esp8266-e24ca-default-rtdb.firebaseio.com"
12 #define FIREBASE_AUTH "cqoKwLKIBGSKWEI905E4bYDeveI88qCxLcKUqRae"
13 #define WIFI_SSID "RedmiA9"
14 #define WIFI_PASSWORD "12345678"
15
16 #define MUX_A D4
17 #define MUX_B D3
18 #define MUX_C D2
19 FirebaseData firebaseData;
20 FirebaseJson json;
21
22 int wait= 1000;
23 float Sensor00;
24 float Sensor11;
```

```
25 float Sensor22;
26 float Sensor33;
27 float Sensor44;
28 float Sensor55;
29 float Sensor66;
30 float Sensor77;
31
32 void setup() {
33   Serial.begin(9600);
34   dht.begin();
35   pinMode(A0, INPUT);
36   WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
37   Serial.print("connecting");
38   while (WiFi.status() != WL_CONNECTED) {
39     Serial.print(".");
40     delay(500);
41   }
42   Serial.println();
43   Serial.print("connected:␣");
44   Serial.println(WiFi.localIP())
45   void changeMux(int c, int b, int a) {
46     digitalWrite(MUX_A, a);
47     digitalWrite(MUX_B, b);
48     digitalWrite(MUX_C, c);
49   void loop(){
50     changeMux(LOW, LOW, LOW ); //Data flow - X0
51     int rain1 = analogRead(A0);
52     Sensor00 = map(rain1, 0, 1023, 0, 100);
53     Serial.println("RAIN1");
54     Serial.println(Sensor00);
55     Firebase.setFloat(firebaseData, "/data/pluie", Sensor00);
56     delay(1000);
57     changeMux(LOW, LOW, HIGH ); //Data flow - X1
58     int moi1 = analogRead(A0);
59     Sensor11 = map(moi1, 0, 1023, 0, 100);
```

```
60 Serial.println("Moisture1");
61 Serial.println(Sensor11);
62 Firebase.setFloat(firebaseData, "/data2/humidite_du_sol", Sensor11);
63 delay(1000);
64 changeMux(HIGH, LOW, LOW);////////////////X4
65 int light=analogRead(A0);
66 Sensor44 = map(light, 0, 1023, 0, 100);
67   Serial.println("light");
68   Serial.println(Sensor44);
69   Firebase.setInt(firebaseData, "/data3/1'intensitelumineuse",
70   Sensor44 );
71   delay(1000);
72 float hum=dht.readHumidity();
73 Sensor66 = map(hum, 0, 1023, 0, 100);
74   Serial.println("Humidite");
75   Serial.println(hum);
76   Firebase.setFloat(firebaseData, "/data4/_humidite", Sensor66);
77   delay(1000);
78 float tem=dht.readTemperature();
79 Sensor66 = map(tem, 0, 1023, 0, 100);
80   Serial.println("Temperature");
81   Serial.println(tem);
82   Firebase.setFloat(firebaseData, "/data5/_temperature", Sensor66);
83   delay(1000);
84 }
```

A.2 Datasheet CD4051

CMOS Analog Multiplexers/Demultiplexers with Logic Level Conversion

The CD4051B, CD4052B, and CD4053B analog multiplexers are digitally-controlled analog switches having low ON impedance and very low OFF leakage current. Control of analog signals up to 20V_{P-P} can be achieved by digital signal amplitudes of 4.5V to 20V (if V_{DD}-V_{SS} = 3V, a V_{DD}-V_{EE} of up to 13V can be controlled; for V_{DD}-V_{EE} level differences above 13V, a V_{DD}-V_{SS} of at least 4.5V is required). For example, if V_{DD} = +4.5V, V_{SS} = 0V, and V_{EE} = -13.5V, analog signals from -13.5V to +4.5V can be controlled by digital inputs of 0V to 5V. These multiplexer circuits dissipate extremely low quiescent power over the full V_{DD}-V_{SS} and V_{DD}-V_{EE} supply-voltage ranges, independent of the logic state of the control signals. When a logic "1" is present at the inhibit input terminal, all channels are off.

The CD4051B is a single 8-Channel multiplexer having three binary control inputs, A, B, and C, and an inhibit input. The three binary signals select 1 of 8 channels to be turned on, and connect one of the 8 inputs to the output.

The CD4052B is a differential 4-Channel multiplexer having two binary control inputs, A and B, and an inhibit input. The two binary input signals select 1 of 4 pairs of channels to be turned on and connect the analog inputs to the outputs.

The CD4053B is a triple 2-Channel multiplexer having three separate digital control inputs, A, B, and C, and an inhibit input. Each control input selects one of a pair of channels which are connected in a single-pole, double-throw configuration.

When these devices are used as demultiplexers, the "CHANNEL IN/OUT" terminals are the outputs and the "COMMON OUT/IN" terminals are the inputs.

Ordering Information

PART NUMBER	TEMP. RANGE (°C)	PACKAGE
CD4051BF, CD4052BF, CD4053BF	-55 to 125	16 Ld CERAMIC DIP
CD4051BE, CD4052BE, CD4053BE	-55 to 125	16 Ld PDIP
CD4051BM, CD4051BNS	-55 to 125	16 Ld SOIC
CD4051BPW, CD4052BPW, CD4053BPW	-55 to 125	16 Ld TSSOP

Features

- Wide Range of Digital and Analog Signal Levels
 - Digital 3V to 20V
 - Analog ≤20V_{P-P}
- Low ON Resistance, 125Ω (Typ) Over 15V_{P-P} Signal Input Range for V_{DD}-V_{EE} = 18V
- High OFF Resistance, Channel Leakage of ±100pA (Typ) at V_{DD}-V_{EE} = 18V
- Logic-Level Conversion for Digital Addressing Signals of 3V to 20V (V_{DD}-V_{SS} = 3V to 20V) to Switch Analog Signals to 20V_{P-P} (V_{DD}-V_{EE} = 20V)
- Matched Switch Characteristics, r_{ON} = 5Ω (Typ) for V_{DD}-V_{EE} = 15V
- Very Low Quiescent Power Dissipation Under All Digital-Control Input and Supply Conditions, 0.2μW (Typ) at V_{DD}-V_{SS} = V_{DD}-V_{EE} = 10V
- Binary Address Decoding on Chip
- 5V, 10V and 15V Parametric Ratings
- 10% Tested for Quiescent Current at 20V
- Maximum Input Current of 1μA at 18V Over Full Package Temperature Range, 100nA at 18V and 25°C
- Break-Before-Make Switching Eliminates Channel Overlap

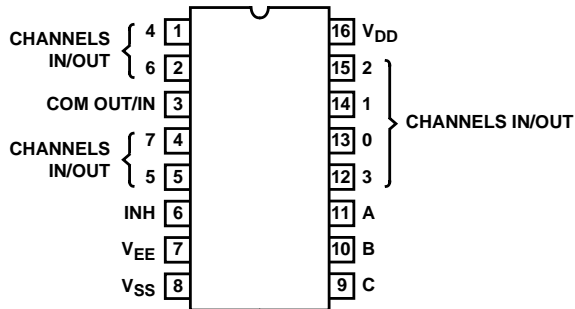
Applications

- Analog and Digital Multiplexing and Demultiplexing
- A/D and D/A Conversion
- Signal Gating

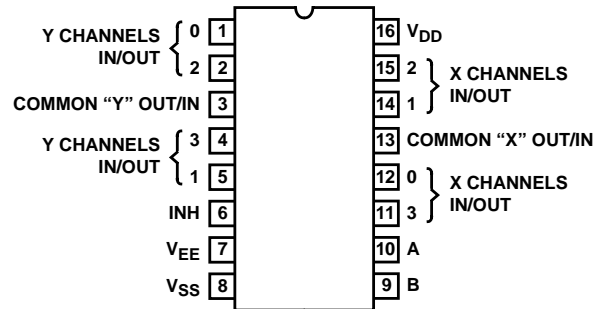
CD4051B, CD4052B, CD4053B

Pinouts

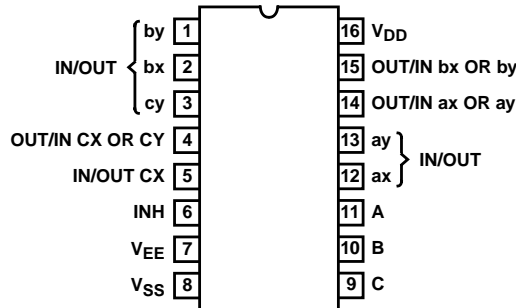
CD4051B (PDIP, CDIP, SOIC, TSSOP)
TOP VIEW



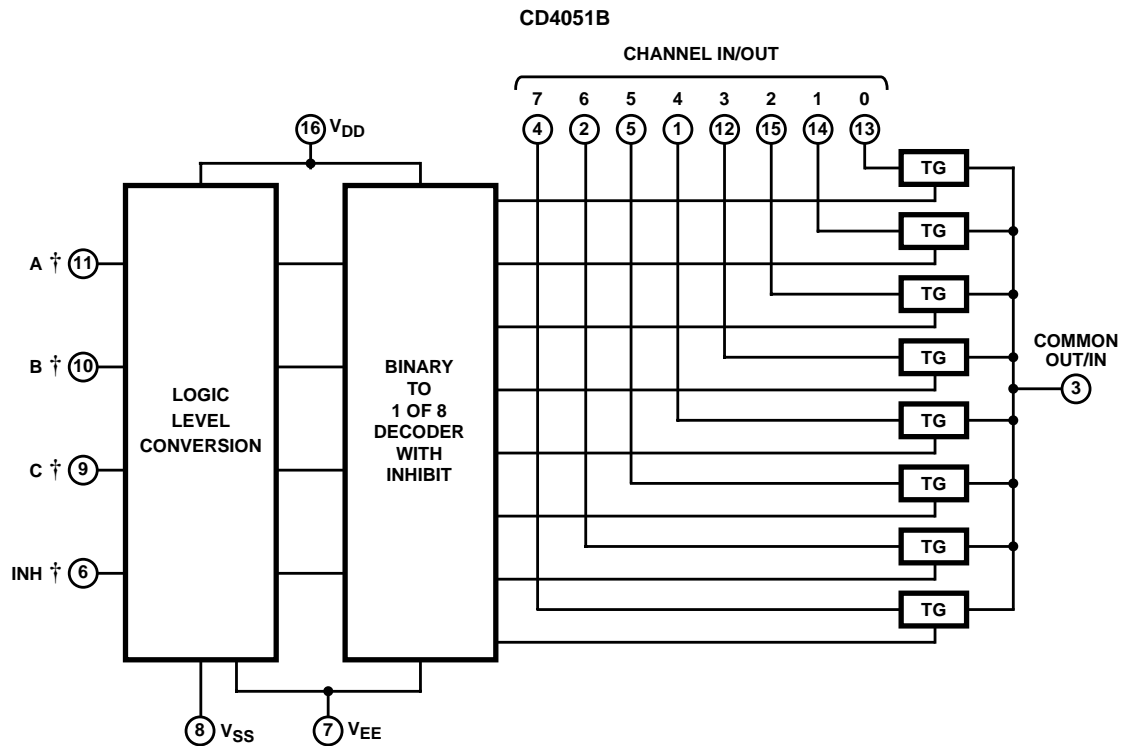
CD4052B (PDIP, CDIP, TSSOP)
TOP VIEW



CD4053B (PDIP, CDIP, TSSOP)
TOP VIEW



Functional Block Diagrams

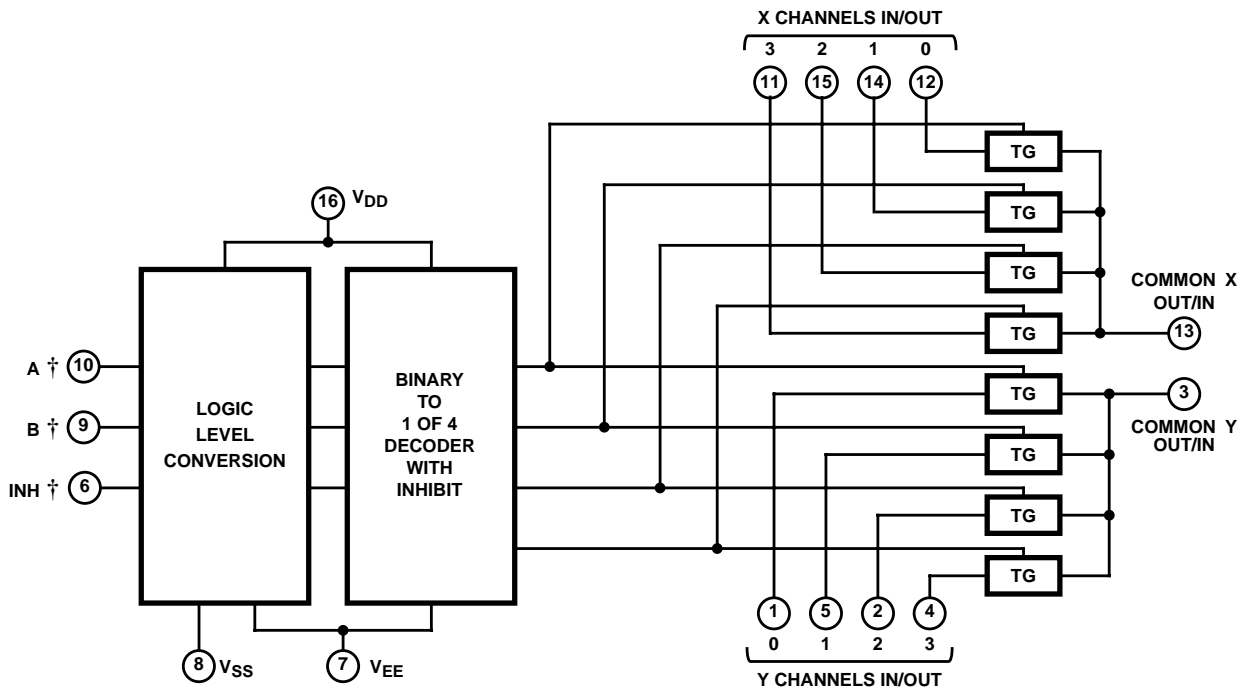


† All inputs are protected by standard CMOS protection network.

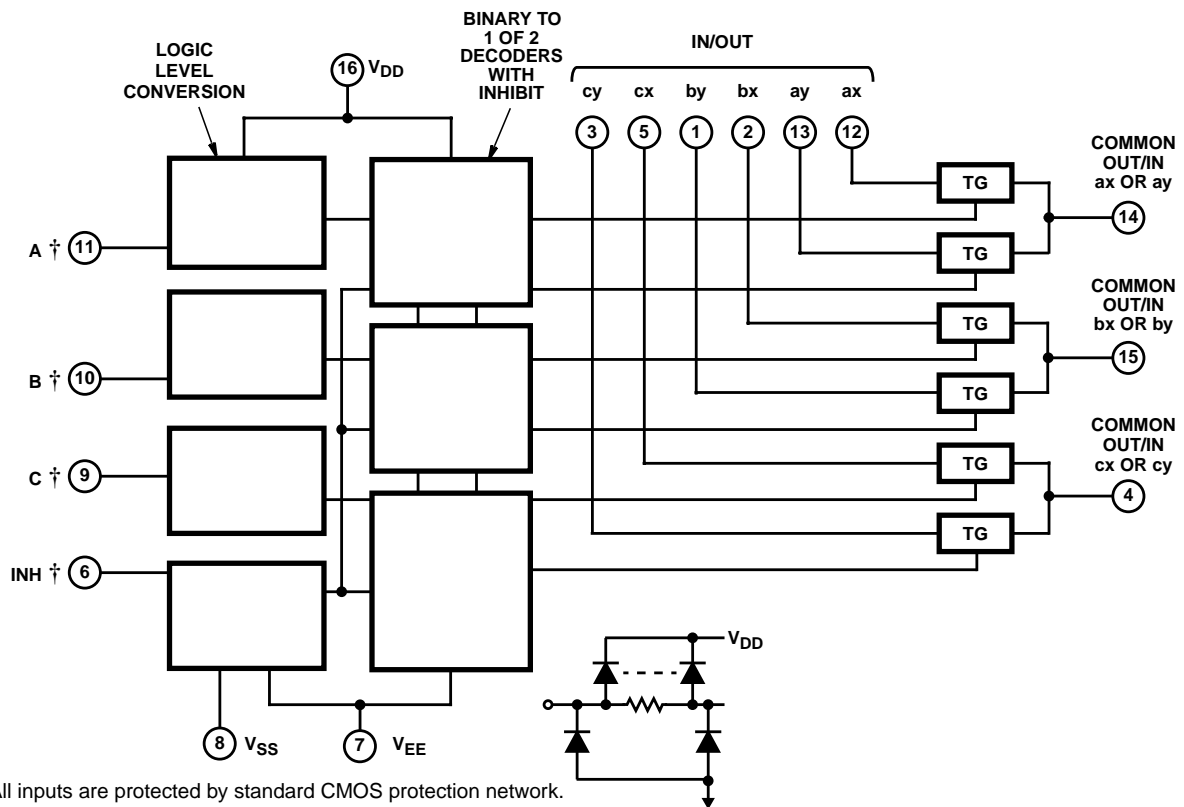
CD4051B, CD4052B, CD4053B

Functional Block Diagrams (Continued)

CD4052B



CD4053B



† All inputs are protected by standard CMOS protection network.

CD4051B, CD4052B, CD4053B

TRUTH TABLES

INPUT STATES				"ON" CHANNEL(S)
INHIBIT	C	B	A	
CD4051B				
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	X	X	X	None
CD4052B				
INHIBIT	B		A	
0	0		0	0x, 0y
0	0		1	1x, 1y
0	1		0	2x, 2y
0	1		1	3x, 3y
1	X		X	None
CD4053B				
INHIBIT	A OR B OR C			
0	0			ax or bx or cx
0	1			ay or by or cy
1	X			None

X = Don't Care

CD4051B, CD4052B, CD4053B

Absolute Maximum Ratings

Supply Voltage (V+ to V-)	
Voltages Referenced to V _{SS} Terminal	-0.5V to 20V
DC Input Voltage Range	-0.5V to V _{DD} +0.5V
DC Input Current, Any One Input	±10mA

Operating Conditions

Temperature Range	-55°C to 125°C
-------------------	----------------

Thermal Information

Thermal Resistance (Typical, Note 1)	θ_{JA} (°C/W)	θ_{JC} (°C/W)
E Package	67	N/A
F Package	115	45
D Package	73	N/A
NS Package	64	N/A
PW Package	108	N/A
Maximum Junction Temperature (Ceramic Package)	175°C	
Maximum Junction Temperature (Plastic Package)	150°C	
Maximum Storage Temperature Range	-65°C to 150°C	
Maximum Lead Temperature (Soldering 10s)	265°C (SOIC - Lead Tips Only)	


CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

NOTE:

- The package thermal impedance is calculated in accordance with JESD51.

Electrical Specifications

Common Conditions Here: If Whole Table is For the Full Temp. Range, V_{SUPPLY} = ±5V, A_V = +1, R_L = 100Ω, Unless Otherwise Specified (Note 3)

PARAMETER	CONDITIONS				LIMITS AT INDICATED TEMPERATURES (°C)							UNITS
	V _{IS} (V)	V _{EE} (V)	V _{SS} (V)	V _{DD} (V)	-55	-40	85	125	25			
									MIN	TYP	MAX	
SIGNAL INPUTS (V_{IS}) AND OUTPUTS (V_{OS})												
Quiescent Device Current, I _{DD} Max	-	-	-	5	5	5	150	150	-	0.04	5	μA
	-	-	-	10	10	10	300	300	-	0.04	10	μA
	-	-	-	15	20	20	600	600	-	0.04	20	μA
	-	-	-	20	100	100	3000	3000	-	0.08	100	μA
Drain to Source ON Resistance r _{ON} Max 0 ≤ V _{IS} ≤ V _{DD}	-	0	0	5	800	850	1200	1300	-	470	1050	Ω
	-	0	0	10	310	330	520	550	-	180	400	Ω
	-	0	0	15	200	210	300	320	-	125	240	Ω
Change in ON Resistance (Between Any Two Channels), Δr _{ON}	-	0	0	5	-	-	-	-	-	15	-	Ω
	-	0	0	10	-	-	-	-	-	10	-	Ω
	-	0	0	15	-	-	-	-	-	5	-	Ω
OFF Channel Leakage Current: Any Channel OFF (Max) or ALL Channels OFF (Common OUT/IN) (Max)	-	0	0	18	±100 (Note 2)		±1000 (Note 2)		-	±0.01	±100 (Note 2)	nA
Capacitance: Input, C _{IS} Output, C _{OS} CD4051 CD4052 CD4053 Feedthrough C _{IOS}	-	-5	5-	5	-	-	-	-	-	5	-	pF
					-	-	-	-	-	30	-	pF
					-	-	-	-	-	18	-	pF
					-	-	-	-	-	9	-	pF
					-	-	-	-	-	0.2	-	pF
Propagation Delay Time (Signal Input to Output)		R _L = 200kΩ, C _L = 50pF, t _r , t _f = 20ns	5	-	-	-	-	-	-	30	60	ns
			10	-	-	-	-	-	15	30	ns	
			15	-	-	-	-	-	10	20	ns	

CD4051B, CD4052B, CD4053B

Electrical Specifications Common Conditions Here: If Whole Table is For the Full Temp. Range, $V_{SUPPLY} = \pm 5V$, $A_V = +1$, $R_L = 100\Omega$, Unless Otherwise Specified **(Continued)** (Note 3)

PARAMETER	CONDITIONS				LIMITS AT INDICATED TEMPERATURES (°C)							UNITS
	V_{IS} (V)	V_{EE} (V)	V_{SS} (V)	V_{DD} (V)	-55	-40	85	125	25			
									MIN	TYP	MAX	
CONTROL (ADDRESS OR INHIBIT), V_C												
Input Low Voltage, V_{IL} , Max	$V_{IL} = V_{DD}$ through $1k\Omega$; $V_{IH} = V_{DD}$ through $1k\Omega$	$V_{EE} = V_{SS}$, $R_L = 1k\Omega$ to V_{SS} , $I_{IS} < 2\mu A$ on All OFF Channels	5	1.5	1.5	1.5	1.5	-	-	1.5	V	
			10	3	3	3	3	-	-	3	V	
			15	4	4	4	4	-	-	4	V	
Input High Voltage, V_{IH} , Min	$V_{IL} = V_{DD}$ through $1k\Omega$	$V_{EE} = V_{SS}$, $R_L = 1k\Omega$ to V_{SS} , $I_{IS} < 2\mu A$ on All OFF Channels	5	3.5	3.5	3.5	3.5	3.5	-	-	V	
			10	7	7	7	7	7	-	-	V	
			15	11	11	11	11	11	-	-	V	
Input Current, I_{IN} (Max)	$V_{IN} = 0, 18$		18	± 0.1	± 0.1	± 1	± 1	-	$\pm 10^{-5}$	± 0.1	μA	
Propagation Delay Time: Address-to-Signal OUT (Channels ON or OFF) See Figures 10, 11, 14	$t_r, t_f = 20ns$, $C_L = 50pF$, $R_L = 10k\Omega$	0	0	5	-	-	-	-	-	450	720	ns
		0	0	10	-	-	-	-	-	160	320	ns
		0	0	15	-	-	-	-	-	120	240	ns
		-5	0	5	-	-	-	-	-	225	450	ns
Propagation Delay Time: Inhibit-to-Signal OUT (Channel Turning ON) See Figure 11	$t_r, t_f = 20ns$, $C_L = 50pF$, $R_L = 1k\Omega$	0	0	5	-	-	-	-	-	400	720	ns
		0	0	10	-	-	-	-	-	160	320	ns
		0	0	15	-	-	-	-	-	120	240	ns
		-10	0	5	-	-	-	-	-	200	400	ns
Propagation Delay Time: Inhibit-to-Signal OUT (Channel Turning OFF) See Figure 15	$t_r, t_f = 20ns$, $C_L = 50pF$, $R_L = 10k\Omega$	0	0	5	-	-	-	-	-	200	450	ns
		0	0	10	-	-	-	-	-	90	210	ns
		0	0	15	-	-	-	-	-	70	160	ns
		-10	0	5	-	-	-	-	-	130	300	ns
Input Capacitance, C_{IN} (Any Address or Inhibit Input)				-	-	-	-	-	5	7.5	pF	

NOTE:

- Determined by minimum feasible leakage measurement for automatic testing.

Electrical Specifications

PARAMETER	TEST CONDITIONS			LIMITS	UNITS		
	V_{IS} (V)	V_{DD} (V)	R_L (k Ω)				
Cutoff (-3dB) Frequency Channel ON (Sine Wave Input)	5 (Note 3)	10	1	V_{OS} at Common OUT/IN	CD4053	30	MHz
					CD4052	25	MHz
					CD4051	20	MHz
				$V_{EE} = V_{SS}$, $20 \text{Log} \frac{V_{OS}}{V_{IS}} = -3dB$			V_{OS} at Any Channel

CD4051B, CD4052B, CD4053B

Electrical Specifications

PARAMETER	TEST CONDITIONS			LIMITS		UNITS	
	V _{IS} (V)	V _{DD} (V)	R _L (kΩ)	TYP			
Total Harmonic Distortion, THD	2 (Note 3)	5	10		0.3	%	
	3 (Note 3)	10			0.2	%	
	5 (Note 3)	15	0.12		%		
	V _{EE} = V _{SS} , f _{IS} = 1kHz Sine Wave						%
-40dB Feedthrough Frequency (All Channels OFF)	5 (Note 3)	10	1	V _{OS} at Common OUT/IN	CD4053	8	MHz
	V _{EE} = V _{SS} , $20 \log \frac{V_{OS}}{V_{IS}} = -40\text{dB}$				CD4052	10	MHz
					CD4051	12	MHz
				V _{OS} at Any Channel		8	MHz
-40dB Signal Crosstalk Frequency	5 (Note 3)	10	1	Between Any 2 Channels		3	MHz
	V _{EE} = V _{SS} , $20 \log \frac{V_{OS}}{V_{IS}} = -40\text{dB}$			Between Sections, CD4052 Only	Measured on Common	6	MHz
					Measured on Any Channel	10	MHz
				Between Any Two Sections, CD4053 Only	In Pin 2, Out Pin 14	2.5	MHz
			In Pin 15, Out Pin 14		6	MHz	
Address-or-Inhibit-to-Signal Crosstalk	-	10	10 (Note 4)			65	mV _{PEAK}
	V _{EE} = 0, V _{SS} = 0, t _r , t _f = 20ns, V _{CC} = V _{DD} - V _{SS} (Square Wave)					65	mV _{PEAK}

NOTES:

- Peak-to-Peak voltage symmetrical about $\frac{V_{DD} - V_{EE}}{2}$
- Both ends of channel.

Typical Performance Curves

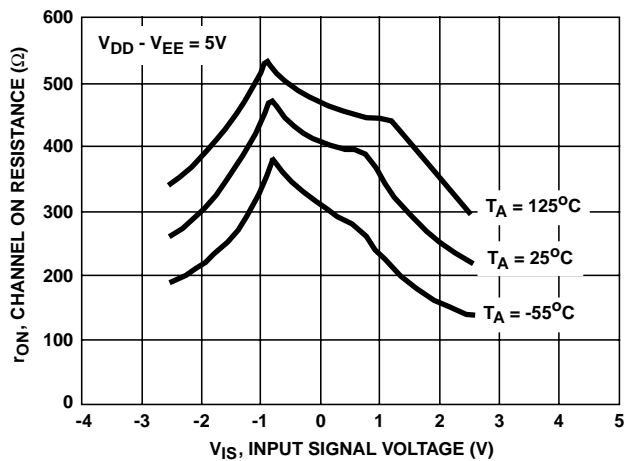


FIGURE 1. CHANNEL ON RESISTANCE vs INPUT SIGNAL VOLTAGE (ALL TYPES)

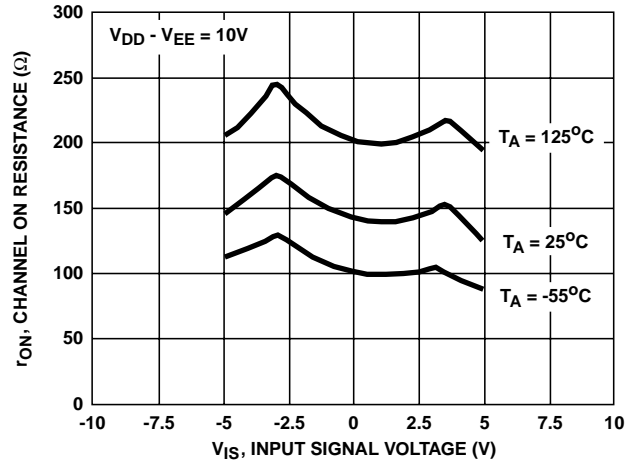


FIGURE 2. CHANNEL ON RESISTANCE vs INPUT SIGNAL VOLTAGE (ALL TYPES)

Typical Performance Curves (Continued)

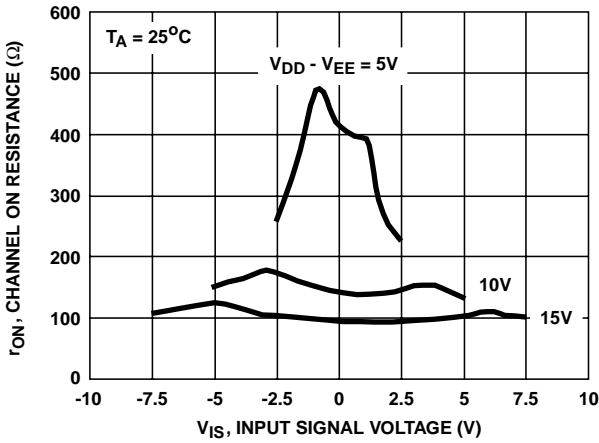


FIGURE 3. CHANNEL ON RESISTANCE vs INPUT SIGNAL VOLTAGE (ALL TYPES)

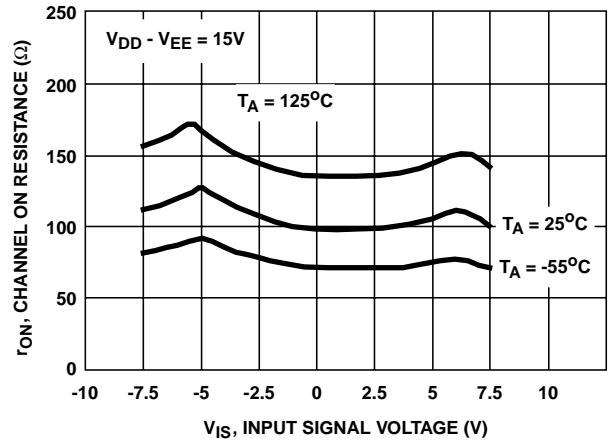


FIGURE 4. CHANNEL ON RESISTANCE vs INPUT SIGNAL VOLTAGE (ALL TYPES)

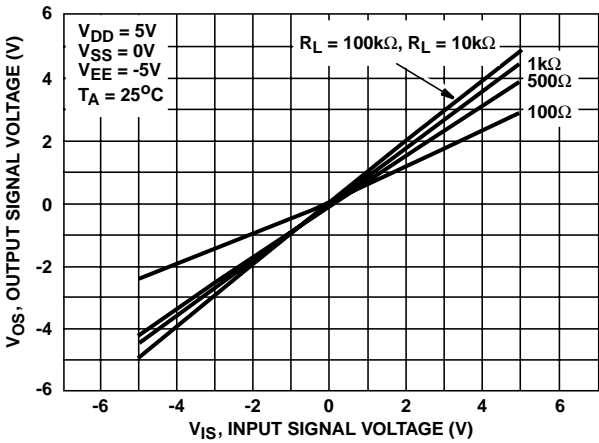


FIGURE 5. ON CHARACTERISTICS FOR 1 OF 8 CHANNELS (CD4051B)

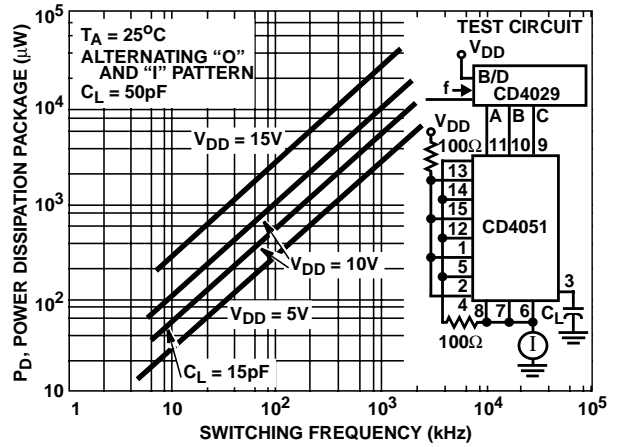


FIGURE 6. DYNAMIC POWER DISSIPATION vs SWITCHING FREQUENCY (CD4051B)

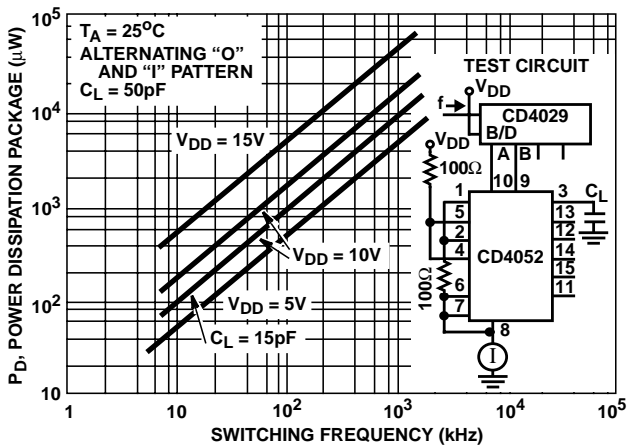


FIGURE 7. DYNAMIC POWER DISSIPATION vs SWITCHING FREQUENCY (CD4052B)

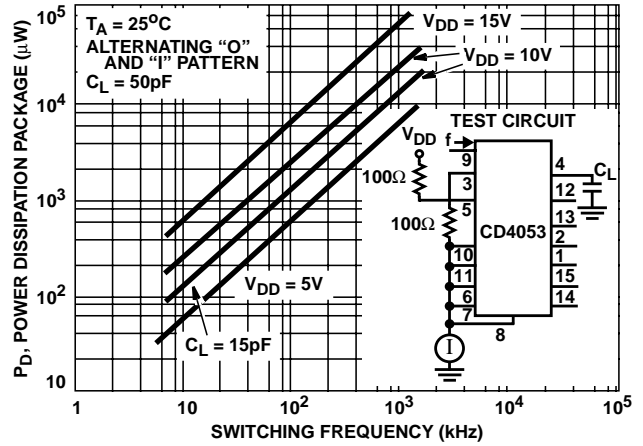
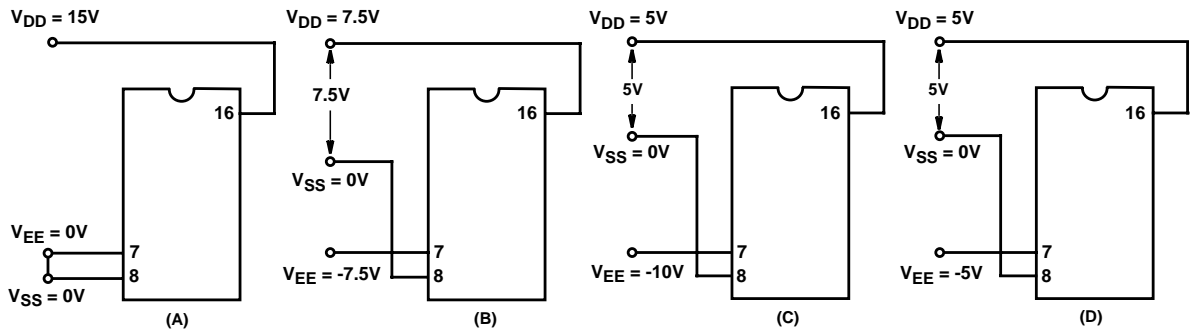


FIGURE 8. DYNAMIC POWER DISSIPATION vs SWITCHING FREQUENCY (CD4053B)

Test Circuits and Waveforms



NOTE: The ADDRESS (digital-control inputs) and INHIBIT logic levels are: "0" = V_{SS} and "1" = V_{DD} . The analog signal (through the TG) may swing from V_{EE} to V_{DD} .

FIGURE 9. TYPICAL BIAS VOLTAGES

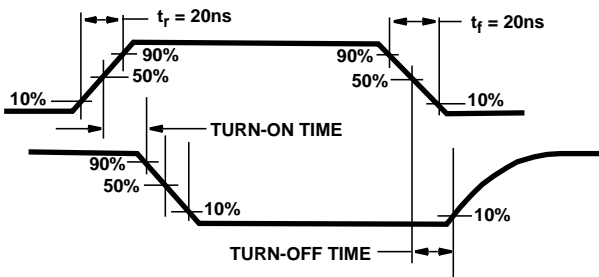


FIGURE 10. WAVEFORMS, CHANNEL BEING TURNED ON ($R_L = 1k\Omega$)

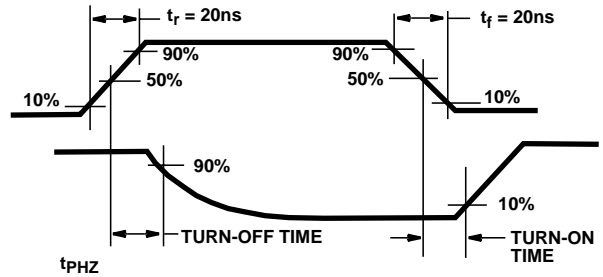


FIGURE 11. WAVEFORMS, CHANNEL BEING TURNED OFF ($R_L = 1k\Omega$)

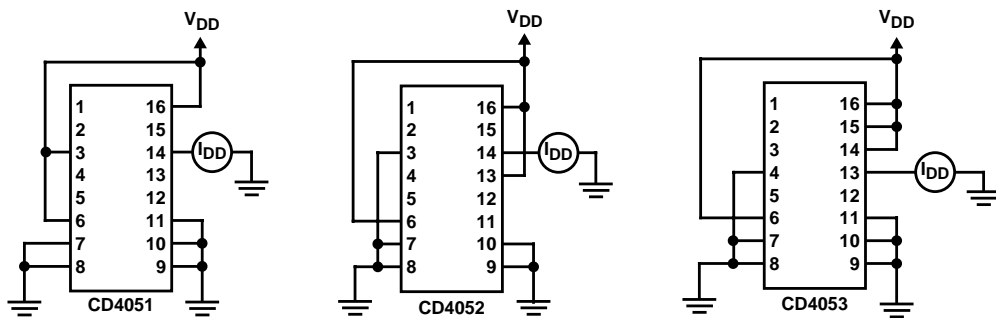


FIGURE 12. OFF CHANNEL LEAKAGE CURRENT - ANY CHANNEL OFF

CD4051B, CD4052B, CD4053B

Test Circuits and Waveforms (Continued)

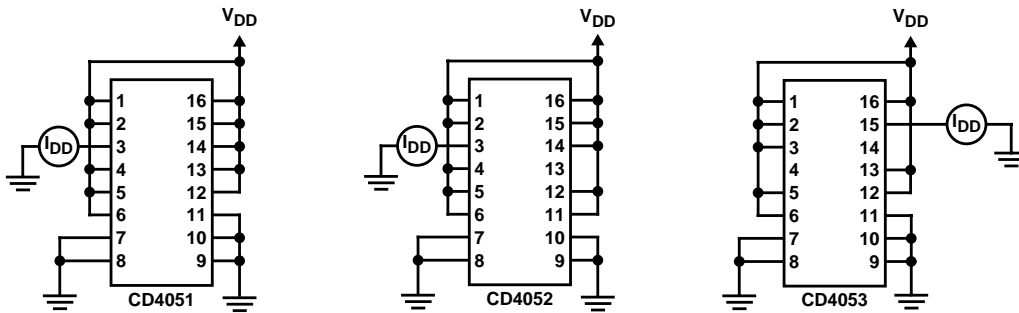


FIGURE 13. OFF CHANNEL LEAKAGE CURRENT - ALL CHANNELS OFF

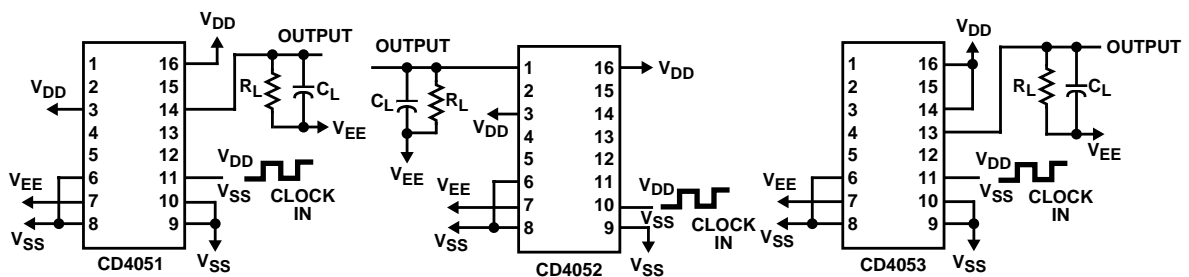


FIGURE 14. PROPAGATION DELAY - ADDRESS INPUT TO SIGNAL OUTPUT

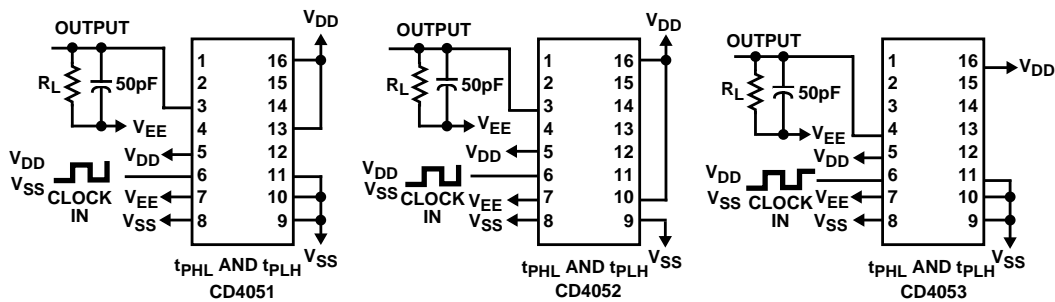


FIGURE 15. PROPAGATION DELAY - INHIBIT INPUT TO SIGNAL OUTPUT

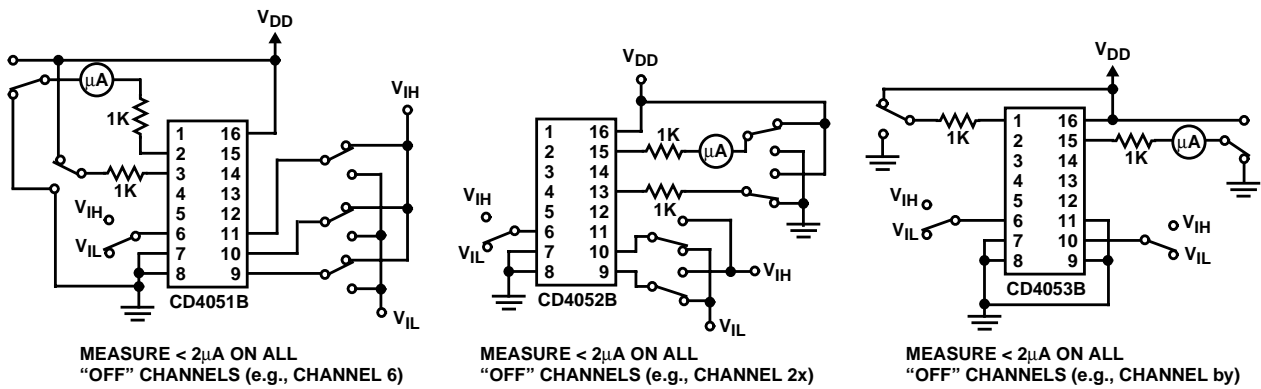


FIGURE 16. INPUT VOLTAGE TEST CIRCUITS (NOISE IMMUNITY)

Test Circuits and Waveforms (Continued)

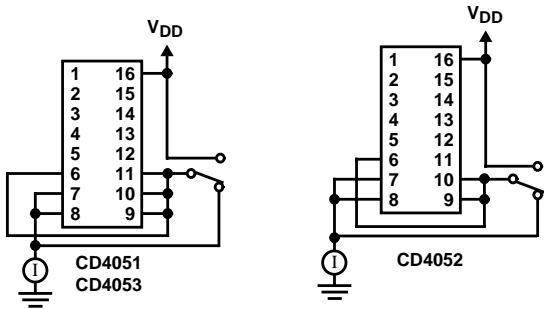


FIGURE 17. QUIESCENT DEVICE CURRENT

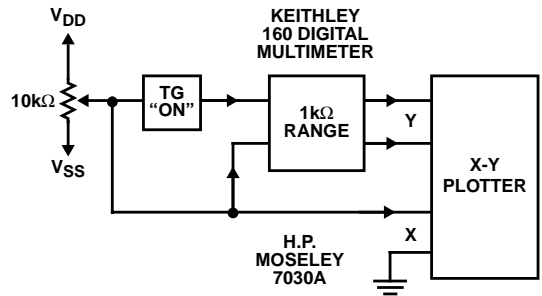


FIGURE 18. CHANNEL ON RESISTANCE MEASUREMENT CIRCUIT

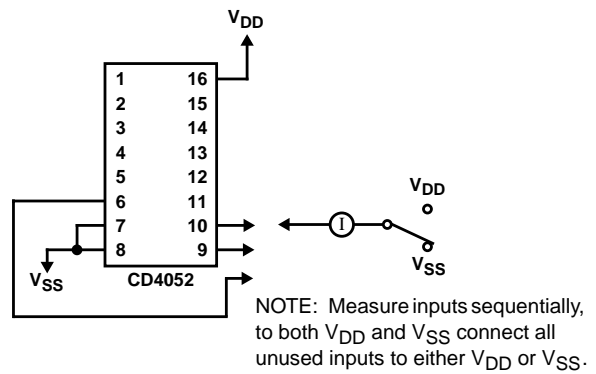
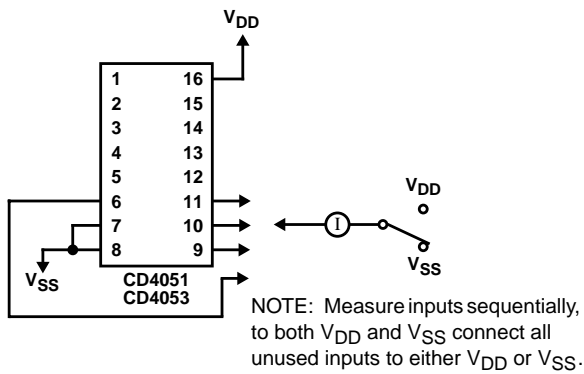


FIGURE 19. INPUT CURRENT

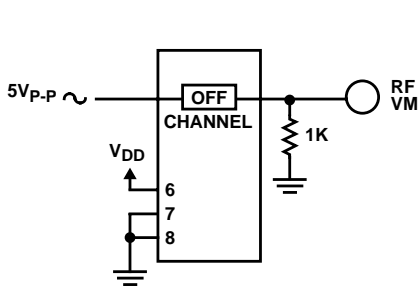


FIGURE 20. FEEDTHROUGH (ALL TYPES)

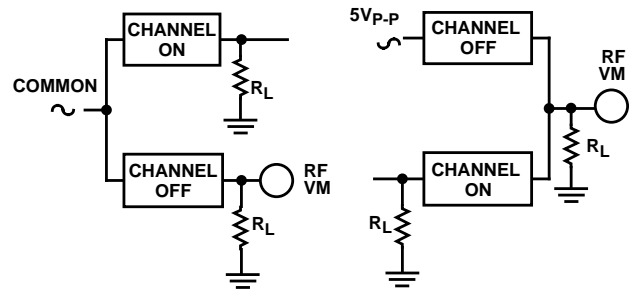


FIGURE 21. CROSSTALK BETWEEN ANY TWO CHANNELS (ALL TYPES)

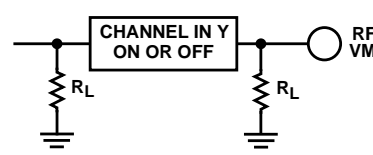
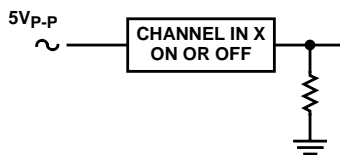


FIGURE 22. CROSSTALK BETWEEN DUALS OR TRIPLETS (CD4052B, CD4053B)

Test Circuits and Waveforms (Continued)

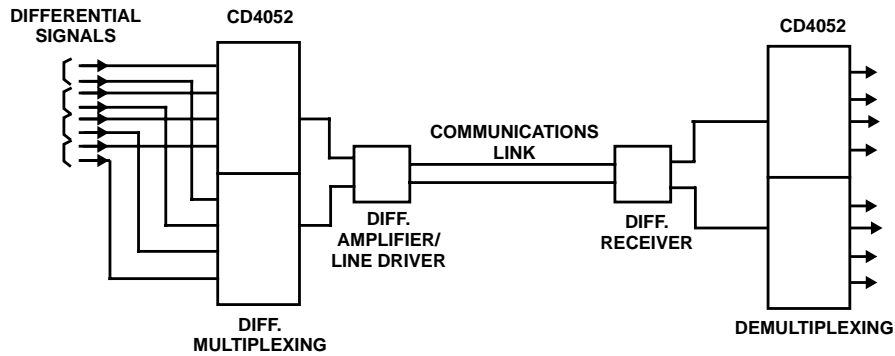


FIGURE 23. TYPICAL TIME-DIVISION APPLICATION OF THE CD4052B

Special Considerations

In applications where separate power sources are used to drive V_{DD} and the signal inputs, the V_{DD} current capability should exceed V_{DD}/R_L (R_L = effective external load). This provision avoids permanent current flow or clamp action on the V_{DD} supply when power is applied or removed from the CD4051B, CD4052B or CD4053B.

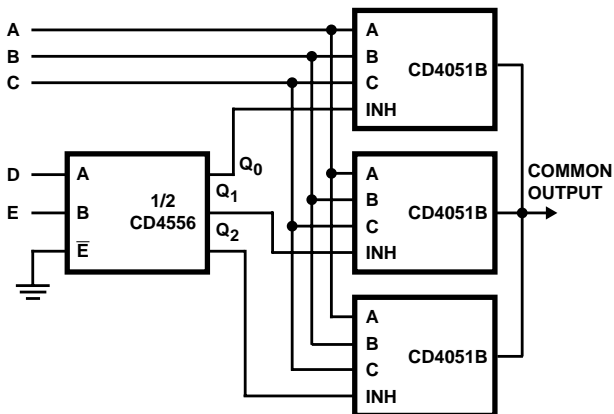


FIGURE 24. 24-TO-1 MUX ADDRESSING

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgment, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Customers are responsible for their applications using TI components.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.

