



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



N° d'ordre : M...../GE/2019

MEMOIRE

Présenté pour obtenir le diplôme de

MASTER EN GENIE ELECTRIQUE

Option : Electronique des Systèmes Embarqués

Par :

TOURE Fatoumata Fousseyni & REZIK Kheira

Intitulé :

Contrôle et Commande d'un système à distance via GSM

Soutenu le 10/07/2019 devant le jury composé de :

Président :	M.REBHI Moustapha	MAA	Université de Mostaganem
Examineur 1 :	M.BENTOUMI Mohamed	MCB	Université de Mostaganem
Examineur 2 :	Mlle DENDANI Djazia	MAA	Université de Mostaganem
Encadrante :	Mme Benchellal Amel	MCB	Université de Mostaganem
Co-Encadrant :	M.BENAOULI Mohamed	MAA	Université de Mostaganem

Année Universitaire 2018/2019

Remerciements

Après avoir rendu grâce à Allah le Tout Puissant et Le Tout-Miséricordieux de nous avoir donné la force nécessaire et le courage d'accomplir ce travail ainsi que tout au long de notre étude.

*Nous tenons à remercier les personnes qui, de près ou de loin ont participé à l'élaboration de ce mémoire, à commencer par notre Encadrante **Mme. Benchellal** et notre Co-encadrant **M. Benaouali** pour leurs disponibilités, leurs aides, leurs conseils et encouragements. Nous remercions très vivement tous nos professeurs, nos promotionnaires, ami(e)s et tous nos **proches** qui ont contribué à la réalisation de ce projet.*

Dédicace

Je dédie ce modeste travail aux personnes qui sont chères à mon cœur :

***Mes parents** sans qui, je ne serais pas là où j'en suis maintenant. Pour tous leurs sacrifices, leurs prières, leurs amours et leurs soutiens.*

***Mes Frères** pour leurs appuis et leurs encouragements permanents.*

*A toute **ma Famille** pour leur soutien tout au long de mon parcours universitaire.*

*Mes **Ami(e)s** ainsi que toutes **les personnes** qui m'ont soutenu et aidé tout au long de ce projet de fin d'étude.*

♥ **TOURE Fatoumata Fousseyni** ♥

Dédicace

Je dédie ce projet :

*A mes **chers parents**, qui n'ont jamais cessé de formuler des prières à mon égard de me soutenir et de m'épauler pour que je puisse atteindre mes objectifs.*

*A mon grand frère **Abdelghani** et sa femme et son fils (**Yanis**).*

*A mes frères **Abdelhalim, Abdelhak** et **Achref**.*

*A ma petite sœur **Yasmine**.*

*A ma chère sœur et son mari et ses enfants (**Badro** et **Issehak**) pour leurs soutiens moral et leurs précieux conseils tout au long de mes études.*

*A mon oncle **Abdelkader**, qui je souhaite une bonne santé.*

*A ma chère binôme **FATOUMATA** pour son entente et sa sympathie.*

*A mes ami(e)s **Mohammed, Billel** et **Hayet** qui m'ont supporté dans les moments difficiles.*

A toute ma famille.

A tous mes autres ami(e)s et mes camarades.

Sans oublier tous les professeurs que ce soit du primaire, du moyen, du secondaire ou de l'enseignement supérieur.

A tous que j'aime et ceux qui m'aiment.

Merci d'être toujours là pour moi.

♥ **REZIK Kheira** ♥

Résumé

Notre projet consiste à la réalisation d'un système contrôlé et commandé à distance via GSM. Plus concrètement à la commande et contrôle de l'arrosage du jardin d'un lieu public par l'envoi et la réception de message sur smartphone, en y ajoutant la commande d'éclairage automatique du lieu avec son adaptation à la lumière jour/nuit.

Notre mémoire est subdivisé en trois principaux chapitres. Le premier chapitre intitulé **Généralité sur la télé-irrigation et éclairage intelligent**, portera sur une étude générale de l'irrigation et l'éclairage intelligent, les différents types, les méthodes, et leurs impacts sur la société. Le second chapitre intitulé **Partie matérielle et logicielle** se consacre à la présentation de l'ensemble des composants utilisés, logiciel utilisé et les commandes nécessaire à la réalisation du projet. Et dans le dernier chapitre intitulé **Conception et réalisation pratique**, on parlera de la réalisation pratique ; et il s'agira bien sûr d'un système fonctionnel et prêt à être exploité.

Summary

The purpose of our project is to realize a system remotely controlled and commanded via GSM; In concrete terms to the command and control of the watering of a public place garden by sending and receiving message on a smartphone, adding to it the automatic lighting command of the Public garden with regard to daylight and night.

Our dissertation is divided into three main chapters. The first chapter titled **GENERALITY ON TELE-IRRIGATION AND INTELLIGENT LIGHTING**, will deal with a general study on irrigation and intelligent lighting, the different types, the different methods, and their impacts on the society. The second chapter titled **MATERIAL AND SOFTWARE PARTS** is devoted to the presentation of the whole used device, used software and the commands necessary to the project realization. And in the last chapter titled **CONCEPTION AND PRACTICAL REALIZATION**, we will talk about the Practical Realization; and for sure it will be a functional system and ready to be exploited.

Table des matières

<i>Liste des Figures</i>	1
Liste des Tableau	2
<i>Introduction générale</i>	3
I. Situation actuelle	3
II. Motif du choix du projet :.....	3
III. Objectifs du projet :	3
<i>Chapitre I : Généralités sur L'irrigation et l'éclairage intelligent</i>	4
1.1 Introduction.....	5
1.2 La télé-irrigation	5
1.2.1 Les techniques d'irrigation.....	5
1.2.1.a Le système d'irrigation par ruissellement ou par déversement.....	5
1.2.1.b Le système d'irrigation par submersion ou inondation	6
1.2.1.c Irrigation par infiltration.....	6
1.2.1.d Le système d'irrigation par aspersion	6
1.2.1.e La micro irrigation (goutte-à-goutte)	7
1.2.2 Matériel de la télé-irrigation.....	7
1.2.2.a 1 ^{ère} Catégorie :.....	7
1.2.2.b 2 ^{ème} Catégorie :.....	7
1.3 L'éclairage Intelligent.....	8
1.3.1 Domaines d'application.....	8
1.3.1.a L'éclairage extérieur intelligent	8
1.3.1.b L'éclairage intérieur	8
1.3.2 Matériels utilisés pour l'éclairage intelligent.....	9
1.3.2.1 Les capteurs :.....	9
1.3.2.2 Les actionneurs :.....	9
1.4 Système de commande centralisée et commande décentralisée	10
1.4.1 Système centralisée	10
1.4.2 Système décentralisée	10
1.5 Conclusion	11
<i>Chapitre II : Partie matérielle et logicielle</i>	12
2.1 Introduction.....	13

2.2	Cahier des Charges souhaité	13
A.	Partie Matérielle	13
2.3	Arduino UNO.....	13
2.3.1	Présentation de la carte Arduino UNO.....	13
2.4	L'Arduino GSM Shield 2.....	15
2.4.1	Description de l'Arduino Shield 2	16
2.4.2	Caractéristiques :.....	17
2.4.3	Brochage :	17
2.5	La photorésistance LDR	18
2.6	Capteur d'humidité (Soil Humidity sensor).....	19
2.7	Capteur LM35	19
2.8	Détecteur de niveau d'eau	20
2.9	La LED.....	21
2.10	Micro pompe à eau avec connecteur USB.....	21
2.11	2channels 5v Relay Module.....	21
B.	Partie Logicielle.....	22
2.12	Logiciel Arduino	22
2.13	Les commandes AT	24
2.11.1	Les types des commandes AT et les réponses.....	24
2.11.2	Commandes AT dédiées au service SMS :	25
2.14	Conclusion	25
	<i>Chapitre III : réalisation pratique du système</i>	26
3.1	Introduction :.....	27
3.2	Organigramme du Programme.....	27
3.3	Schéma électrique et principe de fonctionnement du système	29
3.3.1	Schéma électrique	29
3.3.2	Principe de fonctionnement du système.....	29
3.4	Commande de l'éclairage.....	30
3.4.1	Simulation sur Proteus ISIS	31
3.5	Commande de l'arrosage	32
3.5.1	Mise en marche du module GSM Shield 2	32
3.5.2	Bloc capteur d'humidité	34
3.5.3	Réception SMS via GSM	35

3.5.4	Bloc capteur de température.....	35
3.5.5	Bloc des pompes et le capteur de niveau d'eau.....	36
3.6	Système sur plaque d'essai	37
3.7	Brochage des éléments de notre système avec Arduino Uno	38
3.8	Implémentation sur maquette.....	38
3.9	Tests de fonctionnalité	40
3.10	Coût de réalisation	40
3.11	Conclusion	41
	<i>Conclusion générale et perspectives</i>	42
	<i>Références Bibliographiques</i>	43
	<i>Références Figures</i>	43
	<i>Liste des abréviations</i>	44
	Annexe1	45
	Annexe2	47

Liste des Figures

Figure 1. 1 : Irrigation par Ruissellement [1]	5
Figure 1. 2 : Irrigation par submersion [2]	6
Figure 1. 3 : Irrigation par Infiltration [3]	6
Figure 1. 4 : Irrigation par Aspersion [4]	6
Figure 1. 5 : Micro-Irrigation [5]	7
Figure 1. 6 : Eclairage extérieur [6]	8
Figure 1. 7 : Gestion des infrastructures et bureaux [6]	9
Figure 1. 8 : Système Centralisé [7]	10
Figure 1. 9 : Système Décentralisé[8]	10
Figure 2. 1 : Présentation du cahier des charges	13
Figure 2. 2 : Présentation de la carte Arduino[10]	14
Figure 2. 3 : Arduino GSM Shield 2 [9]	16
Figure 2. 4 : Vue d'ensemble GSM Shield 2	16
Figure 2. 5 : Brochage[9]	18
Figure 2. 6 : La photorésistance(LDR)[11]	18
Figure 2. 7 : Capteur d'humidité[12]	19
Figure 2. 8 : Capteur LM35[13]	19
Figure 2. 9 : Détecteur de niveau d'eau[14]	20
Figure 2. 10 : Image LED.....	21
Figure 2. 11 : Micro pompe à eau[15].....	21
Figure 2. 12 : Module à 2canaux relais [16].....	22
Figure 2. 13 : Interface d'Arduino	22
Figure 2. 14 : Les boutons de l'interface Arduino	23
Figure 2. 15 : Schéma de fonctionnement des commandes AT[17]	24
Figure 3. 1 : L'organigramme du programme	28
Figure 3. 2 : Schéma électrique du système	29
Figure 3. 3 : Simulation ISIS Allumage automatique LED éteintes	31
Figure 3. 4 : Simulation ISIS Allumage automatique LED allumées	32
Figure 3. 5 : allumage manuel sous ISIS	32
Figure 3. 6 : Insertion de la carte SIM.....	33
Figure 3. 7 : Connexion du Shield avec un Arduino UNO	33
Figure 3. 8 : Shield sous tension	33
Figure 3. 9 : Brochage du capteur d'humidité.....	34
Figure 3. 10 : Brochage capteur de température	35
Figure 3. 11 : Affichage de la température sur le moniteur de l'Arduino	36
Figure 3. 12 : Brochage des pompes	36
Figure 3. 13 : Brochage Capteur de niveau	37
Figure 3. 15 : Bloc d'éclairage sur plaque d'essai	37
Figure 3. 16 : Circuit Complet sur plaque d'essai	38
Figure 3. 17 : Plancher de bois	39

Figure 3. 18 : Délimitation des parties	39
Figure 3. 19 : Image maquette finale du Projet	40

Liste des Tableau

Tableau 2. 1 : Caractéristique du GSM Shield 2 Arduino [9].....	17
Tableau 2. 2 : Commande AT dédiées service SMS [15]	25
Tableau 3. 1 : Valeurs du capteur de lumière dans différents conditions	30
Tableau 3. 2 : Valeurs du capteur d'humidité dans différents conditions.....	34
Tableau 3. 3 : Brochage des Composants	38
Tableau 3. 4 : Ensemble des composants et leurs prix.....	41

Introduction générale

I. Situation actuelle

De nos jours nous assistons à une forte évolution de la technologie, plus particulièrement dans le domaine de la communication notamment les « smartphones ». Cet outil nous apporte beaucoup plus de fonctionnalité, en dehors de son utilisation pour les appels téléphoniques. On peut les utiliser pour renforcer notre sécurité et notre confort de près comme à une petite ou grande distance via Bluetooth qui est déjà intégré dans le smartphone ou un Module GSM.

Actuellement le réseau GSM compte des millions d'utilisateurs. Ceci donne la possibilité via l'envoi et la réception des SMS de commander et contrôler un système/processus quelconque.

Sur ce, l'objet de notre mémoire est de commander et contrôler (l'éclairage et l'arrosage) du jardin d'une structure publique par smartphone via GSM.

II. Motif du choix du projet :

Trois (3) points nous ont motivés pour le choix de ce projet qui sont :

- Commander à distance une structure publique.
- Avoir un contrôle sur la consommation de l'eau et de l'électricité : afin d'éviter les gaspillages inutiles.
- Le confort de l'utilisateur.

III. Objectifs du projet :

- Regrouper suffisamment d'informations sur la carte développement Arduino : son langage de programmation, son principe de fonctionnement.
- Comprendre le principe de fonctionnement du Module GSM : comment se fait l'envoi et la lecture d'un SMS ?
- Réaliser un système capable d'exécuter une action entre un smartphone, un Module GSM et une carte Arduino en expliquant les différents blocs.

Chapitre I : Généralités sur L'irrigation et l'éclairage intelligent

1.1 Introduction

La forte augmentation des ventes de smartphone et de tablettes électroniques se fait en même temps qu'une adoption rapide par le grand public des technologies de la domotique ainsi que l'autopilotage. Le smartphone devient une télécommande universelle pour toute la maison et les équipements électriques d'où la télé-irrigation qui est l'objet de notre projet. Par conséquent, nous allons présenter un aperçu général sur les systèmes d'irrigations et d'éclairages intelligents. Les différents types d'irrigation existant et d'éclairage en fonction de leurs fonctions à remplir selon leur manière de fonctionner.

1.2 La télé-irrigation

L'irrigation est un apport artificiel d'eau aux cultures, en complément aux précipitations naturelles, l'objectif est de créer des conditions favorables de production, tant au point de vue quantitatif que qualitatif [1].

La télé-irrigation permet de contrôler à distance une vanne ou une pompe et de collecter des données sur le terrain comme :

- ✓ La pression
- ✓ La température
- ✓ Le débit
- ✓ L'humidité dans le sol
- ✓

1.2.1 Les techniques d'irrigation

Elles peuvent être rangées en cinq classes que l'on nomme les systèmes d'arrosage :

- Le système d'irrigation par ruissellement ou par déversement,
- Le système d'irrigation par submersion ou inondation,
- Le système d'irrigation par infiltration,
- Le système d'irrigation par aspersion,
- La micro irrigation (le goutte-à-goutte).

1.2.1.a Le système d'irrigation par ruissellement ou par déversement

Elle consiste à faire couler sur un sol en pente une mince lame d'eau (figure 1.1) pendant le temps nécessaire pour que la terre s'humecte jusqu'à la profondeur voulue [1].



Figure 1. 1 : Irrigation par Ruissellement [1]

1.2.1.b Le système d'irrigation par submersion ou inondation

Pour cette technique la pratique se fait en recouvrant le sol d'une couche d'eau plus ou moins épaisse (figure 1-2) et en la laissant séjourner pendant le temps nécessaire pour qu'elle pénètre par infiltration à la profondeur utile au développement des cultures [1].



Figure 1. 2 : Irrigation par submersion [2]

1.2.1.c Irrigation par infiltration

Elle se diffère totalement des deux méthodes précédentes. Repartie dans un ensemble dense de rigoles (voir Figure 1.3), l'eau y reste sans jamais déborder et s'infiltré dans le sol latéralement ou de haut en bas, humidifiant ainsi toute la masse de terre comprise entre les rigoles [1].



Figure 1. 3 : Irrigation par Infiltration [3]

1.2.1.d Le système d'irrigation par aspersion

Elle est relativement peu pratique en Afrique tropicale. Cette technique consiste à imiter l'effet des précipitations : l'eau, acheminée sous pression par des tuyaux flexibles, est propulsée en l'air sous forme de gouttelettes (voir Figure 1.4), lesquelles retombent sur les cultures autour de chaque asperseur [2].



Figure 1. 4 : Irrigation par Aspersion [4]

1.2.1.e La micro irrigation (goutte-à-goutte)

La micro irrigation autrement appelé le goutte-à-goutte est une technique où l'eau s'égoutte lentement vers les racines des plantes par un système de tuyaux et goutteurs (Figure 1.5), soit en coulant à la surface du sol soit en irriguant directement la rhizosphère [2].



Figure 1. 5 : Micro-Irrigation [5]

Notre objectif étant l'arrosage d'un jardin gazonné avec des plantes, nous sommes intéressés seulement par le dernier type qui est l'irrigation goutte-à-goutte pour l'arrosage des plantes du jardin de la structure.

1.2.2 Matériel de la télé-irrigation

On peut distinguer deux catégories de matériels ou d'installations nécessaires à l'irrigation :

1.2.2.a 1^{ère} Catégorie :

Ceux-ci servent à amener l'eau depuis les sources disponibles (cours d'eau, lacs ou retenues, nappe phréatique).

Dans cette catégorie on trouve :

- Forage,
- Pompes,
- Réseaux d'irrigations,
- ...

1.2.2.b 2^{ème} Catégorie :

Ceux servant à l'irrigation proprement dite, faisant la distribution d'eau aux plantes.

Elle peut comporter :

- Les asperseurs,
- Les canons d'arrosage,
- Des goutteurs,
- Tuyaux
- ...

1.3 L'éclairage Intelligent

Les systèmes d'éclairages intelligents permettent une forte économie d'énergie. Ces réductions de consommation sont réalisables du fait que les lampes utilisées dans de tels systèmes sont en communication permanente les unes avec les autres et avec l'environnement, grâce à une commande intégrée et adaptent leur performance aux conditions variables, comme la lumière du jour et le flux de personnes. Et la gestion du système peut être avec une Commande centralisée (filaire ou sans fil) ou une commande décentralisée dont nous en parlerons plus loin.

1.3.1 Domaines d'application

1.3.1.a L'éclairage extérieur intelligent

Le pilotage de l'éclairage public était uniquement géré par des systèmes centralisés d'enclenchement et de déclenchement des luminaires, fonctionnant grâce à des automates reliés à des horaires programmés ou des horloges astronomiques. Mais aujourd'hui grâce à la technologie LED (Diode Electroluminescente), le pilotage de l'éclairage à évoluer d'un fonctionnement on/off vers les systèmes de gestion dits intelligents. Par sa flexibilité, la technologie LED permet de prévoir un enclenchement instantané des luminaires [3].

Il est centré sur l'efficacité et le respect de l'environnement.



Figure 1. 6 : Eclairage extérieur [6]

1.3.1.b L'éclairage intérieur

Cet éclairage intelligent intervient dans la gestion technique des bâtiments (GTB).

La GTB est un système informatique généralement installé dans des grands bâtiments ou des installations industrielles afin de superviser l'ensemble des équipements qui y sont installés [4].

C'est un domaine qui remplit de nombreuses fonctions comme :

- Le confort (optimisation de l'éclairage, du chauffage, ouverture automatique de la porte du garage...)
- La gestion d'énergie (programmation, optimisation de la température de la maison pour économiser...)

- La sécurité (détections d'intrusions, fuites de gaz...)
- La communication (comme les commandes à distance ou l'émission de signaux destinés à l'utilisateur).

On retrouve l'éclairage intérieur dans :

- La gestion des infrastructures et bureaux



Figure 1. 7 : Gestion des infrastructures et bureaux [6]

- Le domaine de la santé :
- L'éclairage pour l'hôtellerie
- Le domaine des transports
- L'industrie
- Les commerces

1.3.2 Matériels utilisés pour l'éclairage intelligent

On peut classer le matériel mis en jeu pour la réalisation de l'éclairage intelligent selon deux types :

1.3.2.1 Les capteurs :

Une large gamme de capteur est utilisée dans l'éclairage intelligent, allant des capteurs les plus simples à d'autres à principe plus complexes. Dans cette gamme, on trouve :

- Les capteurs de lumière : Photorésistance (LDR), Photodiode...etc.
- Les détecteurs de présence et de mouvement : infra-rouge, ultrasonique.
- Les capteurs iHF (technologie à haute fréquence intelligente).

1.3.2.2 Les actionneurs :

L'opération principale visée étant l'éclairage, les lampes représentent les actionneurs de ce dernier, par exemples :

- Les luminaires
- Lampe Led
- Les lampadaires solaires

1.4 Système de commande centralisée et commande décentralisée

1.4.1 Système centralisée

Les équipements de ce système sont mis en réseau grâce à un système sans fil (radio, infrarouge, WIFI) ou par des liaisons filaires plus stables (Bus, Ethernet) et contrôler par une intelligence centralisée (modules embarqués, centrale programmable, interface micro-informatique) grâce à laquelle ils communiquent. Les téléphone portables, ordinateur de poche, télécommande ou les tablettes tactiles permettent de centraliser et de programmer les différents équipements localement ou à distance. Ainsi, d'une seule touche nous pouvons contrôler l'arrosage, l'éclairage, la sécurité, la température, les toiles de fenêtres, et de programmer ceux-ci en fonction d'ambiances et de scénarios bien précis [7].

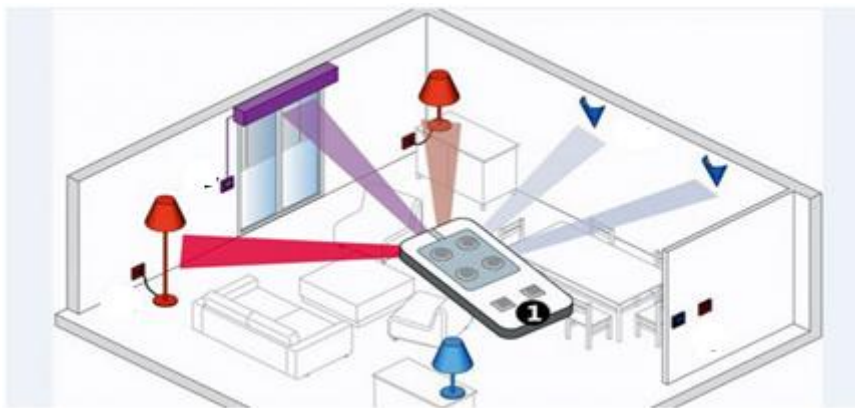


Figure 1. 8 : Système Centralisé [7]

1.4.2 Système décentralisée

Très avantageux pour les systèmes d'éclairage public, permettant de gagner en confort et d'économiser de l'énergie. Dans un système d'éclairage intelligent, où les lampadaires lumineux sont intelligents, l'éclairage se fera uniquement en présence d'utilisateur (en n'allumant les luminaires qu'autour d'un usager). Chaque luminaire contient un module intelligent et les modules sont équipés de capteurs. De plus les lampadaires ou luminaires, mis en réseau, communiquent entre eux de sorte que quand l'un détecte une présence il prévient les autres qui s'allumeront si nécessaire [8].



Figure 1. 9 : Système Décentralisé[8]

1.5 Conclusion

Dans ce chapitre, nous avons d'abord parlé de l'irrigation en générale, puis de différentes modalités d'arrosage. L'objectif étant d'automatiser la commande d'un système comportant l'arrosage et l'éclairage, un tour d'horizon sur le sujet est donc proposé dans ce chapitre.

Chapitre II : Partie matérielle et logicielle

2.1 Introduction

Ce second chapitre portera sur une étude théorique des principaux composants et logiciels utilisés pour la réalisation de notre système. Le système réalisé est un système d'arrosage et l'éclairage intelligent. Il doit commander et contrôler l'arrosage en fonction de la température, l'humidité du sol et la luminosité de la structure publique pour l'éclairage.

2.2 Cahier des Charges souhaité

Notre étude étant le cas d'une structure publique, le cahier des charges est défini ainsi :

- Stocker l'eau de la piscine dans un réservoir pour éventuel utilisation
- Recevoir un message à chaque fois qu'il y a un besoin d'arrosage,
- Déclencher l'arrosage depuis son smartphone,
- Remplir le réservoir d'arrosage après chaque arrosage
- Commander automatiquement l'éclairage.

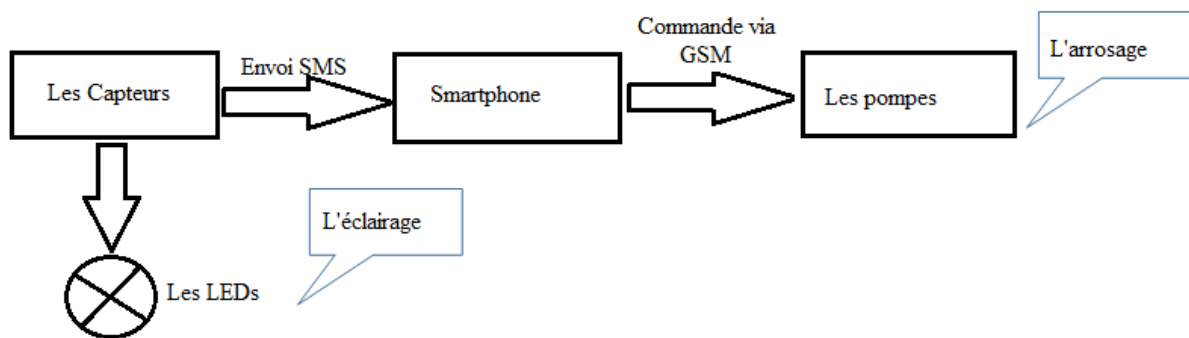


Figure 2. 1 : Présentation du cahier des charges

La réalisation de ce cahier de charges nécessite l'utilisation des modules comme le GSM, la carte Arduino UNO, des capteurs, d'autres composants et des logiciels dont nous ferons une étude plus loin.

A. Partie Matérielle

2.3 Arduino UNO

L'Arduino UNO est une carte électronique basée autour d'un microcontrôleur Atmega328P du fabricant Atmel. Il existe plusieurs autres types comme : le MEGA, DUO...etc. Le programme est stocké et exécuté par le microcontrôleur pour les différentes tâches.

2.3.1 Présentation de la carte Arduino UNO

L'intérêt principal des cartes ARDUINO est leur facilité de mise en œuvre. ARDUINO fournit un environnement de développement s'appuyant sur des outils

open source. Le chargement du programme dans la mémoire du microcontrôleur se fait de façon très simple par port USB. En outre, des bibliothèques de fonctions "clé en main" sont également fournies pour l'exploitation d'entrées sorties courantes : gestion des E/S TOR, gestion des convertisseurs ADC, génération de signaux PWM, exploitation de bus TWI/I2C, exploitation de servomoteurs... [10].

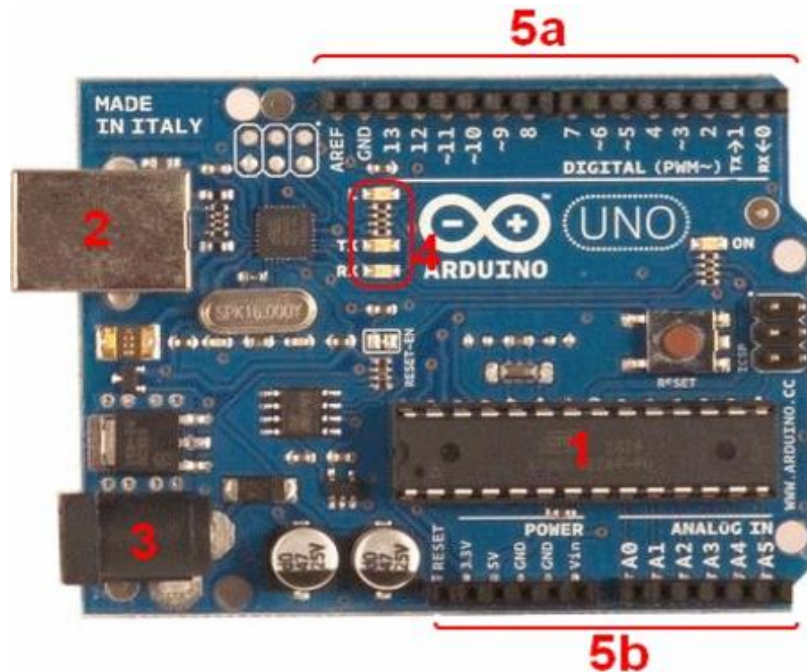


Figure 2. 2 : Présentation de la carte Arduino[10]

1 : le microcontrôleur

Le cerveau de la carte. C'est lui qui va recevoir le programme que nous allons créer et va le stocker dans sa mémoire puis l'exécuter. Grâce à ce programme, il va savoir faire des choses, dont nous avons besoins. Il est constitué par un ensemble d'éléments qui ont chacun une fonction bien déterminée :

Le processeur : c'est le composant principal du microcontrôleur. C'est lui qui va exécuter le programme que nous lui donnerons à traiter. On le nomme souvent le CPU.

La mémoire Flash : c'est celle qui contiendra le programme à exécuter. Cette mémoire est effaçable et réinscriptible (c'est la même qu'une clé USB par exemple).

RAM : c'est la mémoire dite "vive", elle va contenir les variables de notre programme. Elle est dite "volatile" car elle s'efface si on coupe l'alimentation du microcontrôleur (comme sur un ordinateur).

EEPROM : c'est le disque dur du microcontrôleur. On peut y enregistrer des infos qui ont besoin de survivre dans le temps, même si la carte doit être arrêtée. Cette

mémoire ne s'efface pas lorsque l'on éteint le microcontrôleur ou lorsqu'on le reprogramme.

Les registres : c'est un type de mémoire utilisé par le processeur.

La mémoire cache : c'est une mémoire qui fait la liaison entre les registres et la RAM.

2 et 3 : Alimentation

Pour fonctionner, la carte a besoin d'une alimentation. Le microcontrôleur fonctionnant sous 5V, la carte peut être alimentée en 5V par le port USB (en 2) ou bien par une alimentation externe (en 3) qui est comprise entre 7V et 12V. Cette tension doit être continue et peut par exemple être fournie par une pile 9V. Un régulateur se charge ensuite de réduire la tension à 5V pour le bon fonctionnement de la carte. Pas de danger de tout griller, il faut respecter l'intervalle de 7V à 15V.

4 : Visualisation

Les trois "points blancs" entourés en rouge (4) sont des LED dont la taille est de l'ordre du millimètre. Ces LED servent à deux choses :

Celle tout en haut du cadre : elle est connectée à une broche du microcontrôleur et va servir pour tester le matériel. Nota : Quand on branche la carte au PC, elle clignote quelques secondes.

Les deux LED du bas du cadre : servent à visualiser l'activité sur la voie série (une pour l'émission et l'autre pour la réception). Le téléchargement du programme dans le microcontrôleur se faisant par cette voie, on peut les voir clignoter lors du chargement.

5a et 5b : la connectique

La carte Arduino ne possédant pas de composants qui peuvent être utilisés pour un programme, mis à part la LED connectée à la broche 13 du microcontrôleur, il est nécessaire de les rajouter. Mais pour ce faire, il faut les connecter à la carte. C'est là qu'intervient la connectique de la carte (en 5a et 5b).

2.4 L'Arduino GSM Shield 2

C'est ce module qui nous permettra de contrôler et commander depuis notre smartphone le système partout où il y'a un réseau GSM. L'utilisation du GSM shield 2 est plus pratique avec son antenne intégrée et sa compatibilité avec Arduino UNO.



Figure 2. 3 : Arduino GSM Shield 2 [9]

2.4.1 Description de l'Arduino Shield 2

Il existe plusieurs modules GSM : SIM900, SIM800, Shield V2 ...etc.

L'Arduino GSM Shield V2 permet à une carte Arduino de se connecter à Internet, effectuer / recevoir des appels vocaux et envoyer / recevoir des messages SMS. Le Shield utilise un modem radio M10 de Quectel . Il est possible de communiquer avec le shield en utilisant des commandes AT qui sont des jeux de commande textuelle. La bibliothèque GSM dispose d'un grand nombre de méthodes de communication avec le Shield.

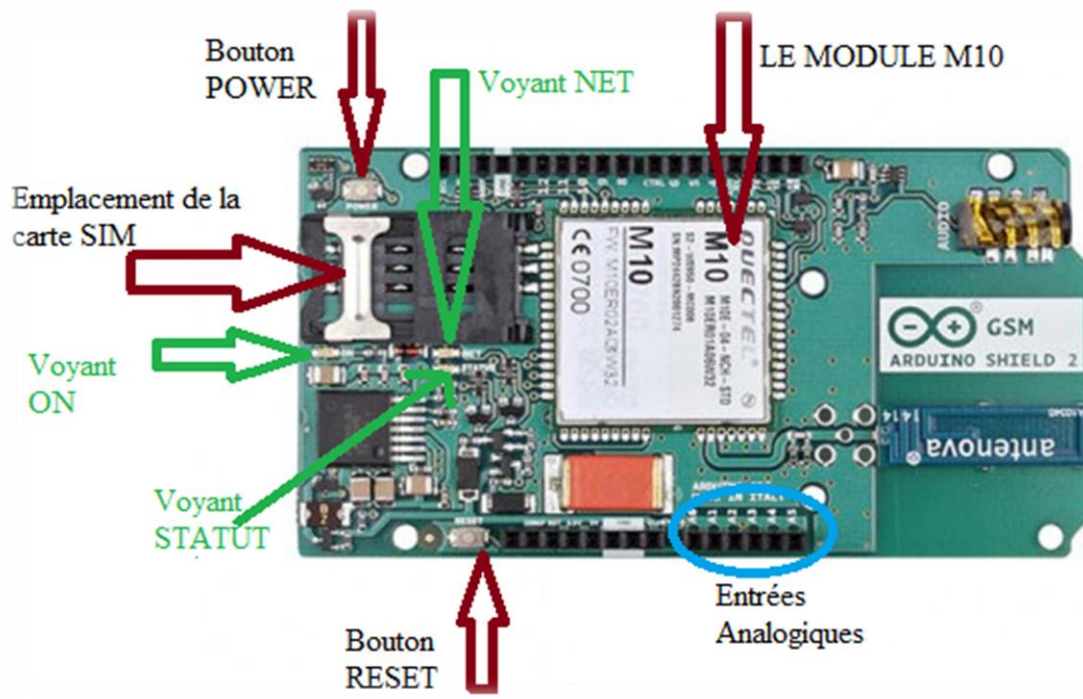


Figure 2. 4 : Vue d'ensemble GSM Shield 2

Le Shield contient :

- ✓ 1 Antenne intégrée.

- ✓ 6 Broches d'entrées analogiques : pour brancher des sondes analogiques de mesures et y programmer des seuils minimum et maximum **et en cas de franchissement des seuils on reçoit un SMS.**
- ✓ 14 Broches numériques : Entrées/Sorties.
Entrées digitales ou numériques : pour brancher des détecteurs du type : Intrusion, présence, contact de porte, détecteur de fumée, Gaz, inondation, thermostat, Hygrométrie, ...etc.
 Les sorties sont activées en envoyant au modem GSM un SMS codé, un relais de puissance est alors activé pour commander tous matériels (équipements électriques) :
 Portes, portails, chauffages, climatisation, volets, moteurs, pompes, ...etc.
- ✓ Deux petits boutons : **RESET** (est lié à la broche de réinitialisation Arduino) et **POWER** (allume et éteint le modem).
- ✓ Prise audio intégrée.
- ✓ Des indicateurs embarqués (voyants d'état) : On, Statut et Net
On : indique que le bouclier est alimenté.
Statut : s'allume lorsque le modem est sous tension et que des données sont transférées vers / depuis le réseau GSM / GPRS.
Net : clignote lorsque le modem communique avec le réseau radio.
- ✓ 5 Broches d'alimentation. [9]

2.4.2 Caractéristiques :

Le M10 est un modem quadri-bande GSM / GPRS fonctionnant :

Fréquences	GSM 850 MHz, GSM 900 MHz, DCS1800 MHz et PCS1900 MHz.
Protocoles	TCP / UDP et HTTP via une connexion GPRS
La vitesse maximale de transfert des données sur la liaison descendante et sur la liaison montante GPRS	85,6 kbps
Tension de fonctionnement	5V
Puissance	Il est recommandé d'alimenter la carte avec une alimentation externe pouvant fournir entre 700 et 1000 mA
Dimension	100 x 53 mm

Tableau 2. 1 : Caractéristique du GSM Shield 2 Arduino [9]

2.4.3 Brochage :

La broche 2 est connectée à la broche TX du M10

La broche 3 à la broche RX.

La broche PWRKEY du modem est connectée à la broche 7 d'Arduino.

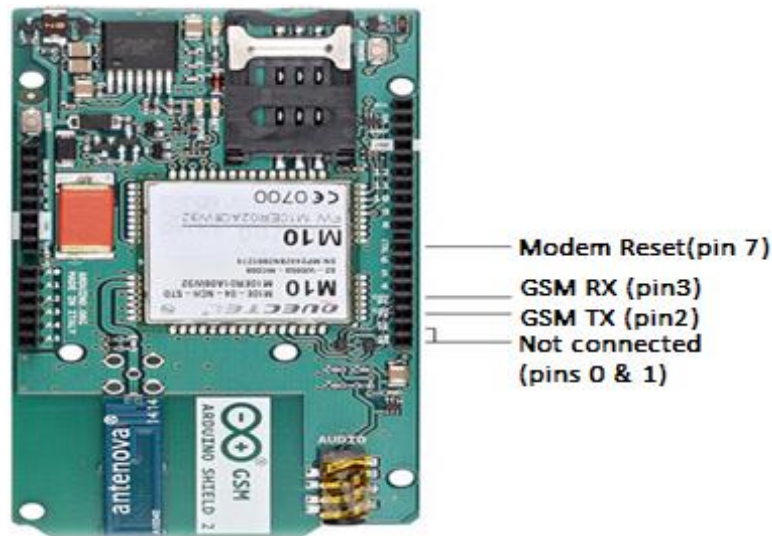


Figure 2. 5 : Brochage[9]

2.5 La photorésistance LDR

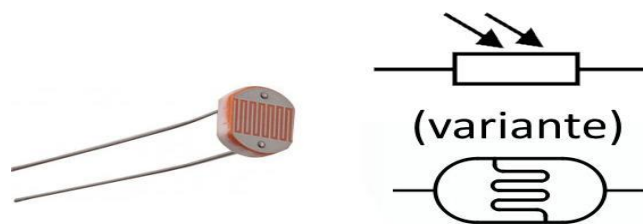


Figure 2. 6 : La photorésistance(LDR)[11]

Notre objectif est de pouvoir mesurer la luminosité jour/nuit pour pouvoir éclairer par la suite la structure la nuit dans l'obscurité. Donc la présence d'un capteur de lumière dans notre circuit est plus qu'évidente.

La photorésistance est l'un des capteurs de lumière dont la résistivité dépend de la luminosité ambiante.

La partie haute du capteur en forme de serpent est sa partie sensible qui est un sulfure de cadmium. L'énergie lumineuse déclenche l'augmentation des porteurs libres dans le matériau de la LDR de sorte que sa résistance électrique diminue [5]. Donc plus elle est éclairée plus sa résistivité baisse.

Notre choix s'est porté sur la LDR parce qu'elle a l'avantage d'avoir des larges gammes spectrales, une sensibilité élevée, facile à trouver et d'être moins chers.

Les capteurs de température de la série LM35 sont des circuits intégrés de précision. Le capteur délivre entre ses bornes une tension évoluant linéairement avec la température. La température est directement en degré Celsius. Et la tension de sortie est divisée par 10mV.

Nous avons les différents types :

LM35 et LM35A ont une plage de -55°C à 150°C ,

LM35C et LM35CA délivrent des températures de -40°C à 110°C ,

LM35D pour une plage de 0°C à 100°C . [12]

Notre choix s'est porté sur le LMDZ parce que nous avons un Arduino qui ne supportera une plage de $-55/150^{\circ}\text{C}$ et aussi en fonction de sa précision, de son coût moins cher et de la facilité de son utilisation.

2.8 Détecteur de niveau d'eau



Figure 2. 9 : Détecteur de niveau d'eau[14]

Il peut être utilisé pour détecter le niveau de liquide dans un réservoir, peut activer une pompe, un indicateur, une alarme ou un autre appareil. Ces caractéristiques sont :

- Capacité de contact : 70W
- Tension de commutation : DC110V
- Courant de commutation : 0.5A
- Résistance d'isolement : > 10
- Température de fonctionnement : $-10 \sim 60$
- Pression de travail : $< 0.6\text{MPA}$ [13]

Nous allons l'utiliser pour détecter le niveau d'eau de notre réservoir qui stocke l'eau servant à faire l'arrosage de notre jardin.

2.9 La LED

DEL en français (diode électroluminescente) est un composant optoélectronique qui émet de la lumière lorsqu'elle est parcourue par un courant électrique dans le sens direct. C'est-à-dire le courant doit la traversée de l'anode (la tension plus élevée) vers la cathode.

Elle a une vaste gamme de couleurs en fonction du semi-conducteur (GaAsp, AlGaAS, GaN, GaP, InGaN, SiC, ...) utilisé et de la longueur d'onde (infrarouge si elle est >700nm, ultraviolet <400nm et les autres couleurs entre les deux). La tension de seuil d'une LED dépend de sa couleur (pour le jaune est compris entre 2.10V et 2.18V) [6].

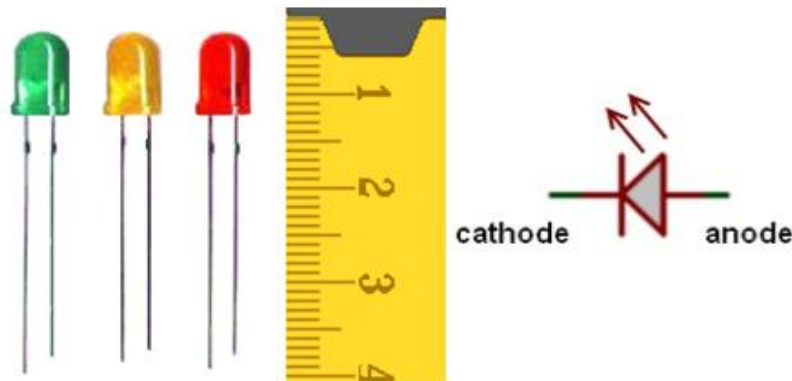


Figure 2. 10 : Image LED

2.10 Micro pompe à eau avec connecteur USB



Figure 2. 11 : Micro pompe à eau[15]

L'utilisation d'une pompe est nécessaire dans notre projet, une pompe qui sera équipé de tuyaux afin de faire circuler l'eau. Nous avons opté pour une micro pompe (voir figure 1.10) dont le débit est réglable et conçu entièrement submersible pour l'eau douce et eau salée.

Elle est en plastique ayant comme caractéristique :

- Une tension d'entrée DC 3.5-9v ;
- La puissance de 1-3 w ;
- Débit : Max. 200L/H (44GPH) ; [14]

2.11 2channels 5v Relay Module

Nous allons utiliser un module de relais pour contrôler la mise en service de nos pompes.



Figure 2. 12 : Module à 2canaux relais [16]

Le 2channels 5v Relay Module C'est un module 5V 2 canaux relais, il peut être commandé directement par une large gamme de microcontrôleurs tels que Arduino, AVR, PIC, ARM et MSP430. Le module contient 2 relais, avec des ports "NC" signifie "Normalement connectés à COM" et "NO" ports signifie "Normalement ouvert à COM". Ce module est également équipé Avec des LED pour indiquer l'état des relais.

B. Partie Logicielle

2.12 Logiciel Arduino

Le logiciel Arduino est un espace de développement intégré (IDE) qui nous permet d'écrire, de compiler et d'envoyer du code sur le circuit imprimé du même nom (carte Arduino). Son interface se présente de manière simple et structurée.

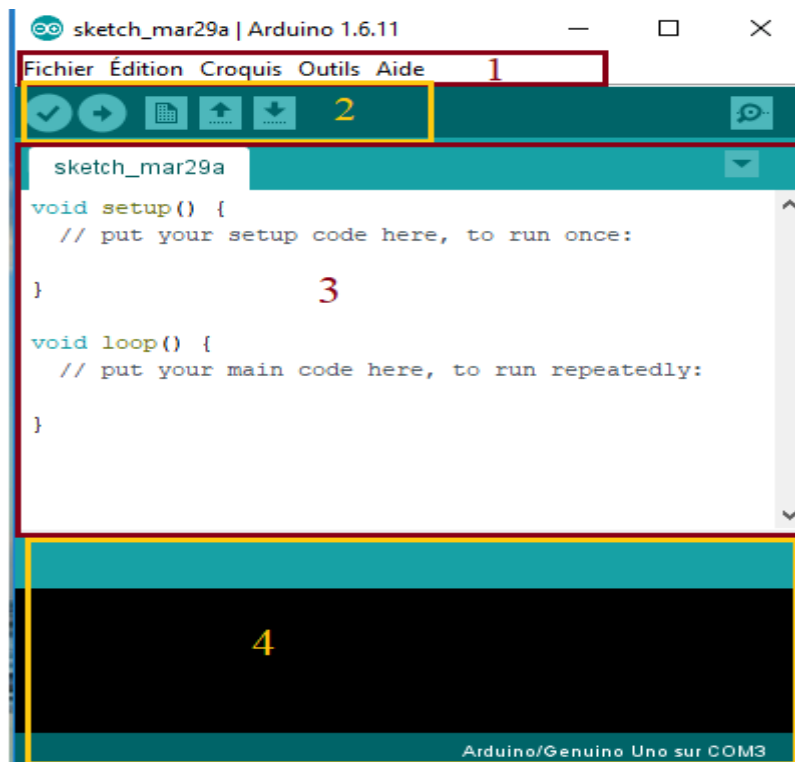


Figure 2. 13 : Interface d'Arduino

- 1 ⇒ Ce sont les options de configuration du logiciel
- 2 ⇒ Les boutons qui vont nous servir lorsque l'on va programmer la carte
- 3 ⇒ Ce bloc va contenir le programme que nous allons créer
- 4 ⇒ Il est important, car il va nous aider à corriger les fautes dans notre programme. C'est le débogueur

Nous allons parler des fonctions des différents boutons du logiciel Arduino que nous avons encadrés et nommés avec des chiffres de 1 à 6.

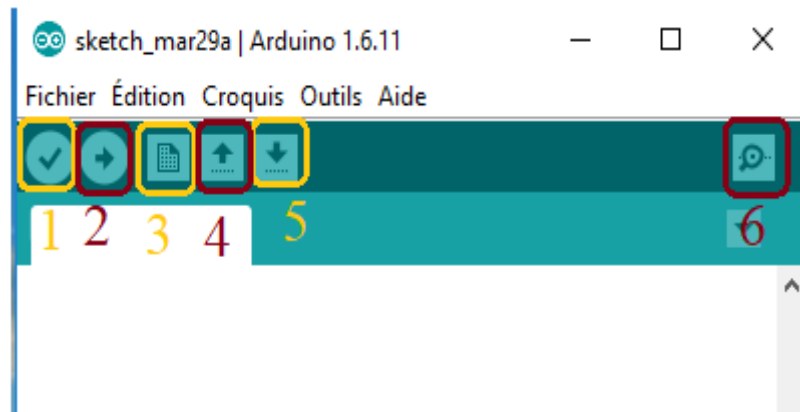


Figure 2. 14 : Les boutons de l'interface Arduino

- 1 ⇒ Permet de vérifier le programme, il actionne un module qui cherche les erreurs dans votre programme.
- 2 ⇒ Charge (téléverse) le programme dans la carte
- 3 ⇒ Crée un nouveau fichier
- 4 ⇒ Ouvre un fichier Bouton
- 5 ⇒ Enregistre le fichier
- 6 ⇒ Ouvre le moniteur série

2.13 Les commandes AT

Les commandes AT sont des jeux de commandes textuelles permettant de gérer la plupart des modems ou des modules GSM. Ces commandes commencent toujours par les lettres « AT » et se terminent obligatoirement par un retour chariot.

AT : Code d'attention. C'est le préfixe d'une ligne de commande qui indique au modem qu'une commande ou une séquence de commandes va être envoyée.

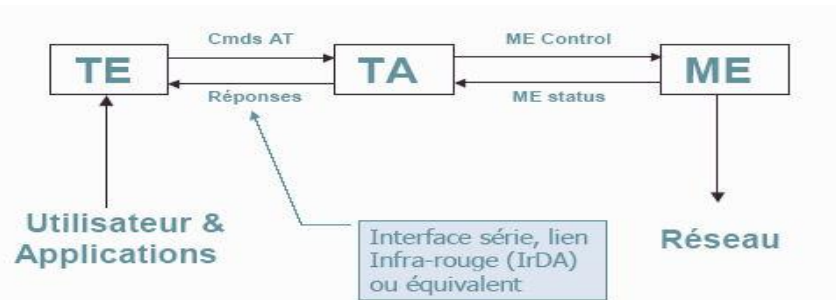


Figure 2. 15 : Schéma de fonctionnement des commandes AT[17]

- ❖ TE (Terminal Equipment) : envoi et affiche les Commandes
- ❖ ME (Mobile Equipement)
- ❖ TA (Terminal Adaptator) : interface entre l'utilisateur et le mobile

2.11.1 Les types des commandes AT et les réponses

➤ **Commande de test: AT+<X>=?**

Pour la commande test, le ME renvoie la liste des paramètres et des plages de valeurs définies avec la commande d'écriture correspondante par processus internes.

➤ **Commande pour lire : AT+<X> ?**

Elle renvoie la valeur actuellement définie du paramètre ou des paramètres.

➤ **Commande pour écrire : AT+<X>=<...>**

Cette commande définit les valeurs de paramètre défini par l'utilisateur.

➤ **Commande d'exécution : AT+<X>**

La commande d'exécution lit les paramètres non variables affectés par les processus internes dans le GSM. [15]

2.11.2 Commandes AT dédiées au service SMS :

AT+CSMS	Sélection du service de messagerie
AT+CPMS	Sélection de la zone mémoire pour le stockage des SMS
AT+CMGF	Sélection du format du SMS (PDU ou TEXT)
AT+CSCA	Définition de l'adresse du centre de messagerie
AT+CSDH	Affiche en mode TEXT le paramétrage des SMS
AT+CSAS	Sauvegarde du paramétrage
AT+CRES	Restauration du paramétrage par défaut
AT+CNMI	Indication concernant un nouveau SMS
AT+CMGL	Liste les SMS stockés en mémoire
AT+CMGR	Lecture d'un SMS
AT+CMGS	Envoie un SMS
AT+CMSS	Envoie d'un SMS stocké en mémoire
AT+CMGW	Écriture d'un SMS
AT+CMGD	Efface un SMS

Tableau 2. 2 : Commande AT dédiées service SMS [15]

2.14 Conclusion

Ce chapitre nous a permis de connaître les notions de bases et les caractéristiques des composants et modules que nous utilisons dans le projet. De comprendre leurs modes de fonctionnement ainsi que le logiciel utilisé.

Chapitre III : réalisation pratique du système

3.1 Introduction :

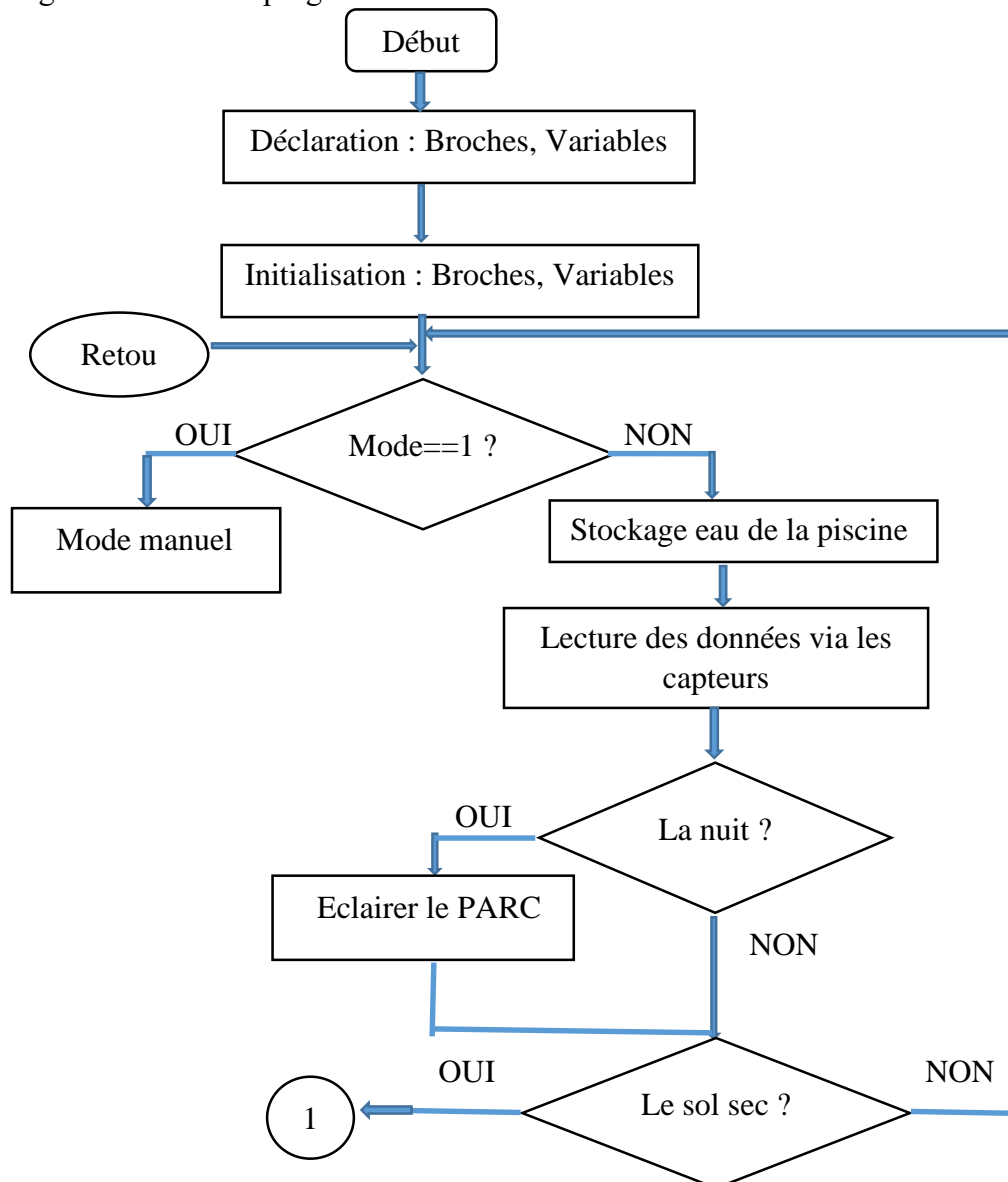
Après avoir fait, précédemment une description théorique sur les Dispositifs que nous allons utiliser, nous proposons dans ce chapitre, la réalisation pratique du système d'éclairage et d'irrigation intelligente selon un cahier des charges que nous avons mis au point.

L'application de ce contrôle commande est d'effectuer sur une structure publique qui compte un certains nombres de services offerts au seul objectif qui est le confort du citoyen (visiteur du lieu).

Deux opérations font alors l'objectif de ce travail : commander automatiquement l'éclairage de la structure publique selon le jour et la nuit, contrôler et commander à distance via GSM, l'arrosage des plantes du jardin d'une structure publique avec la méthode goutte-à-goutte.

3.2 Organigramme du Programme

L'organigramme de notre programme est le suivant :



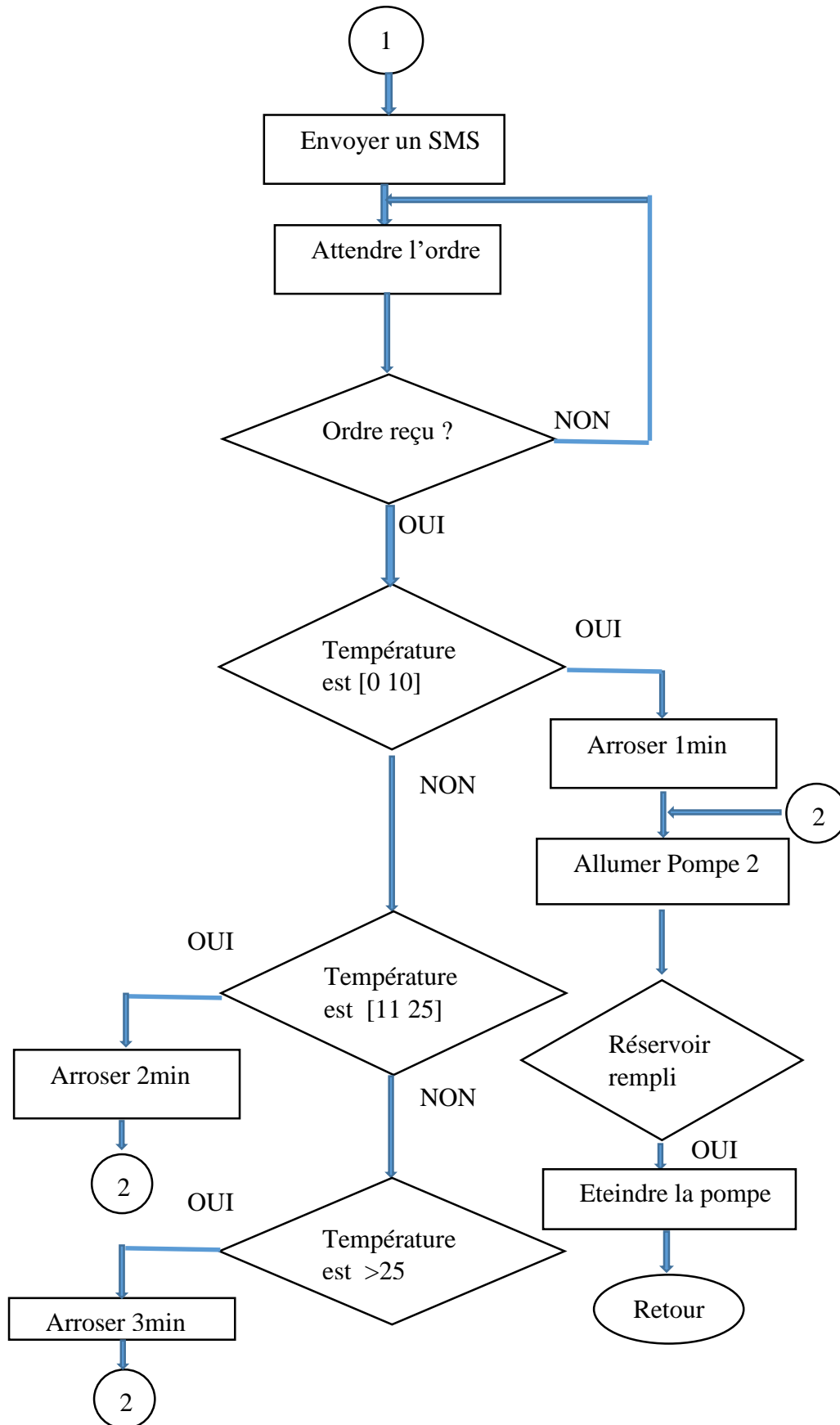


Figure 3. 1 : L'organigramme du programme

3.3 Schéma électrique et principe de fonctionnement du système

3.3.1 Schéma électrique

Le schéma électrique nous permettra de bien expliquer le principe de fonctionnement du système.

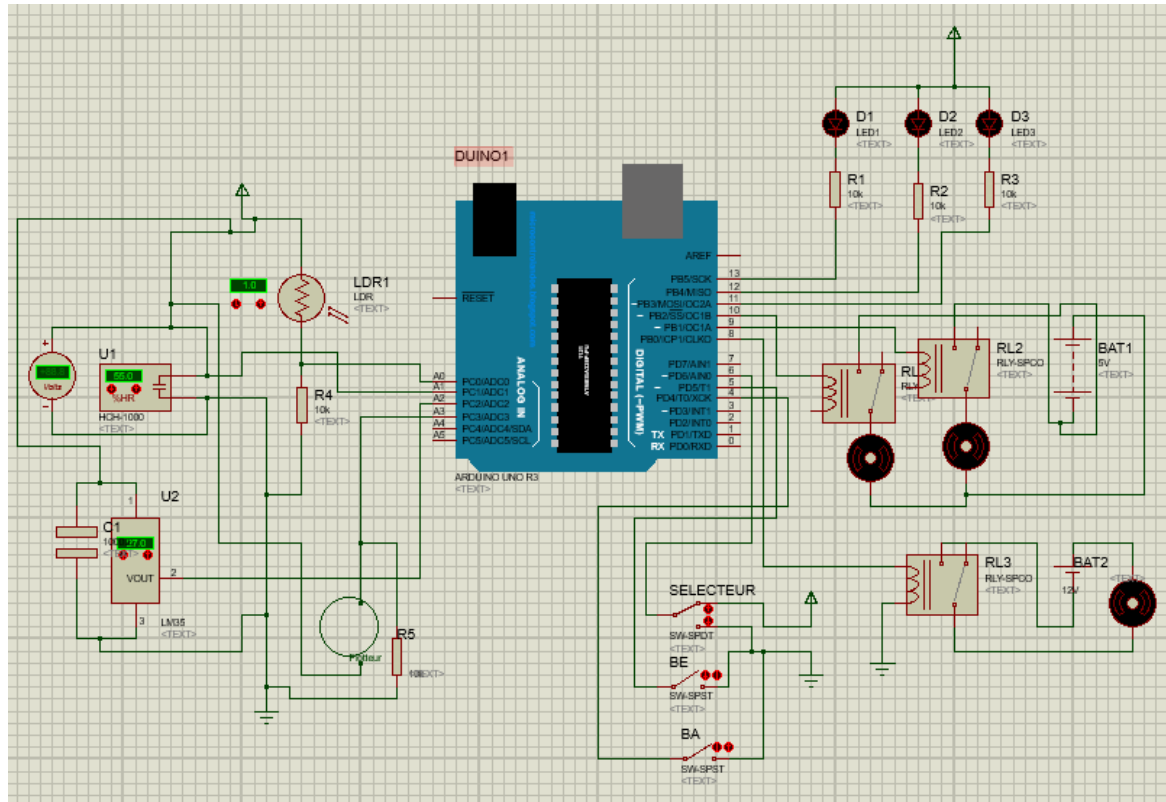


Figure 3. 2 : Schéma électrique du système

3.3.2 Principe de fonctionnement du système

Pour le contrôle de l'éclairage, l'Arduino lit la tension aux bornes du LDR à travers une entrée analogique et la compare à la valeur du seuil. Le seuil étant la limite à laquelle nous passons de la lumière à l'obscurité. Lorsque le seuil est atteint led1, led2 et led3 s'allument automatiquement et restent à cet état jusqu'à dépassement du seuil.

Pour la télé-irrigation, en un premier temps l'eau de la piscine est stockée dans le premier réservoir grâce à la pompe de vidange, PV (voir figure 3.11). Le stockage d'eau se fait de façon périodique.

L'Arduino lit en permanence, la valeur du capteur d'humidité. Si le sol est sec, il nous envoie un SMS nous prévenant qu'il y'a besoin d'arrosage via le module GSM. La confirmation de l'opération d'arrosage est exécutée par un retour de SMS (dans notre cas : l'opérateur chargé de prendre la décision d'arrosage doit confirmer par un message : POMPEon).

La pompe d'arrosage PA (voir figure 3.11) se met en marche et l'arrosage commence. Il est à signaler que la durée d'arrosage est un paramètre qui change en fonction de la température

délivrée par le capteur de température (l'impact est très perceptible d'une saison à l'autre). La durée d'arrosage dépend de la valeur que l'Arduino lit aux bornes du capteur de température

Dès que l'arrosage prend fin le relai contrôlant la pompe de remplissage PR (voir figure 3.11) s'active mettant en marche cette dernière.

Cette dernière s'arrêtera si le niveau haut du réservoir 1 est atteint, cela se fait grâce à la valeur qui correspond à un niveau haut lu par Arduino aux bornes du capteur de niveau.

Comme pour tout système automatisé l'intégration d'une commande manuelle est plus que nécessaire pour la sécurité et la maintenance. Raison pour laquelle nous avons intégré une commande manuelle à notre système.

Un sélecteur simule le mode automatique ainsi que le mode manuel.

En mode manuel l'éclairage et l'arrosage se déclenche respectivement suite à l'appuie sur BE (Bouton Eclairage) et BA (Bouton Arrosage).

3.4 Commande de l'éclairage

Pour cette partie nous utilisons une LDR et trois (3) LED.

La LDR est connecté à l'une des entrées analogiques de l'Arduino. La fonction `analogRead()` retourne un nombre entier (Int) compris entre 0 et 1023 selon la luminosité . Ce nombre correspondant à la tension mesurée, 0 = 0 volt, 1023 = 5 volts (pour la carte Arduino UNO). Pour avoir la valeur de la tension :

$$\text{Tension} = \text{valeur} * (5.0 / 1023.0).$$

Nous avons visualisé sur le moniteur série de l'Arduino à partir de la fonction `analogRead()`, des valeurs entières dans différentes conditions (pleine lumière, Eclairé et obscurité) suite à des tests sur le LDR. Les valeurs obtenues sont les suivantes :

Situation	Valeur LDR
Plein lumière	974-1023
Eclairé	100-974
Obscurité	0-100

Tableau 3. 1 : Valeurs du capteur de lumière dans différents conditions

Les LED sont connectées respectivement aux pins 13,12 et 11 de l'Arduino comme des sorties digitales. En utilisant une résistance d'adaptation entre chaque LED et les pins de l'Arduino. La tension maximale directe (fonctionnement directe de la LED) se trouvant entre 1.2v et 1.6v et une intensité maximale $I=20\text{mA}$ d'après la fiche technique de la LED. En utilisant une alimentation de 5v, nous avons calculé la valeur de la résistance grâce à la formule $U= RI$

$$R = \frac{U}{I} = \frac{5 - 1.2}{0.02} = 190\Omega$$

Pour être sûr de ne pas détériorer nos LED à cause d'un courant trop fort, nous avons placé une résistance dont la valeur est plus grande que celle calculée de valeur $1k\Omega$.

Nous avons relié la plus grande patte (anode) au 5v, et la plus petite à la résistance, elle-même reliée à la broche de la carte. La LED s'allumera lorsque qu'on met la broche à la masse.

3.4.1 Simulation sur Proteus ISIS

Sur Proteus ISIS nous avons fait la simulation de la partie éclairage en mode automatique et manuel.

On a utilisé une TORCH_LDR pour simuler la lumière naturelle. Lorsque la torche est proche de la LDR, les LED sont éteintes et à une certaine distance elles s'allument automatiquement.

Sur la figure ci-dessous la torche est proche, supposant ainsi la présence de la lumière. Nos LED sont alors éteintes.

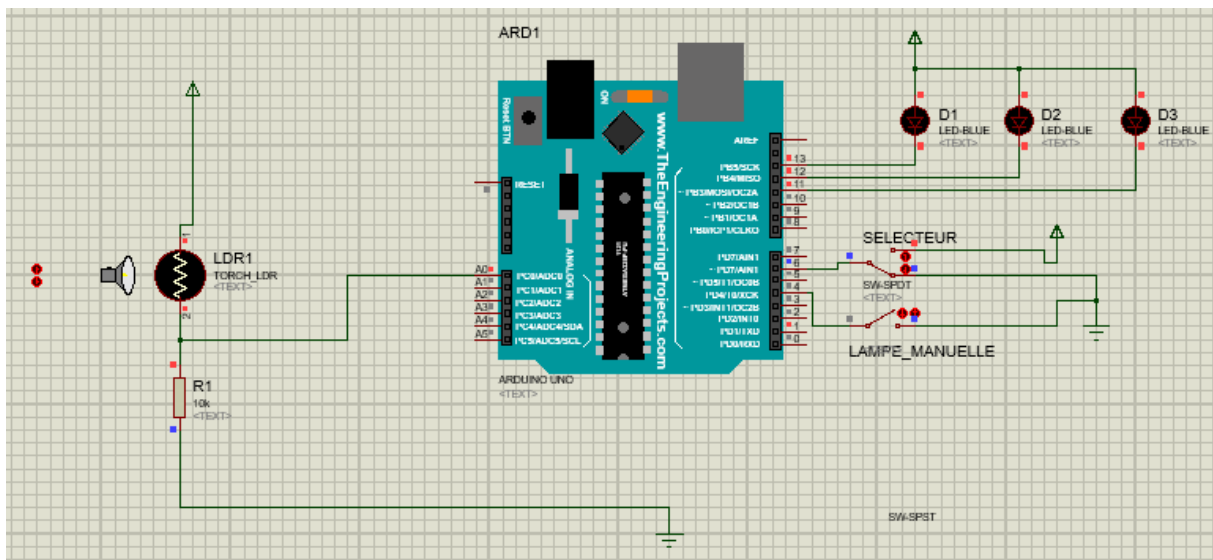


Figure 3. 3 : Simulation ISIS Allumage automatique LED éteintes

La figure suivante montre le phénomène contraire du précédent. La torche est loin du LDR et les sont LED allumés.

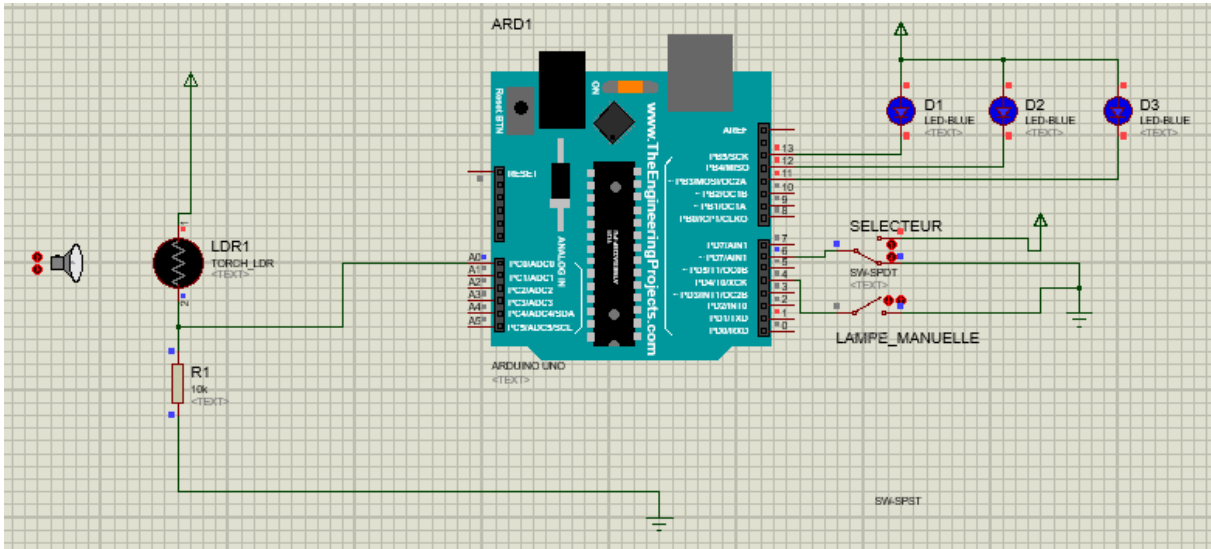


Figure 3. 4 : Simulation ISIS Allumage automatique LED allumées

En mon manuel, les ED s'allument lorsque l'interrupteur (LAMPE_Manuelle) est en état haut et s'éteignent dans le cas contraire. Comme on peut voir sur la figure ci-dessous

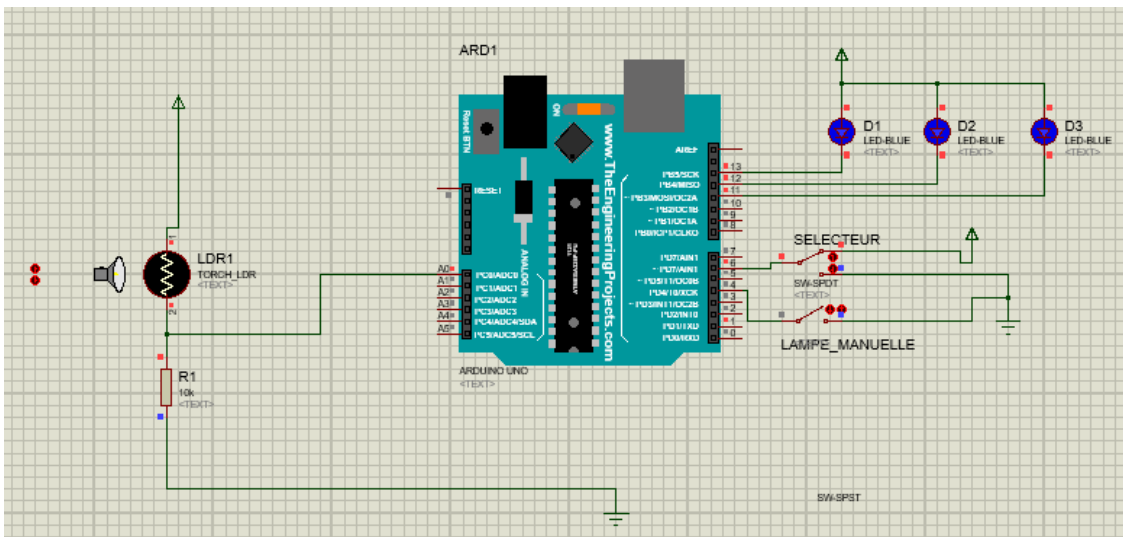


Figure 3. 5 : allumage manuel sous ISIS

3.5 Commande de l'arrosage

C'est ici qu'intervient notre module GSM pour contrôler et commander notre système d'arrosage par SMS (envoi/réception) partout où il y'a un réseau GSM peu importe la distance.

3.5.1 Mise en marche du module GSM Shield 2

1. On insère une carte SIM active dans le support prévu à cet effet.



Figure 3. 6 : Insertion de la carte SIM

2. On connecte le shield avec notre carte Arduino Uno en liant les pins du GSM shield 2 avec ceux de l'Arduino.

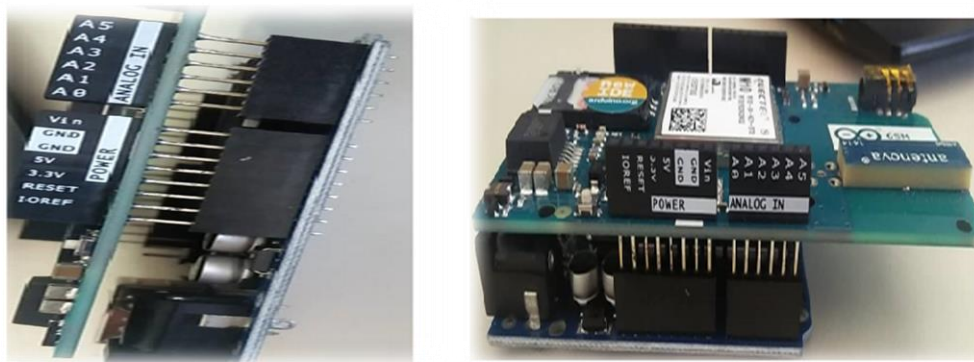


Figure 3. 7 : Connexion du Shield avec un Arduino UNO

3. Ensuite on alimente la carte Arduino. La DEL « ON » doit s'allumer.
4. Puis il faut appuyer sur le bouton POWER pour allumer le Shield.



Figure 3. 8 : Shield sous tension

Pour finir nous avons établi la communication logicielle entre le module GSM Shield 2 et l'Arduino Uno grâce aux broches numériques 2 qui équivaut à Tx et 3 le Rx et à la fonction :

SoftwareSerial gsm(2, 3) dans le programme. Et pour communiquer avec le GSM Shield 2 nous avons utilisé les commandes AT.

3.5.2 Bloc capteur d'humidité

Notre capteur est placé directement au sol pour détecter sa teneur en humidité. Son pin S est relié à une entrée analogique de l'Arduino, le G à GND et le V au 5v. Comme il fonctionne en numérique et analogique nous l'avons réglé en mode analogique en basculant le débit vers le A du module (voir Figure 3.8).

Après plusieurs tests sur notre capteur dans différents conditions (sol sec, peu humide, humide, inonder), en visualisant les valeurs obtenues grâce à la fonction analogRead() sur le moniteur série du logiciel Arduino, nous avons obtenu le tableau suivant :

Conditions	Valeurs brutes	Valeurs convertie
Sol inondé	122-273	0.5 - 1.33
Sol humide	288-774	1.4 - 3.8
Sol très peu humide	775-915	3.8 - 4.47
Sol Sec	915-1023	4.47 - 5

Tableau 3. 2 : Valeurs du capteur d'humidité dans différents conditions

Nous avons défini notre seuil en nous référant sur le tableau ci-dessus. Lorsque le seuil est atteint, un message sera envoyé pour nous avertir de l'état du sol, autrement dit que nos plantes ont besoin d'arrosage.

Nous commençons d'abord par sélectionner le format de notre message grâce à la commande AT+CMPF=1, mode Texte.

Ensuite AT+CMGS=\ "numéro du destinataire"\ commande qui permet l'envoi des messages terminés par Ctrl+Z qui est égale 0x1A et correspond à 26.

Ensuite on écrit le contenu du message grâce à gsm.println(" contenu du message") en terminant par Ctrl+Z gsm.println((char), 26) ;

Le message est directement envoyé après le délai qui suit le Ctrl+Z.



Figure 3. 9 : Brochage du capteur d'humidité

3.5.3 Réception SMS via GSM

Après avoir reçu un message nous prévenant d'un besoin d'arrosage, nous déclenchons l'arrosage par envoi d'un SMS au module.

La réception, la lecture et l'identification du code se font en plusieurs étapes qui sont détaillés dans notre programme.

Dans cette partie les commandes AT utilisées sont :

« AT+CNUM », « AT+CSQ », « AT+CMGF », « AT+CMGD », « AT+CMDR » dont nous avons précisé dans le chapitre précédent.

3.5.4 Bloc capteur de température

La température intervient au niveau de la durée de l'arrosage au cas où celui-ci est déclenché.

Lorsqu'il fait chaud, le soleil brille fort les plantes ont besoins de beaucoup d'eau. Pour cause l'eau sèche vite qui est le cas contraire au temps de fraîcheurs. Alors pour anticiper ce phénomène, plus la température est élevée plus la durée de l'arrosage est longue et plus elle est basse plus la durée est courte. On pourra voir des exemples sur l'organigramme (voir figure 3.1)

La patte du milieu (OUT) du LM35Dz est reliée à l'une des entrées analogiques de notre module, la patte droite au +5v et celle du gauche à GND selon la fiche technique du composant (voir figure 3.9).

La température en degré Celsius est : $Temp * 5.0 / 1023 * 0.01 = Temp * 0.488758553$

Temp : valeur de la sortie du capteur.

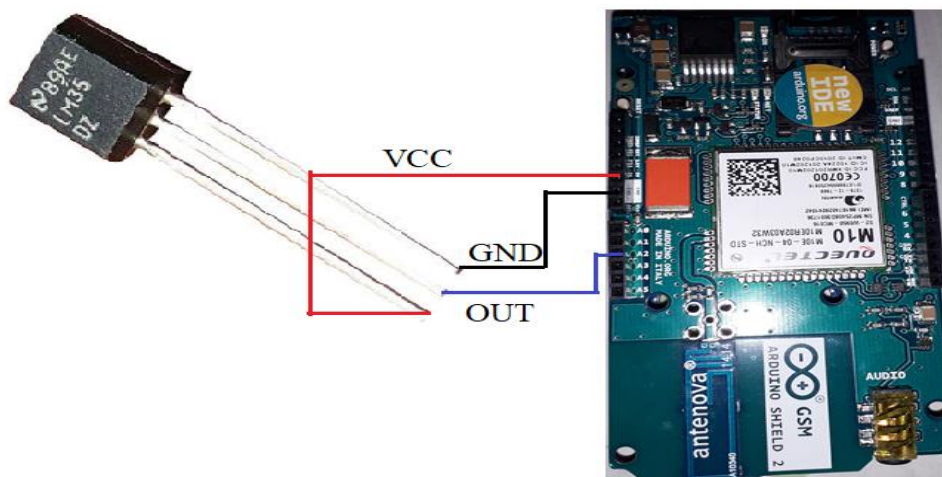


Figure 3. 10 : Brochage capteur de température

Après avoir testé notre capteur, nous avons puis visualiser la valeur de la température sur le moniteur du logiciel Arduino. On le voir sur la figure suivante :

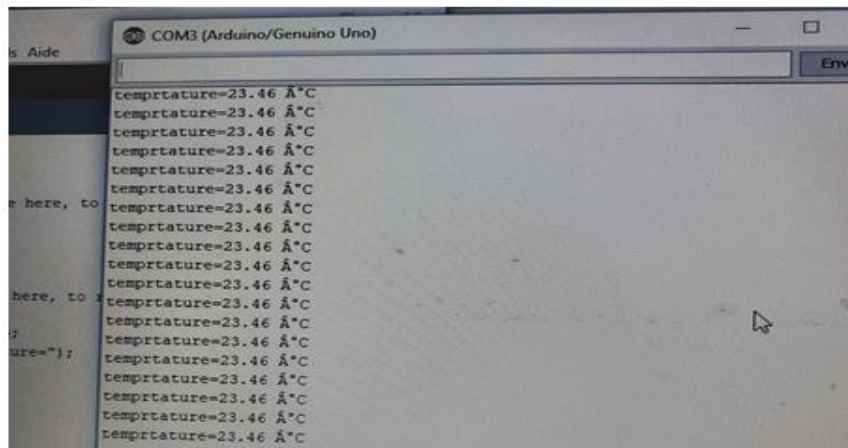


Figure 3. 11 : Affichage de la température sur le moniteur de l'Arduino

3.5.5 Bloc des pompes et le capteur de niveau d'eau

Ici nous avons trois (3) pompes nommées respectivement PA, PR et PV (voir Figure 3.11) pour chacun de nos réservoirs R_A et R_R et la piscine, un capteur de niveau d'eau, et un relai double chaînes pour commander nos deux pompes. En effet R_A est notre réservoir pour l'arrosage et R_R celui qui servira pour remplir R_A (l'eau de R_R venant de la piscine).

PV est commandé par un relai, qui est parcouru par un courant à chaque fois que c'est le moment de la vidange.

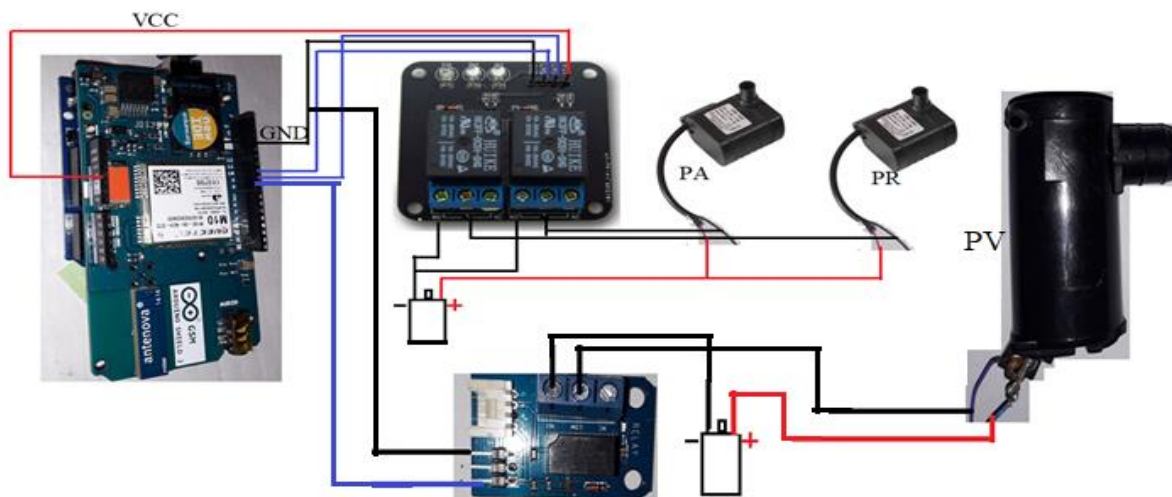


Figure 3. 12 : Brochage des pompes

A la réception du code CH1 est alimenté et PA mise en marche pour une durée qui dépend de la température du lieu. A la fin de l'arrosage CH1 est non alimenté et CH2 s'alimente mettant en marche PR pour remplir le réservoir R_A et ne s'arrêtera que si le niveau haut est atteint.

Et c'est là qu'intervient notre flotteur Ne. Ce dernier relié à une entrée analogique de l'Arduino (voir Figure 3.12). Il est placé est l'intérieur du réservoir R_A, il nous délivre une valeur entre 0 et 5V. Nous avons fixé des seuils comme étant l'état bas (0) pour les valeurs inférieures à 5 et l'état haut (1) égale au 5V.

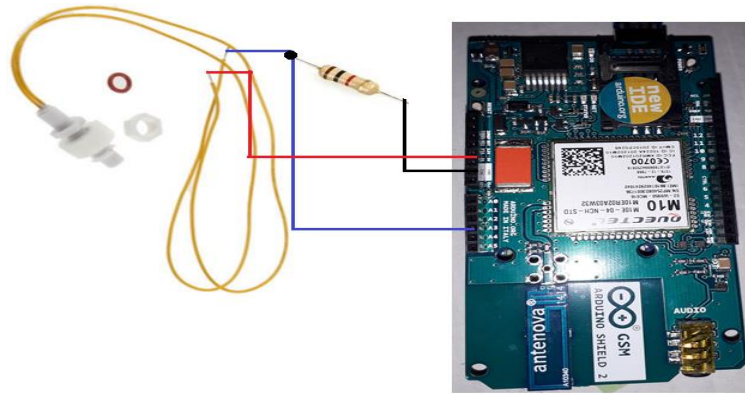


Figure 3. 13 : Brochage Capteur de niveau

Après la sélection du mode manuel par le sélecteur, l'appuie sur BA permet de lancer l'arrosage et un appuie dans l'autre sens permet d'y mettre fin.

3.6 Système sur plaque d'essai

Après des simulations sur Proteus ISIS, nous avons fait les essais sur une plaque d'essai pour tester le fonctionnement de notre système. Et nous avons constaté que notre système fonctionne bien pour l'éclairage de même que pour l'arrosage. Seulement que nous n'avons pas utilisé de tuyaux entre les réservoirs et pour les plantes.

Les images suivantes sont respectivement celle du circuit d'éclairage et celle de l'arrosage.

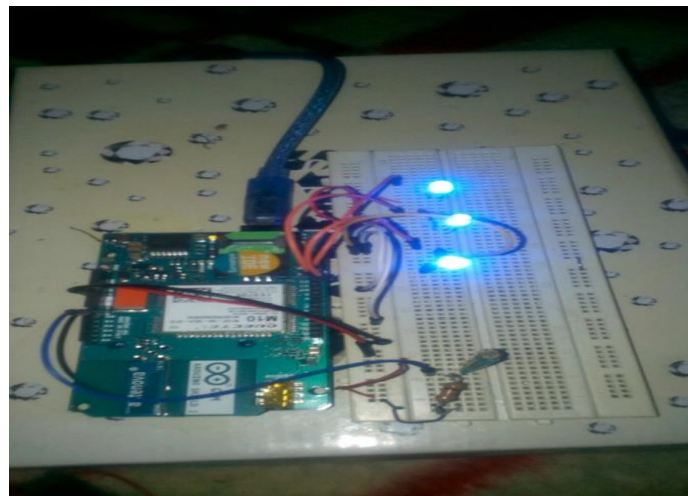


Figure 3. 14 : Bloc d'éclairage sur plaque d'essai

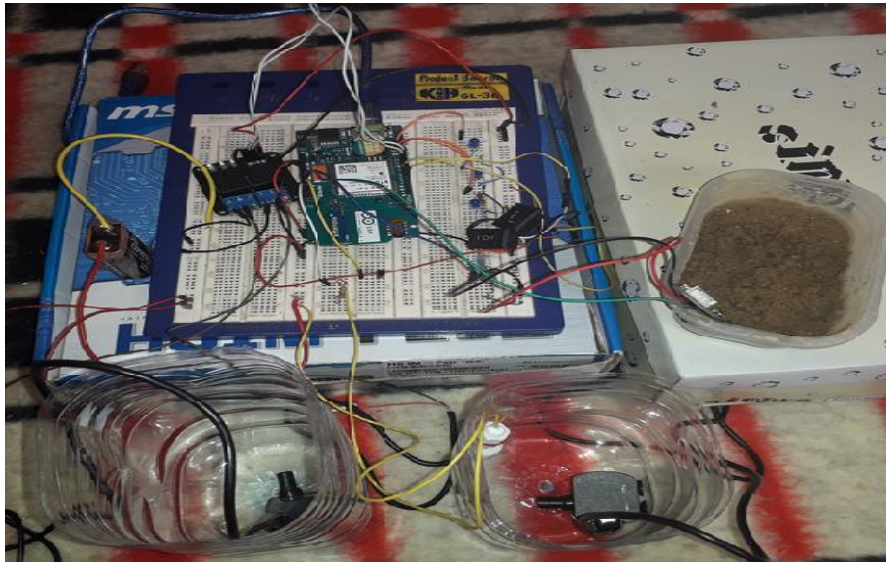


Figure 3. 15 : Circuit Complet sur plaque d’essai

3.7 Brochage des éléments de notre système avec Arduino Uno

Le tableau ci-dessous représente le brochage des différents éléments d’utilisés dans la réalisation du système.

Périphérique E/S	Numéro de la broche	Type
LDR	A0	Entrée analogique
Capteur d’humidité	A1	Entrée analogique
Capteur de température	A2	Entrée analogique
Capteur de Niveau	A3	Entrée analogique
Led1	13	Sortie numérique
Led2	12	Sortie numérique
Led3	11	Sortie numérique
PR	10	Sortie numérique
PA	9	Sortie numérique
PV	8	Sortie numérique
Sélecteur	6	Entrée numérique
BE	5	Entrée numérique
BA	4	Entrée numérique

Tableau 3. 3 : Brochage des Composants

3.8 Implémentation sur maquette

Après des essais sur ISIS et sur une plaque d’essai nous passons à l’implémentation de notre système sur la maquette pour des éventuels tests.

Notre maquette est la représentation d’une structure publique que nous avons nommée « FatKam ». Une structure dont la partie principale est un jardin comportant des plantes

exotiques et une piscine. Et une seconde partie divisée en deux, l'une est un lieu de jeux d'attractions et l'autre un restaurant.

Nous avons réalisé notre maquette sur un plancher de bois de dimension 49/52.

Les images suivantes sont les différentes parties de maquette :



Figure 3. 16 : Plancher de bois

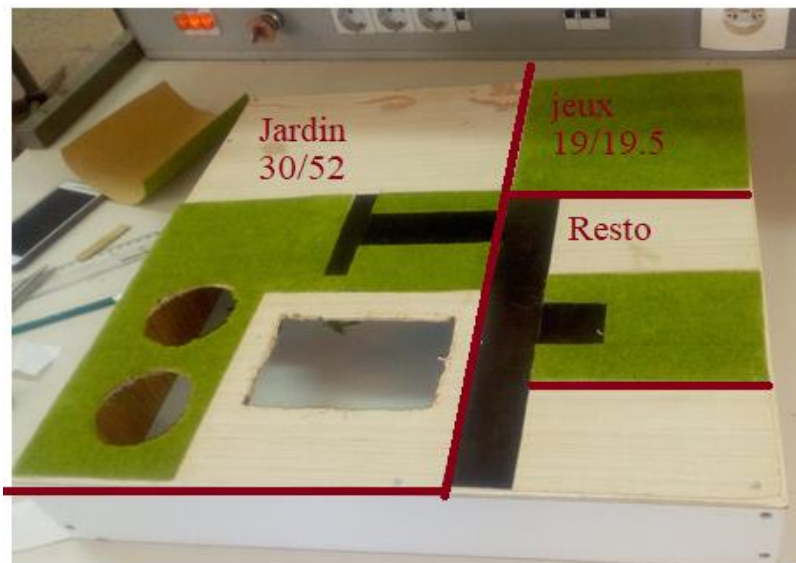


Figure 3. 17 : Délimitation des parties



Figure 3. 18 : Image maquette finale du Projet

3.9 Tests de fonctionnalité

Après avoir implémenté le système nous avons fait des tests pour voir la bonne fonctionnalité de ce dernier et ce fut un succès.

Pour la vidange nous avons donné un délai de 3min, et ainsi on a pu voir que chaque 3min PV se met en marche effectuant le stockage de l'eau de la piscine dans le réservoir R_R.

Et lorsque nous avons placé le capteur H au sol sec. La valeur lue par le capteur fait partie de l'intervalle [4.50 5.00], un SMS a été envoyé à notre téléphone portable dont le contenu était '' **Besoin d'arrosage**'' nous prévenant ainsi l'état du sol. Suite à cela nous avons envoyé un message dont le contenu est ''**POMPEon**'' au module GSM et dès la réception l'arrosage a été déclenché et a duré pendant 30 secondes grâce à la température qui était en dessus de 25°C.

Quand la durée pour l'arrosage s'est écoulé, PR s'est mis en marche remplissant le réservoir R_A jusqu'à atteignant le niveau haut du capteur Ne.

De même l'éclairage automatique fonctionne bien dès que le seuil pour la limite lumière/obscurité a été atteint les LED se sont allumées instantanément.

Et pour finir le système fonctionne parfaitement bien manuellement.

3.10 Coût de réalisation

L'objectif principal de notre projet étant de réaliser un contrôle et commande à distance via un module GSM, nous l'avons réalisé sur une maquette de dimension restreinte. Ceci doit donc être accompagné d'une étude technico économique pour juger de la fiabilité de ce travail qui doit réaliser un gain économique en terme d'énergie (consommation réduite d'électricité : éclairage), économie de ressource naturelle (consommation réduite d'eau : arrosage) rajoutant à cela une baisse facture de frais de conception qui rend le projet réduit en terme de coût de réalisation.

A travers le mode de fonctionnement de ce système, on remarque que l'opération d'arrosage ne s'enclenche qu'en cas de nécessité et que la quantité d'eau est contrôlée en fonction du besoin. De même, pour l'éclairage (lampes LEDs) ne s'allument que dans l'obscurité et cela puisqu'il s'agit d'un lieu public.

Dans le tableau suivant nous avons cité tous les composants intervenant dans la réalisation de notre système et la conception de la maquette ainsi que leurs prix afin de savoir son coût total. Ainsi après calcul, la réalisation de notre projet recommande un total de dépenses allant jusqu'à la somme de 10125DA, ce qui paraît un prix relativement abordable.

Le coût de cette réalisation est abordable, ce qui nous projette à formuler l'idée de le mettre en pratique dans des environnements plus importants intervenant ainsi dans la télé-irrigation.

Composant	Nombre	Prix unitaire	Prix total
Pompe 3.5 à 9V	2	1600DA	3200DA
Pompe 12V	1	1000DA	1000DA
Capteur d'humidité	1	800DA	800DA
Détecteur de niveau d'eau	1	500DA	500DA
LDR	1	50DA	50DA
LED	3	10DA	30DA
Relai 2 chaines	1	700DA	700DA
Relai	1	400DA	400DA
Résistance	5	5DA	25DA
Sélecteur	1	80	80DA
Interrupteurs	2	50	100DA
Maquette			3240DA
Total			10125DA

Tableau 3. 4 : Ensemble des composants et leurs prix

3.11 Conclusion

Ce chapitre étant riche en pratique, nous avons montré les relations entre les différents éléments du système, comment sont branchés les différents composants avec l'Arduino Uno. Et à établir la communication entre le module GSM et la carte Arduino Uno.

Pour mettre en application notre système, plusieurs tests ont donc été effectués pour témoigner de la fiabilité de cette réalisation permettant ainsi de voir si à travers ces tests, on a respecté le cahier des charges proposé, en même temps dans un cas contraire, montrer ses limites.

Les tests ont essentiellement porté sur les capteurs de température, d'humidité pour activer l'arrosage et le capteur de lumière LDR pour activer l'éclairage. Ces résultats ont d'abord été réalisés sur Proteus ISIS, sur plaque d'essai puis sur maquette finale.

Les commandes AT ont aussi fait l'objet de cette étude, servant ainsi à tester, lire, écrire et exécuter des commandes pour communiquer avec le module.

Conclusion générale et perspectives

L'eau, indispensable à la vie, est un bien précieux. L'économie de cette dernière et celle de l'énergie est un vecteur essentiel à l'amélioration des conditions de vie et de la cohésion de la société tout entière.

Dans ce projet de fin d'études, nos objectifs étaient de réaliser un système de télé-irrigation et d'éclairage automatique. Economiser l'eau en utilisant l'eau d'une piscine (destiné à s'écouler dans l'égout) pour faire l'arrosage. Gérer l'eau afin de limiter les gaspillages grâce à un suivi des quantités d'eau utilisées pour les arrosages en fonction de l'humidité, de la température etc.... Piloter à distance des pompes à eau grâce à son téléphone portable qui permet un gain en temps et le confort de l'utilisateur.

Puisqu'il s'agit d'une structure publique, qui en principe diversifie les activités offertes aux visiteurs ou adhérents (piscine, jardin public, aire de jeux, restaurant,), une gestion de la lumière artificielle a été proposée au côté de l'opération d'arrosage en fonction de la lumière naturelle.

La réalisation de ce projet a été très bénéfique pour nous, il nous a permis une amélioration dans les domaines de l'électronique et l'informatique, de découvrir les modules GSM, des nouvelles commandes de communication, des nouvelles fonctions sur de la cartes Arduino et bien d'autres. Mais également d'être confrontés aux problèmes que posent la conception et l'intégration d'un tel projet.

Comme perspectives à ce travail, nous proposons :

- ❖ L'utilisation de Batteries solaires.
- ❖ L'ajout de la commande du contrôle de l'accès à la piscine.
- ❖ Projeter cette étude sur une surface agricole touchant ainsi le domaine de l'irrigation.
- ❖ Pleins d'autres perspectives peuvent être générés en exploitant ces résultats.

Références Bibliographiques

- [1] <http://agripourtous.blogspot.com/2012/10/generalite-sur-lirrigation.html>
- [2] <https://fr.m.wikipedia.org/wiki/Irrigation>
- [3] <https://www.espazium.ch/fr/actualites/leclairage-exterieur-intelligent>
- [4] https://fr.m.wikipedia.org/wiki/Gestion_technique_de_bâtiment
- [5] Photorésistance.pdf
- [6] <http://ressources.univ-lemans.fr/AccesLibre/UM/Pedago/physique/02/electro/caracled.html>
- [7] www.smartgrids-cre.fr/index.php?p=smarhome-systemes-immotiques
- [8] www.julesmart.be/smartnodes.html
- [9] <https://store.arduino.cc/arduino-gsm-shield-2-integrated-antenna>
- [10] [arduino-pour-bien-commencer-en-electronique-et-en-programmation.pdf](#)
- [11] <https://www.dzduino.com/Soil-Humidity-Sensor>
- [12] <http://www.ti.com/lit/ds/symlink/lm35.pdf>
- [13] <https://www.dzduino.com/detecteur-de-niveau-deau>
- [14] <https://www.dzduino.com/micro-pompe-a-eau-avec-connecteur-usb-3-5-9vdc-3w>
- [15] www.technologuepro.com/gsm/command_at/
- [16] <http://www.cf-techno.com/capteur-humidite-sol>

Références Figures

- [1] <https://www.aquaportail.com/definition-6677-ruissellement.html>
- [2] <https://www.canaux-et-territoire.info/d2/index.php?post2009/12/24/38-les-techniques-d-irrigation-i-irrigation-gravitaire>
- [3] <http://calag.ucanr.edu/Archive/?article=ca.v052n02p38>
- [4] <http://saleplas.net/fr/produits/irrigation-et-agriculture/irrigation-par-aspersion>
- [5] <https://agronomie.info/fr/principes-de-fonctionnement-de-systeme-goutte-agoutte/amp/>
- [6] <https://www.ledvance.ch/fr/application-et-projets/domaines/index.jsp>
- [7] www.systemed.fr/conseils-bricolage/domotique-et-vdi-centraliser-commandes,2454.html
- [8] <http://popups.ulg.ac.be/0037-9565/index.php?id=4910&file=1>
- [9] <https://store.arduino.cc/usa/arduino-gsm-shield-2-integrated-antenna>
- [10] [arduino-pour-bien-commencer-en-electronique-et-en-programmation.pdf](#)
- [11] <https://www.carnetdumaker.net/images/photoresistance-et-symbole/>
- [12] <https://www.dzduino.com/Soil-Humidity-Sensor>

- [13] <https://ardushop.ro/en/electronics/192-temperature-sensor-lm35dz.html>
- [14] <https://www.dzduino.com/detecteur-de-niveau-deau>
- [15] <https://www.dzduino.com/micro-pompe-a-eau-avec-connecteur-usb-3-5-9vdc-3w>
- [16] <https://www.dzduino.com/2-channels-5v-relay-module>
- [17] www.technologiepro.com/gsm/command_at/Image1.jpg

Liste des abréviations

GSM : Global Mobile Système

GPRS : General Packet Radio Service

SMS : Short Message Service

SIM : Subscriber Identity Module

AT : Warning

GTB : Gestion Technique des Bâtiments

LDR : Light Dependent Resistor

LED : Light Emitting Diode

TOR : Tout Ou Rien

USB : Universal Serial Bus

PWM : Pulse Width Modulation

ADC : Analog to Digital Converter

TWI/I2C : Two Wire Interface/ Inter-Integrated Circuit

CPU : Central Processing Unit

RAM : Random Access Memory

AlGaAs : Arséniure de Gallium-aluminium

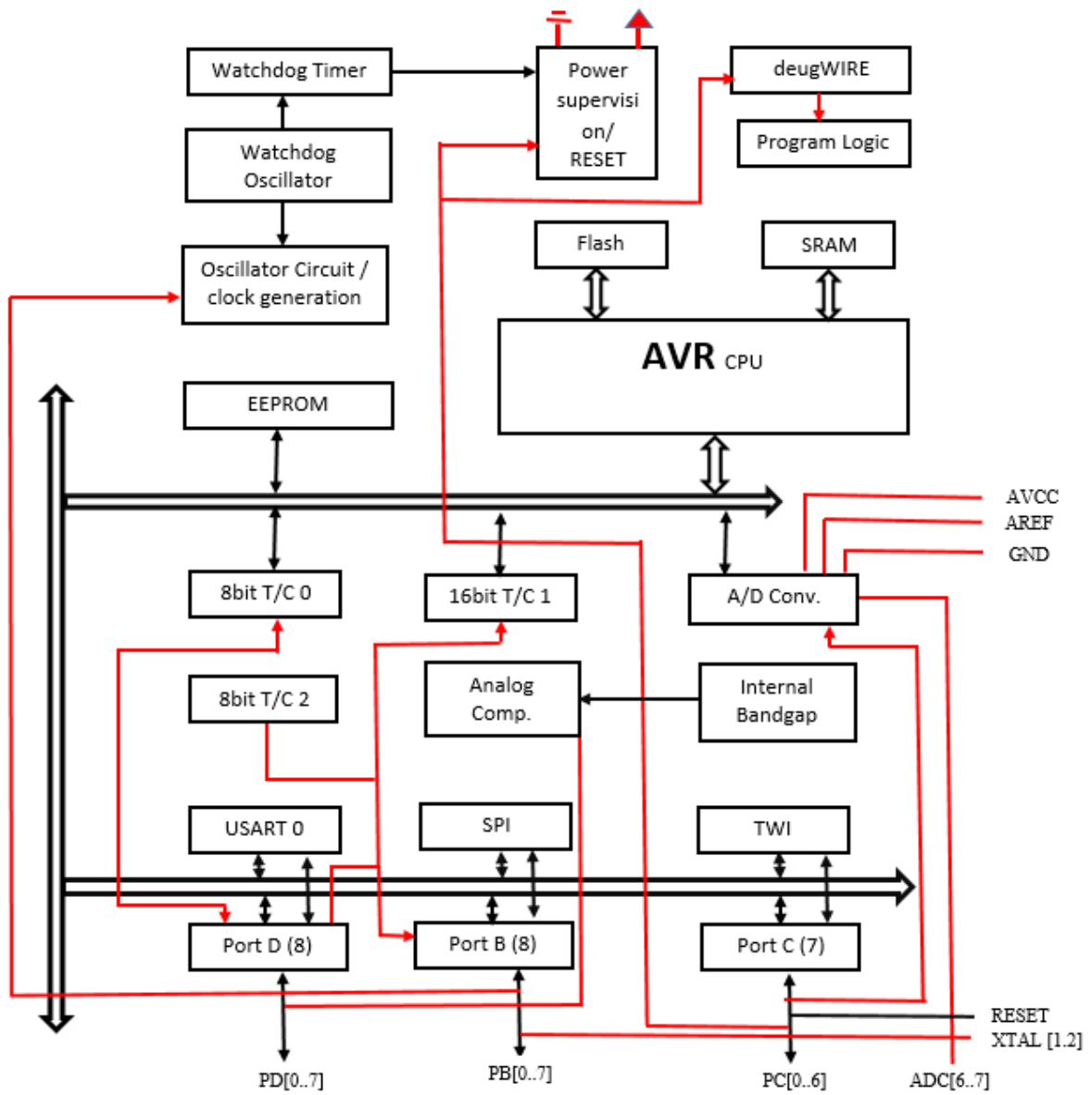
GaAsP : Phospho-arséniure de Gallium

GaN : Nitrure de Gallium

GaP : Phosphure de Gallium

SiC : Carbure de Silicium

InGaN : Nitrure de Gallium-indium



Architecture du microcontrôleur ATmega328

Annexe2

Le programme pour la réception des SMS pour la commande de l'arrosage

```
void arrosage(){
  gsm.println("AT+CMGR=1");
  ecouteGSM("OK", 2000, 1);
  if(H==1.00){
    if (caractCumul.indexOf("POMPEon") > 0) {
      textSMS = "Ordre reçu : Pompe marche !";
      if((T_cel>=2.00) && (T_cel<=10.00)){ digitalWrite(PA, HIGH);
      delay(10000);
      digitalWrite(PA,LOW); }
      if((T_cel>10.00) && (T_cel<=25.00)){
      digitalWrite(PA, HIGH);
      delay(15000);
      digitalWrite(PA,LOW); }
      if(T_cel>25.00){
      digitalWrite(PA, HIGH);
      delay(30000);
      digitalWrite(PA,LOW); }
      statut=1;
      digitalWrite(PR,HIGH);
      Serial.print("statut="); Serial.println(statut);
      delay(100);}
    else textSMS = "Ordre non compris !";}
  gsm.println("AT+CMGD=1,4");
  ecouteGSM("OK", 1000, 0);
  Serial.println(textSMS);
  Serial.println("");
}
```