

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université 8Mai 1945 – Guelma
Faculté des sciences et de la Technologie
Département d'Electronique et Télécommunications



**Mémoire de fin d'étude
Pour l'obtention du diplôme de Master2 Académique**

**Domaine : Sciences et Technologie
Filière : Télécommunications
Spécialité : Systèmes de Télécommunications**

SURVEILLANCE DES MACHINES RELIEE PAR UN RESEAU ETHERNET

Présenté par :

Chenaa Zehor & Beldjazia Fatima Zohra

Sous la direction de Mr :

Chouana Oussama

Juin 2017

Remerciement

Avant de commencer de présenté ce mémoire

Tous d'abord, nous remercions ALLAH tout le puissant de nous avoir donné le courage et la volonté de mener à terme ce travail.

Ce travail, ne serait pas aussi riche et n'aurait pas pu avoir nos valseureux sans l'aide de l'encadreur Mr Chouana.O, nous le remercions pour la qualité de son encadrement exceptionnel, et nous lui somme très reconnaissant pour son aide précieuse pendant les moments difficiles et d'avoir approfondi nos connaissances aussi bien pratiques que théoriques dans le domaine électronique et industriel.

Nous adresse nos remerciements aussi à Mr Himeur.Ch, pour son suivie, son aide dans le domaine informatique.

Nous remercions aussi également tous les professeurs pendant notre cursus universitaire, sans oublier le soutien de nos parents et amis/amies Pour tous ceux qui nous sont chers.

Enfin, Nous formulons toutes notre haute considération envers les respectables membres du jury de soutenance qui auront à se prononcer sur ce modeste travail....

Merci à vous tous.



Dédicace

A ma très chère mère ;

Tu m'as donné la vie, la tendresse et le courage pour réussir.

Tout ce que je peux t'offrir ne pourra exprimer l'amour et la reconnaissance que je te porte.

Et témoignage, je t'offre ce modeste travail pour te remercier pour les sacrifices et pour l'affection dont tu m'as toujours entourée.

A mon très cher père ;

L'épaule solide, l'œil attentif compréhensif et la personne la plus digne de mon estime et de mon respect.

Aucune dédicace ne saurait exprimer mes sentiments, que dieu te préserve et te procure santé et longue vie.

Ames très chères sœurs ; Ames très chers frères ;

Aucun mot ne pourrait exprimer ma reconnaissance et ma gratitude pour le soutien et l'amour que vous m'avez cessé de me prodiguer ;

A mon encadreur,

A mes professeurs,

A mes chers Amis/Amies,

Pour tous ceux qui me sont chers ;

A tous ceux qui m'ont aidé de près ou de lions.

Je vous dis merci.



Zehor Chenaa

Dédicac

Je tiens à remercier dieu le tout puissant pour m'avoir donné la chance et le courage afin de m'avoir permis d'atteindre ce stade dans les études et de finir ce modeste travail malgré mes obligations d'étudiant

Je tiens à exprimer toute ma reconnaissance à mon Encadreur de mémoire Mr 'CHOUANA OUSSAMA' Je le remercie de m'avoir encadré, orienté, aidé et conseillé.

Je dédie ce mémoire

A mes chers parents qui ont sacrifiés pour moi tous les moyens et les conditions les plus favorables pour réaliser ce travail, aussi pour leurs aides et leurs encouragements à fin de suivre nos études.

Mes chers frères et sœurs qui sont toujours à mon côté et m'encourager

Et a toute la famille 'Beldjazia'.

Enfin, je remercie tous mes Ami(e)s que j'aime, Pour leur sincère amitié et confiance, et à qui je dois ma reconnaissance et mon attachement.

À tous ces intervenants, je présente mes remerciements, mon respect et

Ma gratitude.

*Beldjazia fatima
zohra*

Résumé

La carte IDO est une carte électronique standard à base d'un microcontrôleur AVR, qui nous permet de connecter n'importe quelle machine au réseau à travers une carte Ethernet pour la suivre à distance.

La tisseuse circulaire (métier à tissé) est une application parmi plusieurs qu'on peut faire avec cette carte.

Abstract

IDO card is a standard electronic board containing an AVR microcontroller, which allows any machine to be connected to the network via an Ethernet card for remote monitoring.

The circular loom is one of several applications that can be made with this card.

ملخص

بطاقة IDO هي بطاقة إلكترونية موحدة على أساس المتحكم AVR ، و التي تسمح لنا بتوصيل اي جهاز او آلة بالشبكة من خلال بطاقة ايثرنت لرصد و متابعتها عن بعد.

النول الدائري هو تطبيق من بين العديد من التطبيقات التي يمكن ان تفعل بهذه البطاقة.

SOMMAIRE

INTRODUCTION GENERALE

CHAPITRE I : GENERALITE

1. INTERNET OF THINGS	1
1.1. INTRODUCTION	1
1.2. HISTORIQUE	1
1.3. TENDANCES	1
1.4. FONCTIONNEMENT DES OBJETS CONNECTENT'.....	2
1.5. RESEAUX DE TELECOMMUNICATION DES OBJETS	3
1.5.1. Communication GSM directe.....	3
1.5.2. Utilisation du Bluetooth	3
1.5.3. Réseaux M2M dédiés	3
1.5.4. RESEAUX ETHERNET	3
1.6. QUELQUE DOMAINES D'APPLICATIONS.....	4
1.6.1. L'Ido dans le secteur de la santé	4
1.6.2. L'IdO dans le secteur du transport	4
1.6.3. L'IdO dans le secteur du l'industrie.....	5
1.6.4. L'IdO et l'enseignement.....	5
1.6.5. L'IdO dans le secteur de la sécurité	6
1.7. LES DIFFERENTES TOPOLOGIES DES NORMES RESEAUX DE L'IDO	6
1.8. EN QUOI L'IDO EST-IL IMPORTANT ?.....	7
1.9. CONCLUSION	8
2. MICROCONTROLEUR ET ARDUINO	9
2.1. INTRODUCTION	9
2.2. MICROCONTROLEUR	9
2.2.1. Définition:	9
2.2.2. Les types des microcontrôleurs	9
2.2.3. L'architecture des microcontrôleurs	10

2.2.4.	Domaines d'applications des microcontrôleurs	11
2.3.	ARDUINO	13
2.3.1.	C'est quoi l'Arduino?	13
2.3.2.	Bref Historique de l'Arduino	13
2.3.3.	Matériel Arduino	14
2.3.4.	Les différents types de cartes Arduino	15
2.3.5.	Description de la carte	15
2.3.6.	L'alimentation	16
2.3.7.	Mémoire :	16
2.3.8.	Logiciel Arduino	17
2.3.8.1.	Définition arduino IDE	17
2.3.8.2.	Interface d'arduino IDE	17
2.3.9.	L'environnement de développement IDE de l'ARDUINO	18
2.3.10.	Les avantages de l'Arduino	19
2.4.	CONCLUSION	19
3.	COMMUNICATION TCP/IP (MODULE ETHERNET) :	20
3.1.	INTRODUCTION :	20
3.2.	PRESENTATION ET FONCTIONNEMENT :	20
3.2.1.	Le rôle du protocole IP	20
3.2.2.	Le rôle du protocole TCP	21
3.3.	DESCRIPTION DU MODELE :	21
3.3.1.	Un modèle en 4 couches	21
3.3.1.1	La couche hôte réseau	22
3.3.1.2	La couche internet	22
3.3.1.3	La couche transport	23
3.3.1.4	La couche application	23
3.4.	COMPARAISON AVEC LE MODELE OSI	24
3.5.	LE SCHEMA DU PROTOCOLE TCP/IP :	25
3.6.	A QUOI SERT LE PROTOCOLE TCP/IP ?	26
3.7.	LE MODULE ETHERNET	26
3.7.1.	Introduction	26
3.7.2.	Bref d'historique	27
3.7.3.	Définition	27

3.7.4.	Architecture interne du module Arduino Ethernet.....	28
3.8.	COMPLEMENTARITE ENTRE ETHERNET ET TCP/IP	29
3.9.	CONCLUSION.....	30
4.	INTERFACE GRAPHIQUE (C #SHARPE).....	31
4.1.	INTRODUCTION	31
4.2.	QU'EST-CE QUE LE C# ?	31
4.3.	Bref HISTOIRIQUE DE C Sharpe.....	31
4.4.	LE BUT DE C#.....	32
4.5.	ENVIRONNEMENT DE DEVELOPPEMENT C#.....	32
4.5.1.	Présentation du logiciel.....	32
4.6.	AVANTAGES DE C# ?	34
4.7.	CONCLUSION.....	34

CHAPITRE II : APPLICATION

1.	INTRODUCTION.....	35
2.	DESCRIPTION	35
3.	PARTIE HARDWARE DE LA TISSEUSE CIRCULAIRE.....	39
3.1.	μ C ARDUINO.....	39
3.1.1.	Aperçu	39
3.1.2.	Architecture Interne.....	40
3.2.	LES ENTREES LOGIQUES.....	41
3.2.1.	Alarme.....	41
3.2.2.	Bouton.....	42
3.2.3.	Optocoupleur.....	43
3.2.4.	La détection des coupures de file	44
3.3.	LES SORTIES LOGIQUES.....	44
3.3.1.	Relais.....	45
3.3.2.	LED	45
3.4.	SORTIE ANALOGIQUE.....	47
3.4.1.	« L298 »	47
3.5.	D'autres composants actifs et passifs de la carte.....	48
3.5.1.	Le transistor BC547	48

3.5.2.	La Résistance.....	49
3.5.3.	Le Condensateur.....	49
3.6.	INTERFACE TCP/IP « LE MODULE ETHERNET ».....	51
3.6.1.	Le Module Ethernet.....	51
3.6.2.	Aperçu :	51
3.6.3.	Caractéristiques	53
3.7.	INTERFACE I2C « LCD 16*2).....	53
3.7.1.	Aperçu	53
3.7.2.	Le module LCD.....	53
3.7.3.	Spécifications	54
4.	PARTIE SOFTWARE DE LA TISSEUSE CIRCULAIRE.....	55
4.1.	DESCRIPTION DE L'APPLICATION.....	55
4.2.	DESCRIPTION DE L'ALGORITHME.....	55
4.3.	ORGANIGRAMME.....	56
4.4.	CONCEPTION.....	58
4.4.1.	Interface graphique.....	58
4.5.	IMPLEMENTATION	59
4.5.1.	Langage de programmation.....	59
4.5.2.	L'environnement de développement:.....	59
4.5.3.	Bibliothèque	59
4.5.4.	Système d'exploitation :	60
5.	CONCLUSION.....	60

CHAPITRE III: TEST PRATIQUE ET RESULTAT

1.	INTRODUCTION.....	61
2.	TEST DES ENTREES SORTIES et LA TRANSMISSION DES DONNEES PAR RESEAU	61
2.1.	Vérification des entrées.....	61
2.2.	Vérification des sorties.....	63
2.2.1.	Test bouton Marche.....	64
2.2.2.	Test bouton arrêt.....	64
2.2.3.	Coupure dans la zone 1	65

Sommaire

3 .	TEST DES EVENEMENTS PARALLELE ET LA TRANSMISSION DES DONNEES PAR RESEAU.....	67
3.1.	Cas de coupure de plusieurs zones au même temps	67
3.2.	Simulation de la défaillance du capteur.....	68
4 .	CONCLUSION.....	69

CONCLUSION GENERALE

LISTE DES FIGURES

LISTE DES TABLEAUX

ABRIVIATION

REFERENCE BIBLIOGRAPHIES

ANNEXE

Introduction
Générale

Introduction Générale

Le domaine des communications et en particulier l'IdO (Internet Des Objets connectés) a connu une évolution spectaculaire durant ces dernières années .cette technologie émergente permet de prélever certain données physique capté d'un environnement puis de les transmettre à des points de collecte distants.

Le panorama d'applications de la technologie des IdO est très riche et varié. Les applications dans le domaine industriel représentent un des exemples les plus importants. Un IdO pourrait être déployé dans une usine afin de collecter aisément les informations de rendement, état de la machine, et les alarmes.....etc.

Le travail que nous présentons dans notre projet de fin d'étude porte donc sur la réalisation d'une carte électronique standard, qu'elle va nous permettre de visualiser toutes les informations de la machine. Ce travail a été réalisé au niveau du Groupe ABIDI.

Notre mémoire est organisé de la manière suivante :

- le premier chapitre contient des notions générales sur l'internet des objets et leur domaine d'application, ainsi une description de modèle TCP/IP. Aussi, ce chapitre décrit des généralités sur le microcontrôleur, la carte arduino et le module Ethernet.
- Le seconde chapitre contient trois partie : la première partie concerne à une description sur la tiseuse circulaire, la seconde partie décrit la partie hardware de notre application ainsi les dispositifs utilisés dans la carte électronique. la troisième partie contienne la partie software auquel on a présenté notre propre conception ensuite l'implémentation.
- Le dernier chapitre présente les différents tests effectués et les résultats de la transmission des informations sur l'interface graphique.

CHAPITRE I :

Généralité

1. INTERNET OF THINGS

[L'Internet des objets (IdO / IoT) ou les objets connectés]

1.1 INTRODUCTION

L'Internet des objets repose sur l'idée que tous les objets peuvent être connectés un jour à Internet et sont donc capables d'émettre de l'information et éventuellement de recevoir des commandes. L'Internet des objets propose de créer une continuité entre le monde réel et le monde numérique : il donne une existence aux objets physiques dans le monde numérique. [1]

Cependant nous pouvons considérer que l'Internet des objets consiste à pouvoir trouver tout objets physique (lunettes, montre, clé..) en posant tout simplement une question à votre Smartphone ou une machine qui soit connectée à Internet ; Le monde des objets connectés va décupler les problèmes de sécurité pour ses usagers. [2]

Dans cette partie, nous présentons un aperçu général sur la troisième évolution de l'internet ; et un bref historique du «L'Internet des objets» et leurs fonctionnements ainsi que ses différents domaines d'applications et le futur de l'IdO

1.2 HISTORIQUE

Le terme « Internet of Things » (en Français Internet des Objets) est né en 1999 au centre MIT (Massachusetts Institute of Technology), grâce à Kevin Ashton, un chercheur britannique, pionnier dans son domaine (IdO). Son équipe lança la promotion d'une connectivité ouverte de tous les objets en utilisant les étiquettes RFID (Radio Frequency Identification). Grâce à l'apparition du nouveau protocole IPv6, des secteurs comme l'aéronautique s'emparent rapidement du concept de l'internet des objets, et participent aux recherches. Ce concept de l'Internet des Objets commence à connaître une popularité en 2007. On a envisagé alors de mettre en place un Internet des Objets Global, Ubiquitaire. [3]

1.3 TENDANCES

Depuis son lancement au grand public, le réseau Internet n'a cessé de se développer et ce à une vitesse astronomique. Les applications Internet sont de plus en plus abondantes et adressent plusieurs domaines et secteurs de notre vie quotidienne sociale et économique. Parmi ces applications, il y a l'Internet des Objets (IdO), ou plus souvent IoT (Internet of Things). C'est un domaine en plein vogue, qui est encore flou pour plusieurs même certains qui sont familiers avec le domaine des TICs (Technologies de l'Information et la communication).

L'Internet des Objets désigne la « connexion » d'objets communicants à un réseau plus large, moyennant des technologies de communication telles que le WIFI, le Bluetooth,....

C'est un concept encore jeune et se développe rapidement, sans qu'il a une définition officielle et standardisée. Néanmoins, plusieurs experts du domaine s'alignent sur la définition suivante :

L'Internet des objets est considéré comme la troisième évolution de l'internet, baptisée Web 3.0 qui fait suite succédant à l'ère du Web Social 2.0. Il représente l'extension d'internet à des objets ou des lieux du monde physique. Il désigne la connexion de ces objets à un réseau plus large (Wifi, Bluetooth...) que ce soit par l'intermédiaire des Smartphones ou de manière autonome. [4]

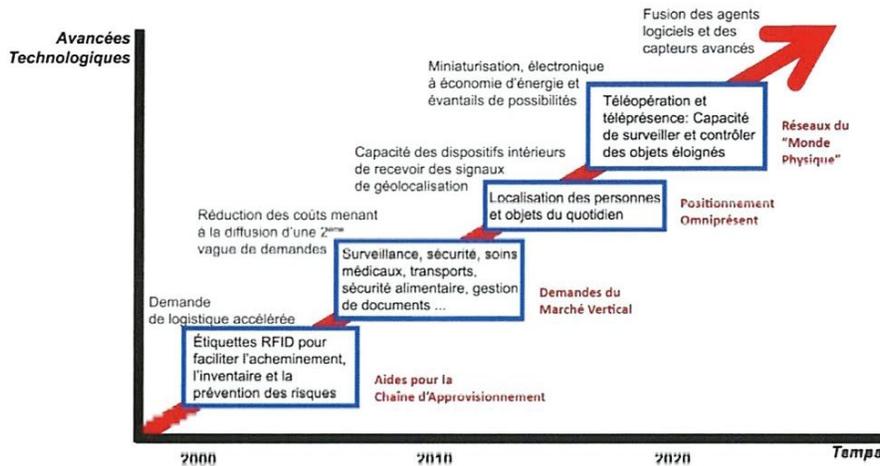


Figure 1.1 : Historique de la Technologie : la connectivité des choses

Le nombre d'objets connectés atteindra 6,4 milliards en 2016, soit 30% de plus qu'en 2015, et atteindra environ 20,8 Milliards d'ici 2020. Les services en relation avec les objets connectés sont estimés à 235 Milliards \$, soit 22% de plus qu'en 2015. Le marché des produits utilisant les objets connectés est estimé à 1,4 Trillion \$ en 2016, soit 19,5 % de plus qu'en 2015, et atteindra 3,01 Trillion \$ en 2020.

1.4 FONCTIONNEMENT DES OBJETS CONNECTÉS

Les objets connectés sont reliés à Internet (on parle d'ailleurs d'Internet des objets, ou web 3.0) : ils peuvent donc communiquer avec d'autres systèmes pour obtenir ou fournir de l'information. Cela est rendu possible par la forte miniaturisation des composants électroniques, mais aussi par l'émergence de nouveaux réseaux de télécommunication de type M2M. Ils pourront par exemple :

- ❖ Collecter et stocker des informations en fonction de leur environnement : rythme cardiaque de l'utilisateur, hygrométrie d'une cave, etc.

- ❖ Déclencher une action en fonction des informations recueillies sur le web, comme par exemple l'arrosage d'une pelouse à la veille d'une forte journée de sécheresse.

1.5 RESEAUX DE TELECOMMUNICATION DES OBJETS

Si certains objets connectés, de par leur nature, peuvent être raccordés à un réseau wifi domestique ou professionnel (par exemple une balance corporelle connectée), de nombreux autres sont destinés à être utilisés partout en mobilité. Une des problématiques va donc être de trouver un moyen de faire communiquer ces objets sans sacrifier l'autonomie énergétique, et pour un coût raisonnable.

1.5.1 Communication GSM directe

La première solution consiste à utiliser les réseaux de données existants (3G, 4G, etc.). Cette solution est très peu adaptée dans la majorité des cas : elle nécessite une carte SIM, et utilise une technologie très consommatrice d'énergie. De plus, elle est relativement onéreuse.

1.5.2 Utilisation du Bluetooth

La seconde solution consiste à se connecter à un Smartphone équipé de Bluetooth pour se servir de la connexion de ce dernier au réseau Internet. Dans ce cas, la consommation énergétique est moindre, mais la dépendance au Smartphone peut être problématique.

1.5.3 Réseaux M2M dédiés

La 3ème, ces dernières années, on a vu apparaître des réseaux de télécommunication dédiés aux objets connectés. Ces derniers ne nécessitent pas de carte SIM et offrent des consommations très réduites.

Ils sont conçus pour transmettre un volume restreint de données (quelques octets). C'est le cas par exemple de Sigfox.[5]

1.5.4 RESEAUX ETHERNET

Le réseau Ethernet est utilisé comme un réseau de communication des objets pour les avantages suivants :

- ✚ Un réseau fiable
- ✚ Connexion permanente, sans interruption
- ✚ Capable d'envoie les donnée pour une grande distance
- ✚ Ethernet est plus rapide et plus sécurisé.
- ✚ Il n'y a pas le problème des obstacles (comme le wifi est Bluetooth)

1.6 QUELQUE DOMAINES D'APPLICATIONS

1.6.1. L'IdO dans le secteur de la santé

L'Internet des Objets apparaît comme une technologie logique pour l'ouverture du monde de la santé, Les accessoires connectés fournissent en direct des informations sur la santé de la personne qui les portes sur elle, contribuant ainsi à alerter les professionnels comme les patients sur les risques éventuels avant qu'ils ne deviennent graves. Des scanners utilisables à domicile permettent de réaliser des auto- diagnostics, ce qui évite une consultation à l'hôpital pour des pathologies bénignes et donne aux établissements une indication plus précise de la nécessité ou non d'un traitement pour le patient.

En effet, dans un hôpital la perte d'une des données à un instant critique peut devenir une question de vie ou de mort, éventuellement. Il n'est donc tout simplement pas acceptable que l'indisponibilité se chiffre en jours ni même en heures.

La disponibilité des données est d'une importance vitale et les pannes, lorsqu'elles surviennent, ne doivent pas durer plus de quelques secondes ou minutes tout au plus avant que l'accès aux données soit rétabli

1.6.2. L'IdO dans le secteur du transport

La voiture connectée est en route. Le marché des transports a déjà anticipé l'arrivée des objets connectés. Parmi les enjeux les plus fréquents que ce domaine fait naître, on retrouve la réduction des accidents et des embouteillages, le partage de voitures, le développement des offres de VTC et de taxis.



Figure I.2 : La voiture connectée est en route

1.6.3. L'IdO dans le secteur de l'industrie

L'Industrie se tourne vers l'usine intelligente et IdO est une chance pour l'industrie. Secteur déjà hautement mécanisé, l'industrie pourrait encore rationaliser une partie de ses ressources. Les objets connectés commencent dès à présent à s'instiller dans les chaînes de production.

L'objectif pour ce secteur est d'optimiser le transport logistique en particulier les flottes de camions, les conteneurs et autres wagons.

=>Diminution des coûts.

- ❖ Maintenance
- ❖ Garantie
- ❖ Productivité

->Gagner des nouveaux clients et des nouveaux produits

- Un objet banale → Un objet haut de gamme.
- Le nombre de clients augmente vu que l'IdO améliore la qualité du produit et joue un rôle dans la compétitivité.

=>L'intégration du client dans la production. Le client peut choisir un produit personnalisé (sous commande) et peut suivre ses états de production. {En cours de construction, à l'étape x, à l'étape y, dans le dépôt, en bateau, chez la boutique...} L'IdO, une chance pour l'industrie.



Figure I.3 : L'Industrie se tourne vers l'usine intelligente

1.6.4. L’IdO et l’enseignement

- ❖ L’IdO peut contribuer massivement au progrès au sein de l’enseignement.
- ❖ L’apprentissage sera alors plus authentique et vivant, dépassant le cadre de la salle de classe.
- ❖ Les établissements ne peuvent plus compter uniquement sur le savoir des professeurs.
- ❖ Les appareils personnels doivent être largement acceptés, et non interdits, pendant les cours. Encourager leur utilisation comme outils favoriser et accélérer l’apprentissage.
- ❖ Grâce au foisonnement des appareils mobiles, les établissements peuvent également collecter les données plus facilement pour interpréter les comportements et les activités des étudiants.
- ❖ Utilisées à bon escient, ces données permettront de cibler l’apprentissage en fonction des besoins, des méthodes d’assimilation et des aspirations des étudiants.

1.6.5. L’IdO dans le secteur de la sécurité

- ❖ Automobile : géolocalisation, contact des secours en cas d’accident, échanges avec les dispositifs de circulation (radars, caméras...), etc.
- ❖ Domotique : suivi en temps réel des dispositifs de chauffage, d’éclairage, de consommation d’eau, de sécurité, etc.
- ❖ Le contrôle parental: Les parents peuvent localiser facilement leurs enfants. Comme ils peuvent contrôler ce qu’il regarde à la télé. [2]

Applications de l’IdO

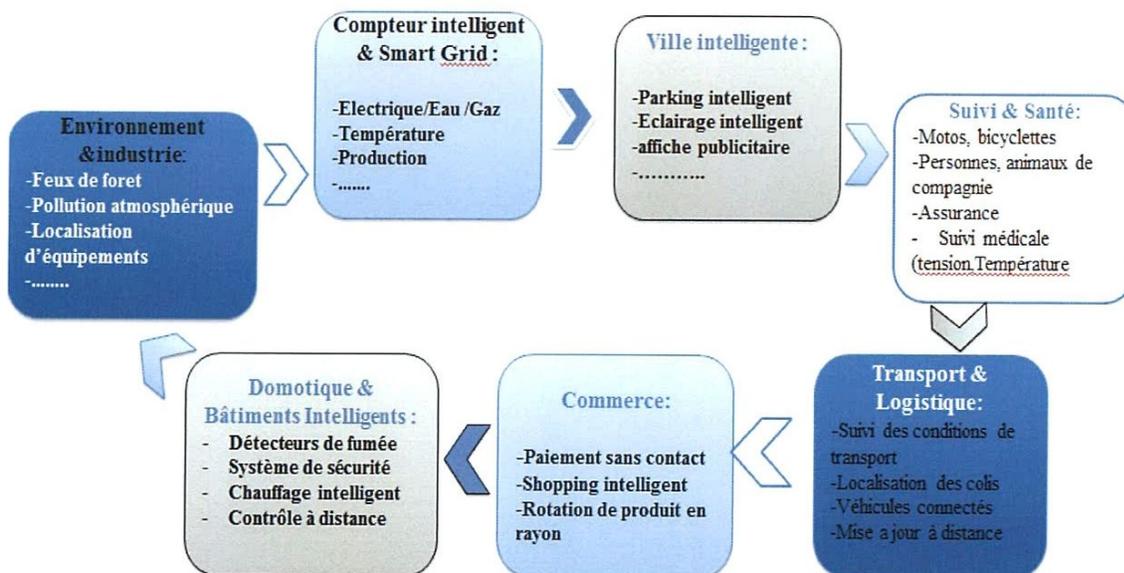


Figure I.4 : Cas et Services de l’Internet des objets

1.7. LES DIFFERENTES TOPOLOGIES DES NORMES RESEAUX DE L'IDO

Les architectures suivantes sont ou ont effectivement été utilisées dans des réseaux informatiques grand public ou d'entreprise. La topologie d'un réseau correspond à son architecture physique. En ce sens où leur structure détermine leur type. [1]

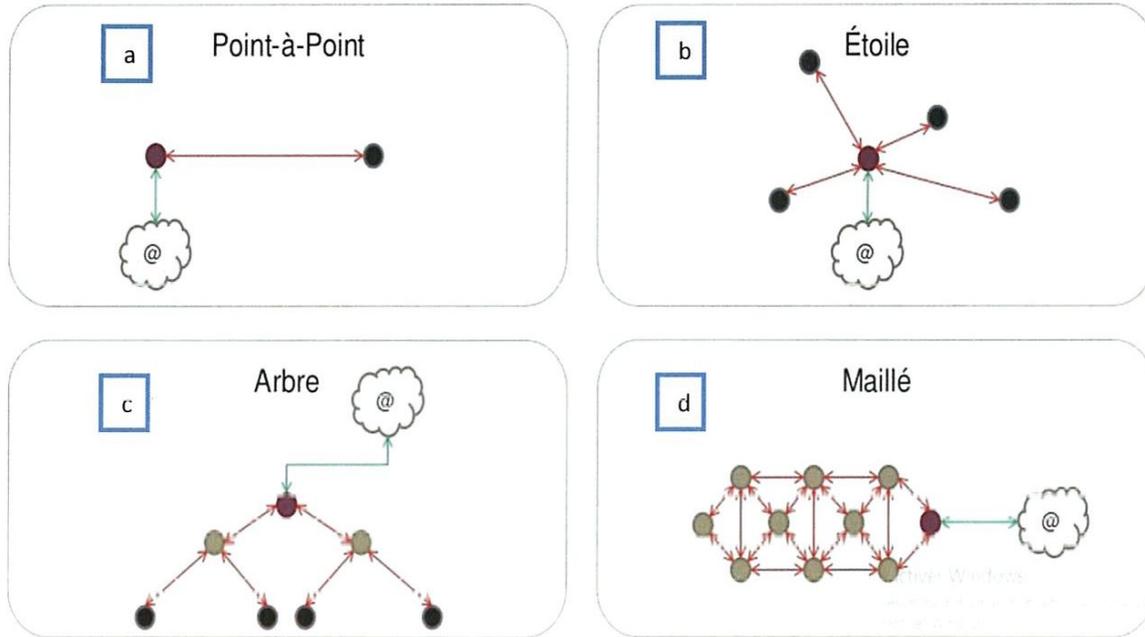


Figure I.5 : les différentes topologies a= Point-à-Point;b= Étoile; c=Arbre ;
 d= Maillé.

Il existe 2 modes de propagation classant ces topologies :

- ✚ Mode de diffusion (par exemple topologie en bus ou en anneau)

Ce mode de fonctionnement consiste à n'utiliser qu'un seul support de transmission. Le principe est que le message est envoyé sur le réseau, ainsi toute unité réseau est capable de voir le message et d'analyser selon l'adresse du destinataire si le message lui est destiné ou non.

- ✚ Mode point à point (par exemple topologie en étoile ou maillée)

Dans ce mode, le support physique ne relie qu'une paire d'unités seulement. Pour que deux unités réseaux communiquent, elles passent obligatoirement par un intermédiaire (le nœud).

1.8. EN QUOI L'IDO EST-IL IMPORTANT ?

Pour pouvoir mesurer l'importance de l'IdO, il faut d'abord comprendre les différences entre l'Internet et le World Wide Web (ou Web), des termes souvent confondus. L'Internet est la couche physique, c'est-à-dire le réseau composé de commutateurs, de routeurs et d'autres équipements.

Sa principale fonction est de transporter les informations d'un point A à un point B de façon rapide, fiable et sécurisée. Le Web, lui, est une couche applicative qui intervient sur l'Internet. Son rôle essentiel est de fournir une interface permettant d'exploiter les informations qui circulent sur l'Internet.

- ❖ Dans 10ans. ...peut être L'IdO, eût une révolution a la puissance dix ! Demain, ces activités qu'on est entrains de faire seront banale.
- ❖ Tout à fait, selon les spécialistes avec les Smartphones puis un nombre croissant d'objets connectés, en dix ans (2015-2025) 150 milliards d'objets devraient se connecter entre eux, avec l'internet et avec plusieurs milliards de personnes.

1.9. CONCLUSION

Dans cette partie on a présenté L'Internet des Objets : une description ainsi que le fonctionnement ; leur domaines d'application et la tendance et historique de la Technologie ; la connectivité des choses ;

Et nous a permis que L'Internet des Objets constitue l'avenir pour les entreprises. Il reste cependant difficile de définir comment et par quels moyens les entreprises vont adopter et s'approprier ces objets. L'abondance de technologies et le manque de visibilité sont les principales raisons qui poussent les entreprises à hésiter le basculement vers le tout connecté.

2. MICROCONTROLEUR ET ARDUINO

2.1 INTRODUCTION

Le terme micro-ordinateur est utilisé pour décrire un système qui comprend au moins un microprocesseur, une mémoire de programme (ROM), la mémoire de données (RAM), et une entrée-sortie (I/O) périphérique. Certains systèmes de micro-ordinateurs comprennent des composants supplémentaires tels que des Timers, des compteurs et des convertisseurs analogique-numérique. Ainsi, un système de micro-ordinateur peut être existé sous plusieurs sortes à partir d'un gros ordinateur ayant disques durs, disquettes, imprimantes et à contrôleur intégré à puce unique.

2.2. MICROCONTROLEUR

2.2.1. Définition: Un microcontrôleur est un circuit hautement intégré qui rassemble, sur une puce, la plus part ou tous les parties nécessaire d'un ordinateur.

Le microcontrôleur comprend généralement: CPU (Central Processing Unit), la mémoire RAM (Random Access Memory), EPROM / PROM / ROM (programmable effaçable Read Only Memory), I / O (entree / sortie), des timers.

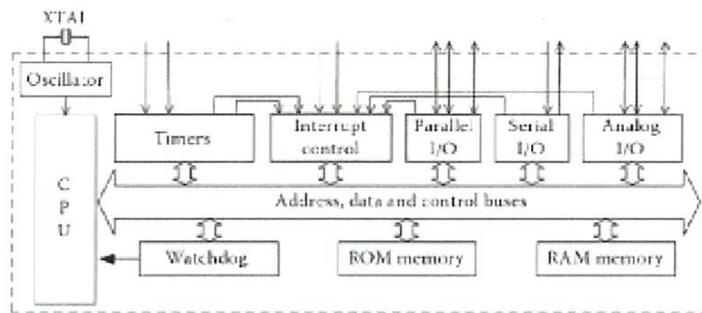


Figure I.6: Schéma des éléments principaux d'un microcontrôleur

2.2.2 Les types de microcontrôleurs

Les microcontrôleurs peuvent être classés selon la largeur des bus interne, l'architecture de la mémoire et de jeu d'instructions.

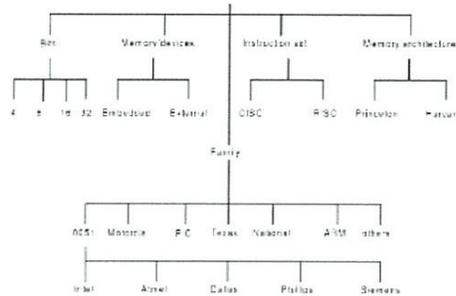


Figure I.7: Les types des microcontrôleurs Les microcontrôleurs de 8, 16 et 32 bits

- ✚ Lorsque l'UAL effectue des opérations arithmétiques et logiques sur un octet (**8 bits**) à une instruction, on dit qu'il est un microcontrôleur de 8 bits. Son bus interne est de largeur de 8 bits.
- ✚ Lorsque l'UAL effectue des opérations arithmétiques et logiques sur un octet (**16 bits**) à une instruction, on dit qu'il est un microcontrôleur de 16 bits. Son bus interne est de largeur de 16 bits. Ses performances et la capacité de calcul sont renforcées avec une plus grande précision par rapport aux microcontrôleurs 8 bits.
- ✚ Lorsque l'UAL effectue des opérations arithmétiques et logiques sur un octet (**32 bits**) à une instruction, on dit qu'il est un microcontrôleur de 32 bits. Son bus interne est de largeur de 32 bits. Ses performances et la capacité de calcul sont plus grandes et plus précis par rapport aux microcontrôleurs 16 bits.

2.2.3. L'architecture des microcontrôleurs

Architecture de Von Neumann-(Princeton): quand un UC a une mémoire commune utilisé pour le programme et les données, on dit qu'il a l'architecture Princeton de mémoire dans le processeur.

Architecture de Harvard: quand la mémoire de programme se distingue de la mémoire de données, le UC a l'architecture de Harvard dans son processeur. Alors l'accès à chacune des deux mémoires se fait via un chemin distinct. Cette organisation permet de transférer une instruction et des données simultanément, ce qui améliore les performances.

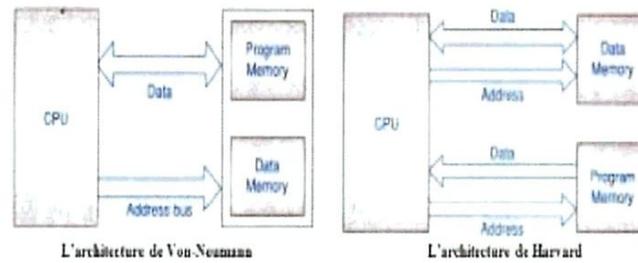


Figure I.8 : L'architecture d'un microcontrôleur

2.2.4. Domaines d'applications des microcontrôleurs

Système de communication: Les microcontrôleurs 8 bits sont souvent utilisés pour les téléphones portables simples et la téléphonie fixe alors que les microcontrôleurs 32 bits se retrouvent plutôt dans les Smartphones et les PDA. [7]



Figure I.9: Les appareils de communication

Technique médicale: Les instruments de mesure (par exemple mesure de la glycémie), les organes artificiels, etc.



Figure I.10 : Appareil médicale

Les technologies de la sécurité: Les systèmes pour gérer la sécurité dans les moyens de transport (par exemple : les passages à niveau), dans les bâtiments (par exemple: alarme incendie, effractions) etc.



Figure I.11 : Appareil de sécurité

Mécatronique et automation industrielle : Installation pour la production de biens, pour la logistique, etc.



Figure I.12 : Les appareils industriels

Moyens de transport: Autos, avions, vélo électrique etc.



Figure I.13 : Les moyens de transports

Electronique de consommation: Appareil hifi, TV, vidéo, télécommande etc.



Figure I.14 : TV, vidéo, HIFI

Application basse consommations : Les appareils à piles, tels que ceux développés par la HESB pour la station ornithologique suisse, qui permettent d'enregistrer des données spatiales et télémétriques.

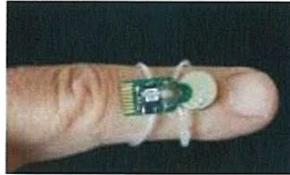


Figure I.15 : Enregistreur de données géographiques à pile et système de fixation

2.3 ARDUINO

2.3.1. C'est quoi l'Arduino?

Elle est considérée comme un grand équipement électronique pour les amateurs ainsi que les professionnels, l'Arduino est un puissant ordinateur à carte unique avec une utilisation facile à adapter.

L'Arduino est un circuit imprimé en matériel libre (dont les plans de la carte elle-même sont publiés en licence libre mais dont certains composants sur la carte, comme le microcontrôleur par exemple, ne sont pas en licence libre), sur lequel se trouve un microcontrôleur qui peut être programmé pour analyser et produire des signaux électriques, de manière à effectuer des tâches très diverses comme la domotique (le contrôle des appareils domestiques – éclairage, chauffage...). [8]

2.3.2. Bref Historique de l'Arduino

Le projet Arduino est né en hiver 2005. Massimo Banzi enseigne dans une école de Design à Ivrea en Italie, et souvent ses étudiants se plaignent de ne pas avoir accès à des solutions bas prix pour accomplir leurs projets de robotique. Banzi en discute avec David Cuartielles, un ingénieur Espagnol spécialisé sur les microcontrôleurs...[7]

Ils décident de créer leur propre carte en embarquant dans leur histoire un des étudiants de Banzi, David Mellis qui sera chargé de créer le langage de programmation allant avec la carte. En deux jours David écrira le code! Trois jours de plus et la carte était créée...Ils décidèrent de l'appeler Arduino (un bar fréquenté par les élèves à proximité de l'école)...

Ça devient un hit tout de suite auprès des étudiants. Tout le monde arrive à en faire quelque chose très rapidement sans même avoir de connaissances particulière ni en électronique ni en informatique: réponse à des capteurs, faire clignoter des leds, contrôler des moteurs... Ils publient les schémas, investissent 3000 euros pour créer le premier lot de cartes: 200.

Les 50 premières partent directement à des élèves de l'école. En 2006 5 000 cartes vendues...En 2007 plus de 30 000! en 2011 : >120 000, sans compter les clones !

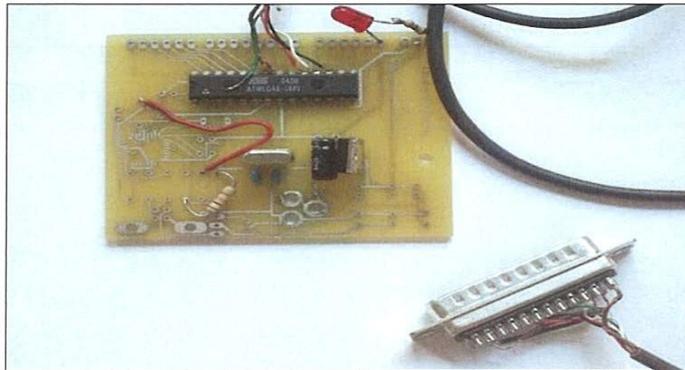


Figure I.16 : Prototype de l'Arduino

2.3.3. Matériel Arduino

La carte Arduino repose sur un circuit intégré (un microcontrôleur) associée à des entrées et sorties qui permettent à l'utilisateur de brancher différents types d'éléments externes :

Côté entrées, des capteurs qui collectent des informations sur leur environnement comme la variation de température via une sonde thermique, le mouvement via un détecteur de présence ou un accéléromètre, le contact via un bouton-poussoir, etc... [7]



Figure I.17 : Différent types des capteurs pour l'Arduino

Côté sorties, des actionneurs qui agissent sur le mode physique telle une petite lampe, un moteur, hautparleur...



Figure I.18: Différent s actionneurs pour l'Arduino

2.3.4. Les différents types de cartes Arduino

Il existe plusieurs types de l'Arduino (Uno, Mēga, Nano, Minietc) la différence entre ces type sont en termes de nombre d'entrées et sorties, ainsi que le microcontrôleur la vitesse du processeur trouvé à l'intérieur, dans notre projet nous allons utiliser Arduino Mēga 2560.

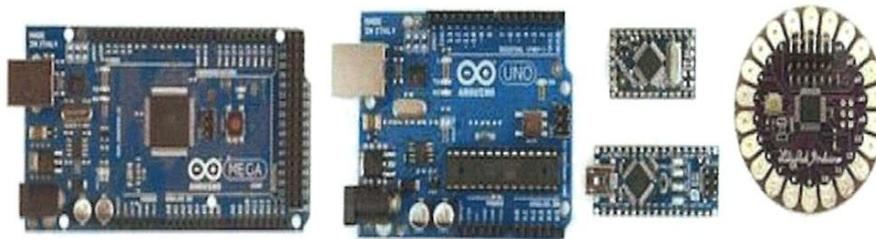


Figure I.19 : Exemples de différents types de cartes Arduino

Nom de la carte	Version du micro contrôleur	Flash (ko)	EEPROM (ko)	SRAM (ko)	Broches		Broches E/S analogique
					E/S Numérique	Avec PWM	
Mega2560	ATmega2560	256	4	8	54	15	16
Uno	ATmega328P	32	1	2	14	6	6
	ATmega168						
Nano	Ou ATmega328P	16/32	0.5/1	1/2	14	6	8
Yun	ATmega32u4	32	1	2.5	20	7	12
Deicimila	ATmega168	16	0.5	1	14	6	6

Tableau I.1 : Différent types des cartes Arduino

2.3.5. Description de la carte

La carte méga 2560 est basée sur le microcontrôleur ATmega2560 qui possède une mémoire flash programmable de 256 K octets avec des capacités en lecture et écriture, et une EEPROM de 4Ko, SRAM de 8 Ko.

Ce microcontrôleur possède aussi 86 I/O lignes, 32 registres généraux de travail, compteur en temps réel (RTC), six minuterie souple/compteurs avec compare modes et PWM, quatre USART, 2 fils Interface série d'octets orientée, 16 canaux, un CAN de 10 bits avec l'option d'étage d'entrée différentiel à gain programmable, minuterie programmable de chien de garde avec oscillateur

interne, un port série SPI, IEEE® std. 1149.1 D'interface de test JTAG, est également utilisé pour accéder au système de débogage sur puce et de la programmation et de six logiciels modes d'économie d'énergie sélectionnable.

✦ Caractéristique d'Arduino Uno

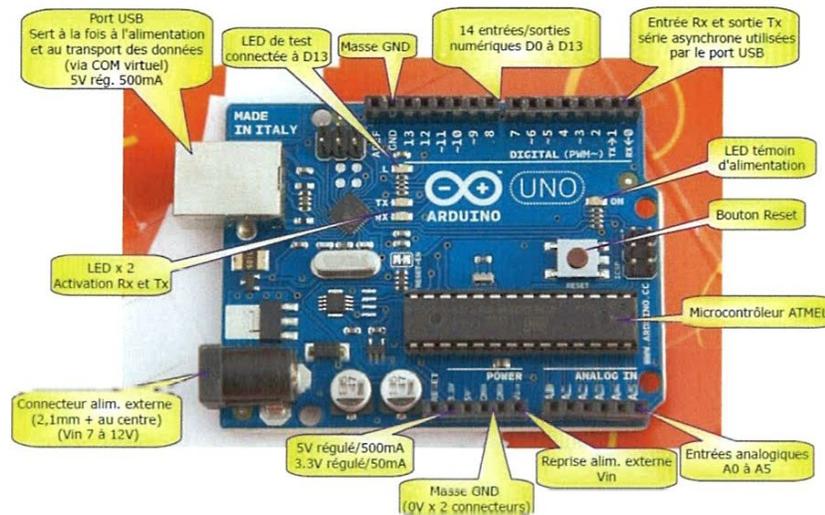


Figure I.20: Caractéristique d'Arduino Uno

2.3.6. L'alimentation

Le Méga 2560 peuvent être alimentés via la connexion USB ou avec une alimentation externe. La source d'alimentation est automatiquement sélectionnée.

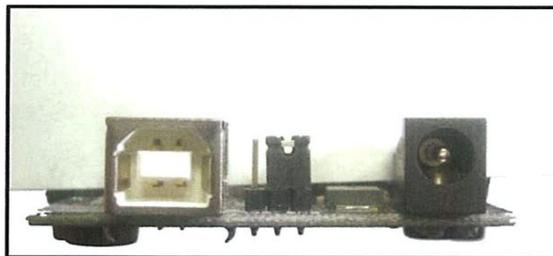


Figure I.21 : Alimentation Arduino Mēga 2560

2.3.7. Mémoire

L'ATmega2560 possède une mémoire flash 2560 Ko pour le stockage de code (dont 8 Ko est utilisé pour le boot loader), 8 Ko de SRAM et 4 Ko de mémoire EEPROM (qui peut être lu et écrit avec la bibliothèque EEPROM). [8]

2.3.8. Logiciel Arduino

Le logiciel Arduino est gratuit (open source) et se télécharger sur le site officiel d'Arduino.

2.3.8.1 Définition Arduino IDE :

IDE (Integrated Development Environment) est un programme spécial exécutable sur votre ordinateur qui vous permet d'écrire des esquisses pour la carte Arduino dans un langage simple sur le modèle du langage de traitement. La magie se produit lorsque vous appuyez sur le bouton qui télécharge l'esquisse à la carte: le code que vous avez écrit est traduit dans la langue de C (qui est généralement assez difficile pour un débutant d'utiliser), et est passé au compilateur avr-gcc, une pièce importante du logiciel open source qui fait la traduction finale dans la langue comprise par le microcontrôleur. Cette dernière étape est très importante, parce que c'est là où Arduino rend votre vie simple en cachant autant que possible de la complexité de la programmation des microcontrôleurs. [7]

2.3.8.2 Interface d'Arduino IDE

✚ Présentation du logiciel

Double-clic sur l'icône IDE Arduino est on obtient la fenêtre vierge ci-dessous :

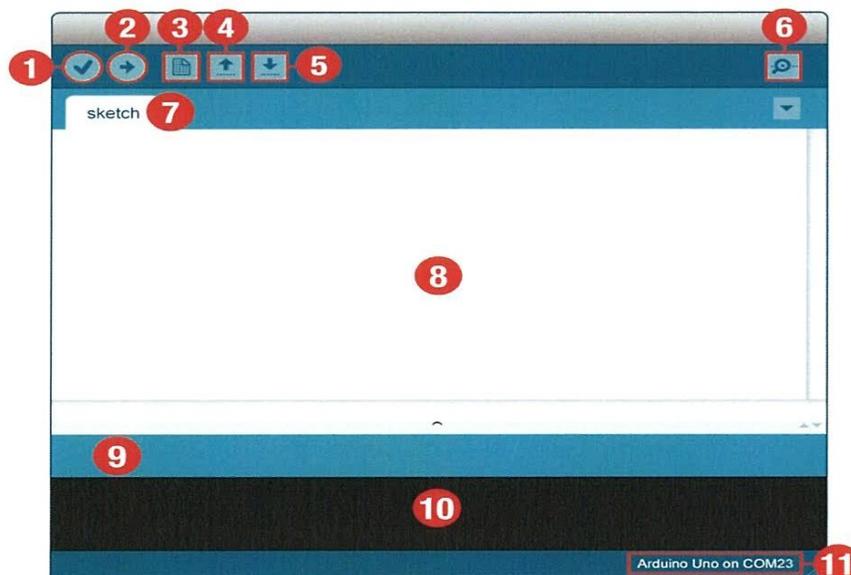


Figure I.22: Interface de logiciel Arduino.

- 1- Test et sauvegarder programme : Compile et approuve votre code. Il va attraper les erreurs de syntaxe (comme des points-virgules ou parenthèses manquantes)
- 2- Télé verser programme : Envoie votre code à l'carte de test. Lorsque vous cliquez dessus, vous devriez voir les lumières sur votre planche clignotent rapidement.
- 3- Nouveau fichier : Ce bouton ouvre un nouvel onglet de la fenêtre de code.
- 4- Ouvrir : Ce bouton vous permettra de vous ouvrir une esquisse existante
- 5- Enregistré : Cela permet d'économiser l'esquisse active.
- 6- ouvrir moniteur de série : Cela va ouvrir une fenêtre qui affiche les informations de série de votre Fil Rouge transmet. Il est très utile pour le débogage. *Le moniteur de série est comme un outil pour "parler avec l'Arduino et vous l'utilisez pour voir les choses d'intérêt'.*
- 7- Sketch Nom: Affiche le nom de l'esquisse que vous travaillez actuellement.
- 8- La place du écrire votre code : Ceci est la zone où vous composez le code pour votre croquis.
- 9- Zone de message: Ceci est où l'IDE vous indique s'il y avait des erreurs dans votre code.
- 10- Console de texte: La console de texte affiche des messages d'erreur complets. Lors du débogage, la console de texte est très utile.
- 11- Conseil et Port série: vous montre ce conseil et les sélections de port série.[8]

2.3.9. L'environnement de développement IDE de l'ARDUINO

Le développeur Arduino fournit l'environnement de développement intégré de l'Arduino (Arduino IDE), qui est une application multiplateforme écrite dans le langage de programmation Java.

Il est conçu pour faciliter la programmation de ces cartes aux artistes (pour lesquels ces cartes ont été développées en premier) et à tout nouvel utilisateur qui ne connaît pas le développement de logiciels.

L'environnement Arduino IDE inclut un éditeur de code et fournit une application qui d'un simple clic permet de compiler et charger des programmes dans une carte Arduino

2.3.10. Les avantages de l'Arduino

- ❖ Elle n'est pas chère!
- ❖ Environnement de programmation clair et simple.
- ❖ Multiplateforme : tourne sous Windows, Macintosh et Linux.
- ❖ Nombreuses bibliothèques disponibles avec diverses fonctions implémentées.
- ❖ Logiciel et matériel open source et extensible.
- ❖ Nombreux conseils, tutoriaux et exemples en ligne (forums, site personnel etc...)
- ❖ Existence de « shield » : ce sont des cartes supplémentaires qui se connectent sur le module Arduino pour augmenter les possibilités comme par exemple : afficheur graphique couleur, interface Ethernet, GPS, etc... [7]

2.4 CONCLUSION

On a pris une vue détaillée sur la carte d'Arduino y compris l'architecture interne aussi son environnement (partie logiciel). L'Arduino représente un logiciel de programmation par code basé sur des cartes électroniques à microcontrôleur open source et qui peut être utilisé pour construire des objets interactifs indépendants (prototypage rapide), ou bien peut être connecté à un ordinateur.

3. COMMUNICATION TCP/IP (MODULE ETHERNET)

3.1. INTRODUCTION

TCP/IP désigne communément une architecture réseau, mais cet acronyme désigne en fait 2 protocoles étroitement liés : un protocole de transport, TCP (Transmission Control Protocol) qu'on utilise "par-dessus" un protocole réseau, IP (Internet Protocol). Ce qu'on entend par "modèle TCP/IP", c'est en fait une architecture réseau en 4 couches dans laquelle les protocoles TCP et IP jouent un rôle prédominant, car ils en constituent l'implémentation la plus courante. Par abus de langage, TCP/IP peut donc désigner deux choses : le modèle TCP/IP et la suite de deux protocoles TCP et IP.

Le modèle TCP/IP, comme nous le verrons plus bas, s'est progressivement imposé comme modèle de référence en lieu et place du modèle OSI. Cela tient tout simplement à son histoire. En effet, contrairement au modèle OSI, le modèle TCP/IP est né d'une implémentation ; la normalisation est venue ensuite. Cet historique fait toute la particularité de ce modèle, ses avantages et ses inconvénients.

L'origine de TCP/IP remonte au réseau ARPANET. ARPANET est un réseau de télécommunication conçu par l'ARPA (Advanced Research Projects Agency), l'agence de recherche du ministère américain de la défense (le DOD : Department of Defense). Outre la possibilité de connecter des réseaux hétérogènes, ce réseau devait résister à une éventuelle guerre nucléaire, contrairement au réseau téléphonique habituellement utilisé pour les télécommunications mais considéré trop vulnérable. Il a alors été convenu qu'ARPANET utiliserait la technologie de commutation par paquet (mode datagramme), une technologie émergente promettant. C'est donc dans cet objectif et ce choix technique que les protocoles TCP et IP furent inventés en 1974. L'ARPA signa alors plusieurs contrats avec les constructeurs (BBN principalement) et l'université de Berkeley qui développait un Unix pour imposer ce standard, ce qui fut fait. [9]

3.2. PRESENTATION ET FONCTIONNEMENT :

Le sigle TCP/IP désigne un protocole de communication utilisé sur Internet. Ce protocole définit les règles que les ordinateurs doivent respecter pour communiquer entre eux sur le réseau Internet.

Nous vous proposons de découvrir plus en détails ses caractéristiques et son fonctionnement.

3.2.1. Le rôle du protocole IP

Le protocole IP pour "Internet Protocol" permet aux ordinateurs reliés à Internet de dialoguer entre eux. Plus précisément, l'Internet Protocol se charge du routage de chaque paquet vers sa destination (c'est-à-dire de l'acheminement des données sur le réseau).

Comme pour un courrier postal, sur Internet, chaque message (c'est-à-dire chaque paquet de données) s'accompagne de différentes informations :

- l'adresse de l'expéditeur
- l'adresse du destinataire
- des données supplémentaires.

Pour que les paquets parviennent à leur destination, une adresse IP unique est attribuée à chaque Ordinateur connecté au réseau.

Cette Adresse IP se présente sous la forme de 4 nombres (entre 0 et 255) séparés par des points : 204.35.129.3

Les paquets de données transitent par des dizaines d'ordinateurs jusqu'à atteindre leur destinataire.

Le protocole TCP/IP fait travailler ensemble IP et TCP pour que les paquets IP ne se perdent pas ou n'arrivent pas en double exemplaire et pour s'assurer que le paquet est bien arrivé à destination.

3.2.2 Le rôle du protocole TCP

Le protocole TCP se charge de la communication entre les applications, c'est-à-dire entre les logiciels utilisés par les ordinateurs.

Il vérifie que le destinataire est prêt à recevoir les données.

Il fractionne les messages en paquets plus petits (car les paquets IP ont une taille limitée) et numérote les paquets.

A la réception, il vérifie que tous les paquets sont bien arrivés et peut redemander les paquets manquants.

Il réassemble les paquets avant de les transmettre aux logiciels.

Il envoie des accusés de réception pour prévenir l'expéditeur que les données sont bien arrivées.

Le protocole TCP/IP permet donc à des logiciels situés sur des ordinateurs différents de communiquer de façon fiable. [10]

3.3 DESCRIPTION DU MODELE

3.3.1 Un modèle en 4 couches

Le modèle TCP/IP peut en effet être décrit comme une architecture réseau à 4 couches :

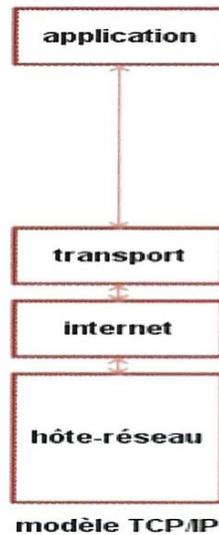


Figure I.23: l'architecture du modèle TCP/IP

3.3.1.1 La couche hôte réseau

Cette couche est assez "étrange". En effet, elle semble "regrouper" les couches physiques et liaison de données du modèle OSI. En fait, cette couche n'a pas vraiment été spécifiée ; la seule contrainte de cette couche, c'est de permettre un hôte d'envoyer des paquets IP sur le réseau. L'implémentation de cette couche est laissée libre. De manière plus concrète, cette implémentation est typique de la technologie utilisée sur le réseau local. Par exemple, beaucoup de réseaux locaux utilisent Ethernet; Ethernet est une implémentation de la couche hôte-réseau.

3.3.1.2 La couche internet

Cette couche est la clé de voûte de l'architecture. Cette couche réalise l'interconnexion des réseaux(hétérogènes) distants sans connexion. Son rôle est de permettre l'injection de paquets dans n'importe quel réseau et l'acheminement de ces paquets indépendamment les uns des autres jusqu'à destination. Comme aucune connexion n'est établie au préalable, les paquets peuvent arriver dans le désordre ; le contrôle de l'ordre de remise est éventuellement la tâche des couches supérieures.

Du fait du rôle imminent de cette couche dans l'acheminement des paquets, le point critique de cette couche est le routage. C'est en ce sens que l'on peut se permettre de comparer cette couche avec la couche réseau du modèle OSI.

La couche internet possède une implémentation officielle : le protocole IP (Internet Protocol).

Remarquons que le nom de la couche ("internet") est écrit avec un i minuscule, pour la simple et bonne raison que le mot internet est pris ici au sens large (littéralement, "interconnexion de réseaux"), même si l'Internet (avec un grand I) utilise cette couche.

3.3.1.3 La couche transport

Son rôle est le même que celui de la couche transport du modèle OSI : permettre à des entités paire sde soutenir une conversation.

Officiellement, cette couche n'a que deux implémentations : le protocole TCP (Transmission Control Protocol) et le protocole UDP (User Datagram Protocol). TCP est un protocole fiable, orienté connexion, qui permet l'acheminement sans erreur de paquets issus d'une machine d'un internet à une autre machine du même internet. Son rôle est de fragmenter le message à transmettre de manière à pouvoir le faire passer sur la couche internet. A l'inverse, sur la machine destination, TCP replace dans l'ordre les fragments transmis sur la couche internet pour reconstruire le message initial. TCP s'occupe également du contrôle de flux de la connexion.

UDP est en revanche un protocole plus simple que TCP : il est non fiable et sans connexion. Son utilisation présuppose que l'on n'a pas besoin ni du contrôle de flux, ni de la conservation de l'ordre de remise des paquets. Par exemple, on l'utilise lorsque la couche application se charge de la remise en ordre des messages. On se souvient que dans le modèle OSI, plusieurs couches ont à charge la vérification de l'ordre de remise des messages. C'est là une avantage du modèle TCP/IP sur le modèle OSI, mais nous y reviendrons plus tard. Une autre utilisation d'UDP : la transmission de la voix. En effet, l'inversion de 2 phonèmes ne gêne en rien la compréhension du message final. De manière plus générale, UDP intervient lorsque le temps de remise des paquets est prédominant.

3.3.1.4 La couche application

Contrairement au modèle OSI, c'est la couche immédiatement supérieure à la couche transport, tout simplement parce que les couches présentation et session sont apparues inutiles. On s'est en effet aperçu avec l'usage que les logiciels réseau n'utilisent que très rarement ces 2 couches, et finalement, le modèle OSI dépouillé de ces 2 couches ressemble fortement au modèle TCP/IP.

Cette couche contient tous les protocoles de haut niveau, comme par exemple Telnet, TFTP (trivial File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), HTTP (HyperText Transfer Protocol). Le point important pour cette couche est le choix du protocole de transport à utiliser. Par exemple, TFTP (surtout utilisé sur réseaux locaux) utilisera UDP, car on part du principe que les liaisons physiques sont suffisamment fiables et les temps de transmission suffisamment courts pour

qu'il n'y ait pas d'inversion de paquets à l'arrivée. Ce choix rend TFTP plus rapide que le protocole FTP qui utilise TCP. A l'inverse, SMTP utilise TCP, car pour la remise du courrier électronique, on veut que tous les messages parviennent intégralement et sans erreurs.

3.4 COMPARAISON AVEC LE MODELE OSI

Tout d'abord, les points communs. Les modèles OSI et TCP/IP sont tous les deux fondés sur le concept de pile de protocoles indépendants. Ensuite, les fonctionnalités des couches sont globalement les mêmes.

Au niveau des différences, on peut remarquer la chose suivante : le modèle OSI faisait clairement la différence entre 3 concepts principaux, alors que ce n'est plus tout à fait le cas pour le modèle TCP/IP.

Ces 3 concepts sont les concepts de services, interfaces et protocoles. En effet, TCP/IP fait peu la distinction entre ces concepts, et ce malgré les efforts des concepteurs pour se rapprocher de l'OSI. Cela est dû au fait que pour le modèle TCP/IP, ce sont les protocoles qui sont d'abord apparus.

Le modèle ne fait finalement que donner une justification théorique aux protocoles, sans les rendre véritablement indépendants les uns des autres.

Enfin, la dernière grande différence est liée au mode de connexion. Certes, les modes orienté connexion et sans connexion sont disponibles dans les deux modèles mais pas à la même couche : pour le modèle OSI, ils ne sont disponibles qu'au niveau de la couche réseau (au niveau de la couche transport, seul le mode orienté connexion n'est disponible), alors qu'ils ne sont disponibles qu'au niveau de la couche transport pour le modèle TCP/IP (la couche internet n'offre que le mode sans connexion). Le modèle TCP/IP a donc cet avantage par rapport au modèle OSI : les applications (qui utilisent directement la couche transport) ont véritablement le choix entre les deux modes de connexion. [9]

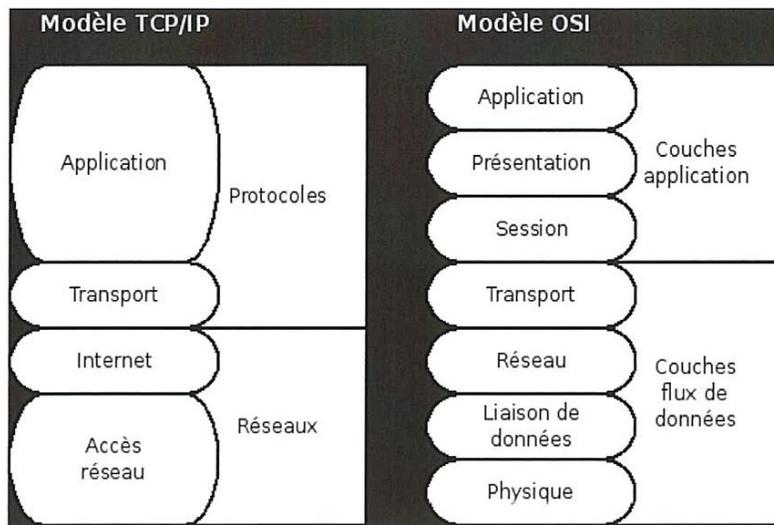


Figure I.24 : le modèle TCP/IP OSI

3.5 LE SCHEMA DU PROTOCOLE TCP/IP :

Le diagramme illustré dans la figure ci-dessous est appelé schéma de protocoles. Il présente certains protocoles communs spécifiés par le modèle de référence TCP/IP. Au niveau de la couche application, vous ne connaîtrez peut-être pas certaines tâches réseau, mais vous l'utilisez probablement tous les jours en tant qu'internaute. Vous étudierez chacune de ces couches dans le cadre du cursus.

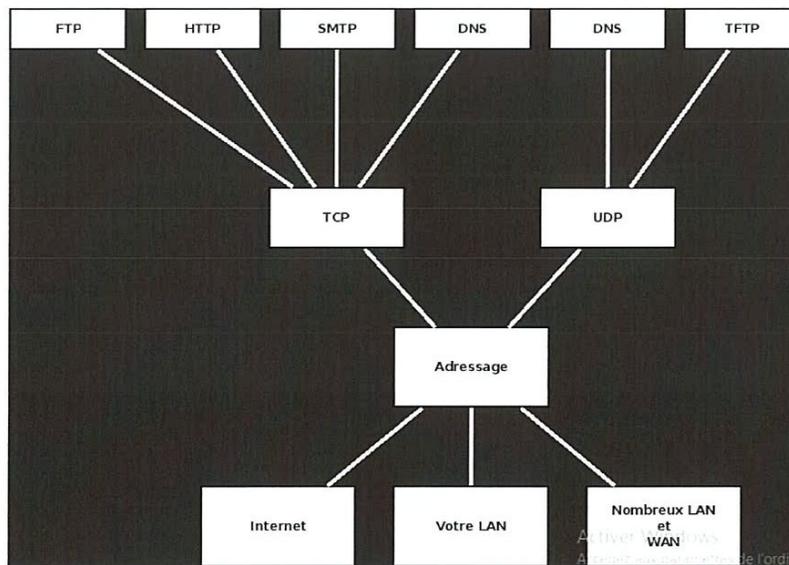


Figure I.25:Le schéma du protocole TCP/IP

Ces applications sont les suivantes :

FTP : Protocole de transfert de fichiers ou protocole FTP

HTTP : Protocole HTTP (Hypertext Transfert Protocol)

SMTP : Protocole SMTP (Simple Mail Transfert Protocol)

DNS : Système DNS (Domaine Name System)

TFTP : Protocole TFTP (Trivial File Transfert Protocol)

Le modèle TCP/IP met l'accent sur une souplesse maximale, au niveau de la couche application, à l'intention des développeurs de logiciels. La couche transport fait appel à deux protocoles : le protocole TCP (protocole de transmission) et le protocole UDP (User datagram Protocol). La couche inférieure, soit la couche d'accès au réseau, concerne la technologie LAN ou WAN utilisée.

Dans le modèle TCP/IP, IP4 est le seul et unique protocole utilisé, et ce, quels que soit le protocole de transport utilisé et l'application qui demande des services réseau. Il s'agit là d'un choix de conception délibéré. IP est un protocole universel qui permet à tout ordinateur de communiquer en tout temps et en tout lieu.

3.6 A QUOI SERT LE PROTOCOLE TCP/IP

Le protocole TCP/IP a de multiples utilisations.

Les navigateurs Internet et les serveurs s'en servent pour se connecter à Internet, ce qui vous permet d'envoyer et de recevoir des pages HTML, des images, des fichiers et toutes sortes d'autres données. [10]

3.7 LE MODULE ETHERNET

3.7.1 Introduction

Juste est arrivé notre nouveau module fantastique d'Ethernet. Ce module facile à utiliser. Aucune installation ou la programmation n'est exigée, le module d'Ethernet est immédiatement détecté sur le réseau. Ce module d'Ethernet est idéal pour la télésurveillance et le contrôle, chacun des points d'entrée-sortie peut être surveillé ou commuté de n'importe quel point sur le réseau, l'accès Internet et le contrôle est également possible.

Le dispositif est automatiquement assigné un IP adresse une fois relié au réseau. Un certain nombre de dispositifs peuvent être reliés au même réseau, à chaque emplacement ayant un IP adresse unique. Le module d'Ethernet soutient un large éventail de protocoles réseau ainsi qu'un connecteur RJ-45 standard.

Le logiciel de Windows est donné pour l'essai et la surveillance des divers points d'entrée-sortie. Le module d'Ethernet est compact et facilement monté dans les contrôleurs existants ou les nouvelles conceptions pour la connectivité facile de réseau.

Notre nouveau module d'Ethernet est idéal pour à télécommande et la surveillance au-dessus d'un réseau. L'unité est parfaite pour l'éclairage de changement dans un bureau ou à la maison au-dessus de votre réseau. Par acquisition de données avec des points d'alarme est possible pour donner le contrôle de visuel et de programme pour n'importe quel dispositif ou machines. La connexion par l'Internet est également facilement réalisable, permettre à télécommande et surveiller à partir de n'importe où dans le monde. Le module d'Ethernet est la solution parfaite pour ajouter les points à télécommande et surveillants d'entrée-sortie à un réseau.

3.7.2 Bref d'historique

Le premier réseau Ethernet fut ainsi mis en place en 1973 au PARC. Ses spécifications ne seront rendues publiques que le 30 septembre 1980 et la norme IEEE 802.3 ne sortira qu'en 1983. Les réseaux locaux (ou LAN pour Local Area Network) se répandront dans les entreprises en 1989 avec le lancement d'Ethernet 10BaseT, qui permet de faire transporter le signal par une paire torsadée semblable à celle du réseau téléphonique de l'établissement. [11]

3.7.3 Définition

3.7.3.1 **ETHERNET** : protocole d'accès au réseau qui décrit deux fonctions principales : d'une part, la communication sur une liaison de données et d'autre part, la transmission physique des données sur le support réseau. Les protocoles d'accès réseau prennent les paquets depuis le protocole IP et les formatent pour les transmettre via les supports.

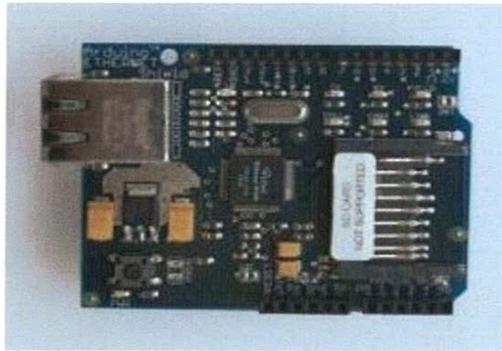


Figure I.26: Carte du module Ethernet

3.7.3.2 LE MODULE ETHERNET Arduino

Permet à une carte Arduino de se connecter à internet ; Il suffit d'utiliser la librairie Ethernet pour écrire des programmes qui se connectent à internet en utilisant ce module.

Le module Ethernet se connecte à une carte Arduino grâce à ses longues broches qui dépassent du circuit imprimé. Ainsi le brochage de la carte Arduino n'est pas modifié et permet d'enficher un autre module par-dessus et laisse l'accès aux broches de la carte Arduino.

3.7.4 Architecture interne du module Arduino Ethernet

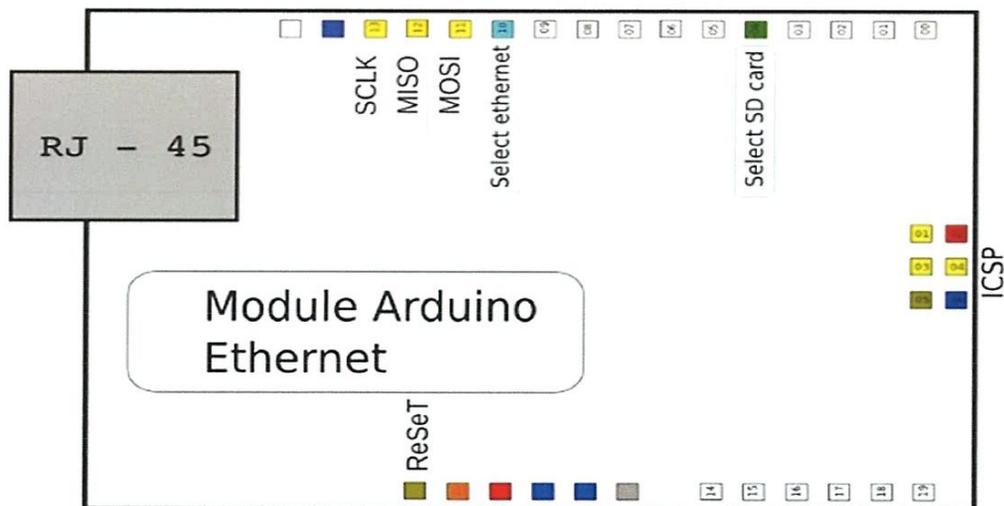


Figure I.27 : Architecture interne du module Arduino Ethernet

3.8 COMPLEMENTARITE ENTRE ETHERNET ET TCP/IP

Dès lors les réseaux d'ordinateurs vont s'appuyer à la fois sur TCP/IP et sur Ethernet. Ethernet sert à la communication entre les ordinateurs connectés à un même réseau local ; TCP/IP assure la communication à distance.

Ethernet est analogue à une conversation dans une salle où chacun prend la parole quand il a quelque chose à dire, s'interrompant en cas de collision avec un autre intervenant ; TCP/IP est (en plus rapide) semblable à l'envoi d'un texte, via la poste, par morceaux successifs qu'il faut classer à l'arrivée. Physiquement, le support de l'Ethernet peut être divers : paire de fils torsadée, câble coaxial, fibre optique, espace hertzien des réseaux WiFi ou câblage électrique de l'immeuble. L'Internet, lui, est composé d'un ensemble de routeurs et de liaisons louées aux opérateurs télécoms. Un réseau Ethernet est relié à l'Internet par une passerelle et une liaison louée.

Ainsi, sous réserve des droits d'accès, n'importe quel ordinateur d'un établissement peut communiquer avec un autre ordinateur d'un autre établissement, la communication empruntant des passerelles entre les réseaux Ethernet et TCP/IP

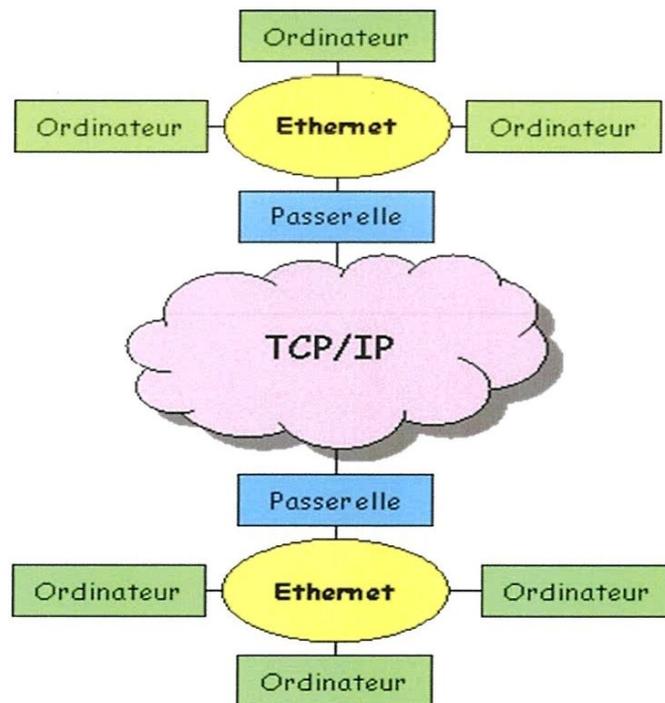


Figure I.28 : Complémentarité entre Ethernet Et TCP/IP

L'utilisateur individuel passe, lui, par le modem et la ligne téléphonique pour se relier à un fournisseur d'accès (IAP, Internet Access Provider), lui-même relié au réseau TCP/IP par une liaison louée.

Pour des raisons de sécurité certaines entreprises cloisonnent physiquement leur réseau : elles utilisent des réseaux privés virtuels fortement protégés (RPV ou VPN pour Virtual Private Network) selon des architectures conçues par les opérateurs télécoms. Sur ces réseaux peuvent entrer en jeu d'autres protocoles que TCP/IP (X25, Frame Relay, ATM etc.). Ces entreprises définissent aussi un Intranet fournissant à l'intérieur de l'entreprise des services analogues à ceux que l'on trouve sur l'Internet (messagerie, documentation électronique, moteur de recherche etc.) Alors le schéma se diversifie et s'enrichit, mais l'articulation entre le protocole du réseau local et celui (ou ceux) du réseau de transport reste la règle. [11]

3.9. CONCLUSION

Dans cette partie on a présenté une étude générale sur les protocoles TCP/IP une description ainsi que le Fonctionnement de chaque protocoles et architecture de ce modèle et nous avons terminé par le module Ethernet complémentarité entre Ethernet et TCP/IP

On conclut qu'il y'à des nombreux protocoles de communication différents sont basés sur les deux protocoles TCP et IP, par exemple les protocoles Http (Hyper Text Transfer Protocol), HTTPS (http sécurisé), SSL (Secure Sockets Layer), SMTP (Simple Mail Transfer Protocol) ou encore FTP (File Transfer Protocol) puisque Le TCP/IP a été développé très tôt et gratuitement soutenu par le système d'exploitation d'UNIX.

4. INTERFACE GRAPHIQUE (C #SHARPE)

4.1. INTRODUCTION

Dans le domaine de la programmation, on utilise aujourd'hui des langages de programmation de haut niveau par opposition aux langages de bas niveau (de type assembleur) qui sont plus orientés vers le langage machine. Parmi les exemples de langages de haut niveau, on peut citer les langages C, C++, C#.

Tout le code de cette documentation a été fait avec le bloc-notes et avec le Framework .NET. Pour pouvoir utiliser cette documentation vous devez être capable d'installer le Framework .NET. Cette documentation fonctionne comme ceci :

- Tous les mots en bleu/italique sont des variables et peuvent avoir des noms différents.
- Tous les mots en vert/italique sont des noms se rapportant au C++.
- Tous les mots en violet/italique sont des noms se rapportant au C#.

L'objectif de cette partie c'est une présentation exhaustive du langage C#. Pour cela une littérature nombreuse existe sur Internet. Le C# peut être considéré comme une synthèse du C++ et de Java.

On va faire une étude générale sur le langage de programmation C# de but ; principe et une bref d'historique ; ainsi que ces caractéristique ; nous verrons en particulier comment faire une interface graphique et le gestionnaires d'évènements ; comment déposer de composants graphique. [12]

4.2. QU'EST-CE QUE LE C# ?

C# doit beaucoup au vaste héritage de C++. Les développeurs C++ et Java ne se sentiront pas dépayés. Langage moderne, intuitif, orienté objet, C# comporte des améliorations substantielles, par exemple un système de types regroupé ou le code "unsafe" pour une maîtrise maximale. Le langage est également enrichi de nouvelles constructions que la majorité des développeurs comprendront aisément. [14]

4.3. Bref HISTOIRIQUE DE C Sharpe

Le projet C# a démarré au milieu des années 1990. Microsoft cherchait à obtenir un nouveau langage de programmation dérivé de C/C++. L'entreprise a tout d'abord apporté des améliorations au langage Java, mais cette voie a été abandonnée lorsque Sun Microsystems - propriétaire du langage Java - a intenté un procès contre eux.

Microsoft se lance alors dans le développement d'une nouvelle plateforme informatique adaptée aux applications web. C'est en septembre 2000 que la plateforme .NET et C# sont présentés au public. C# devient le langage de facto de cette plateforme, il a par ailleurs été utilisé pour implémenter une partie de la plateforme .NET

Le c# est un Langage de programmation crée par Anders Heljberg (Microsoft) en 2001. Il est utilisé pour développer des applications web, ainsi que des applications de bureau, des services web... [13]

*Classement TIOB des langages les plus utilisés au monde: 5ème en janvier 2014.

Année	Version	Bibliothèque	Principal changement
2002	1.0	NET Framework 1.0 et 1.1	
2005	2.0	NET Framework 2.0	généricité ajoutée à C# et au Framework
2008	3.0	NET Framework 3.5	LINQ (Language integrated queries)
2010	4.0	NET Framework 4.0	types dynamiques
2012	5.0	NET Framework 4.5	Méthodes asynchrones
2015	6.0	NET Framework 4.6	version pour Linux

Tableau I.2 : Historique de c Sharpe

4.4. LE BUT DE C#

Est de raccourcir le temps d'élaboration en permettant à des lotisseurs de passer leur temps travaillant à la logique d'application au lieu des détails de programmation de bas niveau.

4.5. ENVIRONNEMENT DE DEVELOPPEMENT C#

4.5.1 Présentation du logiciel

- Double-clic sur l'icône du logiciel est on obtient la fenêtre vierge ci-dessous :

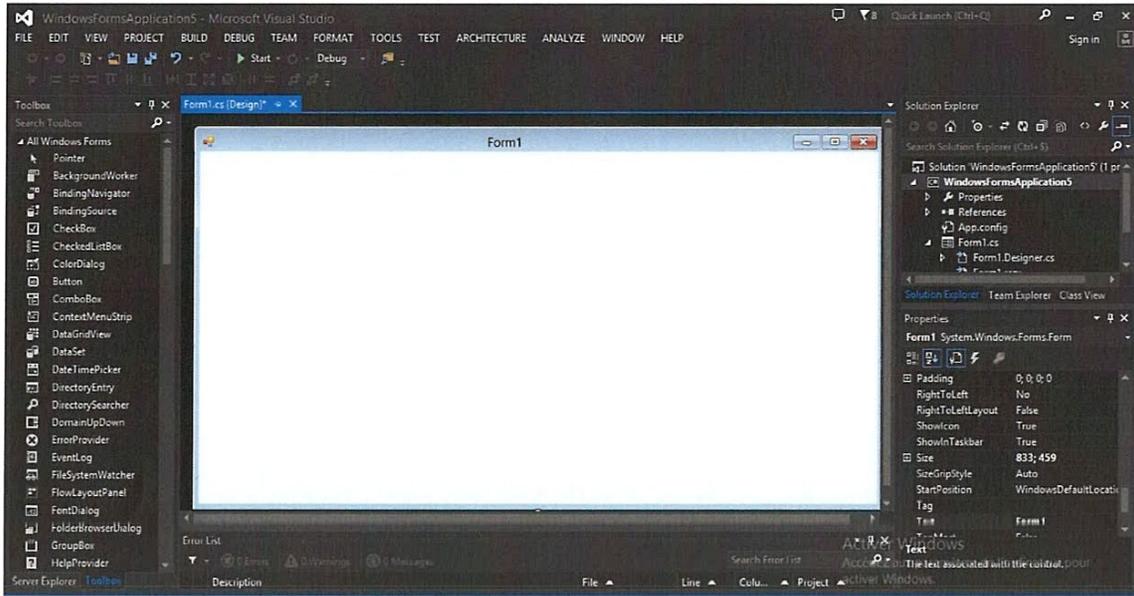


Figure I.29 : Interface de logiciel c#

- Barre d'outils de logiciel :

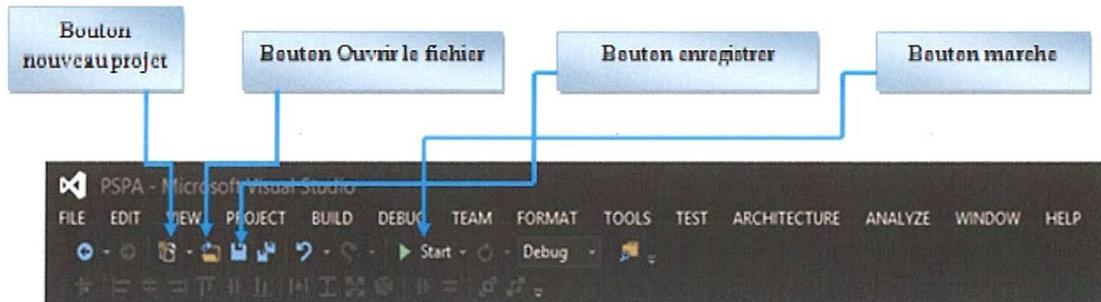


Figure I.30: Barre d'outils et commandes de menu

- solution exploré : contient, des éléments qui représentent les Références, les connexions de données les dossiers et les fichiers



Figure I.31 : solution explorer

- La barre des propriétés :

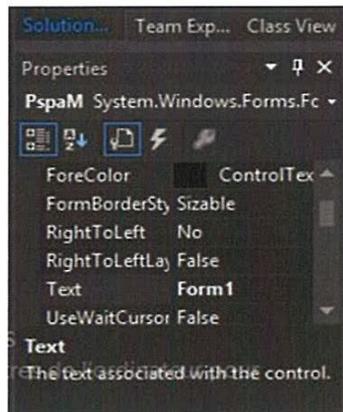


Figure I.32: Propriétés pour les solutions, les projets et les fichiers

4.6. AVANTAGES DE C# ?

- ❖ plus simples que d'autres langages de type objet.
- ❖ sûr et robuste.
- ❖ bon logiciel graphique.
- ❖ connexe à C et à C++.

4.7. CONCLUSION

Dans cette partie on a décrit rapidement le langage C#, le support d'information et aussi on apprend comment coder les informations et on a rappelé de façon brève la structure de programmation et le développement visuel.

CHAPITRE II :

Application

TISSEUSE CIRCULAIRE

1. INTRODUCTION

La Tisseuse Circulaire est une application qui vise la production de Sacs à Tissage Circulaire d'une façon souple, pour cela, en a fait ce travail pour prendre une vue détaillée sur ce type des machines.

2. DESCRIPTION



Figure II.1 :La tisseuse circulaire.

Cette machine se compose :

- ❖ un ensemble des bobines (on l'appelle aussi des antennes) qui nous forme les colonnes des sacs.

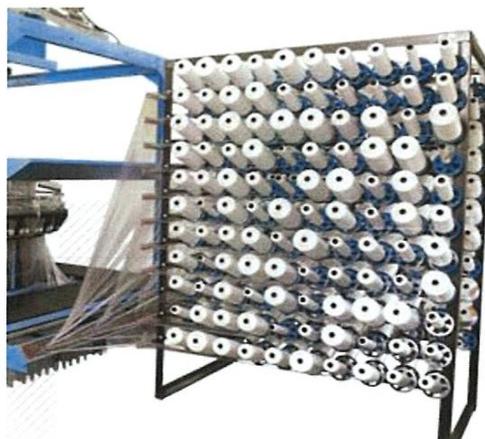


Figure II.2 :Les bobines /ou Les antennes

- ❖ Des navettes qui contiennent des bobines aussi pour former les lignes des sacs.

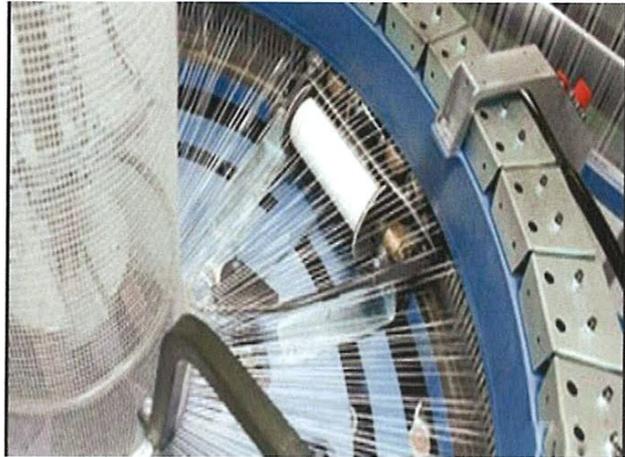


Figure II.3 : Les navettes

- ❖ UN moteur de puissance 4 KW qui assure le mouvement des différentes parties mécaniques de la machine.



Figure II.4 :Le moteur principale

- ❖ Trois pupitres de commande qui contiennent 3 boutons : bouton arrêt, bouton marche et bouton marche impulsionnelle.



Figure II.5 :Boite de commande

- ❖ On a aussi un capteur de proximité qui calcule le métrage des sacs par une pièce métallique fixer dans l'axe du cylindre de récupération des sacs.



Figure II.6 : Capteur De Proximité

- ❖ Un 2eme moteur de puissance 0.18 KW qui permet de récupérer le rouleau des sacs tissés



Figure II.7 : moteur de puissance qui récupéra le rouleau

- ❖ Un variateur de vitesse qui varie la vitesse de la machine à travers le moteur principale.



Figure II.8 : Un variateur de vitesse

Pour réaliser la commande de cette machine en utilisant une carte à microcontrôleur de type Arduino Méga largement disponible sur le marché.

Nous allons présenter dans les lignes suivantes une description simple comme rappelle de cette carte à microcontrôleur.

L'objectif de notre application est de suivre cette machine (les défauts, le contrôle de la vitesse, la production des sacs,) par graphe ou par état ou on va les présenter sur une interface graphique à l'aide de C#.

3. PARTIE HARDWARE DE LA TISSEUSE CIRCULAIRE

Le matériel nécessaire utilisé au développement de notre application :

- ✦ Une carte Arduino.
- ✦ Module Ethernet.
- ✦ Module LCD 16*2interfaceI2C.
- ✦ Une carte électronique fabriquée.
- ✦ PC.
- ✦ Routeur

3.1. µC ARDUINO

3.1.1. Aperçu

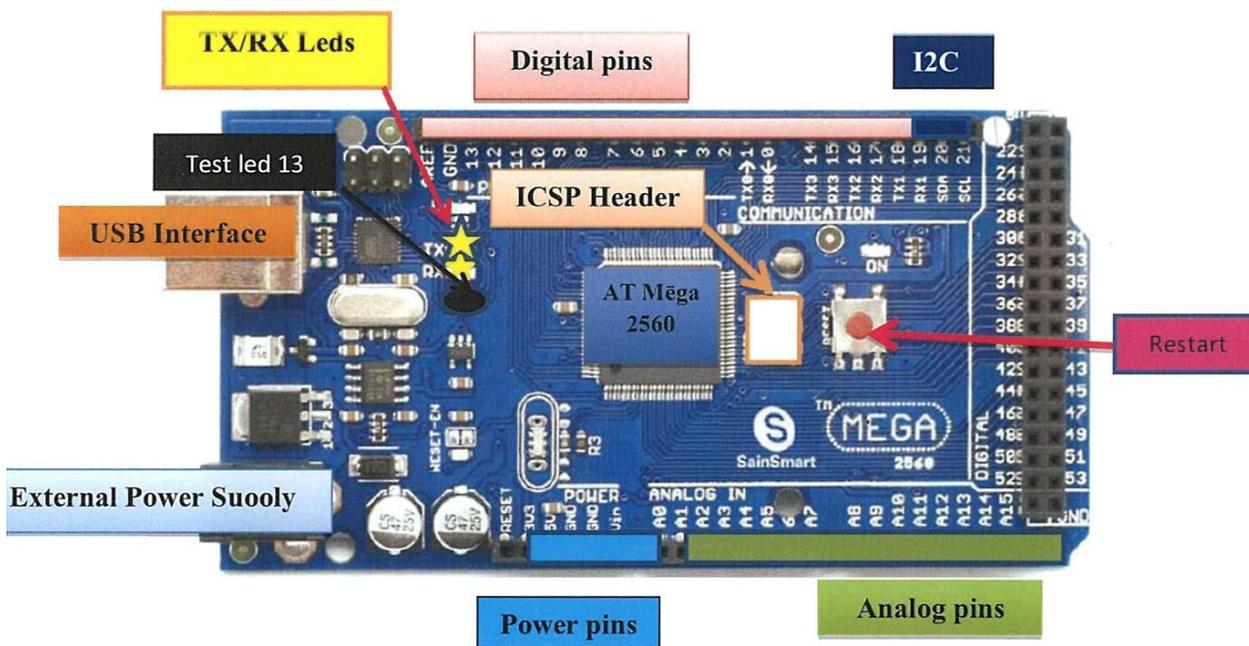


Figure II.9: Carte de développement ARDUINO Méga 2560.

L'Arduino Méga 2560 est une carte microcontrôleur basée sur le microcontrôleur ATmega2560.

Il possède :

- ❖ 54 entrées / sorties numériques : des broches (dont 14 peuvent être utilisées comme sorties PWM).
- ❖ 16 entrées analogiques.
- ❖ 4 UART (ports série matériels).
- ❖ Un oscillateur de 16 MHz (cristal de quartz).

- ❖ Une connexion USB, une prise d'alimentation.
- ❖ D'une embase ICSP.
- ❖ Un bouton de réinitialisation.

✚ La description simplifiée de la carte Arduino ainsi que le principe de fonctionnement ont été décrits dans le chapitre précédent.

3.1.2. Architecture Interne

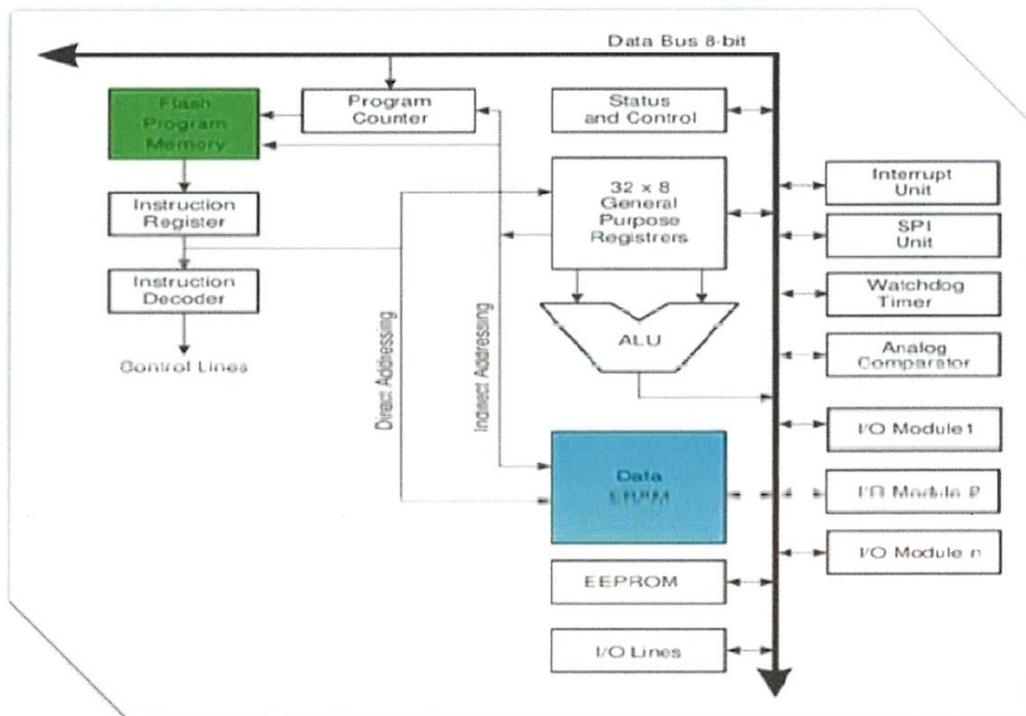


Figure II.10 : Architecture interne du microcontrôleur ATmega 2560.

L'ATmega 2560 est un microcontrôleur 8 bits à faible puissance (technologie CMOS) basé sur l'architecture RISC2. En exécutant des instructions dans un seul cycle d'horloge, l'ATmega2560 réalise des vitesses approchant 1 MIPS.

3.2. LES ENTREES LOGIQUES

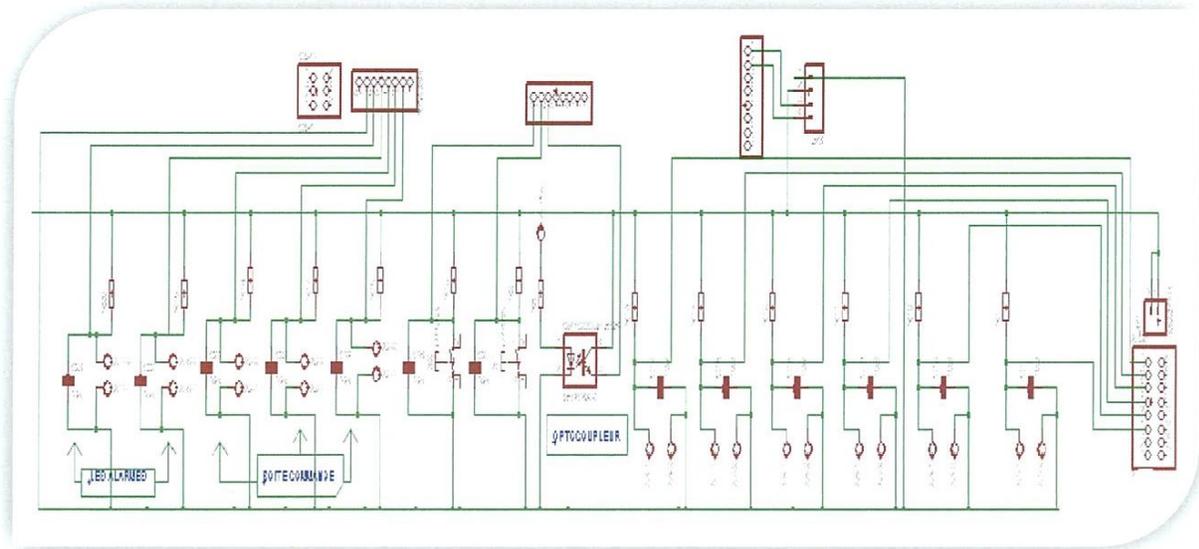


Figure II.11 : schéma des entrées logiques

La figure au-dessus présente le schéma électrique des entrées logiques de la carte électronique fabriqué réalisé par le logiciel Eagle.

- ❖ On a 6 entrées pour détecter les coupures dans chaque zone.
- ❖ Une entrées d'un optocoupleur relié à un capteur de proximité qui permet de calculer le rendement de la machine.
- ❖ Les trois entrées de pupitre de commande.
- ❖ deux entrées d'alarme du moteur et du variateur de vitesse.

Nous détaillons les composants électriques au niveau de la carte électronique comme suivant:

3.2.1. Alarme

Une alarme est :

- ✚ Un dispositif de sécurité visant à avertir (le plus souvent à l'aide d'un signal sonore) le propriétaire de la mise en danger du lieu où est installée l'alarme
- ✚ En d'autres termes, est un système de sécurité composé d'un ensemble d'éléments permettant de détecter un danger potentiel et, de cette manière, d'en prévenir les conséquences.

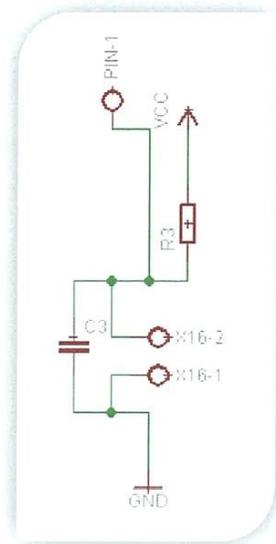


Figure II.12 : schéma d'entrée d'alarme

Ces éléments sont généralement constitués :

- ❖ d'une centrale d'alarme, qui centralise les informations envoyées par les détecteurs et prend la décision de lancer l'alerte,
- ❖ des détecteurs,
- ❖ d'un dispositif (sirène, système d'appel, voyant) destiné à donner l'alerte
- ❖ des dispositifs de commande (télécommande, clavier) permettant aux utilisateurs de mettre le système en marche ou à l'arrêt.

3.2.2. Bouton

- ✚ Boites commandes : "marche, arrêt, marche impulsionnelle".
- ✚ Variation de la vitesse du moteur principale.

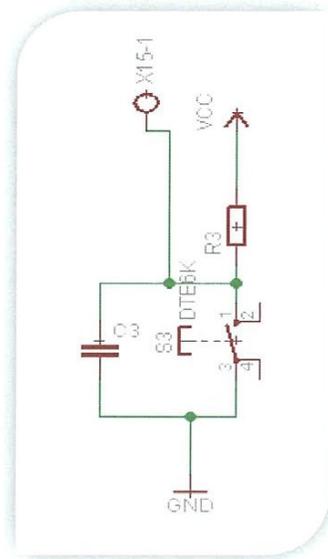


Figure II.13 : schéma d'entrée de Bouton

3.2.3. Optocoupleur

Un Optocoupleur est un dispositif composé de deux éléments électriquement indépendant, mais optiquement couplés, à l'intérieur d'une enveloppe, parfaitement étanche.

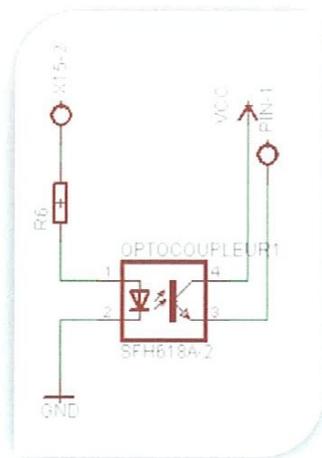


Figure II.14 : schéma d'un Optocoupleur

- ✓ Leur rôle est soit d'assurer une isolation galvanique (aucune liaison électrique) entre deux systèmes électriques pour des utilisations diverses comme :
 - ❖ Interface pour la transmission de données.
 - ❖ Commande de structures Basse Tension (Secteur EDF).

- ❖ Variation de puissance.

3.2.4. La détection des coupures de file

Nous avons 6 zones identique ou les fils passe pour entrer à la machine, notre but est de signaler les coupures de ces fils pour assurer une bonne qualité du sac tissé.

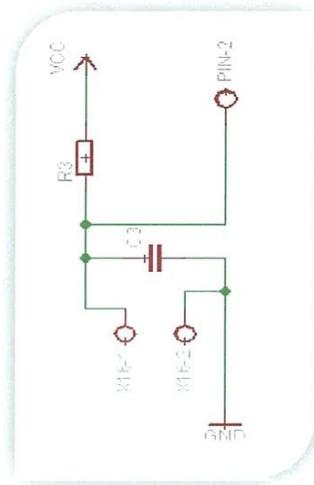


Figure II.15 :schéma d'une zone

3.3. LES SORTIES LOGIQUES

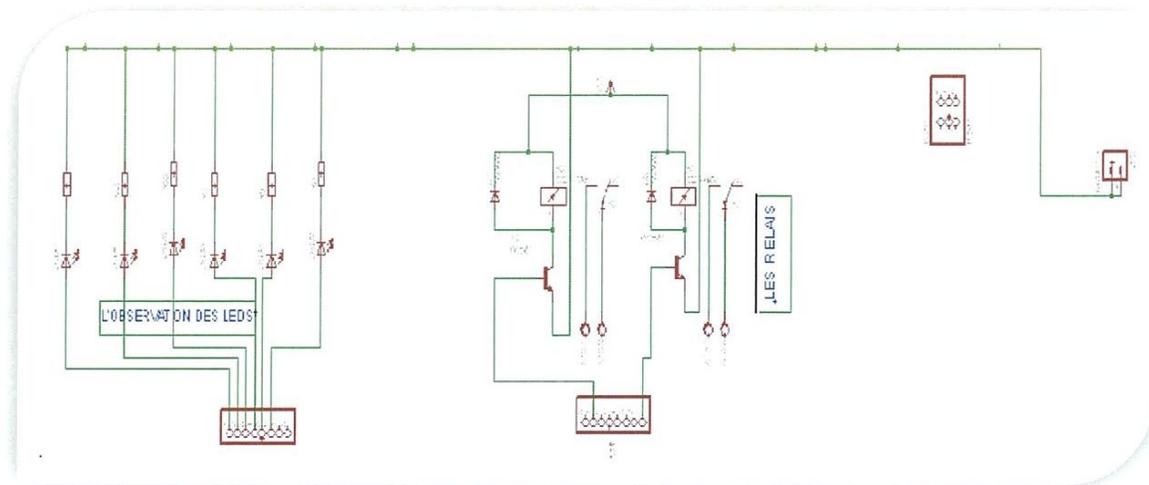


Figure II.16 : schéma des sorties logiques

Nous présentons sur cette figure les sorties logiques de la carte électronique, on a :

- ❖ Deux relais pour faire commander les deux moteurs.
- ❖ 6 leds a l'aide de son éclairage on observe dans quel zone on a détecté la coupure.

3.3.1. Relais

Un relais électronique est un interrupteur qui se commande avec une tension continue de faible puissance. La partie interruptrice sert à piloter des charges secteur de forte puissance.

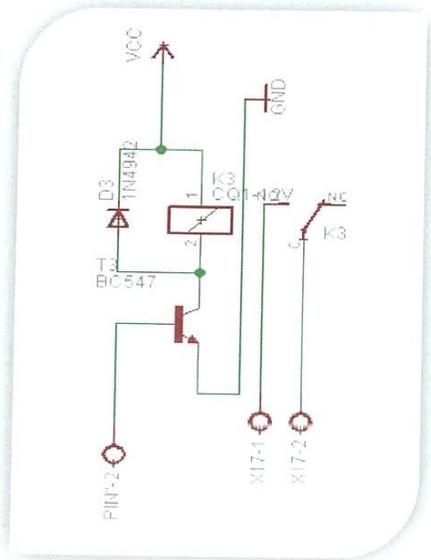


Figure II.17 : schéma du Relais

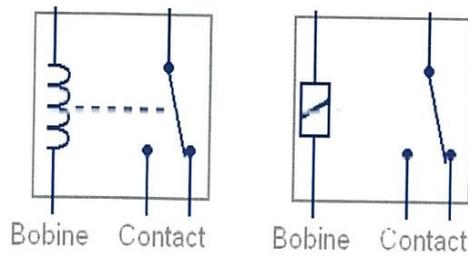


Figure II.18 : Symboles du relais

Le rôle de relais dans ce circuit est envoyé les commandes à les deux moteurs.

3.3.2. LED

Une LED est un composant optoélectronique : il s'agit donc d'un composant électronique capable d'émettre de la lumière lorsqu'il est parcouru par un courant électrique.

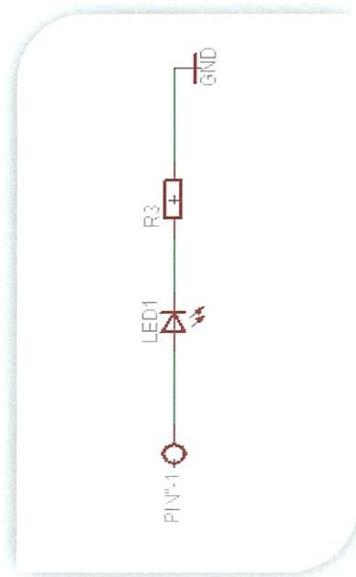


Figure II.19 : Schéma de LED

Le rôle de la Led ici c'est l'observation visuelle (il s'allume si il y'à une coupure dans l'un des zones).

3.4. SORTIE ANALOGIQUE

3.4.1. « L298 »

Le circuit intégré L298 pour les moteurs à courant continu. Ce circuit offre un double pont en H pour former une interface avec microcontrôleur.

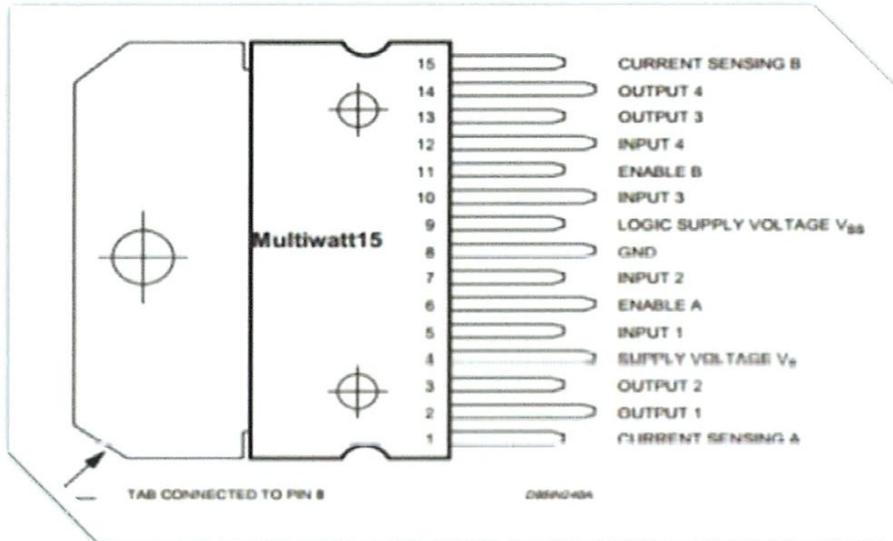


Figure II.20: Le L298N se décline dans un boîtier Multi watt à 15 pins

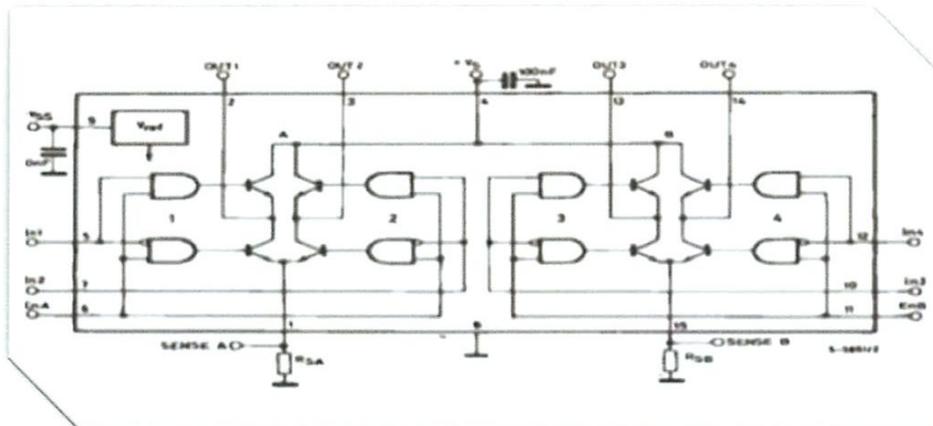


Figure II.21: Structure interne de L-298

❖ Fonctionnement du L298

Le principe de fonctionnement est résumé dans le tableau ci-dessous

Entrés		Sorties	Fonction
ENA=1	Int=1 ; Int2=0	Out1=1 ; Out2=0	La marche avant
	Int=0; Int2=1	Out1=0 ; Out2=1	La marche arrière
	Int1=Int2	Out1=1 ; Out2=1 ou Out1=0 ; Out2=0	Le frein
ENA=0	Int=X ; Int2=X	Rien	

Tableau II.1 : Fonctionnement de circuit intégré I.298

3.5. D'autres composants actifs et passifs de la carte

3.5.1. Le transistor BC547

Le BC547 est un petit transistor polyvalent et économique, idéal pour réaliser toutes sortes de circuits électroniques standard.

Le transistor BC547 appartient à une gamme qui contient aussi :

BC546; BC547;BC548 ; BC549 ; BC550

Les complémentaires en transistor PNP sont respectivement :

BC556 ; BC557 ; BC558 ; BC559 ; BC560

Ses principales applications sont en amplification (mode linéaire) et en commutation (mode tout ou rien).



Figure II.22 : Brochage du BC547C-B-E (collecteur, base, émetteur)

Le transistor BC547 est un classique de l'électronique faible puissance avec ses caractéristiques de 45V et 100mA. Si on souhaite une tranche de gain spécifique, il faut choisir entre BC547A, BC547B et BC547C.

3.5.2. La Résistance

- ✓ La résistance est un composant électrique dont la principale caractéristique, précisément, est de montrer une forte résistance au passage du courant électrique.
- ✓ grandeur mesurée en ohms
- ✓ Il est en effet destiné à protéger les autres composants d'un circuit électronique.
- ✓ Les résistances sont faites d'un mélange de carbone, de fil résistif ou de films de métal,
- ✓ La résistance est par ailleurs à l'origine d'une perte d'énergie sous forme de chaleur, c'est ce qu'on appelle l'effet Joule

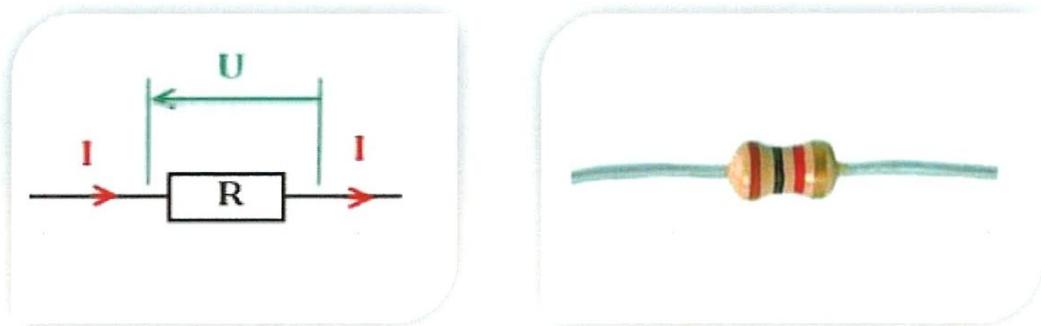


Figure II.23 : Symbole / Brochage de la Résistance

3.5.3. Le Condensateur

- ✓ Est un composant électronique élémentaire, constitué de deux armatures conductrices (appelées « électrodes ») en influence totale et séparées par un isolant polarisable (ou « diélectrique »).
- ✓ Sa propriété principale est de pouvoir stocker des charges électriques opposées sur ses armatures. La valeur absolue de ces charges est proportionnelle à la valeur absolue de la tension qui lui est appliquée.

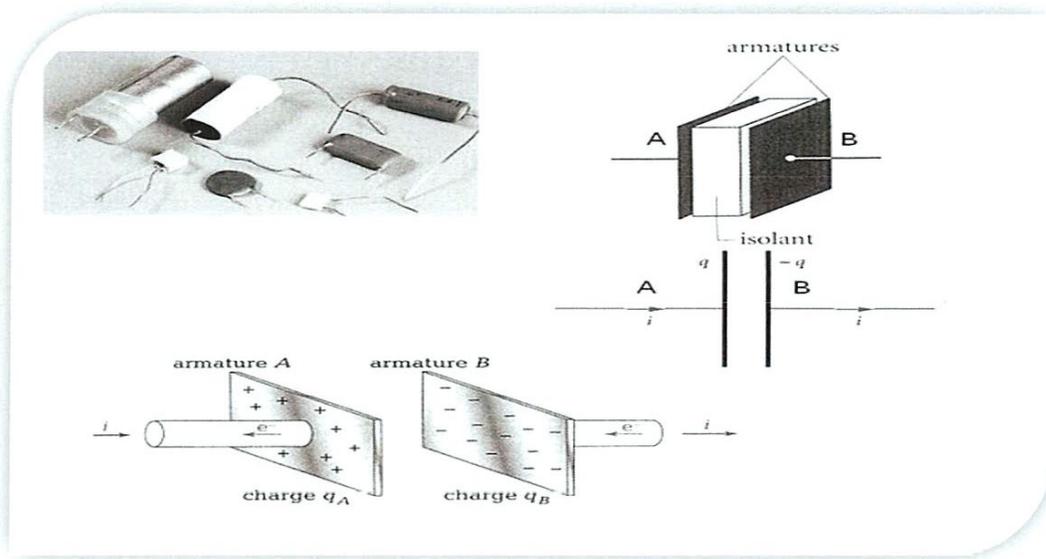


Figure II.21 : Quelque type de condensateur simplifié « électrodes »

Leur rôle :Le Condensateur est utilisé principalement pour :

- ❖ Sa première raison d'utilisation est d'emmagasiner temporairement des charges électriques et donc de l'énergie électrique.
- ❖ stabiliser une alimentation électrique (il se décharge lors des chutes de tension et se charge lors des pics de tension).
- ❖ traiter des signaux périodiques (filtrage.....).

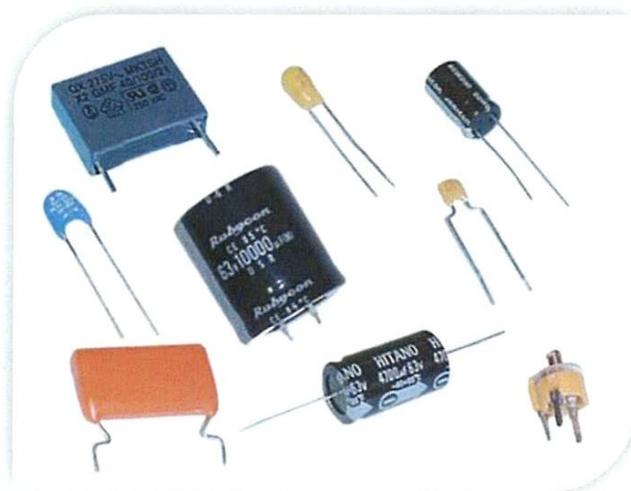


Figure II.25 : Quelques Exemples de différents condensateurs

3.6. INTERFACE TCP/IP « LE MODULE ETHERNET »

3.6.1. Le Module Ethernet est un module :

- ❖ idéal pour la télésurveillance et le contrôle.
- ❖ soutient un large éventail de protocoles réseau ainsi qu'un connecteur RJ-45 standard.
- ❖ la solution parfaite pour ajouter les points à télécommande et surveillants d'entrée-sortie à un réseau .
- ❖ est compact et facilement monter dans les contrôleurs existants ou les nouvelles conceptions pour la connectivité facile de réseau.

3.6.2. Aperçu :

La plupart des cartes d'interface réseau basées sur Ethernet.

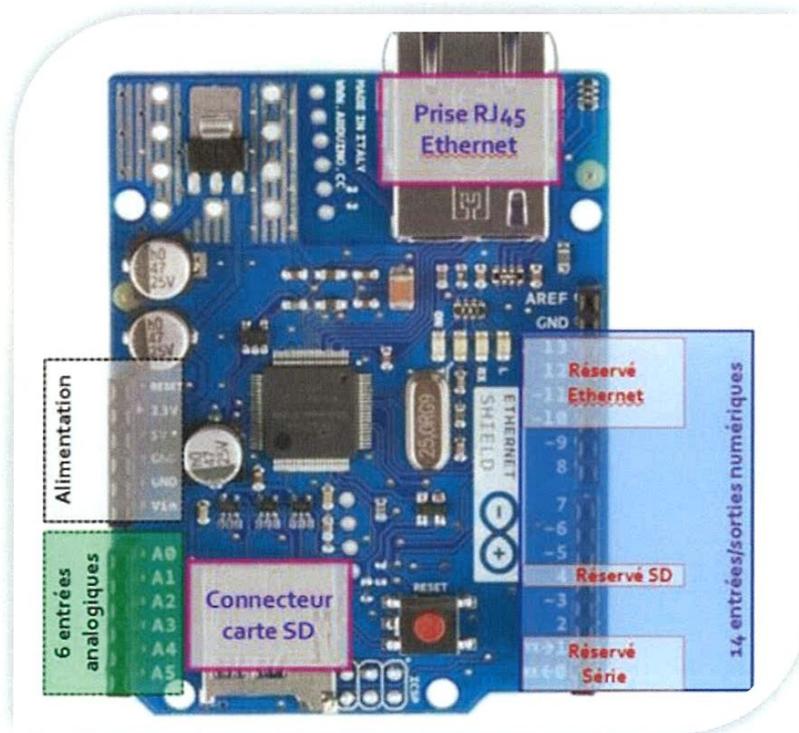


Figure II.26: Carte de développement de module Ethernet

✚ La carte Arduino communique avec le Ethernet et la carte SD (mémoire stockage) en utilisant le bus SPI (via le connecteur ICSP - Pour info : les broches 13 (SCK), 12 (MISO) et 11 (MOSI) sont connectées respectivement aux broches 3, 1 et 4 du connecteur ICSP).

✚ **SPI:** est un protocole communication série utilisant 3 broches. La communication SPI se fait en utilisant :

- ❖ les broches 11,12 et 13 de la carte Duemilanove ou UNO
- ❖ les broches 50,51 et 52 sur la carte Mēga.

Remarque : bien que les broches 13, 12 et 11 du module soient directement connectées aux broches 13,12 et 11 de la carte Arduino, il faut bien comprendre que la connexion SPI entre le module et la carte Arduino se fait via le connecteur ICSP et pas par la connexion directe des broches 13,12 et 11.

- ❖ la broche 4 est utilisée pour sélectionner la carte SD.
- ✚ Le module dispose d'un connecteur d'un connecteur réseau Ethernet standard RJ45.
- ✚ Le module comporte également de plusieurs Leds indicatrices :
 - ❖ PWR : indique que la carte et le module sont sous tension.
 - ❖ LINK : indique la présence d'un lien réseau et clignote quand le module transmet ou reçoit des données.
 - ❖ FULLD : indique que la connexion réseau est "full duplex"
 - ❖ 100M : indique la présence d'une connexion réseau à 100 Mbits/sec (10 Mbits/sec sinon)
 - ❖ RX : clignote quand le module reçoit des données
 - ❖ TX : clignote quand le module envoie des données.
 - ❖ COLL: clignote quand des collisions sont détectées.



Figure II.27 : le module Ethernet

3.6.3. Caractéristiques

- ❖ 24 points d'entrée-sortie peut être surveillé ou commuté facilement (de n'importe quel emplacement sur le réseau).
- ❖ Idéal pour à télécommande au-dessus de l'Internet.
- ❖ IP fixe programmable ou tâche dynamique d'IP.
- ❖ Essai de Windows et programme utilitaire de programmation disponibles.
- ❖ Installation et configuration automatiques.
- ❖ Le dispositif est immédiatement détecté sur le réseau.
- ❖ Interface industriellement compatible de l'Ethernet 10BaseT avec le connecteur RJ-45 standard.
- ❖ Idéal pour l'éclairage, les systèmes bureautiques et la domotique de contrôle.

3.7. INTERFACE I2C ((LCD 16*2))

3.7.1. Aperçu

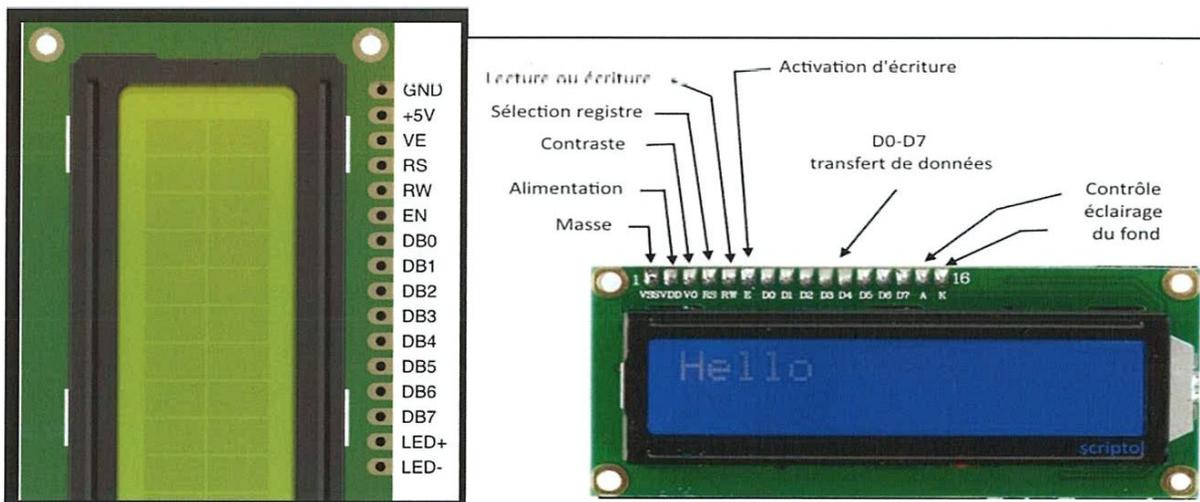


Figure II.28 : Carte de développement d'interface I2C

3.7.2. Le module LCD

- ❖ à 2 lignes de 16 caractères (blancs sur fond bleu) avec interface I2C/TWI est compatible Arduino et Gadgeteer.
- ❖ L'interface i2c est très pratique en comparaison avec l'interface parallèle couramment utilisée sur ce genre d'afficheur, puisqu'elle utilise beaucoup moins de broches (2 voies + alimentation + GND).

- ❖ Ce module peut être alimenté en 5V via le connecteur 4 broches ou en 3.3V via le connecteur IDC10.

L'alimentation en 3.3V par le connecteur 4 broches nécessite l'utilisation d'un convertisseur de niveau logique .Il est alors possible d'utiliser ce module non seulement avec des systèmes 5V (ArduinoUno, Méga, Pro Mini,...) mais aussi des systèmes 3.3V (Carambola, Raspberry Pi, BeagleBone, Arduino Due,...).

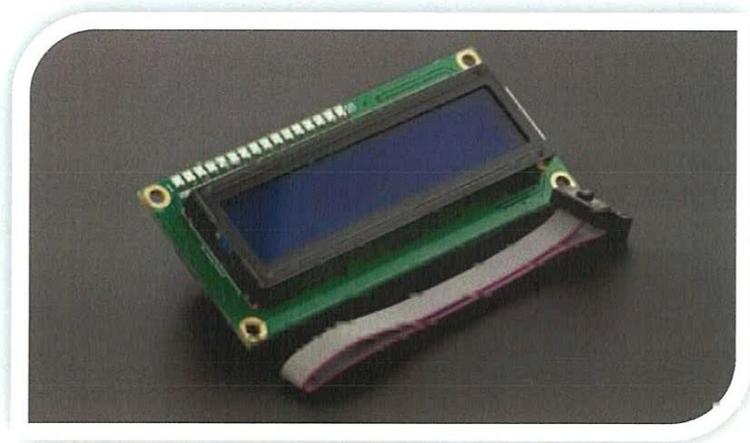


Figure II.29: LCD 16x2 interface i2c

3.7.3. Spécifications

- ❖ Adresse i2c: 0x20-0x27 (0x20 par défaut).
- ❖ Nombre de caractères : 16 caractères x 2 lignes.
- ❖ Caractères blancs sur fond bleu.
- ❖ Contraste ajustable.
- ❖ Alimentation : 5V (via connecteur 4 broches) ou 3.3V (via connecteur IDC10).
- ❖ Interface: IIC/TWI x1, IDC10 x2
- ❖ Taille: 82x35x12mm.

4. PARTIE SOFTWARE DE LA TISSEUSE CIRCULAIRE

4.1. DESCRIPTION DE L'APPLICATION

Notre application à l'objectif de gérer la production et la maintenance avec fiabilité et instantanément, et pour ce but on a créé un réseau Ethernet qui fait connecter la machine avec une interface graphique pour la visualisation des différents défauts et la production en temps réel. La machine envoie les informations au module Arduino, Le serveur reçoit les informations de l'Arduino à travers le réseau Ethernet (module Ethernet) utilisant l'architecture client-serveur.

4.2. DESCRIPTION DE L'ALGORITHME

La machine envoie à l'Arduino plusieurs informations, ce dernier renvoie ces informations sur le réseau après 5s de temps le serveur affiche les informations sur l'interface graphique

On va expliquer ces opérations en utilisant l'algorithme suivant .

Algorithme

Si on appui sur le bouton marche

Démarrer la machine

Si on appui sur le bouton arrêt

Arrêter la machine

Tant que le bouton de marche impulsioneelle est toujours appuyée

Démarrer la machine

Sinon

Arrêter la machine

Fin

Si il y'a une coupure dans une zone

Allumer la Led

Arrêter la machine

Incrémenter la coupure de la zone coupée

Sinon

Eteint la Led

Fin

Si le capteur lire une impulsion

Incrémenter la valeur $v+= 0.435$

Si le temps de lecture de capteur dépasser 5s

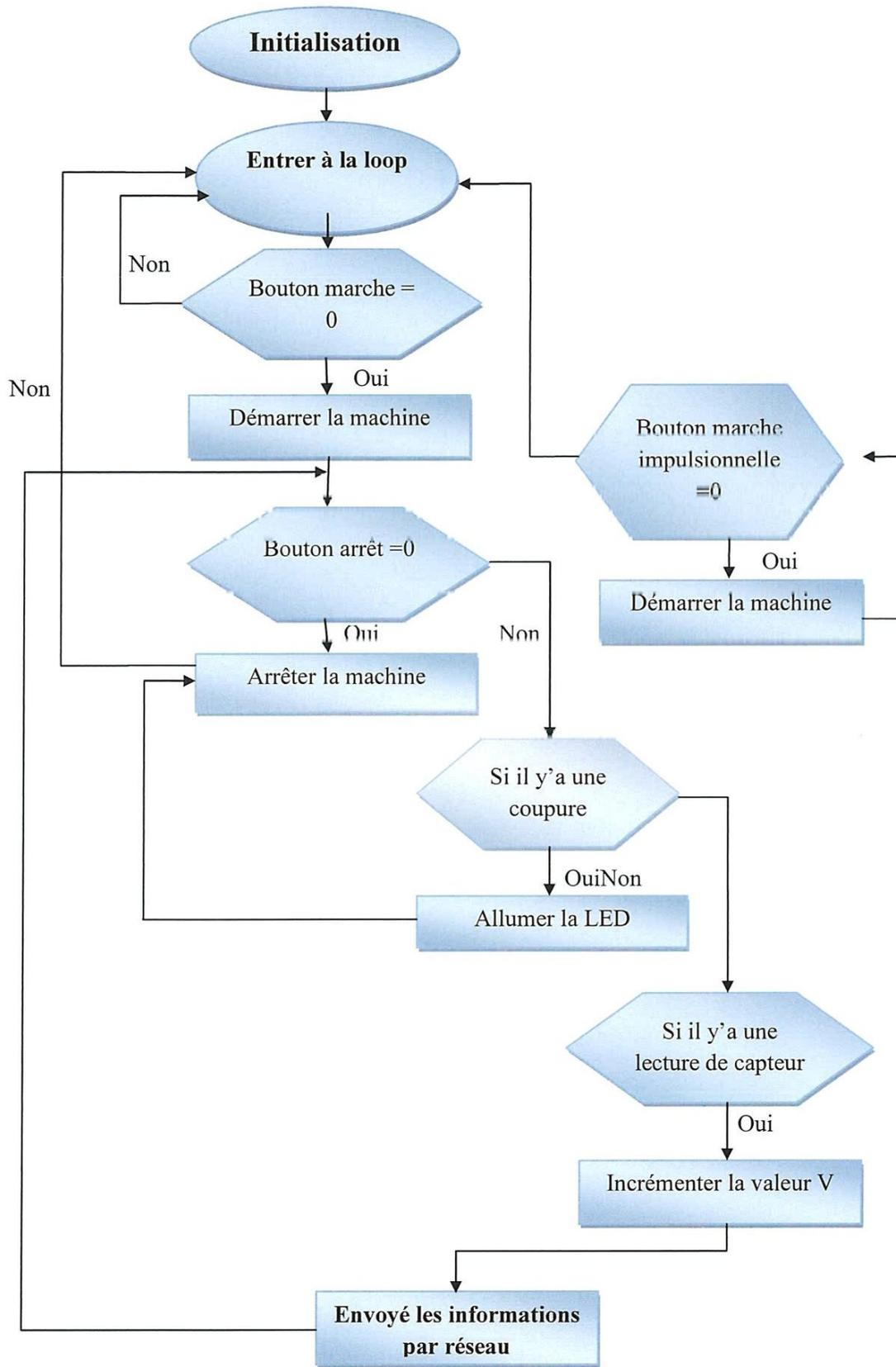
Arrêter la machine

Allumer les 3 Led

Envoyer les informations par réseau.

4.3. ORGANIGRAMME

Nous présentons ci-dessous l'organigramme qui permet de gérer la commande de la machine tisseuse circulaire :



4.4. CONCEPTION

4.4.1. Interface graphique

Figure II.30: Interface graphique.

L'interface graphique se compose de :

Plusieurs champs qui nous permettent d'afficher les informations reçues par le réseau,

- ✚ On a des champs pour afficher les coupures dans chaque zone.
- ✚ D'autres champs pour afficher le nombre de coupure pour chaque zone.
- ✚ des champs dans lesquels on va afficher l'état de la machine, le moteur principal et le variateur de vitesse.
- ✚ Deux champs pour l'affichage de rendement et le débit de la machine sous forme des graphes.
 - Un graphe démonstratif en bas en fonction de débit /temps.
 - Un graphe démonstratif en haut en fonction de rendement/ temps.

4.5. IMPLEMENTATION

4.5.1. Langage de programmation

Afin de mettre en œuvre notre interface , nous avons choisi le langage de programmation C# parce qu'il est conçu pour être un langage de programmation simple, moderne, à usage général et de type sécurisé et orienté objet qui permet aux programmeurs en entreprise de créer une vaste gamme d'applications, empruntant des concepts clés à partir de plusieurs autres langues, notamment Java.

4.5.2. L'environnement de développement

L'environnement de développement intégré (IDE) Visual C# est une collection d'outils de développement exposée à travers une interface utilisateur ordinaire. Certains outils sont partagés avec d'autres langages Visual Studio tandis que d'autres, tels que le compilateur C#, sont propres à Visual C#. Cette rubrique fournit des liens vers les outils Visual C# les plus importants.

Nous avons utilisé la version visuelle studio 2013.

4.5.3 Bibliothèque

✚ **Ethernet** : Avec Arduino Ethernet Shield, cette bibliothèque permet à une carte Arduino de se connecter à Internet. Il peut servir comme un serveur acceptant les connexions entrantes ou un client faisant des clients sortants. La bibliothèque prend en charge jusqu'à quatre connexions simultanées (entrantes ou sortantes ou combinées).

Arduino communique avec le Shield en utilisant le bus SPI.

✚ **SPI** : Serial Peripheral Interface (SPI) est un protocole de données série synchrone utilisé par les microcontrôleurs pour communiquer avec un ou plusieurs périphériques rapidement sur de courtes distances. Il peut également être utilisé pour la communication entre deux microcontrôleurs. (l'Arduino étant le périphérique principal).

Généralement, il existe trois lignes communes à tous les périphériques:

- MISO (Master In Slave Out) - La ligne esclave pour l'envoi de données au maître,
- MOSI (Master Out Slave In) - La ligne Master pour l'envoi de données sur les périphériques,
- SCK (Serial Clock) - Les impulsions d'horloge qui synchronisent la transmission de données générée par le maître

Et une ligne spécifique pour chaque périphérique:

- SS (Slave Select) - la broche sur chaque périphérique que le maître peut utiliser pour activer et désactiver des périphériques spécifiques.

4.5.4. Système d'exploitation :

Afin de mettre en œuvre notre application, nous avons utilisé le système d'exploitation Microsoft Windows 8.1, 64-bit.

5. CONCLUSION

Dans ce chapitre Nous avons donné un bref aperçu de la carte à microcontrôleur et le module Ethernet ainsi que l'I2C .Par la suite nous avons présenté les méthodes et les équipements pour appliquer notre système de surveillance des machines et notre propre conception et en suite notre implémentation. Enfin, nous avons décrit l'algorithme et l'organigramme de la commande de cette machine.



CHAPITRE III :

Test Pratique et Résultat

1. INTRODUCTION

La réalisation pratique d'une machine de production des sacs a tissage circulaire Qui s'appelle « La tisseuse circulaire » par une carte électronique et la carte Arduino (programmation du μc) ainsi que le module ETHERNET (création du réseau Ethernet qui connecte la machines avec une interface graphique).

Dans ce chapitre, nous présentons un aperçu général sur les problèmes qu'on à rencontrer au long des tests pratiques ; et Suivi les résultats obtenu par notre système et la correction des erreurs. On va faire une étude sur les règles utilisées lors d'une conception de cette application.

2. TEST DES ENTREES SORTIES et LA TRANSMISSION DES DONNEES PAR RESEAU

2.1. Vérification des entrées

✚ Une entrée est activée si elle est passée de l'état 1(haut) à l'état 0 (bas).

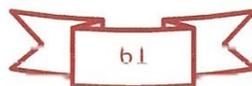
➤ Pour cela un pseudo-programme a été écrit pour faire cette vérification.

```

z1 = digitalRead (pz1);           Serial.print ("z1:"); Serial.println (z1);
z2 = digitalRead (pz2);           Serial.print ("z2:"); Serial.println (z2);
z3 = digitalRead (pz3);           Serial.print ("z3:"); Serial.println (z3);
z4 = digitalRead (pz4);           Serial.print ("z4:"); Serial.println (z4);
z5 = digitalRead (pz5);           Serial.print ("z5:"); Serial.println (z5);
z6 = digitalRead (pz6);           Serial.print ("z6:"); Serial.println (z6);
m= digitalRead (pm);              Serial.print("m:"); Serial.println (m);
v = digitalRead (pv);              Serial.print("v:");Serial.println (v);
sti= digitalRead (mip);            Delay(500);
start= digitalRead (mar);
stopp= digitalRead (arr);

```

✚ Le moniteur série nous permet d'afficher les informations transmises via USB par Arduino pour observer les états des entrées.



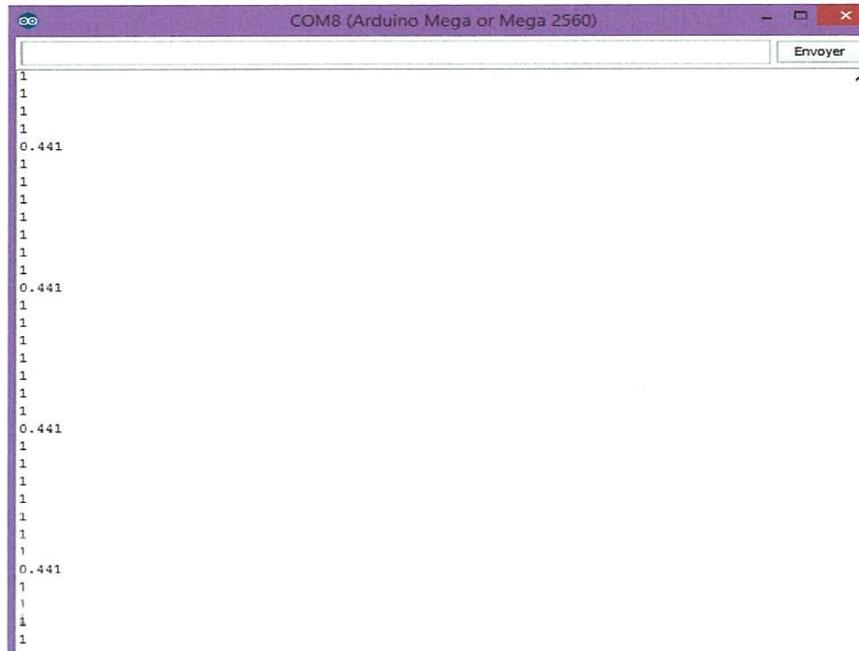


Figure III.1 : Affichage de l'état des entrées dans le moniteur série

✚ L'affichage des informations sur l'interface transmis par réseau Ethernet

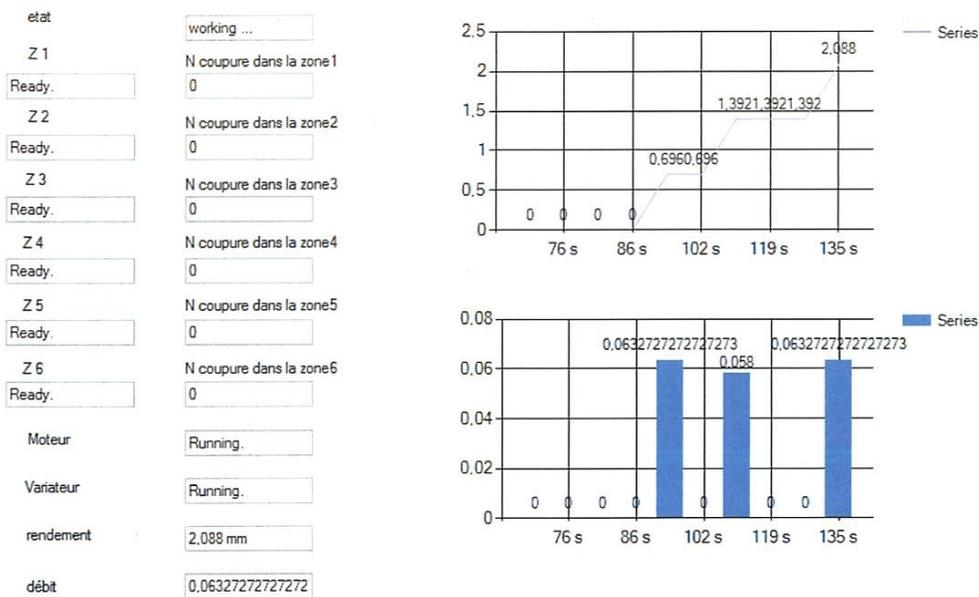


Figure III.2 : Interface de l'application

2.2. Vérification des sorties :

✚ Pour vérifier le bon fonctionnement des sorties, un programme relie l'état de chaque sortie avec une entrée. Donc pour chaque entrée activée, une sortie va être excitée. Et la vérification se fait d'une manière visuelle.

➤ Ce programme a été écrit pour faire cette vérification des sorties :

<pre>Start= digitalRead (mar); digitalWrite (A8,Start);</pre>	}	<ul style="list-style-type: none"> ✓ le bouton marche appuyé ✓ le relai 1 excité
---	---	--

<pre>sti= digitalRead (mip); digitalWrite (A14,HIGH);</pre>	}	<ul style="list-style-type: none"> ✓ le bouton marche impulsionnelle appuyé ✓ le relai 2 excité
---	---	---

<pre>z1 = digitalRead (pz1) digitalWrite (A0,HIGH);</pre>	}	<ul style="list-style-type: none"> ✓ Zone 1 coupée ✓ La led 1 allumée
---	---	---

<pre>z2 = digitalRead (pz2) digitalWrite (A1,HIGH);</pre>	}	<ul style="list-style-type: none"> ✓ Zone 2 coupée ✓ La led 2 allumée
---	---	---

<pre>z3 = digitalRead (pz3) digitalWrite (A2,HIGH);</pre>	}	<ul style="list-style-type: none"> ✓ Zone 3 coupée ✓ La led 3 allumée
---	---	---

<pre>z4 = digitalRead (pz4) digitalWrite (A3,HIGH);</pre>	}	<ul style="list-style-type: none"> ✓ Zone 4 coupée ✓ La led 4 allumée
---	---	---

<pre>z5 = digitalRead (pz5) digitalWrite (A4,HIGH);</pre>	}	<ul style="list-style-type: none"> ✓ Zone 5 coupée ✓ La led 5 allumée
---	---	---

<pre>z6 = digitalRead (pz6) digitalWrite (A5,HIGH);</pre>	}	<ul style="list-style-type: none"> ✓ Zone 6 coupée ✓ La led 6 allumée
---	---	---

2.2.1. Test bouton Marche

On procède à un premier traitement du démarrage de la machine:

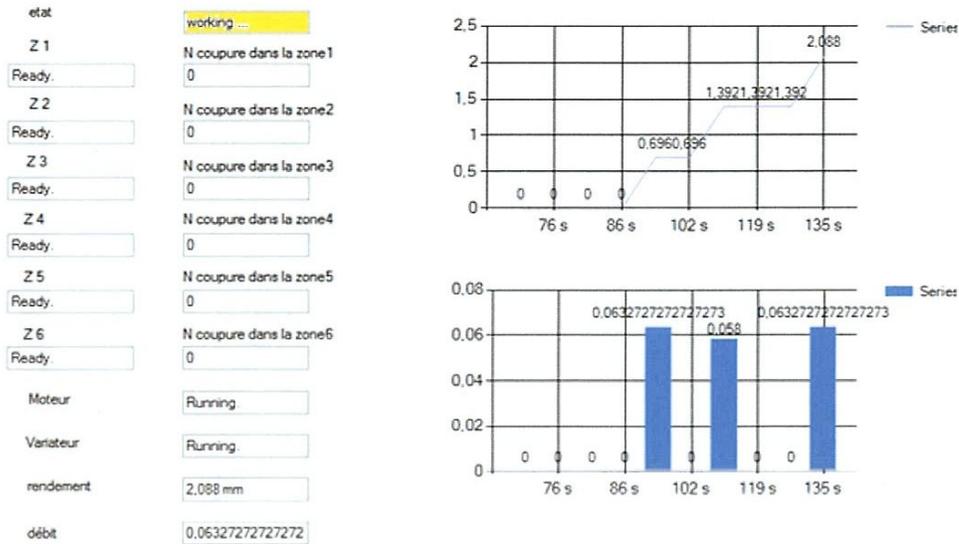


Figure III.3 : Affichage du démarrage sur l’interface graphique

2.2.2. Test bouton arrêt:

La figure qui suit montre que la carte électronique se désactive quand on éteint la machine.

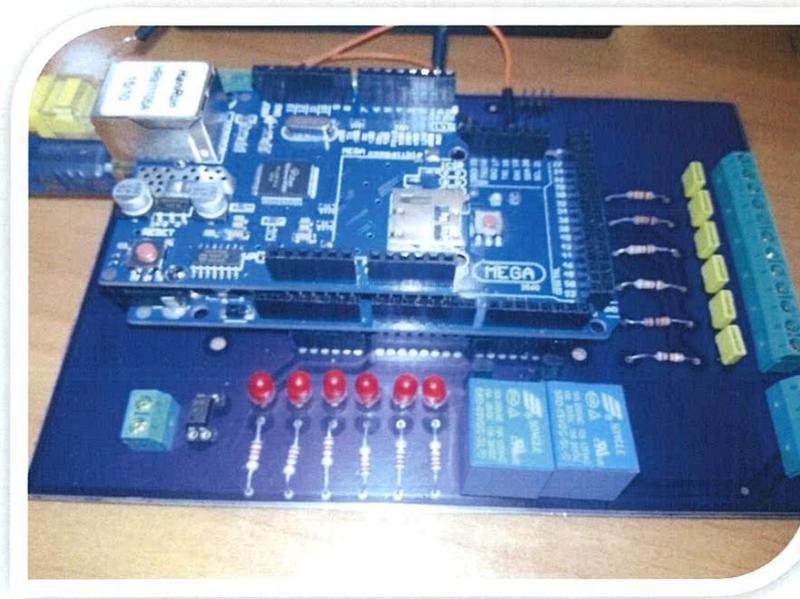
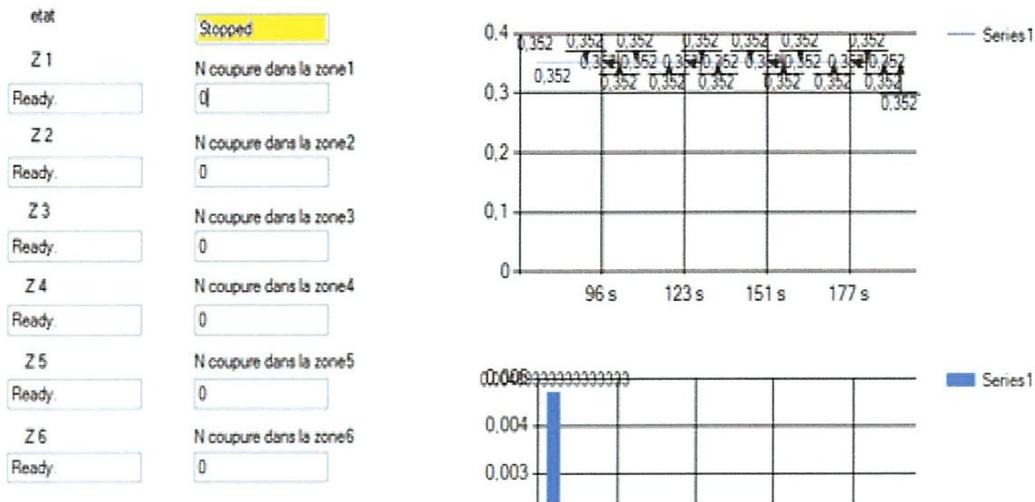


Figure III.4: La carte de commande à l’état arrêt



FigureIII.5:Affichage de l'arrêt sur l'interface graphique

2.2.3. Coupure dans la zone 1 :

Sur la figure ci-dessous on observe que l'une des leds (celle de la zone 1) est allumée. Cela indique qu'une coupure a été effectuée sur ce niveau.



FigureIII.6 :Coupure de la zone 1.

Après avoir simulé une coupure au niveau de la première zone, on a constaté qu'elle est affichée sur l'interface graphique.

A chaque fois qu'on refait la même procédure, on remarque l'incrémentation du nombre des coupures dans la zone coupée.

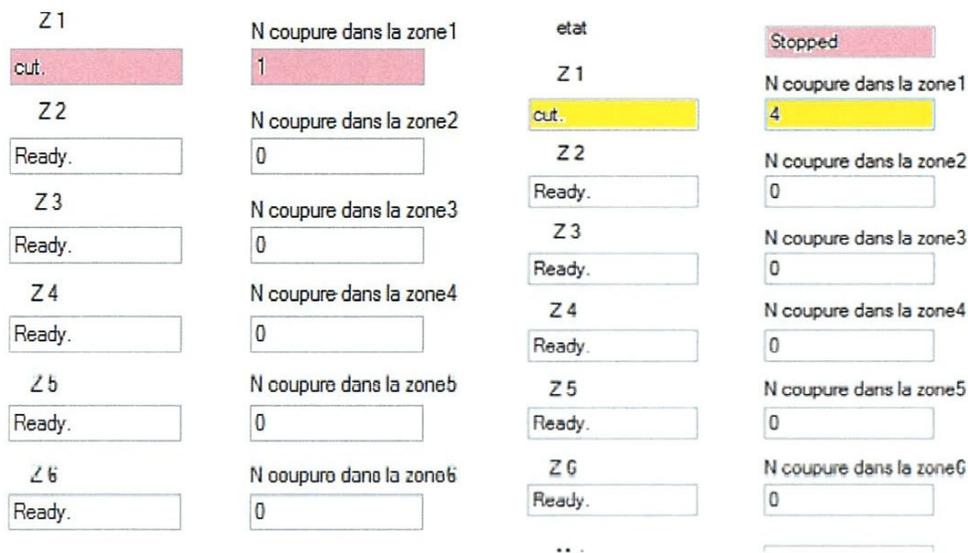


Figure III.7 : Affichage de coupure sur l'interface graphique

Après avoir effectué la coupure, et relancé toute la procédure de nouveau, on a constaté que les anciennes coupures restent mémorisées.

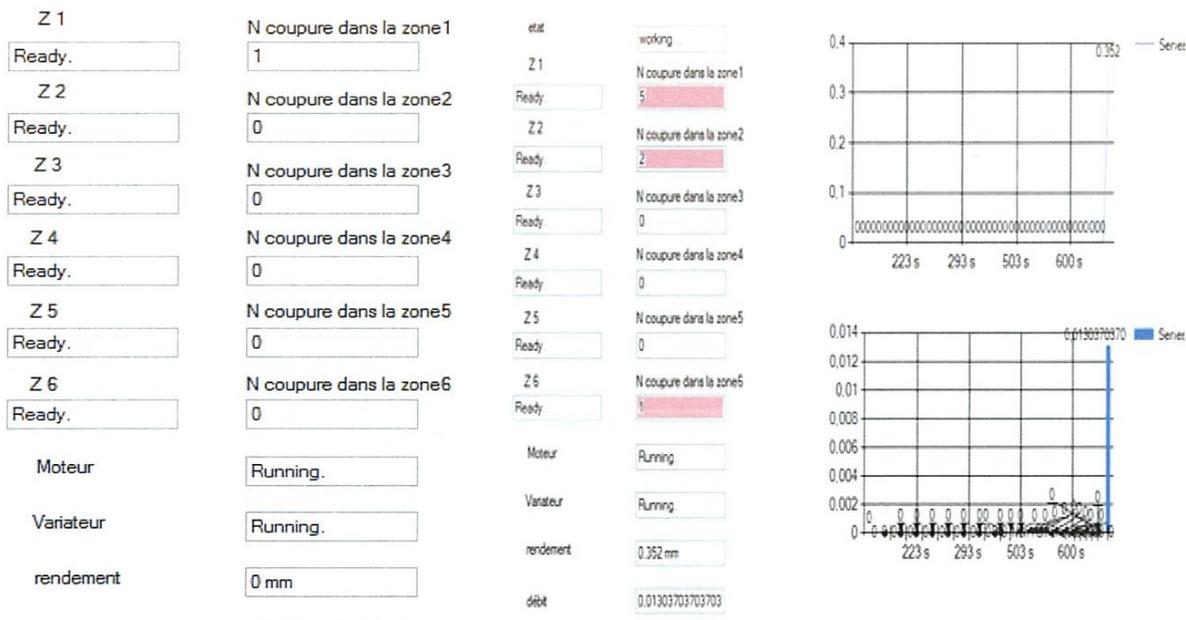


Figure III.8: Affichage du nombre de coupure sur l'interface graphique

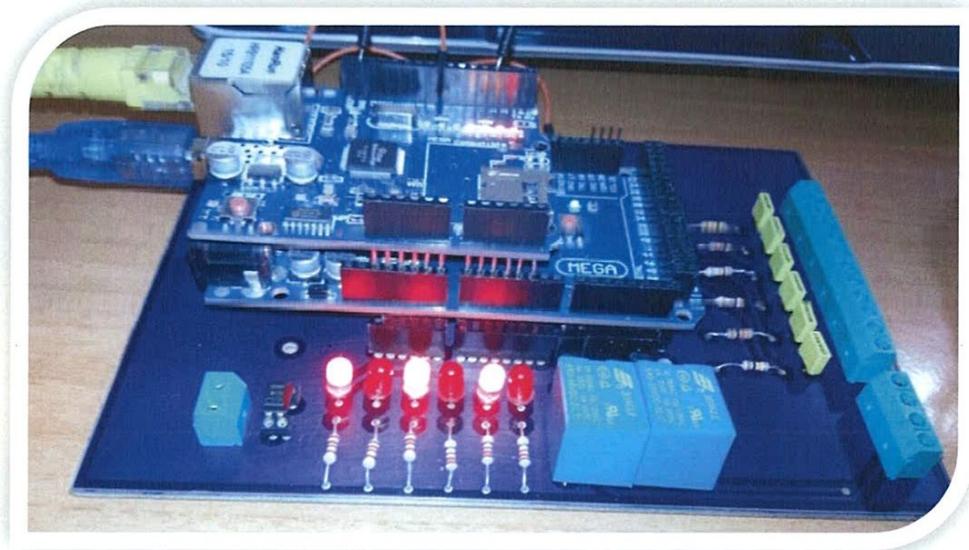
3. TEST DES EVENEMENTS PARALLELE ET LA TRANSMISSION DES DONNEES PAR RESEAU

Notre système est séquentiel, donc il faut tester le cas ou en a des évènements parallèle pour vérifier le bon fonctionnement de la carte dans ce cas.

3.1. Cas de coupure de plusieurs zones au même temps

Pour vérifier s'il est possible de lancer plusieurs coupures, on a excité 3 zones (zone 1, zone3, zone5) au même temps

On a remarqué que 3 Leds ont été allumé.



FigureIII.9:Coupure des 3 zones

Les 3 zones mentionnées précédemment, il s'est affiché au niveau de l'interface graphique 3 coupures au même temps.

etat	Stopped
Z 1	N coupure dans la zone1
cut.	1
Z 2	N coupure dans la zone2
Ready.	0
Z 3	N coupure dans la zone3
cut.	1
Z 4	N coupure dans la zone4
Ready.	0
Z 5	N coupure dans la zone5
cut.	1
Z 6	N coupure dans la zone6
Ready.	0

FigureIII.10:Affichage de coupure 3zones en parallèle sur l’interface graphique

3.2. Simulation de la défaillance du capteur

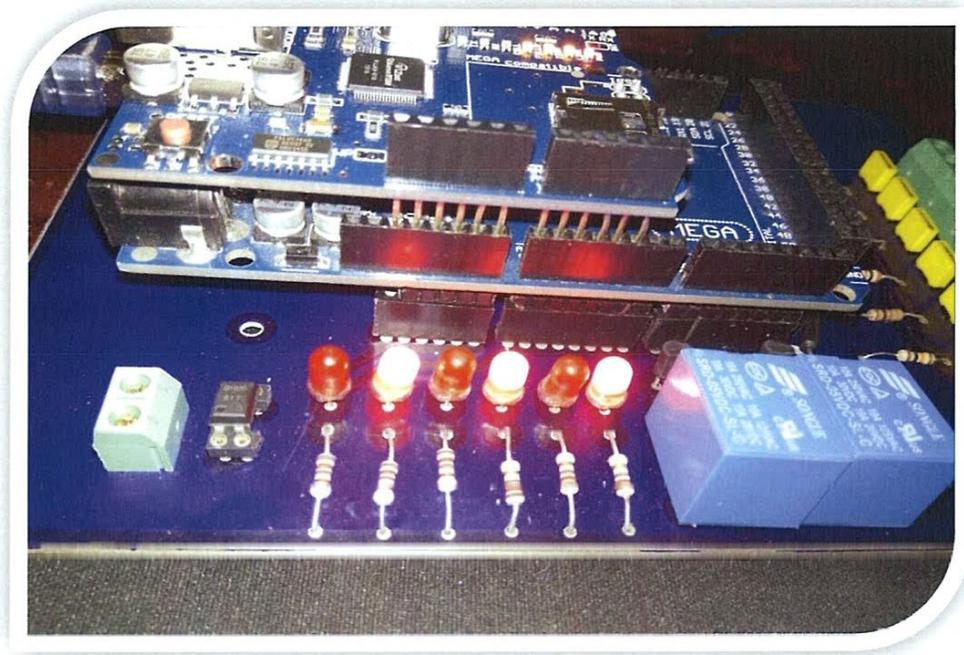
Pour simuler la défaillance du capteur, on va maintenir la lecture de ce dernier. Et voir l’influence de cette action sur le fonctionnement de la carte.

✚ Observation

Blocage de toutes les commandes jusqu’à le débranchement du capteur. Donc le programme est bloqué dans l’instruction de lecture du capteur.

✚ La solution

Création d’un chien de garde qu’il va compter 3 second et sortir de l’instruction pour continuer l’exécution du reste du programme. Et en cas on trouve le même problème, la machine va être arrêtée et un défaut de capteur est signalé par allumage de 3 LED.



FigureIII.11: Allumage de 3 LED de capteur

4. CONCLUSION

Après la réalisation pratique, et plusieurs tests nous affirmons que nous avons achevé notre tâche demandée: commande d'une Tisseuse Circulaire à base d'un ARDUINO et le Réseau ETHERNET.

L'avantage de notre application :

- ✓ Suivre la production et les pannes ainsi que les erreurs qui puissent surgir.
- ✓ Faciliter l'accès direct à ce réseau et la fluidité de la gestion à distance.

Nous avons rencontré plusieurs problèmes durant la conception et la réalisation pratique mais grâce à ce que nous avons appris durant nos études académiques, nous pouvions confronter ces difficultés.

Conclusion Générale

Conclusion Générale

Le travail présenté dans ce mémoire a porté sur la réalisation d'une carte IDO pour la surveillance des machines. Ce projet nous a permis de découvrir et d'appréhender les défis liés à la conception et à la réalisation des systèmes électroniques du point de vue compatibilité, choix de composantsetc.

Les résultats obtenus de ce travail ont été satisfaisants :

- la détection des commandes et des alarmes avec fiabilité.
- Transmission des données par réseau sans rupture.
- affichage des données et des informations en temps réel.

En perspectives, nous pouvons signaler que ce travail peut être plus autonome, plus pratique, et assez évolutif vu le pas géant et les progrès réalisés dans les domaines de la technologie aussi l'évolution technique des microsystèmes électromécaniques. Nous pouvons envisager des améliorations en ce qui concerne :

- Réduire la taille des circuits, on pourra utiliser des composants électroniques CMS.
- Améliorer le temps de repense du système utilisant des processeurs (exécution parallèle)
- Utiliser des applications sur Smartphone pour faciliter l'accès à l'information.

Liste des figures

CHAPITRE I

Figure I.1 : Historique de la Technologie : la connectivité des choses.	2
Figure I.2 : La voiture connectée est en route.....	4
Figure I.3 : L'Industrie se tourne vers l'usine intelligente.....	5
Figure I.4 : Cas et Services de l'Internet des objets	6
Figure I.5 : les différentes topologies a= Point-à-Point; b= Étoile; c= Arbre ; d= Maillé.....	7
Figure I.6 : Schéma des éléments principaux d'un microcontrôleur.....	12
Figure I.7 : Signal analogique par la technique PWM	12
Figure I.8 : Les types des microcontrôleurs Les microcontrôleurs de 8, 16 et 32 bits.....	12
Figure I.9 : L'architecture d'un microcontrôleur.....	13
Figure I.10 : Les appareils de communication.....	14
Figure I.11 : Appareil médicale.....	14
Figure I.12 : Appareil de sécurité.....	14
Figure I.13 : Les appareils industriels.....	15
Figure I.14 : Les moyens de transports.....	15
Figure I.15 : TV, vidéo, HIFI.....	15
Figure I.16 : Enregistreur de données géographiques à pile et système de fixation.....	15
Figure I.17 : Prototype de l'Arduino.....	17

Figure I.18 : Différent types des capteurs pour l'Arduino.....	17
Figure I.19 : Différents actionneurs pour l'Arduino.....	18
Figure I.20 : Exemples de différents types de cartes Arduino.....	18
Figure I.21 : Caractéristique d'Arduino Uno.....	19
Figure I.22 : Alimentation Arduino Mēga 2560.....	19
Figure I.23 : Interface de logiciel Arduino.....	20
Figure I.24 : l'architecture du modèle TCP/IP.....	25
Figure I.25 : le modèle TCP/IP-OSI	28
Figure I.26 : Le schéma du protocole TCP/IP.....	28
Figure I.27 : Architecture du modèle TCP/IP.....	30
Figure I.28 : Carte du module Ethernet.....	31
Figure I.29 ; Architecture interne du module Arduino Ethernet.....	32
Figure I.30 : Complémentarité entre Ethernet Et TCP/IP	33
Figure I.31 : Interface de logiciel c#.....	37
Figure I.32 : Barre d'outils et commandes de menu	37
Figure I.33 : solution explorer.....	38
Figure I.34 : Propriétés pour les solutions, les projets et les fichiers.....	38

CHAPITRE II

Figure II.1 : Les bobines /ou Les antennes.....	39
Figure II.2 : Les navettes.....	40
Figure II.3 : Le moteur principal.....	40
Figure II.4 : Boite de commande.....	40
Figure II.5 : Capteur De Proximité.....	41
Figure II.6 : moteur de puissance qui récupéra le rouleau.....	41

Figure II.7 : Un variateur de vitesse.....	41
Figure II.8 : Carte de développement ARDUINO Méga 2560	43
Figure II.9 : Architecture interne du microcontrôleur ATmega 2560	44
Figure II.10 : schéma des entrées logiques.....	45
Figure II.11 : schéma d'entrée d'alarme	45
Figure II.12 : schéma d'entrée de Bouton	46
Figure II.13 : schéma d'un Optocoupleur.....	47
Figure II.14 : schéma d'une zone.....	47
Figure II.15 : schéma des sorties logiques.....	48
Figure II.16 : schéma du Relais.....	48
Figure II.17 : Symboles du relais.....	48
Figure II.18 : Schéma de LED.....	49
Figure II.19 : Le L298N se décline dans un boîtier Multi watt à 15 pins.....	50
Figure II.20 : Structure interne de L-298.....	50
Figure II.21 : Brochage du BC547C-B-E (collecteur, base, émetteur).....	51
Figure II.22 : Symbole / Brochage du Résistance.....	52
Figure II.23 : Quelque type de condensateur simplifié « électrodes ».....	53
Figure II.24 : Quelques Exemples de différents condensateurs.....	53
Figure II.25 : Carte de développement de module Ethernet.....	54
Figure II.26 : le module Ethernet.....	55
Figure II.27 : Carte de développement d'interface I2C.....	56
Figure II.28 : LCD 16x2 interface i2c.....	57

Figure II.29 : Interface graphique61

CHAPITRE III

Figure III.1: Affichage de l'état des entrées dans le moniteur série.....65

Figure III.2: Interface de l'application.....65

Figure III.3 : Affichage du démarrage sur l'interface graphique.....67

Figure III.4: La carte de commande à l'état arrêt.....67

Figure III.5: Affichage de l'arrêt sur l'interface graphique.....68

Figure III.6: Coupure de la zone 1.....68

Figure III.7: Affichage de coupure sur l'interface graphique.....69

Figure III.8: Affichage du nombre de coupure sur l'interface graphique.....69

Figure III.9: Coupure des 3 zones.....70

Figure III.10: Affichage de coupure 3zones en parallèle sur l'interface graphique.....71

Figure III.11: Allumage de 3 LED de capteur72

Liste des Tableaux

CHAPITRE I

Tableau I.1 : Les développements historiques dans les produits de microprocesseurs.....	11
Tableau I.2 : Différent types des cartes Arduino.....	18
Tableau I.3 : Historique de c Sharpc.....	36

CHAPITRE II

Tableau II.1 : Fonctionnement de Le circuit intégré L298	51
--	----

Les Abréviations

IDO	L'Internet Des Objets
MIT	Massachusetts Institute Of Technology
RFID	Radio Frequency Identification
TICs	Technologies De l'Information Et La Communication.
WIFI	Wireless fidelity
M2M	Machine To Machine
GSM	Global System For Mobile Communication
TCP/IP	Transmission Contrôle Protocole/ Internet Protocole
UDP	User Datagram Protocol
VTC	Voiture De Transport Avec Chauffeur
MIT	Massachusetts Institute Of Technology
IPv6	Internet Protocol Version 6
IEEE	Institute Of Electrical And Electronics Engineers
ROM	Read Only Memory
RAM	Random Access Memory
SRAM	Static Random Access Memory
EPROM	Programmable Effaçable Read Only Memory
PROM	Programable Read Only Memory
HIFI	Hight Fidelity
CMOS	Complementary Metal Oxide Semiconductor
MCS	Microcontroller Series
PIC	Peripheral Interface Controller
ATMEL	American Manufacturer Of Semi-conductors
AVR	Alf & Vegard's RISC Processor
CPU	Central Processing Unit
UAL	Unité Arithmétique Et Logique
LED	Light -Emeting Diode

CAN	Convertisseur Analogique-Numérique
CNA	Un Convertisseur Numérique-Analogique
RTC	Compteur En Temps Réel
PW	Pulse Width Modulation
I/O	In/Out
UC	Unité Centrale
UNIX	Uniplexed Information And Computing Service
PC IBM	Personal Computer Of International Busines Machines
ANSI	American National standards Institute
PDA	Personal Digital Assistant
RTC	Compteur En Temps Réel
SPI	Serial Peripheral Interface
JTAG	Joint Test Action Groupe
RX	Récepteur
TX	Emetteur
USB-TTL	Universal Serial Bus- Time To Live
MISO	Master In Slave Out
MOSI	Master Out Slave In
SCK	Serial Clock
SS	Slave Select
TWI	Two Wire Interface
SDA	Serial Data Line
SCL	Serial Clock Line
AREF	Analogue REference
IDE	Integrated Development Environment
avr-gcc	Alf & Vegard's RISC Processor - GNU Compiler Collection
GPS	Global Positioning System
OSI	Open Systems Interconnection
ARPAET	Advanced Research Projects Agency Network
ARPA	Advanced Research Projects Agency
DOD	Department Of Defense
TFTP	Trivial File Transfer Protocol
SMTP	Simple Mail Transfer Protocol
HTTP	Hypertext Transfer Protocol
HTTP	Hypertext Transfer Protocol Sécurisé
FTP	File Transfer Protocol
DNS	Domaine Name System
FDDI	Fiber Distributed Data Interface
xDSL	Digital Subscriber Line
ATM	Asynchrone Transfer Mode
LAN	Local Area Network
WAN	Wide Area Network

Les Abréviations

HTML	Hypertext Markup Language
PARC	Palo Alto Research Center
IAP	Internet Access Provider
RPV ou VPN	Virtual Private Network
USART	Universal Asynchronous Receiver Transmitter

REFERENCE BIBLIOGRAPHIQUE

- [1] BASSEL ASSAF ; MARTIAL ABISSI, « L'Internet Des Objets - Est-Il La Prochaine Evolution De l'Internet ? », Réalisé Le 24 Septembre 2015, Publié Le 7 Déc. 2015 Sur <https://fr.slideshare.net/martialabissi/Internet-Des-Objets-55908847>.
- [2] LOGHMARI DHIA EDDINE, SEFFEN NARDJES, SGHIR MARWA ET LATIRI WAEL, L'internet Des Objets Présentation D'exposé Publié Le 9 Avril 2016 Sur <https://fr.slideshare.net/electronomica/Internet-Des-Objets-60703444>, Université De Sousse, Réalisé Le 4 Avril 2016.
- [3] <http://jeanchavanat.wixsite.com/les-objets-connectes/-propos1-cec3>
- [4] <https://universitesesame.com/linternet-of-things-iot-levier-de-croissance-economique>,
- [5] <https://www.1min30.com/dictionnaire-du-web/objet-connecte>
- [6] Auteur DAVE EVANS, ' L'Internet des objets, Comment l'évolution actuelle d'Internet transforme-t-elle le monde ?', Livre blanc, Avril 2011, Cisco Internet Business Solutions Group (IBSG).
- [7] HOCINE TAKHI, « Conception et réalisation d'un robot mobile à base d'arduino », mémoire master, Université Amar Telidji, instrumentation 2014-2015.
- [8] GUERUI R, «Etude réalisation et programmation d'un bras manipulateur à base d'arduino » mémoire master, Université 8 mai 45, 2015-2016.
- [9] <http://repository.root-me.org/R%C3%A9seau/FR%20-%20Mod%C3%A8le%20TCP-IP.pdf>
- [10] <https://www.gralon.net/articles/internet-et-webmaster/logiciel/article-le-protocole-tcp-ip---presentation-et-fonctionnement-1597.htm>,
- [11] <http://www.volle.com/travaux/tcpip.htm> , 23 août 2004, site de Michel Volle.

- [12] SEBASTIENVARRETTE, NICOLAS BERNARD, 'Cours de programmation avancée'. Le langage C', Université du Luxembourg 2005–2006. https://repo.zenk-security.com/Programmation/cours_C.pdf
- [13] BILEL 183, C# et .NET version1 a 4, publier le 6 octobre 2012
<https://fr.slideshare.net/bilel183/c-et-net>
- [14] ZAN, cour c#, Publier le 2/04/2003, <https://fr.slideshare.net/webzan2008/cours-c-27336645>
- [15] <http://www.guide-systemes-alarmes.be/quest-ce-quun-systeme-dalarme/>
- [16] <https://www.astuces-pratiques.fr/electronique/le-relais-principe-de-fonctionnement>.
- [17] MOUHCINE NID BELKACEM, projet de fin d'année 2^{ème} année du cycle d'ingénieur systèmes électriques et télécommunication," conception et réalisation des interfaces de puissance et gestion de l'alimentation d'un micro-drone", publier le 5 mars 2014.
<https://fr.slideshare.net/moumoutoubkal/ppt-pfa-nidbelkacem>
- [18] <http://www.futura-sciences.com/maison/definitions/maison-resistance-10889/>.

Annexe

Programme sous Arduino IDE :

```
#include <SPI.h>
#include <Ethernet.h>
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress server(192,168,1,200);
IPAddress ip(192,168,1, 222);
IPAddress gateway(192,168,1, 1);
IPAddress subnet(255, 255, 255, 0);
EthernetClient client;
double value = 0.0;
int z1,z2,z3,z4,z5,z6,m,v;
int pz1 = 27, pz2 = 29, pz3=31, pz4 = 33, pz5 = 35, pz6 =37, pm = 36, pv = 9, arr = 26, mip =28,
mar = 34;
int start=1 ,stopp=1 ,sti=1;
int t=0;
float va=0;
String data;
boolean marche=false,flag=false ;
void setup() {
  Serial.begin(9600);
  while (!Serial) {
    ; }
  pinMode(3,INPUT_PULLUP);
  pinMode(pz1,INPUT_PULLUP);
  pinMode(pz2,INPUT_PULLUP);
  pinMode(pz3,INPUT_PULLUP);
  pinMode(pz4,INPUT_PULLUP);
  pinMode(pz5,INPUT_PULLUP);
```

```
pinMode(pz6,INPUT_PULLUP);
pinMode(pm,INPUT_PULLUP);
pinMode(pv,INPUT_PULLUP);
pinMode(arr,INPUT_PULLUP);
pinMode(mip,INPUT_PULLUP);
pinMode(mar,INPUT_PULLUP);
pinMode(A0,OUTPUT);
pinMode(A1,OUTPUT);
pinMode(A2,OUTPUT);
pinMode(A3,OUTPUT);
pinMode(A4,OUTPUT);
pinMode(A5,OUTPUT);
pinMode(A8, OUTPUT);
pinMode(A14,OUTPUT);
Ethernet.begin(mac, ip);
Serial.println(Ethernet.localIP());
}
void loop() {
  sti= digitalRead (mip);
  if (!marche &&sti==0){
    digitalWrite (A8,HIGH);
    digitalWrite (A14,HIGH);
    while (sti==0) {
  sti= digitalRead (mip);
  }
    digitalWrite (A8,LOW);
    digitalWrite (A14,LOW);
  }
  start= digitalRead (mar);
  if(start ==0){
  while(start==0)
```

```
{start= digitalRead (mar);
}
if(!marche)
{digitalWrite (A8,HIGH);
digitalWrite (A14,HIGH);
}
marche=true;
}
if (!marche )return ;
stopp= digitalRead (arr);
if(stopp ==0){
while(stopp==0)
{stopp= digitalRead (arr);
}
if(marche)
{digitalWrite (A8,LOW);
digitalWrite (A14,LOW);
}
marche=false;
return ;
}
z1 = digitalRead(pz1);
z2 = digitalRead(pz2);
z3 = digitalRead(pz3);
z4 = digitalRead(pz4);
z5 = digitalRead(pz5);
z6 = digitalRead(pz6);
m= digitalRead(pm);
v = digitalRead(pv);
```

```
if (z1==0)
{
digitalWrite (A0,HIGH);
digitalWrite (A8,LOW);
digitalWrite (A14,LOW);
marche = false;
flag=false;
t=49;
}
else digitalWrite (A0,LOW);
if (z2==0){
digitalWrite (A1,HIGH);
digitalWrite (A8,LOW);
digitalWrite (A14,LOW);
marche = false;
flag=false;
t=49;
}
else digitalWrite (A1,LOW);
if (z3==0){
digitalWrite (A2,HIGH);
digitalWrite (A8,LOW);
digitalWrite (A14,LOW);
marche = false;
flag=false;
t=49;
}
else digitalWrite (A2,LOW);
```

```
if (z4==0){
```

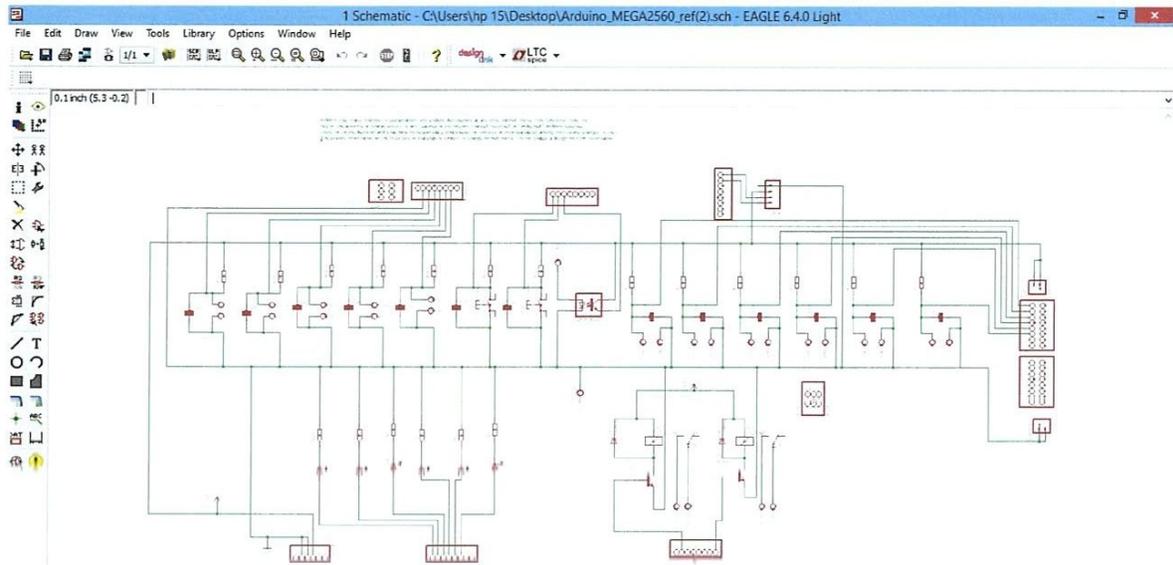
```
digitalWrite (A3,HIGH);
digitalWrite (A8,LOW);
digitalWrite (A14,LOW);
marche = false;
flag=false;
t=49;
}
else digitalWrite (A3,LOW);
if (z5==0){
digitalWrite (A4,HIGH);
digitalWrite (A8,LOW);
digitalWrite (A14,LOW);
marche = false;
flag=false;
t=49;
}
else digitalWrite (A4,LOW);
if (z6==0){
digitalWrite (A5,HIGH);
digitalWrite (A8,LOW);
digitalWrite (A14,LOW);
marche = false;
flag=false;
t=49;
}
else digitalWrite (A5,LOW);
if (digitalRead(3)==0){
int tw =0;
while(digitalRead(3)==0 && tw <50){
tw++;
delay(100);
```

```
}
if(flag && tw>=50){
digitalWrite (A8,LOW);
  digitalWrite (A14,LOW);
  marche = false;
  flag=false;
  digitalWrite (A0,HIGH);
  digitalWrite (A2,HIGH);
  digitalWrite (A4,HIGH);
  return;
} else {
  flag=false;
}
if(tw >=50) flag=true;
va+=0.435;
}
Serial.print(va);
Serial.println(z1);
Serial.println(z2);
Serial.println(z3);
Serial.println(z4);
Serial.println(z5);
Serial.println(z6);
Serial.println(m);
Serial.println(v);
delay(100);
t++;
/////
if(t==50){
  if (client.connect(server, 5000)) {
    value =va;
```

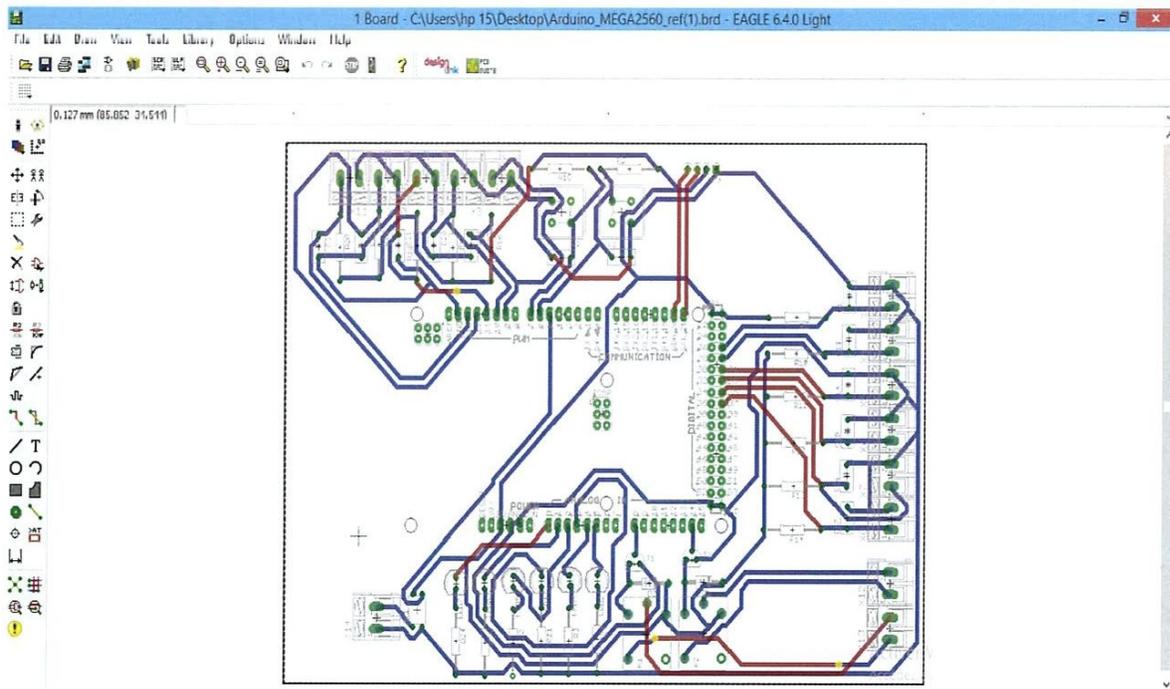
```
Serial.println("connected");
client.println(value);
client.print(z1);
client.print(",");
client.print(z2);
client.print(",");
client.print(z3);
client.print(",");
client.print(z4);
client.print(",");
client.print(z5);
client.print(",");
client.print(z6);
client.print(",");
client.print(m);
client.print(",");
client.println(v);
} else {
  Serial.println("connection failed");
}
client.stop();
t=0;
}
}
```

La carte électronique :

- Schématique



- La carte



- Après l'impression :

