

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira – Bejaia
Faculté de la technologie.

Département De Génie Electrique.



Mémoire de Master

En vue de l'obtention du diplôme de master en électronique

Option : Automatique

Thème

Réalisation d'un système embarqué d'acquisition de données pour l'analyse et la diagnostique d'un système photovoltaïque

Présenté par :

- Abdoune Fahim.
- Saadi Brahim.

Encadré par :

- M^{dm}: Mezah Samia

Année Universitaire : 2015 / 2016

Remerciement :

*Nos remerciements vont tout premièrement à **Dieu** tout puissant pour la volonté, la santé et la patience, qu'il nous a donnée durant toutes ces longues années.*

*Nous exprimons nos profondes gratitudees à **nos parents** pour leurs encouragements, leur soutien et pour les sacrifices qu'ils ont enduré.*

*Nous tenons également à exprimer nos vifs remerciements à notre encadreur Madame "**Mezah Samia**" pour avoir d'abord proposé ce thème.*

*Nous remercions **les membres de jury** d'examen pour l'honneur qu'ils nous font en participant au jugement de ce travail.*

Nous tenons à remercier vivement tous ceux qui nous ont aidés de près ou de loin à accomplir ce travail.

*Nous remercions vont aussi à tous **les enseignants** du département d'électronique qui ont contribué à notre formation.*

*Enfin nous tenons à exprimer notre reconnaissance à tous **nos amis** et **collègues** pour leur soutien moral et matériel...*

Dédicace :

Je dédie ce modeste travail à :

Mes très chers parents qui ont tant sacrifié pour moi ;

Mes frères et mes sœurs ;

Mes oncles et mes tantes, cousins et cousines ;

Tous mes amis(es) ;

Toute notre promotion Automatique 2016 ;

Tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Fahim

Dédicace :

Je dédie ce modeste travail à :

Mes très chers parents qui ont tant sacrifié pour moi ;

Mes frères et mes sœurs ;

Mes oncles et mes tantes, cousins et cousines ;

Tous mes amis(es) ;

Toute notre promotion Automatique 2016 ;

Tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Brahim

Table de matières

Introduction générale	1
Chapitre I : Le système photovoltaïque	
I.1 Introduction	3
I.2 Principe de Fonctionnement.....	3
I.3 Technologie et structures photovoltaïque	4
I.4 Avantages et inconvénients d'une installation PV	4
I.4.1 Avantages	4
I.4.2 Inconvénients.....	5
I.5 Les Différents Types du Systèmes Photovoltaïques	5
I.5.1 Alimentations électriques faibles puissances.....	5
I.5.2 Installations électriques photovoltaïques autonomes.....	5
I.5.3 Installations au centrales électriques photovoltaïques raccordées au réseau.....	6
I.5.4 Installations au centrale électriques photovoltaïques hybride	7
I.6 Evaluation des performances.....	7
I.7 Conclusion.....	8
Chapitre II : Systèmes embarqués d'acquisition de données	
II.1 Un système d'acquisition de donnée.....	10
II.1.1 Introduction aux systèmes d'acquisition de données	10
II.1.2 Définitions et caractéristiques	10
II.1.3 Les Capteurs	11
II.2 Systèmes embarqués	12
II.2.1 Définition d'un système embarqué.....	12
II.2.2 Contraintes	13
II.2.3 Architecture	14
II.2.4 Caractéristiques	14
II.2.5 Interface utilisateur.....	14
II.2.6 Fiabilité.....	15
II.3 Conclusion	15
Chapitre 3 : La plateforme de développement Arduino	
III.1 Introduction	17
III.2 Les cartes Arduino.....	17
III.3 Les gammes de la carte Arduino	18

Arduino Nano.....	19
III.4 Pourquoi Arduino UNO.....	20
III.5 La constitution de la carte Arduino UNO.....	20
III.5.1 Le Microcontrôleur ATmega328.....	20
III.5.2 Les sources de l'alimentation de la carte.....	21
III.5.3 Les connecteurs.....	22
III.5.4 Les LED de contrôle.....	22
III.5.5 Entrée analogique.....	22
III.5.6 Entrées /sorties numérique.....	22
III.5.7 Les sorties électriques.....	23
III.6 Les cartes d'extensions Arduino.....	23
III.7 La programmation.....	25
III.7.1 L'environnement de la programmation.....	25
III.7.2 Structure générale du programme.....	25
III.7.3 Téléchargement du programme.....	26
III.7.4 Description du programme.....	27
III.8 Exemple d'un programme Arduino.....	27
III.8.1 Ajouter les bibliothèques et déclaration des variables.....	27
III.8.2 Fonction de configuration setup().....	27
III.8.3 Programmation des interactions loop() :.....	28
III.9 Les étapes de téléchargement du programme.....	28
III.10 Conclusion.....	30
Chapitre IV : Présentation de la réalisation finale et discussion des résultats	
IV.1 Introduction.....	32
IV.2 Matériels et composants utilisés.....	32
IV.2.1 Panneau photovoltaïque.....	32
IV.2.2 La batterie de panneau et ces caractéristiques.....	33
IV.2.3 Bluetooth.....	33
IV.2.4 Les capteurs.....	33
IV.2.4.5 Capteur de courant de la batterie.....	37
IV.2.5 Programmation de la carte arduino.....	40
IV.2.6 L'interface graphique de Matlab.....	40
IV.3 Conclusion.....	43



Conclusion générale :	44
Référence bibliographies	45

List des figures :

Figure I.1 Schéma d'une cellule photovoltaïque .	3
Figure I.2 Schéma typique d'une installation photovoltaïque autonome .	6
Figure I.3 Installation au centrale électrique photovoltaïque raccordée au réseau .	6
Figure I.4 Système d'alimentation autonome hybride photovoltaïque .	7
Figure II.1 Système typique d'acquisition de données.	10
Figure II.2 Diagramme de système de chaine d'acquisition.	12
Figure II.3 Un système embarqué dans son environnement.	13
Figure III.1 La carte Arduino UNO	19
Figure III.2 Microcontrôleur ATmega328.	21
Figure III.3 Exemple de shield (a) et de modules (b) d'Arduino	24
Figure III.4 Interface IDE Arduino	25
Figure III.5 Paramétrage de la carte	26
Figure III.6 Les étapes de téléchargement du code.	29
Figure IV.1 Bilan d'acquisition de donné.	32
Figure IV.2 Schéma de connexion Bluetooth avec arduino.	33
Figure IV.3 Schéma de réalisation de capteur de température	34
Figure IV.4 Schéma de capteur de tension PV	35
Figure IV.5 Schéma de réalisation d'un capteur de courant pour PV	36
Figure IV.6 Schéma de capteur de tension de Batterie	37
Figure IV.7 Schéma de capteur de courant Batterie	38
Figure IV.8 Schéma global de la carte sur ISIS	38
Figure IV.9 Schéma global de la carte sur ARSS	39
Figure IV.10 Plan global de la carte sur ISIS	39
Figure IV.11 Résultat de simulation pour le premier essai	41
Figure IV.12 Résultat de simulation pour le deuxième essai	42
Figure IV.13 Résultat de simulation pour le troisième essai	43

Introduction générale

Les ressources d'énergie fossile sont limitées, le pétrole et le gaz naturel seront pratiquement épuisés dans les quarante années à venir, à cette préoccupation première de développement d'énergie alternatif aux énergie fossiles, est venue se greffer la nécessité de réduire considérablement la production de gaz à effet de serre qui cause de grave dommage à notre environnement et provoque des changements climatiques de grande envergure.

La filière photovoltaïque est un moyen souple d'exploitation d'une des énergies renouvelables les plus abondantes qui est l'énergie solaire. En effet, pour cette forme d'exploitation de l'énergie solaire, on produit directement de l'électricité, ils utilisent des matériaux exploitant les propriétés des semis conducteurs et fournissant un courant continu lorsqu'ils sont éclairés.

L'énergie solaire n'est pas disponible la nuit, il est nécessaire d'équiper les systèmes photovoltaïques autonome par des batterie d'accumulateur qui permettent de stocker l'énergie et de restituer en temps voulu.

Dans ce travail nous nous sommes intéressés à la réalisation d'un système embarqué d'acquisition de données pour l'analyse et diagnostic d'un système photovoltaïque, ce mémoire est partagé en quatre chapitres :

Dans le premier chapitre nous allons commencer par une introduction aux systèmes photovoltaïques, dans le deuxième temps nous montrons le principe de l'effet photovoltaïque, nous allons citer les différents types des systèmes PV, et nous finissons ce chapitre par l'évolution des performances de système photovoltaïque.

Le deuxième chapitre présente les différents systèmes embarqués d'acquisition de données et nous donnerons quelque exemple pour ces systèmes et aussi leurs caractéristiques, nous présenterons aussi la description des différents outils de développement utilisés.

Le troisième chapitre présente la plateforme de développement Arduino et nous allons aussi donner les différents cartes Arduino.

Dans le quatrième chapitre nous allons présenter une réalisation finale de la carte et simulation complète dans ISIS et aussi la visualisation des cinq paramètres (V_{pv} , I_{pv} , V_{batt} , I_{batt} , température) avec leurs interfaces graphiques.



Chapitre I : Le système photovoltaïque

I.1 Introduction

L'augmentation du coût des énergies fossiles d'un côté est la limitation de ses ressources, de l'autre côté a conduit à l'apparition d'autres formes d'énergie telle que l'énergie photovoltaïque comme une solution prometteuse parmi les autres comme l'énergie éolienne et hydraulique. La production photovoltaïque est devenue une technologie mature de l'énergie et la demande sur cette forme énergétique ne cesse de croître [1].

I.2 Principe de Fonctionnement

Une cellule photovoltaïque est basée sur le phénomène physique appelé effet photovoltaïque qui consiste à établir une force électromotrice lorsque la surface de cette cellule est exposée à la lumière, la tension générée peut varier entre 0.3V et 0.7V en fonction du matériau utilisé et de sa disposition ainsi que la température de la cellule, la figure (I.1) illustre une cellule PV typique et sa constitution détaillée.

Ainsi, une différence de potentiel entre les deux faces de la cellule est créée. Les grilles métalliques à l'avant et à l'arrière de la cellule PV collectent les électrons et les trous qui vont donc fournir à un circuit extérieur le courant électrique produit. Si le photon est très énergétique, il ne peut tout de même extraire qu'un seul électron. L'énergie excédentaire est perdue en chaleur. La zone N est couverte par une grille métallique qui sert de cathode, tandis qu'une plaque métallique (contact arrière) recouvre l'autre face du cristal et joue le rôle d'anode. L'épaisseur totale du cristal est de l'ordre du millimètre.

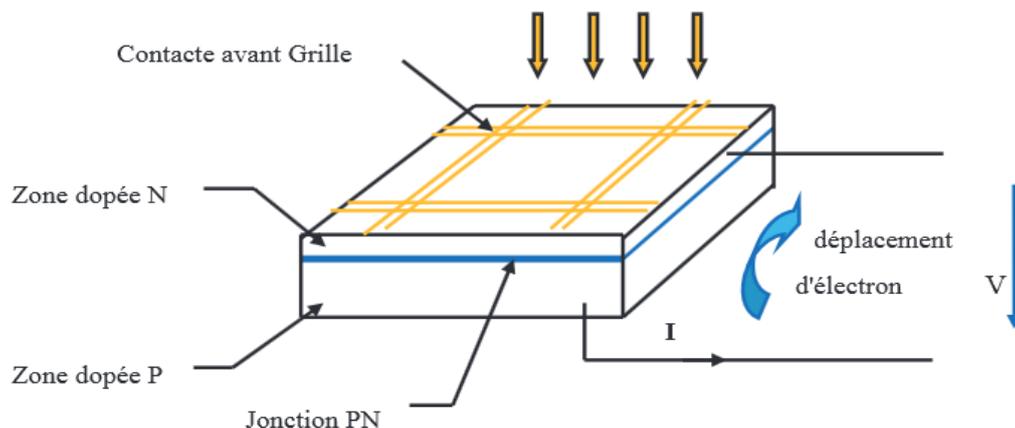


Figure I.1 Schéma d'une cellule photovoltaïque [2].

I.3 Technologie et structures photovoltaïque

La plupart des cellules photovoltaïques sont faites de silicium, l'un des éléments les plus abondants sur la terre; le sable des plages est fait d'oxyde de silicium (silice). Les premières cellules PV offertes mises sur le marché étaient composées de silicium monocristallin; c'est à dire qu'elles contenaient un seul cristal. D'autres techniques de fabrication ont donné les cellules de silicium poly cristallin contenant plusieurs cristaux. Les cellules de silicium poly cristallin sur le marché sont à peine moins efficaces que les cellules monocristallines et leurs utilisations sont très répandues parce que le rapport coût-rendement est le même.

Le développement des technologies des cellules à couche mince a permis de réduire plus encore les coûts en diminuant la qualité de matériaux requis pour fabriquer une cellule. Les cellules de silicium amorphe peuvent contenir une couche mince de silicium et conviennent à la production en série. Par ailleurs, de nouvelles techniques de production ont conduit à la fabrication de cellules de silicium amorphe multi jonction, composées de deux ou trois couches de semi-conducteur. Puisqu'elles sont moins efficaces, il faut de plus grands modules pour produire une quantité d'électricité. Les technologies des cellules à couche mince ont donné également les cellules de tellure de cadmium et de di sélénure de cuivre-indium que l'on voit apparaître sur le marché.

I.4 Avantages et inconvénients d'une installation PV

I.4.1 Avantages

- L'installation ne comporte pas de pièces mobiles qui la rendent particulièrement appropriée aux régions isolées (c'est la raison de son utilisation sur les engins spatiaux)
- le caractère modulaire des panneaux photovoltaïques permet un montage simple et adaptable à des besoins énergétiques divers, les systèmes peuvent être dimensionnés pour des applications de puissances allant du milliwatt au mégawatt Le coût de fonctionnement est très faible vu les entretiens réduits et il ne nécessite ni combustible, ni transport, ni personnel hautement spécialisé.
- La technologie photovoltaïque présente des qualités sur le plan écologique car le produit fini est non polluant, silencieux et n'entraîne aucune perturbation du milieu, si ce n'est par l'occupation de l'espace pour les installations de grandes dimensions.

I.4.2 Inconvénients

- La fabrication du module photovoltaïque relève de la haute technologie et requiert des investissements d'un coût élevé.
- Le rendement réel de conversion d'un module est faible, de l'ordre de 10-15 %. Les générateurs photovoltaïques ne sont pas compétitifs par rapport aux générateurs diesel que pour des faibles demandes d'énergie en régions isolées.
- Tributaire des conditions météorologiques.
- Lorsque le stockage de l'énergie électrique sous forme chimique (batterie) est nécessaire, le coût du générateur est accru.
- Le stockage de l'énergie électrique pose encore de nombreux problèmes.

I.5 Les Différents Types du Systèmes Photovoltaïques

I.5.1 Alimentations électriques faibles puissances

Il s'agit des alimentations électriques faibles telles que les calculettes ou les chargeurs de piles.

I.5.2 Installations électriques photovoltaïques autonomes

Il existe deux systèmes autonomes ; au fil du soleil sans stockage et avec stockage :

I.5.2.1 Système au fil du soleil sans stockage

C'est le système le plus simple puisque l'énergie photovoltaïque est utilisée directement à partir des panneaux. L'appareil alimenté par ce système ne fonctionnera qu'en présence de la lumière, et dès que l'éclairage sera suffisant pour atteindre la puissance demandée. Ce type de système est intéressant pour toutes les applications qui n'ont pas besoin de fonctionner dans l'obscurité, et pour lesquelles le besoin en énergie coïncide avec la présence de la lumière. L'inconvénient de ce système, c'est qu'on ne profite pas toujours des éclairages les plus élevés ; pas de stockage, donc pas de récupération des surplus d'énergie solaire.

I.5.2.2 Système au fil du soleil avec stockage

En site isolé, le champ photovoltaïque peut fournir directement l'énergie électrique nécessaire pour faire fonctionner les récepteurs (éclairage et équipement domestique). Un système de régulation et une batterie d'accumulateurs permettent de stocker l'énergie électrique qui sera ensuite utilisée en l'absence du Soleil. Les batteries sont utilisées pour stocker l'énergie

électrique sous une forme chimique. Elles restituent l'énergie électrique au besoin selon ses caractéristiques.

Le régulateur de charge a pour fonction principale de protéger la batterie contre les surcharges et les décharges profondes. Il est un élément essentiel pour la durée de vie de la batterie. En site isolé, on peut aussi utiliser des récepteurs fonctionnant en courant alternatif. Dans ce cas, l'installation comprendra un onduleur. On peut citer quelques exemples de systèmes autonomes, comme les balises en mer, les lampadaires urbains, le pompage solaire et les maisons en sites isolés.

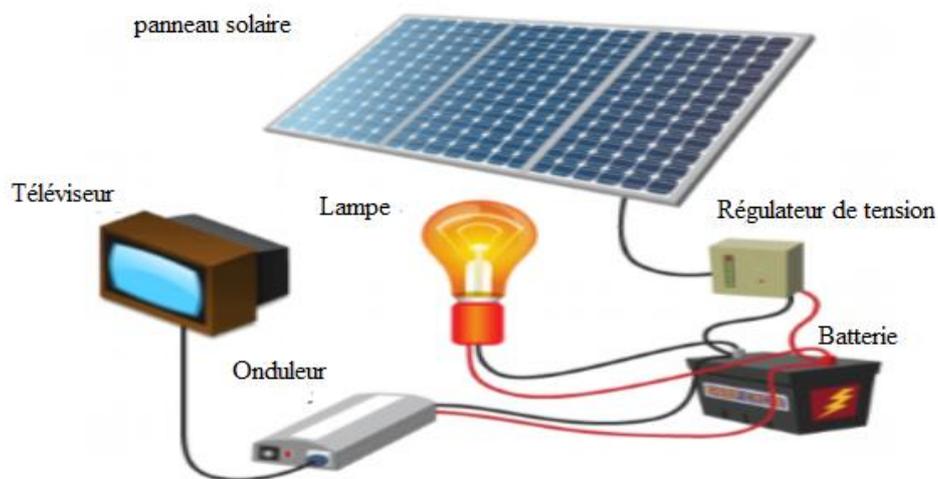


Figure I.2 Schéma typique d'une installation photovoltaïque autonome [4].

I.5.3 Installations au centrales électriques photovoltaïques raccordées au réseau

Un générateur photovoltaïque connecté au réseau n'a pas besoin de stockage d'énergie, donc le maillon le plus problématique (et le plus cher) d'une installation autonome. C'est en fait le réseau dans son ensemble qui sert à stocker l'énergie.

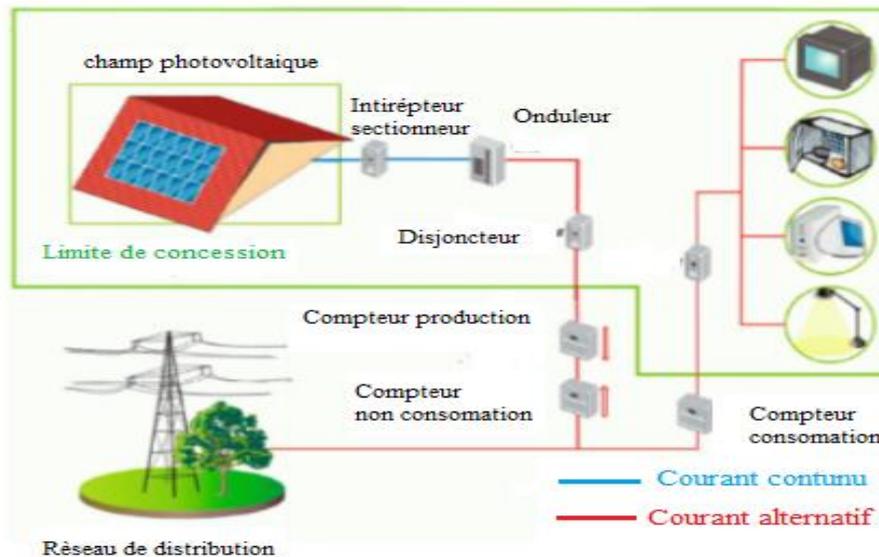


Figure I.3 Installation au centrale électrique photovoltaïque raccordée au réseau [5].

I.5.4 Installations au centrale électriques photovoltaïques hybride

Une des limites d'un système autonome purement photovoltaïque, comme on vient de le décrire, est qu'il fournit une puissance donnée, variable selon la saison, mais que l'on ne peut pas dépasser au risque de détruire la batterie, or les consommateurs ont des besoins qui évoluent et pas forcément en phase avec les saisons.

Avoir un système hybride, c'est disposer d'une autre source d'électricité pour compléter l'apport photovoltaïque. Cette autre source peut être un groupe électrogène (appelé aussi génératrice) ou une éolienne. De tels systèmes ont habituellement des accumulateurs de stockage d'énergie et donc ils permettent en outre de recharger la batterie lorsqu'elle est faible.

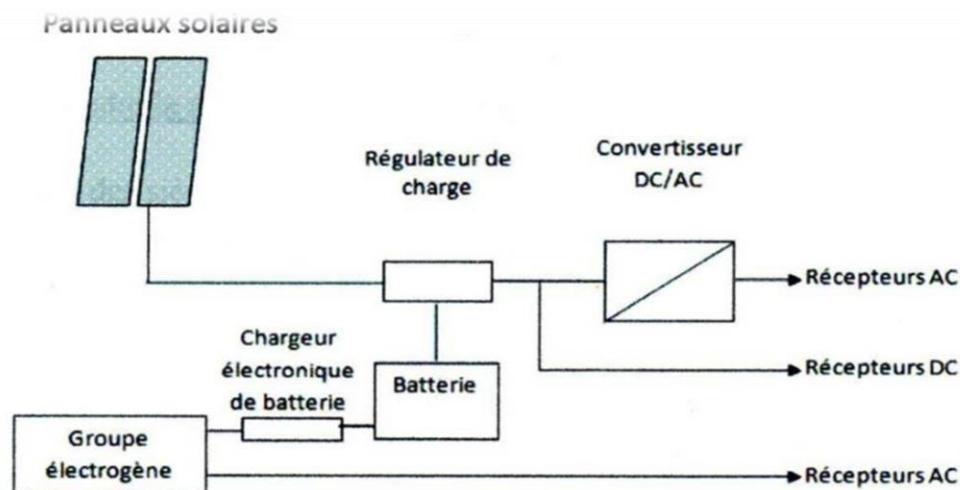


Figure I.4 Système d'alimentation autonome hybride photovoltaïque [7].

Un tel système photovoltaïque hybride fournit une grande fiabilité et peut représenter la solution adéquate pour les sites isolés ; il convient aussi lorsque la demande en énergie est élevée (pendant l'hiver ou tout le long de l'année) [6].

I.6 Evaluation des performances

Les performances d'une installation photovoltaïque dépendent de plusieurs facteurs : la puissance des panneaux photovoltaïques et leur rendement, la qualité des appareils annexes (onduleur, batterie, régulateur...), le dimensionnement adéquat du système et l'ensoleillement. La méthode d'évaluation de ces performances dépend évidemment du type d'installation (autonome, hybride, ...) et elle est toujours basée sur un ensemble de mesures qui doivent être effectuées : à titre d'exemple, pour une installation autonome simple, les mesures à effectuer sont principalement la tension et courant du panneau photovoltaïque, tension et courant de batterie, la température et l'éclairement.

I.7 Conclusion

Dans ce chapitre, nous avons donné un aperçu général sur le domaine photovoltaïque. Nous avons présenté le principe de fonctionnement d'une cellule solaire, les différents types d'installation de systèmes photovoltaïques et la base de la méthode d'évaluation de leurs performances.



Chapitre II : Systèmes embarqués d'acquisition de données

II.1 Un système d'acquisition de donnée

II.1.1 Introduction aux systèmes d'acquisition de données

Le système d'acquisition de données est un ensemble d'appareils interconnectés de manière à permettre l'obtention automatique d'une série de mesure sur un objet (figure II.1). L'analyse et le choix des caractéristiques d'un système d'acquisition est fortement liés l'application. Le choix d'une réalisation doit être le résultat d'un travail de synthèse entre les connaissances des moyens de mesure, des structures possibles et des éléments disponibles [8].

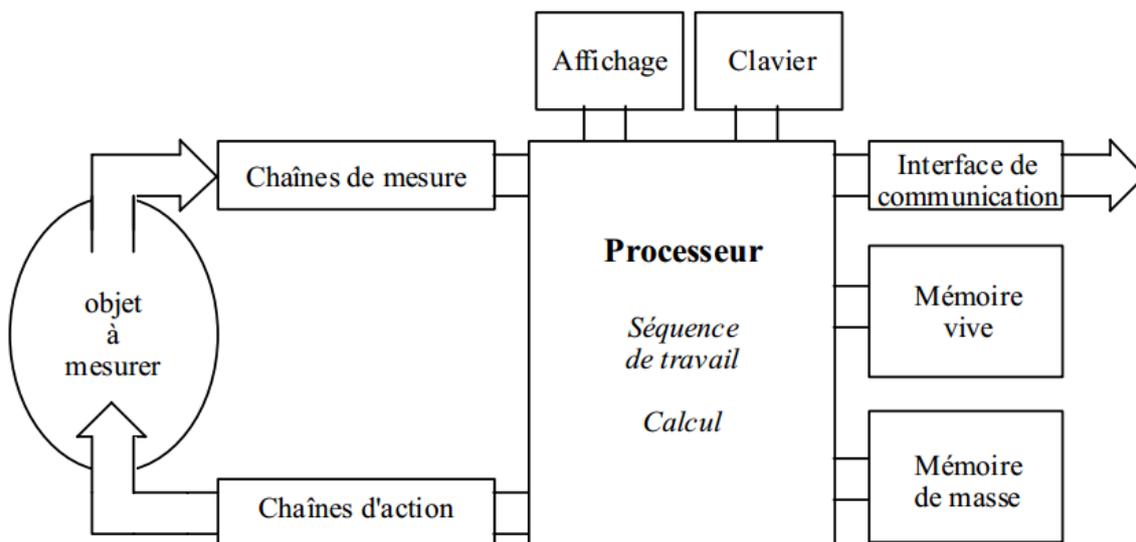


Figure II.1 Système typique d'acquisition de données.

II.1.2 Définitions et caractéristiques

Tout système d'acquisition comprend au moins une chaîne de mesure. Dès que l'on est en présence de plusieurs chaînes, se pose la question d'une économie de matériel en exploitant un multiplexage, soit sous forme analogique (sortie des conditionneurs ou transmetteurs connectées successivement à un seul convertisseur A/D), soit sous forme numérique (les convertisseurs sont reliés au processeur au travers d'un bus numérique). Pour pouvoir exécuter ses tâches, le processeur doit disposer d'une mémoire vive pour stocker les données, éventuellement la séquence de travail demandée par l'utilisateur. De plus il doit disposer d'un moyen de communication, soit directement avec l'utilisateur par un clavier et un panneau d'affichage, soit par l'intermédiaire d'un processeur hôte. Enfin une mémoire de masse est généralement nécessaire (PROM, mémoire flash, disquette, disque dur ...), pour le programme

d'exploitation du processeur et / ou l'archivage des données. La grandeur physique objet de la mesure (déplacement, température, pression etc...) est désignée comme le mesurande et représentée par m ; l'ensemble des opérations expérimentales qui concourent à la connaissance de la valeur numérique du mesurande constitue son mesurage. Lorsque le mesurage utilise des moyens électroniques de traitement de signal, il est nécessaire de produire à partir du mesurande une grandeur électrique qui en soit une représentation aussi exacte que possible ; ceci signifie que la grandeur électrique et ses variations apportent toute l'information nécessaire à la connaissance du mesurande. Le capteur est le dispositif qui soumis à l'action d'un mesurande non électrique présente une caractéristique de nature électrique (charge, tension, courant, ou impédance) désignée par s et qui est fonction du mesurande

$$S = F(m)$$

S : grandeur de sortie ou réponse du capteur

m : grandeur d'entrée ou excitation

La relation $S = F(m)$ résulte dans sa forme théorique des lois physiques qui régissent le fonctionnement du capteur et dans son expression numérique de sa construction (géométrie, dimensions), des matériaux qui le constituent et éventuellement de son environnement et de son mode d'emploi (température, alimentation).

II.1.3 Les Capteurs

Les capteurs sont l'un des éléments de fin de chaîne d'acquisition de données (figure II.2). Ils sont sensibles aux phénomènes physiques ou chimiques. Leur rôle consiste à produire un signal électrique, le plus souvent une tension analogique, mais aussi parfois une intensité, une fréquence ou une série de pulsations. Ces signaux sont liés au milieu dans lequel ils sont placés ou au phénomène qu'ils doivent détecter. Leurs caractéristiques importantes sont le domaine d'utilisation, la sélectivité, la sensibilité, la dérive et la reproductibilité, l'encombrement, la fiabilité, le coût, ...etc. Le problème d'utilisation d'un capteur est dû à la perturbation du phénomène mesuré, au niveau de signal, aux parasites lors de la transmission de leurs signaux, à la non-linéarité de la réponse.

Les signaux issus du capteur ne sont pas toujours directement utilisables par le dispositif de traitement, un conditionnement est alors nécessaire avant leurs exploitation tel que: amplifier les signaux, décaler leur origine de manière à exploiter au mieux l'amplitude acceptée par

l'organe de mesure électrique, transformer des courants ou des fréquences en tensions, filtrer pour éliminer des interférences ...etc.

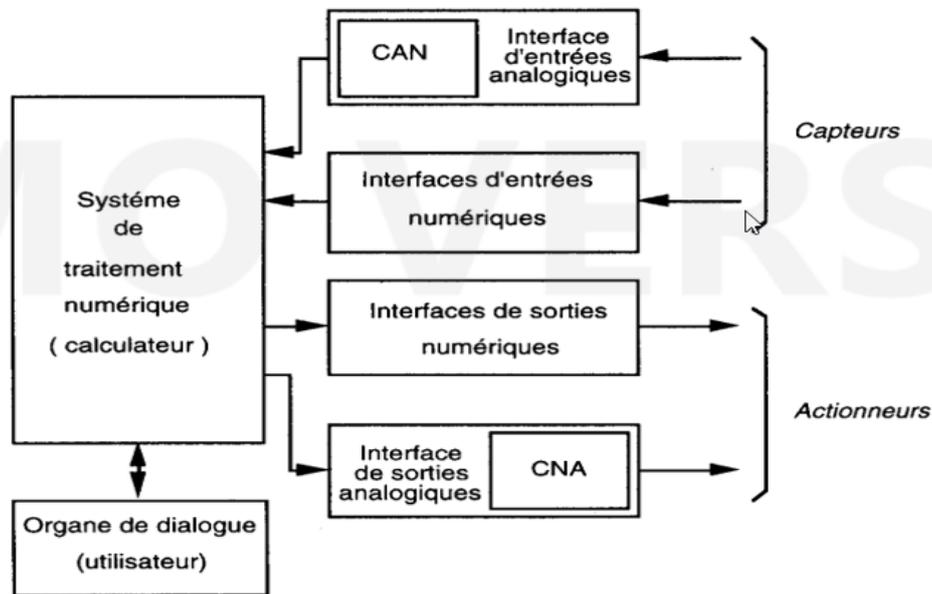


Figure II.2 Diagramme de système de chaîne d'acquisition.

Lorsque la distance entre le capteur et le système convertisseur analogique-numérique est grande, on a intérêt à transmettre un signal amplifié, à relativement basse impédance, et éventuellement symétrique, pour diminuer l'importance relative des tensions parasites ajoutées le long des câbles de liaison. C'est un autre rôle du conditionnement du signal.

II.2 Systèmes embarqués

II.2.1 Définition d'un système embarqué

Un système embarqué est défini comme un système électronique et informatique autonome, souvent temps réel, spécialisé dans une tâche bien précise, le terme désigne bien le matériel informatique que le logiciel utilisé. Ses ressources sont généralement limitées ; cette limitation est généralement d'ordre spatiale (encombrement réduit) et énergétique (consommation restreinte) [9].

La complexité croissante des systèmes embarqués, nécessite de conception globale qui tienne compte des fonctionnalités et des constituants (capteurs, actionneurs, contrôleurs, réseaux), des

perturbations et des fortes contraintes de coût ou environnementale (retard, incertitudes, ...) (figure II.3).

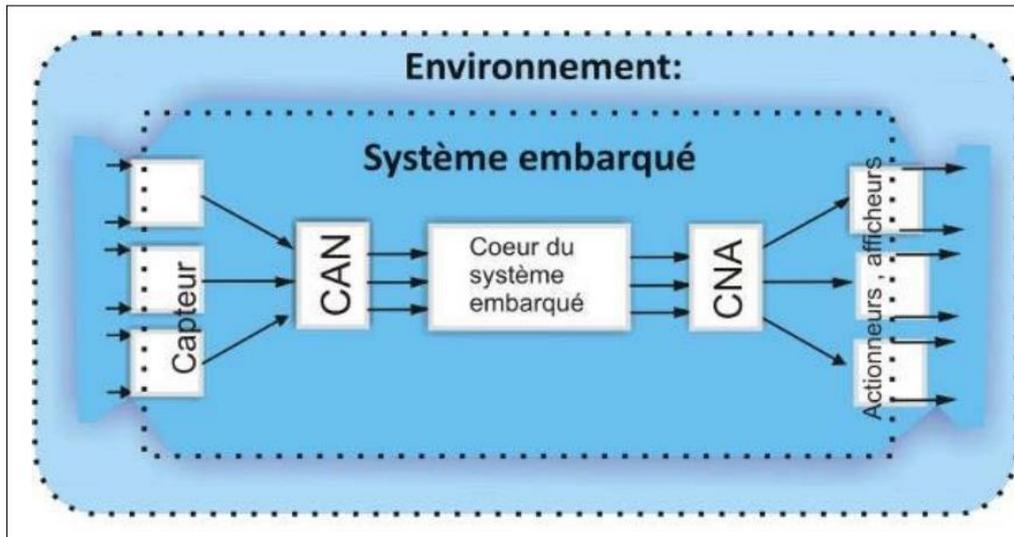


Figure II.3 Un système embarqué dans son environnement.

II.2.2 Contraintes

Les systèmes embarqués exécutent des tâches prédéfinies et ont un cahier des charges contraignant à remplir, qui peut être d'ordre :

- De coût : Le prix de revient doit être le plus faible possible surtout s'il est produit en grande série.
- D'espace compté
- De puissance de calcul : il convient d'avoir la puissance de calcul juste nécessaire pour répondre aux besoins et aux contraintes temporelles de la tâche prédéfinie.
- D'autonomie : la consommation énergétique doit être la plus faible possible, due à l'utilisation de batteries et/ou de panneaux solaires.
- Temporel : dont les temps d'exécution et l'échéance temporelle d'une tâche sont déterminés (les délais sont connus ou bornés a priori). Cette dernière contrainte fait que généralement de tels systèmes ont des propriétés temps réel.
- De sûreté de fonctionnement : si certains de ces systèmes embarqués mettent des vies humaines en danger en cas de défaillance ; Ils sont alors dits « critiques ».

- De sécurité : pour certains systèmes d'acquisition et de transmission d'informations confidentielles, il devient primordial de conserver et protéger ces informations.

II.2.3 Architecture

Les systèmes embarqués utilisent généralement des microprocesseurs à basse consommation d'énergie ou des microcontrôleurs, dont la partie logicielle est en partie ou entièrement programmée dans le matériel, généralement en mémoire de différents types, EPROM, EEPROM, FLASH, etc.

II.2.4 Caractéristiques

- Plutôt que des systèmes universels effectuant plusieurs tâches, les systèmes embarqués sont étudiés pour effectuer des tâches précises. Certains doivent répondre à des contraintes de temps réel pour des raisons de fiabilité et de rentabilité. D'autres ayant peu de contraintes au niveau performances permettent de simplifier le système et de réduire les coûts de fabrication.
- Les systèmes embarqués ne sont pas toujours des modules indépendants. Le plus souvent ils sont intégrés dans le dispositif qu'ils contrôlent.

II.2.5 Interface utilisateur

Certains systèmes embarqués peuvent ne pas avoir d'interface utilisateur (ils sont alors spécialisés dans une seule tâche). Mais cette interface peut également être similaire à celle d'un système d'exploitation d'ordinateur. Les systèmes les plus simples comportent uniquement des boutons, des LED.

- Les systèmes les plus complexes peuvent avoir un écran tactile ou encore un écran comportant des boutons de façon à minimiser l'espace.
- Écran et peu de mémoire.
- Des boutons changent selon l'écran et la sélection se fait naturellement en pointant la fonction désirée.
- Les ordinateurs de poche possèdent en général un bouton au style de « joystick » pour la navigation.

- Avec l'explosion du web, les fabricants de systèmes embarqués ont proposé une nouvelle option, une interface au style d'une page web sur une connexion au réseau. Cela permet d'éviter le coût d'un système sophistiqué tout en conservant une interface complète sur un autre ordinateur, quand cela est nécessaire. Interface couronnée de succès pour les installations permanentes à distance, les routeurs en particulier.

II.2.6 Fiabilité

Les systèmes embarqués sont la plupart du temps dans des machines qui doivent fonctionner en continu pendant de nombreuses années, sans erreurs, et dans certains cas, réparer eux-mêmes les erreurs quand elles arrivent. C'est pourquoi les logiciels sont toujours développés et testés avec plus d'attention que ceux pour les PC. Les pièces mobiles non fiables (par exemple les lecteurs de disques, boutons ou commutateurs) sont proscrites.

La question de la fiabilité peut inclure :

- Le système ne peut pas être éteint pour des réparations ou ce sont des réparations inaccessibles. La solution peut être des pièces détachées supplémentaires ou un "mode mou" du logiciel qui fournit un fonctionnement partiel. Par exemple, les câbles sous-marins, les balises de navigation, les puits de forage...
- Le système doit rester en marche pour des raisons de sécurité. Souvent, les sauvegardes sont effectuées par un opérateur. Dans ce cas, le « mode mou » est toléré. Par exemple : les systèmes de contrôle des réacteurs, les usines chimiques, les signaux de train...
- Un arrêt du système peut provoquer des pertes monétaires énormes s'il s'éteint. Par exemple : les systèmes de ponts ou d'ascenseurs, les transferts de fond, les salles de bourse, les ventes ou services automatiques...

II.3 Conclusion

Dans ce chapitre nous avons exposé des généralités concernant les systèmes d'acquisition de données et les systèmes embarqués à travers la présentation de certaines caractéristiques.

Le chapitre suivant sera consacré à la présentation de la plateforme Arduino utilisée pour réaliser notre système d'acquisition de données pour l'analyse et le diagnostic d'un système photovoltaïque.



Chapitre III : La plateforme de développement Arduino

III.1 Introduction

L'utilisation des systèmes embarqués permet de simplifier énormément les schémas électroniques et réduire l'utilisation de composants électroniques, réduisant ainsi le coût de fabrication d'un produit.

Pour des raisons de simplicité et de rapidité de développement et de prototypage, on évitera de concevoir et de réaliser des cartes à microcontrôleurs et on utilise des plateformes de développement à base de ces composants programmables. Ces cartes de développement sont dotées de certains nombres de circuits annexes tels que les connecteurs, switch, afficheurs, capteurs ...etc, qui permettent de tester et valider des applications en moins de temps et avec peu d'effort. Le choix de la carte de développement dépend de ses caractéristiques en termes de prix, caractéristiques techniques, et de l'environnement logiciel de développement.

III.2 Les cartes Arduino

La plateforme de prototypage OpenSource est basée sur les 2 éléments suivants :

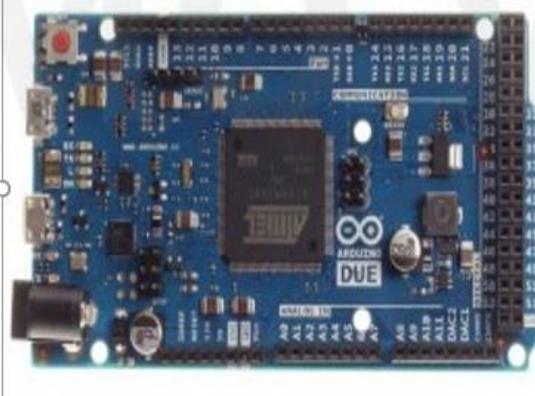
- Matériels : les cartes Arduino et une multitude de carte d'extensions spécifiques (Modules et shields).
- Logiciels : un IDE et une multitude de fonctions dédiées aux cartes d'extensions.

Les cartes Arduino sont généralement construites autour d'un microcontrôleur Atmel AVR (ATmega328, ATmega32u4 ou ATmega2560 pour les versions récentes, ATmega168, ATmega1280 ou ATmega8 pour les plus anciennes), et de composants complémentaires qui facilitent la programmation et l'interfaçage avec d'autres circuits (figure I.13.). Chaque carte possède au moins un régulateur linéaire 5 V et un oscillateur à quartz 16 MHz.

Le microcontrôleur est préprogrammé avec un bootloader de façon à ce qu'un programmeur dédié ne soit pas nécessaire. Les cartes sont programmées au travers d'une connexion série. Les premiers Arduino possédaient un port série RS-232, puis l'USB est apparu sur les modèles actuels, tandis que certains modules destinés à une utilisation portable comme le Lillypad ou le Pro-mini se sont affranchis de l'interface de programmation, relocalisée sur un module USB-série dédié (sous forme de carte ou de câble).

III.3 Les gammes de la carte Arduino

Actuellement, il existe plus de 20 versions de carte Arduino, nous citons quelques-unes dans le tableau ci-dessous, afin d'éclaircir l'évaluation de ce produit scientifique et académique :

Cartes Arduino	Caractéristiques
 <p data-bbox="373 898 675 931">Carte Arduino Due</p>	<ul style="list-style-type: none"> ➤ processeur Atmel SAM3X8E, basé sur une architecture ARM Cortex M3 à 32 bit cadencé à 84MHz. ➤ Une interface USB 2.0 fonctionnant à 480 Mega bits permettant à l'Arduino Due d'agir comme un Host USB (ce qui permet d'interfacer des périphériques USB tels que souris, clavier, appareil photo, téléphone mobile et plus). ➤ 4 ports de communication série à haute vitesse. ➤ 70 connecteurs d'entrée/sortie.
 <p data-bbox="443 1458 616 1491">Arduino Yun</p>	<ul style="list-style-type: none"> ➤ processeur ATmega32u4 ➤ Tension d'entrée (limites) : 6-20V ➤ Entrée/sortie digital : 14 ➤ PWM : 7 canaux ➤ Entrées analogiques : 6 (plus 6 multiplexées sur 6 broches digital) ➤ Courant continu par entrée/sortie : 40 mA ➤ Courant continu pour broche 3.3V : 50 mA ➤ Mémoire Flash : 32 KB (ATmega32u4) dont 4 KB utilisé par le boot loader ➤ SRAM : 2.5 KB (ATmega32u4) ➤ EEPROM : 1 KB (ATmega32u4) ➤ Vitesse d'horloge : 16 MHz
 <p data-bbox="440 2007 608 2040">Arduino uno</p>	<ul style="list-style-type: none"> ➤ Microcontrôleur : ATmega328. ➤ Tension de fonctionnement : 5V. ➤ Tension d'entrée (limites) : 6-20. ➤ Broches I/O digitales : 14 (dont 6 en sortie PWM). ➤ Broches analogiques d'entrée : 6. ➤ Courant par broche I/O : 40 mA. ➤ Courant par broche sous 3.3V : 50 mA. ➤ Flash Memory: 32 KB (ATmega328) of which 0.5 KB used by bootloader. ➤ SRAM : 2 KB (ATmega328). ➤ EEPROM : 1 KB (ATmega328). ➤ Vitesse d'horloge : 16 MHz.

 <p style="text-align: center;">Arduino Nano</p>	<ul style="list-style-type: none"> ➤ processeur Atmel ATmega328 Tension de fonctionnement : 5 V. ➤ Entrées/Sorties digitales : 14 entrées/sorties (dont 6 supportant la sortie PWM). ➤ Entrée analogique : 8 ➤ Courant CC (DC) par broche d'Entrée/Sortie : 40 mA. ➤ Mémoire Flash : 32 KB (ATmega328) dont 2 Kb réservé pour le bootloader. ➤ SRAM : 2 KB (ATmega328). ➤ EEPROM : 1 KB (ATmega328). ➤ Fréquence de l'horloge : 16 MHz.
--	---

Tableau III.1 Les gammes de la carte Arduino

Parmi ces types, nous avons choisi une carte Arduino UNO, l'intérêt principal de cette carte est de faciliter la mise en œuvre d'une telle commande qui sera détaillée par la suite.

L'Arduino fournit un environnement de développement s'appuyant sur des outils open source comme interface de programmation. L'injection du programme déjà converti par l'environnement sous forme d'un code dans la mémoire du microcontrôleur se fait d'une façon très simple par la liaison USB. En outre, des bibliothèques de fonctions "clé en main" sont également fournies pour l'exploitation d'entrées-sorties. Cette carte est basée sur un microcontrôleur ATmega 328 et des composants complémentaires. Cette carte est illustrée dans la figure si dessous.



Figure III.1 La carte Arduino UNO

III.4 Pourquoi Arduino UNO

Il y a de nombreuses cartes électroniques qui possèdent des plateformes basées sur des microcontrôleurs disponibles pour l'électronique programmée. Tous ces outils prennent en charge les détails compliqués de la programmation et les intègrent dans une présentation facile à utiliser. De la même façon, le système Arduino simplifie la façon de travailler avec les microcontrôleurs tout en offrant à toutes personnes intéressées, plusieurs avantages cités comme suit :

- **Le prix (réduits) :** les cartes Arduino sont relativement peu coûteuses comparativement aux autres plates-formes, (les cartes Arduino préassemblées coûtent moins de 2500 Dinars).
- **Multi plateforme :** le logiciel Arduino, écrit en JAVA, tourne sous les systèmes d'exploitation Windows, Macintosh et Linux.
- **Un environnement de programmation clair et simple :** l'environnement de programmation Arduino (le logiciel Arduino IDE) est facile à utiliser, tout en étant assez flexible pour que les utilisateurs avancés puissent en tirer profit également.
- **Logiciel Open Source et extensible :** le logiciel Arduino et le langage Arduino sont publiés sous licence open source, disponible pour être complété par des programmeurs expérimentés.

III.5 La constitution de la carte Arduino UNO

Les différents composants que nous pouvons d'identifier visuellement sur une carte Arduino sont :

III.5.1 Le Microcontrôleur ATmega328

Un microcontrôleur ATmega328 est un circuit intégré qui rassemble sur une puce plusieurs éléments complexes dans un espace réduit au temps des pionniers de l'électronique. Aujourd'hui, en soudant un grand nombre de composants encombrants ; tels que les transistors ; les résistances et les condensateurs, tout peut être logé dans un petit boîtier en plastique noir muni d'un certain nombre de broches dont la programmation peut être réalisée en langage C. la figure III.3 montre un microcontrôleur ATmega 328, qu'on trouve sur la carte Arduino.

Le microcontrôleur ATmega328 est constitué par un ensemble d'éléments qui ont chacun une fonction bien déterminée. Il est en fait constitué des mêmes éléments que sur la carte mère

d'un ordinateur. Globalement, l'architecture interne de ce circuit programmable se compose essentiellement sur :

- **La mémoire Flash** : C'est celle qui contiendra le programme à exécuter. Cette mémoire est effaçable et réinscriptible mémoire programme de 32Ko.
- **La mémoire RAM** : c'est la mémoire dite "vive", elle va contenir les variables du programme. Elle est dite "volatile" car elle s'efface si on coupe l'alimentation du microcontrôleur. Sa capacité est 2 ko

La mémoire EEPROM : C'est le disque dur du microcontrôleur. On y enregistre 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.

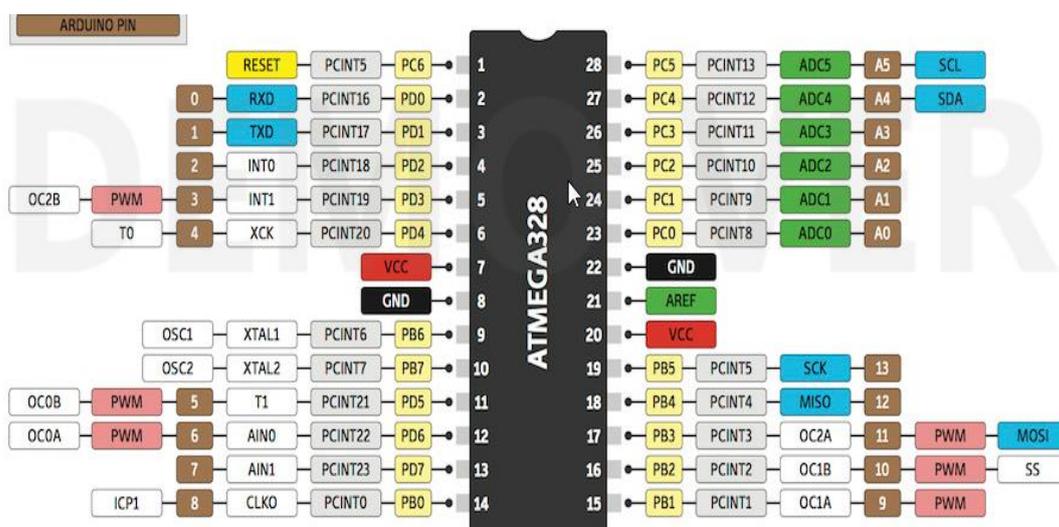


Figure III.1 Microcontrôleur ATmega328.

III.5.2 Les sources de l'alimentation de la carte

On peut distinguer deux genres de sources d'alimentation (Entrée Sortie) et cela comme suit :

- **Vin** : La tension d'entrée positive lorsque la carte Arduino est utilisée avec une source de tension externe (à distinguer du 5V de la connexion USB ou autre source 5V régulée). On peut alimenter la carte à l'aide de cette broche, ou, si l'alimentation est fournie par le jack d'alimentation, accéder à la tension d'alimentation sur cette broche.
- **5V** : La tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (pour info : les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite "tension régulée" obtenue à l'aide d'un composant appelé un régulateur et qui est intégré à la carte Arduino). Le 5V régulé fourni par cette broche peut donc provenir soit de la tension d'alimentation Vin via le

régulateur de la carte, ou bien de la connexion USB (qui fournit du 5V régulé) ou de tout autre source d'alimentation régulée.

- **3V3** : Une alimentation de 3.3V fournie par le circuit intégré FTDI (circuit intégré faisant l'adaptation du signal entre le port USB de notre ordinateur et le port série de l'ATmega) de la carte est disponible :

Ceci est intéressant pour certains circuits externes nécessitant cette tension au lieu du 5V : L'intensité maximale disponible sur cette broche est de 50mA.

III.5.3 Les connecteurs

Les ranges de connecteur qui se nomment aussi les pins, ont plusieurs fonctions, certains sont des entrées et d'autres peuvent agir comme des entrées ou des sorties, certains sont analogiques et d'autres numériques, ces caractéristiques dépendent de modèle de la carte, les descriptions qui suivent s'applique à la carte d'Arduino uno.

III.5.4 Les LED de contrôle

Les petites LED TX et RX permettent de vérifier que les informations sont bien transmises entre la carte et l'ordinateur en expédition (TX) et en réception (RX). La LED ON, qu'on elle est allumé lorsque la carte Arduino est alimenté en électricité est fonctionné, le deuxième présent sur la carte indique lorsque pin 13 est parcouru par un courant électronique.

III.5.5 Entrée analogique

Les connecteurs de A0 à A5 sont des entrées analogiques dont nous pouvons connaître la valeur codée sur 10 bits (de 1 à 1024), grâce à la commande `analogRead()`. Ce sont ces ports que l'on utilisera, par exemple pour un potentiomètre ou un capteur de luminosité par défaut, ces connecteurs sont prévus pour mesurer une tension allant de 0 à 5volt mais cet écart peut être modifier à l'aide de la commande `analogreference()`.

Les ports A5 et A4 peuvent être utiliser comme interface de communication à deux files TWI (TWI Wire interface aussi appelé I2C), dans ce cas, le port A4 est utiliser pour la ligne SDA (serial data ligne) et le port A5 en SCL (serial clock ligne).

III.5.6 Entrées /sorties numérique

Les connecteurs numériques de 0 à 13 peuvent être régler en entrée ou en sortie, grâce à la fonction `pinMode()`, selon le mode choisis. On accédera à leur valeur à l'aide de la fonction `digitalRead()`, ou en y ouvrera une valeur à l'aide de la fonction `digitalWrit()`.

Certains de ces ports ont en outres propriétés particulières :

- Le port « 0 » est capable d'agir comme un port série en réception (RX) ;

- Le port « 1 » est capable d'agir comme un port série en émission (TX) ;
- Les ports «3,5,6,9,10 et 11 permettent d'envoyer PWM (Pulse Width Modulation) codée sur 8 bits (0 à 255) à l'aide de la fonction analogWrite ().
- Les ports 10,11,12 et 13 peuvent être utilisés pour communiquer selon le protocole SPI (serial peripheral interface), de MOTOROLA qui est intéressant pour sa vélocité, dans ce mode le pin 10 gèrera du MOSI (Master Output Slave Input), le pin 12 servira pour le master MISO (Master Input Slave Output) et le pin 13 du serial clock (sck).

III.5.7 Les sorties électriques

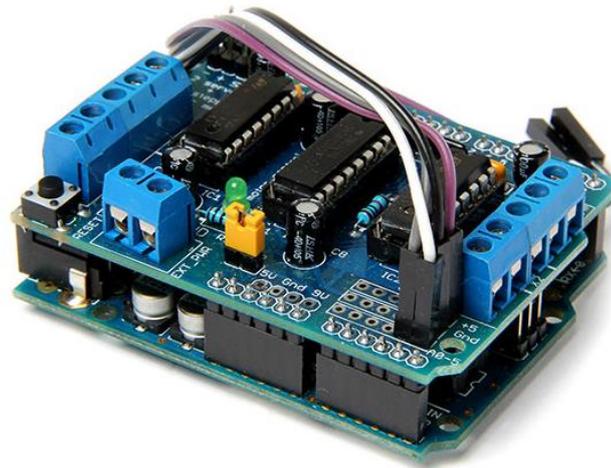
La carte arduino peut envoyer de l'électricité afin d'alimenter les appareils qui y sont connectés. Un cerveau moteur par exemple est connecté à la carte d'Arduino par trois fils la masse, une alimentation de 5v et enfin un câble cerveau pour envoyer les l'ordre à l'appareil. Celui-ci pourrait être alimenté indépendamment de la carte Arduino, mais il est pratique que cette dernière soit capable d'envoyer de l'électricité.

- Les trois connecteurs GND sont des masses. Tout appareil relie à un port GND.
- Le port 3.3 volts, comme son nom le suggéré, envoie une tension 3.3 volts
- Le port 5 volts, de la même manière envoie une tension de 5 volts.
- Le port Vin (9 volts) est en revanche prévu pour recevoir une alimentation électrique.

III.6 Les cartes d'extensions Arduino

Dans le vocabulaire Arduino une carte d'extension se nomme 'shield'. Un shield s'encfiche directement à la carte Arduino pour étendre ses fonctionnalités (figure III.3). Les modules et les shields d'extensions sont très variés et couvrent plusieurs fonctions:

- Communication : Ethernet, wifi, bluetooth, Xbee, ZeegBee, GSM, GPS, ...
- Actionneurs : contrôleurs moteurs (DC, pas à pas , servo moteur), relais,
- Capteurs : température, pression, force, son, caméra, ...
- Entrée/Sorties : boutons, Leds, claviers, hautparleur, tactile
- Affichage : afficheur 7 segments, afficheurs LCD monochrome et graphique, afficheurs TFT ...
- Divers : lecteurs carte microSD,.....



(a)



(b)

Figure III.2 Exemple de shield (a) et de modules (b) d'Arduino

III.7 La programmation

Une telle carte d'acquisition qui se base sur sa construction sur un microcontrôleur, doit être dotée d'une interface de programmation, comme est le cas de notre carte. L'environnement de programmation open-source pour Arduino peut être téléchargé gratuitement (pour Mac OS X, Windows, et Linux).

III.7.1 L'environnement de la programmation

Le logiciel de programmation de la carte Arduino sert d'éditeur de code (langage proche du C). Une fois, le programme tapé ou modifié au clavier, il sera transféré et mémorisé dans la carte à travers de la liaison USB. Le câble USB alimente à la fois en énergie la carte et transporte aussi l'information ce programme appelé IDE Arduino.

III.7.2 Structure générale du programme

Les cartes Arduino sont programmées en C. L'IDE Arduino possède une interface souple et simple est compatible aux différents systèmes d'exploitation.

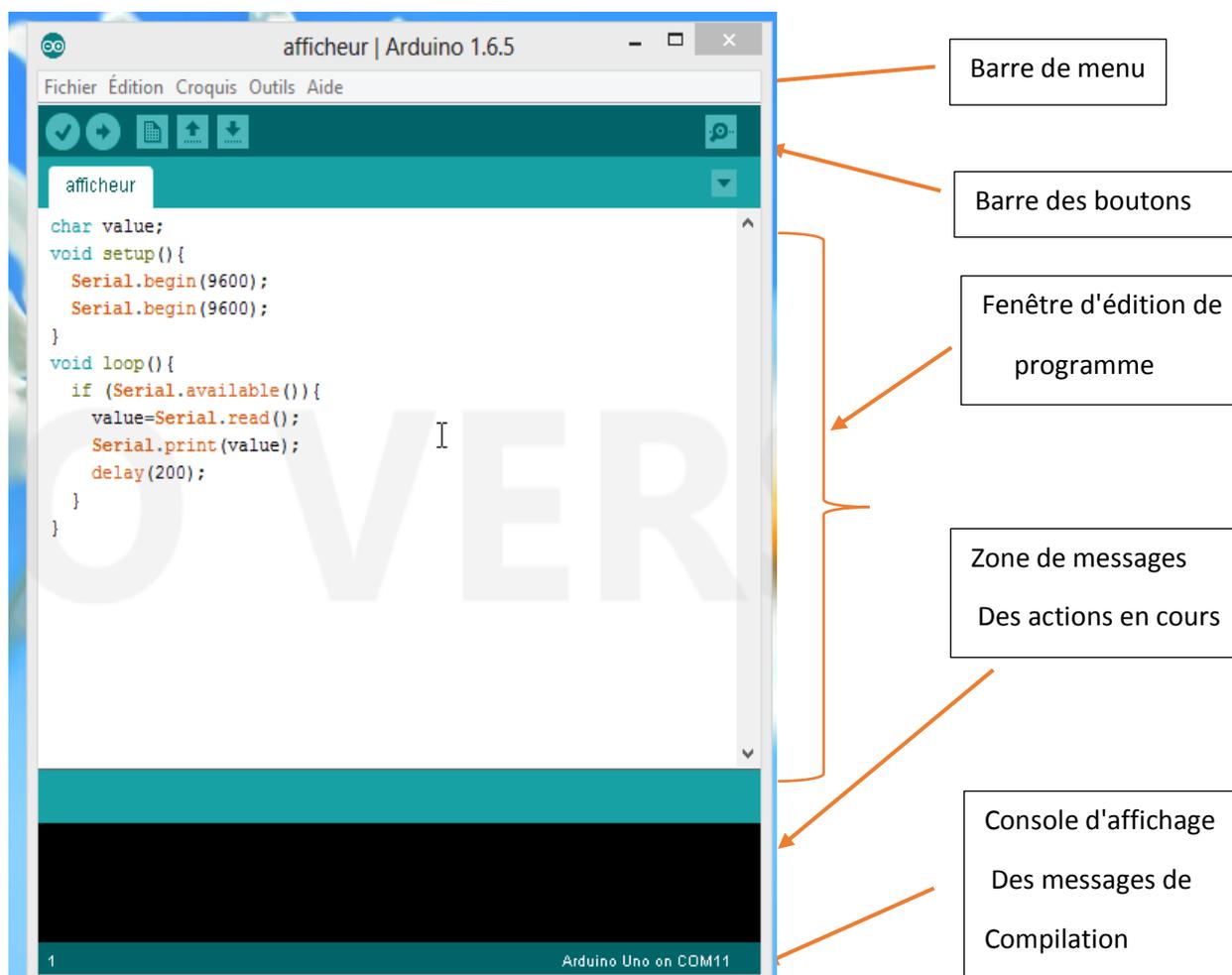


Figure III.3 Interface IDE Arduino

III.7.3 Téléchargement du programme

Avant d'envoyer un programme dans la carte, il est nécessaire de sélectionner le type de la carte (Arduino UNO) et le numéro de port USB (COM 13) comme à titre d'exemple cette figure suivante :

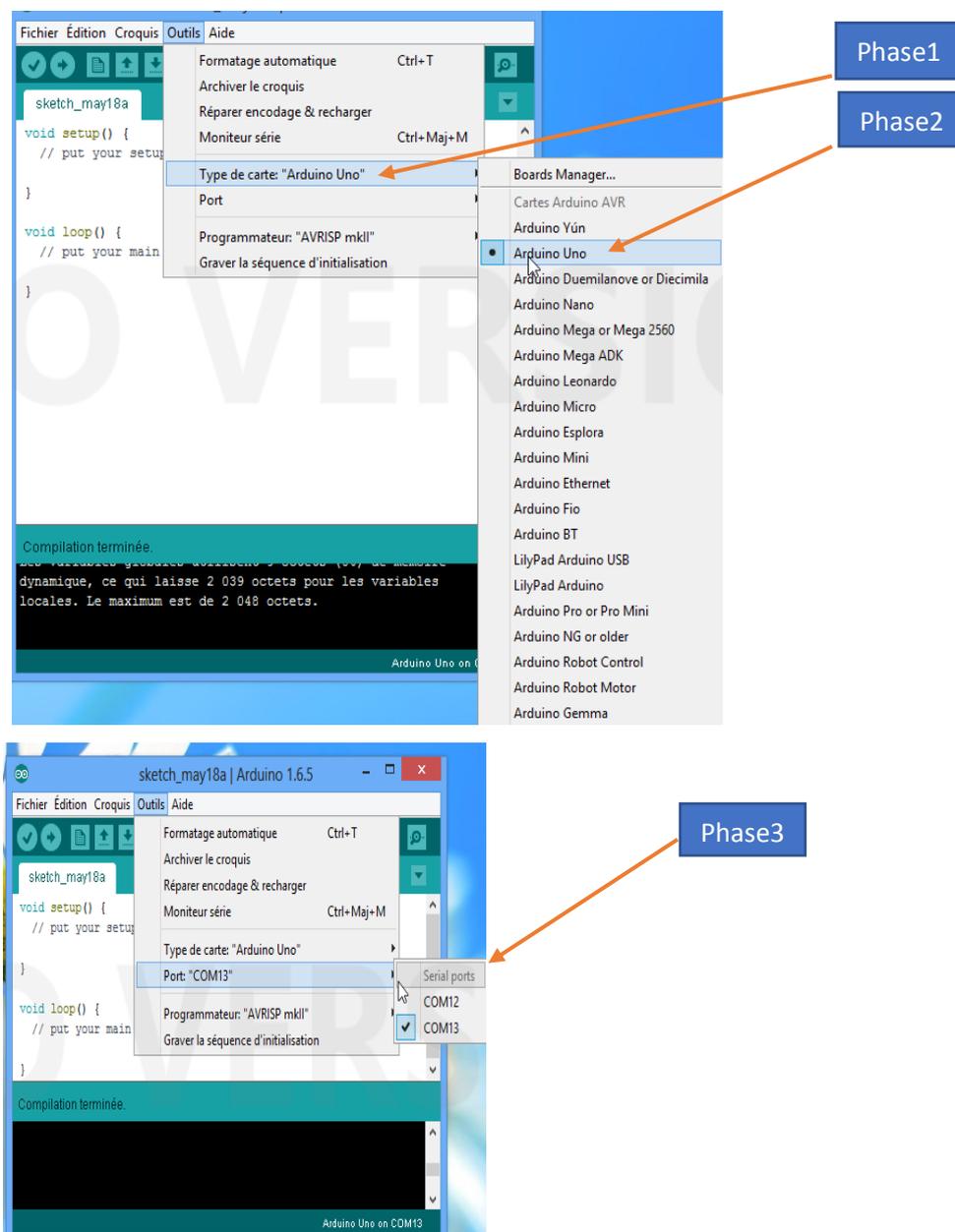


Figure III.4 Paramétrage de la carte

III.7.4 Description du programme

Un programme Arduino est une suite d'instructions élémentaires sous forme textuelle (ligne par ligne). La carte lit puis effectue les instructions les unes après les autres dans l'ordre défini par les lignes de code.

Commentaires : Les commentaires sont, en programmation informatique, des portions du code source ignorées par le compilateur ou l'interpréteur, car ils ne sont pas censés influencer l'exécution du programme.

III.8 Exemple d'un programme Arduino

Supposant que l'on veut mesurer la tension d'un panneau photovoltaïque et afficher cette tension sur un afficheur LCD. Le programme est organisé en trois parties :

III.8.1 Ajouter les bibliothèques et déclaration des variables

```
#include <Wire.h> // ajouter la bibliothèque de I2C
#include <LiquidCrystal_I2C.h> // ajouter la bibliothèque de LCD
int Tension = A0;
int value;
float vout;
float vin;
float R1= 2400.0;
float R2= 1000.0;
LiquidCrystal_I2C lcd(0x27,16,2);
```

III.8.2 Fonction de configuration setup()

Les broches numériques de l'Arduino peuvent aussi bien être configurées en entrées numériques ou en sorties numériques ; pinMode(nom, état) est l'une des quatre fonctions relatives aux entrées / sorties numériques :

```
void setup()
{
  Serial.begin(9600);
  pinMode(Tension, INPUT);
  lcd.init(); // initialisation de l'écran LCD
  lcd.backlight();
  lcd.home();
  lcd.print("Tension :"); // Envoyer le message à LCD.
}
```

III.8.3 Programmation des interactions loop() :

Dans cette boucle, on définit les opérations à effectuer. La fonction `digitalwrite` (nom, état) est une autre des quatre fonctions relatives aux entrées / sorties numériques.

```
void loop()
{
  value = analogRead(Tension);
  vout = (value * 5.0) / 1024.0;
  vin = vout / (R2/(R1+R2));
  lcd.setCursor(0,1);
  lcd.print(vin,1);
  Serial.println(vin);
  Serial.println(vout);
  delay(500);
}
```

III.9 Les étapes de téléchargement du programme

Une simple manipulation enchaînée doit être suivie afin d'injecter un code vers la carte Arduino via le port USB.

1. On conçoit ou on ouvre un programme existant avec le logiciel IDE Arduino.
2. On vérifie ce programme avec le logiciel Arduino (compilation).
3. Si des erreurs sont signalées, on modifie le programme.
4. On charge le programme sur la carte.
5. On câble le montage électronique.
6. L'exécution du programme est automatique après quelques secondes.
7. On alimente la carte soit par le port USB, soit par une source d'alimentation autonome (pile 9 volts par exemple).
8. On vérifie que notre montage fonctionne.

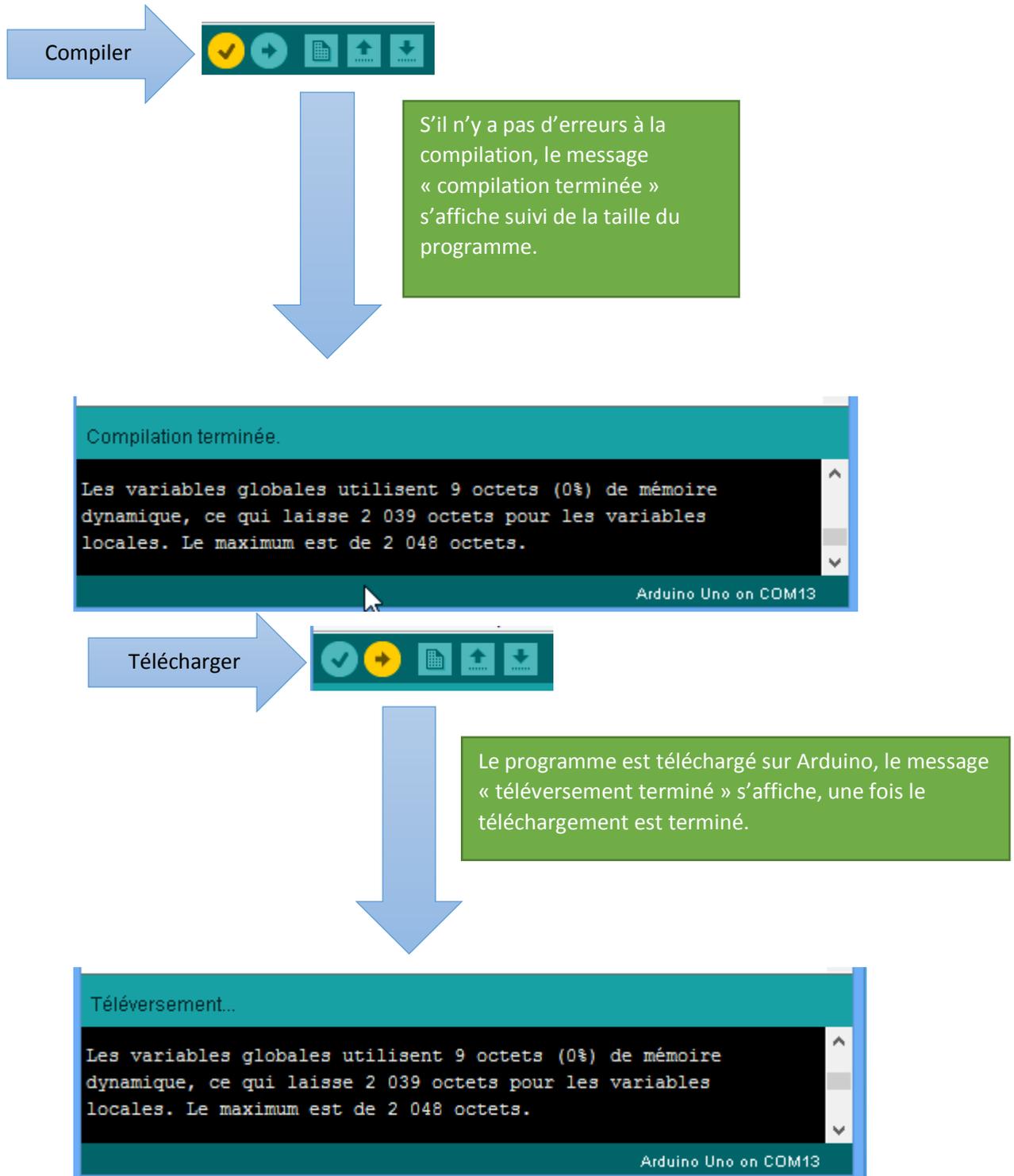


Figure III.5 Les étapes de téléchargement du code.

III.10 Conclusion

Dans ce chapitre, nous avons projeté la lumière sur une carte d'Arduino UNO, donnant ainsi les raisons pour lesquelles nous l'avons choisi, nous avons aussi expliqué les deux parties essentielles relatives à l'exploitation de cette carte (la partie matérielle et la partie programmation).



Chapitre IV : Présentation de la réalisation finale et discussion des résultats

IV.1 Introduction

Notre travail consiste à réaliser une carte électronique à base d'Arduino pour la mesure de cinq grandeurs et leurs transmissions à un ordinateur via une liaison Bluetooth dans le but d'évaluer les performances d'un système photovoltaïque simple constitué d'un panneau solaire, une batterie et une charge résistive. Les grandeurs à mesurer sont : la tension et le courant du panneau, la tension et le courant de la batterie et la température ambiante.

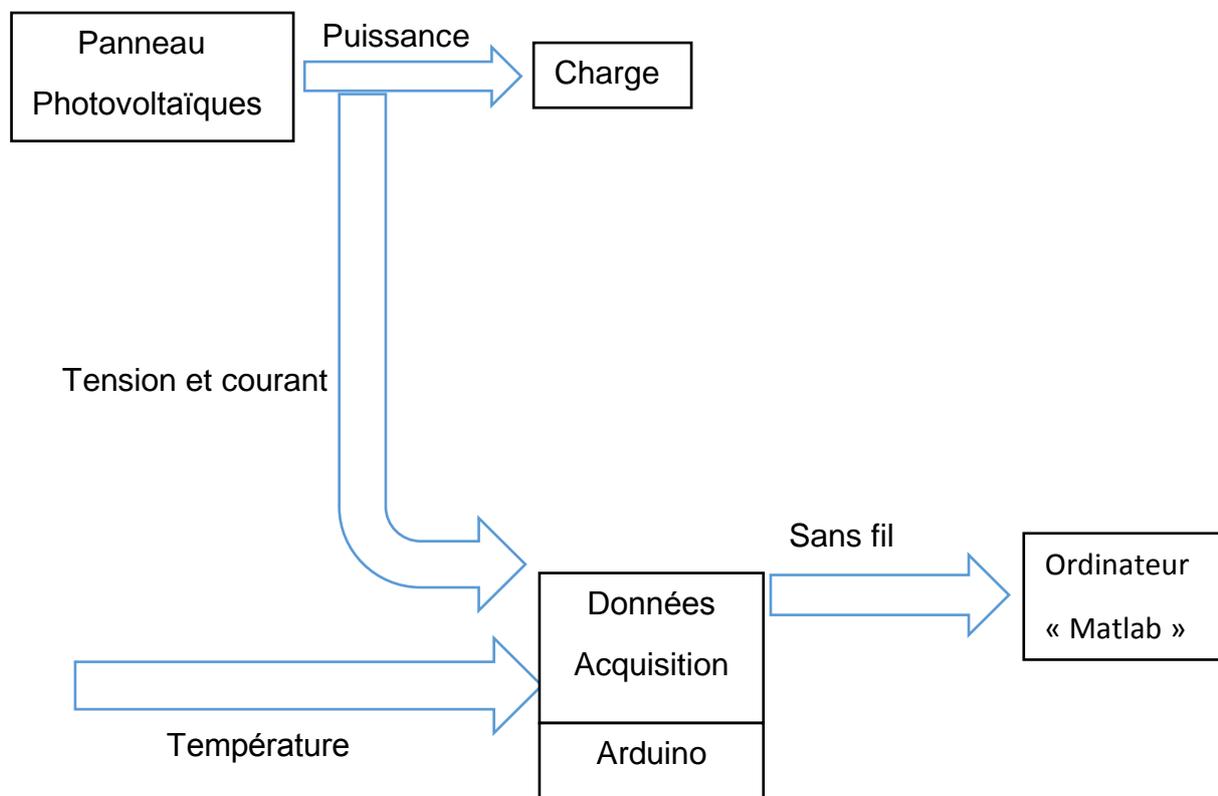


Figure IV.1 Bilan d'acquisition de donn e.

IV.2 Mat eriels et composants utilis es

IV.2.1 Panneau photovolta ique

Nous avons utilis e des caract eristiques suivantes :

- Puissance maximal(P_{mx}) : 10 w.
- Tension en circuit ouvert(V_{co}) : 21V.
- Courant de court-circuit(I_{cc}) : 0,65A
- Tension a la puissance maximale(V_{pm}) : 17V
- Courant a la puissance maximale(I_{pm}) : 0,6A
- La tension de r eseau : 600Vmax

IV.2.2 La batterie de panneau et ces caractéristiques

Toute batterie est un ensemble d'éléments ou de cellules électrochimiques capable de stocker de l'énergie électrique sous forme chimique et Les paramètres électriques suivants sont généralement employés pour caractériser une batterie :

- Batterie ZL P05
- Tension nominale : 12V
- Capacité nominale : 4,5Ah

IV.2.3 Bluetooth

Pour la transmission des données de la carte Arduino vers l'ordinateur, nous utilisons un module Bluetooth avec les caractéristiques suivantes

Modèle : JY-MCU / HC06

- Avec adaptateur de tension 3.6 - 6V intégré.
- Tension d'alimentation 3.3 à 5V DC.
- Les broches TX et RX se connectent pour contrôler les entrées sorties E/S.

La Figure IV.2 montre le schéma de connexion sur ISIS.

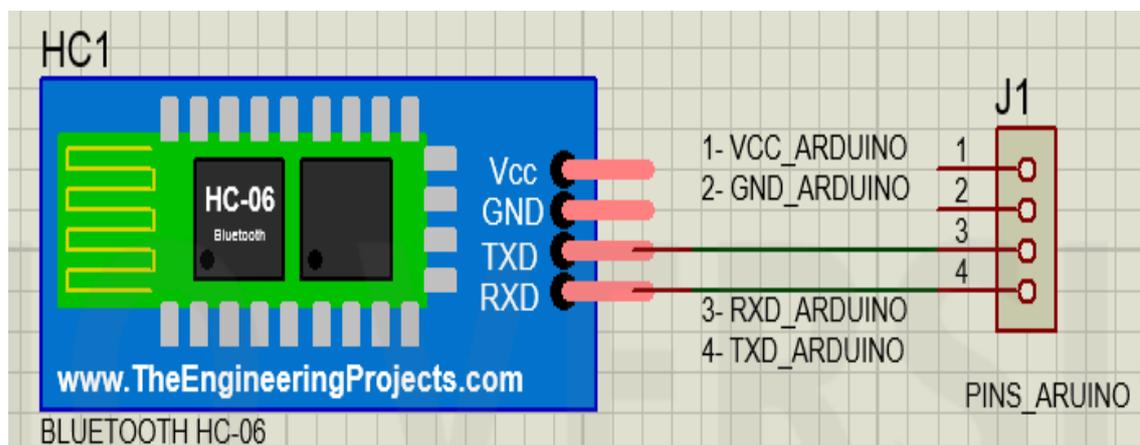


Figure IV.2 Schéma de connexion Bluetooth avec arduino.

IV.2.4 Les capteurs

Nous avons utilisé les différents capteurs suivants :

IV.2.4.1 Capteur de température LM35

Pour mesurer la température de milieu extérieur on a utilisé le capteur de température LM35 et voici ses caractéristiques :

- Tension d'alimentation : 35V
- Tension de sortie : 6V

La figure IV.4 nous montre le schéma de réalisation du capteur de tension

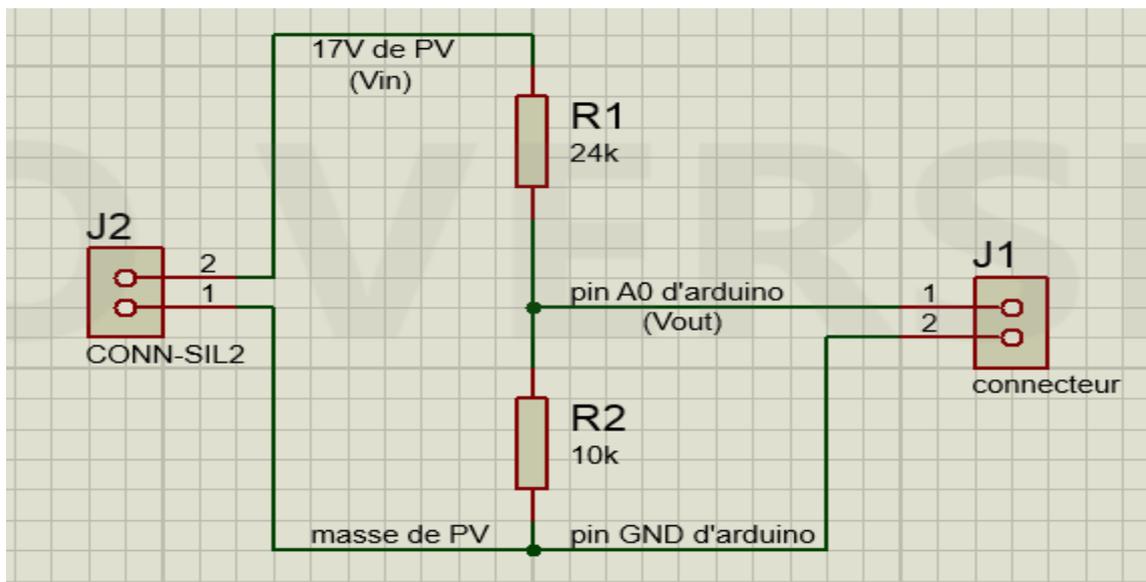


Figure IV.4 Schéma de capteur de tension PV

IV.2.4.3 Capteur de courant pour (PV)

Pour mesurer le courant de PV nous avons réalisé un capteur de courant par la méthode shunt, le problème rencontré c'est que le principe de cette méthode consiste à choisir une faible résistance, de quelque Ω . Et la tension au borne de cette résistance est très faible, de quelque mv, pour régler ce problème nous avons utilisé un amplificateur LM358 pour amplifier la tension de sortie.

Déterminant quelques paramètres pour la réalisation de capture :

Nous avons :

$$U_{shunt} = R_{shunt} * I_e \Rightarrow U_{shunt} = 0.01 * 0.6$$

$$\Rightarrow U_{shunt} = 0.006V$$

$$U = U_{shunt} * \left(1 + \frac{R_2}{R_1}\right) \Rightarrow U = R_{shunt} * I_{shunt} * \left(1 + \frac{R_2}{R_1}\right)$$

Donc : $U=17V$, $U_{shunt} = 0.06V$. Reste à déterminer R_1 et R_2 .

Pour déterminer R_1 et R_2 , il se fit de fixé une résistance et déterminer la deuxième. Dans notre cas nous avons fixé R_1 à $0.001K\Omega$ et nous avons déterminé R_2 comme suite :

$$U = U_{shunt} * \left(1 + \frac{R_2}{R_1}\right) \Rightarrow U - U_{shunt} = U_{shunt} * R_2/R_1$$

$$\Rightarrow R_2 = (U - U_{shunt}) * R_1 / U_{shunt}$$

$$\Rightarrow R_2 = (17-0.006)*0.001 / 0.006$$

$$\Rightarrow R_2 = 2.83 \text{ K}\Omega$$

Après la détermination de ces paramètres nous avons réalisé le capteur de courant pour PV. La figure IV.5 illustre ce capteur.

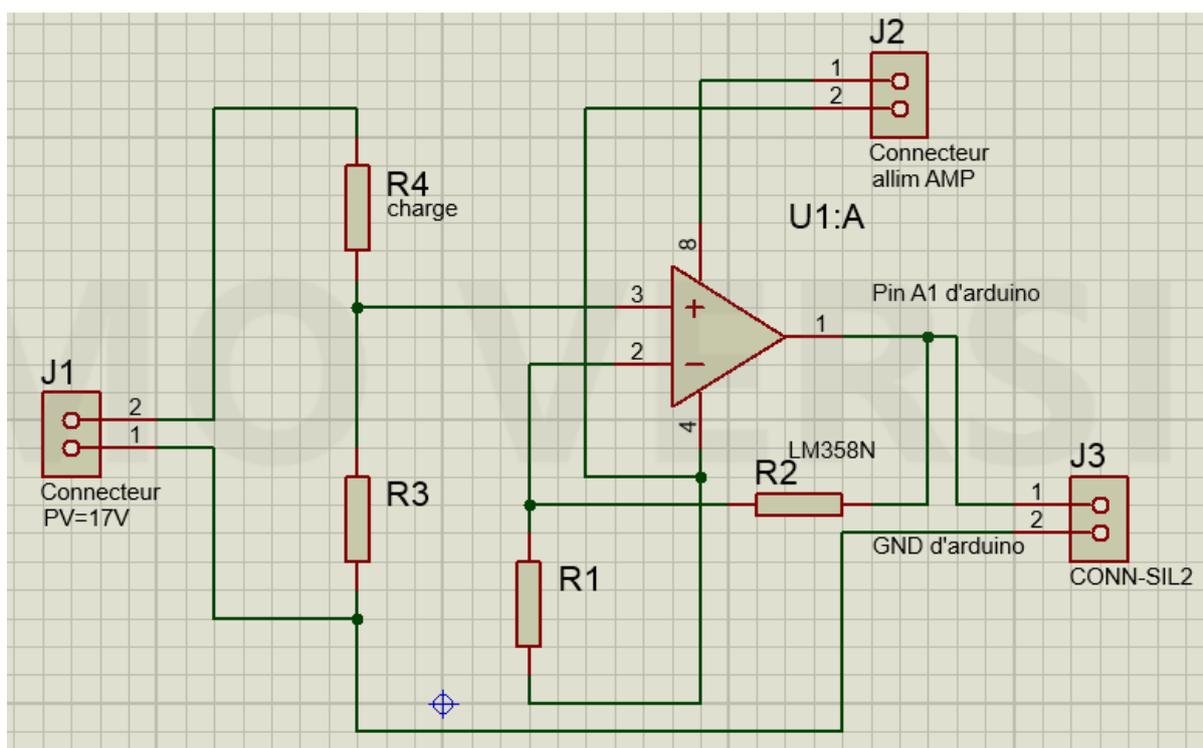


Figure IV.5 Schéma de réalisation d'un capteur de courant pour PV

IV.2.4.4 Capteur de tension de la batterie

Dans ce cas aussi on utilise la même méthode que le panneau, alors on utilise le diviseur de tension :

$$V_s = \frac{R_4}{R_3 + R_4} * V_e$$

- V_e : tension d'entrée
- V_s : tension de sortie

Tel que :

$$V_e = 17V \text{ et } V_s = 5V$$

Nous avons fixé R_2 à $10K\Omega$ et on détermine R_1 comme suite :

$$V_s \cdot (R_1 + R_2) = R_2 \cdot V_e \Rightarrow R_1 \cdot V_s + R_2 \cdot V_s = R_2 \cdot V_e$$

$$\Rightarrow R_1 = R_2 \cdot (V_e / V_s - 1)$$

AN: $R_1 = 10 \cdot (\frac{12}{5} - 1) = 14 \text{ K}\Omega$

La figure IV.6 nous montre le schéma de réalisation le capteur de tension

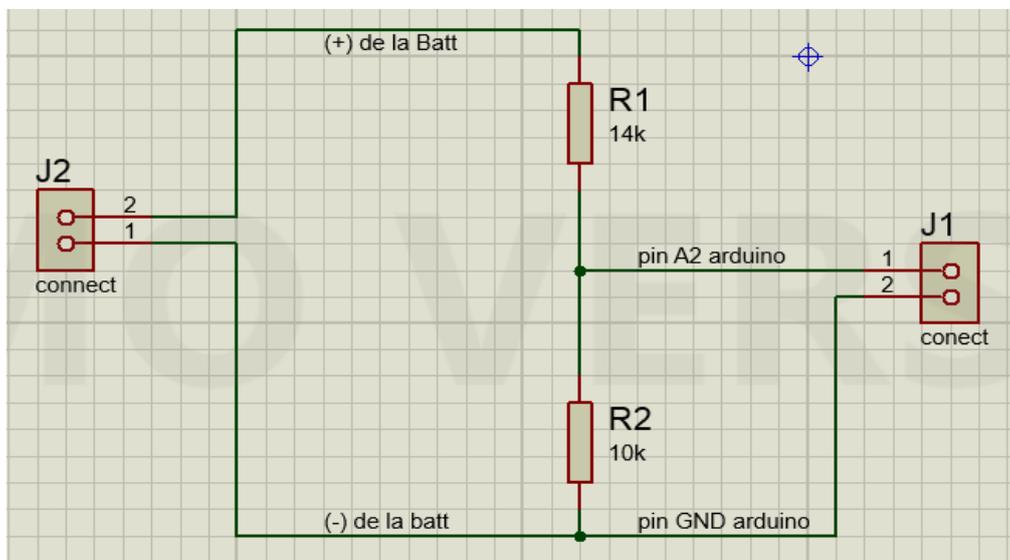


Figure IV.6 Schéma de capteur de tension de Batterie

IV.2.4.5 Capteur de courant de la batterie

Pour mesurer le courant de la batterie on a utilisé le capteur de courant, et pour cela on va déterminer les paramètres suivants :

Alors nous avons :

$$U_{shunt} = R_{shunt} \cdot I_e \Rightarrow U_{shunt} = 0.01 \cdot 4.5$$

$$\Rightarrow U_{shunt} = 0.045V$$

$$U = U_{shunt} \cdot (1 + \frac{R_2}{R_1}) \Rightarrow U = R_{shunt} \cdot I_{shunt} \cdot (1 + \frac{R_2}{R_1})$$

Donc : $U = 12V$, $U_{shunt} = 0.045V$. Reste à déterminer R_1 et R_2 .

Pour déterminer R_1 et R_2 , il se fit de fixé une résistance et déterminer la deuxième. Dans notre cas nous avons fixé R_1 à $0.001K\Omega$ et nous avons déterminé R_2 comme suite :

$$U = U_{shunt} \cdot (1 + \frac{R_2}{R_1}) \Rightarrow U - U_{shunt} = U_{shunt} \cdot R_2 / R_1$$

$$\Rightarrow R_2 = (U - U_{shunt}) \cdot R_1 / U_{shunt}$$

$$\Rightarrow R_2 = (12 - 0.045) * 0.001 / 0.045$$

$$\Rightarrow R_2 = 0.265 \text{ K}\Omega$$

La figure IV.7 nous montre le schéma de réalisation de ce capteur

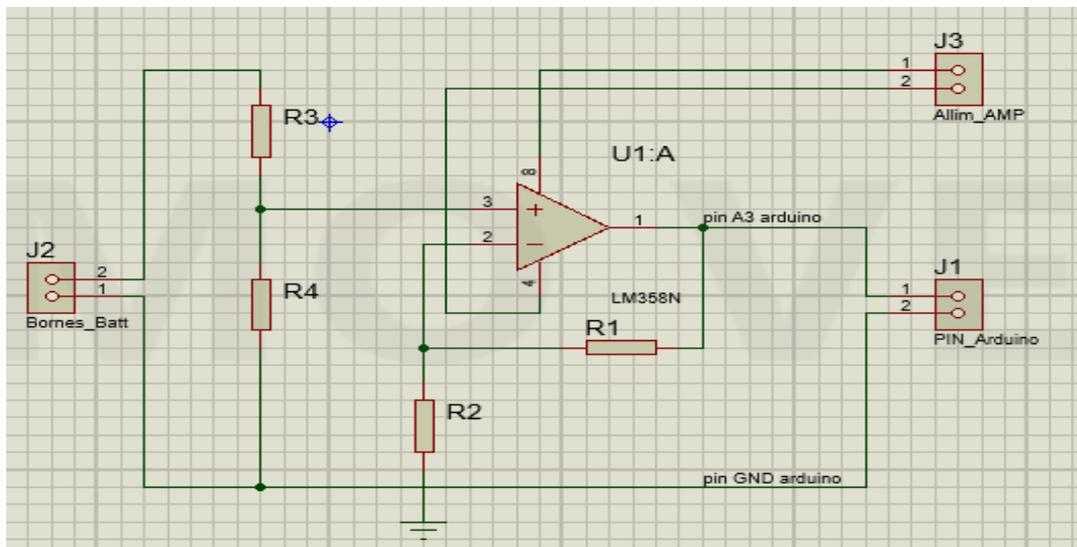


Figure IV.7 Schéma de capteur de courant Batterie

La figure IV.8 nous montre le schéma global de la carte sur ISIS

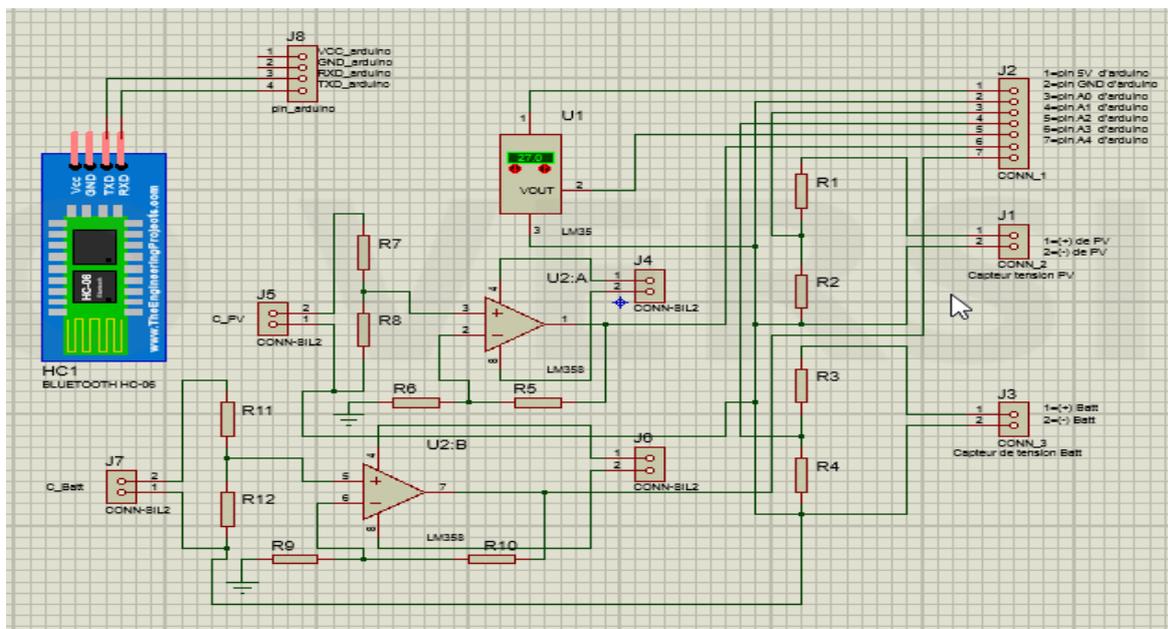


Figure IV.8 Schéma global de la carte sur ISIS

La figure IV.9 nous montre le schéma global de la carte sur ARSS

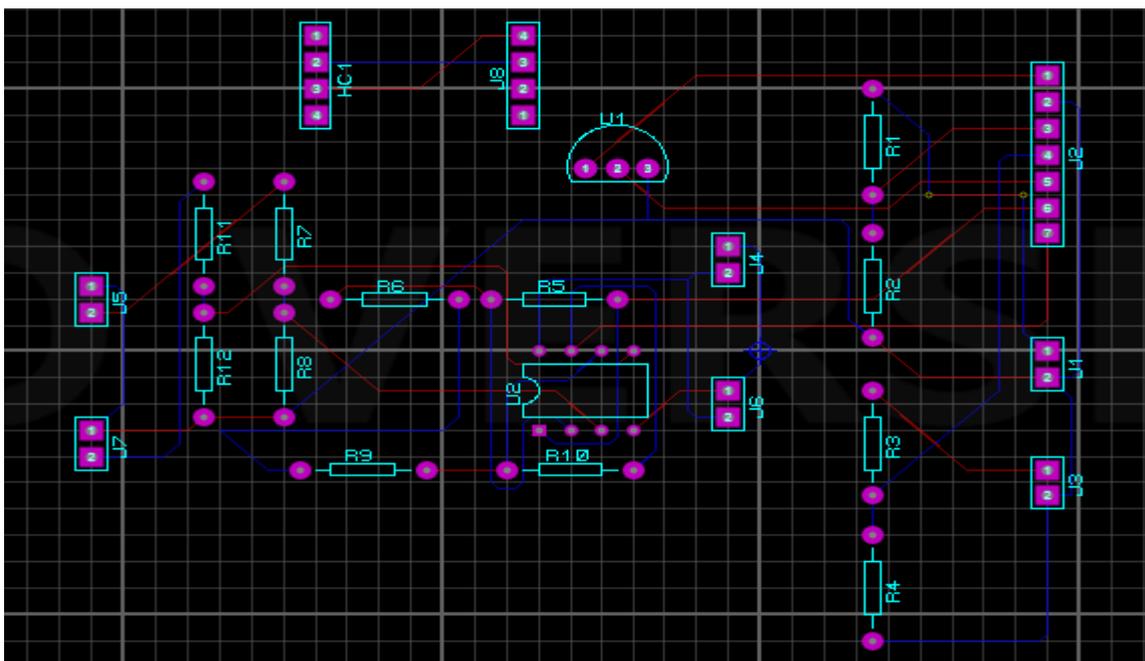


Figure IV.9 Schéma global de la carte sur ARSS

La figure IV.11 nous montre le plan global de la carte sur ISIS

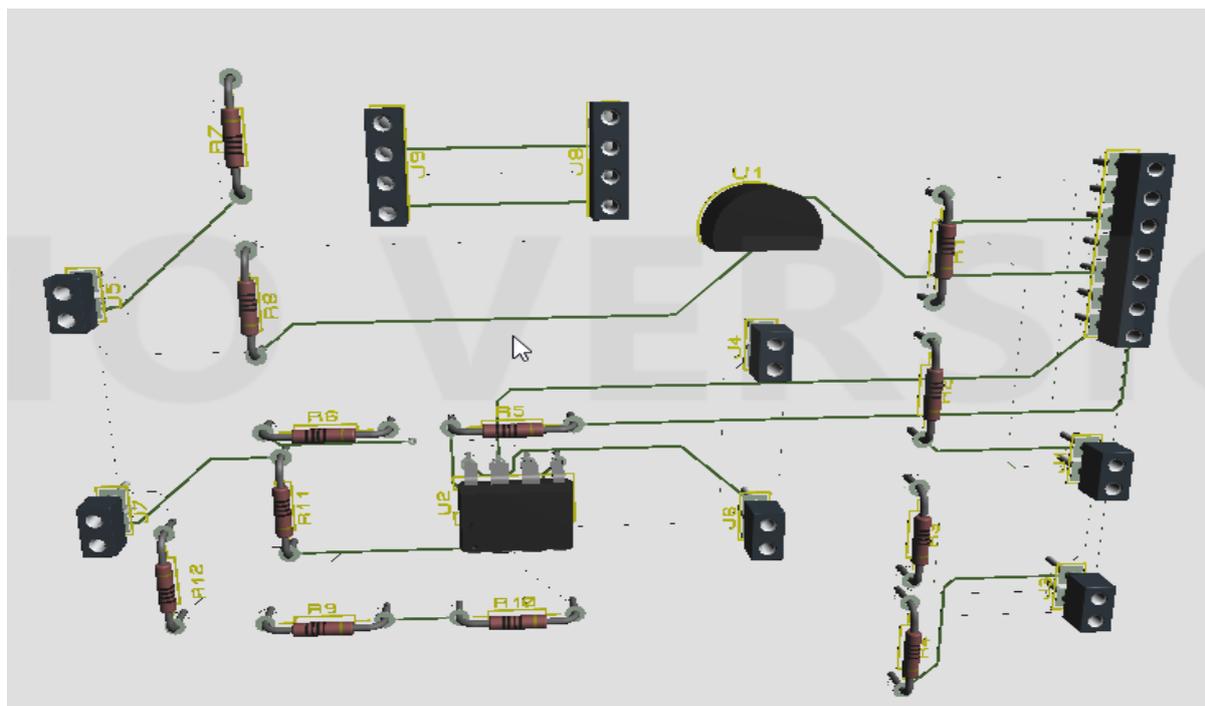


Figure IV.10 Plan global de la carte sur ISIS

IV.2.5 Programmation de la carte arduino

Nous avons utilisé cinq capteurs décrètent dans la partie précédente, ces capteurs vont transmettre des données vers la carte arduino uno, et cette dernière va transmettre aussi ces données vers l'ordinateur (Matlab) via Bluetooth, mais après de les avoir traiter ; comme nous avons déclaré précédemment que la carte arduino ne peut accepter dans ses entrées analogiques que de 0V à 5V.

Pour régler ce problème nous avons programmé la carte arduino de façon à recevoir dans ces entrées analogiques que des données de 0V à 5V, lire ces données, faire des calculs et d'envoyer une valeur réelle selon des formules que nous avons déjà cité ci-dessus, nous allons prendre un exemple, la tension de panneau (PV) pour bien expliquer, la valeur à transmettre d'arduino à Matlab via Bluetooth est :

$$V_r = V_m * 5/n , \text{ tel que}$$

V_r: valeur réelle à transmettre d'arduino à l'ordinateur.

V_m: valeur de la tension mesurée à l'instant 't'.

n: est le nombre de bits.

$$n = 2^{10} = 1024.$$

Pour envoyer des données d'arduino à l'ordinateur, nous avons utilisé Bluetooth, la communication de ce dernier avec la carte arduino est série par contre la communication Bluetooth avec l'ordinateur est sans fil. Nous avons fait un programme sur Matlab qui permet cette communication à l'aide de deux fonction `fopen()` et `fscanf()`, tel que la fonction `fopen()` permet à la carte d'arduino de se connecter avec Matlab via Bluetooth et la fonction `fscanf()` permet de recevoir des données de la carte arduino via Bluetooth.

IV.2.6 L'interface graphique de Matlab

Notre objectif d'utiliser Matlab est de visualiser instantanément les données envoyer par arduino via Bluetooth, donc cela était fait avec trois essais différents c.-à-d. avec la variation des courants et les tensions du panneau et de batterie, alors nous avons relevé trois cas différents, les résultats et les figure suivantes nous montrent ses résultats :

Premier cas :

Tension de panneau solaire = 15V, courant de panneau = 0,55A

Tension batterie(V_{pv}) = 12V, courant de batterie(I_{batt}) = 4,5A

La puissance de panneau(P_{pv}) = 8,25 watt

Le rendement de panneau = 0,82

La température = 22°

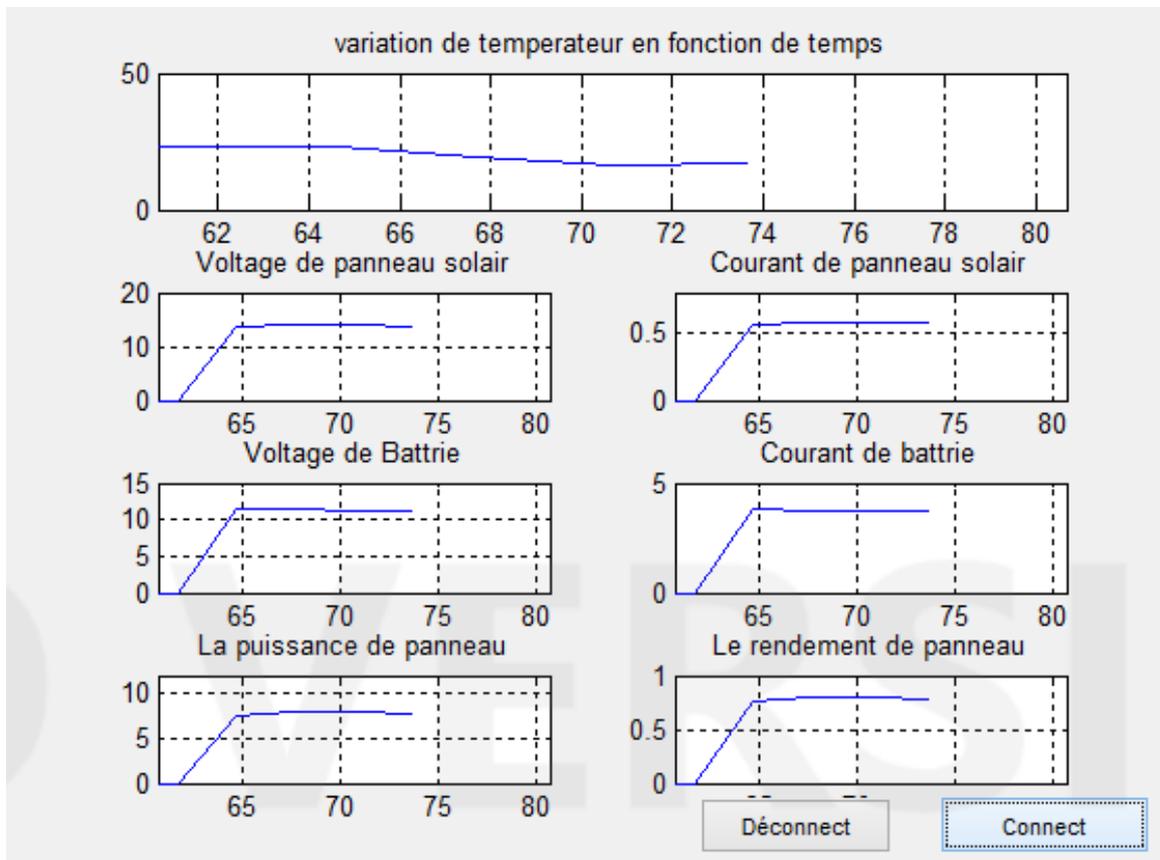


Figure IV.11 Résultat de simulation pour le premier essai

Deuxième cas :

Tension de panneau solaire = 7V, courant de panneau = 0.3A

Tension batterie(V_{pv}) = 4V , courant de batterie(I_{batt}) = 2A

La puissance de panneau(P_{pv}) = 2,1 watt

Le rendement de panneau = 0,21

La température = 22°

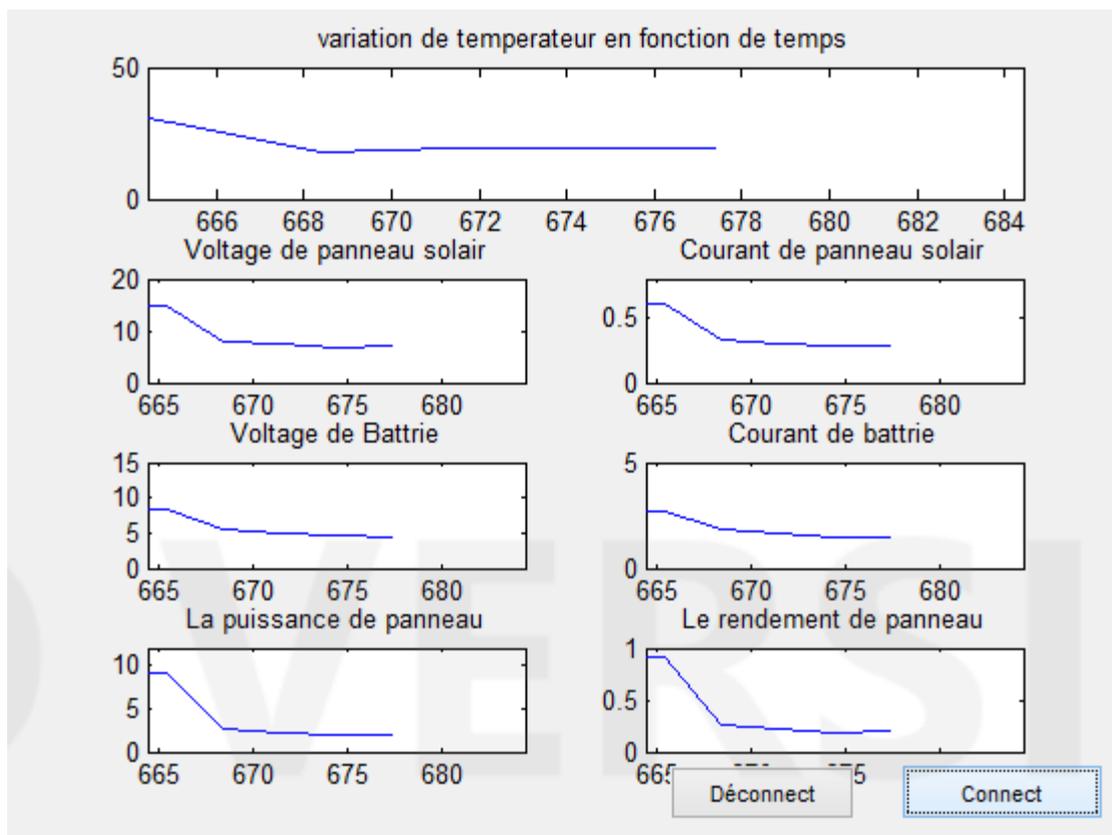


Figure IV.12 Résultat de simulation pour le deuxième essai

Troisième cas :

Tension de panneau solaire = 2V, courant de panneau = 0.1A

Tension batterie(V_{pv}) = 2V, courant de batterie(I_{batt}) = 1A

La puissance de panneau(P_{pv}) = 0,2 watt

Le rendement de panneau = 0,02

La température = 25°

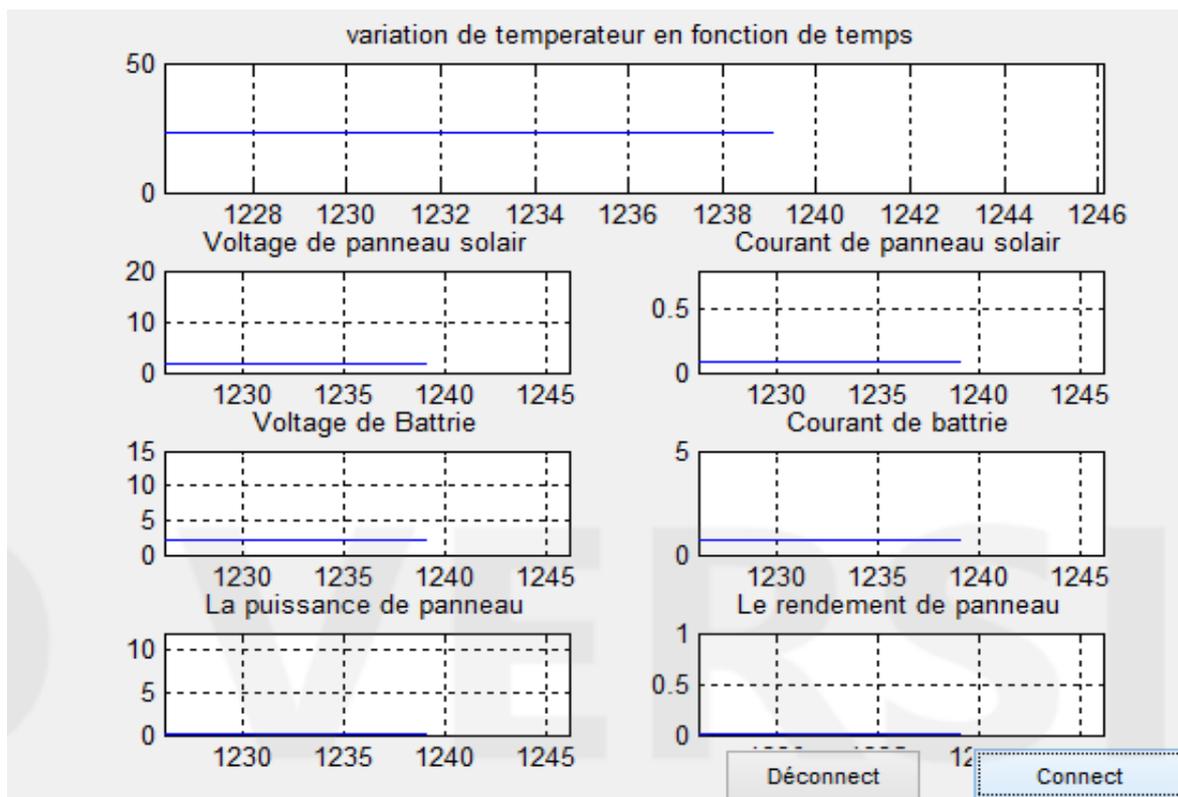


Figure IV.13 Résultat de simulation pour le troisième essai

IV.3 Conclusion

Ce chapitre est consacré à la présentation de la réalisation matérielle de notre système d'acquisition. Dans le premier temps c'est la présentation de matériels utilisés comme le panneau solaire et la batterie ainsi le Bluetooth et après les différents capteurs utilisés avec leurs caractéristiques et leurs schémas de réalisation.

La phase finale de ce travail c'est la réalisation complète avec l'utilisation de matériels et logiciels (Arduino UNO, Matlab) et la présentation des valeurs mesurées affichées par l'interface graphique de Matlab.

Conclusion générale :

La demande mondiale en énergie évolue rapidement et les ressources naturelles de l'énergie telles que l'uranium, le gaz et pétrole diminuent en raison d'une grande diffusion et développement de l'industrie ces dernières années. Pour couvrir les besoins en énergie, des recherches sont conduites à l'énergie renouvelable. Une des énergies renouvelables qui accomplir la demande est l'énergie solaire photovoltaïque, c'est une énergie propre, silencieuse, disponible et gratuite. C'est d'ailleurs ce qui explique que son utilisation connaît une croissance significative dans le monde.

Le travail effectué dans ce mémoire porte sur la réalisation d'un système embarqué d'acquisition de données pour l'analyse et le diagnostic d'un système photovoltaïque.

Dans le premier lieu notre objectif est de mesurer cinq grandeurs (V_{pv} , I_{pv} , V_{batt} , I_{batt} et la température), et les transmettre de la carte arduino à un PC via Bluetooth.

Ensuite nous avons réalisé une carte électronique qui contient les éléments suivants : capteur de courant, capteur de tension, capteur de température et un module Bluetooth. Pour le dimensionnement nous avons pris en considération les caractéristiques de panneau PV et de la batterie et carte arduino. Après la finalisation de cette réalisation nous avons passé à la simulation de programme et après trois essais nous avons obtenu des résultats différents, sont un peu près des résultats réels voulu cette différence revient à la perturbation de la tension de sortie. C'était prévu d'imprimer et de réaliser une carte finale mais le facteur de temps ne nous a pas laissé de finir avec ceci, donc tous les résultats obtenus sont à partir d'une plaque d'essai ordinaire.

Finalement les perspectives que nous pouvons proposer pour l'amélioration et l'évaluation des performances de système, une commande simple et efficace peut agir et apporter des améliorations, cette commande est la commande MPPT son rôle est d'amener la puissance obtenue et la stabiliser à un point de puissance maximale, plusieurs solutions ont été proposées pour l'algorithme de recherche de point de puissance maximale (PPM), commandant le convertisseur statique, donc nous citons quelques méthodes utilisées :

La méthode PetO (perturbation et observation), la méthode Hill Climbing, la méthode d'Incrément de conductance.

Références bibliographiques:

[1] : A.Houssein, N.Heraud, I.Souleiman, G.Pellet, "Monitoring and fault diagnosis of photovoltaic panels", EFEEA'10 International Symposium on Environment Friendly Energies in Electrical Applications, Ghardaïa, Algeria, 2-4 November 2010.

[2] : www.photovoltaique.guidenr.fr/cours-photovoltaique-zutonne.

[3] : www.ulfg.ul.udn.lb/final_year_project/final_year_project_file.

[4] : www.voliculteurs.fr/photovoltaique/particulier_energie

[6] : www.transtech-africa.de/fr/photovoltaique

[7] Antonio Luque and Steven Hegedus, "Handbook of Photovoltaic Science and Engineering", JohnWiley & Sons Ltd, 2003

[8] : M.BOUKLI.HACEN omar « conception et réalisation d'un générateur photovoltaïque muni d'un convertisseur MPTT pour une meilleure gestion énergétique. Thèse de magistère Décembre 2011

[9] : <http://www.ni.com/data-acquisition/what-is/f/>