

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université Ibn Khaldoun de Tiaret
Faculté des Sciences Appliquées
Département de Génie Mécanique



MÉMOIRE DE FIN D'ETUDES

Pour l'obtention du Diplôme de Master
Domaine : Sciences et Technologie
Filière : Electromécanique
Spécialité : Maintenance Industrielle

Thème

**L'étude et l'amélioration d'un système de
gestion de la maintenance assisté par
ordinateur (GMAO)**

Préparé par :

DENFAR MOHAMED et ZERROUKI SID AHMED.

Soutenu publiquement le : 13 / 07 / 2021, devant le jury composé de :

M. MAKHFI Souad	Maître de Conférences "B" (Univ. Ibn Khaldoun)	Président
M. SAAD Mohamed	Maître de Conférences "B" (Univ. Ibn Khaldoun)	Examineur
M. BENAMAR Badr	Maître assistant "A" (Univ. Ibn Khaldoun)	Examineur
M. BENSATTALAH. A	Maître de Conférences "B" (Univ. Ibn Khaldoun)	Encadreur

Année universitaire : 2020 - 2021

REMERCIEMENTS

Nous tenons tout d'abord à remercier l'université "**Ibn Khaldoun**" de **TIARET**, spécialement le département de Génie mécanique et tous nos enseignants.

Nous adressons également nos remerciements à Mr le président du jury et Mrs les membres du jury pour avoir accepté d'examiner et de juger notre travail.

Comme, nous exprimons notre profonde gratitude et nos sincères remerciements à notre encadreur Mr. **BENSATTALAH Aissa**, pour la qualité de son encadrement et pour leur soutien sa disponibilité et leur conseils nous ont été d'une grande utilité afin de mener à bien notre modeste travail, nous avoir dirigés durant la période de préparation de notre mémoire de Master.

On remercie infiniment tous nos amis de leurs encouragements.

DEDICACES

J'ai toujours pensé faire où offrir quelque chose à mes parents en

Signe de reconnaissance pour tout ce qu'ils ont consenti comme efforts, rien que pour me voir réussir, et voilà, l'occasion est venue.

A ceux qui m'ont donné la vie, symbole de beauté, et de fierté, de sagesse et de patience.

A ceux qui sont la source de mon inspiration et de mon courage, à qui je dois de l'amour et de la reconnaissance.

 *A tous mes Amis sans exception.*

 *A tout le Département de **G.M** ainsi que les étudiants de **M.I***

TABLE DES MATIERES

INTRODUCTION GENERALE	9
1 CHAPITRE 1 : GMAO	12
1.1 INTRODUCTION	12
1.2 DEFINITION LA MAINTENANCE.....	12
1.2.1 OBJECTIFS DE LA MAINTENANCE	12
1.2.2 ROLE DE LA MAINTENANCE	12
1.2.3 TYPES DE MAINTENANCE	13
1.3 GESTION DE MAINTENANCE ASSISTEE PAR ORDINATEUR.....	17
1.3.1 CARACTERISTIQUES GENERALES	17
1.3.2 OBJECTIFS ET BUTS D'UN SYSTEME GMAO	17
1.3.3 ROLE D'UN SYSTEME DE GMAO.....	18
1.3.4 STRUCTURE DES LOGICIELS GMAO	18
1.3.5 LOGICIEL POUR GMAO.....	19
1.3.6 PRISE EN MAIN DE OPTIMAIN V 6.3.4	22
1.4 CONCLUSION	24
2 CHAPITRE2 : VIBRATION MECANIQUE.....	26
2.1 INTRODUCTION	26
2.2 DEFINITION D'UNE VIBRATION.....	26
2.2.1 SIGNAL VIBRATOIRE	26
2.2.2 TYPE DES SIGNAUX	27
2.2.3 DEFAUTS DES MACHINES TOURNANTES	29
2.3 METHODE DE L'ANALYSE VIBRATOIRE	32
2.3.1 METHODE DE NIVEAU GLOBAL	32
2.3.2 METHODE DE L'ANALYSE EN FREQUENCE	33
2.3.3 LES GROUPES DES MACHINES	38
2.3.4 LES DEFAUTS DES MACHINES TOURNANTES ET LEURS FREQUENCES	41
2.4 CONCLUSION	42
3 CHAPITRE 3 : LES SYSTEMES EMBARQUES	44
3.1 INTRODUCTION	44
3.2 PRESENTATION DES SYSTEMES EMBARQUES	44
3.2.1 DEFINITION	44
3.2.2 COMPOSITION D'UN SYSTEME EMBARQUE	44
3.2.3 CARACTERISTIQUES	45
3.2.4 ARCHITECTURE D'UN SYSTEME EMBARQUE	46
3.2.5 CLASSIFICATION DES SYSTEMES EMBARQUES	47
3.2.6 DOMAINES D'APPLICATION DES SYSTEMES EMBARQUES	48
3.3 CARTE ARDUINO	49
3.3.1 LE SYSTEME ARDUINO	49
3.3.2 LE MODULE ARDUINO	50

TABLE DES MATIERES

3.3.3 LA CARTE ARDUINO UNO	53
3.4 LES CAPTEURS DE VIBRATION	54
3.4.1 LES PROSIMETRES	54
3.4.2 VELOCIMETRIE	55
3.4.3 LES ACCELEROMETRES	55
3.5 LES CAPTEURS DE TEMPERATURES	56
3.5.1 LE CAPTEUR DHT11	56
3.5.2 CAPTEUR DE TEMPERATURE LM 35	57
3.6 LES APPLICATIONS DES VIBRATIONS MECANIQUES DANS LA MAINTENANCE	58
3.7 CONCLUSION	60
4 CHAPITRE 4 : LA CONCEPTION ET LA REALISATION D'UN SYSTEME DE GMAO AMELIORE	62
4.1 INTRODUCTION	62
4.2 ETUDE DE SYSTEME DE MAINTENANCE PAR LA METHODE UML	62
4.2.2 ANALYSE DE TEMPERATURE D'UN GROUPE MOTOPOMPE	68
4.2.3 MODELISATION PAR UML	68
4.2.4 LA REALISATION DE GMAO	71
4.3 REALISATION DE LA PARTIE DE VIBRATION ET DE TEMPERATURE	74
4.3.2 STRUCTURE GENERALE DU PROGRAMME (IDE ARDUINO)	75
4.3.3 CABLAGE	75
4.3.4 DEMARCHE DE PROGRAMMATION	76
4.3.5 PROTOTYPE	82
4.3.6 LE CONTROLE DE TEMPERATURE	85
4.4 CONCLUSION	87
CONCLUSION GENERALE	89
BIBLIOGRAPHIES	91

LISTE DES FIGURES

Numéro	Titre	Page
Figure 1.1	Types de maintenance	16
Figure 1.2	<i>Aperçu de l'interface graphique utilisateur du logiciel OptiMaint</i>	22
Figure 1.3	<i>Barre d'outils du logiciel OptiMaint V 6.3.4</i>	23
Figure 1.4	<i>Boite de dialogue demande d'intervention</i>	23
Figure 2.1	<i>Signal vibratoire</i>	22
Figure 2.2	<i>Représentation fréquentielle d'une réponse vibratoire simple</i>	23
Figure 2.3	<i>Représentation vectorielle d'une réponse vibratoire simple</i>	22
Figure 2.4	<i>Défaut de balourd [Landolsi]</i>	29
Figure 2.5	<i>Représentation graphique de diagramme de Bode</i>	31
Figure 2.6	<i>Défaut d'engrenage.</i>	32
Figure 2.8	<i>Exemple des points de mesure lors de l'analyse vibratoire.</i>	32
Figure 2.9	<i>Spectre d'amplitude d'une vibration harmonique.</i>	33
Figure 2.10	<i>Vibration périodique non sinusoïdale.</i>	34
Figure 2.11	<i>Spectre d'amplitude d'une vibration périodique.</i>	34
Figure 2.12	<i>Spectre de vitesse vibratoire d'une vibration périodique.</i>	35
Figure 2.13	<i>Spectre d'accélération vibratoire d'une vibration périodique.</i>	36
Figure 2.14	<i>Exemple de limites vibratoires proposées par les normes</i>	38
Figure 3.1	<i>Exemple de limites vibratoires proposées par les normes</i>	45
Figure 3.2	<i>Architecture d'un système embarqué</i>	47
Figure 3.3	<i>Domaines d'application des systèmes embarqués.</i>	48
Figure 3.4	<i>La carte Arduino UNO</i>	49
Figure 3.5	<i>Arduino UNO</i>	45
Figure 3.6	<i>Arduino Méga</i>	46
Figure 3.6	<i>Arduino Nano</i>	46
Figure 3.7	<i>LilyPad Arduino</i>	46
Figure 3.8	<i>Arduino Leonardo</i>	47
Figure 3.9	<i>RedBoard</i>	47
Figure 3.10	<i>Description des entrées/sorties de la carte Arduino Uno</i>	50
Figure 3.11	<i>Description des entrées/sorties de la carte Arduino Uno</i>	51
Figure 3.12	<i>Prosimètre monté sur un palier</i>	52
Figure 3.13	<i>Schéma de principe d'une vélocimétrie</i>	54
Figure 3.14	<i>Schéma de principe d'un accéléromètre.</i>	55
Figure 3.15	<i>Mode de fixation de l'accéléromètre</i>	55
Figure 4.1	<i>Un groupe motopompe électrique</i>	53
Figure 4.2	<i>Découpage de système motopompe.</i>	54
Figure 4.3	<i>Groupe électropompe</i>	65
Figure 4.4	<i>Les différents cas d'utilisation de ce groupe.</i>	69
Figure 4.5	<i>Fonctionnalités de groupe de gestion des organes.</i>	70
Figure 4.6	<i>Diagramme des classes UML de GMAO</i>	70

LISTE DES FIGURES

Numéro	Titre	Page
Figure 4.7	<i>L'interface principale de GMAO</i>	70
Figure 4.8	<i>L'interface principale de GMAO</i>	71
Figure 4.9	<i>L'interface principale de GMAO</i>	71
Figure 4.10	<i>Analyser les vibratoires</i>	72
Figure 4.11	<i>Le gyro-accéléromètre MPU6050</i>	72
Figure 4.12	<i>Caractéristique technique du gyro-accéléromètre MPU 6050</i>	73
Figure 4.13	<i>L'IDE Arduino.</i>	73
Figure 4.14	<i>Schéma de Circuit des connexions de la réalisation.</i>	74
Figure 4.15	<i>Dossier " bibliothèques " de Arduino</i>	74
Figure 4.16	<i>Les bibliothèques I2C et MPU6050 sous l'IDE Arduino.</i>	75
Figure 4.17	<i>La fenêtre de l'IDE Arduino pour rétablir le code de configuration.</i>	76
Figure 4.18	<i>L'inclusion des bibliothèques « I2Cdev » et « MPU6050 ».</i>	77
Figure 4.19	<i>La compilation du sketch.</i>	77
Figure 4.20	<i>Valeurs retournées par le module MPU6050.</i>	78
Figure 4.21	<i>Résultat de la calibration.</i>	78
Figure 4.23	<i>le test de fonctionnement de l'MPU6050.</i>	79
Figure 4.23	<i>Les résultats de calibration remplacée dans le code.</i>	78
Figure 4.24	<i>La stabilité des mesures après calibration.</i>	80
Figure 4.25	<i>Des alertes température élevée</i>	81
Figure 4.26	<i>Les résultats de calibration remplacée dans le code.</i>	82
Figure 4.27	<i>Programme de mesure de température et d'humidité</i>	82
Figure 4.28	<i>Des alertes température élevée</i>	84
Figure 4.29	<i>L'interface principale de GMAO</i>	85
Figure 4.30	<i>Programme de mesure de température et d'humidité</i>	85
Figure 4.31	<i>Programme de mesure de température et d'humidité</i>	86
Figure 4.32	<i>Résultat de température et l'humidité</i>	87

Liste des tableaux

Numéro	Titre	Page
Tableau3.1	<i>Les défauts et leurs fréquences.</i>	37
Tableau3.2	<i>Choix d'une platine Arduino en fonction des objectifs recherchés</i>	47
Tableau4.1	<i>Les éléments et leurs fonctions et les techniques</i>	56
Tableau4.2	<i>Les éléments et leurs fonctions et les techniques</i>	57
Tableau4.3	<i>Les éléments et leurs fonctions et les techniques</i>	57
Tableau4.4	<i>Broche et connexion du notre circuit électronique.</i>	65

**INTRODUCTION
GÉNÉRALE**

INTRODUCTION GENERALE

Les entreprises actuellement donnent l'importance des coûts induits par les problèmes et les pannes des systèmes de production. Pour cela la maintenance joue un rôle très important dans n'importe quelle structure organisationnelle d'une entreprise industrielle, l'entreprise elle met des dépenses importantes pour la maintenance, elle met des compétences humaines spécialisées dans la maintenance, des stratégies et des plans de maintenance, des systèmes de management d'informations et des connaissances, mettre en œuvre l'avancement technologique pour la maintenance.

La maintenance préventive est basée sur une surveillance du fonctionnement du bien selon un calendrier, ou à la demande, par contre la maintenance conditionnelle est basée sur l'analyse vibratoire pour objectif la détection des défauts. Alors que les deux types de maintenances préventives (systématique et conditionnelle) pour but d'éviter l'intervention de la maintenance corrective c'est-à-dire les problèmes d'arrêts des systèmes et les pannes. Notre travail de master s'inscrit dans ce contexte pour objectif d'exploiter les technologies de l'informatique et la communication tel que les systèmes d'informations et les systèmes électroniques embarqués afin d'utiliser pour la collecte des informations pertinentes de la maintenance préventive systématique et conditionnelle pour améliorer un système de GMAO. Pour réaliser ce travail nous allons faire une étude sur un système de maintenance d'un groupe motopompe électrique au niveau de l'entreprise Algériennes Des Eaux (ADE).

Afin d'atteindre les objectifs que nous fixés, ce mémoire est structuré en quatre chapitres :

Le premier chapitre est consacré à la gestion de maintenance assistée par ordinateur GMAO, nous faisons une étude sur les différents types de maintenance (Maintenance préventive, maintenance conditionnelle, maintenance corrective...etc.) puis nous présentons les caractéristiques d'un système informatique de management de la maintenance.

Dans le deuxième chapitre nous présentons une recherche sur la vibration mécanique car l'analyse vibratoire est un moyen utilisé pour suivre l'état des machines. Nous commençons ce chapitre par l'explication des concepts de base de la vibration puis nous faisons une étude sur les différents types des signaux vibratoires, les méthodes de l'analyse vibratoire, les critères pour trouver et choisir les seuils de jugement.

Dans le troisième chapitre nous présentons les systèmes embarqués, nous présentons premièrement les composants d'un système embarqué, puis nous présentons les cartes Arduino, car la carte Arduino c'est le cœur d'un système embarqué et enfin nous faisons une étude sur les capteurs de vibration et de température.

Le quatrième chapitre est consacré à la modélisation et la réalisation de notre système de GMAO améliorée, le système de GMAO amélioré se base sur le développement d'une application informatique en utilisant le SGBD Microsoft Access et la programmation de l'Arduino avec les capteurs des vibrations et de température.

Chapitre 01

**GESTION DE LA MAINTENANCE
ASSISTEE PAR ORDINATEUR**

1.1 Introduction

Pour pouvoir informatiser les activités de maintenance, on a besoin un système informatique de gestion de la maintenance sous la forme d'un logiciel ou d'un progiciel.

Le logiciel ou le progiciel choisi doit être organisé autour d'une base de données qui permettra à partir de terminaux déployés dans les bureaux techniques, les ateliers, les magasins et les bureaux d'approvisionnement, de programmer et de suivre toutes les activités d'un service de maintenance et les objets de cette activité (services, lignes d'atelier, machines, équipements, sous-ensembles, pièces, etc.) sous les trois aspects techniques, budgétaire et organisationnel. Dans ce chapitre nous présentons les différents types de maintenance et leur objectif puis nous présentons les caractéristiques d'un système informatique de management de la maintenance.

1.2 Définition la maintenance

D'après la norme AFNOR X 60-010, la maintenance est définie comme étant « l'ensemble des actions permettant de maintenir ou de rétablir un bien dans un état spécifié ou en mesure d'assurer un service déterminé ». En effet, maintenir, c'est donc effectuer des opérations (dépannage, graissage, réparation, amélioration, vérification, ...) qui permettent de conserver le potentiel du matériel pour assurer la continuité et la qualité de la production ainsi que la sécurité d'opération. [1]

1.2.1 Objectifs de la maintenance

Les objectifs que la maintenance réalise à travers son organisation, sa gestion et ses interventions, sont très nombreux. Ils peuvent toutefois être groupés en sept axes : [2]

- La disponibilité,
- L'économie,
- La qualité,
- La durabilité,
- La sécurité,
- La productivité,
- La protection de l'environnement.

1.2.2 Rôle de la maintenance

La maintenance doit assurer la rentabilité des investissements matériels de l'entreprise, en maintenant le potentiel d'activités en tenant compte de la politique de maintenance définis par l'entreprise. [2]

1.2.2.1 Court terme

Objectifs essentiels :

- Réduire les durées d'immobilisations.

- Réduire les coûts des interventions.

1.2.2.2 . Moyen terme

Elaboration d'un plan de prévisions des interventions à l'immobilisation des équipements.

1.2.2.3 Long terme

Réalisation d'opérations directement liées à la politique générale de l'entreprise et fin de minimiser l'ordonnancement :

- Des charges.
- Des stocks.
- Des investissements.

1.2.3 Types de maintenance

D'après la définition, on a plusieurs types de maintenance. [3]

1.2.3.1 Maintenance préventive

Et / ou des critères prédéterminés significatifs de l'état de dégradation du bien ou de service. Cette politique de maintenance s'adresse aux machines provoquant une perte de production ou des coûts d'arrêts imprévisibles classés comme important pour l'entreprise. Telle est le cas des machines des chantiers de forages. Il convient donc d'organiser un système de maintenance visant à minimiser ces arrêts souvent trop onéreux. Ainsi on aura à pratiquer trois formes de maintenance dite préventive :

1.2.3.2 Maintenance Systématique

Maintenance préventive effectuée selon un échancier établi à partir d'un nombre prédéterminé d'unités d'usage. La mise en pratique de cette maintenance nécessite de décomposer les machines en éléments maintenables. Ces éléments doivent être visités ou changés régulièrement.

La périodicité de ces visites s'établit par l'étude des lois de durée de vie. On harmonisera ces périodicités de façon à les rentres multiple les unes des autres.

Des gammes d'entretien seront élaborées de façon à préciser le travail à exécuter par l'équipe de maintenance, un rapport sera rédigé mettant en relief les résultats des diverses mesures et les observations. L'intérêt de cette méthode est de diminuer les risques de défaillance.

1.2.3.2.1 Les opérations de maintenance préventive systématique peuvent être

- Le remplacement,
- De l'huile des boites de vitesse,

- Des pièces d'usure normale,
- Le réglage et l'étalonnage,
- Des niveaux de pressions hydrauliques et pneumatiques,
- Des tensions de courroies,
- Des jeux glissières ou des cales d'ajustement,
- Le contrôle de l'état général,
- Des niveaux d'huile,
- Des divers blocages,
- Apparence d'usure ou de fissure.

1.2.3.3 Maintenance conditionnelle

Maintenance préventive subordonnée au franchissement d'un seuil prédéterminé significatif de l'état de dégradation du bien, Le franchissement du seuil peut être mis en évidence par l'information donnée par un capteur ou par tout autre moyen.

1.2.3.4 Maintenance prévisionnelle

Maintenance préventive subordonnée à l'analyse de l'évolution surveillée des paramètres significatifs de la dégradation du bien, permettant de retarder et de planifier les interventions.

1.2.3.5 Maintenance corrective

Ensemble des activités réalisées après la défaillance d'un bien, ou la dégradation de son fonctionnement, pour lui permettre d'accomplir une fonction requise, au moins provisoirement,

La maintenance corrective comprend en particulier :

- La localisation de la défaillance et son diagnostic,
- La remise en état avec ou sans modification,
- Le contrôle du bon fonctionnement.

1.2.3.6 Maintenance palliative

Activités de maintenance corrective destinées pour permettre à un bien d'accomplir provisoirement tout ou une partie d'une fonction requise.

Appelée couramment "dépannage", la maintenance palliative est principalement constituée d'actions à caractère provisoire qui devront être suivies d'actions curatives.

1.2.3.7 Maintenance curative

Activités de maintenance corrective ayant pour objet de rétablir un bien dans un état spécifié ou de lui permettre d'accomplir une fonction requise.

Le résultat des activités réalisées doit présenter un permanent, ces activités peuvent être :

- Des modifications ou améliorations ayant pour objet de supprimer la (ou les) défaillance (s).

1.2.3.8 Maintenance a améliorative

Ce type de maintenance permet, après réflexion et étude, d'éliminer le problème. Elle nécessite obligatoirement une concertation entre services production-bureau d'étude et maintenance.

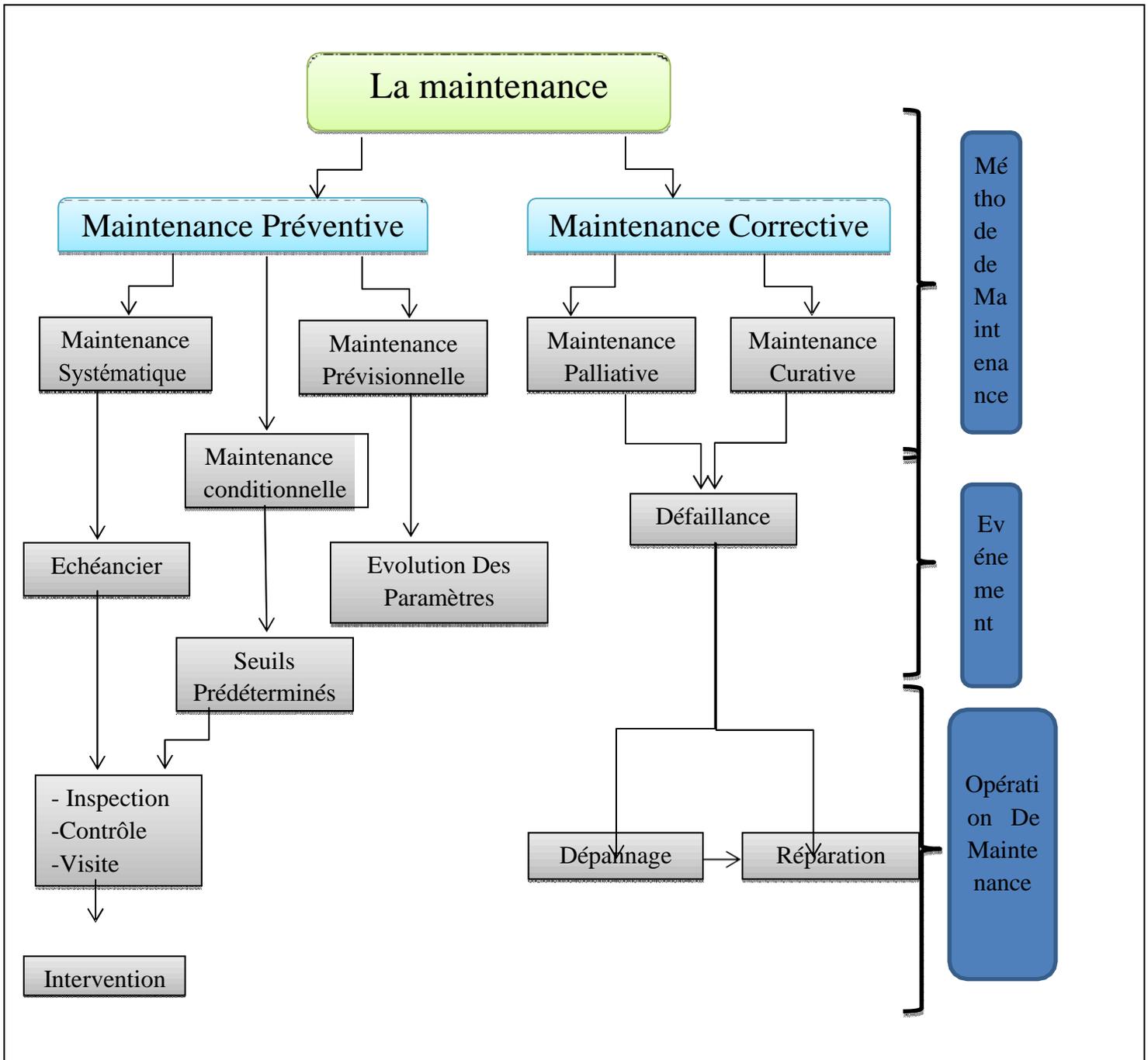


Figure I.1. Types de maintenance [3]

1.3 Gestion de maintenance assistée par ordinateur

La GMAO signifie la gestion de maintenance assistée par ordinateur il s'agit d'un logiciel spécialisé pour réaliser la gestion d'un service technique. La GMAO est constituée d'une base de données « historique » qui est alimentée par le personnel de maintenance via un formulaire. Chaque GMAO est personnalisée selon les besoins spécifiques d'exploitation de l'historique ou le fonctionnement d'un site.

1.3.1 Caractéristiques générales

Un système informatique de management de la maintenance est un progiciel organisé autour d'une base de données permettant de programmer et de suivre sous les trois aspects techniques, budgétaires et organisationnel, toutes les activités d'un service de maintenance et les objets de cette activité (services, lignes d'atelier machines ,équipements , sous-ensembles , pièces , etc.) à partir de terminaux répandus dans les bureaux techniques, les ateliers ,les magasins et bureaux d'approvisionnement.

Le logiciel de GMAO permet de construire une base de données dans laquelle on retrouvera [4]

- Les articles du magasin, les fournisseurs,
- La gestion des demandes d'interventions. Des entrées et sorties des articles,
- La gestion des achats,
- La gestion des actifs (équipements et sous-ensembles),
- La gestion des interventions correctives, préventives,
- Les analyses financières et le suivi des indicateurs de maintenance,
- La gestion des contacts clients et la facturation.

1.3.2 Objectifs et buts d'un système GMAO [5]

1.3.2.1 Objectifs :

Un système de GMAO doit ;

- Permettre la réduction des coûts du service maintenance
- Aider à faire augmenter la fiabilité des équipements avec une disponibilité maximale.
- Fournir à tout acteur du service maintenance des informations plus détaillées et plus fiables.

1.3.2.2 Buts

Pour atteindre ces objectifs, les buts visés par un système de GMAO sont:

- Améliorer le contrôle des coûts

- Optimiser le budget maintenance
- Optimiser la gestion des achats et des stocks (réduction des coûts)
- Augmenter la disponibilité des équipements (diminution du manque à produire)
- Améliorer et faciliter la planification de la maintenance
- Capitaliser sur l'expérience (consultation facile et rapide de l'historique)
- Diminuer le nombre, la fréquence et la gravité des pannes
- Améliorer la qualité de service (répondre aux besoins et augmenter le taux de satisfaction)
- Augmenter la productivité de la maintenance
- Réduire au maximum les interventions urgentes
- Diminuer les délais d'intervention

1.3.3 Rôle d'un système de GMAO[4]

- Un système de GMAO a pour rôle de participer à une meilleure organisation, à un gain de temps et d'efficacité, et à l'allègement des tâches.
- Un système de GMAO est conçu pour gérer, piloter, suivre et analyser l'activité, la prise en charge de la planification par exemple, est difficilement réalisable car une partie des travaux maintenance n'est même pas planifiable.

1.3.4 Structure des logiciels GMAO

- Tous les logiciels GMAO ont en commun la même structure modulaire proposant les mêmes fonctions.
- Logiciel GMAO → différents modules



Figure 1.2 : Structure des logiciels GMAO [4]

Gestion du suivi opérationnel des équipements,

- Gestion des interventions en interne et en externe,
- Gestion du préventif,
- Gestion des stocks,
- Gestion des approvisionnements et des achats,
- Analyse des défaillances,
- Gestion du budget et suivi des dépenses,
- Gestion des ressources humaines,
- Tableaux de bord et statistiques,
- Autres modules et interfaçages possibles.

Domaines à gérer en maintenance :

- Les activités du service de maintenance,
- Le matériel,
- Les stocks de rechanges et l'approvisionnement,
- La gestion budgétaire : ventilation des coûts par matériel, service, type de maintenance.
- Les investissements,
- Les ressources humaines.

1.3.5 Logiciel pour GMAO :[5]

Selon les systèmes de GMAO, les fonctions remplies sont dénommées, réparties et organisées de diverses manières. Mais, tous les progiciels de GMAO ont en commun la même structure modulaire proposant les mêmes fonctions. Dans notre cas, il s'agit du logiciel **OptiMaint** édité par l'entreprise Française **Apisoft international**

- **OptiMaint** est une solution globale de gestion et d'organisation de la fonction maintenance adaptée aux différents secteurs de l'industrie, du tertiaire, des services, des collectivités etc ...
- **OptiMaint** a été conçu avec des responsables de maintenance pour répondre aux besoins concrets d'une petite, moyenne ou grande structure.
- **OptiMaint** peut s'inscrire dans une logique de maintenance multi sites.
- **OptiMaint** est l'alternative entre une solution de GMAO complexe, coûteuse et longue à mettre

en place et une solution avec des fonctionnalités trop simples et limitées.

- **OptiMaint** est le fruit de plus de 10 ans d'expérience de développement en solution de GMAO.
- **OptiMaint** couvre les fonctionnalités suivantes:
 - **Gestion du patrimoine** : Description détaillée des équipements à maintenir.
 - **Gestion des interventions** : Optimisation de la gestion des interventions préventives et corrective avec un historique.
 - **Gestion des achats** : Optimisation de la gestion des achats depuis la commande jusqu'à la réception.
 - **Gestion des stocks** : Optimisation de la gestion des stocks en évitant les ruptures de stocks ou les surstocks.
 - **Gestion des budgets** : Permettre le suivi du budget de fonctionnement.
 - **Gestion des projets** : Permettre le suivi des budgets alloués à des projets d'investissement.
 - **Analyse** : Tableau de bord et suivi statistique des équipements.

1.3.5.1.1 FONCTIONNALITÉS DU LOGICIEL GMAO :

1.3.5.1.1.1 Gestion du patrimoine

La gestion du patrimoine, concerne essentiellement le suivi des équipements à travers les taches suivantes;

- Codification des équipements, des organes et des articles.
- Arborescence des équipements avec les organes et les articles.
- Visualisation graphique de l'arborescence.
- Calendrier individuel des équipements avec gestion des exceptions (révisions, arrêts exceptionnels ...).
- Définition de famille et de sous famille.
- Commentaires libres.
- Fonction de recopie partielle ou complète d'un équipement.
- Création de ses propres champs saisie sur les fiches (format alphanumérique, numérique, monétaire, date, oui/non, image).
- Définition de liste déroulante avec choix paramétrables.
- Fonction de recherche multi critères.

1.3.5.1.1.2 Gestion des interventions :

1°. Les intervenants :

- Définition des rôles / qualifications des intervenants.
- Taux horaire par intervenant.
- Paramétrage de l’Affichage/impression des temps et des coûts fonction des profils utilisateurs.
- Création de ses propres champs saisie sur les fiches (format alphanumérique, numérique, monétaire, date, oui/non, image).
- Définition de liste déroulante avec choix paramétrables.
- Fonction de recherche multi critères

2. Les demandes d’intervention (DI) :

- Impression et/ou envoi par email automatique des **DI**.
- Suivi par statut des **DI** (acceptée –convertie en bon de travail –engagée –clôturée –refusée).
- Conversion des **DI** en bon de travail **BT**.
- Gestion des priorités des **DI**.
- Date souhaitée / Date début de panne.
- Fonction de recherche multi critères.

1.3.5.1.1.3 La maintenance préventive :

- Création et édition automatique ou manuel des bons de travaux.
- Gamme opératoire (description libre de l’intervention).
- Association de fichiers joints à une maintenance préventive.
- Association de document modèle.
- Qualification rattachée à une gamme opératoire.
- Temps et coût prévu.
- Article(s) nécessaire(s).
- Fusion de plusieurs bons de travaux.
- Planning visuel du préventif.
- Planification calendaire.
- Planification selon des compteurs (nombre illimité de compteurs pour un même équipement).
- Fonction de recopie d’une maintenance préventive sur un ou plusieurs équipements.

1.3.6 Prise en main de OPTIMAINt v 6.3.4 :[5]

- La prise en main du logiciel OptiMaint (**Figure 1.2**) passe essentiellement par la découverte de son **IGU** (Interface Graphique Utilisateur) ou bien en anglais **GUI** (pour **G**raphical **U**ser **I**nterface).
- Pour la découverte et la prise en main du logiciel OptiMaint, deux composantes de l'IGU sont à considérer:
 - La barre des menus standards
 - La barre d'outils

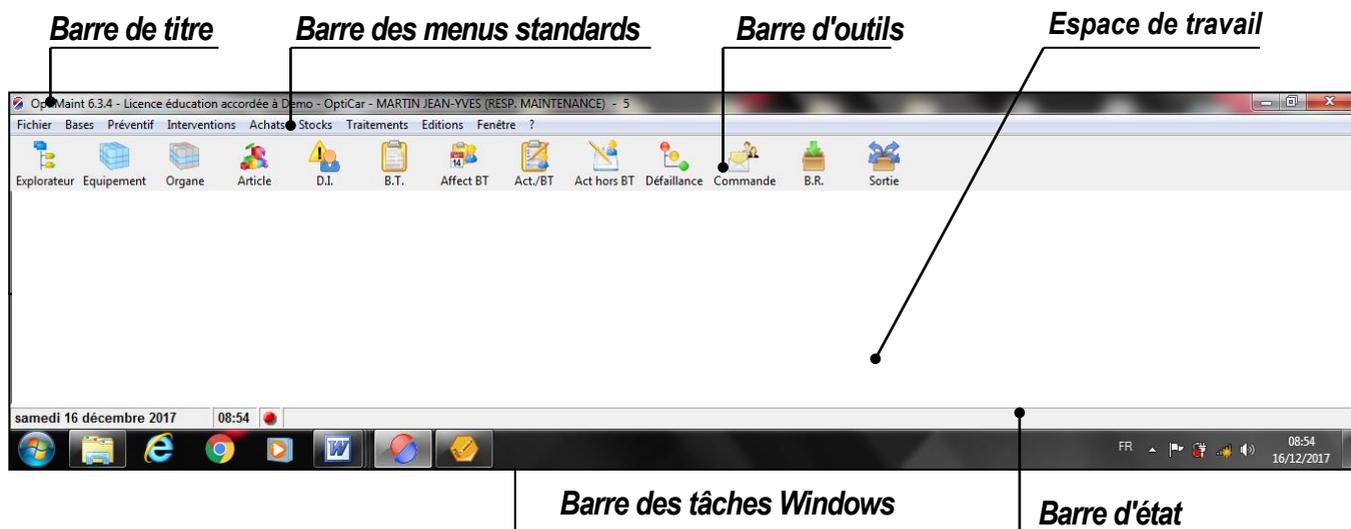


Fig 12.: Aperçu de l'interface graphique utilisateur du logiciel OptiMaint

1.3.6.1 Barre des menus standards :

La barre des menus standards, est une composante de IGU et contient les menus suivants :



Fig 12: Barre principale des menus standards du logiciel OptiMaint

Pour faciliter la découverte du logiciel **OptiMaint**, les menus standards, sont regrouper comme suit :

1. Menu [**Fichier**] → Accès au paramétrage, à la sécurité et la confidentialité d'OptiMaint.
2. Menus [**Bases**],[**Préventif**] → Accès à toutes les définitions des données.
3. Menus [**Interventions**],[**Achat**],[**Stocks**] → Accès à tous les mouvements.
4. Menu [**Traitements**] → Accès à tous les traitements.
5. Menu [**Editions**] → Accès aux éditions.

6. Menus [Fenêtre],[?] |→ Accès aux fenêtres déjà ouvertes ou spécifiques et à l'aide.

1.3.6.2 Barre d'outils :

- Le logiciel **OptiMaint** intègre une barre d'outils composée d'un certain nombre d'icônes associées

à des boutons de commandes

Bouton de commande **DI**

Icône du bouton de commande

Demande d'intervention

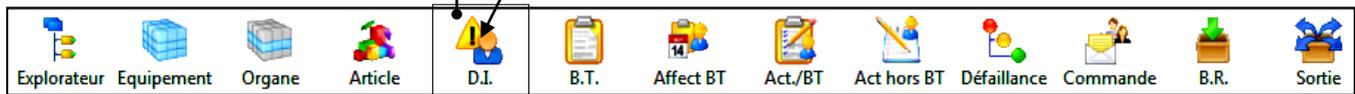


Fig 1.3: Barre d'outils du logiciel OptiMaint

17624

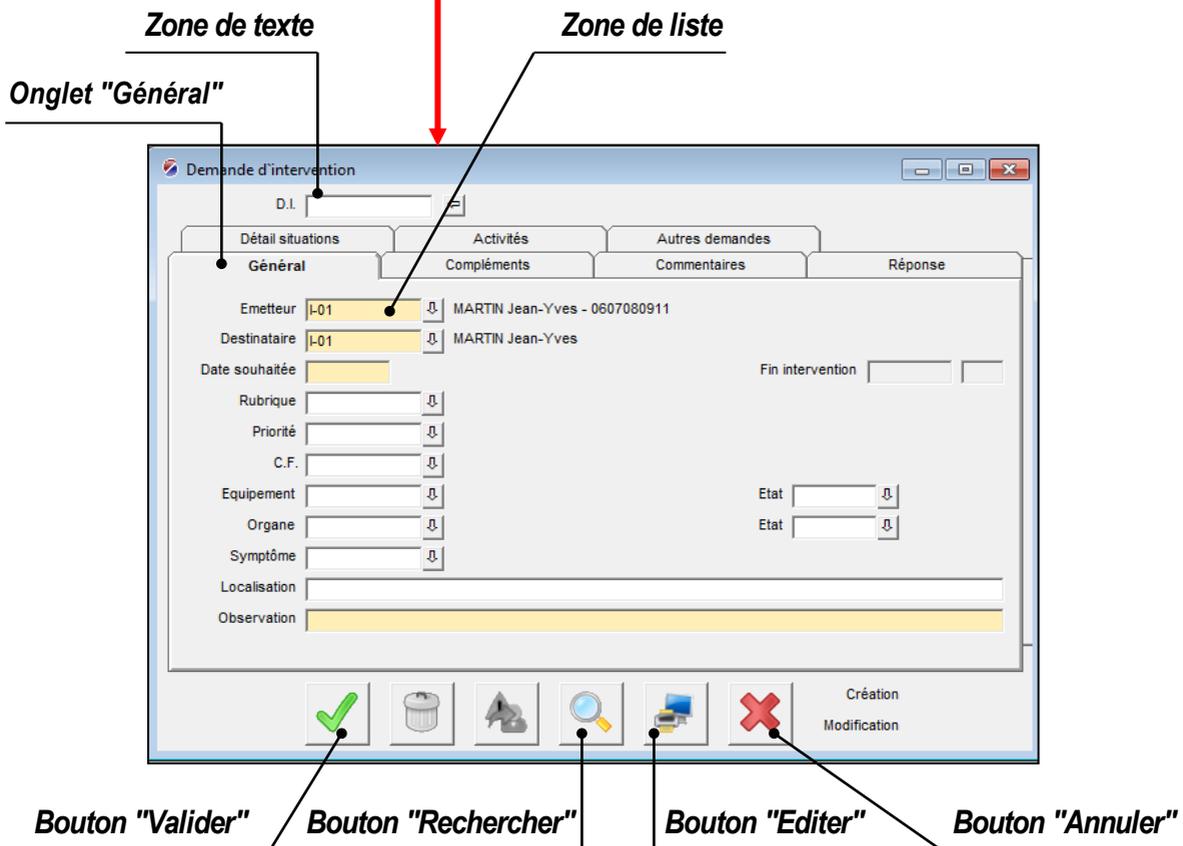
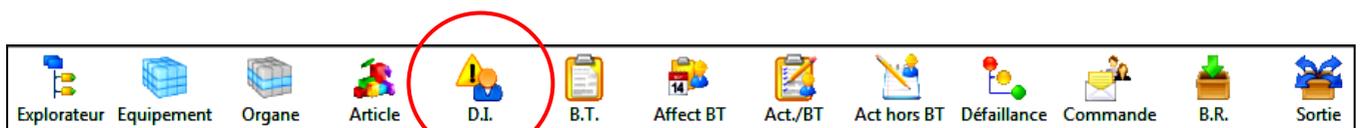


Fig 1.4: Boite de dialogue demande

1.4 Conclusion :

Dans ce chapitre nous avons connu ce que c'est un logiciel GMAO, ces différents modules et de les relier aux concepts dans les outils actuels de gestion de la maintenance.

Enfin, la solution GMAO est un excellent outil pour améliorer les processus, gagner en productivité, réduire les coûts et éviter des incidents de qualité. Le chapitre suivant sera consacré aux vibrations mécaniques et l'analyse vibratoire.

Chapitre 02

L'ANALYSE VIBRATOIRE

2 Introduction

L'analyse vibratoire est un moyen utilisé pour suivre l'état des machines tournantes En fonctionnement.

A partir des vibrations régulièrement recueillies sur une machine tournante, l'analyse vibratoire consiste à détecter d'éventuels les dysfonctionnements et à suivre leur évolution dans le but de planifier ou reporter une intervention mécanique.

2.1 Définition d'une vibration

La norme ISO 2041 « Vibrations et chocs - Vocabulaire (Août 1990) » définit la notion de vibration : Variation avec le temps de l'intensité d'une grandeur caractéristique du mouvement ou de la position d'un système mécanique, lorsque l'intensité est alternativement plus grande et plus petite qu'une certaine valeur moyenne ou de référence. En fait, un corps est dit en vibration lorsqu'il est animé d'un mouvement oscillatoire autour d'une position d'équilibre ou de référence. Les textes de normalisation AFNOR relatifs aux vibrations sont : NF E 90-001, NF E 90 002.[6]

2.1.1 Signal vibratoire

Une vibration peut se représenter sous diverses formes.

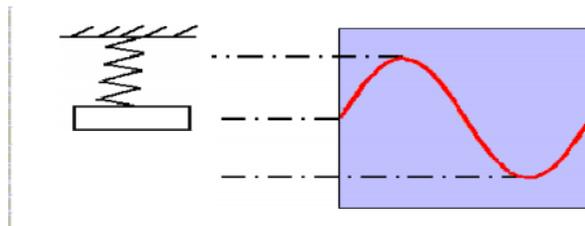


Figure 2.1 : Signal vibratoire

Si l'on observe un système mécanique simple constitué d'une masselotte suspendue à un ressort, on constate que le mouvement de la masselotte se traduit par un déplacement, une vitesse et une accélération.

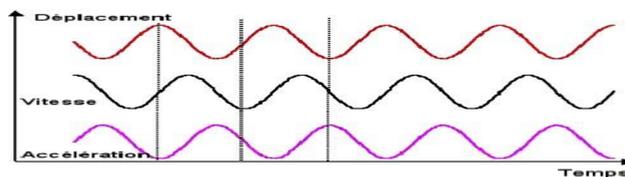


Figure 2.2 : Signal vibratoire

On remarque sur la figure ci-dessus un décalage entre les maximas de chaque sinusoïde. Ce décalage est appelé la phase du signal.

La représentation en **accélération** accentue les hautes fréquences La représentation en

déplacement accentue les basses fréquences

Il faut également noter qu'il existe une relation mathématique entre déplacement, vitesse et accélération.

$$x(t) = A \sin(2\pi \cdot f t)$$

$$v(t) = \frac{dx}{dt} = 2\pi \cdot f A \cos(2\pi \cdot f t)$$

$$a(t) = \frac{d^2x}{dt^2} = -(2\pi \cdot f)^2 A \sin(2\pi \cdot f t)$$

2.1.2 Type des signaux [6]

2.1.2.1 Déterministe

Des signaux (périodique ou non périodique) dont l'évolution en fonction du temps peut être parfaitement décrite par un modèle mathématique. Ces signaux proviennent de phénomènes pour lesquels on connaît les lois physiques correspondantes et les conditions initiales, permettant ainsi de prévoir le résultat. Les signaux non périodiques se composent d'une part des signaux pseudopériodiques formés d'une somme de sinusoides de périodes différentes et d'autre part des signaux transitoires dont l'existence est limitée dans le temps. Ces signaux "certains" peuvent en principe être reproduits rigoureusement identiques à eux-mêmes.

2.1.2.2 Aléatoire

Le signal aléatoire fait intervenir deux notions :

- La notion de signal c'est à dire une mesure qui dépend d'un ou plusieurs paramètres d'espace, le plus souvent le temps.
- La notion d'aléatoire qui fait que, à un instant t donné, la mesure n'est pas certaine mais dépend du hasard. Cet aspect est appelé l'aspect stochastique et on parle indifféremment de signal aléatoire ou de signal stochastique.

Ce sont les signaux qui ont caractère non reproductible et imprévisible.

[Caen]

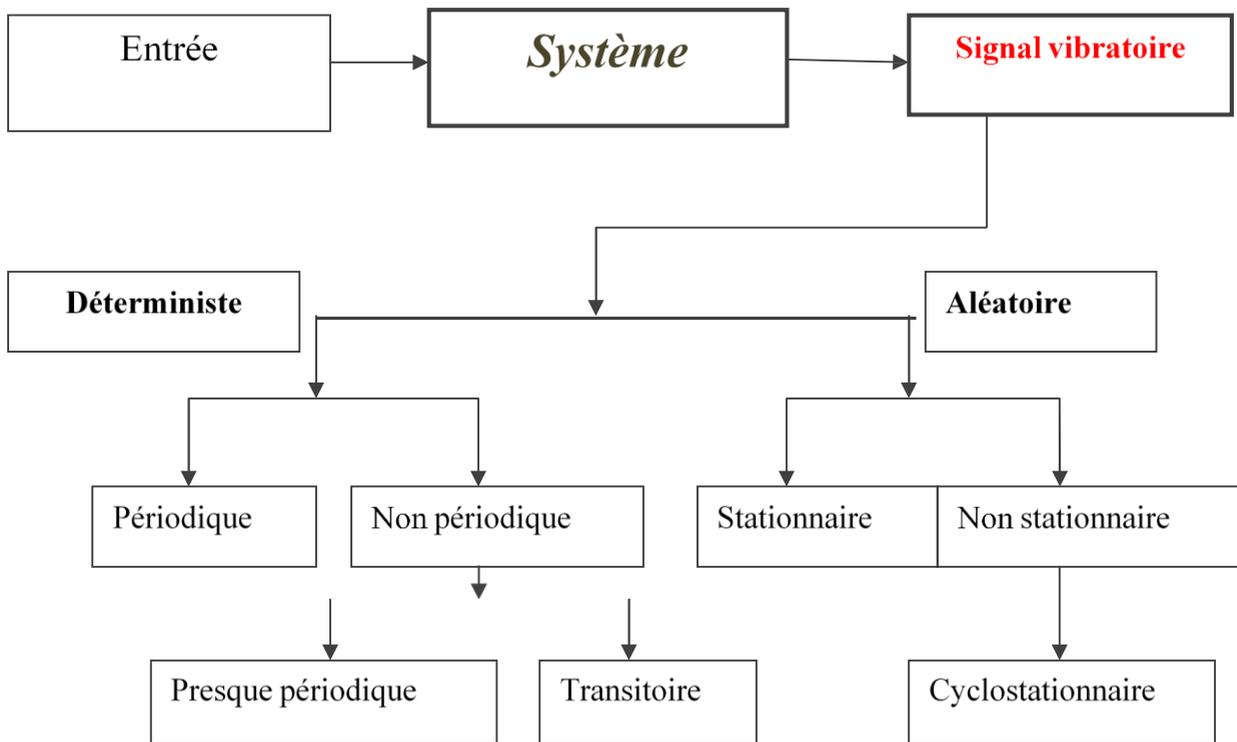


Figure I.4 : les différents types des signaux vibratoires

2.1.3 Défauts des machines tournantes

2.1.3.1 Balourd

Le balourd (figure 2.4) est la cause de vibrations la plus commune et la plus fréquemment rencontrée. En dépit du soin apporté à la construction des rotors, il est donc impossible de les équilibrer parfaitement et il existe donc toujours une **vibration à la fréquence de rotation** dont l'amplitude est directement proportionnelle à l'importance du balourd et au carré de vitesse de rotation. Une modification brusque de l'amplitude correspond systématiquement à une modification du balourd dont l'origine peut-être une rupture ou la déformation d'une partie du rotor (ailettes de turbine par exemple). Si cette modification d'amplitude est plus lente, il peut s'agir d'une usure ou d'un encrassement de la partie tournante (suie, dépôt, etc.). Le phénomène se caractérise par une vibration radiale importante à la fréquence de rotation.[6]

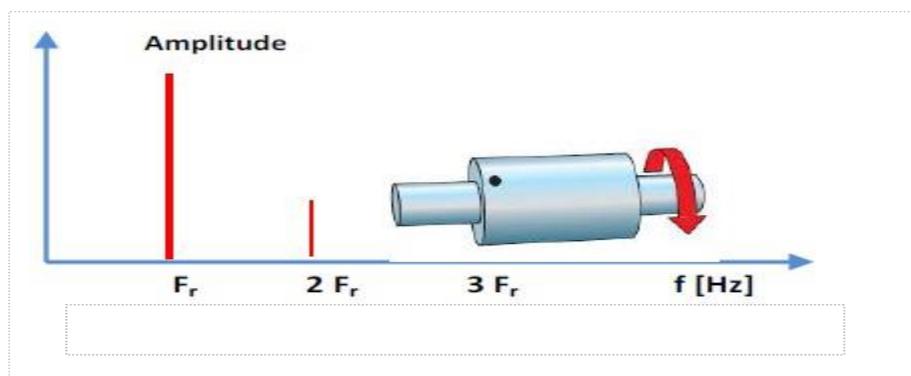


Figure 2.4 Défaut de balourd [6]

2.1.3.2 Désalignement :

Un désalignement (Figure 2.5) provoque **des vibrations à la fréquence de rotation** ainsi qu'aux harmoniques d'ordre 2, 3 et parfois 4 (double, triple, quadruple de la fréquence de rotation et parfois même davantage en particulier pour les accouplements à denture où l'on rencontre les harmoniques liés au nombre de dents et à la fréquence de rotation). La composante axiale de la vibration est particulièrement importante pour l'harmonique d'ordre.

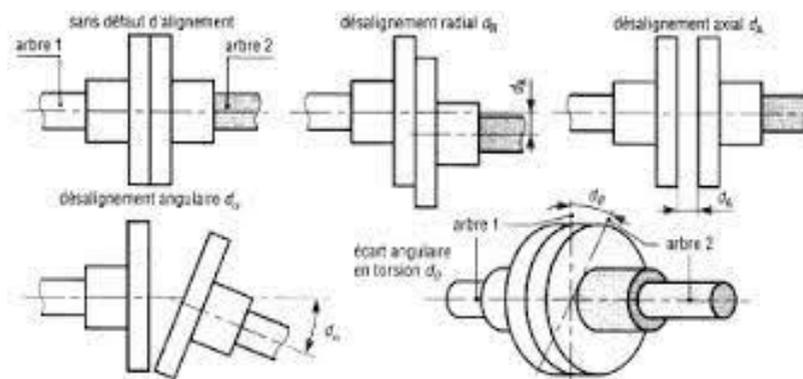


Figure 2.5 : défaut désalignement

2.1.3.3 Mauvais serrage mécanique

Lorsqu'un palier est desserré ou présente une possibilité de mouvement partiel dans le plan radial, il apparaît une vibration radiale à une fréquence égale à deux fois la vitesse de rotation. Cette vibration se produit sous l'effet du balourd initial et elle peut prendre une amplitude élevée en fonction du degré de desserrage du palier.

2.1.3.4 Engrènement

L'engrenage est un dispositif élémentaire constitué de deux organes rigides et dentés R1 et R2 généralement cylindriques ou coniques, appelés roues, possédant N1 et N2 dents qui tournent aux vitesses V1 et V2 autour d'axes fixes.

Un défaut d'engrenage (Figure 2.6) provoque des vibrations à la fréquence est égale à la vitesse de rotation du pignon multipliée par son nombre de dents ($f_{eng} = N1 f1 = N2 f2$).



Figure 2.6 : défaut d'engrenage. [6]

2.1.3.5 Défauts des roulements

L'écaillage d'une piste de roulements (Figure 2.7) provoque des chocs et une résonance du palier qu'il est facile d'identifier avec un appareil à mesure d'ondes de chocs. En analyse spectrale, ce phénomène apparaît aux hautes fréquences par une densité spectrale qui augmente au fur et à mesure que les roulements se détériorent.

Si l'avarie du roulement ne comportait qu'un seul point, on pourrait voir apparaître un pic à une fréquence liée à la vitesse du rotor et aux dimensions du roulement (diamètre de pistes intérieures et extérieures, nombre de billes).

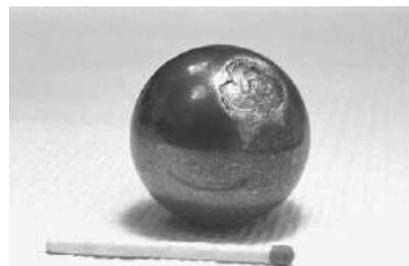


Figure 2.7 : défaut des roulements [6]

2.2 Méthode de l'analyse vibratoire

Toutes machines possédant des composantes en mouvement génèrent un certain niveau de vibration quel que soit le soin apporté à leur fabrication. Le niveau de ces vibrations est le paramètre le plus significatif de l'état d'une machine tournante. Quand ce niveau change, la première manifestation physique d'une anomalie apparaîtra, causent une potentielle de dégradation et de panne. Aujourd'hui, l'emploi des technologies intégrant les méthodes avancées de traitement de signal occupe une place privilégiée pour faire un diagnostic d'un équipement mécanique. Ces caractéristiques font de la surveillance par analyse vibratoire, un outil indispensable pour une maintenance moderne, puisqu'elle permet d'améliorer la disponibilité et la sécurité par détection précoce et le suivi de la dégradation des éléments critiques, ainsi localiser l'organe défectueux sans démontage de la machine. [7]

Deux méthodes sont appliquées pour le suivi vibratoire des machines tournantes :

- Par niveau global.
- Par l'analyse en fréquence.

2.2.1 Méthode de niveau global [7]

La méthode du niveau global, consiste à apprécier le niveau vibratoire de la machine en mesurant la valeur efficace de ces vibrations et de la comparer aux seuils de jugement préconisés par les normes. Selon la norme AFNOR.E90-30, l'intensité vibratoire est une grandeur caractérisant de façon simple et globale, l'état vibratoire d'une machine.

La mesure du niveau global permet de suivre de manière fiable une machine. Elle ne permet pas d'établir un diagnostic. En effet sous une valeur en déplacement, vitesse ou accélération, tous les phénomènes mécaniques sont confondus (Balourd, lignage, engrènement, roulement). Donc la mesure du niveau global permet d'émettre un avis sur la situation du phénomène en fréquence (d'origine basse ou haute fréquence).

2.2.2 Méthode de l'analyse en fréquence [7]

La mesure de la sévérité vibratoire permet de savoir si le comportement vibratoire d'une machine dépasse les limites admissibles. Mais au vu des résultats, il n'est pas possible d'émettre une hypothèse sur les causes et les excitateurs de vibration. Ces informations ne pourront être obtenues qu'en réalisant une analyse en fréquence.

Le diagnostic vibratoire se base sur l'identification du phénomène mécanique, à la fréquence de la vibration qu'il génère. Pour cela il faut donc représenter la vibration dans l'espace des fréquences ; cette opération se réalise mathématiquement en effectuant une transformée de FOURIER du signal temporel vibratoire. Le résultat est appelé SPECTRE fréquentiel. Chaque composant du spectre correspond à une fréquence caractéristique bien définie (Déséquilibre, résonance, engrènement, lignage...). L'analyse en fréquence s'effectue en général lorsque le niveau vibratoire de la machine est jugé supérieur au seuil admissible.

2.2.2.1 Les étapes de mesure

La mesure de vibration est détectée par le capteur, relié à un appareil de mesure portable. Cet appareil qui permet l'instauration d'un suivi très efficace, et permet d'élaborer une banque de données, relié à un logiciel expert permettant l'établissement des routes de relevés et l'exploitation des résultats de mesure :

2.2.2.1.1 Les points de mesurage

La plupart des vibrations de machines sont issues des parties tournantes ou oscillantes. Elles peuvent être d'origine mécanique, électromagnétique, hydraulique, etc. Elles sont transmises à la structure par l'intermédiaire des paliers, et aux fondations par l'intermédiaire des fixations. On conçoit aisément que les meilleurs points de mesurage dans le cadre de la maintenance des machines sont les paliers et qu'il serait déraisonnable de prendre des mesures sur les carters.

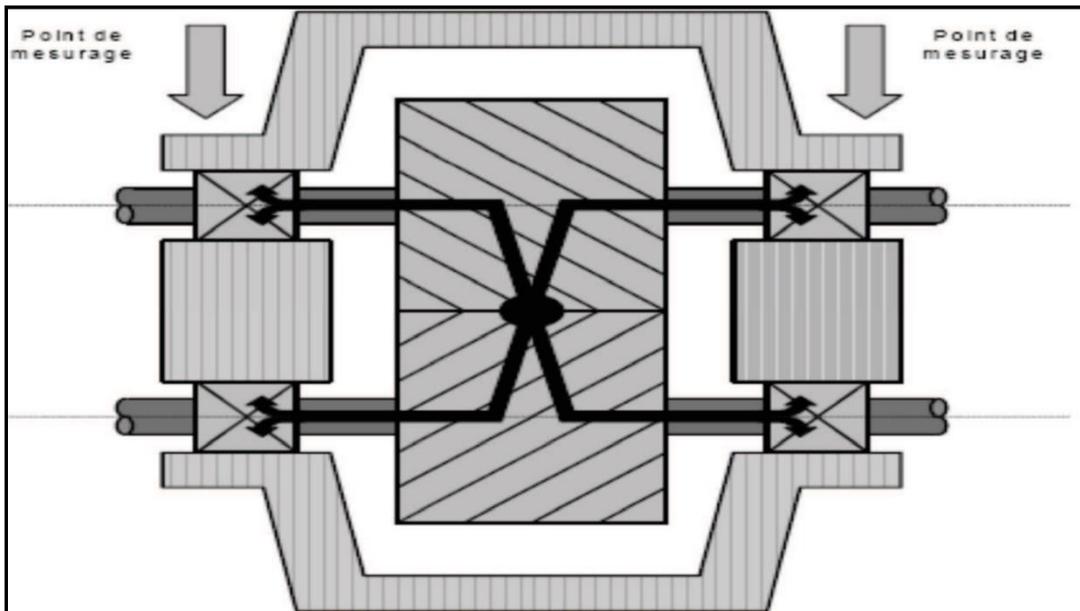


Figure 2.8 : Exemple des points de mesurage lors de l'analyse vibratoire.[7]

2.2.2.1.1.1 Exploitation des résultats

Après la collecte, on peut éditer différents rapports sur les mesures réalisées à l'aide d'un système expert intégré à un micro-ordinateur. Parmi ces rapports on cite :

- 1- Points en déplacement d'alarme avec courbes et spectre associés.
- 2- Tableaux des valeurs.
- 3- Points de mesures.
- 4- Liste des commentaires.

2.2.2.1.1.2 Représentation spectrale [7]

Les spectres mémorisés sont représentés sur un écran ou imprimés avec la liste des pics pour un diagnostic rapide ou une éventuelle comparaison avec le spectre de référence de la machine, ou des spectres déjà établis.

A- Cas d'une vibration harmonique :

Le mouvement vibratoire le plus simple est sinusoïdal (harmonique). Comme il a été vu précédemment, ce phénomène est caractérisé par sa période T , mesurée en secondes. La fréquence f est telle que $f = 1/T$, et la pulsation $\omega = 2\pi f$.

Le déplacement vibratoire s'exprime alors [8]

$$S(t) = S_c \sin(2\pi ft) = S_c \sin \omega t$$

Dans le cas d'une vibration sinusoïdale, on vérifie que :

$$S_{\text{eff}} = \frac{S_c}{\sqrt{2}}$$

Ayant mesuré cette valeur, et connaissant la fréquence f , on peut construire un diagramme appelé « spectre d'amplitude » où l'on va porter la valeur efficace en fonction de la fréquence. Pour une vibration sinusoïdale, on obtient un point unique, de coordonnées (f, s) . En pratique, on préfère tracer une raie de hauteur s , située à la fréquence f .

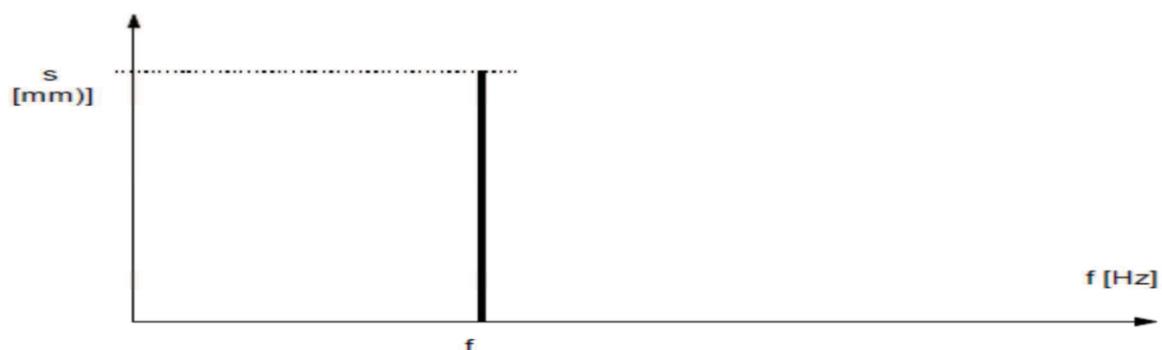


Figure 2.9 : Spectre d'amplitude d'une vibration harmonique.

B- Cas d'une vibration périodique, mais non-harmonique :

Cette vibration est également de période T, mais sa forme n'est pas sinusoïdale.

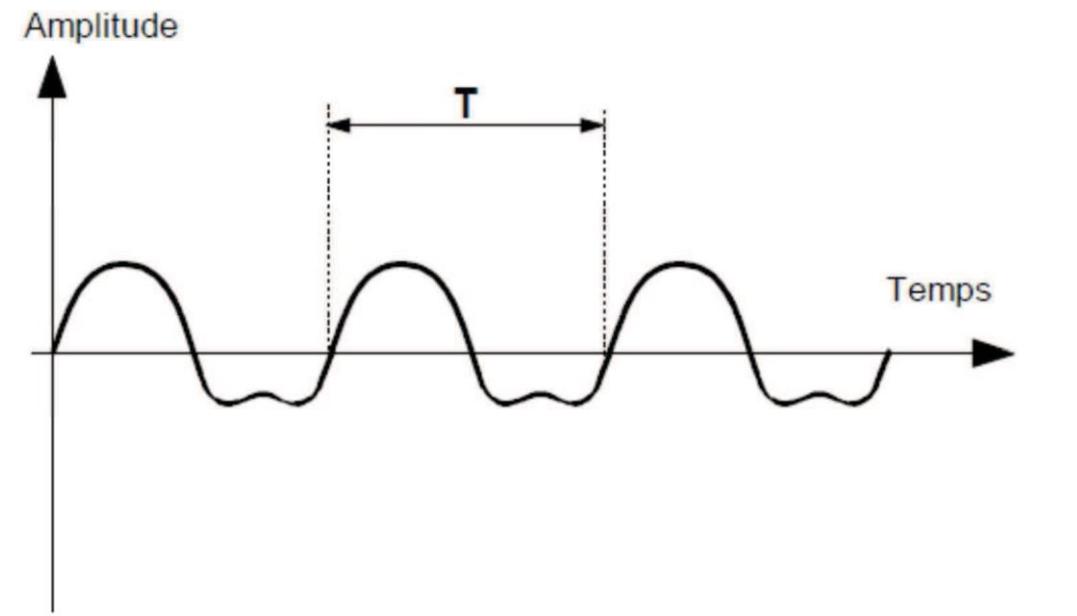


Figure 2.10 : vibration périodique non sinusoïdale.

2.2.2.1.2 Déplacement vibratoire

La valeur efficace du déplacement vibratoire s'exprime alors :

$$S_{\text{eff}} \neq \frac{S_c}{\sqrt{2}}$$

On montre qu'une vibration périodique est une somme de vibrations sinusoïdale de fréquences multiples de f . Son spectre est donc constitué de raies aux fréquences $f, 2f, 3f, \dots$, dont les hauteurs sont égales aux valeurs efficaces de chaque composante sinusoïdale.

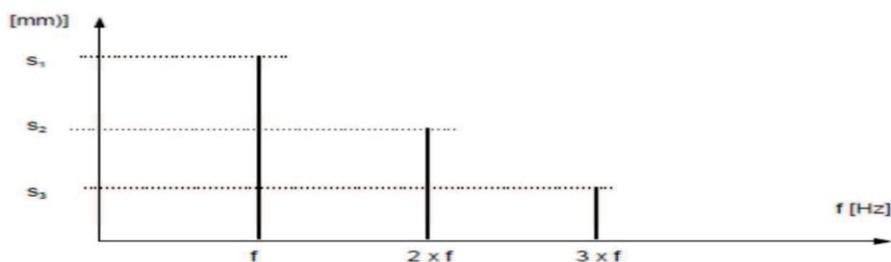


Figure 2.11 : spectre d'amplitude d'une vibration périodique.

2.2.2.1.3 Vitesse vibratoire

Si l'on applique l'équation à ce spectre, on constate que les différentes composantes, exprimées en vitesse vibratoire, ont des amplitudes respectives de :

- $v_1 = s_1 \times 2 \cdot f \cdot \pi$
- $v_2 = s_2 \times 2 \pi(2f)$.

$$- v_n = s_n \times 2\pi (nf).$$

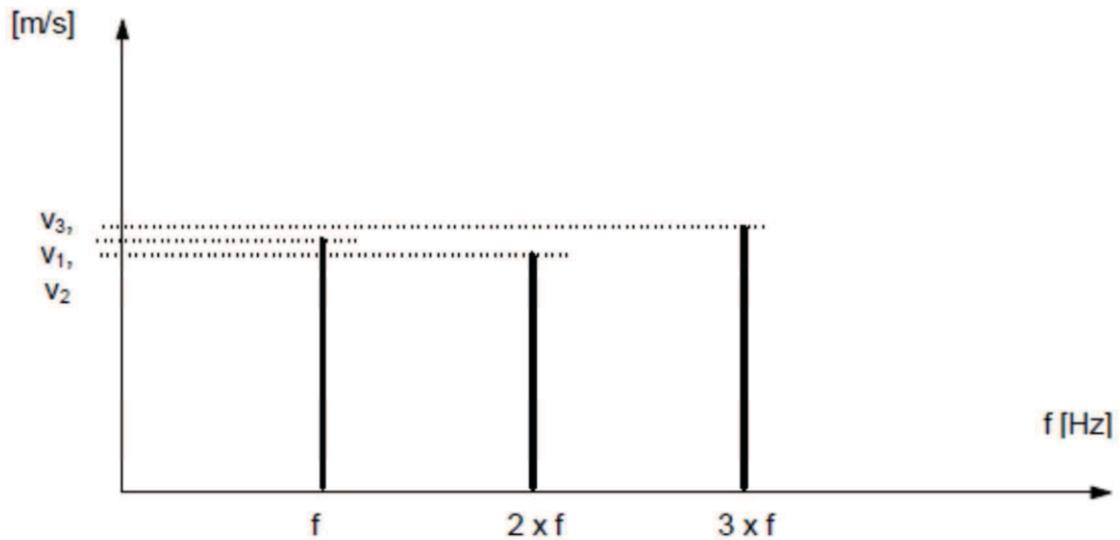


Figure 2.12 : spectre de vitesse vibratoire d'une vibration périodique.

2.2.2.1.4 Accélération vibratoire

Si l'on applique l'équation 12 à ce nouveau spectre, on ne constate que les différentes composantes, exprimées en accélération vibratoire, ont des amplitudes respectives de :

- $a_1 = s_1 \times (2\pi f)^2.$
- $a_2 = s_2 \times (2\pi(2f))^2.$
- $a_n = s_n \times (2\pi(nf))^2.$

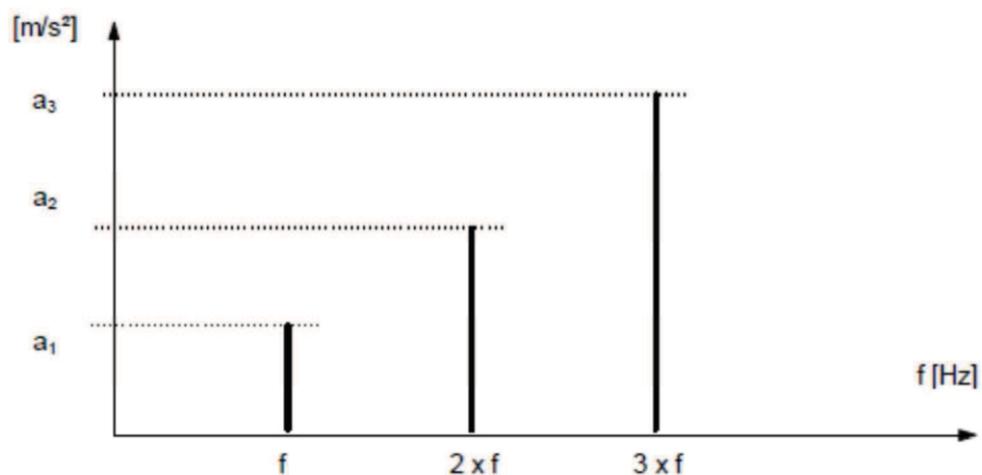


Figure 2.13 : spectre d'accélération vibratoire d'une vibration périodique.

On ne constate que le fait de mesurer une grandeur plutôt qu'une autre n'apporte pas d'information supplémentaire sur l'origine des vibrations, mais donne un poids plus important à une zone du spectre.

2.2.3 Les groupes des machines

L'analyse vibratoire est réalisée sur plusieurs machines tournantes pour cela elles sont regroupées selon les normes par des critères de distinction qui sont la puissance, les fondations et la présence d'effets de masse alternatifs non compensables (par exemple les machines à piston). [8]

Ces groupes sont :

2.2.3.1 Groupe I

Eléments de moteurs ou de machines qui, dans leurs conditions normales de fonctionnement, sont intimement solidaires de l'ensemble d'une machine (par exemple moteur électrique produit en série, puissance jusqu'à (15 kW).

2.2.3.2 Groupe II

Machines de taille moyenne (en particulier moteurs électriques de puissance comprise entre (15 et 75 kW) sans fondations spéciales. Moteurs montés de façon rigide ou machines (puissance jusqu'à 300 kW) sur fondations spéciales.

2.2.3.3 Groupe III

Moteurs de grandes dimensions et autres grosses machines ayant leurs masses tournantes montées sur des fondations lourdes et relativement rigides dans la direction des vibrations.

2.2.3.4 Groupe IV

Moteurs de grandes dimensions et autres grosses machines ayant leurs masses tournantes montées sur des fondations relativement souples dans la direction des vibrations (par exemple groupes turbogénérateurs, particulièrement ceux qui sont installés sur des fondations légères).

2.2.3.5 Groupe V

Machines et dispositifs mécaniques d'entraînement avec effets d'inertie non équilibrés (dus au mouvement alternatif des pièces), montés sur des fondations relativement rigides dans la direction des vibrations.

2.2.3.6 Groupe VI

Machines et dispositifs mécaniques d'entraînement avec effets d'inertie non équilibrés (dus au mouvement alternatif des pièces), montés sur des fondations relativement souples dans la direction des vibrations. Machines avec masses

tournantes accouplées souplement (par exemple arbres de broyeurs). Machines telles que centrifugeuses avec déséquilibres variables, capables de fonctionner isolément, sans l'aide d'éléments de liaison. Cribles, machines à tester la fatigue dynamique et génératrice de vibrations pour les industries de transformation.

2.2.3.7 Seuils de jugement

La norme propose pour chacun des quatre premiers groupes des seuils de jugement qui déterminent les domaines suivants :

- Bon.
- Admissible.
- Encore admissible.
- Inadmissible.

Ces seuils ne sont qu'une proposition basée sur une statistique regroupant de très nombreuses machines de types très différents. Il est de la responsabilité de l'utilisateur d'affiner ces seuils machine par machine, en fonction de leur historique. Ces seuils pourront ainsi être modifiés à la baisse ou à la hausse.

La norme ne propose aucun seuil de jugement pour les machines des groupes V et VI, car ils comprennent les machines alternatives, présentant des comportements très variables en fonction du nombre de cylindres, de l'angle entre ces cylindres et du calage des manetons.

Ces groupes comprennent également des machines à balourd variable ou des machines construites spécialement pour vibrer. Les seuils de jugement pour les machines des groupes V et VI ne pourront donc être déterminés que par le constructeur ou l'utilisateur.

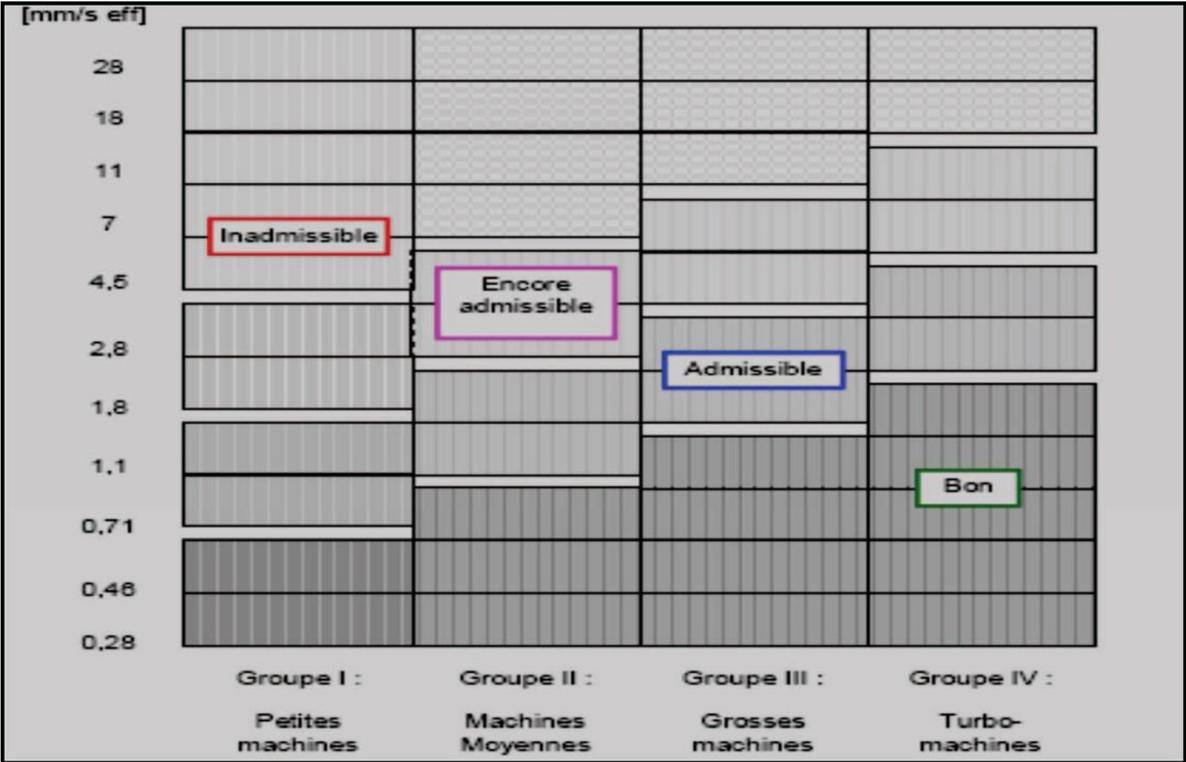


Figure 2.14 : Exemple de limites vibratoires proposées par les normes AFNOR E 90-300 ou ISO 2372.

2.2.4 Les défauts des machines tournantes et leurs fréquences [8]

Cause (défaut)	Fréquence	Amplitude	Remarques	Direction
Balourd	$1 * f_{rot}$	Proportionnelle à la vitesse de rotation Un déphasage de 90° sur deux mesures orthogonales	Amplitude max dans la direction radiale	Radiale
Alignement	$2.3 * f_{rot}$	En plus des vibrations radiale vibrations axiale Souvent importante	Les mesures axiales doivent effectuée sur le palier fixe de la ligne d'arbre	Radiale Axiale
Fixation Serrage	$1.2.3 * f_{Rot}$		Aucune déphasage sure deux mesure orthogonale	Radiale
Courroie défectueuse	$1.2.3 * f_{Rot}$	Importante dans la direction radiale	Le nombre de battement selon l'état de la courroie	Radiale
Engrenage	$Z * f_{rot}$	Niveau important en accélération	Bonds latiaux autour de f_e	Radiale axiale
Palier lisse (frottement)	(0.42-0.48) $* f_{rot}$	Très important si l'arbre faiblement charger	Vibration rencontrée sure les paliers avec arbre grand vitesse	Radiale
Défaut électrique	$1.2.3 * f_{Électrique}$	Constante et reproductible	Apparaissent seulement sous tension	Radiale axiale
Défaut hydraulique	Nbre pole $* f_{rot}$	Instable		Radiale axiale
Roulement	Haute fréquence	Non constante	Amplitude max	Radiale axiale
Faux round d'un pignon	$f_e = f_{rot} \text{ pignon}$		Bandes latérales autour de la fréquence f_e	Radiale axiale

Tableau 3.1 : Les défauts et leurs fréquences.[8]

La légende du tableau

- f_{rot} : fréquence de rotation.
- f_e : fréquence engrènement.
- Z : nombre des dents.

Ce tableau sera l'outil pour identifier les défauts existants dans notre dispositif d'étude.

2.3 Conclusion :

Dans ce chapitre, nous avons présenté la méthode de l'analyse vibratoire, qui permettra de mettre en évidence un grand nombre des défauts causeront les pannes dans les machines tournantes par l'identification de leurs fréquences.

Le but de cette méthode est d'éviter les arrêts de production non programmé, augmenter la durée de vie des équipements et assurer un contrôle de qualité permettent de mettre une procédure de maintenance efficace.

Chapitre 03

**LES SYSTEMES
EMBARQUES**

3 Chapitre 3 : les systèmes embarqués

3.1 INTRODUCTION

L'objectif de ce chapitre est de mettre l'accent sur la description des différents matériels qu'on a utilisés pour la réalisation, programmation et création d'une application. À savoir :

- Systèmes Embarqués
- La Carte Arduino
- Les Capteurs

3.2 PRESENTATION DES SYSTEMES EMBARQUES

3.2.1 Définition :

Un système embarqué est un système électronique et informatique autonome, souvent temps réel est constitué de puce électronique sur laquelle fonctionne un logiciel dédié à l'exécution de fonction spécifique ; le tout étant destiné à être intégré dans des sous-ensembles, équipements, appareils et produits divers. [9]

Le logiciel a une fonctionnalité fixe à exécuter qui est spécifique à une application. L'utilisateur n'a pas la possibilité de modifier les programmes. Bien souvent, il n'a pas conscience d'utiliser un système à base des microprocesseurs.

3.2.2 .Composition d'un système embarqué

Quelle que soit la nature et la complexité du système, un système embarqué (figure3.1) est composé d' :

- Une partie matérielle utilisée pour la performance
 - Microprocesseur, microcontrôleurs, DSP.
 - Mémoires.
 - Interfaces d'entrées /sorties.
- Une partie logicielle
 - Programmes.

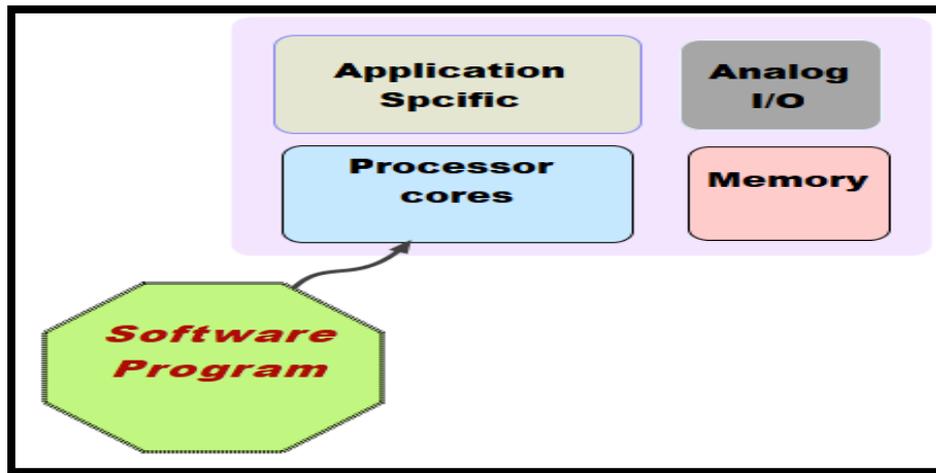


Figure3.1 : composition d'un système embarqué [9]

3.2.3 Caractéristiques

Les systèmes embarqués traitent certaines caractéristiques spécifiques et ces caractéristiques sont uniques dans chaque système embarqué. Certaines des caractéristiques importantes d'un système embarqué sont [10].

3.2.3.1 Spécification de domaine d'application

Un système embarqué exécute certaines fonctions prévues mais ne pouvant être employées dans aucun autre but. Des spécifications sont généralement attribuées et contribuent à la réduction du prix global du système embarqué.

3.2.3.2 . Temps-réel

Un système temps réel est un système qui répond à des contraintes temporelles n'est pas forcément un système qui va vite. Il doit pour cela garantir un déterminisme logique et temporel. Il doit également être fiable.[10]

On distingue trois types de temps réel :

- **Le temps réel dur** : C'est un système pour lequel un retard dans la délivrance du résultat rend le système inutile. C'est par exemple le cas du contrôle d'un missile.
- **Le temps réel mou** : C'est un système pour lequel un retard dans la délivrance du résultat n'est pas catastrophique. C'est par exemple le cas du système d'ouverture d'une barrière automatique.
- **Le temps réel ferme** : C'est un système pour lequel on peut tolérer quelques retards (peu souvent) dans la délivrance du résultat. C'est par exemple le cas d'un système multimédia type téléphone mobile.

. Réactivité

Un système embarqué doit par définition surveiller et contrôler un environnement. Si ce dernier présente lui-même un comportement changeant ou de type événementiel.

. Criticité, Fiabilité

Du fait de leur portabilité et de la mobilité des produits dans lesquels ils sont incorporés, les systèmes embarqués évoluent généralement dans de conditions environnementales non déterministes et souvent non maîtrisées. Ils sont exposés à des variations et autres contraintes environnementales susceptibles d'induire des défaillances. En même temps que s'accroît leur sophistication, les systèmes embarqués sont utilisés dans des applications de plus en plus critiques dans lesquels leur dysfonctionnement peut générer des nuisances. Ce type de systèmes doit garantir une très haute fiabilité et doit pouvoir réagir en cas de panne de l'un de ses composants. [11]

. L'autonomie

Les systèmes embarqués doivent en général être autonomes, c'est-à-dire remplir leur mission pendant de longues périodes sans intervention humaine. Cette autonomie est nécessaire lorsque l'intervention humaine est impossible, mais aussi lorsque la réaction humaine est trop lente ou insuffisamment fiable. [11]

. La consommation énergétique

La gestion de puissance est un autre facteur important qui doit être pris en considération lors de la conception des systèmes embarqués. Les processeurs utilisés dans les systèmes embarqués sont 2 à 3 décades moins puissantes qu'un processeur d'un ordinateur PC. La consommation énergétique est très faible, due à l'utilisation de batteries et/ou, de panneaux solaires voir de pile à combustible pour certain prototype.

3.2.4 . Architecture d'un système embarqué

Quelle que soit la nature et la complexité du système, on décompose un système embarqué en : [12]

- **Système contrôlé** : environnement (procédé) équipé d'une instrumentation qui réalise l'interface avec le système de contrôle.

- **Système de contrôle** : éléments matériels (microprocesseurs...) et logiciels dont la mission est d'agir sur le procédé via les actionneurs en fonction de l'état de ce procédé indiqué par les capteurs de manière à maintenir ou conduire le procédé dans un état donné.

L'architecture d'un système embarqué se définit par le schéma ci-dessous (figure 3.2).

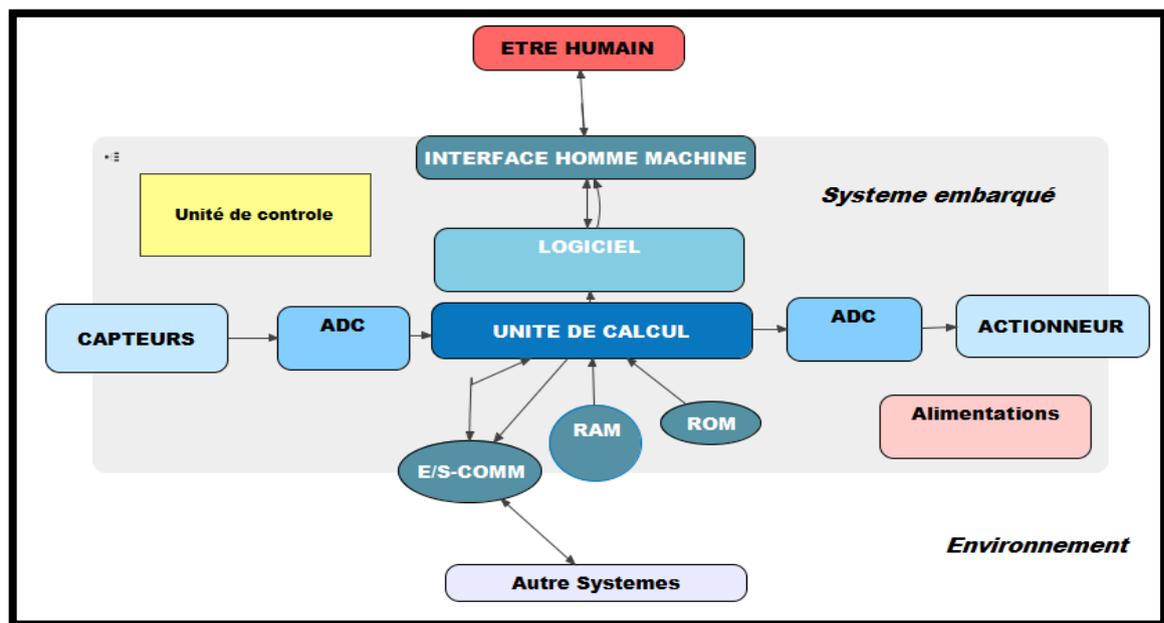


Figure 3.2 : architecture d'un système embarqué [12]

3.2.5 Classification des systèmes embarqués [12]

Nous pouvons classer les systèmes embarqués dans trois types comme suit :

- **Systèmes embarqués à petite échelle (SSES)** : Ces systèmes sont conçus avec un microcontrôleur simple de 8 ou 16 bits, ils ont peu de complexité de matériel et de logiciel et impliquent la conception de niveau conseil. Ils peuvent même être à piles.
- **Systèmes embarqués à échelle moyenne (MSES)** : Ces systèmes sont habituellement conçus avec des microcontrôleurs de 16 ou 32 de bit ou des DSPs simples ou RISCs. Ceux-ci ont des complexités de matériel et de logiciel avérées.
- **Systèmes embarqués sophistiqués (SES)** : Les systèmes embarqués sophistiqués ont d'énormes complexités de matériel et de logiciel et peuvent avoir besoin de processeurs extensibles ou de processeurs configurables et des réseaux logiques programmables (PLA).

3.2.6 Domaines d'application des systèmes embarqués

Les systèmes embarqués sont désormais utilisés dans des applications diverses tels que le transport (avionique, espace, automobile, ferroviaire), dans les appareils électriques et électroniques (appareils photo, jouets, postes de télévision, électroménager, systèmes audio, téléphones portables), dans la distribution d'énergie, dans l'automatisation, etc. (figure 3.3)



Figure 3.3 : domaines d'application des systèmes embarqués.[12]

3.3 Carte Arduino

Les cartes Arduino sont conçues pour réaliser des prototypes et des maquettes de cartes électroniques pour l'informatique embarquée. Ces cartes permettent un accès simple et peu coûteux à l'informatique embarquée. De plus, elles sont entièrement libres de droit, autant sur l'aspect du code source (Open Source) que sur l'aspect matériel (Open Hardware). Ainsi, il est possible de refaire sa propre carte Arduino dans le but de l'améliorer ou d'enlever des fonctionnalités inutiles au projet.

Le langage Arduino se distingue des langages utilisés dans l'industrie de l'informatique embarquée de par sa simplicité. En effet, beaucoup de bibliothèques et de fonctionnalités de base occultent certains aspects de la programmation de logiciel embarquée afin de gagner en simplicité. Cela en fait un langage parfait pour réaliser des prototypes ou des petites applications dans le cadre de hobby [13].

3.3.1 Le système Arduino permet de

- Contrôler les appareils domestiques.
- Fabriquer votre propre robot.
- Faire un jeu de lumières.
- Communiquer avec l'ordinateur.
- Télécommander un appareil mobile (modélisme) Etc...

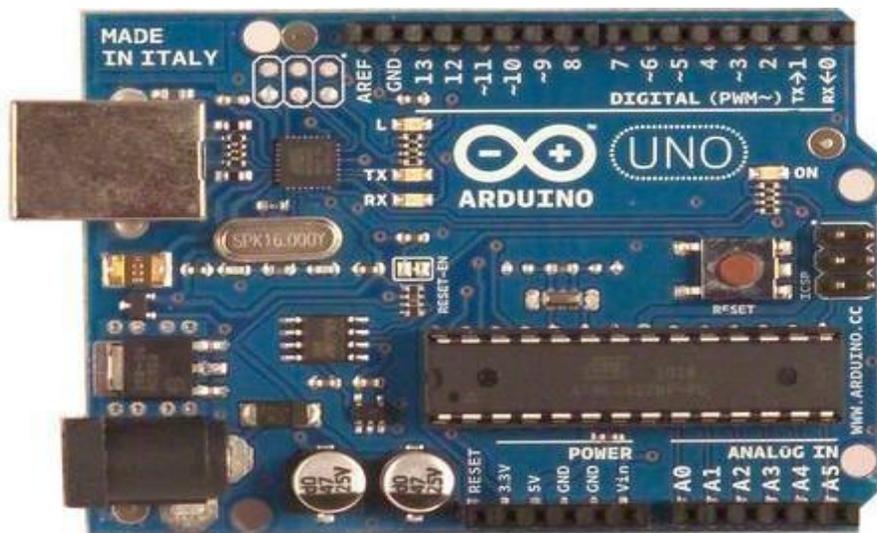


Figure 3.4 : La carte Arduino UNO

3.3.2 Le module Arduino

3.3.2.1 Présentation générale

Arduino est une gamme de circuits électroniques open source basée pour la plupart sur un microcontrôleur du fabricant Atmel. Ces circuits intègrent les composants nécessaires pour permettre une utilisation rapide et simple du microcontrôleur. Cette simplification vise à rendre accessibles à tous la création et la programmation d'objets ou dispositifs interactifs. Ces objets peuvent contenir toutes sortes de capteurs, d'indicateurs lumineux ou d'interrupteurs que l'on souhaite faire intervenir.

Entre autres, les cartes Arduino sont équipées de connecteurs standardisés pour brancher des modules compatibles appelés shields. Ces derniers sont des circuits d'une taille plus ou moins semblable à celle de l'Arduino et qui viennent s'empiler sur ces connecteurs. Ils proposent des extensions matérielles qui permettent d'ajouter des fonctionnalités originales à son projet. En plus de ces connecteurs, les cartes possèdent toutes une connectique USB permettant de programmer facilement le microcontrôleur qu'elles embarquent.

3.3.2.2 Types de cartes Arduino

Il existe plusieurs types de cartes Arduino, nous citons : [14]

Arduino Uno



Figure 3.5: Arduino UNO

- **Arduino Méga**



Figure 3.6 : Arduino Méga

- **Arduino Nano**

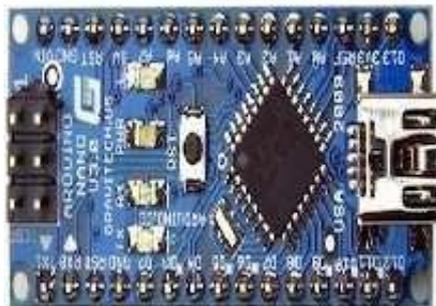


Figure 3.7 : Arduino Nano

- **LilyPad Arduino**

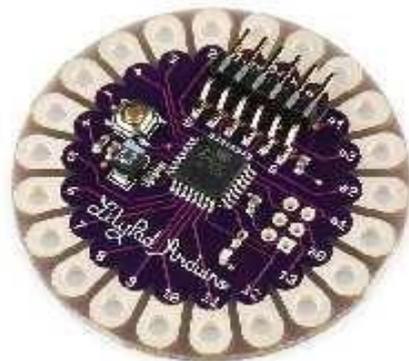


Figure 3.8 : LilyPad Arduino

- **Arduino Leonardo**



Figure 3.9 : Arduino Leonardo

- **RedBoard**

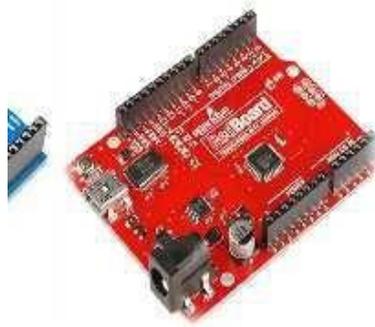


Figure 3.10 : RedBoard

3.3.2.3 Critère de choix d'une platine Arduino

L'intérêt principal des cartes Arduino est leur facilité de mise en œuvre. Arduino fournit un environnement de développement s'appuyant sur des outils open source. Le chargement du programme dans la mémoire du microcontrôleur se fait de façon très simple par port USB. Le choix d'une platine de développement Arduino, est dicté par le besoin du concepteur du montage. Le tableau 3.2

Objectifs recherchés	Arduino
Commencer à programmer à moindre coût	Uno
Chercher à brancher une multitude de capteurs, utiliser plusieurs périphériques En série	<i>Mega</i>
Souhaiter les mêmes propriétés que l'Arduino Mega mais avec une puissance de Calcul supérieure	<i>DUE</i>
Chercher à miniaturiser le montage	<i>Mini.</i>
Cherchez un véritable ordinateur embarqué comparable à une platine Raspberry™	<i>Yun</i>

Tab 3.2: Choix d'une platine Arduino en fonction des objectifs recherchés

[14]

3.3.3 La carte Arduino Uno

3.3.3.1 Présentation

La carte Arduino Uno est le produit populaire parmi les cartes Arduino. Parfaite pour débiter la programmation Arduino, elle est constituée de tous les éléments de base pour construire des objets d'une complexité relativement faible.

La carte Arduino Uno, comme son nom l'indique, a été la première à utiliser la version de programmation Arduino 1.0, et elle est devenue le symbole de l'univers Arduino.

La carte Arduino Uno est constituée de 14 broches d'entrées/sorties digitales, dont six sont utilisables en PWM, de 6 broches d'entrées analogiques, d'une connectique USB, d'une connectique d'alimentation, d'un port ICSP et d'un bouton RESET. [14]

La description de toutes les connectiques de la carte Arduino UNO est présentée dans les deux images ci-dessous :

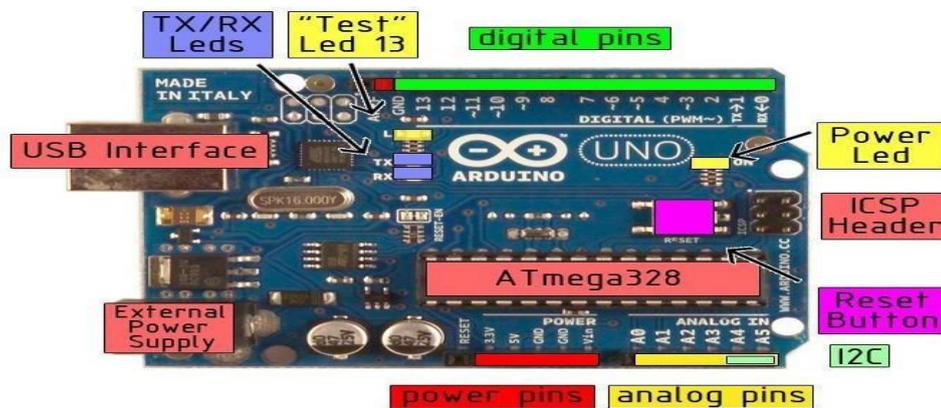


Figure 3.11 : Description des entrées/sorties de la carte Arduino Uno

3.3.3.2 Ces avantages

- Pas cher !
- Environnement de programmation clair et simple.
- Multiplateforme : tourne sous Windows,
- Nombreuses bibliothèques disponibles avec diverses fonctions implémentées.
- Logiciel et matériel open source et extensible.
- Nombreux conseils, tutoriaux et exemples en ligne (forums, site perso etc....)
- Existence de « shield » (boucliers en français) : ce sont des cartes supplémentaires qui se

3.4 Les capteurs de vibration

La première étape conduisant à l'obtention d'une lecture de vibration consiste à convertir la vibration mécanique produite par une machine en un signal électrique équivalent. Cette opération est réalisée au moyen des capteurs de vibrations, on retrouve parmi les capteurs les plus couramment utilisés le proximètre (mesure de déplacement), la vélocimétrie (mesure de vitesse) et l'accéléromètre (mesure d'accélération). [15]

3.4.1 Les proximètres

Le proximètre, ou sonde capteur de déplacement sans contact directement proportionnel au déplacement relatif de la vibration d'un arbre ou d'un rotor, il est monté en permanence à l'intérieur du palier [Figure 3.12], les mesures en déplacement ne sont pas quantifiables dans toutes les gammes de fréquence, ces mesures seront limitées aux basses fréquences (< 100 Hz).

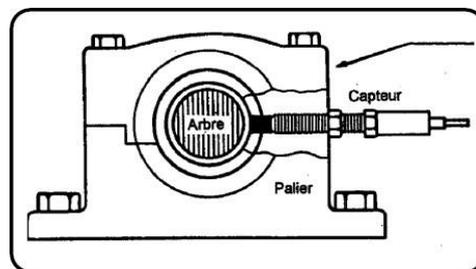


Figure 3.12 : proximètre monte sur un palier [15]

3.4.1.1 Les avantages est les inconvénients

Avantages

- Mesure directement les mouvements d'arbre
- Même capteur pour les butées axiales, les vibrations radiales et la vitesse
- Mesure directement le déplacement
- Pas de pièce mobile

Inconvénients

- Gamme de fréquence limitée. Pas de détection des défauts de roulements
- Installation
- Sensible au matériau de l'arbre
- Restriction de températures.

❖ Problèmes et défauts détectés

- Arbre
- Palier lisse
- Buté
- Généraux : balourd, désalignement, usure, et...

3.4.2 Vélométrie

Les capteurs de vitesse, ou vélocimétries, sont constitués d'une sonde à contact dite sonde sismique qui mesure le mouvement absolu de l'organe sur lequel elle est fixée.

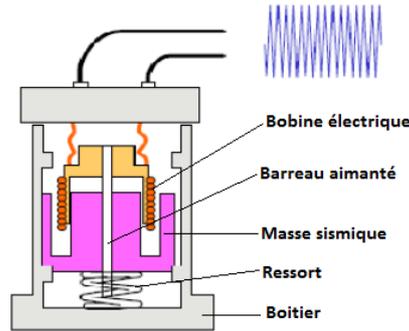


Figure 3.13 : Schéma de principe d'une vélocimétrie [15]

3.4.3 Les accéléromètres :

Un accéléromètre piézoélectrique [Figure 3.14] est composé d'un disque en matériau piézoélectrique (quartz), qui joue le rôle d'un ressort sur lequel repose une masse sismique précontrainte.

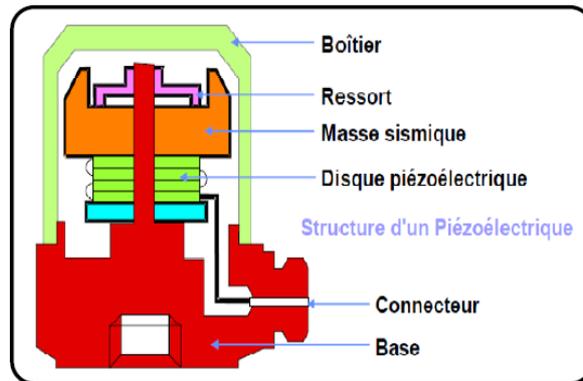


Figure 3.14 : Schéma de principe d'un accéléromètre.[15]

Le mode de fixation de l'accéléromètre sur la structure a une influence considérable sur la réponse du capteur : Plus la fixation est rigide, plus la réponse s'élargit vers les hautes fréquences

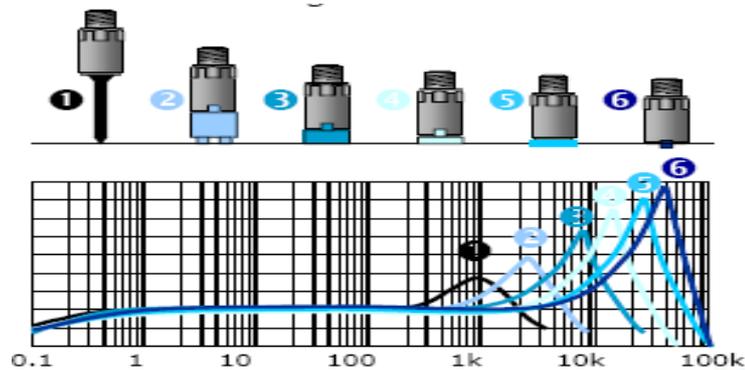


Figure 3.15 : Mode de fixation de l'accéléromètre [15]

Les accéléromètres piézoélectriques tendent à devenir les capteurs de vibration absolue les plus utilisés pour la surveillance. Ils possèdent les propriétés suivantes :

- Utilisables sur de très grandes gammes fréquentielles
- Excellente linéarité sur une très grande gamme dynamique (typiquement 140 dB).
- Le signal d'accélération peut être intégré électroniquement pour donner le déplacement et la vitesse
- Aucun élément mobile, donc extrêmement durable.

3.4.3.1 Les avantages est les inconvénients :

Les avantages :

- Facile à installer
- Petit, léger
- Supporte les hautes températures
- Pas de pièce mobile

Les inconvénients :

- Nécessite une double intégration pour le déplacement
- Nécessite une source extérieure
- Fournit des informations limitées sur la dynamique d'arbre
- Médiocre pour les faibles vitesses

Problèmes et défauts détectés :

- Roulements
- Engrenages
- Machine à pâles
- Machine électrique
- Généraux : balourd, désalignement, usure, etc.

3.5 Les capteurs de températures

Vue l'importance de contrôler la température pour les machines industrielles, pour cela nous citons quelques types des capteurs.

3.5.1 Le capteur DHT11 [16]

Le capteur DHT11 est lui capable de mesurer des températures de 0 à +50°C avec une précision de

+/- 2°C et des taux d'humidité relative de 20 à 80% avec une précision de +/- 5%. Une mesure peut être réalisée toutes les secondes.

Le DHT11 compatibles 3.3 volts et 5 volts (le fabricant recommande cependant de toujours alimenter le capteur en 5 volts pour avoir des mesures précises).



Figure 3.16 : Capteur DHT11 [16]

Les caractéristiques du capteur DHT11

Humidité (relative %)	20 ~ 80%
Précision (humidité)	+/- 5%
Température	0 ~ +50°C
Précision (température)	+/- 2°C
Fréquence mesure max	1Hz (1 mesure par seconde)
Tension d'alimentation	3 ~ 5 volts
Stabilité à long terme	+/- 1% par an

Tableau 3.3 : caractéristiques du capteur DHT11 [16]

3.5.2 Capteur de température LM 35

Le capteur de température LM35 est un capteur de température analogique fabriqué par Texas Instruments, il est extrêmement populaire en électronique, car précis, peu coûteux, très simple d'utilisation et d'une fiabilité à toute épreuve, Il peut fonctionner dans une plage de -55°C à +150°C.

Il dispose de trois pattes, la première pour la tension d'entrée (l'alimentation), la deuxième c'est la tension de sortie V_{out} la troisième pour la masse. Dont la tension de sortie V_{out} est linéairement proportionnelle à la température exprimée en degré Celsius, plus exactement, la tension de sortie V_{out} augmente de 10 mV chaque fois que la température augmente de 1°C, selon les données fournies par le constructeur [16].

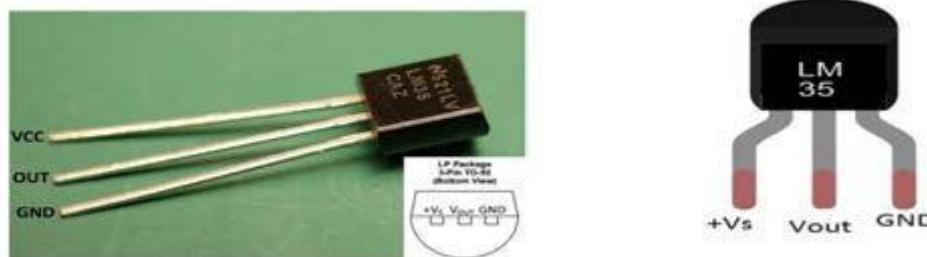


Figure 3.17 : Vue du Capteur LM35[16]

Caractéristiques : [16]

Nombre de broches	3 broches
Dimension	4.30 mm x 4.30 mm
Tension d'alimentation min	4V
Tension d'alimentation max	20V
Température Min	-55 °C
Température Max	+155°C
Consommation	60μA
Gain de la sonde (sortie)	10mV /°C
Précision	±0.75°C

Tableau 3.4 : caractéristiques du capteur LM35 [16]

3.6 Les applications des vibrations mécaniques dans la maintenance

Dans cette partie nous présentons des travaux qui ont utilisé l'analyse de vibration dans la maintenance.

- [MECHTA Soumia et EL-EUDJAMA Marwa, 2020 TIARET] elles ont développé une application didactique qui a pour but d'analyser les vibrations des machines tournantes. Elles ont utilisé un capteur de l'accéléromètre GY-61 ADXL 335 et un module LCD pour l'affichage de résultat de vibration. La figure suivante présente le travail réalisé par [17].



Figure 3.18: Montage et localisation du capteur ADXL [17]

- [KOUCHIH Mourad et BOUCHOUKA Hocine, 2017 TIARET] Leur objectif est la maintenance des machines tournantes par l'analyse vibratoire. L'accéléromètre AS-065 est utilisé pour la mesure de l'accélération vibratoire. Il est raccordé au collecteur analyseur **VIBROTEST 60**. Ils ont utilisé le collecteur **SCHENK VIBROTEST 60** (Figure 3.19) qui permet de mesurer la valeur globale, les spectres en fréquences et les spectres enveloppes (**BCS**). Les données collectées sont stockées dans une carte

PCMCA.



Figure 3.19 : Collecteur SCHENK VIBROTEST 60 [18]

- Selon [HACHEMI Mohammed,2012] La surveillance vibratoire des machines tournante dans l'industrie jeux un rôle très importante pour prolongée la durée de vie des équipements pour cela le travail de [19] a réalisé une application sur la pompe centrifuge P105 de module 4 du site HASSI R'MEL, par l'utilisation de la technique de visualisation de mouvement vibratoire de La déformée opérationnel (ODS) pour analyser des problèmes vibratoires.



Figure3.20 : La pompe centrifuge P105 [19]

-Selon [CHAIB Sid ali et BOUSSAIDI Azeddine, 2016] La maintenance des machines tournantes par l'analyse des vibrations a considérablement évolué ces dernières années grâce aux techniques avancées de traitement du signal. Ces techniques permettent maintenant de détecter à un stade précoce l'existence d'un défaut, d'identifier sa nature, de déterminer sa gravité et de suivre son évolution. Pour cela le travail de [20] s'inscrit sur la surveillance et diagnostic des défauts mécaniques par l'application d'une technique de traitement de signal. Pour mesurer les effets des vibrations engendrées

par le fonctionnement ou les dysfonctionnements d'un roulement à billes, ils ont utilisé des capteurs externes au roulement. Des accéléromètres mesurant les accélérations subies. Les accéléromètres sont souvent de nature piézo-électrique. Puis ils ont fait une simulation sous Matlab dans le but d'extraire des signatures spectrales d'un roulement à l'état sain (fonctionnement normal), et à l'état défaillant. Ils ont fait une comparaison entre le spectre de référence et les spectres des états défaillants afin de suivre l'évolution de chaque défaut d'un élément de roulement en fonction de son diamètre.

3.7 Conclusion

Dans ce chapitre nous avons présenté les différents matériels utilisés dans notre projet pour réaliser la partie de la maintenance conditionnelle.

Vue l'importance des capteurs de vibrations et de températures pour bien suivre l'état de fonctionnement des machines.

Dans ce chapitre nous avons présentés le principe de fonctionnement des capteurs de vibration et de température, et nous avons terminé ce chapitre par des applications qui ont utilisés l'analyse de vibrations dans le domaine de maintenance.

Chapitre 04

**La Conception et la réalisation
d'un système de GMAO
amélioré**

4 Chapitre 4 : la Conception et la réalisation d'un système de GMAO amélioré

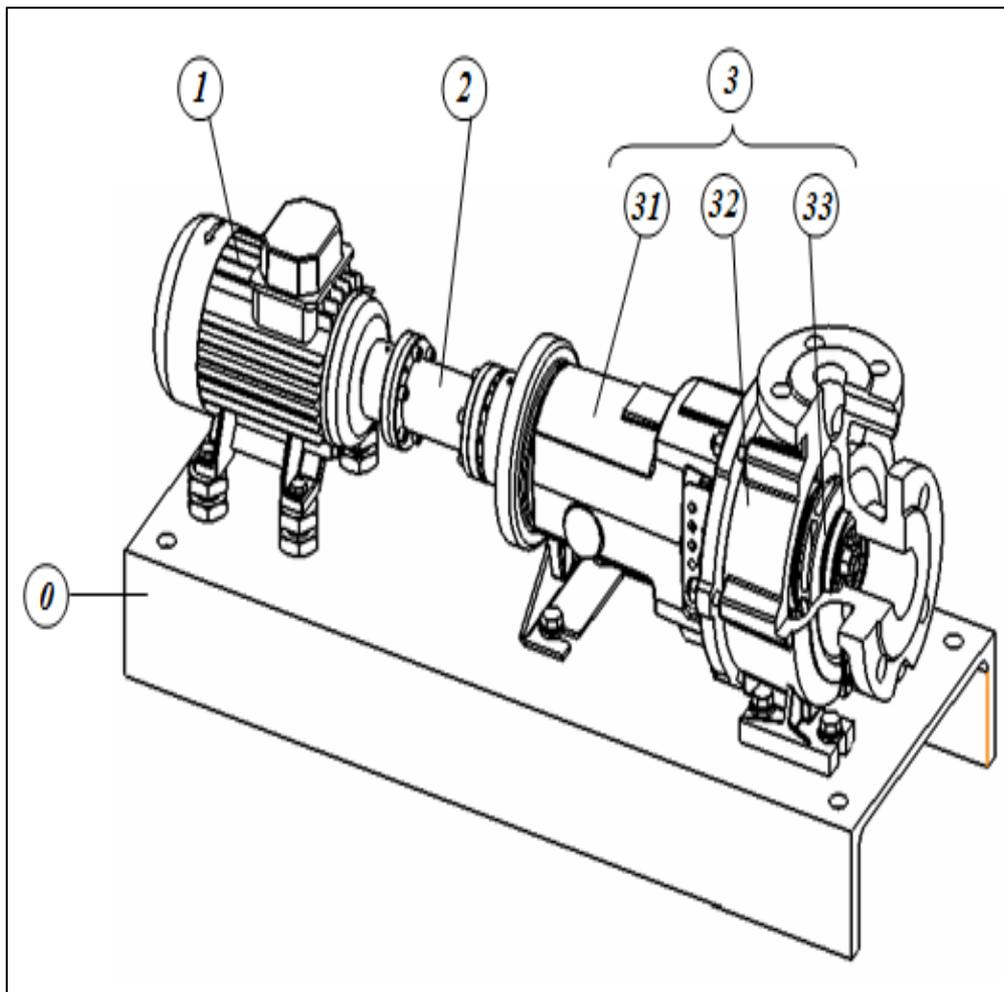
4.1 Introduction

L'amélioration de la GMAO peut intervenir par rapport d'une part la maintenance préventive. Et d'autre part, par rapport la maintenance conditionnelle, La mise en œuvre d'un système de GMAO amélioré inclut : Des informations sur le niveau vibratoire et sur les mesures de température, des informations sur maintenance préventive et L'historique des interventions. Ces informations permettant de comprendre et d'analyser bien le système de maintenance. Dans ce chapitre, nous utilisons la notation UML (Unified Modeling Language), pour représenter les différents modèles de notre système de maintenance. Nous utilisons le SGBD Microsoft Access pour l'implémentation et la réalisation de base de données de maintenance et un système embarqué pour la réalisation de la collecte des informations vibratoires.

4.2 Etude de système de maintenance par la méthode UML

La méthode UML nous a permet d'étudier et de modéliser un système de maintenance, nous avons pris l'étude de cas d'un groupe motopompe électrique. Pour utiliser la méthode UML il faut premièrement faire une étude bien détaillée sur l'analyse fonctionnel du système à modéliser pour collecter les informations et les besoins de l'utilisateur. Pour cela nous avons fait un stage au niveau de l'entreprise ADE pour comprendre bien le principe de fonctionnement de la pompe et l'importance de la surveillance vibratoire et température pour suivre état de la machine afin d'éviter les défaillances. Et aussi l'analyse de rapport de stages qui nous a aidée à connaitre les différents composants de la pompe pour améliore notre système par l'injection des capteurs de vibration et de température par réaliser un système de GMAO amélioré.

La **figure 4.1** représente un groupe motopompe électrique, Il est composé d'un moteur électrique [1] entraînant une pompe hydraulique centrifuge [3] au moyen d'un accouplement élastique [2]. L'ensemble est monté sur un châssis fixe [0]. En plus de ce composant, il comprend souvent des composants pour la régulation de la pression, du débit ou d'autres paramètres.



La figure 4.1 : un groupe motopompe électrique

La figure suivante présente le Découpage du système :

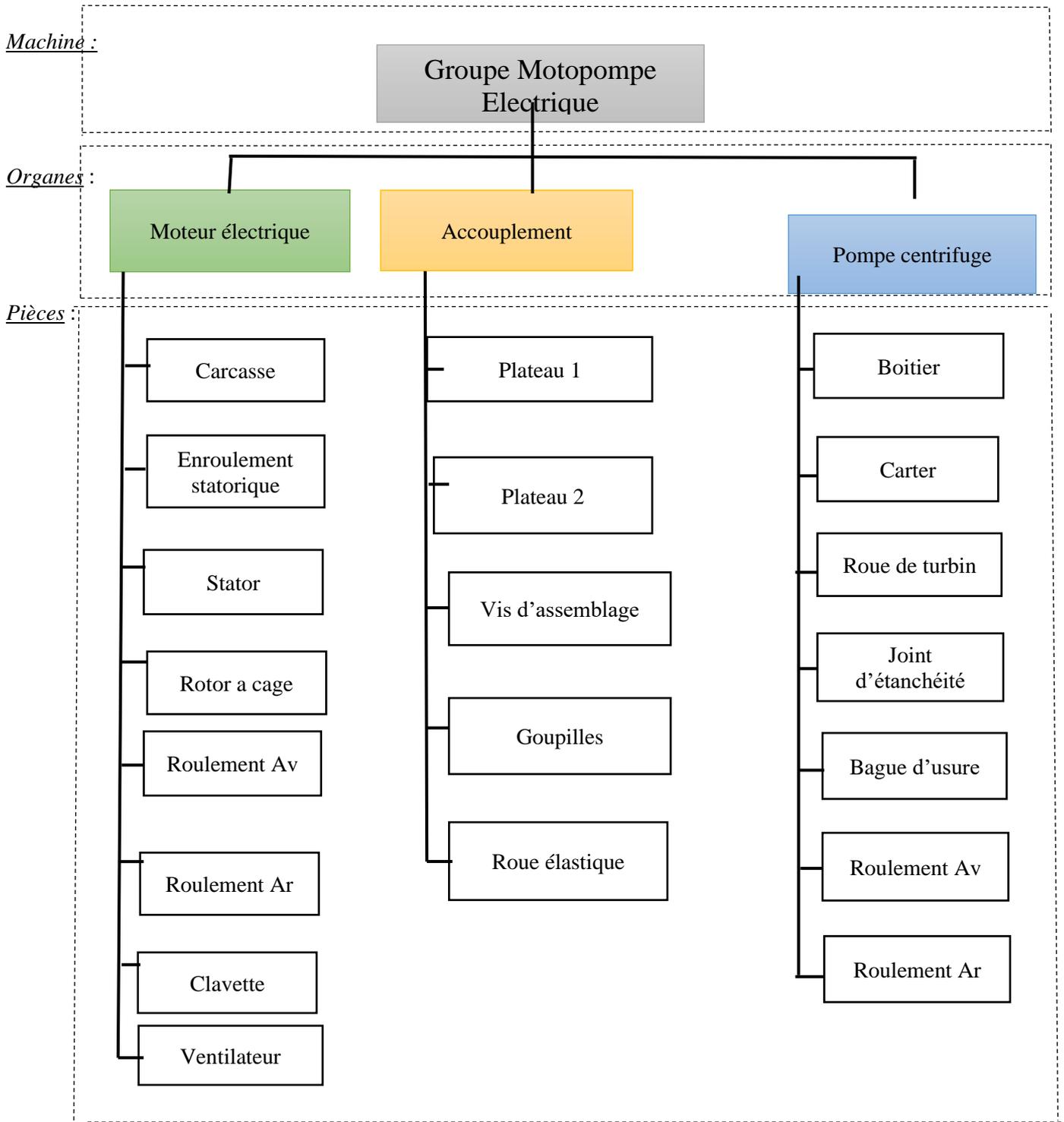


Figure.4.2 : Découpage de système motopompe.

Les groupes électropompes sont constitués de deux parties :

- **Le moteur** : permet un mouvement de rotation pour faire tourner la pompe.
- **La pompe** : c'est la partie entraînée par le moteur pour aspirer l'eau et faire le refoulement.
- Les deux sont liés par **un accouplement** rigide à manchon élastique qui diminue les frottements dus au choc de démarrage.



Figure4.3 : groupe électropompe au niveau de l'entreprise ADE.

4.2.1.1 Les fonctions principales et complémentaires d'une pompe centrifuge :

- Doit pomper le fluide.
- Assurer le bon fonctionnement de la pompe.
- Doit résister les conditions de l'environnement.
- Respecter les normes et les consignes de sécurité.
- Minimiser les couts de maintenance.
- Débitier le fluide sous pression souhaitée.
- Transmettre la puissance vers pompe.
- Entraîner la pompe centrifuge.

4.2.1.2 Les fonctions principales et complémentaires d'accouplement

- Transmettre la puissance du moteur vers la pompe.
- Assurer le bon fonctionnement.
- Doit résister les conditions environnementales.
- Doit respecter les normes de sécurité.
- Doit convenir au système motopompe (puissance, vitesse de rotation, dimensions, etc.).

- Entrainer la pompe (émis de puissance).
- Entrainer par le moteur (réception de puissance).

4.2.1.3 Les fonctions principales et complémentaires de Moteur électrique

- Transmettre l'énergie du moteur via l'accouplement.
- Diminuer les couts de maintenance du moteur par l'homme de maintenance.
- Assurer le bien fonctionnement du moteur.
- Doit résister les conditions environnementales.
- Doit respecter les normes de sécurité de fonctionnement et les interventions.
- Doit réduire les couts de maintenance.
- Transmettre le couple de rotation du moteur.
- Convertir l'énergie électrique en énergie mécanique.

Nous intéressons à l'Analyse fonctionnelle du système et les techniques de détections utilisées afin de mettre des capteurs pour objectif de suivre la maintenance, les trois tables présentent chacune les éléments et leurs fonctions et les techniques utilisées pour détecter la défaillance.

Machine : Motopompe		
Organe : Pompe centrifuge		
Pièce/Elément	Fonction	Détection
Arbre	Transmettre une puissance Sous forme d'un couple et d'un mouvement de rotation	Analyse Vibratoire
Roue de turbine	Donner une vitesse au liquide	Bruit Signal d'alarme
Roulements	Supportent une charge maintien de l'axe Réduire la friction	Vibration Bruit
Joint d'étanchéité	Assurer l'étanchéité	Fuite
Garniture	Assure l'étanchéité de la pompe	Fuite d'eau Et d'huile
Ecrou	Faire blocage de la roue	Visuel
Bague d'usure	Assure-le non-passage de fluide	Fuit

Tableau 4.1. : les éléments et leurs fonctions et les techniques utilisées pour détecter la défaillance.

Machine : Motopompe		
Organe : Moteur électrique		
Pièce/Élément	Fonction	Détection
Bobinage	Crée le champ magnétique nécessaire à la rotation du moteur.	Visuel
Spires du rotor	Produire la force magnétique	Signalisation Sonore
Roulement du moteur	Supportent une charge maintien de l'axe Réduire la friction	Vibration Bruit
Ventilateur	Refroidir le moteur	Vibration Echauffement Bruit
Boite à corne	Alimenter et commander le moteur	Signalisation Sonore
Bâti	Renforme le moteur	Echauffement Bruit

Tableau 4.2. : les éléments et leurs fonctions et les techniques utilisées pour détecter la défaillance.

Machine : Motopompe		
Organe : Accouplement et autres composants		
Pièce/Élément	Fonction	Détection
Vis d'assemblage	Assurer l'assemblage des parties de l'accouplement	Visuel (capteur de vibration)
Roue élastique	Absorber les vibrations des arbres	Vibration Bruit
Boitier (corps de pompe)	Protéger la pompe	Visuel
Palier de pompe	Guider et supporter l'arbre de rotation	Capteur
Châssis	Porter le groupe motopompe	Visuel
Impulseur	Transmettre l'énergie de l'arbre au produit	Capteur
Limiteur de pression	Abaisse de pression de fluide	Signal d'alarme (capteur de pression)

Tableau 4.3 : les éléments et leurs fonctions et les techniques utilisées pour détecter la défaillance.

4.2.2 Analyse de température d'un groupe motopompe

Selon le document technique de groupe motopompe l'entreprise ADE, la température doit être observée pendant son fonctionnement :

- La température d'entrée
- La température dans la tubulure de refoulement
- La température des garnitures mécaniques
- La température du liquide de refroidissement à la sortie du refroidisseur (différence de température : 10 °C ou maximum).

Donc la température a une importance énorme pour le bon fonctionnement de système.

4.2.3 Modélisation par UML

Afin de mieux expliciter les fonctionnalités de système de GMAO et modéliser ses objets pour le cas d'un groupe motopompe électrique, nous utilisons le langage de modélisation UML, (cas d'utilisation, diagramme de classe). La modélisation par UML se base les composants d'un groupe motopompe électrique avec le découpage du système pour créer le diagramme de classe et sur l'analyse fonctionnelle de chaque Pièce/Élément sa ou ses fonctions et ses techniques de détection.

4.2.3.1 Diagramme des cas d'utilisation

Pour décrire les fonctionnalités de notre système de gestion de la maintenance, nous utilisons le diagramme de cas d'utilisation. Nous modélisons chaque fonctionnalité par un diagramme de cas d'utilisation. Un cas d'utilisation « use case » représente un ensemble de séquences d'actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier. Chaque cas d'utilisation permet de décrire ce que le futur système devra faire, sans spécifier comment il le fera.

Un acteur représente un rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié. Dans notre cas l'acteur représente technicien de maintenance ou un chef service de maintenance etc. impliqué dans le système de la maintenance et les capteurs de vibrations et les capteurs de température (des agents automatiques) qui sont installés dans des endroits bien définis pour collecter des informations sur l'état de système.

4.2.3.1.1 Fonctionnalités du groupe « gestion des utilisateurs »

La figure ci-après montre les différents cas d'utilisation de ce groupe.

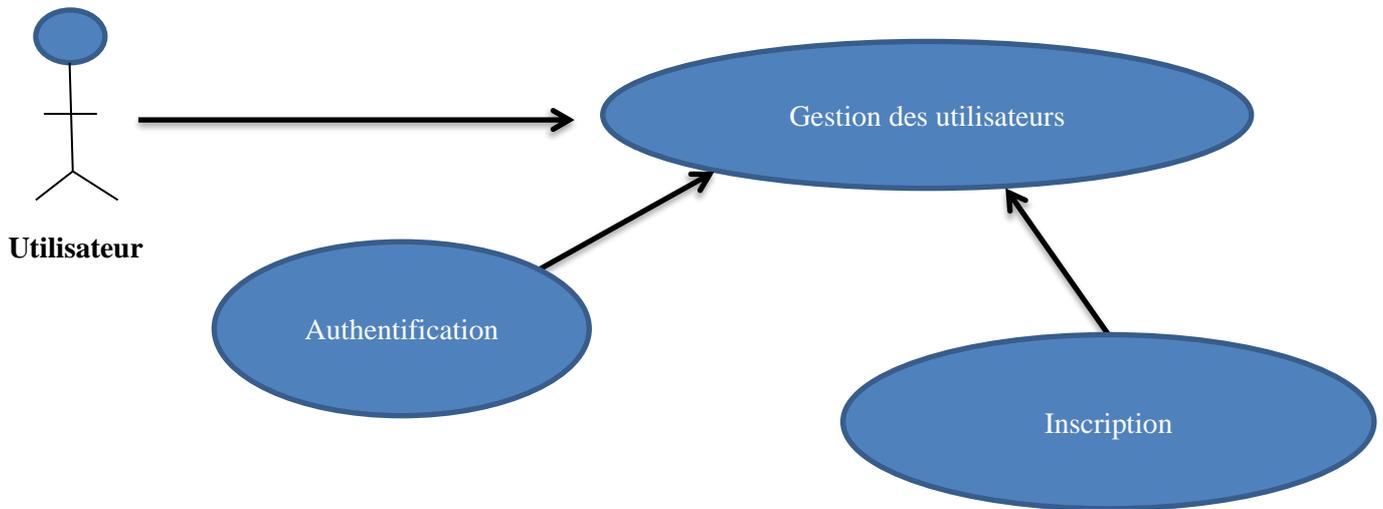


Figure4.4 : les différents cas d'utilisation de ce groupe.

- **Cas d'utilisations « Authentification » Pré condition. Néant**

Enchaînement.

- L'utilisateur choisit de s'authentifier dans la GMAO
- L'utilisateur saisit son nom utilisateur et son mot de passe
- Si l'authentification s'effectue avec succès, les fonctionnalités de GMAO deviennent accessibles pour l'utilisateur
- Sinon, un message d'erreur s'affiche à l'utilisateur

- **Cas d'utilisations « inscription » Pré condition. Néant**

Enchaînement.

- L'utilisateur qui veut utiliser la GMAO doit remplir un formulaire d'inscription
- Après l'avoir rempli, l'utilisateur envoie le formulaire
- Si l'utilisateur n'est pas inscrit, alors le système utilise la base de données des acteurs impliqués dans le système de maintenance pour vérifier l'existence de l'utilisateur
- Si l'acteur existe alors il est enregistré comme utilisateur de GMAO
- Sinon, le cas d'utilisations se termine par un échec

Post condition. Néant

- **Fonctionnalités du groupe « Gestion des organes ».**

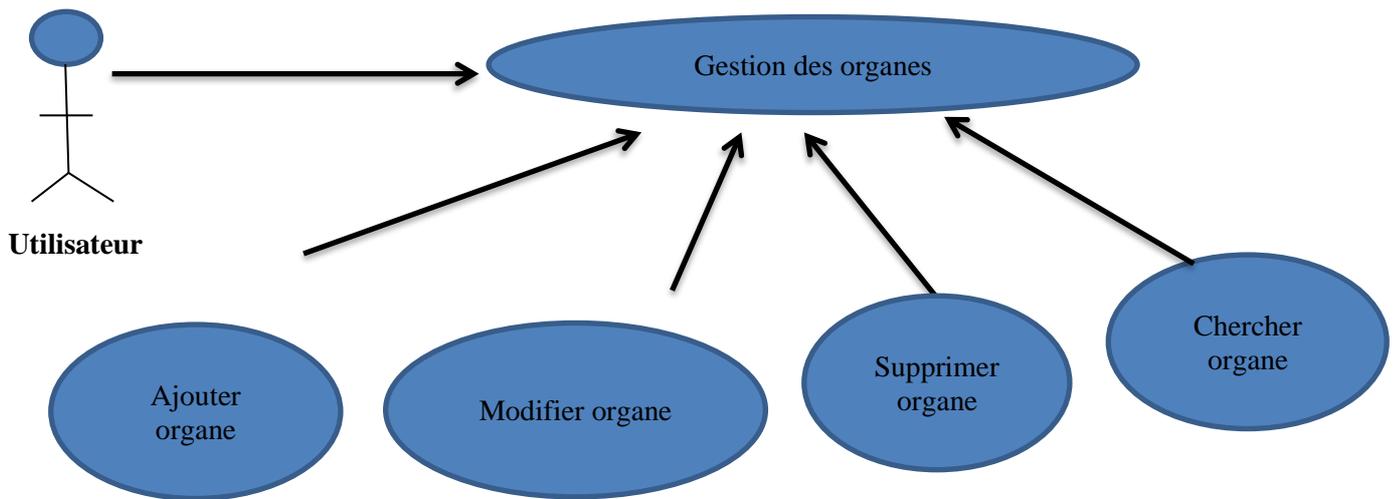


Figure4.5 : Fonctionnalités de groupe de gestion des organes

4.2.3.1.2 Fonctionnalités du groupe « gestion des maintenances »

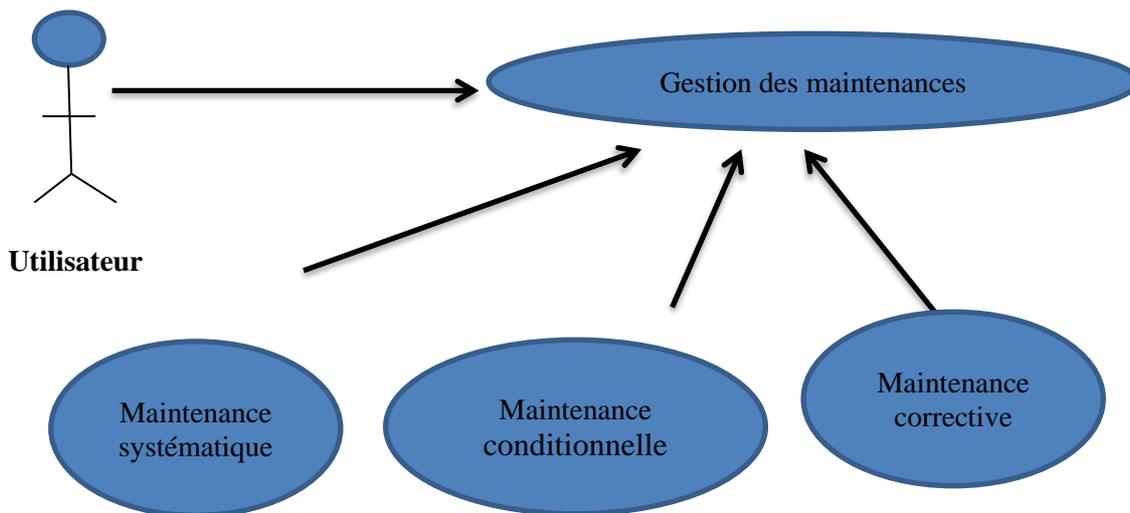


Figure4.6 : Fonctionnalités de groupe de gestion des organes

4.2.3.1.3 Fonctionnalités du groupe « analyse de vibration »

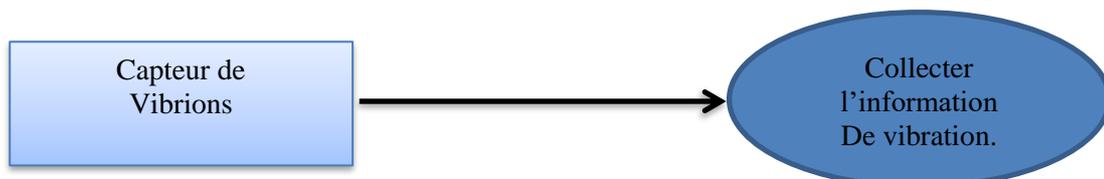


Figure4.7 : Fonctionnalités de groupe « analyse de vibration »

4.2.3.1.4 Fonctionnalités du groupe « analyse de tempérer »

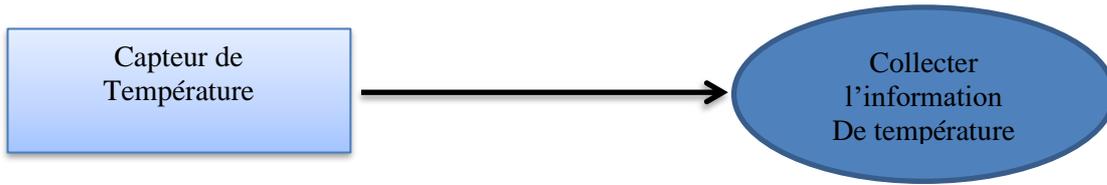


Figure4.8 : Fonctionnalités de groupe « analyse de tempérer »

4.2.3.2 Conceptions statiques des composants GMAO :

Nous représentons l’aspect statique du système par un diagramme de classes. La figure 4.9 montre le digramme des classes de la GMAO

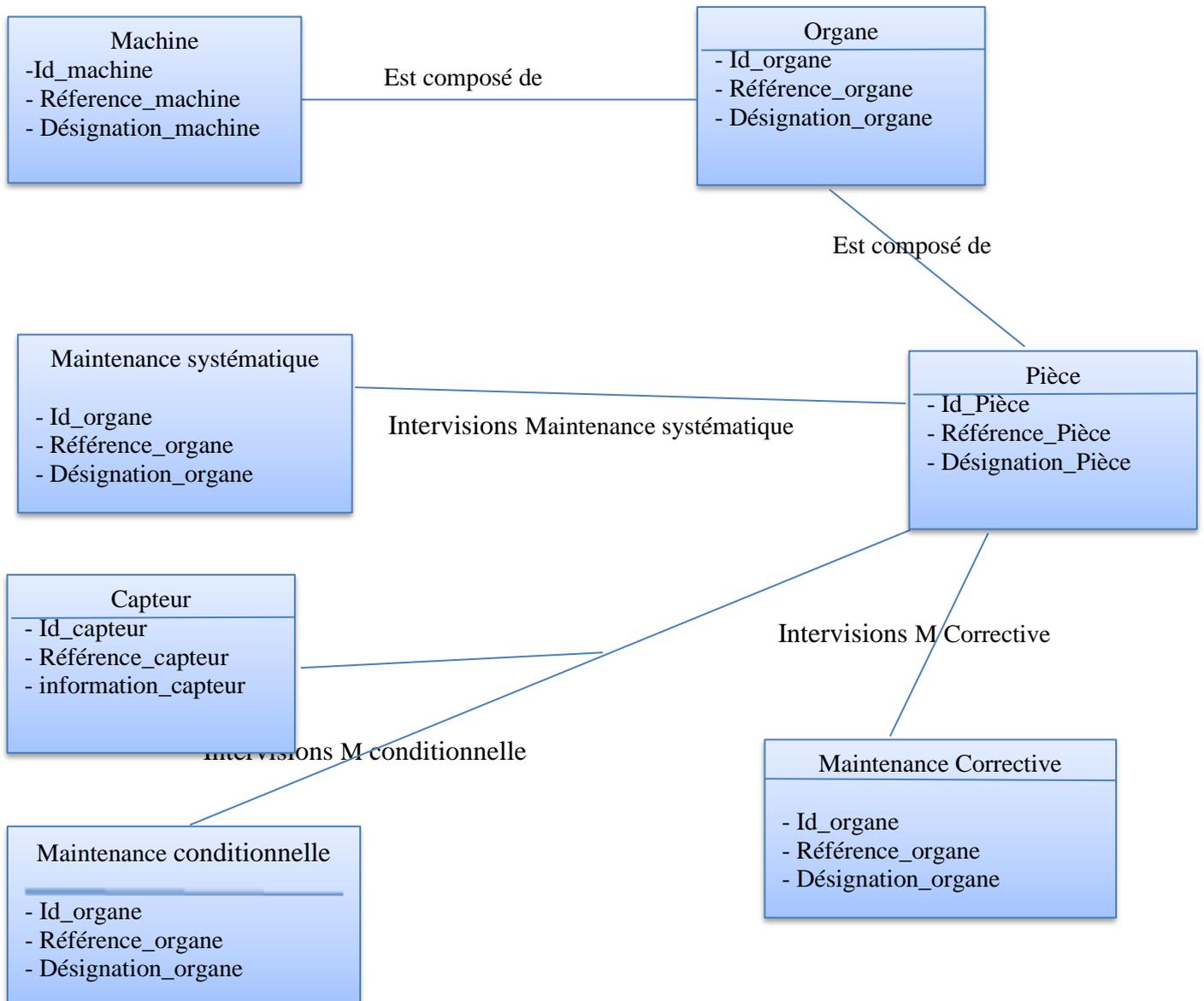


Figure4.9 : Diagramme des classes UML de GMAO améliorée

4.2.4 La réalisation de la GMAO

Pour implémenter un système d'information de maintenance, il faut une base de données

associée pour stocker ces informations. Pour cela Nous avons utilisé le système de gestion des basées de données (SGBD) Microsoft Access.

Microsoft Access est une base de données relationnelle éditée par Microsoft. Ce logiciel fait partie de la suite Microsoft Office. Microsoft Access permet la création des tables, des relations entre les tables, les requetés SQL ...etc. La figure4.9 suivante présente l’interface principale de GMAO développé par Microsoft Access.



Figure 4.9 : l’interface principale de GMAO

La figure4.10 suivantes présentent les deux maintenances systématiques Conditionnel :

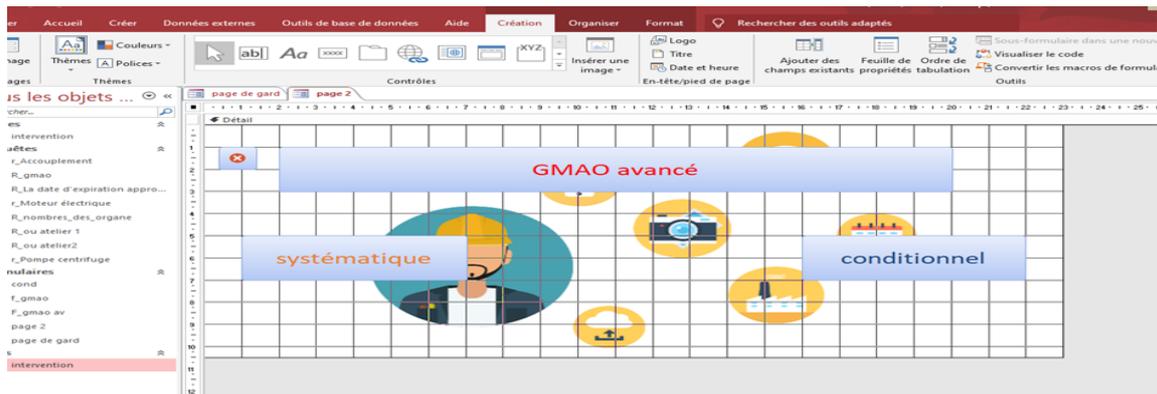


Figure4.10 : l’interface principale de GMAO

Une fois on clique sur le bouton systématique la flique4.11 s’affiche pour faire les opérations nécessaires.

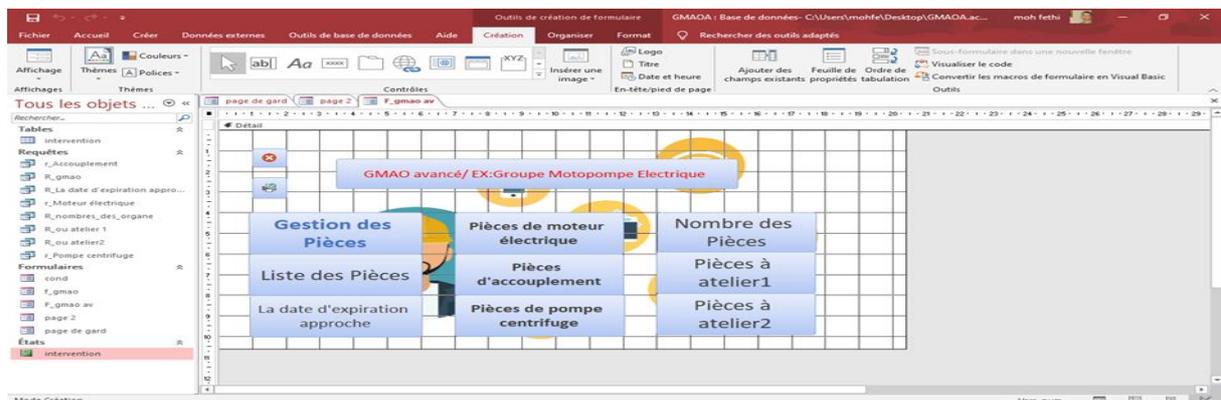


Figure4.11 : l’interface principale de GMAO

Figure4.12 l’interface de saisir les données d’intervention.

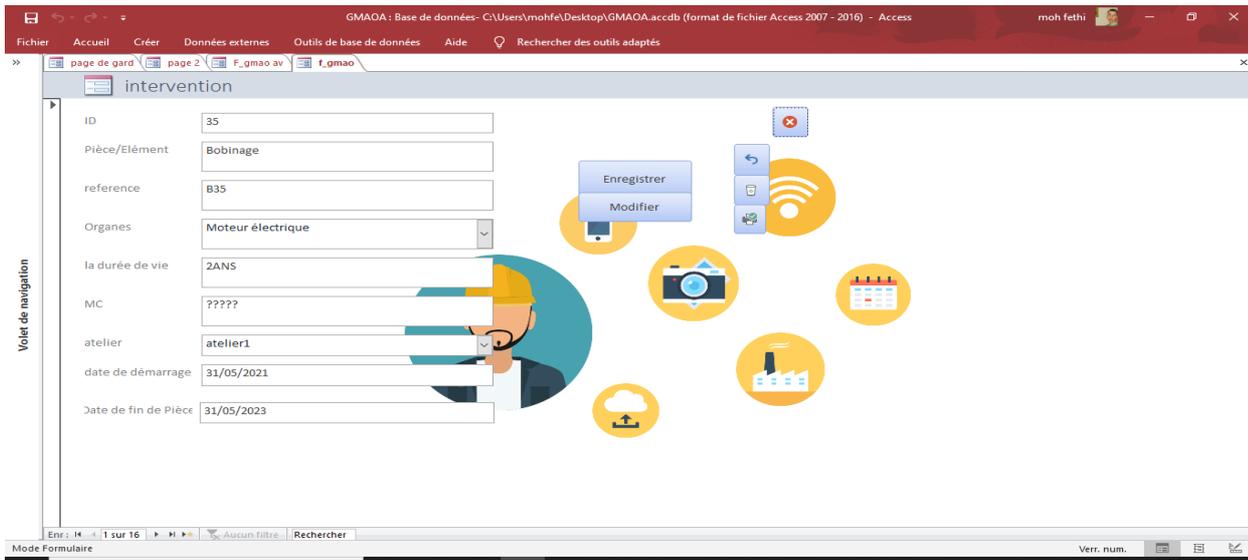


Figure4.12 : l’interface principale de GMAO

Une fois cliqué sur Conditionnel la figure4.13 s’affiche pour analyser les informations vibratoires et de température.

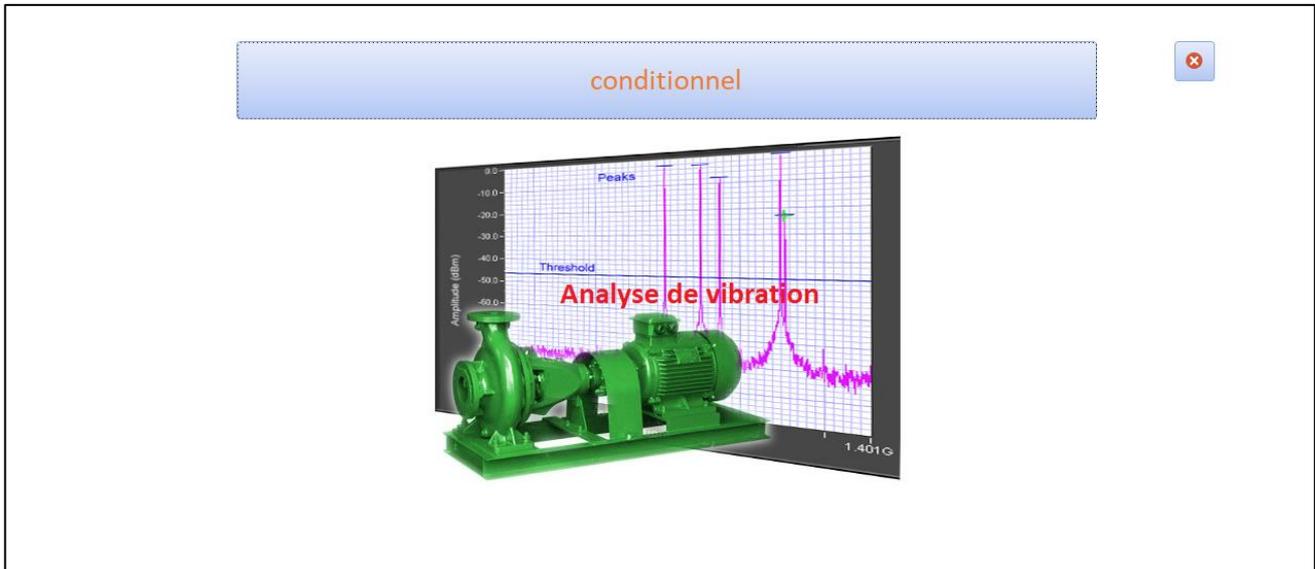


Figure4.13 : analyser les vibratoires et les températures

4.3 Réalisation de la partie de vibration et de température

Pour réaliser cette partie nous avons utilisé le capteur de Gyro-accéléromètre MPU 6050 et La carte Arduino Uno pour analyser l’information vibratoire et le capteur de température pour suivre et mesure la température d’un groupe motopompe.

4.3.1.1 Gyro-accéléromètre MPU 6050 :

Définition : Le gyro-accéléromètre 3 axes est un composant combinant un gyroscope 3 axes et un accéléromètre 3 axes, il est très utile pour détecter la vitesse angulaire pour pouvoir calculer l’angle de rotation. [21]

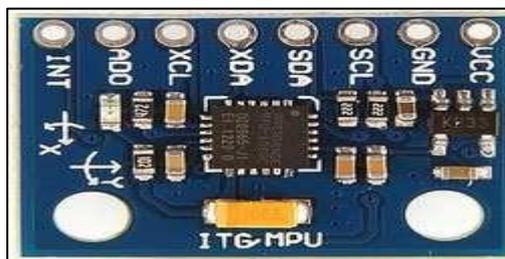


Figure 4.14 : le gyro-accéléromètre MPU6050[21]

Le gyroscope est un capteur de position angulaire, il donne la position angulaire selon deux ou trois axes de son référentiel par rapport à un référentiel inertiel (galiléen).

4.3.1.1.1 Caractéristiques techniques du gyro-accéléromètre 3 axes :

- Capteur de vitesse angulaire tri-axiale (gyroscope) avec une sensibilité allant jusqu’à 131 LSB/dps et une gamme a pleine échelle de +250, +500, +1000, et +2000dps.
- Accéléromètre tri-axiale pleine échelle programmable de +-2g, +-4g, +-8g, +-16g.
- Capteur de température à sortie numérique.
- Algorithmes embarqués pour la calibration de la boussole sans besoin d’intervention de l’utilisateur.

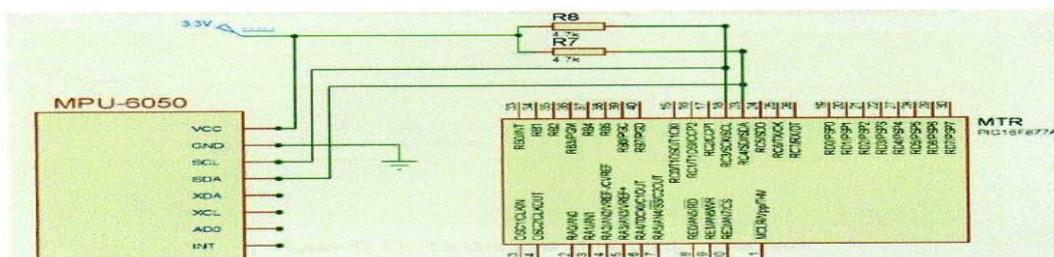


Figure 4.15: caractéristique technique du gyro-accéléromètre MPU 6050[21]

4.3.1.2 La carte Arduino Uno :

La carte Arduino Uno est le produit populaire parmi les cartes Arduino. Parfaite pour débiter la programmation Arduino, elle est constituée de tous les éléments de base pour construire des objets d'une complexité relativement faible.

La carte Arduino Uno, comme son nom l'indique, a été la première à utiliser la version de programmation Arduino 1.0, et elle est devenue le symbole de l'univers Arduino.

La carte Arduino Uno est constituée de 14 broches d'entrées/sorties digitales, dont six sont utilisables en PWM, de 6 broches d'entrées analogiques, d'une connectique USB, d'une connectique d'alimentation, d'un port ICSP et d'un bouton RESET. [13]

4.3.2 Structure générale du programme (IDE Arduino)

Comme n'importe quel langage de programmation, une interface souple et simple est exécutable sur n'importe quel système d'exploitation Arduino basé sur la programmation en C.

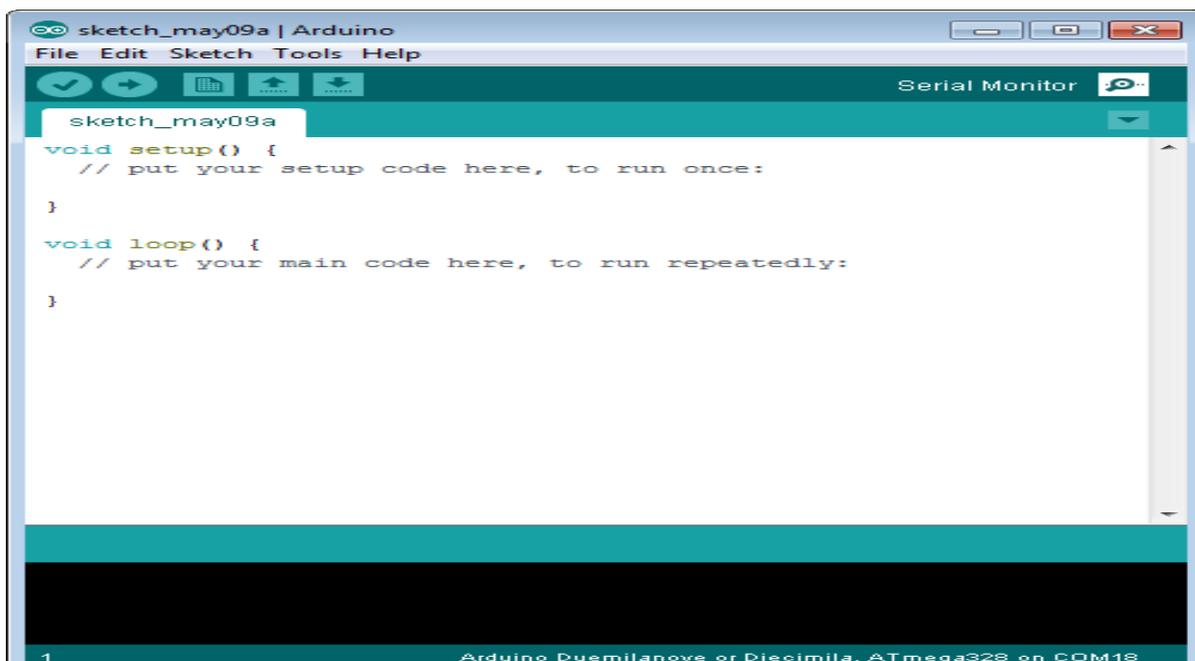


Fig.4.16 : L'IDE Arduino.

4.3.3 Câblage :

Pour le schéma de câblage et le profil de connexion, on connecte les broches des composants suivant :

Les différentes connexions et numéro des pins de circuit sont présentés par le tableau suivant :

Pin GY-521	Pin Arduino UNO
VCC	+5V
SCL	A5
SDA	A4
D2	INT
GND	GND

Tableau 4.4. : Broche et connexion du notre circuit électronique.

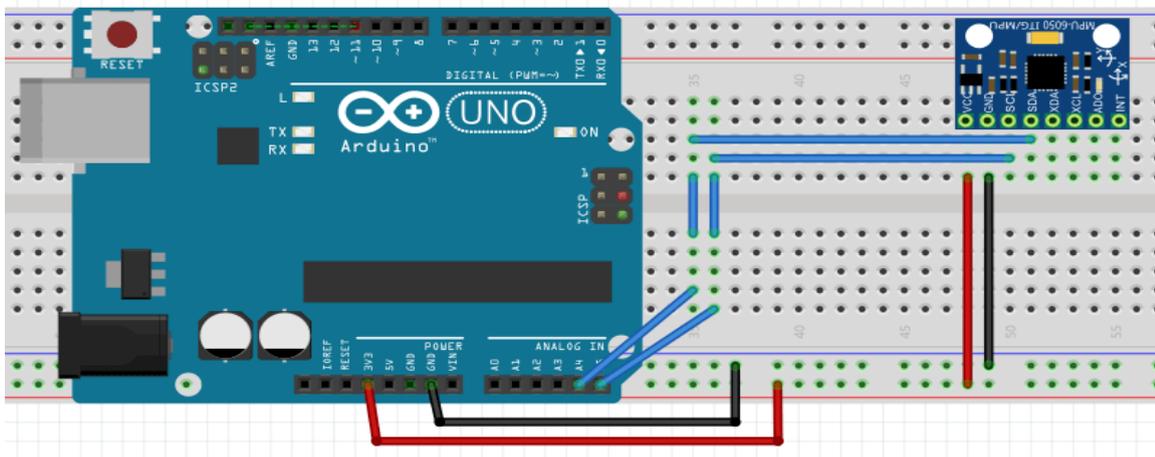


Fig.4.17 : Schéma de Circuit des connexions de la réalisation.

4.3.4 Démarche de programmation :

Tout d'abord, afin de tester le MPU 6050, on doit installer deux bibliothèques indispensables pour le fonctionnement de notre projet.

Bibliothèque MPU 6050 pour Arduino Pour interagir le module MPU6050 avec Arduino, Tout d'abord, il faut télécharger la librairie mpu6050 disponible sur internet, décompresser et copier

les deux fichiers nommés « MPU6050.LIB et MPU6050.IDX » et les placer dans le dossier des bibliothèques de notre logiciel Arduino.

Bibliothèque de I2C pour Arduino Pour assurer la communication entre l'Arduino et le module IMU mpu6050 on doit aussi télécharger la bibliothèque I2C nommé "I2Cdev.zip" disponible sur internet, qui présente trois fichiers nommés « I2C.IDX, I2CTEP.LIB, I2CTEP.HEX » Maintenant, on les place dans la bibliothèque de logiciel Arduino [22].

Alors maintenant, dans le dossier "librairies" d'Arduino, nous avons deux nouvelles entités Fig.4.18.

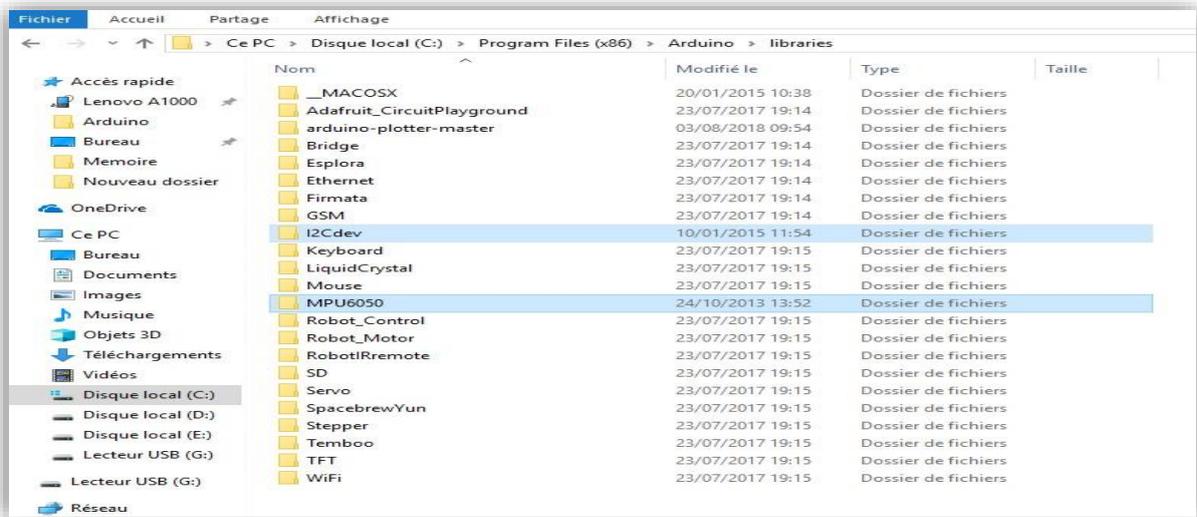


Fig.4.18 : Dossier " librairies " de Arduino

Ensuite on clique sur l'IDE Arduino pour voir si ces nouvelles bibliothèques sont visibles (Fig.4.19).

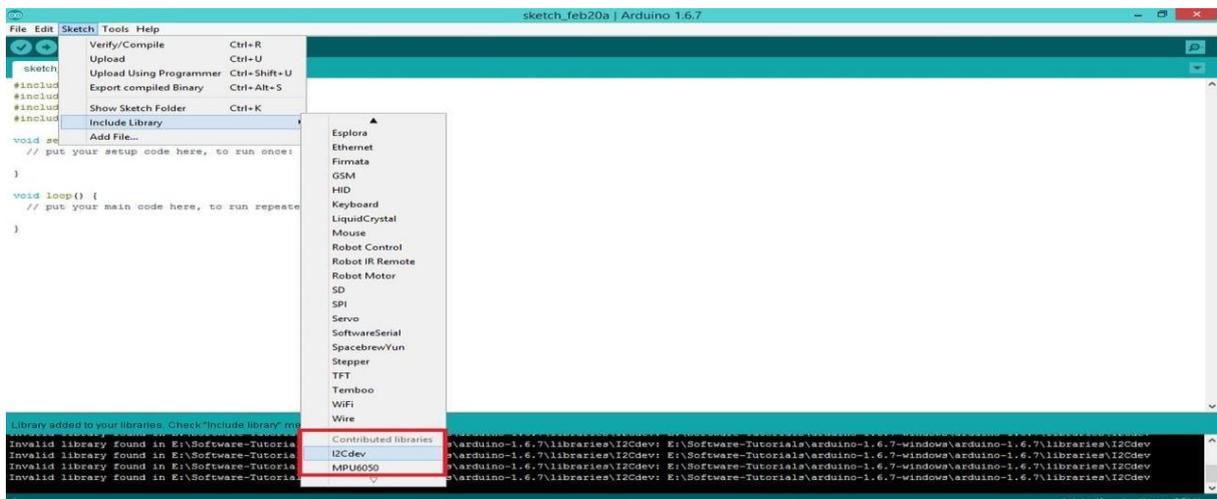


Fig.4.19 : les bibliothèques I2C et MPU6050 sous l'IDE Arduino.

Avant d'inclure ces bibliothèques dans notre sketch, on doit rétablir le code de configuration de MPU6050.

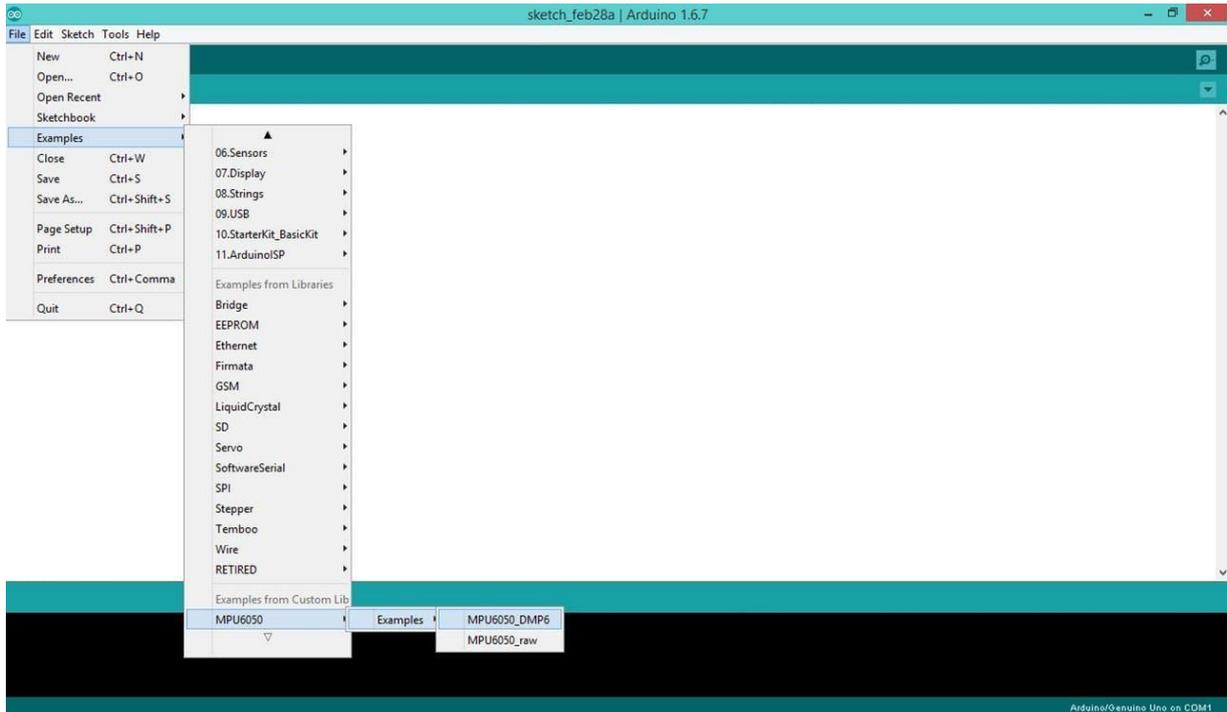


Fig.4.20 : La fenêtre de l'IDE Arduino pour rétablir le code de configuration.

Ensuite, on inclue les bibliothèques "I2Cdev" et "MPU6050" dans ce sketch.

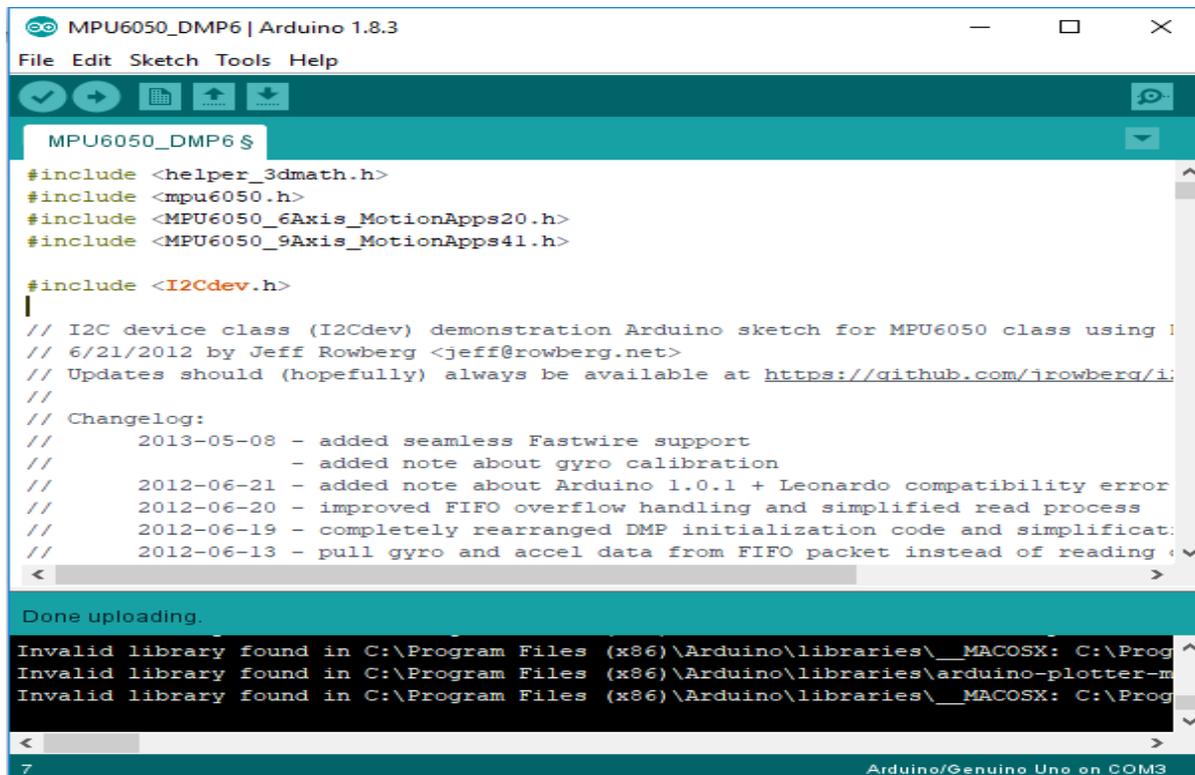


Fig.4.21 : l'inclusion des bibliothèques « I2Cdev » et « MPU6050 ».

Après avoir suivi toutes les étapes, on compile le sketch.

```

MPU6050_DMP6 | Arduino 1.8.3
File Edit Sketch Tools Help
MPU6050_DMP6
// I2C device class (I2Cdev) demonstration Arduino sketch for MPU6050 class using DMP (MotionApps v2.0)
// 6/21/2012 by Jeff Rowberg <jeff@rowberg.net>
// Updates should (hopefully) always be available at https://github.com/jrowberg/i2cdevlib
//
// Changelog:
//   2013-05-08 - added seamless Fastwire support
//               - added note about gyro calibration
//   2012-06-21 - added note about Arduino 1.0.1 + Leonardo compatibility error
//   2012-06-20 - improved FIFO overflow handling and simplified read process
//   2012-06-19 - completely rearranged DMP initialization code and simplification
//   2012-06-13 - pull gyro and accel data from FIFO packet instead of reading directly
//   2012-06-09 - fix broken FIFO read sequence and change interrupt detection to RISING
//   2012-06-05 - add gravity-compensated initial reference frame acceleration output
//               - add 3D math helper file to DMP6 example sketch
//               - add Euler output and Yaw/Pitch/Roll output formats
//   2012-06-04 - remove accel offset clearing for better results (thanks Sungon Lee)
//   2012-06-01 - fixed gyro sensitivity to be 2000 deg/sec instead of 250
//   2012-05-30 - basic DMP initialization working
//
// =====
// I2Cdev device library code is placed under the MIT license
// Copyright (c) 2012 Jeff Rowberg
//
// Permission is hereby granted, free of charge, to any person obtaining a copy
// of this software and associated documentation files (the "Software"), to deal
// in the Software without restriction, including without limitation the rights
// to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
// copies of the Software, and to permit persons to whom the Software is
  
```

Fig.4.22 : La compilation du sketch.

Après avoir compilé le sketch on aura cette fenêtre qui signifie que notre capteur MPU6050 est prête à être manipuler.

```

COM3 (Arduino/Genuino Uno)
2
Initializingsful

Send any character to begin DMP programming and demo:
Initializing DMP...
Initializing I2C devices...
Testing device connections...
MPU6050 connection successful

Send any character to begin DMP programming and demo:
  
```

Fig.4.23 : le test de fonctionnement de l'MPU6050.

On voit une ligne qui indique “Send any character to begin DMP programming and demo”.

Alors maintenant que nous avons configuré le matériel et logiciel, il est temps de compiler le programme sous Arduino IDE, et le téléverser vers la carte pour l’exécution via le câble USB. Après la compilation du code on aura les valeurs des degrés tridimensionnels qui sont Phi, Thêta, Psi suivant les axes Roulis (**R**oll), Tangage (**P**itch), Lacet (**Y**aw), sur MPU6050 en utilisant le moniteur série de l’IDE Arduino.

```

COM3
Temperature: 28.08 degC
Acceleration X: 1.03, Y: -0.28, Z: 9.23 m/s^2
Rotation X: -0.06, Y: 0.03, Z: -0.03 rad/s
Temperature: 28.08 degC
Acceleration X: 1.02, Y: -0.24, Z: 9.23 m/s^2
Rotation X: -0.06, Y: 0.03, Z: -0.03 rad/s
Temperature: 28.08 degC
Acceleration X: 1.01, Y: -0.28, Z: 9.23 m/s^2
Rotation X: -0.06, Y: 0.03, Z: -0.03 rad/s
Temperature: 28.08 degC
Acceleration X: 1.03, Y: -0.24, Z: 9.20 m/s^2
Rotation X: -0.06, Y: 0.03, Z: -0.03 rad/s
Temperature: 28.08 degC
Acceleration X: 1.02, Y: -0.24, Z: 9.22 m/s^2
Rotation X: -0.06, Y: 0.03, Z: -0.03 rad/s
Temperature: 28.08 degC
Acceleration X: 1.07, Y: -0.24, Z: 9.23 m/s^2
Rotation X: -0.09, Y: 0.02, Z: -0.03 rad/s
Temperature: 28.08 degC
Acceleration X: 1.03, Y: -0.28, Z: 9.22 m/s^2
Rotation X: -0.06, Y: 0.03, Z: -0.03 rad/s
Temperature: 28.08 degC
Acceleration X: 1.03, Y: -0.27, Z: 9.19 m/s^2
Rotation X: -0.06, Y: 0.03, Z: -0.03 rad/s
Temperature: 28.08 degC
Acceleration X: 1.02, Y: -0.27, Z: 9.22 m/s^2
Rotation X: -0.09, Y: 0.02, Z: -0.03 rad/s
Temperature: 28.08 degC
Acceleration X: 1.03, Y: -0.28, Z: 9.24 m/s^2
Rotation X: -0.06, Y: 0.03, Z: -0.03 rad/s
Temperature: 28.12 degC
Acceleration X: 1.02, Y: -0.28, Z: 9.22 m/s^2
Rotation X: -0.06, Y: 0.03, Z: -0.03 rad/s
Temperature: 28.13 degC
Acceleration X: 1.02, Y: -0.28, Z: 9.25 m/s^2
Rotation X: -0.06, Y: 0.02, Z: -0.03 rad/s
Temperature: 28.13 degC
Acceleration X: 1.01, Y: -0.28, Z: 9.20 m/s^2
Rotation X: -0.09, Y: 0.03, Z: -0.03 rad/s
Temperature: 28.13 degC
  
```

Fig.4.24 : Valeurs retournées par le module MPU6050.

Après avoir configuré notre capteur, on passe à l’étape de calibration comme suit :

4.3.4.1 Calibration du MPU-6050

La calibration est une étape indispensable pour avoir d’un bon résultat de calcul, il sert à fixer les références (0, 0,0) du module. Pour calibrer notre module, nous avons appliqué l’algorithme suivant :

- Mettre le module sur une plate horizontale.
- Mettre tous les offsets à 0
- Passé le programme à Arduino et nous ouvrons le moniteur en série pour voir lesvaleurs retournées.
- Laisser le module opérant pour quelques minutes (5-10) donc la température eststabilisée
- Vérifier les valeurs dans le moniteur en série et on note-les.
- Mettre à jours nos offsets et on passe le programme à nouveau à la carte.

- Répéter ce processus jusqu'à ce que notre programme retourne à zéro pour chaquegyros, à zéro pour l'accéléromètre suivant X et Y et +16384 suivant Z.

```

COM3 (Arduino Uno)

MPU6050 Calibration Sketch

Your MPU6050 should be placed in horizontal position, with package letters facing up.
Don't touch it until you see a finish message.

MPU6050 connection successful

Reading sensors for first time...

Calculating offsets...
...
...
...
...
...

FINISHED!

Sensor readings with offsets:  4      6    16390  0      0      -1
Your offsets:  2795  -728  1133  116  -14  -23

Data is printed as: accelX accelY accelZ gyroX gyroY gyroZ
Check that your sensor readings are close to 0 0 16384 0 0 0
If calibration was successful write down your offsets so you can set them in your projects using something similar to xpu.setXaccelOffset(youroffset)

 Autoscroll
    
```

Fig.4.25 : Résultat de la calibration.

Pour la figure ci-dessus, les décalages d'étalonnage sont atteints pour les 6 degrés de liberté ; les trois premiers décalages étaient pour l'accéléromètre (X, Y, Z) et les trois derniers pour le gyroscope.

L'échelle de sortie pour n'importe quel réglage est [-32768, +32767] pour chacun des six axes. Le paramètre par défaut dans la classe I2Cdevlib est +/- 2g pour l'accélération et +/- 250 deg / sec pour le gyroscope. Si l'appareil est parfaitement horizontal et ne bouge pas, on aura :

- Les axes X / Y doivent être 0
- L'axe Z accel doit indiquer 1g, soit +16384 pour une sensibilité de 2g
- Les axes gyroscopiques X / Y / Z doivent indiquer 0.

Ensuite, ces décalages résultants ont été pris et remplacés dans le code MPU6050, comme indiqué ci-dessous pour obtenir un résultat et une simulation précis. Ainsi, l'étalonnage est compensé par le code :

```
// supply your own gyro offsets here, scaled for min sensitivity
mpu.setXGyroOffset(116);
mpu.setYGyroOffset(-14);
mpu.setZGyroOffset(-23);
mpu.setXAccelOffset(2795);
mpu.setYAccelOffset(-728);
mpu.setZAccelOffset(1133); // 1688 factory default for my test chip
```

Fig.4.26 : Les résultats de calibration remplacée dans le code.

Pendant 10 à 20 secondes, on constate que la valeur du lacet augmente, puis, d'un coup, elle chute et fini par se stabiliser. Les roulis et tangage conserve leur valeur initiale pendant la même durée puis se stabilisent.

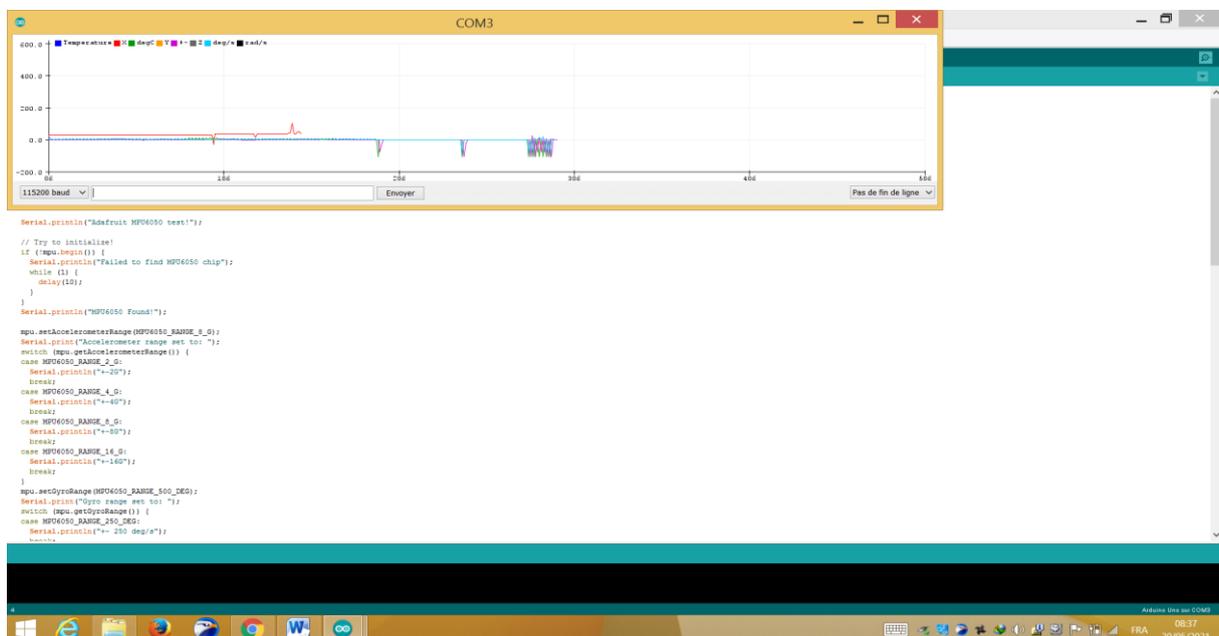


Fig.4.27 : la stabilité des mesures après calibration.

4.3.5 Prototype:

Nous avons maintenant ajouté une programmation pour les alertes de vibration et de température.

4.3.5.1 Le code source :

```
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
#include <Wire.h>

Adafruit_MPU6050 mpu;

void setup(void) {
  Serial.begin(115200);
  while (!Serial)
    delay(10);

  Serial.println("Adafruit MPU6050 test!");

  if (!mpu.begin()) {
    Serial.println("Failed to find MPU6050 chip");
    while (1) {
      delay(10);
    }
  }
  Serial.println("MPU6050 Found!");

  mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
  Serial.print("Accelerometer range set to: ");
  switch (mpu.getAccelerometerRange()) {
  case MPU6050_RANGE_2_G:
    Serial.println("+2G");
    break;
  case MPU6050_RANGE_4_G:
    Serial.println("+4G");
    break;
  case MPU6050_RANGE_8_G:
    Serial.println("+8G");
    break;
  case MPU6050_RANGE_16_G:
    Serial.println("+16G");
    break;
  }
  mpu.setGyroRange(MPU6050_RANGE_500_DEG);
  Serial.print("Gyro range set to: ");
  switch (mpu.getGyroRange()) {
  case MPU6050_RANGE_250_DEG:
    Serial.println("+ 25 deg/s");
    break;
  case MPU6050_RANGE_500_DEG:
    Serial.println("+ 50 deg/s");
    break;
  case MPU6050_RANGE_1000_DEG:
    Serial.println("+ 100 deg/s");
    break;
  case MPU6050_RANGE_2000_DEG:
    Serial.println("+ 200 deg/s");
    break;
  }
}
```

:

```

Serial.println(" ACCELERATION DÉPASSEACCELERATION . ");
}
/* Serial.print("Rotation X: ");
Serial.print(g.gyro.x);
Serial.print(", Y: ");
Serial.print(g.gyro.y);
Serial.print(", Z: ");
Serial.print(g.gyro.z);
Serial.println(" rad/s");*/
Serial.print("Temperature: ");
Serial.print(temp.temperature);
Serial.println(" degC");
if (temp.temperature>29){
Serial.print("Température dépasse.");
}

Serial.println("");
delay(1000);
}

```

Maintenant, l'application nous donne des alertes en cas de vibrations ou de température élevée.

```

-----
15:58.217 ->
15:59.239 -> Acceleration X: -0.41, Y: 0.17, Z: -9.88 m/
15:59.239 -> Temperature: 26.48 degC
15:59.239 ->
16:00.223 -> Acceleration X: -0.42, Y: 0.15, Z: -9.86 m/
16:00.270 -> Temperature: 26.52 degC
16:00.270 ->
16:01.254 -> Acceleration X: -0.42, Y: 0.16, Z: -9.88 m/
16:01.254 -> Temperature: 26.49 degC
16:01.254 ->
16:02.239 -> Acceleration X: -0.42, Y: 0.16, Z: -9.87 m/
16:02.239 -> Temperature: 26.51 degC
16:02.239 ->
16:03.270 -> Acceleration X: -3.67, Y: -1.39, Z: -9.58 m
16:03.270 -> Temperature: 26.52 degC
16:03.270 ->
16:04.254 -> Acceleration X: 10.05, Y: -3.45, Z: -10.70
16:04.254 -> ACCELERATION DÉPASSEACCELERATION .
16:04.254 -> Temperature: 26.52 degC
16:04.301 ->
16:05.285 -> Acceleration X: 10.75, Y: -5.40, Z: -10.82
16:05.285 -> ACCELERATION DÉPASSEACCELERATION .
16:05.285 -> Temperature: 26.54 degC
16:05.285 ->
16:06.270 -> Acceleration X: 6.18, Y: -7.43, Z: -10.42 m
16:06.317 -> ACCELERATION DÉPASSEACCELERATION .
16:06.317 -> Temperature: 26.56 degC
16:06.317 ->
16:07.301 -> Acceleration X: -0.48, Y: 0.08, Z: -9.86 m/
16:07.301 -> Temperature: 26.59 degC

```

Fig.4.28 : des alertes température élevée

```

17:17:59.077 -> Temperature: 29.04 degC
17:17:59.077 -> Température dépasse.
17:18:00.108 -> Acceleration X: -0.52, Y: 0.21, Z: -9.96 m/s^2
17:18:00.108 -> Temperature: 29.04 degC
17:18:00.108 -> Température dépasse.
17:18:01.093 -> Acceleration X: -0.53, Y: 0.21, Z: -9.97 m/s^2
17:18:01.093 -> Temperature: 29.08 degC
17:18:01.093 -> Température dépasse.
17:18:02.124 -> Acceleration X: -0.50, Y: 0.22, Z: -9.92 m/s^2
17:18:02.124 -> Temperature: 29.09 degC
17:18:02.124 -> Température dépasse.
17:18:03.108 -> Acceleration X: -0.53, Y: 0.21, Z: -9.99 m/s^2
17:18:03.108 -> Temperature: 29.10 degC
17:18:03.108 -> Température dépasse.
17:18:04.140 -> Acceleration X: -0.53, Y: 0.21, Z: -9.97 m/s^2
17:18:04.140 -> Temperature: 29.12 degC
17:18:04.140 -> Température dépasse.
17:18:05.124 -> Acceleration X: -0.51, Y: 0.22, Z: -9.98 m/s^2
17:18:05.124 -> Temperature: 29.13 degC
17:18:05.124 -> Température dépasse.
17:18:06.116 -> Acceleration X: -0.51, Y: 0.23, Z: -9.96 m/s^2
17:18:06.163 -> Temperature: 29.14 degC
17:18:06.163 -> Température dépasse.
17:18:07.147 -> Acceleration X: -0.51, Y: 0.22, Z: -9.98 m/s^2
17:18:07.147 -> Temperature: 29.14 degC
17:18:07.147 -> Température dépasse.
17:18:08.162 -> Acceleration X: -0.52, Y: 0.20, Z: -9.96 m/s^2
17:18:08.162 -> Temperature: 29.16 degC
17:18:08.162 -> Température dépasse.
17:18:09.147 -> Acceleration X: -0.52, Y: 0.22, Z: -9.94 m/s^2
17:18:09.147 -> Temperature: 29.17 degC
17:18:09.147 -> Température dépasse.
17:18:10.178 -> Acceleration X: -0.50, Y: 0.19, Z: -9.98 m/s^2
17:18:10.178 -> Temperature: 29.16 degC
17:18:10.178 -> Température dépasse.

```

Fig.4.29 : La compilation du sketch.

4.3.6 Le contrôle de température

La figure suivante présente le montage de capture de température DH22 avec l'Arduino, il suffit d'installer sur des endroits bien définis sur la motopompe électrique.

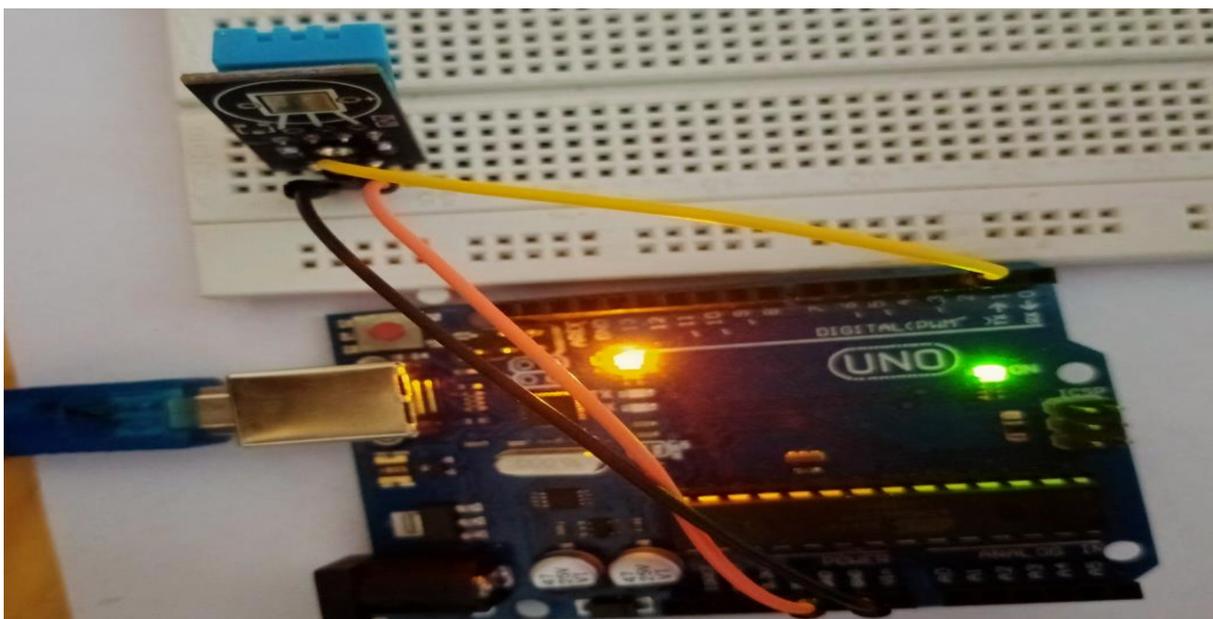
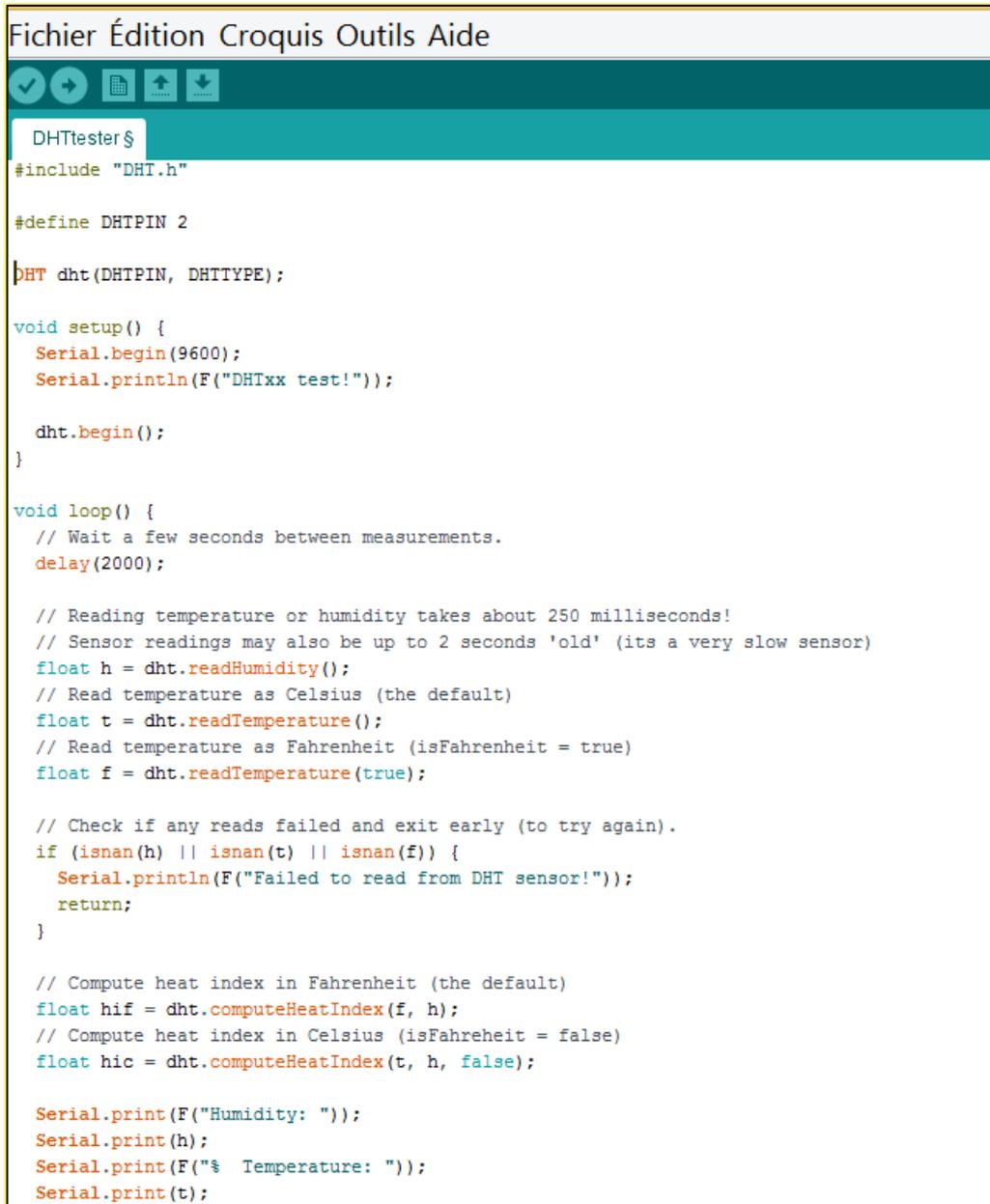


Figure4.30 : Programme de mesure de température et d'humidité

La programmation de capteur de température elle est disponible au niveau des bibliothèques d'Arduino, il suffit de l'adapter à nos besoins.

The image shows a screenshot of an Arduino IDE window titled "Fichier Édition Croquis Outils Aide". The window contains a C++ program for measuring temperature and humidity using a DHT sensor. The code includes the DHT.h library, defines the DHTPIN as 2, and initializes a DHT object. The setup function starts the serial port at 9600 baud and prints a test message. The loop function delays 2000ms between measurements, reads humidity and temperature, and prints the results. It also includes error handling for failed reads and heat index calculations.

```
DHTtester$
#include "DHT.h"

#define DHTPIN 2

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));

  dht.begin();
}

void loop() {
  // Wait a few seconds between measurements.
  delay(2000);

  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);

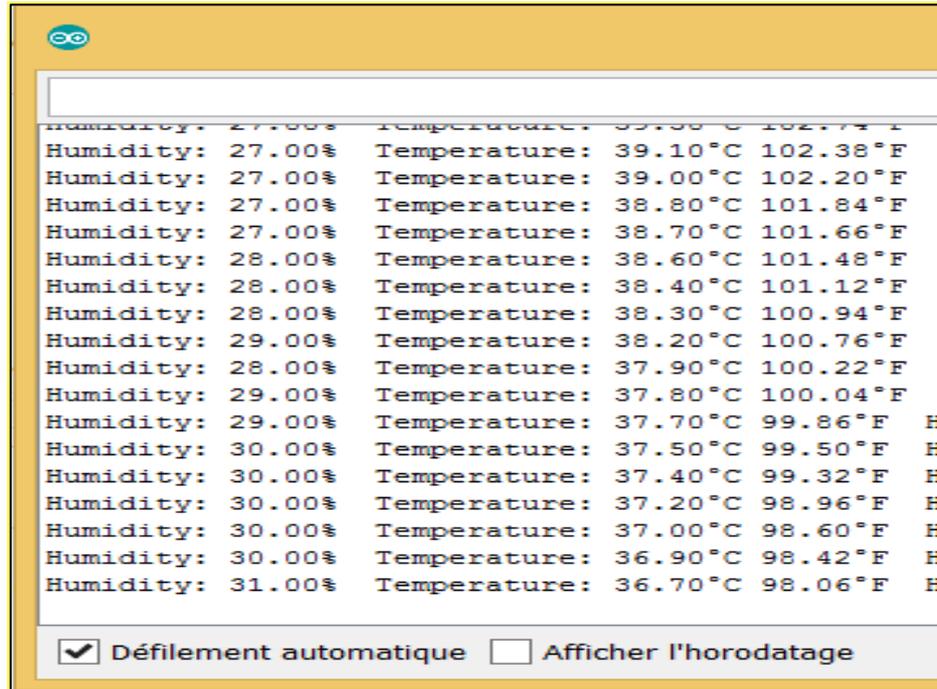
  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;
  }

  // Compute heat index in Fahrenheit (the default)
  float hif = dht.computeHeatIndex(f, h);
  // Compute heat index in Celsius (isFahreheit = false)
  float hic = dht.computeHeatIndex(t, h, false);

  Serial.print(F("Humidity: "));
  Serial.print(h);
  Serial.print(F("% Temperature: "));
  Serial.print(t);
```

Figure4.31 : Programme de mesure de température et d'humidité

La figure suivante présente le résultat d'exécution pour afficher les températures et les humidités en temps réel.



Humidity: 27.00%	Temperature: 39.10°C	102.38°F	
Humidity: 27.00%	Temperature: 39.00°C	102.20°F	
Humidity: 27.00%	Temperature: 38.80°C	101.84°F	
Humidity: 27.00%	Temperature: 38.70°C	101.66°F	
Humidity: 28.00%	Temperature: 38.60°C	101.48°F	
Humidity: 28.00%	Temperature: 38.40°C	101.12°F	
Humidity: 28.00%	Temperature: 38.30°C	100.94°F	
Humidity: 29.00%	Temperature: 38.20°C	100.76°F	
Humidity: 28.00%	Temperature: 37.90°C	100.22°F	
Humidity: 29.00%	Temperature: 37.80°C	100.04°F	
Humidity: 29.00%	Temperature: 37.70°C	99.86°F	H
Humidity: 30.00%	Temperature: 37.50°C	99.50°F	H
Humidity: 30.00%	Temperature: 37.40°C	99.32°F	H
Humidity: 30.00%	Temperature: 37.20°C	98.96°F	H
Humidity: 30.00%	Temperature: 37.00°C	98.60°F	H
Humidity: 30.00%	Temperature: 36.90°C	98.42°F	H
Humidity: 31.00%	Temperature: 36.70°C	98.06°F	H

Défilement automatique
 Afficher l'horodatage

Figure4.32 : résultat de température et l'humidité

4.4 Conclusion :

Ce travail consiste à améliorer le processus de Système de Maintenance par l'intégration de l'information vibratoire dans un système de maintenance préventive, pour cela nous avons utilisé langage UML pour la conception et la modélisation de système de maintenance, nous avons pris un groupe motopompe électrique comme étude de cas de son système de maintenance, et nous avons utilisé SGBD Microsoft Access pour l'implémentation de GMAO, pour l'analyse de l'information vibratoire nous avons utilisé l'Arduino avec les capteurs de vibrations en particulière le capteur de Gyro-accéléromètre MPU6050, pour l'analyse de température nous avons utilisé capteur DH11. Le système de GMAO amélioré que nous avons développé peut-être utilise avec n'importe quel système industriel et des machines pour la gestion de la maintenance.

CONCLUSION GÉNÉRALE

Conclusion générale :

Ce travail consiste à améliorer le processus de Système de Maintenance par l'intégration de l'information vibratoire dans un système de maintenance préventive systématique. L'amélioration de la GMAO peut intervenir par rapport d'une part la maintenance préventive. Et d'autre part, par rapport à la maintenance conditionnelle. Alors que les deux types de maintenance préventive (systématique et conditionnelle) pour but d'éviter l'intervention de la maintenance corrective c'est-à-dire les problèmes d'arrêts des systèmes et les pannes.

Pour cela la maintenance préventive joue un rôle très important dans n'importe quelle structure organisationnelle d'une entreprise industrielle afin de réduire les coûts et les dépenses et aussi pour être concurrente. L'objectif de ce travail est la réalisation d'une GMAO améliorée qui inclut toutes les informations de la maintenance systématique et conditionnelle. La mise en œuvre de la GMAO améliorée inclut : Des informations sur le niveau vibratoire et des informations sur la température, des informations sur la maintenance préventive et l'historique des interventions. Ces informations permettant de comprendre et d'analyser puis protéger bien le système industriel.

La GMAO améliorée a été appliquée sur une étude de cas d'un groupe motopompe électrique. Nous avons utilisé SGBD Microsoft Access pour l'implémentation de GMAO, et pour l'analyse de l'information vibratoire nous avons utilisé l'Arduino avec les capteurs de vibrations en particulier le capteur de Gyro-accéléromètre MPU6050 et pour analyse la température nous avons utilisé le capteur DH22.

La GMAO améliorée que nous avons développée peut-être utilisée dans n'importe quel système industriel .

Avec ce travail de master nous avons enrichi nos connaissances sur plusieurs disciplines comme le domaine de développement des applications informatiques pour la maintenance et le développement des systèmes embarqués.

Ce travail ouvre d'autres perspectives, premièrement au niveau d'installation de plusieurs capteurs dans différents endroits bien définis dans un groupe motopompe électrique pour bien analyser les vibrations et localiser les défauts et ainsi dans la partie de développement que nous recommandons la continuation des développements la GMAO puis les tests unitaires de génie logiciel sur des entreprises dans la zone industrielle de Tiaret qui ne sont pas encore installées GMAO.

**RÉFÉRENCES
BIBLIOGRAPHIQUES**

Bibliographies :

- [1] BOULENGER, Vers la zéro panne avec la maintenance conditionnelle, AFNOR, PARIS 1998.
- [2] M. Soussan, T. Dib, Etude critique et propositions d'amélioration de la gestion de la maintenance cas de l'adduction EL KANSERA, Mémoire d'ingénieur d'état, 2012/2011.
- [3] D. richet, Maintenance basée sur la fiabilité : un outil pour la certification, Ed. Masson.1996.
- [4] Nicolas Buchy « La gestion de la maintenance assistée par ordinateur et la maintenance des logiciels » www.lrgl.uqam.ca. 2002.
- [5] Jean-Pierre Vernier. « Maintenance et GMAO : Tableaux de bord, organisation et procédures Reliées », 2010.
- [6] LandolsiFoued, « Cours De Techniques de Surveillance, Partie I Surveillance Des Machines Par Analyse Vibratoire ».
- [7] A. BOULENGER, C. PACHAUD :« Diagnostic vibratoire en maintenance préventive » (Dunod, Paris, octobre 1999).
- [8] Omar DJEBILI ; « Contribution à la maintenance prédictive par analyse vibratoire des composants mécaniques tournants. Application aux butées à billes soumises à la fatigue de contact de roulement ». Thèse de doctorat, Université De Reims Champagne Ardenne (2013).
- [9] C. ALONSO, B. Estivals, livre “ Conception et Commande de Systèmes Electriques Embarqués “Université –Toulouse, France 2002.
- [10] Shibu KV, "Introduction to Embedded System", 1er Edition, McGraw-Hill, 2009.
- [11] W. Thurber, "Une méthodologie de modélisation multi-modèles distribuée par Métier pour les systèmes embarqués", Thèse de Doctorat, ENSAE de Toulouse, France, 2006.
- [12] Cours systèmes embarqués : Introduction", document internet disponible à : [www.technologuepro.com/cours-systemes-embarques/ cours-systemes- Embarques introduction.html](http://www.technologuepro.com/cours-systemes-embarques/cours-systemes-Embarques-introduction.html).
- [13] A. KOUACHE, "Etude et développement d'un système flou reconfigurable ; Application : Système embarqué de ventilation d'un tunnel", Thèse de Master, Université Ferhat Abbas – Setif1, 2015.
- [14] MEMOIRE 'GHEMARI ZINE ' "Modélisation Et Simulation D'un Capteur De Vibration (Accéléromètre) {Université Annaba. Année 2005 / 2006}.
- [15] MEMOIRE' “KOUCHIH Mourad et BOUCHOUKA Hocine”“, Maintenance des Machines Tournantes par Analyse Vibratoire 'Université TIARET.2017
- [16] <https://www.carnetdumaker.net/articles/utiliser-un-capteur-de-temperature-et-dhumidite-dht11-dht22-avec-une-carte-arduino-genuino/> [10/105/2021]
- [17] MEMOIRE“ MECHTA Soumia et EL-EUDJAMA Marwa“ étude et réalisation d'une maquette didactique pour analyse vibratoire des machines tournantes en mesure globale, Université TIARET 2020
- [18] MEMOIRE' “KOUCHIH Mourad et BOUCHOUKA Hocine”“, Maintenance des Machines Tournantes par Analyse Vibratoire' Université TIARET2017 .
- [19] MEMOIRE “ HACHEMI Mohammed, “ La maintenance des équipements par L'analyse vibratoire ‘Université Constantine 2012.

[20] MEMOIRE CHAIB Sid ali et BOUSSAIDI Azeddine, détection et caractérisation des défauts des roulements par l'analyse spectrale 'Université Ferhat Abbas 2016.

[21] InvenSense Inc, Document Number: PS-MPU-6000A-00 , «MPU-6000 and MPU-6050 Product Specification» , 1197 Borregas Ave, Sunnyvale, CA 94089 U.S.A,08/19/2013 ,Revision: 3.4 ,52 pages

[22] Embeddeb Adventures, 2013 Embedded Adventures, WRL-3000 V2, 2013, [en ligne], date de dernière mise à jour :27juin 2013, Disponible sur :
http://www.embeddedadventures.com/datasheets/WRL-3000_hw_v2_doc_v1.pdf ,13juin 2014.

[23] Pascal R (2006) « UML 2 par la pratique Etude de cas et exercices corrigés » (Livre), Editions Eyrolles Edition. ISBN: 2-212-12014-1. 2006.

[24] <https://www.arduino.cc/>. [10/105/2021]

[25] <https://www.arduino.cc/en/Main/ArduinoBoardUno>. [10/105/2021]

[26] www.arduino.cc/en/Main/Software. [10/105/2021]

[27] <https://www.adafruit.com>[10/105/2021]

[28] <http://www.instructables.com/id/Sending-data-from-Arduino-to-Excel-and-plotting-it>[10/105/2021]

[29] <https://www.mathworks.com/videos/using-matlab-and-arduino-to-acquire-analog-signals-100739.html>. [10/105/2021]

[30] <https://www.google.com/sheets/about/>[10/105/2021]

[31] <https://zestedesavoir.com/tutoriels/686/arduino-premiers-pas-en-informatique-embarquee/>[10/105/2021]

Résumé :

La maintenance préventive (systématique et conditionnelle) joue un rôle important dans n'importe quelle entreprise afin de réduire le coût de dépenses, d'éviter la maintenance corrective, et d'être concurrente, vu que l'électronique embarquée actuellement intégrer jour après jour dans notre vie quotidienne, dans les bâtiments pour rendre plus intelligent, dans les voitures pour aider à la conduite, dans les supermarchés, etc.). L'objectif de ce sujet de master est l'étude et l'amélioration d'un système de gestion de la maintenance assisté par ordinateur (GMAO) en utilisant les systèmes embarqués en particulier les capteurs de vibrations, et les capteurs de température.

-Pour réaliser ce travail nous avons choisi un groupe motopompe comme étude de cas.

Mots clés : GMAO, l'électronique embarquée, la maintenance systématique, la maintenance conditionnelle, Le capteur de vibration, le capteur de température.

ملخص:

تلعب الصيانة الوقائية دورًا مهمًا في أي شركة من أجل تقليل تكلفة النفقات ونظرًا لاندماج الأجهزة الإلكترونية يوميًا بعد يوم في حياتنا اليومية، في المباني لتصبح أكثر ذكاءً، في السيارات للمساعدة في القيادة وما إلى ذلك. الهدف من موضوع هذا الماستر هو دراسة وتحسين نظام إدارة صيانة الكمبيوتر بواسطة مستشعرات الاهتزاز في المحركات والمضخات في القطاع الصناعي لتحسين الصيانة التنبؤية. لتنفيذ هذا العمل اخترنا وحدة ضخ الماء.

summary:

Today's on-board electronics are integrating day after day into our daily lives, in buildings to make smarter, in cars to aid driving, in supermarkets, in human beings, etc.). The aim of this master is the study and improvement of a computer-assisted maintenance management system (GMAO) using on-board systems in particular vibration sensors in motors, pumps, compressors in the industrial sector to improve predictive maintenance.

-To carry out this work we have chosen a motor pump unit as a case study.