



LICENCE SCIENCES ET TECHNIQUES
Génie Electrique

RAPPORT DE FIN D'ÉTUDES

Intitulé :

Réalisation d'un système d'alarme

Réalisé par :

- ELHADJ Fatimetou
- KALAYDI Nadia

Encadré par :

Pr ABARKAN El Hossain (FST FÈS)

Mr ELKHYATI Naoufel (YAZAKI Kenitra)

Soutenu le 10 Juin 2016 devant le jury

Pr ABARKAN El Hossain (FST FÈS)

Pr EL MARKHI Hassan (FST FÈS)

Pr LAMHAMDI Tijani (FST FÈS)

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

وَأَقْرَبُ أَهْلِكُمْ وَسَيَرَّتِي رَبِّي فَأَسْأَلُكُمْ فِيهَا لَمَّا خَلَّيْتُمْ وَأَنْتُمْ تَأْمُرُونَ وَاللَّهُ يَسْمَعُ الصَّوْتِ وَالنَّوْاحِ وَهُوَ سَمِيعٌ عَلِيمٌ وَأَنْتُمْ لَا تَأْمُرُونَ وَاللَّهُ يَسْمَعُ الصَّوْتِ وَالنَّوْاحِ وَهُوَ سَمِيعٌ عَلِيمٌ وَأَنْتُمْ لَا تَأْمُرُونَ وَاللَّهُ يَسْمَعُ الصَّوْتِ وَالنَّوْاحِ وَهُوَ سَمِيعٌ عَلِيمٌ وَأَنْتُمْ لَا تَأْمُرُونَ

اللَّهُ
الْعَظِيمُ

Dédicace

À ceux qui nous avons indiqué la bonne voie en nous rappelant que la volonté fait toujours les grands hommes et qui ont attendus avec patience les fruits de leur bonne éducation...

« Mes Parents »

À ceux qui nous avons été toujours la garantie d'une existence paisible et d'un avenir radieux...

« Notre belle famille »

À Ceux qui nous avons soutenus, encouragés, appréciés nos efforts et créés le milieu favorable, l'ambiance joyeuse et l'atmosphère joviale pour procurer ce travail.

« Mes chers amis »

À toutes ces personnes que nous sentons redoutables de leur dédier ce modeste travail avec nos vifs remerciements et expressions respectueuses de notre profonde gratitude.

Fatimetou & Nadia

Remerciements

On commencerait, avant tout, ce discours en faisant au Dieu vivant, qui nous a donné la force et la patience nécessaires à l'accomplissement de ce travail, des remerciements solennels.

Nous tenons à exprimer nos sincères gratitudees et nos profondes reconnaissances à toutes les personnes qui nous ont aidées à la réalisation de ce modeste travail.

De prime à bord, nous tenons à remercier, M. ABARKAN pour sa patience, sa diligence et surtout pour sa disponibilité et son encadrement, le temps et les précieux conseils qu'il nous a prodigués tout au long de ce travail.

Ainsi que les membres de jury d'avoir accepté d'évaluer ce travail, ainsi que tous les enseignants de la Faculté des Sciences et Techniques de nous avoir apporté leur savoir-faire, leur expérience et leur disponibilité tout au long de notre formation.

Également, nos remerciements s'adressent à M. ELKHAYATI ingénieur de système informatique à YAZAKI Kenitra pour son encadrement et ses informations.

Enfin, nous remercions Monsieur le responsable de la licence M. JORIO pour la qualité d'enseignement et son effort fourni pour notre formation.

Sommaire

<i>Dédicace</i>	2
<i>Remerciements</i>	3
Sommaire	4
Liste des figures	7
Liste des tableaux	8
Liste des abréviations	8
Introduction générale.....	9
Chapitre 1 : Présentation de l'entreprise	10
Introduction.....	11
I. YAZAKI monde	11
II. YAZAKI Kenitra.....	12
1. Fiche technique YAZAKI Kenitra	12
2. Organigramme YAZAKI Kenitra.....	13
3. Activités principales des services	13
4. Câblage automobile.....	14
4.1. Définition	14
4.2. Composants d'un câble	14
5. Processus de fabrication des câblages automobile.....	15
5.1. La coupe.....	16
5.2. Pré-assemblage	16
5.3. L'assemblage	16
Conclusion	17
Chapitre 2 : Présentation de projet	18
Introduction.....	19
I. Cahier de charge.....	19
1. Acteurs du projet.....	19
2. Contexte pédagogique	19
3. Sujet de projet	19
4. Description	19
5. Contraints à respecter	19
II. La problématique	20
1. Description de la chaine de production des câbles.....	20
2. Les Étapes de la production du câble	20

3. Vision sur la problématique	21
III. Cadrage du problème	22
1. Définir la problématique	22
2. Analyse de problème	23
3. Propositions	24
4. Solution	25
IV. Planning du Travail	25
Conclusion	25
Chapitre 3 : Conception d'un système d'alarme.....	26
Introduction.....	27
I. Description des outils utilisés	27
II. Étude sur l'Arduino.....	27
1. Qu'est-ce qu'Arduino?.....	27
III. Le contenu du projet	31
1. Partie électrique	31
1.2. Disjoncteur	31
1.3. Alimentation stabilisée	31
1.4. Fusible.....	32
1.5. Gyrophare.....	33
2. Partie électronique	33
2.1. Arduino	33
2.2. Capteur de température d'humidité DHT11	33
2.3. Les Relais	34
2.4. Afficheur LCD	34
2.5. Capteur de courant ACS712	34
2.6. La liaison RS232	35
3. Partie informatique	35
3.1. Microsoft Visual C#.....	35
3.2. Langage MySQL	36
3.3. Logiciel de programmation Arduino.....	36
3.4. Logiciel de simulation ISIS	37
3.5. WinRelais.....	37
Conclusion	38
Chapitre 4 : Réalisation et simulation	39

Introduction.....	40
I. Simulation du système d'alarme	40
1. Fonctionnement du système.....	40
2. Déroulement de la simulation.....	41
II. Réalisation matérielle.....	47
Conclusion	47
Conclusion générale	48
Webographie.....	49
Annexe : Code application sur Arduino UNO :.....	50
Annexe : Code application de vérification sur Microsoft Visual C# :	52
Annexe : Code application identification sur Microsoft Visual C# :.....	53

Liste des figures

Figure 1:M.Yasuhiko YAZAKI Président Conseil YAZAKI Corporation	11
Figure 2 : Les principaux clients de YAZAKI	11
Figure 3 : Société YAZAKI Kenitra	12
Figure 4 : Clients d'YMK.....	12
Figure 5: Organigramme de YAZAKI Kenitra.....	13
Figure 6:Câble principale.....	14
Figure 7: Les parties du câblage dans l'automobile	14
Figure 8: Composants d'un faisceau.....	15
Figure 9:Processus de production	15
Figure 10:Machine de coupe KOMAX	16
Figure 11:Equipements d'inspection et de test	17
Figure 12: Schéma descriptif de la ligne Main body KSK.....	21
Figure 13: Diagramme des temps de cycle de tous les processus	24
Figure 14 : Diagramme de Gantt	25
Figure 15:Carte Arduino	27
Figure 16: Interrupteur différentiel.....	31
Figure 17: Disjoncteur.	31
Figure 18:Power supply	31
Figure 19:Alimentation 220V alternatif /24V continu.....	32
Figure 20 : Convertisseur alternatif / alternatif	32
Figure 21 : Convertisseur alternatif / continu	32
Figure 22:Fusible	32
Figure 23:Gyrophare 24V/1 .8A.....	33
Figure 24:Carte Arduino	33
Figure 25 : Capteur DHT11	33
Figure 26 : Carte relais.....	34
Figure 27 : Afficheur LCD	34
Figure 28 : Capteur ACS712	34
Figure 29 : Câble RS232	35
Figure 30:Interface Microsoft Visual C#	36
Figure 31 : Logiciel de l'Arduino	36
Figure 32:Interface ISIS.....	37
Figure 33 : Interface WinRelais.....	38
Figure 34:Organigramme du système	40
Figure 35 : Schéma du montage électronique	41
Figure 36:Visualisation d'état des capteurs	42
Figure 37 : Schéma de commande du gyrophare.....	42
Figure 38: Liaison Arduino relais	42
Figure 39: Schéma d'alimentation (puissance)	43
Figure 40 : Algorithme de programme sur C#.....	44
Figure 41:Interface de vérification dans C#.....	45
Figure 42: Interface d'identification dans C #	45
Figure 43:Les codes compatibles.....	46
Figure 44:Les codes ne sont pas compatibles	46

Figure 45: Mise en place du premier test du système d'alarme..... 47

Liste des tableaux

Tableau 1: Fiche signalétique de YAZAKI KENITRA..... 12
Tableau 2: Les postes de la zone P2 16
Tableau 3 : Défaut KSK main body 22
Tableau 4: Description de la problématique étudiée via l’outil QQQQCP 23
Tableau 5: Caractéristiques de la carte Arduino 28

Liste des abréviations

YMK : YAZAKI Maroc Kenitra.

QQQQCP : Qui, Quoi, Où, Quand, Comment, Combien, Pourquoi.

P1 : Zone de coupe.

P2 : Zone de pré-assemblage.

P3 : Zone d’assemblage.

KSK : consommateur spécifique câble.

Introduction générale

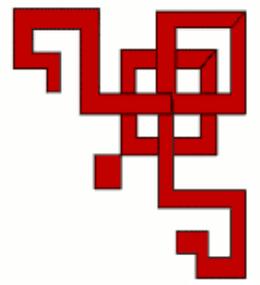
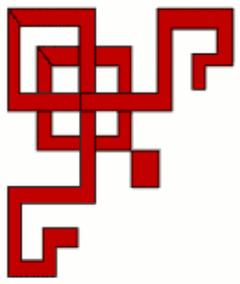
L'industrie automobile a connu dans les dernières décennies une mutation aussi bien importante, et cela d'une part est dû au développement technologique qui a contribué d'une manière cruciale à l'expansion de ce marché, qui est de plus en plus croissant et d'une autre part du mode de vie des clients qui sont devenus plus exigeants et sélectifs dans leurs choix surtout en matière de qualité, ce qui pousse les industriels à être plus compétitif et ouvrant pour non seulement répondre aux exigences de leurs clients, mais aussi de rester dans les normes imposées par le marché mondial.

Allant de ce point de vue, la société YAZAKI, leader dans la production des câbles automobiles, au niveau international ouvre depuis son installation au Maroc à fournir à sa clientèle des produits répondant à leur besoin et cela dans le cadre de sa vision de développement durable, de gestion des ressources et de diminution des gaspillages.

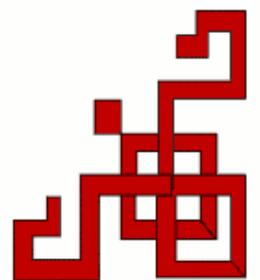
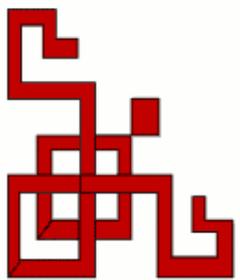
L'idée majeure de ce travail et en collaboration avec le département informatique et technologique est d'optimiser la production des câbles automobiles en mettant en place un système automatique permettant l'anticipation et la détection des erreurs au cours de la production et d'éviter les pertes qui lui sont liées.

Le présent rapport est structuré comme présenté ci-dessous :

Le premier chapitre comprendra une présentation de l'organisme d'accueil, il donnera une présentation de YAZAKI en tant qu'organisme d'accueil, son domaine d'activité et son processus de production. Le deuxième chapitre comportera une présentation du projet : la problématique de la société, l'analyse et la solution proposée. Quant au troisième chapitre, on parle sur la conception matérielle de notre système. Le quatrième chapitre est consacré au programme réalisé et la simulation avec la carte Arduino sur ISIS. Une conclusion mettra fin à ce rapport.



Chapitre 1 : Présentation de l'entreprise



Introduction

Dans ce premier chapitre, nous allons donner un aperçu général sur YAZAKI monde, puis YAZAKI Kenitra tout en détaillant son processus de fabrication.

I. YAZAKI monde



Figure 1: M. Yasuhiko YAZAKI Président Conseil YAZAKI Corporation

Créé en 1929 par le père SADAMI YAZAKI, le groupe YAZAKI a fait ses débuts dans la vente du câblage automobile. En octobre 1941 YAZAKI est devenue l'un des composants pour automobile avec un capital de 3.1915 milliards Yen, actuellement YAZAKI est représentée dans 38 pays, elle compte à son actif plus que 153 sociétés et 410 unités réparties entre usines de production, centres de service au client, centres techniques et technologiques, et fait employer plus de 180000 employés dans le monde. Son activité principale est le câblage, la fabrication de composants électriques pour l'automobile et différentes machines électriques. Ses autres activités sont :

- La fabrication de fils et câbles électriques
- La fabrication de produits pour gaz
- La climatisation

Sur le marché du câblage, YAZAKI figure parmi les leaders au niveau mondial. Grâce au niveau de qualité/ prix qu'elle offre, elle compte, parmi ses clients, des sociétés de réputation, telles que : MERCEDES, JAGUAR, PEUGEOT, NISSAN MOTOR, FIAT, TOYOTA...



Figure 2 : Les principaux clients de YAZAKI

II. YAZAKI Kenitra



Figure 3 : Société YAZAKI Kenitra

Le groupe YAZAKI a installé le deuxième site au Maroc de câblage automobile à Kenitra, une région qui ambitionne de devenir un pôle industriel spécialisé particulièrement dans la fabrication d'équipements d'automobile. Son activité principale est le câblage pour Automobile et la totalité de sa production de câbles électriques est destinée pour l'équipement des marques Jaguar et Land Rover.

Principaux clients de YAZAKI Kenitra jaguar et land rover :



Figure 4 : Clients d'YMK

I. Fiche technique YAZAKI Kenitra

La fiche signalétique de YAZAKI Maroc Kenitra (YMK) est présentée dans ce tableau. Raison sociale	YAZAKI MOROCCO Kenitra
Activité	Câblage automobile
Forme juridique	Société anonyme
Date de création	Juillet 2010
Capital	89.327.000,00Dhs
Effectif	2732
Certification	ISO 14001 version 2004 ISO 18000-18001 version 2007
Contact	Adresse : Route Km 9 Quartier Al Assam Kenitra Tél : 0537369600

Tableau 1: Fiche signalétique de YAZAKI KENITRA

2. Organigramme YAZAKI Kenitra

La structure de l'organigramme est une structure fonctionnelle qui lie l'ensemble des différentes activités, permettant une circulation de l'information qui assure une certaine coordination tout en minimisant les défauts et les dysfonctionnements internes.

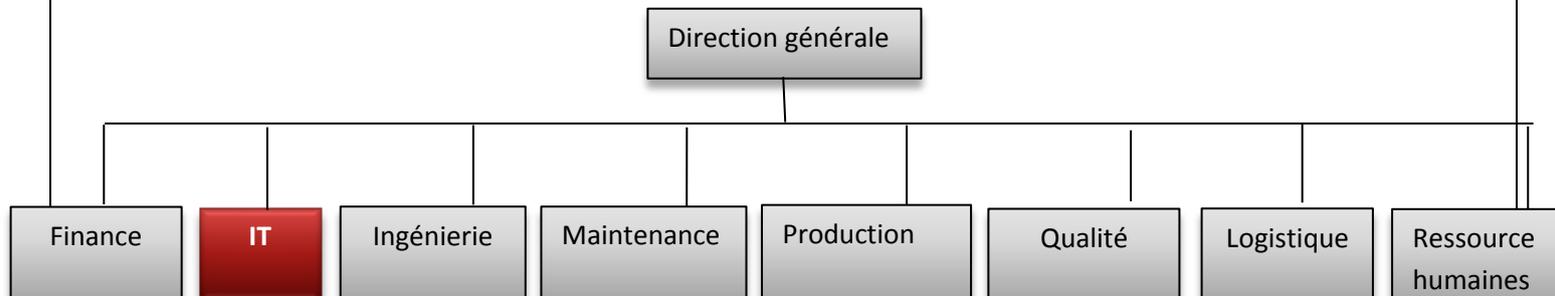


Figure 5: Organigramme de YAZAKI Kenitra

3. Activités principales des services

Le département des ressources humaines : disposer à temps des effectifs suffisants et en permanence, assurer une gestion performante individuelle et collective du personnel par la formation.

Le département financier : assurer les fonctions financières et comptables de l'entreprise, développer et implanter les pratiques, les procédures financières et le contrôle de gestion.

Le département logistique: son rôle est d'optimiser la mise en place et le lancement des programmes de fabrication tout en assurant une gestion optimale du stock et une expédition à temps aux clients.

Le département qualité : c'est le garant de la politique et du système qualité de l'entreprise à travers l'implantation d'un système qualité fiable.

Le département engineering : qui a pour mission d'adapter les procédés de fabrication conformément aux règles définies par les Directions Engineering et Qualité (plans de surveillance, control plan, ...) du groupe.

Le département production : qui a pour principale mission la réalisation des programmes de production tout en assurant une bonne qualité du produit en respectant les délais fixés au préalable et en optimisant les performances.

Le département maintenance : il assure l'installation et la maintenance de tous les équipements de l'usine avec une fiabilité optimale et une efficacité maximale d'équipement d'YMK.

Le département IT: Ce département est chargé d'animer et d'assurer la cohérence des divers systèmes traitant l'information et les mettre à la disposition des utilisateurs, il est chargé également de la gestion des réseaux, des postes et des logiciels de bureautique

4. Câblage automobile

4.1. Définition

Qu'est-ce qu'un câblage ?

Le câblage est un ensemble de fils qui relie l'ensemble des composants qui portent les fonctions électriques et électroniques du véhicule. Il existe plusieurs familles de câblages connectées, les plus importantes sont :

- **Le câblage principal** : il est monté sur caisse et assure les liaisons entre le compartiment moteur et l'habitacle.
- **Le câblage habitacle** : il est monté sur caisse et assure les liaisons des différents équipements de l'habitacle et boîtier de servitude habitacle.
- **Le câblage moteur** : il assure les liaisons entre les sondes, électrovannes, injecteurs, capteurs et le calculateur moteur.



Figure 6: Câble principale

La figure ci-dessous regroupe les différents types de câbles fabriqués :

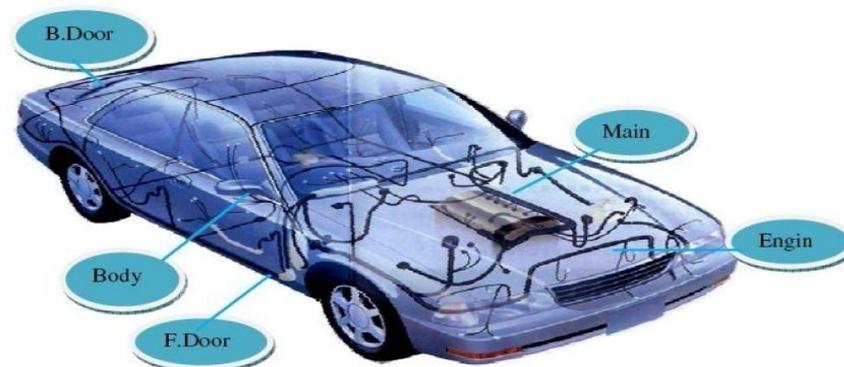


Figure 7: Les parties du câblage dans l'automobile

4.2. Composants d'un câble

Un câblage est constitué d'un ensemble de conducteurs électriques, terminaux, connecteurs et matériels de protection.

Fil conducteur : Conduit le courant électrique d'un point à un autre.

Terminal : Assure une bonne connexion entre deux câbles (l'un est une source d'énergie, l'autre est un consommateur d'énergie).

Connecteur : Ce sont des pièces où les terminaux seront insérés, ils permettent d'établir un circuit électrique, d'établir un accouplement mécanique séparable et d'isoler électriquement les parties conductrices.

Accessoires : Ce sont des composants pour la protection et isolation du câblage : les rubans d'isolement, les tubes.

Matériel de protection (Fusible) : Sont des pièces qui protègent le câble et tous ses éléments de la surcharge du courant qui pourrait l'endommager.

Clips ou agrafes : Les clips sont des éléments qui permettent de fixer le câble à la carrosserie de l'automobile. Sans les clips le montage serait impossible, le câble restera détaché provoquant des bruits et sera exposé aux détériorations à cause des frottements.

La figure si dessous regroupe les différents composants du câble:

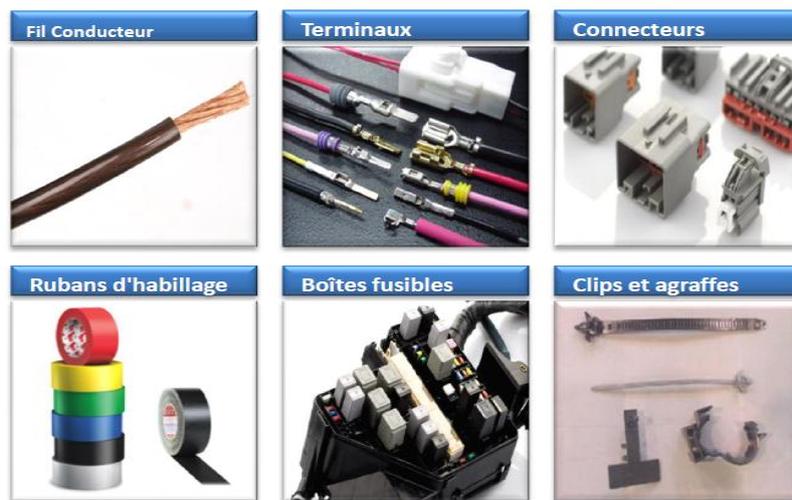


Figure 8: Composants d'un faisceau

5. Processus de fabrication des câblages automobile

Pour arriver à réaliser un produit fini, le processus de production d'un produit s'effectue sur trois étapes fondamentales:

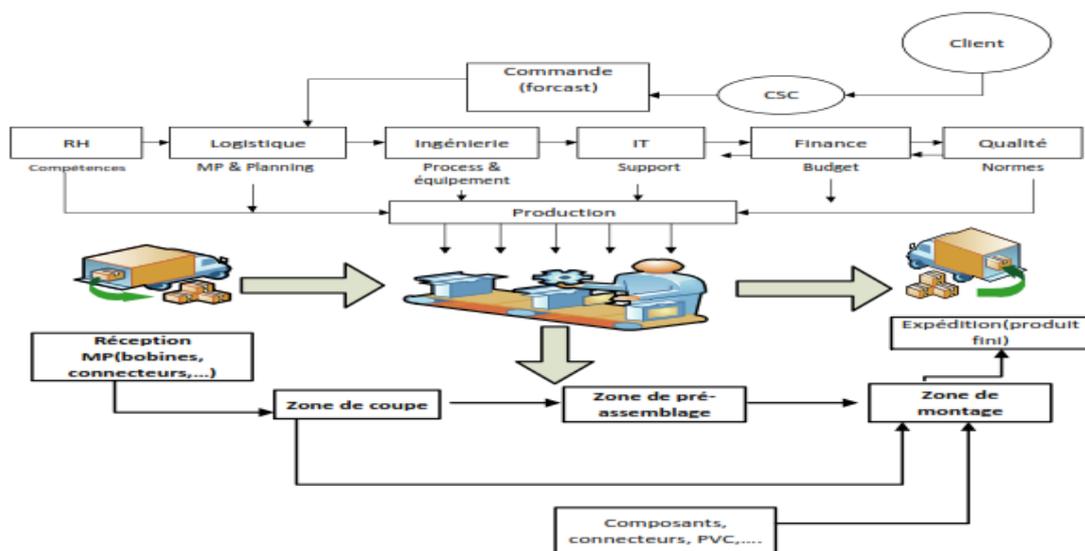


Figure 9: Processus de production

5.1. La coupe

La coupe (s'appelle P1) est en réalité la première étape de fabrication physique du câble, elle se fait après stockage et préparation et de la matière première et bien sûr après réception des DATA (liste des fils à couper) du département ingénierie. Pour ce faire la coupe est équipée par deux types de machines automatiques qui servent à la coupe des fils, qui sont utilisées comme illustrées dans la figure suivante: KOMAX et YACC.

- KOMAX: machine standard utilisée par toutes les entreprises de câblage.
- YACC : c'est une machine fabriquée spécialement pour YAZAKI.



Figure 10: Machine de coupe KOMAX

5.2. Pré-assemblage

Est une étape intermédiaire, mais pas toujours nécessaire, entre la coupe et le montage (assemblage) puisque certains circuits finissent au niveau de la coupe et passent directement vers le secteur montage pour être utilisés, d'autres circuits selon leur nature passent par l'une ou toutes les étapes décrites dans le tableau suivant :

	Description	Machine utilisée
Poste sertissage	L'insertion des terminaux se fait automatiquement sur les machines de la coupe, cependant pour des raisons de capacité et de puissance de la machine, ils ont assuré des petites machines consacrées spécialement à cette opération dans la zone de pré assemblage. La machine Mecal permet de faire le sertissage manuel.	
Poste joint	la jointure des fils se fait par trois techniques différentes : <ul style="list-style-type: none"> ✓ Par vibration : elle s'exécute avec la machine Ultra Sonic ✓ Par chaleur : se fait avec la machine Raychem qui sert à joindre les fils et insérer le bouchon (Shrink) à la place de la jonction pour lui donner plus de résistance. Par soudage: dans le but de protéger la partie visible entre le terminal et le fil.	<ul style="list-style-type: none"> ✓ Schunk ✓ Raychem
Poste twist	Sert à twister 2 fils (rarement 3) par des machines différentes, dont chacune a des caractéristiques propres.	<ul style="list-style-type: none"> ✓ Machine TWIST YAZAKI, ✓ Machine TWIST SLOVAKIA. ✓ Machine KOMAX.
Poste d'accessoire	Pour l'insertion des accessoires (par exemple le bouchon).	

Tableau 2: Les postes de la zone P2

5.3. L'assemblage

C'est la zone où les fils venant de la coupe sont assemblés. L'assemblage se fait soit sur des tableaux fixes pour les câbles de petites dimensions soit sur des tableaux

roulants dans les chaînes de montage pour les câbles longs. Les câbles passent généralement par trois étapes principales lors du montage : l'insertion, l'enrubannage et l'inspection. Chacune de ces étapes comporte des opérations qui varient en fonction de la nature du câble.

5.3.1. L'insertion

Cette étape consiste à insérer manuellement les terminaux des circuits dans les connecteurs qui leur correspondent. Des supports sous forme de fiches comprenant le processus d'assemblage sont mis à la disposition des opérateurs.

5.3.2. Enrubannage

C'est l'opération qui permet de recouvrir les fils une fois insérée par des rubans et des protecteurs. Ils consistent à couvrir les faisceaux électriques soit avec des rubans adhésifs ou avec des tuyaux, afin de les protéger de la haute température, des éraflures et pour assurer des cotes adéquates aux spécifications client.

5.3.3. Le contrôle électrique

Le câble subit un contrôle électrique qui représente la dernière étape du processus d'assemblage et qui a pour but l'assurance de la qualité et du bon fonctionnement des faisceaux électriques en vérifiant la continuité électrique entre les différentes extrémités du circuit et la présence des éléments secondaires (sécurité des connecteurs, les fusibles ...). Les tests standards réalisés sur les câbles sont illustrés dans la figure suivante :

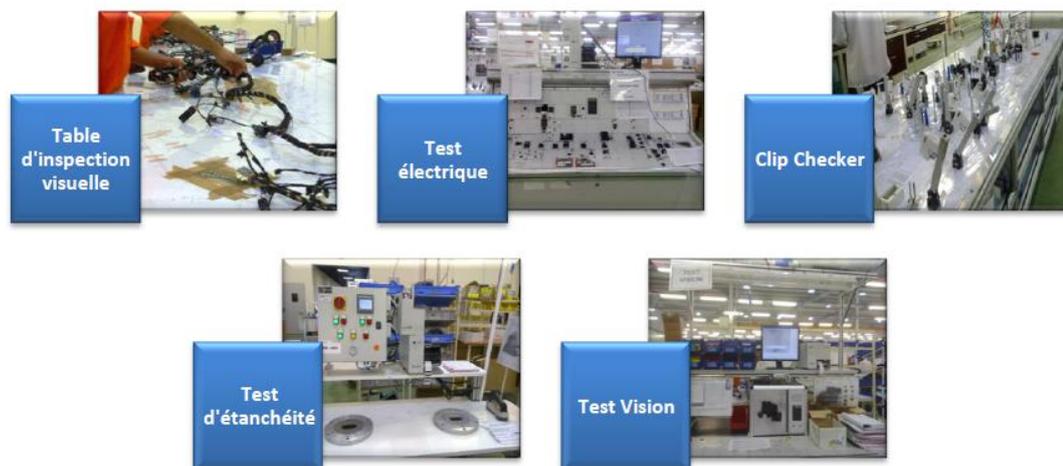
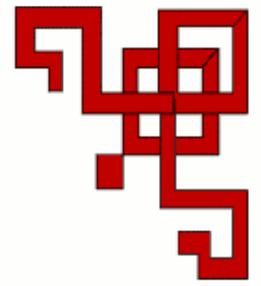
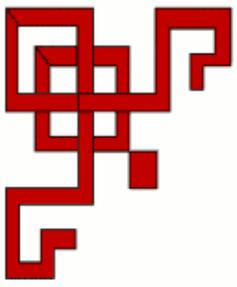


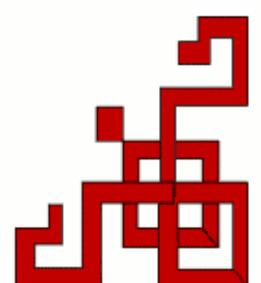
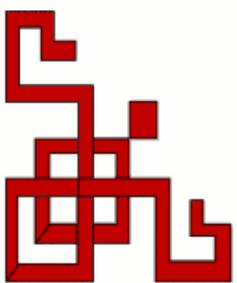
Figure 11: Equipements d'inspection et de test

Conclusion

Ce chapitre introductif avait pour but de présenter l'environnement général dans lequel le projet se déroule. Nous avons présenté la société YAZAKI Kenitra, ses activités et sa politique.



Chapitre 2 : Présentation de projet



Introduction

Après avoir introduit l'organisme d'accueil, nous passons dans ce deuxième chapitre à illustrer les outils utilisés pour la réalisation d'un système d'alarme. Cette étape consiste à étudier la problématique de la société YAZAKI Kenitra et définir la solution proposée.

I. Cahier de charge

1. Acteurs du projet

1.1. Maître d'ouvrage

Le maître d'ouvrage est la société YAZAKI qui est une société de câblage d'automobile appartenant au groupe YAZAKI installée à Kenitra.

1.2. Maître d'œuvre

Faculté des sciences et techniques Fès, filière Génie électrique, présentée par Fatimetou ELHADJ et Nadia KALAYDI. Avec le suivi et l'encadrement de :

- M. ELKHAYATI: Encadrant de stage
- M. ABARKAN : Encadrant pédagogique

2. Contexte pédagogique

Ce projet s'inscrit dans le cadre du stage de projet de fin d'études, dont les étudiants de FST appartenant à cette filière sont censés faire un Projet dans le domaine électrique.

Au cours de ce stage, nous devons mettre en place nos acquis pour résoudre des problèmes et trouver des solutions pratiques.

3. Sujet de projet

L'objet de ce projet est de faire l'étude la conception et la mise en œuvre d'un système d'alarme commandé par un Arduino afin de contrôler et vérifier les références des câbles lors du montage de l'infotainment sur le convoyeur assemblage.

4. Description

Le présent sujet est scindé en trois étapes :

- L'étude de la carte Arduino.
- La conception du système d'alarme.
- La réalisation et la programmation du système.

5. Contraints à respecter

- Les solutions proposées doivent être rentables et efficaces.
- L'investissement demandé pour mettre en place la solution proposée doit être raisonnable et bien justifié.
- Le projet doit être terminé dans l'intervalle de temps défini entre le 04 avril 2016 et le 04 juin 2016.

II. La problématique

Cette phase va nous permettre de définir la problématique, les objectifs du projet, le processus du projet et les ressources nécessaires pour commencer notre étude.

1. Description de la chaîne de production des câbles

La chaîne de production des câbles passe par trois étapes séparées :

- SPS LINE : dans laquelle une partie du câble est fabriquée et qui sera codée plus tard scannée en fonction de ce code avant d'être montée sur le convoyeur assemblage.
- Convoyeur assemblage: C'est l'étape principale de cet assemblage, au cours de laquelle la grande partie du câble sera fabriquée.
 - Celle-ci comporte deux parties, qui sont: les postes d'insertions et les postes d'enrubannage.
- L'infotainment: dans cette étape aussi, la fabrication d'une partie du câble aura lieu afin d'être montée au convoyeur après que les postes d'insertions seront terminés.

⇒ Les câbles se distinguent par un code-barres qui est variant de l'un à l'autre.

2. Les Étapes de la production du câble

La production du câble commence au niveau du SPS Line, qui contient cinq postes.

Chacune de ces dernières a des ordres qui lui sont propres, après l'exécution de chaque ordre, la poste fait passer son travail à celle qui la succède.

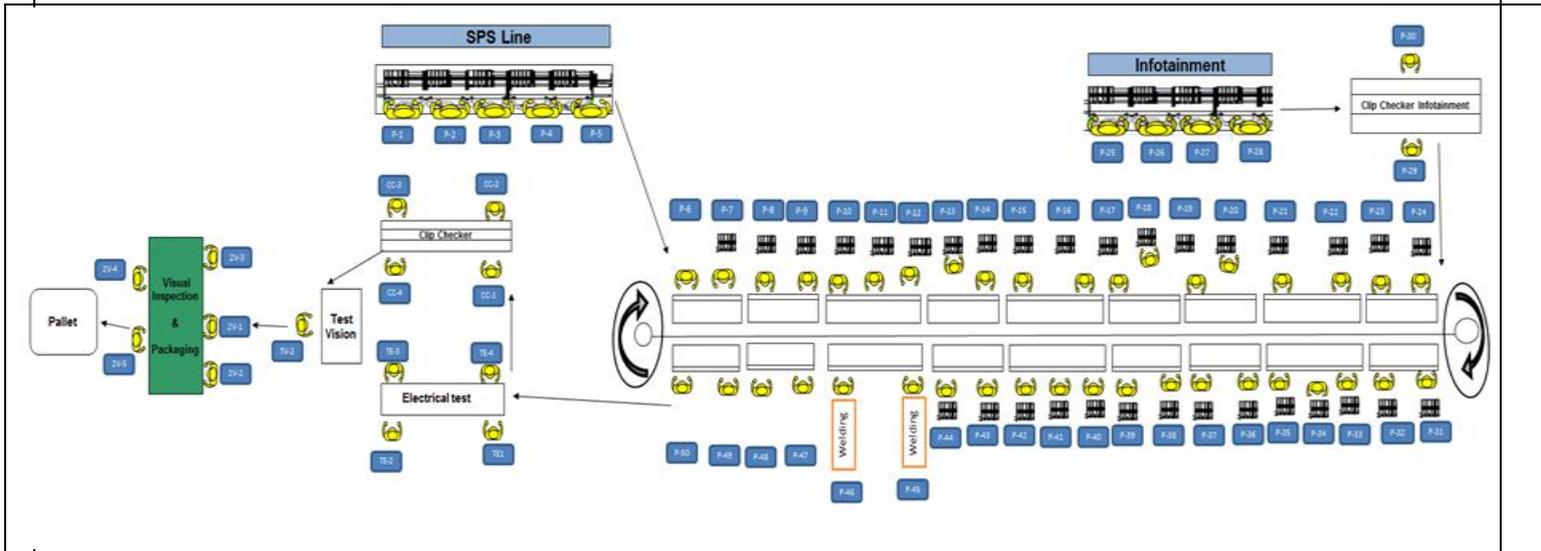
Après l'exécution de tous les ordres donnés aux postes, la partie du câble qui a été tout de suite fabriqué sera codée par un code, qui uniquement appartient à ce câble, afin de pouvoir le distinguer des autres.

Ensuite le câble sera scanné et monté sur le convoyeur assemblage dans le premier poste de l'assemblage qui est aussi la sixième poste. Parallèlement, la partie infotainment sera établie avec l'assemblage en même temps.

Avant d'entamer l'enrubannage l'infotainment sera combiné avec la partie du câble venant de l'insertion selon leurs compatibilités et suivant la similarité de leurs codes,

À la fin de l'enrubannage, le câble mère sera mené par un opérateur pour subir un test électrique au niveau duquel on vérifie sa conductibilité.

La figure suivante permettra de voir la ligne d'assemblage en gros :



: Opérateur de la zone de production.

: Poste

: Palette

: Les fils

: Machine de shunk

Figure 12: Schéma descriptif de la ligne Main body KSK

3. Vision sur la problématique

Durant le test, il s'avère parfois que des câbles non compatibles se sont combinés au niveau de l'étape infotainment, par erreur de la part de l'opérateur avec un mal lecture visuellement.

Le problème est que le câble erroné n'est détecté qu'à la fin de la chaîne d'assemblage au niveau de la dernière étape qui est le test électrique ce qui rend l'opération de rework (recyclage) du câble plus au moins compliqué.

Pour avoir une idée sur les câbles et les postes où il y'a souvent un problème de confusion, le tableau suivant donne une idée sur les câbles erronés et les postes responsables sur cette confusion :

Code défaut	Opérateur	chaîne	Client	modèle	famille	Processeur	Quantité	shift	Date
AW26	P24	Main Body KSK 01	Land Rover	L538	Main body	Test électrique	1	S	4/23/2016 12:00:00 AM
AW26	P24	Main Body KSK 01	Land Rover	L538	Main body	Test électrique	1	S	4/23/2016 12:00:00 AM
AW26	P24	Main Body KSK 01	Land Rover	L538	Main body	Test électrique	1	S	4/19/2016 12:00:00 AM
AW21	P18	Main Body KSK 01	Land Rover	L538	Main body	Test électrique	1	M	4/17/2016 12:00:00 AM
AA50	P6	Main Body KSK 01	Land Rover	L538	Main body	Test électrique	1	M	4/15/2016 12:00:00 AM
AW26	P24	Main Body KSK 01	Land Rover	L538	Main body	Test électrique	1	M	4/15/2016 12:00:00 AM
AW22	P9	Main Body KSK 01	Land Rover	L538	Main body	Test électrique	1	M	4/13/2016 12:00:00 AM
AW26	P24	Main Body KSK 01	Land Rover	L538	Main body	Test électrique	1	M	4/13/2016 12:00:00 AM
AW26	P24	Main Body KSK 01	Land Rover	L538	Main body	Test électrique	1	S	04/09/2016 00:00

Tableau 3 : Défaut KSK main body

Remarque :

Après avoir analysé ces différents problèmes détectés, on trouve que notre problème étudié de confusion arrive toujours au niveau du poste d'assemblage 24 pour la référence de câble AW26.

III. Cadrage du problème

1. Définir la problématique

Dans le but de décrire d'une manière structurée notre problématique, nous avons utilisé les principales questions - réponses de l'outil QQQCP décrit dans le tableau:

Qui ?	Les départements d'YMK: ingénierie, production, qualité et logistique.
Quoi ?	Risque de montage des sous-éléments infotaiment erronés.
Où ?	Au niveau du test électrique.
Quand ?	Lors de la combinaison de l'infotaiment avec les câbles à assembler.
Comment ?	Par une vérification des codes.
Pourquoi ?	Améliorer la productivité et réduire le gaspillage du temps.

Tableau 4: Description de la problématique étudiée via l'outil QQQQCP

Des réponses plus détaillées aux questions QQQQCP sont formulées dans ce qui suit :

➤ **Qui ? Qui est concerné par problème**

Les départements d'YMK: ingénierie, production, qualité et logistique.

➤ **Quoi ? C'est quoi le problème ?**

Risque de montage des sous éléments infotaiment erronées sur Convoyeur assemblage.

➤ **Où ? Où apparaît le problème ?**

Arrivant au test électrique le problème apparait.

➤ **Quand ? Quand est-ce qu'apparait le problème ?**

Lorsque l'opérateur vérifie manuellement le Numéro de la partie du câble venant de l'infotaiment avec le câble à la fin de l'insertion tout en assimilant le code qui accompagne chaque partie.

➤ **Comment ? Comment mesurer le problème et ses solutions ?**

Pour mesurer le problème on a fixé un poste de vérification automatique pour ne pas laisser l'opérateur commis l'erreur.

➤ **Pourquoi ? Pourquoi il faut résoudre ce problème ?**

On souhaite résoudre ce problème pour minimiser les pertes de la société dues au recyclage de ces câbles.

2. Analyse de problème

Cette partie d'analyse consiste à montrer toutes les pertes et les dégâts dus à ce problème d'incompatibilité.

Les pertes engendrées par le problème de compatibilité sont multiples, à noter que chaque câble infotaiment monté d'une manière erronée provoque l'arrêt immédiat de toute la chaine de production, car au moins le câble qui le suit sera évidemment erroné aussi.

Pour une bonne analyse de dégâts et pertes engendrés par cette problématique on a utilisé la méthode de chronométrage pour définir le temps de production pour chaque processus. C'est l'action de chronométrer les durées des opérations effectuées par un opérateur sur chaque poste afin de définir son temps de cycle.

⇒ **Le temps de cycle (cycle time) d'un câble** : est le temps mesuré entre deux sorties successives de produits finis.

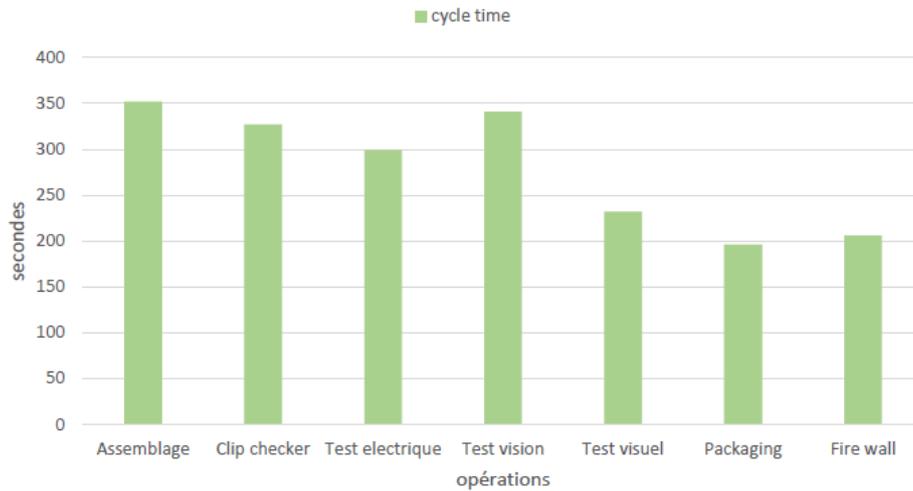


Figure 13: Diagramme des temps de cycle de tous les processus

D'après le diagramme ci-devant, on déduit que l'erreur de compatibilité des codes commise lors de l'infotaiment et constatée au cours du test électrique retarde la chaîne de production par au moins 980 secondes qui est la somme du cycle time du poste responsable de l'infotaiment, de clip checker qui le suit et le test électrique ou le problème de confusion est toujours détecté, ce qui est beaucoup de temps perdu et par suite coutera à l'entreprise un beau montant d'argent.

Ensuite, mis à part le temps perdu lors de détection de ce problème on constate d'autres conséquences sur la chaîne de production dont les plus importants sont comme suit :

- Retour du câble pour rework.
- Retard de la production plus.
- Augmentation du cout du câble à cause de son rework.
- Risque d'arrêt de la chaîne de production.

3. Propositions

Après analyse on constate que les causes principales du problème sont :

- Dépendance sur la Vérification visuelle de l'opérateur.
- Détection des erreurs qu'après passage au test électrique.
- Manque d'un système automatique susceptible de faire la compatibilité entre les câbles.

Pour toutes ces raisons, on propose une solution d'automatisation de l'opération de vérification au niveau du poste 24.

4. Solution

Pour remédier à ce problème ,notre cahier de charge consiste en l'élaboration d'un système d'alarme ,qui sera placé à la fin de l'étape d'insertion et qui sera capable de comparer les codes scannés sur l'ensemble des câbles venant de l'étape d'insertion et les câbles venant de l'étape infotainment.

En cas de similarité des codes, rien ne se passe, et la chaîne de production continue, et en cas de non-similarité, le système d'alarme se déclenche, informant l'opérateur de l'existence d'une erreur.

IV. Planning du Travail

Pour bien organiser notre travail et pour situer nos objectifs dans notre cadre temporelle limitée qui est de deux mois, le diagramme de GANTT semble idéal pour planifier notre travail et rendre plus simple le suivi de son avancement.

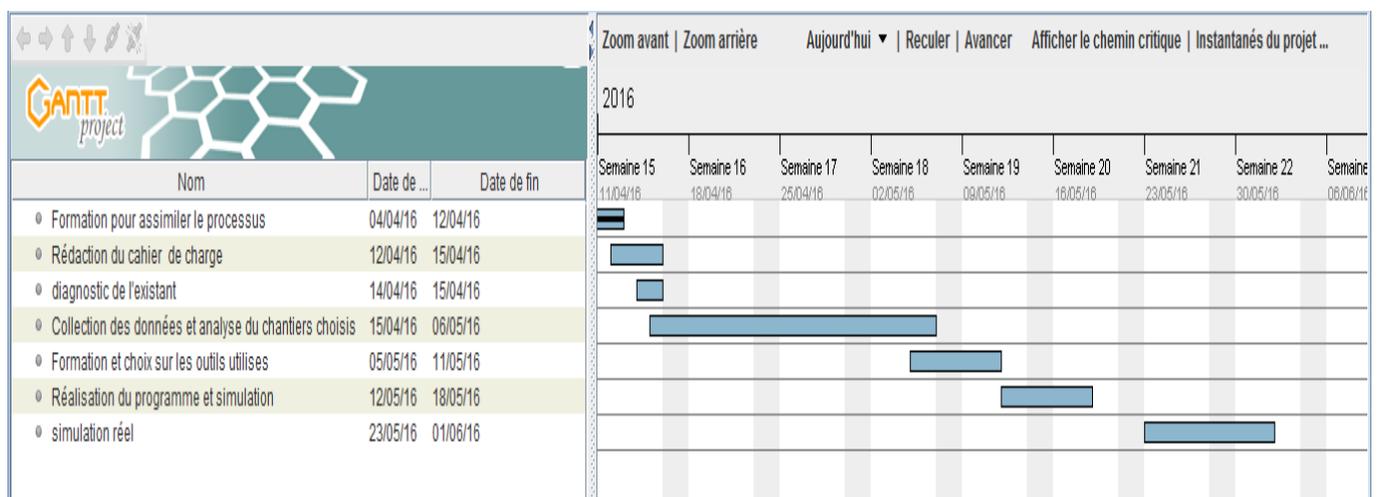
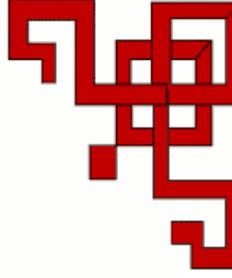
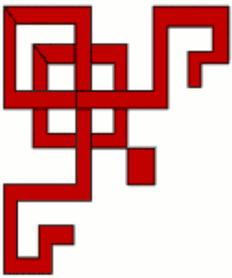


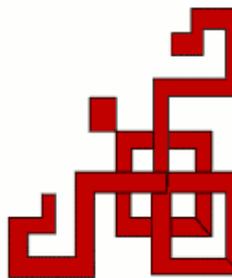
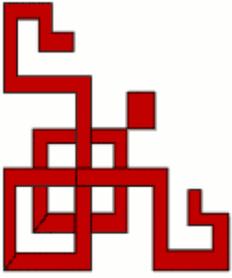
Figure 14 : Diagramme de Gantt

Conclusion

Après avoir analysé notre problématique, précisé les causes et les conséquences du problème étudié et défini notre cahier de charge, l'objectif et l'étude préalable du chapitre suivant sont de mettre l'application dans son cadre conceptuel.



Chapitre 3 : Conception d'un système d'alarme



Introduction

Dans ce chapitre nous présenterons une description détaillée de la solution adoptée pour répondre aux spécifications de notre cahier de charge en abordant la conception détaillée de chaque partie du système afin d'obtenir une schématisation complète et précise.

I. Description des outils utilisés

Matériel :

- ✓ Carte Arduino Uno.
- ✓ Capteur humidité et température DHT11.
- ✓ Capteur de courant ACS712 ;
- ✓ LCD, relais, interrupteur différentiel, disjoncteur, alimentation stabilisée, fusible, gyrophare.

Logiciel :

- ✓ Compilateur Arduino.
- ✓ Logiciel de simulation ISIS.
- ✓ Winrelais.
- ✓ Microsoft visual C#.
- ✓ MySQL.

II. Étude sur l'Arduino

I. Qu'est-ce qu'Arduino?

Le système Arduino est un outil pour fabriquer de petits ordinateurs qui peuvent contrôler d'avantage les choses du monde matériel que votre ordinateur de bureau. C'est une plateforme open source d'électronique programmée qui est basée sur une carte à microcontrôleur et un logiciel, véritable environnement de développement intégré, pour écrire, compiler et transférer le programme vers la carte.

Parmi toutes les cartes de la famille Arduino, nous avons choisi la version UNO vu ces caractéristiques et c'est à partir de cette base que nous avons développé toute la suite de ce projet.

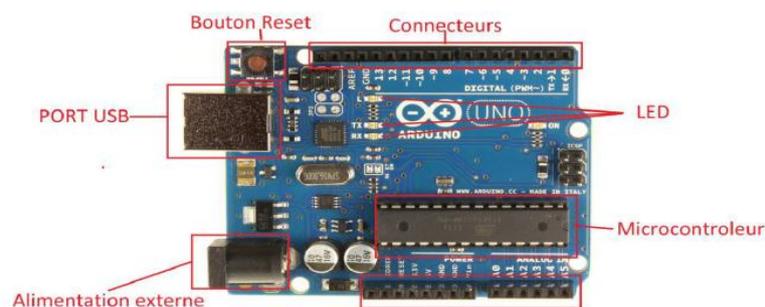


Figure 15: Carte Arduino

La carte Arduino Uno comme illustrée dans la figure ci-dessus est une carte à microcontrôleur basée sur l'ATmega328.

Elle dispose :

- de 14 broches numériques d'entrées/sorties (dont 6 peuvent être utilisées en sorties PWM (largeur d'impulsion modulée)),
- de 6 entrées analogiques (qui peuvent également être utilisées en broches entrées/sorties numériques),
- d'un quartz 16Mhz,
- d'une connexion USB,
- d'un connecteur d'alimentation jack,
- d'un connecteur ICSP (programmation "in-circuit"),
- et d'un bouton de réinitialisation (reset).

Elle contient tout ce qui est nécessaire pour le fonctionnement du microcontrôleur; pour pouvoir l'utiliser et se lancer, il suffit simplement de la connecter à un ordinateur à l'aide d'un câble.

1.1. Synthèse des caractéristiques

Le tableau ci-dessous représente les caractéristiques de la carte Arduino utilisée :

Microcontrôleur	ATmega328
Tension de fonctionnement	5V
Tension d'alimentation (recommandée)	7-12V
Tension d'alimentation (limites)	6-20V
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM)
Broches d'entrées analogiques	6 (utilisables en broches E/S numériques)
Intensité maxi disponible par broche E/S (5V)	40 mA (ATTENTION : 200mA cumulés pour l'ensemble des broches E/S)
Intensité maxi disponible pour la sortie 3.3V	50 mA
Intensité maxi disponible pour la sortie 5V	Fonction de l'alimentation utilisée - 500 mA max si port USB utilisé seul
Mémoire Programme Flash	32 KB (ATmega328)
Mémoire SRAM (mémoire volatile)	2 KB (ATmega328)
Mémoire EEPROM (mémoire non volatile)	1 KB (ATmega328)
Vitesse d'horloge	16 MHz

Tableau 5: Caractéristiques de la carte Arduino

1.1.1. Alimentation

La carte Arduino Uno peut-être alimentée soit via la connexion USB (qui fournit 5V sous 500mA) ou à l'aide d'une alimentation externe. La source d'alimentation est sélectionnée automatiquement par la carte.

L'alimentation externe (non-USB) peut être soit un adaptateur secteur (pouvant fournir typiquement de 3V à 12V sous 500mA) ou des piles (ou des accus). L'adaptateur secteur peut être connecté en branchant une prise 2.1mm positive au centre dans le connecteur jack de la carte. Les fils en provenance d'un bloc de piles ou d'accus peuvent être insérés dans les connecteurs des broches de la carte appelées Gnd (masse ou 0V) et Vin (Tension positive en entrée) du connecteur d'alimentation.

La carte peut fonctionner avec une alimentation externe de 6 à 20 volts. Cependant, si la carte est alimentée avec moins de 7V, la broche 5V pourrait fournir moins de 5V et la carte pourrait être instable. Si on utilise plus de 12V, le régulateur de tension de la carte pourrait chauffer et endommager la carte. Aussi, la plage idéale recommandée pour alimenter la carte Uno est entre 7V et 12V.

Les broches d'alimentation sont les suivantes :

- **VIN.** La tension d'entrée positive lorsque la carte Arduino est utilisée avec une source de tension externe (à distinguer des 5V de la connexion USB ou autre source 5V régulés). Vous pouvez alimenter la carte à l'aide de cette broche, ou, si l'alimentation est fournie par le jack d'alimentation, accéder à la tension d'alimentation sur cette broche.
- **5V.** La tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (pour info : les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite "tension régulée" obtenue à l'aide d'un composant appelé un régulateur et qui est intégré à la carte Arduino). Les 5V régulés fournis par cette broche peuvent donc provenir soit de la tension d'alimentation VIN via le régulateur de la carte, ou bien de la connexion USB (qui fournit des 5V régulés) ou de toute autre source d'alimentation régulée.
- **3.3V.** Une alimentation de 3.3V fournie par le circuit intégré FTDI (circuit intégré faisant l'adaptation du signal entre le port USB de votre ordinateur et le port série de l'ATmega) de la carte est disponible : ceci est intéressant pour certains circuits externes nécessitant cette tension au lieu des 5V). L'intensité maximale disponible sur cette broche est de 50mA.
- **GND.** Broche de masse.

1.1.2. Entrées et sorties numériques

Chacune des 14 broches numériques de la carte UNO (numérotées des 0 à 13) peut être utilisée soit comme une entrée numérique, soit comme une sortie numérique.

De plus, certaines broches ont des fonctions spécialisées :

- **Communication Série:** Broches 0 (RX) et 1 (TX). Utilisées pour recevoir (RX) et transmettre (TX) les données sériées de niveaux TTL. Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega8U2

programmé en convertisseur USB-vers-série de la carte, composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur.

- **Interruptions externes:** Broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement de valeur. Voir l'instruction `attachInterrupt()` pour plus de détails.
- **Impulsion PWM (largeur d'impulsion modulée):** Broches 3, 5, 6, 9, 10, et 11. Fournissent une impulsion PWM 8-bits à l'aide de l'instruction `analogWrite()`.
- **SPI (Interface Série Périphérique):** Broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la librairie pour communication SPI. Les broches SPI sont également connectées sur le connecteur ICSP qui est mécaniquement compatible avec les cartes Mega.
- **I2C:** Broches 4 (SDA) et 5 (SCL). Supportent les communications de protocole I2C (ou interface TWI (Two Wire Interface - Interface "2 fils"), disponible en utilisant la librairie `Wire/I2C` (ou `TWI - Two-Wire interface - interface "2 fils"`).
- **LED:** Broche 13. Il y a une LED incluse dans la carte connectée à la broche 13. Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte.

1.1.3. Broches analogiques

La carte Uno dispose de 6 entrées analogiques (numérotées de 0 à 5) comme montrées dans la figure ci-dessus, chacune pouvant fournir une mesure d'une résolution de 10 bits (c.-à-d. sur 1024 niveaux soit de 0 à 1023) à l'aide de la très utile fonction `analogRead()` du langage Arduino. Par défaut, ces broches mesurent entre le 0V (valeur 0) et le 5V (valeur 1023), mais il est possible de modifier la référence supérieure de la plage de mesure en utilisant la broche AREF et l'instruction `analogReference()` du langage Arduino.

Note : les broches analogiques peuvent être utilisées en tant que broches numériques, elles sont numérotées en tant que broches numériques de 14 à 19.

1.1.4. Autres broches

Il y a deux autres broches disponibles sur la carte :

- **AREF :** Tension de référence pour les entrées analogiques (si différent du 5V).
- **Reset :** Mettre cette broche au niveau BAS entraîne la réinitialisation du microcontrôleur. Typiquement, cette broche est utilisée pour ajouter un bouton de réinitialisation sur le circuit qui bloque celui présent sur la carte.

III. Le contenu du projet

La réalisation d'un système d'alarme dans une usine sert évidemment à faciliter la détection des erreurs avant que l'assemblage des câbles soit terminé ce qui donne une bonne gestion de temps en évitant le rework. Et par suite ce travail, a été divisé en trois parties :

- ✓ Partie électrique.
- ✓ Partie électronique.
- ✓ Partie informatique.

I. Partie électrique

Cette partie comportera les différents équipements qu'on a utilisés dans l'armoire électrique et le rôle de chacun.



1.1. Interrupteur différentiel

L'interrupteur différentiel : il établit, supporte et coupe des courants dans les conditions de service normales et provoque l'ouverture des contacts quand le courant différentiel atteint, dans des conditions spécifiques, la valeur de sensibilité.

Figure 16: Interrupteur différentiel.

1.2. Disjoncteur



Disjoncteur, un appareil électromécanique de connexion qui permet d'assurer la protection contre les court-circuités, les surcharges et la protection des personnes contre les contacts indirects. D'un ou plusieurs circuits en cas de surintensité ou surcharge, et a pour but de protéger les installations électriques, des courants d'intensités trop élevées susceptibles de les endommager .Sa fonction principale est d'assurer la protection du circuit qu'il alimente.

Figure 17: Disjoncteur.

Les caractéristiques principales d'un disjoncteur :

- ✓ La tension assigné ou tension d'utilisation
- ✓ Le courant assigné ou courant d'utilisation dans les cas normaux
- ✓ Pdc pouvoir de coupure ou courant maximal que peut couper le disjoncteur
- ✓ La courbe de déclenchement



1.3. Alimentation stabilisée

Cette alimentation stabilisée contient : un transformateur pour passer du 220V alternatif à une tension plus basse de 24V alternatif par exemple ; un dispositif redresseur à diodes, permettant d'avoir un courant pulsé, mais circulant toujours dans le même sens et enfin un dispositif de lissage qui permet d'avoir une tension ou un courant constant, appelé courant

Figure 18: Power supply

continu. Il en est de même des alimentations que l'on trouve dans l'unité centrale de tous les ordinateurs de bureau.

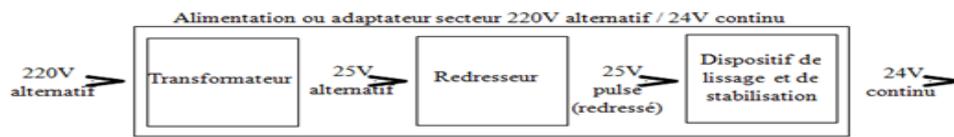


Figure 19:Alimentation 220V alternatif /24V continu

Le transformateur monophasé convertit de l'énergie électrique alternative en énergie électrique alternative ou encore, si on considère l'échange d'énergie pendant une seconde, il convertit une puissance électrique alternative en une puissance électrique alternative. C'est un convertisseur alternatif / alternatif, dont le symbole est le suivant :

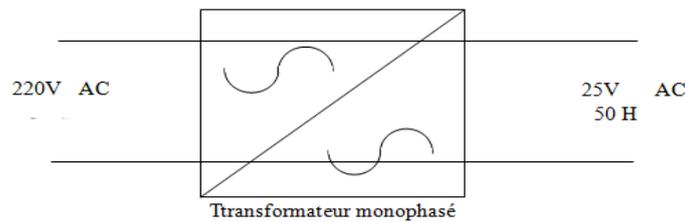


Figure 20 : Convertisseur alternatif / alternatif

Le redresseur convertit une puissance électrique alternative en une puissance électrique redressée. En électrotechnique, on dit que c'est un convertisseur alternatif / continu, dont le symbole est le suivant :

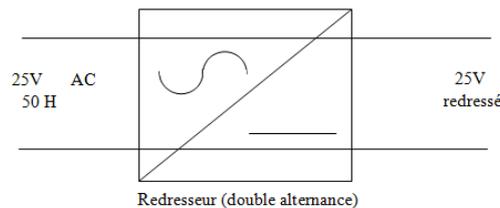


Figure 21 : Convertisseur alternatif / continu

Le lissage de tension par condensateur est réservé aux courants faibles (typiquement inférieurs à 5A dans les adaptateurs secteurs et les alimentations).

1.4. Fusible



Sert à protéger l'installation contre les surtensions fortes et courts ou courts circuits.

Figure 22:Fusible

1.5. Gyrophare



Figure 23:Gyrophare 24V/1 .8A

C'est un composant électronique de la famille des composants optoélectroniques capable d'émettre de la lumière lorsqu'un courant électrique le parcourt, c'est l'outil qu'on utilise pour déclarer l'alarme.

2. Partie électronique

Cette partie sera consacrée aux listings de toutes les composantes électroniques qui seront utilisés dans ce système d'alarme.

2.1. Arduino



Figure 24:Carte Arduino

L'Arduino joue un rôle déclencheur dans ce projet, c'est lui la base de ce système, le rôle qui lui est attribué sera de lier toutes ces parties (électriques, électroniques et informatique) pour ensuite traiter les informations entrantes et sortantes.

2.2. Capteur de température d'humidité DHT11



Figure 25 : Capteur DHT11

2.2.1.Description

Ce circuit comprend un capteur DHT11 qui fournit une information numérique proportionnelle à la température et l'humidité mesurée par le capteur. La technologie utilisée par le capteur DHT11 garantit une grande fiabilité, une excellente stabilité à long terme et un temps de réponse très rapide.

Chaque élément DHT11 est calibré avec précision en laboratoire. Le coefficient d'étalonnage est mémorisé dans la mémoire OTP interne.

L'interface unifilaire série permet l'intégration de ce capteur dans le système numérique d'une manière rapide et simple.

L'interface physique du capteur est réalisée grâce à un connecteur au pas de 0,1 "à 3 broches : +5 V, GND et DATA. Les deux premiers pins sont l'alimentation et la

masse et ils sont utilisés pour alimenter le capteur, le troisième est le signal de sortie du capteur numérique.

2.2.2. Caractéristiques

- Alimentation : 5V
- Consommation : 0.5 mA en nominal / 2.5 mA maximum
- Étendue de mesure température : 0°C à 50°C \pm 2°C
- Étendue de mesure humidité : 20-90%RH \pm 5%RH

2.3. Les Relais



Figure 26 : Carte relais

Il sert à faire une transition entre un courant faible et un courant fort. Mais il sert également à commander plusieurs organes simultanément grâce à ses multiples contacts synchronisés. Il permet également la transition entre deux sources différentes en isolant ces dernières.

2.4. Afficheur LCD

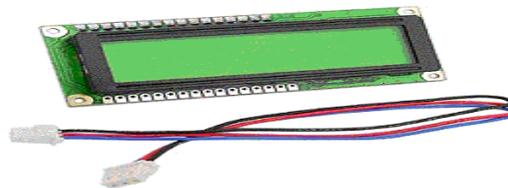


Figure 27 : Afficheur LCD

Les afficheurs à cristaux liquides sont des modules compacts intelligents et nécessitent peu de composants externes. Ils sont utilisés avec beaucoup de facilité. Ils sont pratiquement les seuls à être utilisés sur les appareils à alimentation par pile.

2.5. Capteur de courant ACS712



Figure 28 : Capteur ACS712

Le capteur ACS712 donne une mesure précise du courant pour les deux signaux AC et DC. L'ACS712 émet un signal de sortie analogique de tension qui varie linéairement avec courant détecté. Le dispositif exige 5V DC et un couple de

condensateurs de filtrage.

Caractéristiques :

- Faible niveau de bruit analogique chemin du signal.
- 1,5% d'erreur de sortie à 25 degrés C.
- Factory parée pour la précision.
- La tension de sortie proportionnelle aux courants AC ou DC.

2.6. La liaison RS232



Figure 29 : Câble RS232

La liaison RS232 est une liaison série qui permet de transmettre des informations sans avoir à transmettre l'horloge de synchronisation. Elle utilise des tensions de fonctionnement non compatibles avec la logique 5V, car elle fonctionne sur les niveaux +12V ou -12V. Son principal intérêt est un nombre de fils réduit, puisque trois suffisent à la transmission d'informations : un fil pour émettre, un autre pour recevoir et un dernier pour la masse.

Pour notre projet, nous avons dû lier la connexion du port série du PC avec Arduino.

3. Partie informatique

3.1. Microsoft Visual C#

Aujourd'hui, de nombreuses personnes développent en C et C++ .Néanmoins, ces langages ont aussi un certain nombre de défauts dû à leur ancienneté.

VB fait partie des langages suivant le concept orienté objet, il a un environnement de développement intégré (EDI) qui permet de développer facilement des interfaces graphiques.

Critère de choix du logiciel :

Nous avons décidé de travailler avec Microsoft Visual C# pour développer l'application, ce choix est basé sur les points suivants :

- Le langage est relativement facile et simple à apprendre.
- Le logiciel est gratuit et disponible dans la société.
- Le langage semble être le nouveau langage pour développer des applications Windows, avec Visual Basic.

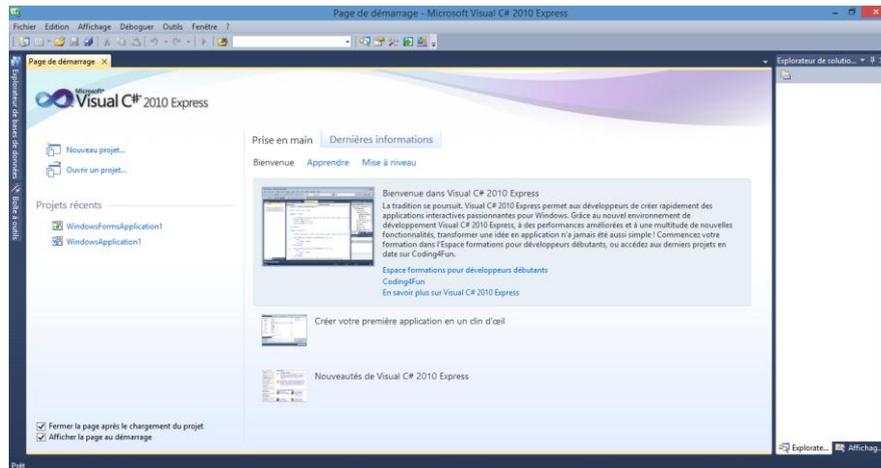


Figure 30: Interface Microsoft Visual C#

3.2. Langage MySQL

MySQL est un système de gestion de bases de données relationnelles. Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, Informix et Microsoft.

Critère de choix du MySQL :

- La gratuité et la disponibilité du code source.
- La simplicité de liaison avec des bases de données.

3.3. Logiciel de programmation Arduino

L'environnement de programmation Arduino (IDE en anglais) est une application écrite en java inspiré du langage « Processing ». L'IDE permet d'écrire, de modifier un programme et de le convertir en une série d'instruction compréhensible pour la carte.

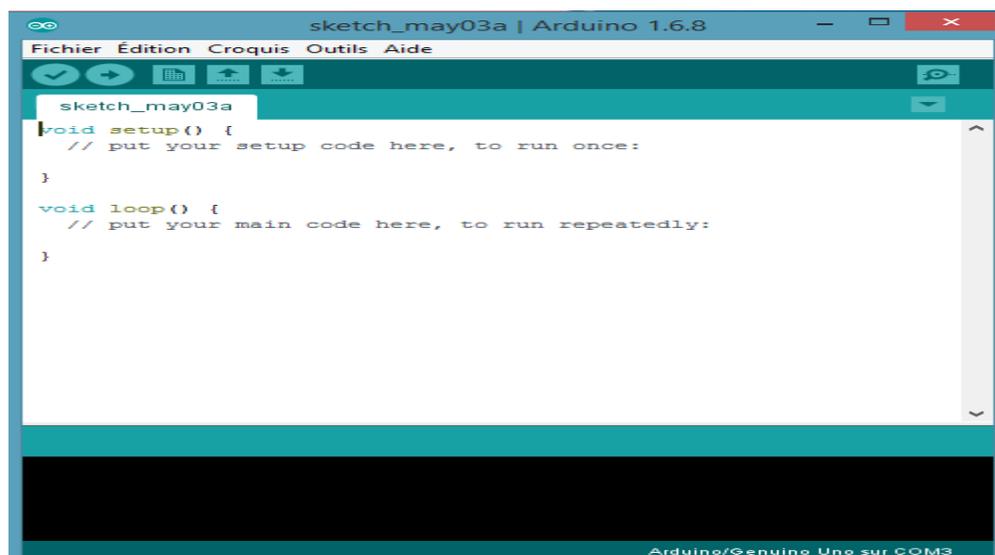


Figure 31 : Logiciel de l'Arduino

3.4. Logiciel de simulation ISIS

ISIS est un éditeur de schémas qui intègre un simulateur analogique, logique ou mixte. Toutes les opérations se passent dans cet environnement, aussi bien la configuration des différentes sources que le placement des sondes et le tracé des courbes.

La simulation permet d'ajuster et de modifier le circuit comme si on manipulait un montage réel. Ceci permet d'accélère le prototypage et de réduire son cout. Il faut toujours prendre en considération que les résultats obtenus de la simulation sont un peu différent de celles du monde réel, et ça dépend de la précision des modèles.

Son interface d'utilisation est le suivant :

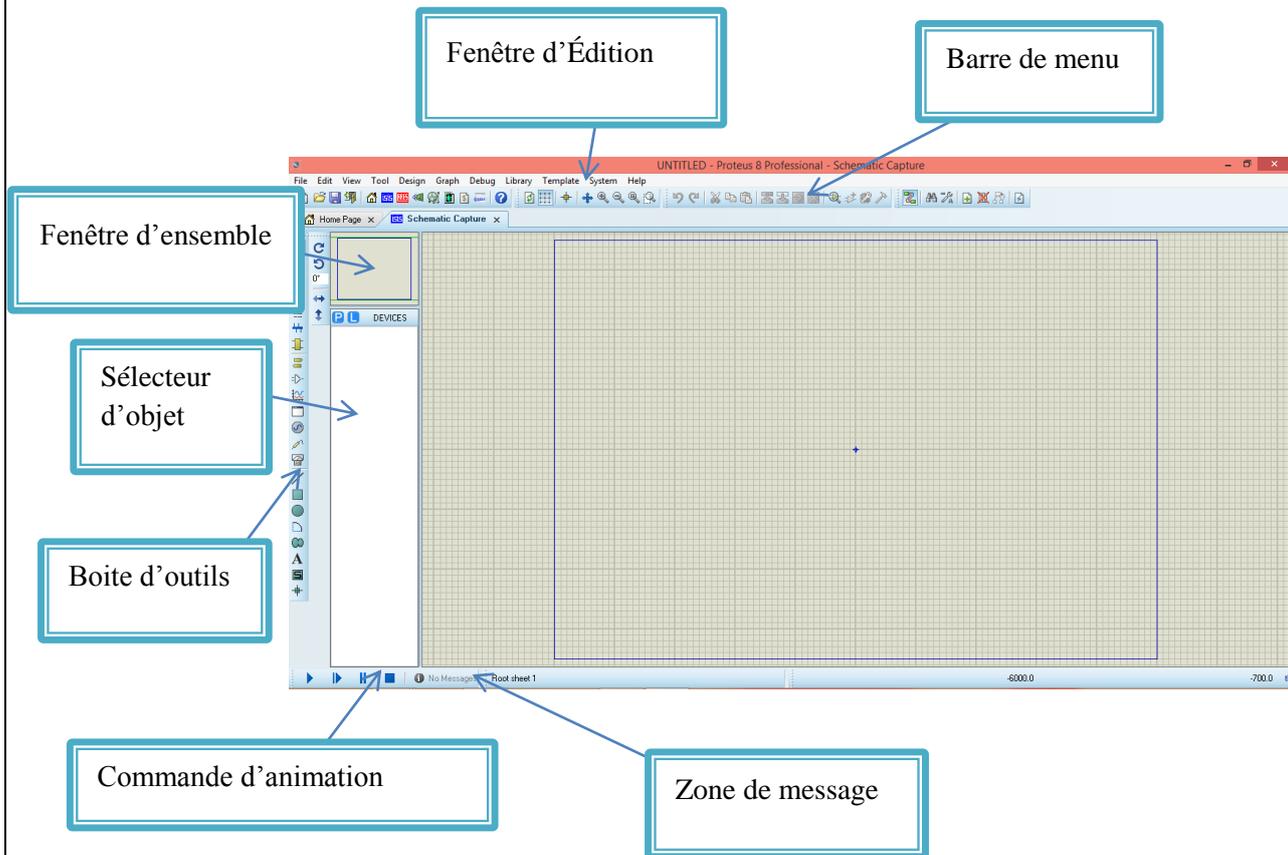


Figure 32:Interface ISIS

3.5. WinRelais

WinRelais est un logiciel de saisie de schémas électrotechniques de types unifilaires, multifilaires, architecturaux et développés, pour le bâtiment et l'industrie. Il permet aussi de dessiner des grafquets ainsi que des schémas pneumatiques et hydrauliques.

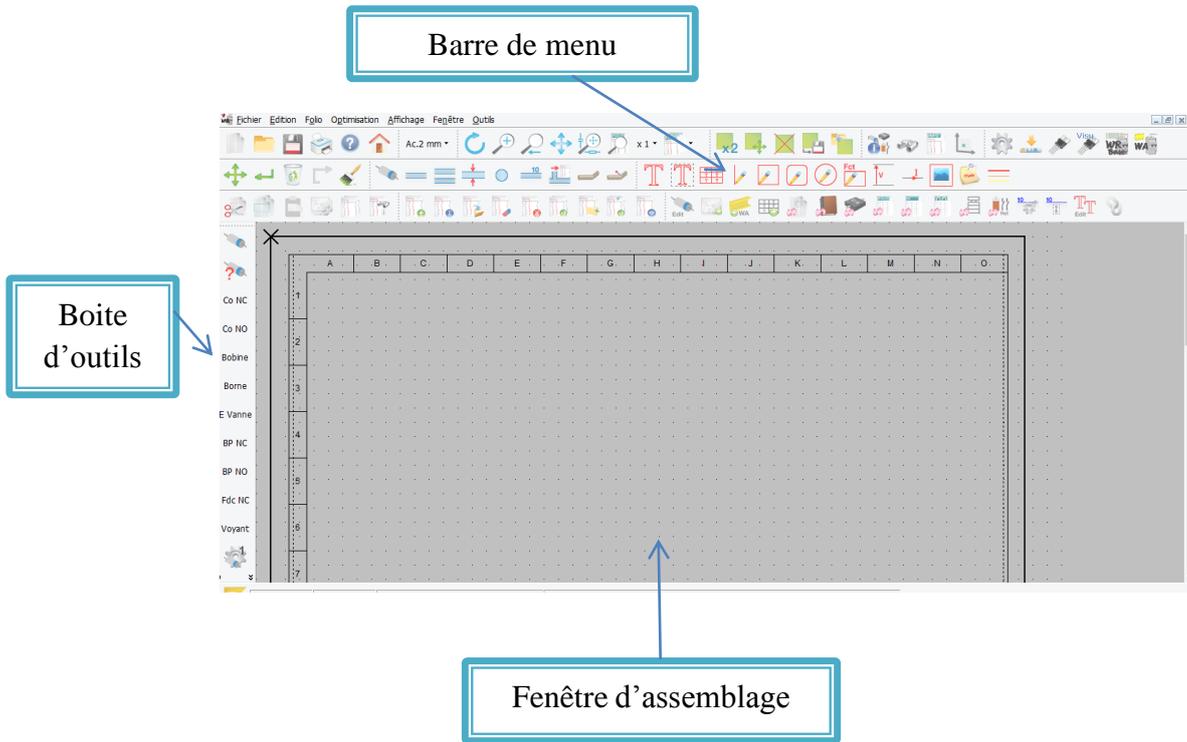
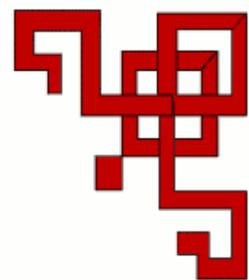
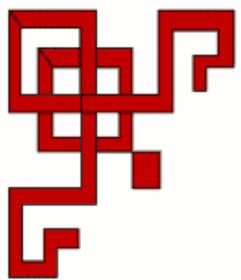


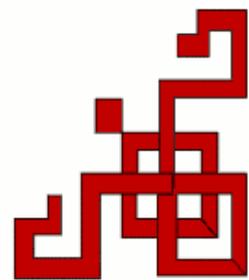
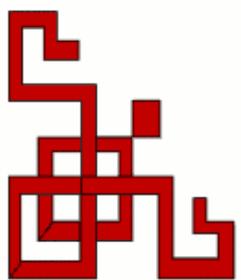
Figure 33 : Interface WinRelais

Conclusion

En guise de conclusion ce chapitre a été dédié à l'initiation des différents étages du dispositif tout en procurant au lecteur une idée générale sur les caractéristiques techniques.



Chapitre 4 : Réalisation et simulation



Introduction

La phase de réalisation est la phase qui englobe l'étape de la programmation, l'étude technique, l'étude d'environnement de travail et les différents outils de développement.

I. Simulation du système d'alarme

1. Fonctionnement du système

Le rôle de notre système est de vérifier la compatibilité de la partie du câble codée arrivant de l'infotainment avec le câble de convoyeur assemblage.

Cette comparaison des deux codes barre se fait grâce à l'application C# qu'on a créé et qui permet de comparer des deux codes-barres qu'on fait rentrer et donner le résultat de la vérification, si les codes-barres se coïncident rien ne se passe sinon il envoie la commande à l'Arduino pour déclencher l'alarme.

Le gyrophare est alimenté grâce à une boîte électrique et sa marche/arrêt est commandée par l'Arduino.

L'organigramme ci-dessous montre le fonctionnement de notre système :

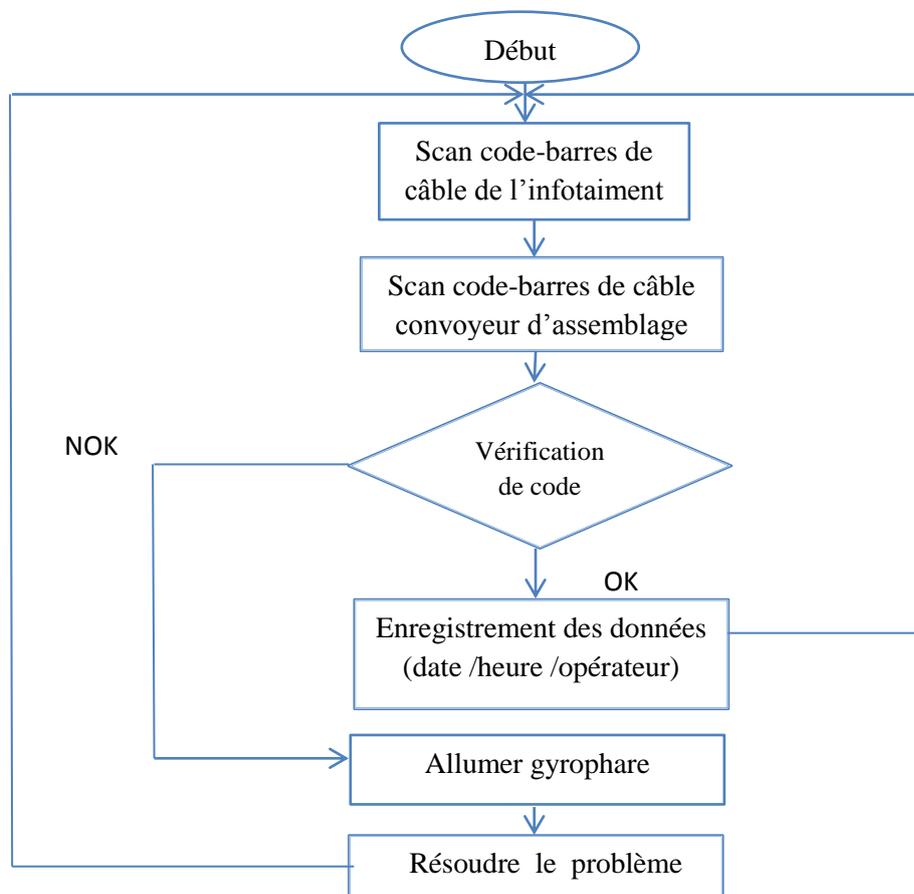


Figure 34: Organigramme du système

2. Déroulement de la simulation

2.1. Connexion d'Arduino

Avant de passer à la réalisation pratique de notre système, nous avons fait recours à la simulation de différentes parties du système. Pour cela on a utilisé le logiciel ISIS de simulation.

Voici le schéma de montage complet :

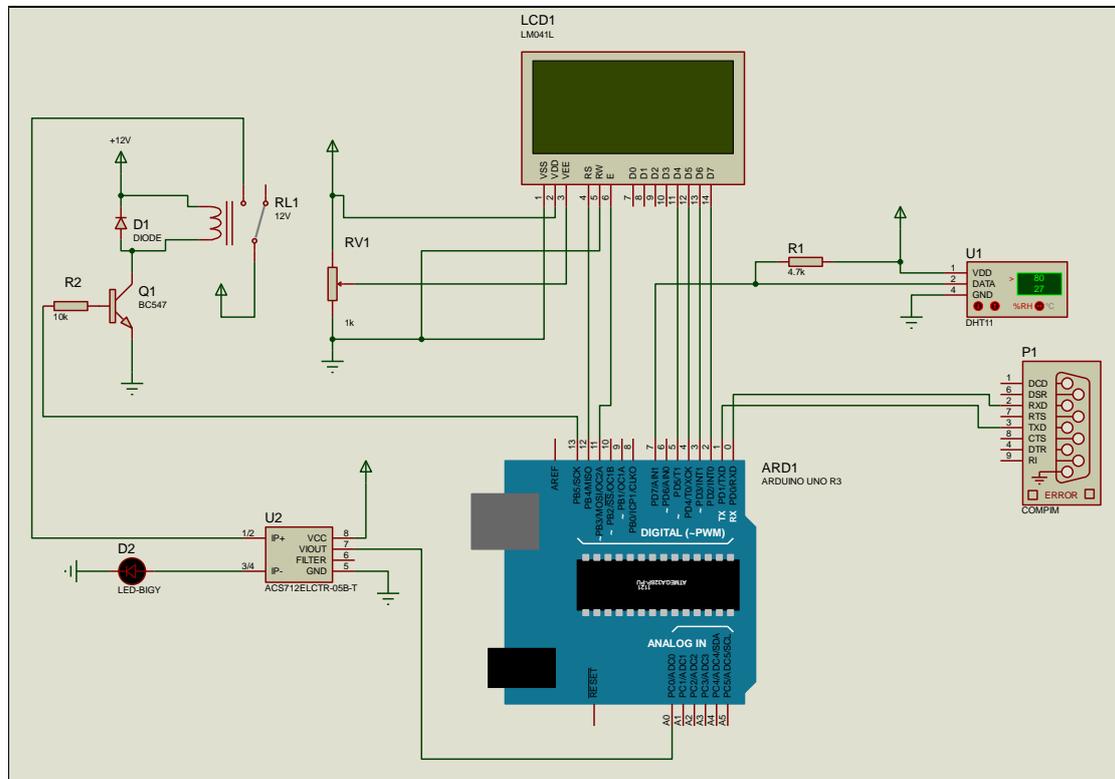


Figure 35 : Schéma du montage électronique

Pour bien simuler le processus de fonctionnement du système et éviter une erreur sans pouvoir la détecter, il est nécessaire de simuler chaque partie indépendante du système global :

2.1.1. Capteur DHT11

Comme son nom l'indique, il capte la température et l'humidité dans l'armoire électrique.

Pour superviser l'état de température et d'humidité on affiche les valeurs détectées par le capteur DHT11 sur un afficheur LCD comme ça au cas de dépassement de limite une intervention sera faite.

2.1.2. Capteur du courant

Son rôle sera de savoir la quantité du courant délivrée au gyrophare qui est schématisé par une LED jaune.

Une fois le module est sous tension les valeurs de température, humidité et courants s'affichent dans l'écran LCD.

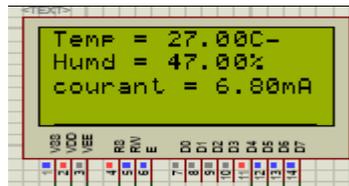


Figure 36: Visualisation d'état des capteurs

⇒ Lorsque le gyrophare est atteint ou en panne la valeur du courant sera de 0A.

2.1.3. Interface commande puissance

Pour schématiser le relais sur le logiciel de simulation ISIS, on le remplace par un bloc transistor, sur le schéma donné ci-devant, le transistor Q1 est l'élément de commande qui commute le courant dans la bobine du relais.

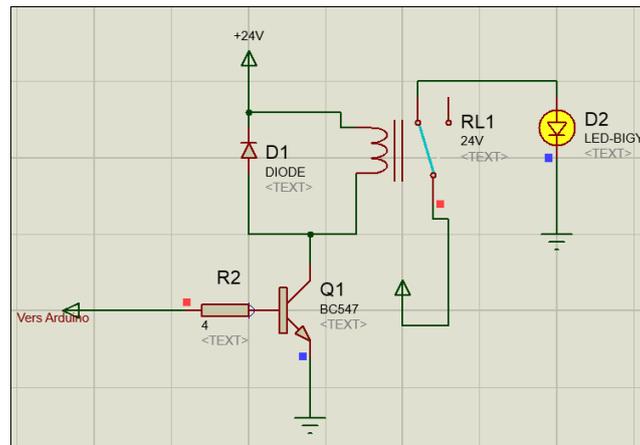


Figure 37 : Schéma de commande du gyrophare

La liaison entre l'Arduino et le relais se fait de la manière suivante ; connecter le pin « entré » du relais au pin (pin 13) « sorti » de l'Arduino UNO. Le schéma suivant représente cette liaison :

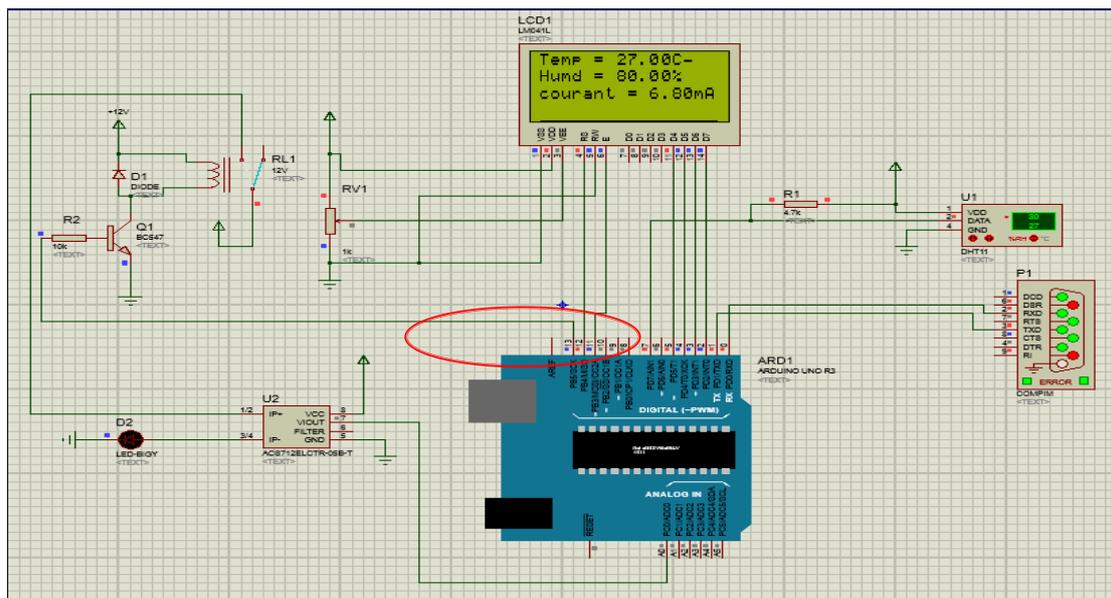


Figure 38: Liaison Arduino relais

2.2. Alimentation relais

L'alimentation stabilisée est liée directement avec le relais qui est lié à son tour au gyrophare. Ce dernier est alimenté à 24V.

Le relais constitue l'intermédiaire entre les deux parties électrique et électronique, c'est lui qui transporte la commande de l'Arduino vers le gyrophare.

On a réalisé ce système émettant 24v et qui sera illustré dans le schéma ci-après :

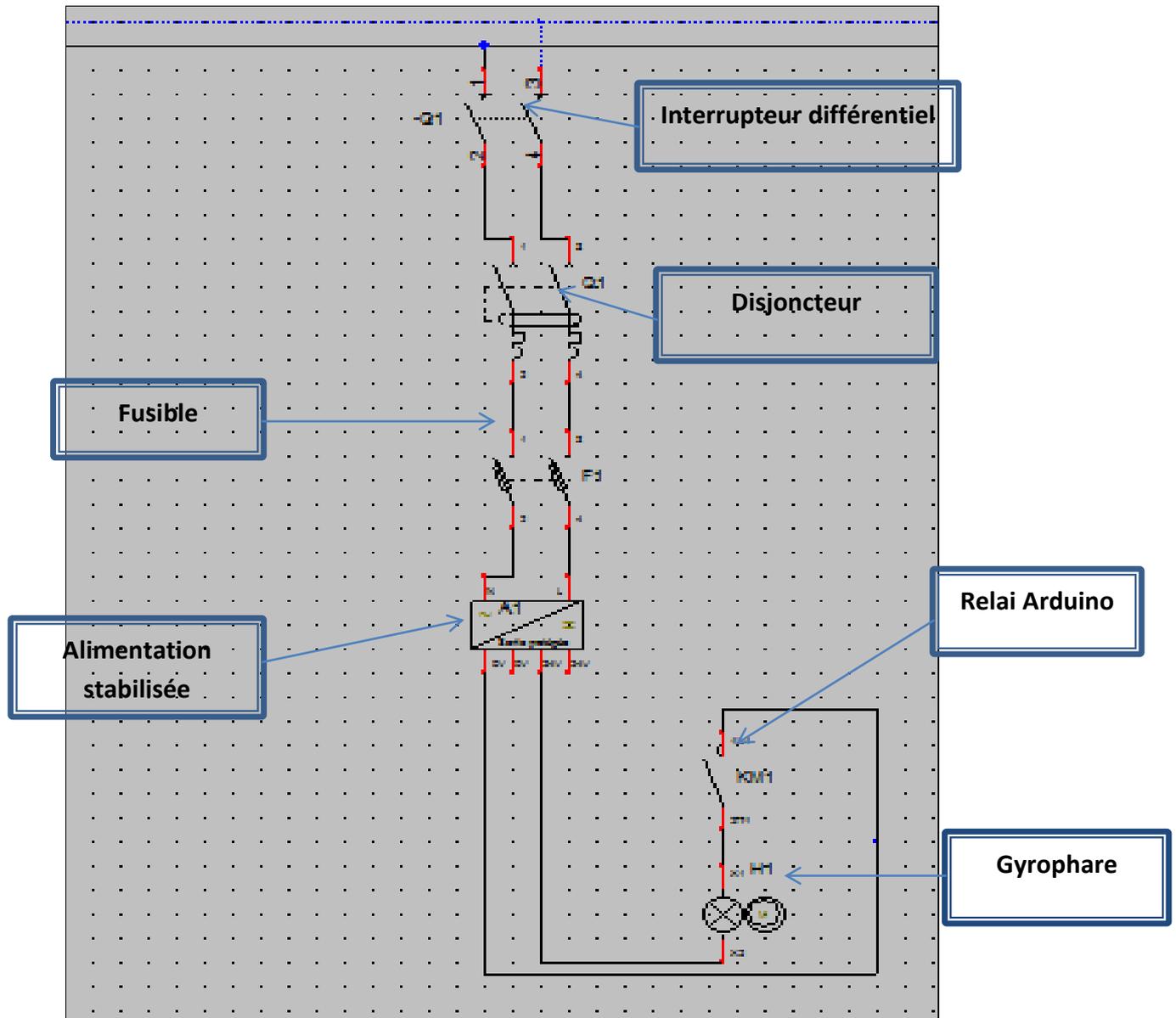


Figure 39: Schéma d'alimentation (puissance)

Ce schéma a été dessiné par l'intermédiaire de WinRelais.

Après avoir réalisé la partie technique du projet, le système est déjà prêt à accueillir la commande. Cette commande va dépendre du résultat de la comparaison comme déjà mentionné. Et pour faire, il faut lier la carte Arduino à l'application C#. Cette liaison se fait à travers le Protocole RS232.

L'application d'identification et de vérification de C# et la liaison de cette dernière avec la carte Arduino seront détaillées dans la partie suivante.

2.3. Connexion de l'Arduino avec C#

2.3.1. Introduction

C'est dans cette partie qu'on doit programmer notre carte Arduino et de créer l'application C# afin d'accomplir le fonctionnement désiré. Cela consiste à créer l'application de vérification des codes-barres et lier ça avec l'Arduino via la liaison de communication série RS232. L'Arduino sera programmé à son tour pour déclencher le gyrophare au cas de non-conformité.

L'algorithme de notre système est le suivant :

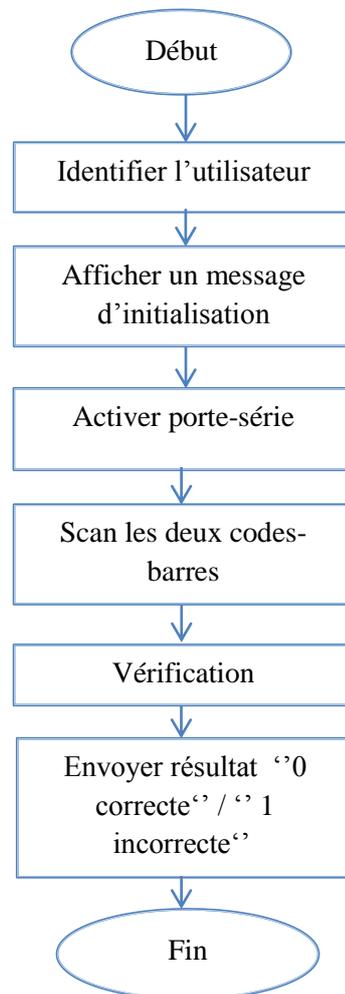


Figure 40 : Algorithme de programme sur C#

2.3.2. Interface de vérification C#

L'interface de la vérification que nous avons réalisée est le suivant :

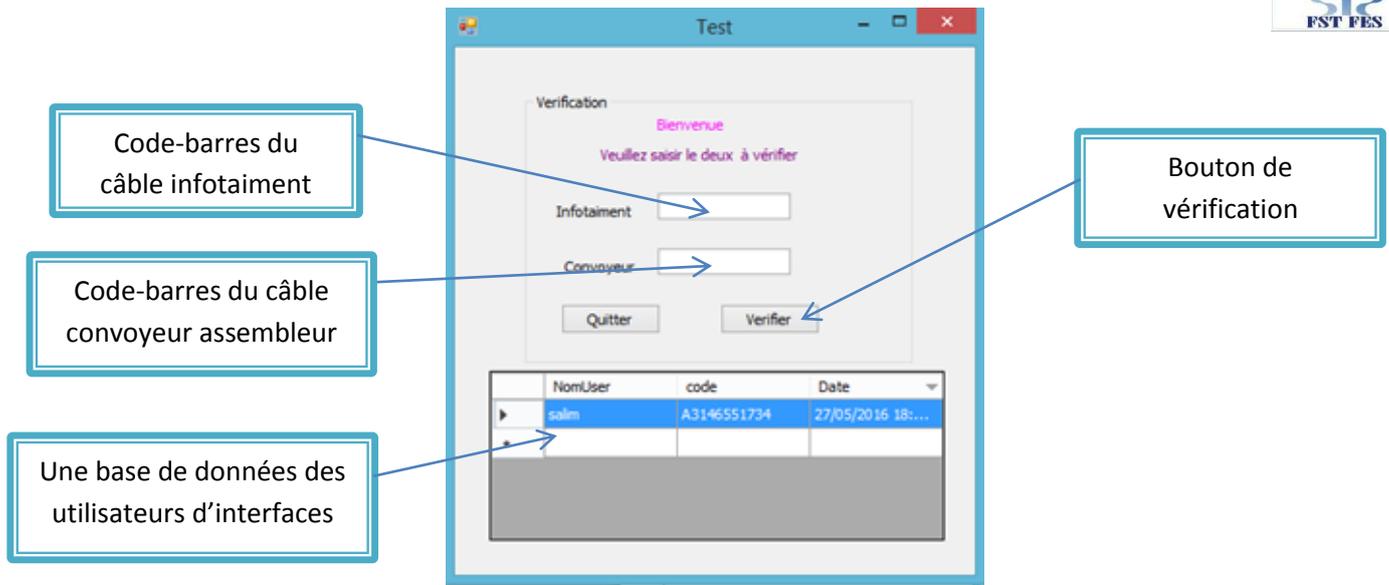


Figure 41: Interface de vérification dans C#

Comme déjà mentionné c'est à ce niveau que la comparaison entre les deux codes-barres se fait. Et c'est le résultat de cette comparaison (envoyé à l'Arduino via la communication série) qui le commande.

2.4. Connexion C# et la base de données

Pour des raisons de sécurité, l'accès à cette application de vérification serait protégé par un nom d'utilisateur et un code formé d'au moins de 4 caractères.

Dans notre cas, il nous a tout d'abord fallu créer des identifiants pour l'accès de l'opérateur à l'application. L'inscription de l'opérateur n'étant pas requise, nous avons tout simplement créé une table « Admin » avec un champ pour le pseudonyme et un autre pour le mot de passe.

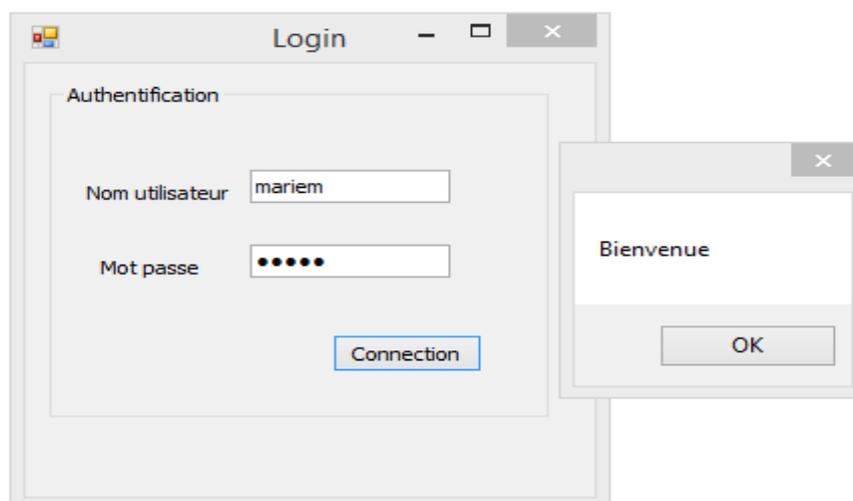


Figure 42: Interface d'identification dans C #

2.5. Simulation du système final sur ISIS

Cas 1 : les deux codes sont compatibles :

Dans le cas où les deux codes-barres saisis sont compatibles, le message suivant s'affiche « les deux codes-barres sont compatibles » et le gyrophare reste éteint.

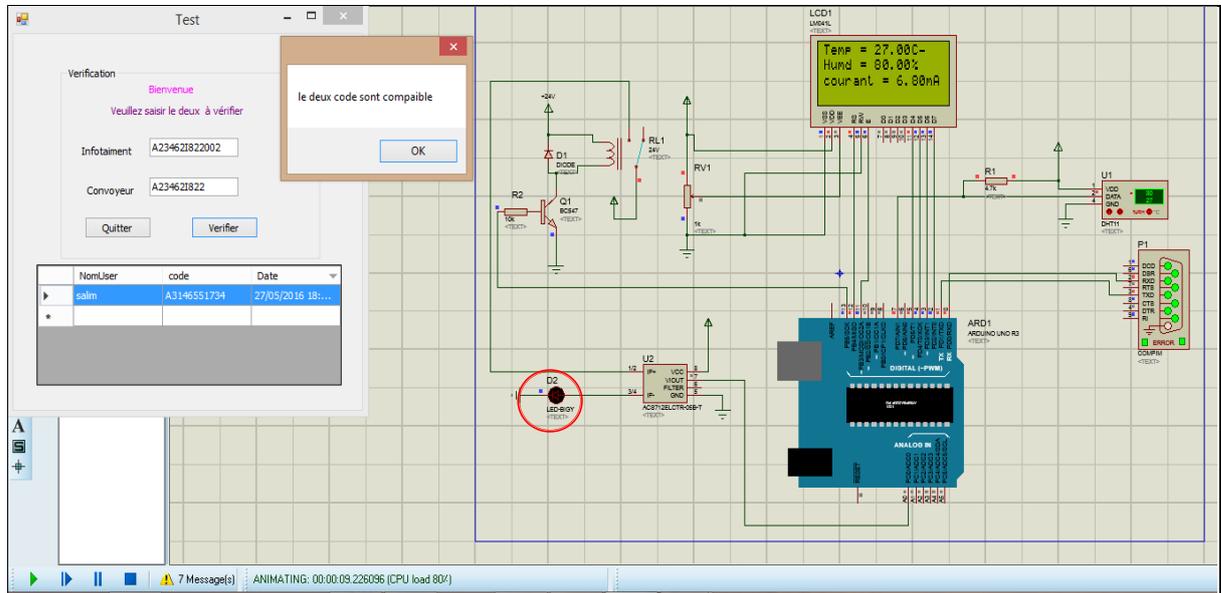


Figure 43: Les codes compatibles

Cas 2 : les deux codes ne sont pas compatibles :

Dans le cas où les deux codes-barres saisis ne sont pas compatibles pas, le message suivant s'affiche « les deux codes-barres ne sont pas compatibles » et le gyrophare s'allume pour mettre alarme à l'opérateur.

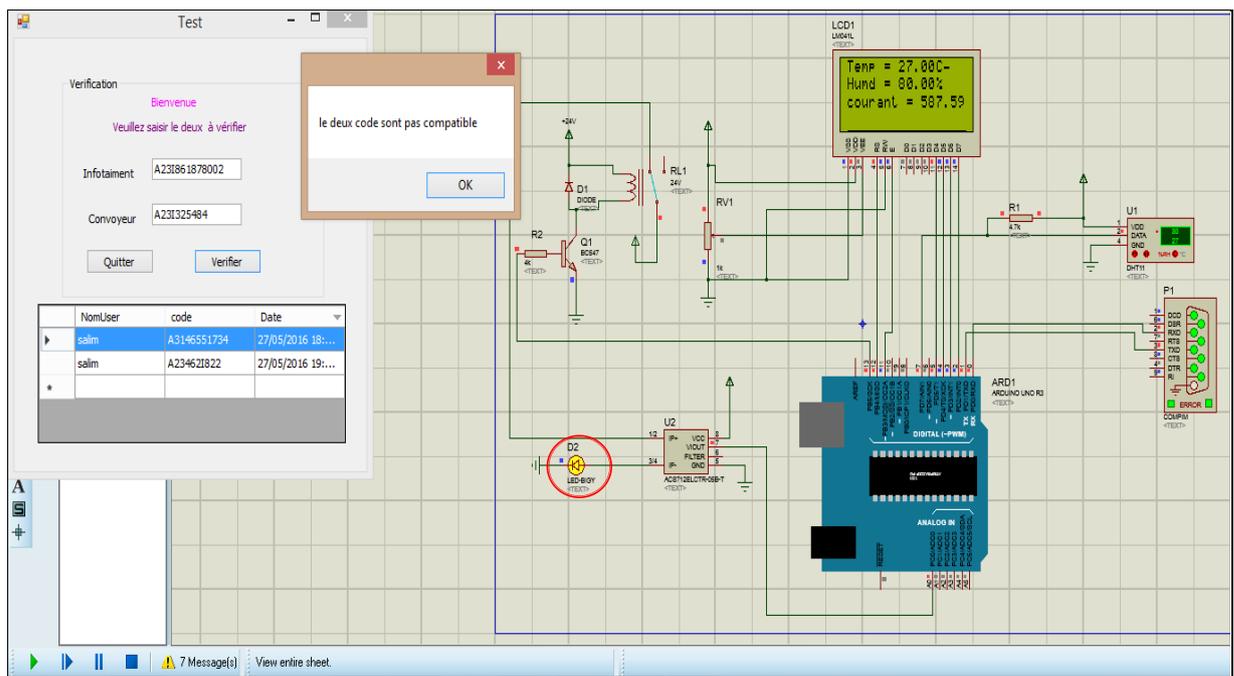


Figure 44: Les codes ne sont pas compatibles

II. Réalisation matérielle

Un premier test était déjà fait à la fin de la période de notre stage pour tester l'efficacité de nos codes et pour mettre en place le système.

Le code marche bien et le gyrophare s'allume à chaque fois que les deux codes-barres ne sont pas compatibles.

Mais le système n'est pas encore mis en place officiellement pour la ligne, on attend toujours l'accord de l'administration pour pouvoir la dupliquer et la mettre en place.



Figure 45: Mise en place du premier test du système d'alarme.

Conclusion

Dans ce chapitre nous nous sommes intéressées à l'amélioration d'une partie importante dans la société YAZAKI qui est la partie organisationnelle. En effet, l'application réalisée a pour but d'améliorer la productivité de la ligne de production.

Dans un premier temps, nous sommes consacrées à la réalisation du branchement de notre système et dans un deuxième temps, nous avons décrit ses différentes fonctionnalités.

Et à la fin un premier test était primordial pour valoriser notre travail.

Conclusion générale

L'objectif de ce projet était de réaliser un système d'alarme grâce à une carte Arduino tout en améliorant la productivité du système de production. Dans la finalité sont d'annuler les erreurs commises lors du montage de l'infotaiment.

Grâce au logiciel C#, il nous a été possible de développer une application de vérification. Le couplage avec un Arduino nous apporte l'avantage de pouvoir programmer les différentes tâches et conditions nécessaires à la réalisation du système.

Le but de notre application étant de vérifier les codes-barres du câble infotaiment et le câble principal d'une façon automatique, il était nécessaire d'avoir recours à une vérification d'utilisateur en créant une base de données qui limite l'accès au système.

Notre choix s'est immédiatement porté sur le programme MySQL pour gérer cette base de données.

Ensuite, avec de la programmation, nous avons défini les limites de l'application et les différentes conditions nécessaires à son bon fonctionnement.

Le logiciel ISIS avait un grand rôle pour simuler la démarche du système et assurer la liaison entre C# et Arduino.

En conclusion nous pouvons constater le bon fonctionnement de notre produit. L'application réalisée est opérationnelle en ce moment dans la société, et utilisée primordialement pour être générée plus tard sur toutes les lignes de production de la société qui souffre de cette confusion.

Ce projet a été intéressant et bénéfique pour nous dans plusieurs sens et nous a permis de :

- ✓ Nous perfectionner en améliorant nos connaissances en programmation et conception.
- ✓ Mettre en œuvre le déroulement d'une application qui nous permettra de bien vérifier le montage du câble avant d'entamer le test électrique.
- ✓ Découvrir le monde de l'entreprise « Fonctionnement ».

Webographie

Arduino :

- <http://arduino.cc/en/Main/Software> / (consulté le 12 /04/2016)
- <http://arduino.cc/> (consulté le 14 /04/2016)
- <http://www.multimedialab.be/cours/logiciels/arduino.htm/> (consulté le 14 /04/2016)

Capteur de courant ACS712 :

- datasheet-ebs011.pdf/ (consulté le 20 /04/2016)
- <http://www.electrodragon.com/acs712-current-sensor-read/> (consulté le 24/04/2016)
- mr003-006_datasheet.pdf/ (consulté le 20 /04/2016)

Capteur DHT11 :

- DHT11.pdf/ (consulté le 26 /04/2016)
- <https://arduino-info.wikispaces.com/DHT11-Humidity-TempSensor/> (consulté le 26 /04/2016)

Autres :

- <https://en.wikipedia.org/> (consulté le 20 /04/2016)
- [http://www.yazaki-europe.com /](http://www.yazaki-europe.com/) (consulté le 05 /04/2016)

Annexe : Code application sur Arduino UNO :

```
#include "DHT.h"

#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

#define DHTPIN 7

#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE)

void setup() {

  pinMode(13,OUTPUT);

  Serial.begin(9600);

  dht.begin();

  lcd.begin(16, 4);

}

void loop() {

  float h = dht.readHumidity();

  float t = dht.readTemperature();

  lcd.setCursor(0,0);

  lcd.print("Temp = ");

  lcd.print(t);

  lcd.print("C°");

  lcd.setCursor(0,1);

  lcd.print("Humd = ");

  lcd.print(h);

  lcd.print("% ");

  Serial.println(t);

  Serial.println(h);

  float average = 0;
```

```
for(int i = 0; i < 1000; i++) {  
  
    average = average + (.0264 * analogRead(A0) -13.51);  
  
}  
  
lcd.setCursor(0,2);  
  
lcd.print("courant = ");  
  
lcd.print(average);  
  
lcd.print("mA");  
  
if(Serial.available())  
  
    {  
  
        int c = Serial.read();  
  
        if (c == '1')  
  
            {  
  
                digitalWrite(13,HIGH);  
  
            }  
  
        else if (c == '0')  
  
            {  
  
                digitalWrite(13,LOW);  
  
            }  
  
    }  
  
    delay(500);  
  
}
```

Annexe : Code application de vérification sur Microsoft Visual C# :

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Projet_stage
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
            serialPort1.PortName = "COM1";
            serialPort1.BaudRate = 9600;
            serialPort1.Open();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            string x = txtinf.Text;
            string y = txtmer.Text;
            if (x.Substring(0, 10) == y.Substring(0, 10))
            {
                serialPort1.Write("0");
                MessageBox.Show("le deux code sont compaible");
                dataGridView1.Rows.Add(Login.username, txtmer.Text,
DateTime.Now.ToString());
                txtinf.Clear();
                txtmer.Clear();
            }

            else
            {
                serialPort1.Write("1");
                MessageBox.Show("le deux code sont pas compatible ");
                txtinf.Clear();
                txtmer.Clear();
            }
        }

        private void button2_Click(object sender, EventArgs e)
        {
            Application.Exit();
        }
    }
}
```

Annexe : Code application identification sur Microsoft Visual C# :

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;
namespace Projet_stage
{
    public partial class Login : Form
    {
        public static string username;

        public Login()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            SqlConnection Cn = new SqlConnection("Data
Source=INTEL\\SQLEXPRESS;Initial Catalog=FIL;Integrated Security=True");
            Cn.Open();
            SqlCommand Cmd = new SqlCommand("select *from Utilisateur where NOM
= '" + txtUtil.Text + "'and MOTPASS = '" + txtPass.Text + "'", Cn);
            SqlDataReader dr;
            dr = Cmd.ExecuteReader();
            int Cmpt = 0;

            while(dr.Read())
            {
                Cmpt += 1;
            }
            if(Cmpt == 1)
            {
                Login.username = txtUtil.Text;
                MessageBox.Show("Bienvenue" );
                Form2 x = new Form2();
                x.Show();
            }
            else if(Cmpt > 0){
                MessageBox.Show("duplication du no et le mot de pass");
            }
            else {
                MessageBox.Show("le nom ou le mot de pass incorrecte");
            }

            txtUtil.Clear();
            txtPass.Clear();
            this.Hide()

        }
    }
}

```