

II .1/ Introduction

Après qu'on vous a décrit le principe et la systématique de la localisation, nous allons maintenant entamer la partie électronique qui va commander tout le système. Commençons tout d'abord par une description théorique des différents composants qu'on a utilisés pour pouvoir réaliser notre travail.

II.2/ ETUDE DES CAPTEURS

Les capteurs assurent l'interface de mesure entre le système de capteur. Utilisant pour les phénomènes physiques les plus variés pour mesurer des paramètres analogiques.

Un capteur utilisé pour une application quelconque doit être relatif aux grandeurs caractérisant ce système : tension, courant, température et éclairement.

Certains de ces capteurs délivrent de très faible tension de l'ordre de quelques millivolts et il faut prendre les précautions nécessaires pour que leurs connexions avec l'unité de traitement n'introduisent pas d'erreurs inacceptables.

En effet ces capteurs ont une importance dans le système d'acquisition puisque la précision en dépend en grande partie.

II.2.1/CAPTEUR DE TENSION :

La tension maximale délivrée par le champ photovoltaïque est de l'ordre de 200V. Cependant, avec une telle valeur de tension, on ne peut pas attaquer directement l'entrée du capteur, d'où la nécessité de mettre un diviseur de tension. [15]

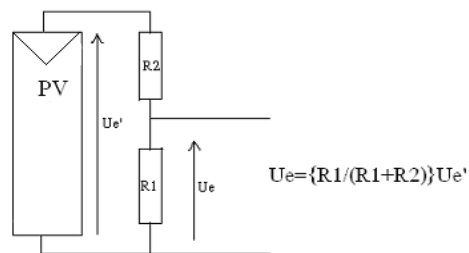


Figure (II.1) diviseur de tension

II.2.2/CAPTEUR D'ECLAIREMENT

Pour une cellule photovoltaïque le courant de court-circuit I_{cc} est proportionnel à l'éclairement E .

$I_{cc} = K.E$ connaissant K on pourra déterminer l'éclairement E correspondant à un courant de court-circuit. $E = I_{cc}/K$

Pour notre système nous avons utilisé une cellule étalant qui délivre une tension voisinant 100mV pour un éclairage de 1000W/m².

Cependant il faut un signal sous forme de tension a l'entré du convertisseur de la carte d'acquisition (≈ 5 V), d'où la nécessité d'amplifier cette tension délivrée par la cellule.



Figure (II .2) cellule photovoltaïque

II.2.3/CAPTEUR DE TEMPERATURE :

La température est appréciée par un simple capteur de type LM35, qui s'alimente entre +4 V et +20 V. Sa sortie délivre une tension de valeur proportionnelle à la température, à raison de 10 mV par degré Celsius, avec pour base la valeur de 0 V à 0 °C. On dispose ainsi d'une tension de 1,0 V pour une température de 100 °C. la relation entre la tension de sortie V_T et la température T est donnée par la formule :

$$V_T \text{ en mV} = (\text{Temp en } ^\circ\text{C}) \times 10 \quad (2-1)$$

Le capteur peut travailler de 0°C à +100°C. La caractéristique correspondant à l'équation (2-1) est donnée à la **Figure II-2**

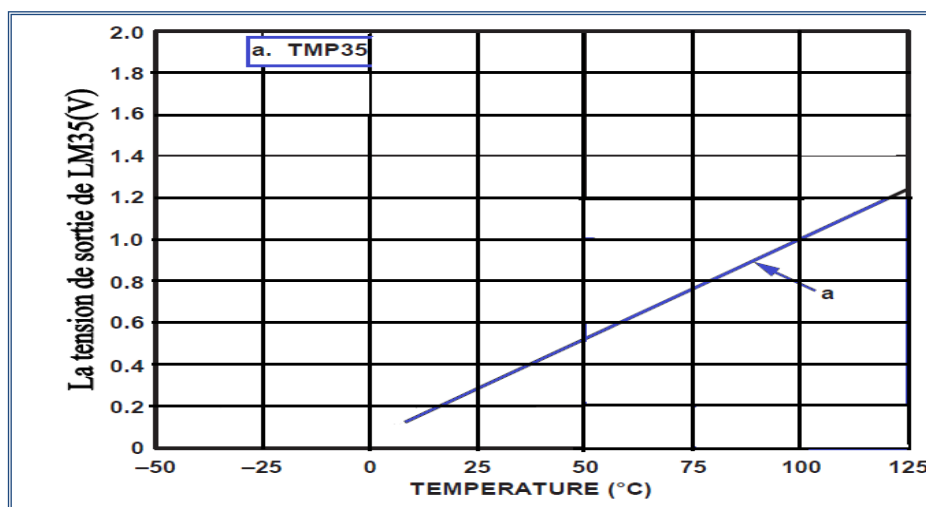


Figure (II-3) : Caractéristique tension-température du capteur LM35

Le circuit de brochage de la LM35 comprend trois bornes :

- * la borne (1) est connectée à +5V.
- * la borne (2) est la tension de sortie proportionnelle à la température
- * la borne (3) est connectée à la masse.[16]

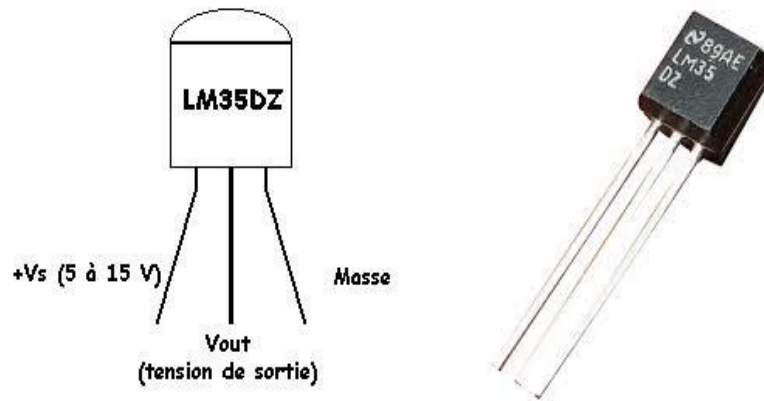


Figure (II.4) Capteur de température LM35

II .2.4/CAPTEUR DE COURANT :

Le principe de mesure de courant est de convertir ce dernier, capté à la sortie du champ photovoltaïque par l'intermédiaire d'un capteur à effet Hall, en une tension voisinant de 5V. Ce type de capteur permettant la mesure de courants alternatifs ou continus ainsi que d'impulsions.

Un composant un peu particulier va être utilisé.

L' ACS712 est un capteur d'intensité de courant.

Ce shield trouvé sur le Net est un capteur de courant AC, mais aussi DC

Ce capteur à effet Hall lit aussi bien l'intensité positive que négative

La valeur 0 est donc la demi-mesure de la tension d'alimentation...

Le montage étant alimenté en 5V, la valeur 0 sera de 2.5V.

- I_{max} (A) évoluera de 2.5V vers le 0V

+ I_{max} (A) évoluera de 2.5V vers le 5V.

Pour un modèle de 20A, la résolution sera donc de $2.5 \text{ (V)} / 20 \text{ (A)}$ soit 0.125A par pas.

C'est on ne peut plus honorable pour notre usage. [16]

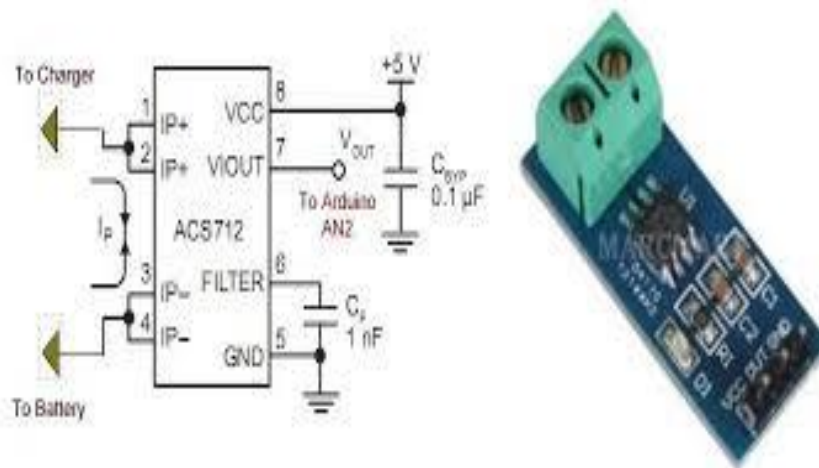


Figure (II.5) capteur à effet Hall ACS712

II .3/ ETUDE DE LA CHARGE ELECTRONIQUE :

Pour relever la caractéristique I-V d'un champ PV, on branche ce dernier à un condensateur en série avec un thyristor **Th**, et en parallèle avec une résistance **Rd** (voir **Figure II.5**).

II.3.1/Principes de fonctionnement

Le microcontrôleur envoie une impulsion tantôt sur la gâchette du thyristor pour assurer la charge du condensateur, tantôt sur le relais pour faire la décharge du condensateur dans la résistance **Rd**.

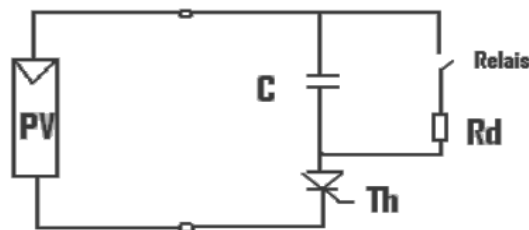


Figure (II.6) principe de fonctionnement

II.2.2 Charge :

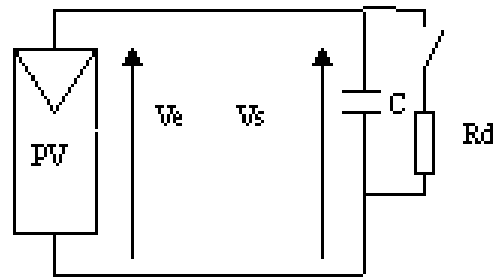


Figure (II.7) charge du condensateur

$$U_s = U_e (1 - e^{-t/T})$$

avec

U_e Tension de charge en volt

$$T = R_0 \cdot C$$

U_s Tension aux bornes du condensateur en volt

t Temps de charge en seconde.

T Constante de temps du circuit en seconde

R_0 le rapport de V_{co}/I_{cc} exprimé en Ohm.

C Capacité du condensateur en farad .

e la base du logarithme népérien ($\approx 2,718$) .

On a que $V_{co} \approx 200 \text{ V}$, $I_{cc} \approx 20 \text{ A}$ donc $R_0 \approx 10 \text{ k}\Omega$

On a que le temps de charge est $t = 10 \text{ ms}$ donc $C \approx 1000 \text{ }\mu\text{f} / 200 \text{ V}$

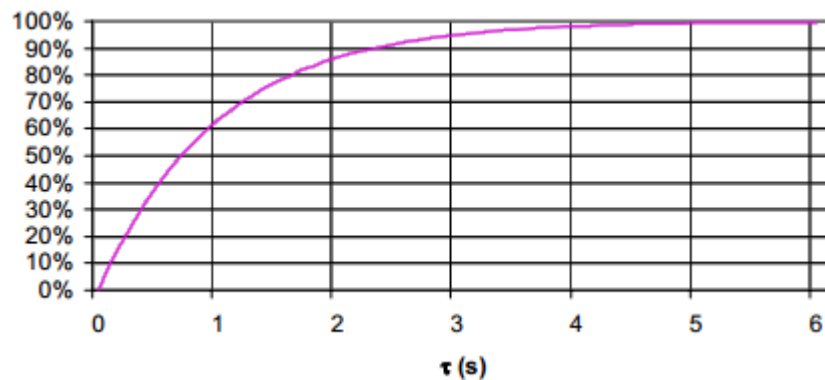


Figure (II.8) courbe charge du condensateur

II.3.3 Décharge :

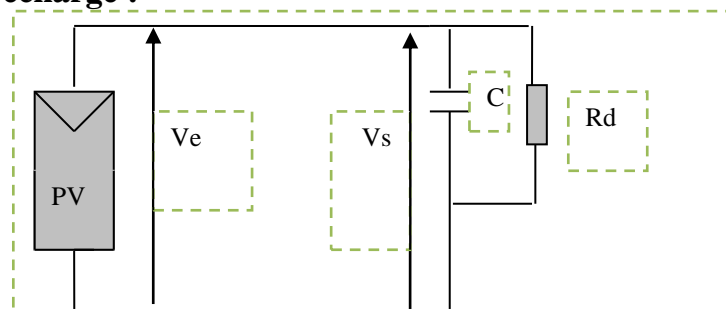


Figure (II.9) decharge du condensateur

$$U_s = U_e \cdot e^{-t/T} \quad \text{avec}$$

$$T = R_d \cdot C$$

U_e Tension de charge en volt

U_s Tension aux bornes du condensateur en volt

t Temps de décharge en seconde.

T Constante de temps du circuit en seconde

R_d la résistance de la décharge en Ohm.

C Capacité du condensateur en farad.

e la base du logarithme népérien ($\approx 2,718$).

On a que le temps de décharge est 20s donc $R_d = 20 \text{ k}\Omega$

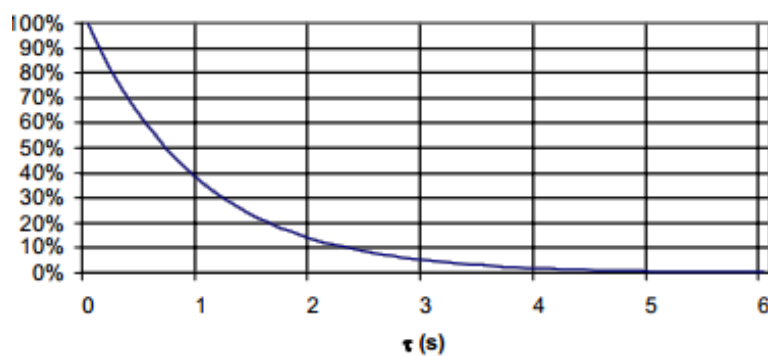


Figure (II.10) courbe decharge du condensateur

Dans notre cas on a utilisé 9 capacités de $C = 1000\mu\text{f} / 63 \text{ V}$ associées entre elles de façon à avoir une charge qui répond à notre besoin ($C = 1000\mu\text{f} / 200\text{V}$)

C_1, C_4 et C_7 sont associées en série donc on aura une somme sur les tensions ($63 \times 3 = 188\text{V}$) avec une division de charges :

$$\frac{1}{C} = \frac{1}{C_1} + \frac{1}{C_4} + \frac{1}{C_7} \quad \text{Donc } C = \frac{C_1}{3} = 1000/3 = 333.333 \mu\text{f} / 188 \text{ V}$$

On a 3 branches en parallèle : donc on aura une conservation de tension (188V) et sommation de capacités : $C_{\text{total}} = 3 \times C = 999.999 \mu\text{f} / 188$

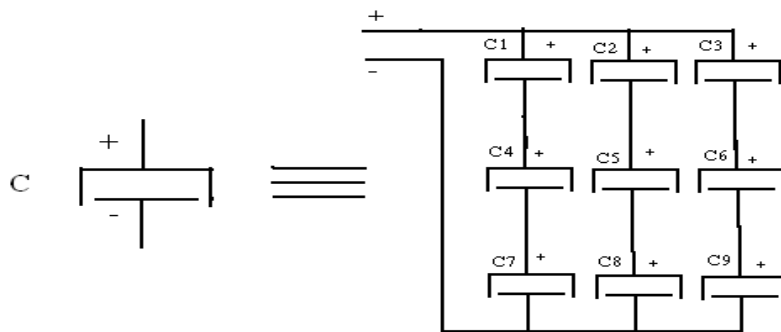


Figure (II.11) branches en parallèle d'un condensateur

II.4/ ATmega328 (ARDUINO):

Dans cette partie on s'intéresse plus par les caractéristiques de microcontrôleur qu'on va choisir, le choix est dépendu de plusieurs facteurs importants selon le besoin...

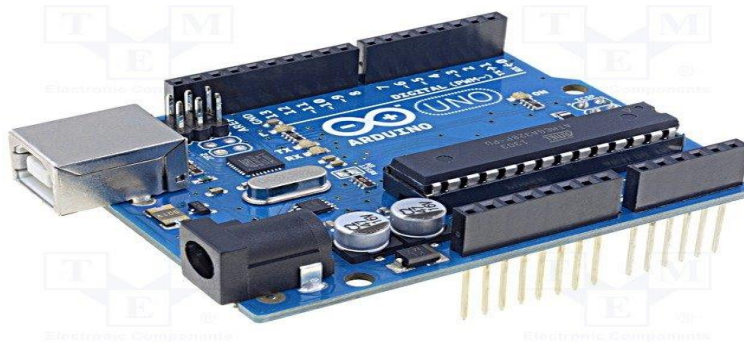
C'est une plate-forme open-source d'électronique programmée qui est basée sur une simple carte à microcontrôleur (de la famille AVR), et un logiciel, véritable environnement de développement intégré, pour écrire, compiler et transférer le programme vers la carte à microcontrôleur.

Arduino peut être utilisé pour développer des objets interactifs, pouvant recevoir des entrées d'une grande variété d'interrupteurs ou de capteurs, et pouvant contrôler une grande variété de moteurs ou toutes autres sorties matérielles. Les cartes Arduino peuvent être autonomes, ou bien ils peuvent communiquer avec des logiciels tournant sur votre ordinateur (tels que LabVIEW, Flash, Processing, MaxMSP, **MekerPlot**). Les cartes électroniques peuvent être fabriquées manuellement ou bien être achetées pré-assemblées, dans notre cas, nous avons acheté une carte Arduino UNO prête.

Le langage de programmation Arduino est une implémentation de Wiring, une plate-forme de développement similaire, qui est basée sur l'environnement multimédia de programmation Processing.

II.4.1/ Pourquoi l'Arduino :

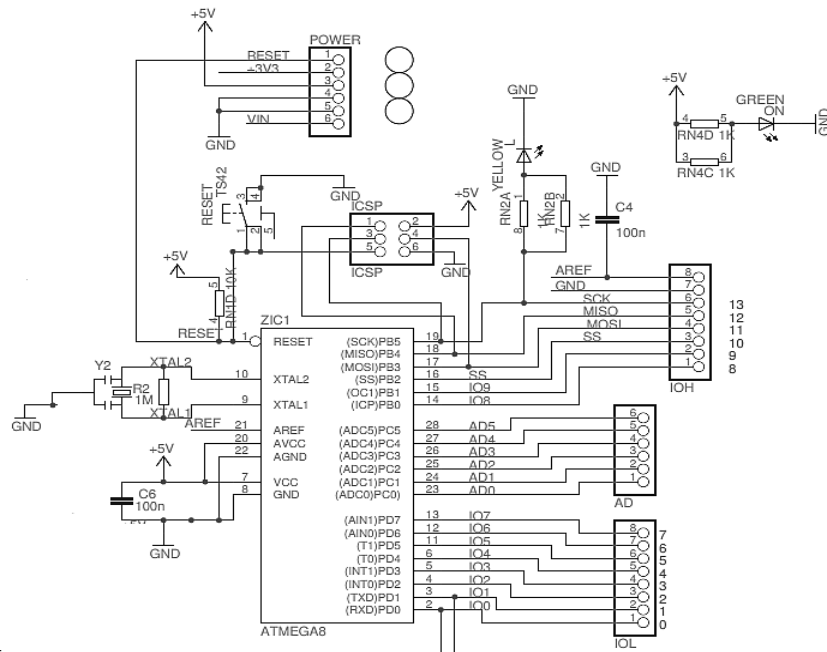
Il y a de nombreux microcontrôleurs et de nombreuses plateformes basées sur des microcontrôleurs disponibles pour l'électronique programmée, qui offrent des fonctionnalités comparables. Tous ces outils prennent en charge les détails compliqués de la programmation des microcontrôleurs et les intègrent dans une présentation facile à utiliser. De la même façon, le système Arduino simplifie la façon de travailler avec les microcontrôleurs, de plus, tourne sous les systèmes d'exploitation Windows, Macintosh et Linux. La plupart des systèmes à microcontrôleurs sont limités à Windows.[17]



Figure(II.12) : La carte Arduino Uno.

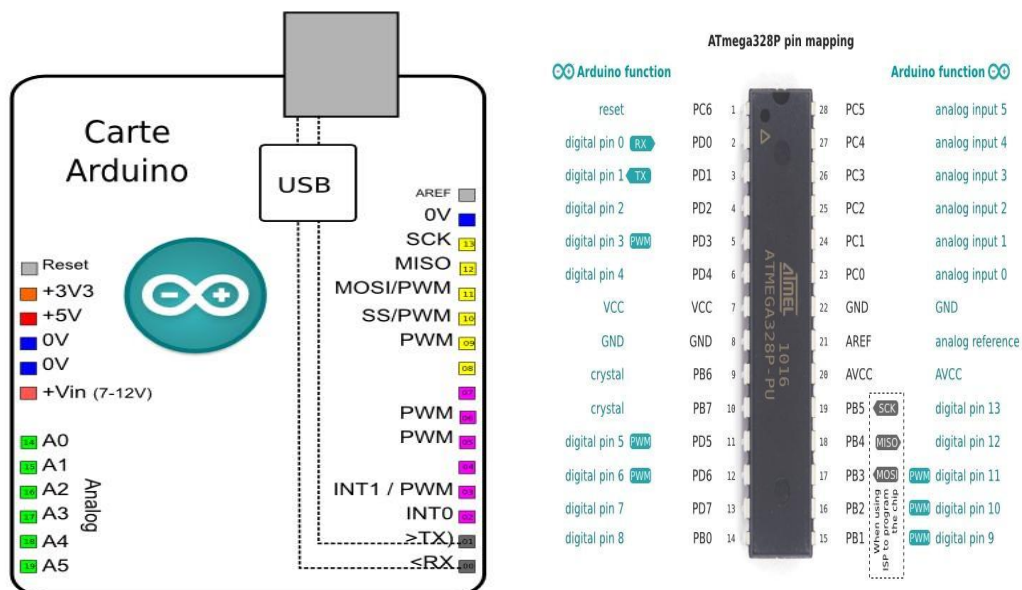
II.4.2/ Schéma simplifié de la carte Arduino UNO

Les signaux d'entrée-sortie du microcontrôleur sont reliés à des connecteurs selon le schéma ci- dessous [18]



Figure(II.13) Schéma simplifié de la carte Arduino UNO

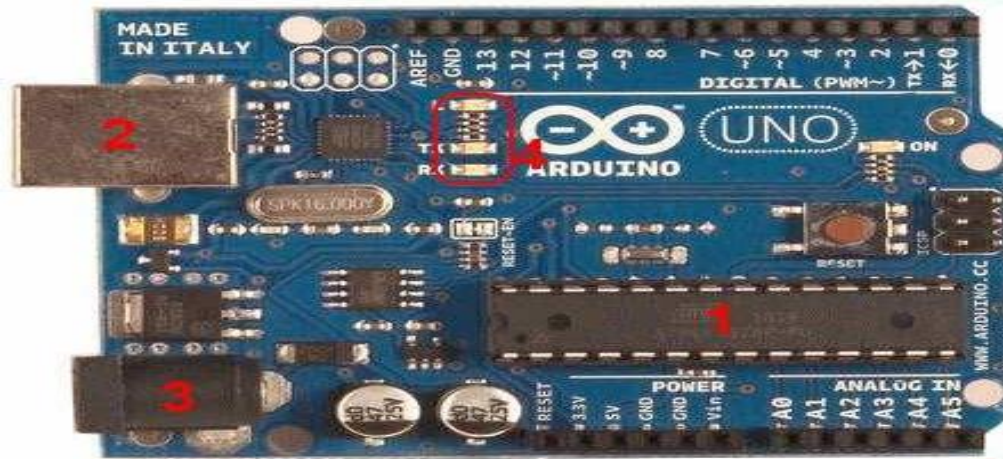
• Brochage de la carte Arduino et « mapping » du microcontrôleur :



Figure(II.14) Brochage de la carte Arduino et « mapping » du microcontrôleur. [18]

II.4.3/Présentation de la carte:

Pour commencer notre découverte de la carte Arduino, je vais vous présenter la carte en elle-même. Nous allons voir comment s'en servir et avec quoi. J'ai représenté en rouge sur cette photo les points importants de la carte.



Figure(II.15) *Présentation de la carte Arduino*

II.4.4/Constitution de la carte

Voyons quels sont ces points importants et à quoi ils servent.

Le micro-contrôleur

Voilà le cerveau de notre carte (en 1). C'est lui qui va recevoir le programme que vous aurez créé et qui va le stocker dans sa mémoire puis l'exécuter. Grâce à ce programme, il va savoir faire des choses, qui peuvent être: faire clignoter une LED, afficher des caractères sur un écran, envoyer des données à un ordinateur ...

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 donc! Veuillez seulement à respecter l'intervalle de 7V à 15V (même si le régulateur peut supporter plus, pas la peine de le retrancher dans ses limites)

Visualisation

Les trois "points blancs" entourés en rouge (4) sont en fait 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

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.

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). Par exemple, on veut connecter une LED sur une sortie du microcontrôleur. Il suffit juste de la connecter, avec une résistance en série, à la carte, sur les fiches de connections de la carte. Cette connectique est importante et a un brochage qu'il faudra respecter. Nous le verrons quand nous apprendrons à faire notre premier programme. C'est avec cette connectique que la carte est "extensible", car l'on peut y brancher tous types de montages et modules ! Par exemple, la carte Arduino Uno peut être étendue avec des shields, comme le « Shield Ethernet » qui permet de connecter cette dernière à internet.

II.4.5/ Les caractéristiques de l'ARDUINO UNO:

Les caractéristiques techniques de l'Arduino.

Microcontrôleur	ATmega328
Tension de fonctionnement	5V
Tension d'alimentation (recommandée)	7-12V
Tension d'alimentation (limites)	6-20V
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM)
Broches d'entrées analogiques	6 (utilisables en broches E/S numériques)
Intensité maxi disponible par broche E/S (5V)	40 mA (ATTENTION : 200mA cumulé pour l'ensemble des broches E/S)
Intensité maxi disponible pour la sortie 3.3V	50 Ma
Intensité maxi disponible pour la sortie 5V	Fonction de l'alimentation utilisée - 500 mA max si port USB utilisé seul
Mémoire Programme Flash	32 KB (ATmega328) dont 0.5 KB sont utilisés par le bootloader (programme de base préprogrammé conçu pour établir la communication entre l'Atmega et le logiciel Arduino)
Mémoire SRAM (mémoire volatile)	2 KB (ATmega328)

Mémoire EEPROM (mémoire non volatile)	1 KB (ATmega328)
Vitesse d'horloge	16 MHz

Tableau (II.1) caractéristiques techniques de l'Arduino [18]

L'Arduino Uno peut être alimenté via la connexion USB ou avec une alimentation externe.

La source d'alimentation est sélectionnée automatiquement.

Le microcontrôleur utilisé sur la carte Arduino UNO est un microcontrôleur ATmega328.

C'est un microcontrôleur ATMEL de la famille AVR

8bits.

Les principales caractéristiques sont :

FLASH = mémoire programme de 32Ko SRAM = données (volatiles) 2Ko

EEPROM = données (non volatiles) 1Ko

Digital I/O (entrées-sorties Tout Ou Rien) =

Timers/Counters : Timer0 et Timer2 (comptage 8 bits), Timer1 (comptage 16bits) Chaque timer peut être utilisé pour générer deux signaux PWM. (6 broches OCxA/OCxB)

Plusieurs broches multi-fonctions : certaines broches peuvent avoir plusieurs fonctions différentes choisies par programmation

PWM = 6 broches OC0A(PD6), OC0B(PD5), OC1A(PB1), OC1B(PB3), OC2A(PB3), OC2B(PD3)

Analog to Digital Converter (résolution 10bits) = 6 entrées multiplexées ADC0(PC0) à ADC5(PC5)

Gestion bus I2C (TWI Two Wire Interface) = le bus est exploité via les broches SDA(PC5)/SCL(PC4).

Port série (USART) = émission/réception série via les broches TXD(PD1)/RXD(PD0)

Comparateur Analogique = broches AIN0(PD6) et AIN1 (PD7) peut déclencher interruption Watchdog Timer programmable.

Gestion d'interruptions (24 sources possibles (cf interrupt vectors)) : en résumé

- Interruptions liées aux entrées INT0 (PD2) et INT1 (PD3)
- Interruptions sur changement d'état des broches PCINT0 à PCINT23
- Interruptions liées aux Timers 0, 1 et 2 (plusieurs causes configurables)
- Interruption liée au comparateur analogique
- Interruption de fin de conversion ADC
- Interruptions du port série USART

- Interruption du bus TWI (I2C) []

II.4.6/ Le convertisseur Analogique-numérique (AD) :

Les microcontrôleurs ATmega328 utilisés avec les cartes Arduino Uno contiennent un convertisseur analogique-numérique (A/N) à 6 voies (A0, A1, A2, A3, A4, A5).

Ce convertisseur a une résolution de 10 bits, renvoyant des résultats de mesure sous forme d'entiers de 0 à 1023 (pour l'acquisition les données analogique). Bien que la fonction principale des broches analogiques pour la plupart des utilisateurs Arduino soit de mesurer la tension de capteurs analogiques, les broches analogiques ont cependant toutes les fonctionnalités des broches numériques d'Entrée/Sortie (identique aux autres broches de la carte Arduino). Cela signifie qu'il est possible de transformer la tension d'entrée entre 0 et 5V en une valeur numérique entière comprise entre 0 et 1023. Il en résulte une résolution (écart entre 2 mesures) de : 5 volts / 1024 intervalles, autrement dit une précision de 0.0049 volts (4.9 mV) par intervalle. [17]

II.5/ Conclusion :

Dans ce chapitre, nous avons présenté les différents capteurs utilisés dans cet projet et la présentation de la carte Arduino UNO