

REPUBLIQUE DU SENEGAL
UNIVERSITE CHEIKH ANTA DIOP DE DAKAR



Gm.0040

ECOLE SUPERIEURE POLYTECHNIQUE
CENTRE DE THIES
DEPARTEMENT GENIE MECANIQUE, OPTION ELECTROMECHANIQUE

Projet de Fin d'Etudes

*En vue de l'obtention du Diplôme d'Ingénieur de Conception
en Génie Mécanique, Option Electromécanique*

TITRE :

**ETUDE ET MODERNISATION DU SYSTEME
D'ACQUISITION ET DE TRAITEMENT DE
DONNEES D'UNE UNITE DE TURBINES
HYDRAULIQUES**

Présenté par : Mlle. Ndèye Ndiaye CISSE

Directeur interne : M. Salam SAWADOGO

Juillet 2003

Directeur externe : M. Ousmane CISSE

DEDICACES

Je dédie ce travail particulièrement :

A ma mère, pour ses prières et ses encouragements qui m'ont donné la force d'aller toujours de l'avant.

A mon père, qui a toujours su placer sa confiance en moi et m'a inculqué le culte de la rigueur, de la persévérance et du sens des devoirs, quoiqu'il puisse nous en coûter.

A toute ma famille qui m'est chère.

A tous mes amis.

SOMMAIRE

Le but de ce travail consistait à étudier le système d'acquisition et de traitement de données, utilisé pour piloter l'installation d'une unité de turbines hydrauliques, et de le moderniser afin de conférer aux travaux pratiques sur machines hydrauliques un grand degré d'automatisation des tâches.

La difficulté, qui a rendu le projet vaste, fut l'obligation d'une progression dans un environnement indéterminé où toutes les données devaient, soit être recherchées sur le terrain, soit en obtenir confirmation par une confrontation avec celles trouvées dans la documentation. Ensuite seulement, les câblages électroniques devaient se faire et l'ensemble des programmes informatiques s'écrire, sur la base des données enregistrées lors de travaux pratiques antérieurs.

Les tests réalisés sur l'installation, en utilisant le nouveau système d'acquisition mis en place, ont donné des résultats positifs quant à la bonne marche du système réalisé en parallèle avec l'ancien. Mais, de tous les paramètres, seule la vitesse de rotation varie de façon normale et conformément à la rotation réelle du rotor. Les paramètres de pressions et de force restent fixent ou varient de façon négligeable.

Les tests ont pu ainsi confirmer l'hypothèse selon laquelle il y aurait une défection du matériel de l'installation. Cette dernière, en ne permettant pas d'enregistrer les données adéquates, rend impossible le déroulement de la partie des travaux pratiques concernant les calculs, les tracés des graphiques et l'interprétation des résultats. L'alternative proposée serait de recourir à tous les moyens, dans la mesure du possible, afin d'effectuer une réparation de l'installation, gage de la pleine utilisation des capacités du nouveau système d'acquisition de données.

Mots clés : turbines – capteurs – carte d'acquisition de données CIO-DAS08-PGH – logiciel d'acquisition et d'affichage, le Labtech – programmation informatique en VBA.

REMERCIEMENTS

J'adresse mes sincères remerciements à tous ceux qui ont contribué à la réalisation de ce travail :

Monsieur Séni Tamba, professeur rattaché au Département de Génie Civil, pour avoir bien voulu me proposer un sujet de fin d'études enrichissant d'expériences.

Monsieur Salam Sawadogo, professeur rattaché au département de Génie Mécanique, pour avoir fait preuve d'un total engagement dans son rôle d'encadreur interne du projet.

Monsieur Ousmane Cissé, ancien étudiant de l'école actuellement en formation D.E.S.S. en France, pour avoir accepté d'être mon directeur externe et sut me conseiller même si ce n'était pas évident avec la distance.

Mention spéciale à **Monsieur Masseck Fall**, technicien responsable du laboratoire d'hydraulique, pour sa grande disponibilité et ses utiles informations qui ont grandement facilité ce travail.

Un merci aussi à **Madame Gaye**, responsable du laboratoire de chimie, pour avoir facilité les accès au laboratoire d'hydraulique.

Enfin, je tiens à remercier, tous ceux qui, de près ou de loin, ont participé à la réalisation de ce travail de fin d'études.

TABLE DES MATIERES

	<u>pages</u>
LISTE DES ANNEXES.....	VIII
LISTE DES TABLEAUX.....	IX
LISTE DES FIGURES.....	X
LISTE DES SYMBOLES ET DES ABREVIATIONS.....	XII
INTRODUCTION	1
I. ENONCE SUR LES SYSTEMES D'ACQUISITION ET DE TRAITEMENT DE DONNEES.....	4
I.1 Notions générales.....	4
I.2 Les capteurs et leurs conditionneurs.....	7
I.2.1 Les capteurs analogiques.....	8
I.2.2 Les capteurs logiques.....	9
I.2.3 Les capteurs numériques.....	9
I.2.4 Les capteurs digitaux.....	9
I.3 Les convertisseurs Analogiques/Numériques (CAN)	9
I.3.1 La période d'échantillonnage.....	11
I.3.2 La résolution.....	11
I.3.3 Les entrées et sorties digitales.....	11
I.3.4 Les sorties analogiques.....	12

II. L'INSTALLATION HYDRAULIQUE.....	13
II.1 Description et Principe de fonctionnement	13
II.2 Les capteurs et leurs conditionneurs.....	15
III. LIBELLE DU CAHIER DES CHARGES.....	17
IV. EXPLICATIONS SUR LE TYPE DE SOLUTION ADOPTEE POUR LA RESOLUTION PRATIQUE DU PROBLEME	19
V. SOLUTION DE CABLAGE DES APPAREILS ELECTRONIQUES.....	21
VI. LA CARTE D'ACQUISITION DE DONNEES CIO-ADO8-PGH.....	23
VI.1 Introduction.....	23
VI.2 Connexions des signaux.....	24
VI.2.1 Le diagramme de connexions.....	24
VI.2.2 Entrées différentielles.....	25
VI.2.3 Entrées et sorties digitales.....	26
VI.3 Installation de la carte dans l'ordinateur.....	26
VI.4 Installation de la carte point de vue logiciel.....	26
VI.4.1 L'adresse de base.....	28
VI.4.2 Sélection du niveau d'interruption.....	28
VI.4.3 Sélecteur d'horloge.....	29
VI.4.4 Etat d'attente.....	30
VI.5 Tests de vérification.....	31

VII. CABLAGE DES CAPTEURS.....	33
VII.1 Principe du montage diviseur de tension	33
VII.2 Principe du montage amplificateur opérationnel.....	34
VII.3 Principe du montage régulateur de tension.....	35
VII.4 Solution adoptée.....	37
VIII. LE LOGICIEL D'ACQUISITION ET D'AFFICHAGE : LE LABTECH....	40
VIII.1 Le logiciel Labtech Notebook.....	40
VIII.1.1 Le système de menus.....	40
VIII.1.2 Le système d'icônes.....	41
▪ Les icônes.....	42
▪ Les types d'icônes.....	44
VIII.2 Le logiciel Real Time Vision.....	45
VIII.2.1 Système requis.....	45
VIII.2.2 La fenêtre de Vision.....	46
VIII.2.3 Les objets de Vision.....	46
▪ Les objets de visualisation.....	47
▪ Les objets de contrôle.....	47
▪ Les objets texte.....	48
▪ Les objets dessin.....	48
▪ Les objets d'animation.....	49

IX. PROGRAMMATION GRAPHIQUE DE L'INSTALLATION ET TESTS	
SUR LES DONNEES A ACQUERIR.....	50
IX.1 Programmation graphique pour l'acquisition de données.....	50
IX.2 Tests	51
X. DEVELOPPEMENTS SUR EXCEL	53
X.1 Enregistrement des données sous Excel et mises en forme.....	53
X.2 Calcul des paramètres permettant le tracé des graphiques.....	56
X.3 Tracé des graphiques.....	59
XI. DETERMINATION DE L'ETAT DE MARCHE DU MATERIEL	
COMPOSANT L'UNITE DE TURBINES.....	61
CONCLUSION.....	63
BIBLIOGRAPHIE.....	65
ANNEXES	67

LISTE DES ANNEXES

Annexes de la partie programmation :

AP1 : Programme écrit dans le langage VBA.....	68
AP2 : Exemples de graphiques tracés par le programme grâce à l'utilisation de données simulées.....	88
AP3 : Exemple de données enregistrées lors des tests sur l'unité de turbines.....	90

LISTE DES TABLEAUX

<u>Tableau V.1:</u> Signal délivré par chaque fil dans un câble.....	21
<u>Tableau XI.1:</u> Résultats des tests réalisés sur les turbines.....	61

LISTE DES FIGURES

Figure I.1 : Exemple détaillé d'un ensemble mesure-contrôle de procédé.....	6
Figure I.2 : Structure générale d'une chaîne d'acquisition de données.....	10
Figure II.1 : Schématisation de l'installation hydraulique.....	14
Figure VI-1 : La carte CIO-AD08-PGH et ses composants.....	24
Figure VI.2 : Diagramme de connexions du CIO-AD08-PG.....	25
Figure VI.3 : Les menus du programme « Instacal ».....	27
Figure VI.4 : Commutateur d'adresses de base.....	28
Figure VI-5 : Sélecteur de niveau d'interruption.....	29
Figure VI-6 : Sélecteur d'horloge.....	29
Figure VI-7 : Sélecteur d'état.....	30
Figure VI.8 : Configuration graphique de la carte après son Installation.....	31
Figure VI.9 : Ecran montrant la connexion pour le test de la voie d'entrée 0.....	32
Figure VII.1 : schématisation du circuit diviseur de tension.....	34
Figure VII.2 : montage non-inverseur.....	35
Figure VII.3 : circuits et symbolisation des régulateurs de tension.....	37
Figure VII.4 : schéma de câblage des régulateurs de tension.....	38
Figure VII.5 : schéma de câblage des capteurs.....	39
Figure VIII.1 : Atlas du menu de Notebook.....	41
Figure VIII.2 : l'interface graphique de Labtech Notebook.....	42
Figure VIII.3 : Fenêtre de configuration d'un bloc.....	43

Figure VIII.4 : Les différents types de blocs existant dans le Labtech.....	45
Figure VIII.5 : Fenêtre de visualisation de Real Time Vision.....	46
Figure VIII.6 : Les objets de visualisation.....	47
Figure VIII.7: Les objets de contrôle.....	48
Figure VIII.8 : Les objets dessin.....	48
Figure IX.1 : Résultats de la programmation graphique sur labtech avec en arrière- plan le schéma réalisé sur le « Vision ».....	50
Figure IX.2 : Fenêtre affichée lors de l'acquisition.....	51
Figure X.1: Fenêtre d'affichage des données enregistrées depuis le labtech et mises en forme (simulation à partir de la carte de démonstration).....	55
Figure X.2: Fenêtre d'affichage des données calculées (réalisée à partir d'une simulation).....	59

LISTE DES SYMBOLES ET DES ABREVIATIONS

Alpha = coefficient de débit = 0.6300

d = diamètre de l'orifice = 0.0090

DACO = Analog output channel 0

DAC1 = Analog Output channel 1

D.O.S = Disk Operating System

dPo = Pression différentielle de l'orifice (Pa)

ETA_t = Rendement de la turbine (%)

F_b = Force du frein (N)

g = Pesanteur = 9.8100

H_i = Charge à l'entrée (m)

N = Vitesse de rotation (Hz)

P_b = Puissance mécanique (puissance au frein) (W)

P_h = Puissance hydraulique (W)

P₁ = Pression à l'entrée du rotor (Pa)

Q_m = débit massique (kg/s)

Q_v = débit volumique (m³/s)

r = rayon de poulie = 0.0240

R_{how} = densité de l'eau = 998.20001

SIG = Signal

T = Couple (mNm)

T.P. = Travaux Pratiques

VBA = Visual Basic For Applications

INTRODUCTION

La technologie, en ce dernier siècle, a connu une avancée fulgurante dans tous les domaines, de la pure mécanique à la médecine avec notamment des systèmes de surveillance et de sécurité toujours plus performants.

Cette avancée a pu être conservée et améliorée grâce notamment au développement toujours plus pointu des systèmes de contrôle des processus associés. De simples appareils de mesure tels les potentiomètres, on est passé aujourd'hui, grâce aux progrès énormes dans le domaine de l'électronique et de l'informatique, à des appareillages cent fois plus légers et à des processus de mesure cent fois plus courts, pour une même précision. On peut notamment multiplier le nombre de points de mesure et agir directement en temps réel, sans l'intervention de l'homme, dans les dispositifs contrôlés.

De tels systèmes appelés « Systèmes d'Acquisition et de Traitement de données » ont conquis aujourd'hui leur place dans l'univers technologique.

Ainsi, dans les écoles d'ingénieurs, ils sont également utilisés dans les équipements pour des travaux pratiques tels les mini-installations de machines hydrauliques.

Récemment, en effet, les problèmes d'espace, de temps et d'argent pouvant être consacrés aux manipulations dans les laboratoires, ont conduit à choisir cette alternative qui permet, non seulement de simuler les mêmes performances que les grosses machines industrielles, mais aussi à éliminer l'aléa d'une prise répétitive de valeurs. Ceci donne à l'étudiant l'opportunité, de visualiser en temps réel les changements des paramètres du système étudié et de se consacrer à l'analyse des résultats, dans des délais raisonnables.

Aussi, dans le laboratoire d'hydraulique de l'Ecole Supérieure Polytechnique de Thiès, se trouve installé un banc d'essai pour turbines hydrauliques (type action, réaction et impulsion) relié à un Système d'Acquisition et de Traitement de données.

Mais ce dernier, avec un fonctionnement sous un environnement D.O.S, présente une incommodité dans l'exploitation par rapport aux avancées opérées récemment dans ce domaine. Il faudrait alors réaliser une modernisation du système, permettant sa réhabilitation, en une nouvelle installation hydraulique pérenne, conviviale et facile d'exploitation avec une utilisation plus commode, sans oublier, la minimisation des coûts par rapport à l'éventuelle acquisition d'une nouvelle unité. Toutes les capacités de l'ancien système devront être reconduites et notamment améliorées.

La résolution du problème passera alors par la réalisation d'un nouveau système pouvant permettre l'utilisation d'un logiciel d'acquisition et de traitement de données garantissant la réalisation d'une programmation sous l'environnement Windows et de type essentiellement graphique.

Ce rapport porte alors sur l'étude de l'installation hydraulique du point de vue mécanique, électrique et électronique, puis sur sa modernisation avec une description des solutions et méthodes utilisées dans la conception et la réalisation dudit nouveau système, des capteurs à l'ordinateur.

L'étude comparative sera la principale méthodologie utilisée, avec la conception d'un système parallèle à celui existant déjà, pour visualiser et comparer directement les possibilités et les limites de l'un et l'autre des deux systèmes, lors de la résolution pratique du problème.

Il va s'agir alors :

- ❖ de mettre en place le nouveau système, parallèle à l'existant, grâce au câblage d'appareils électroniques (capteurs, conditionneurs, ordinateur...);
- ❖ de travailler sur une version téléchargée d'un logiciel d'acquisition et de traitement de données pour la réalisation d'un programme graphique de type moderne plus performant que celui existant ;

- ❖ d'acquérir le logiciel pouvant permettre l'exploitation continue du programme graphique réalisé sur la version téléchargée du logiciel.

I. ENONCE SUR LES SYSTEMES D'ACQUISITION ET DE TRAITEMENT DE DONNEES

I.1 Notions générales

Une chaîne d'acquisition est chargée de recueillir les informations nécessaires à la connaissance et au contrôle d'un procédé donné ; et ces informations recueillies doivent ensuite être délivrées sous une forme appropriée à leur traitement et à leur exploitation.

En effet, l'état d'un procédé est, à chaque instant, caractérisé par les valeurs d'un certain nombre de grandeurs physiques ou / (et) chimiques : ces valeurs sont les informations que le système d'acquisition doit fournir. Et l'assignation d'une valeur à une grandeur physique ou chimique est une opération de mesure et la grandeur objet de la mesure est appelée le mesurande.

Le mesurande est caractérisé par son intensité et par son évolution temporelle et ce sont ces informations que, pour chaque mesurande, la chaîne d'acquisition doit pouvoir délivrer sous la forme de signaux électriques représentatifs et exploitables.

Ainsi, de façon générale, un système d'acquisition et de contrôle de données est constitué d'un certain nombre de composants et de sous-systèmes intégrés pour assurer les fonctions suivantes :

- ❖ l'extraction de l'information relative à chacune des grandeurs physiques à connaître et sa traduction en signal électrique au moyen de capteurs et conditionneurs ;
- ❖ le traitement analogique du signal destiné en particulier à en éviter la dégradation par le bruit et les parasites : amplification, filtrage ;
- ❖ la sélection, parmi l'ensemble des signaux disponibles, du seul signal requis à l'aide du multiplexeur ;

- ❖ la conversion du signal sous forme numérique adaptée au calculateur chargé de l'exploiter , au moyen de l'échantillonneur-bloqueur et du convertisseur analogique numérique ;
- ❖ la coordination des opérations précédentes assurée par un calculateur qui, en plus, peut être chargé de divers traitements sur les signaux numériques qu'il a acquis, traitements devant faciliter leur exploitation (linéarisation) ou corriger des défauts de l'appareillage (dérives thermiques compensées après auto zéro et auto étalonnage par exemple) ;
- ❖ le traitement, l'analyse et l'archivage sur le disque dur du PC des informations recueillies ;
- ❖ la génération d'une sortie appropriée basée sur l'algorithme de contrôle.

Un exemple de structure d'un système d'acquisition et de contrôle de données est représenté par la figure I.1.

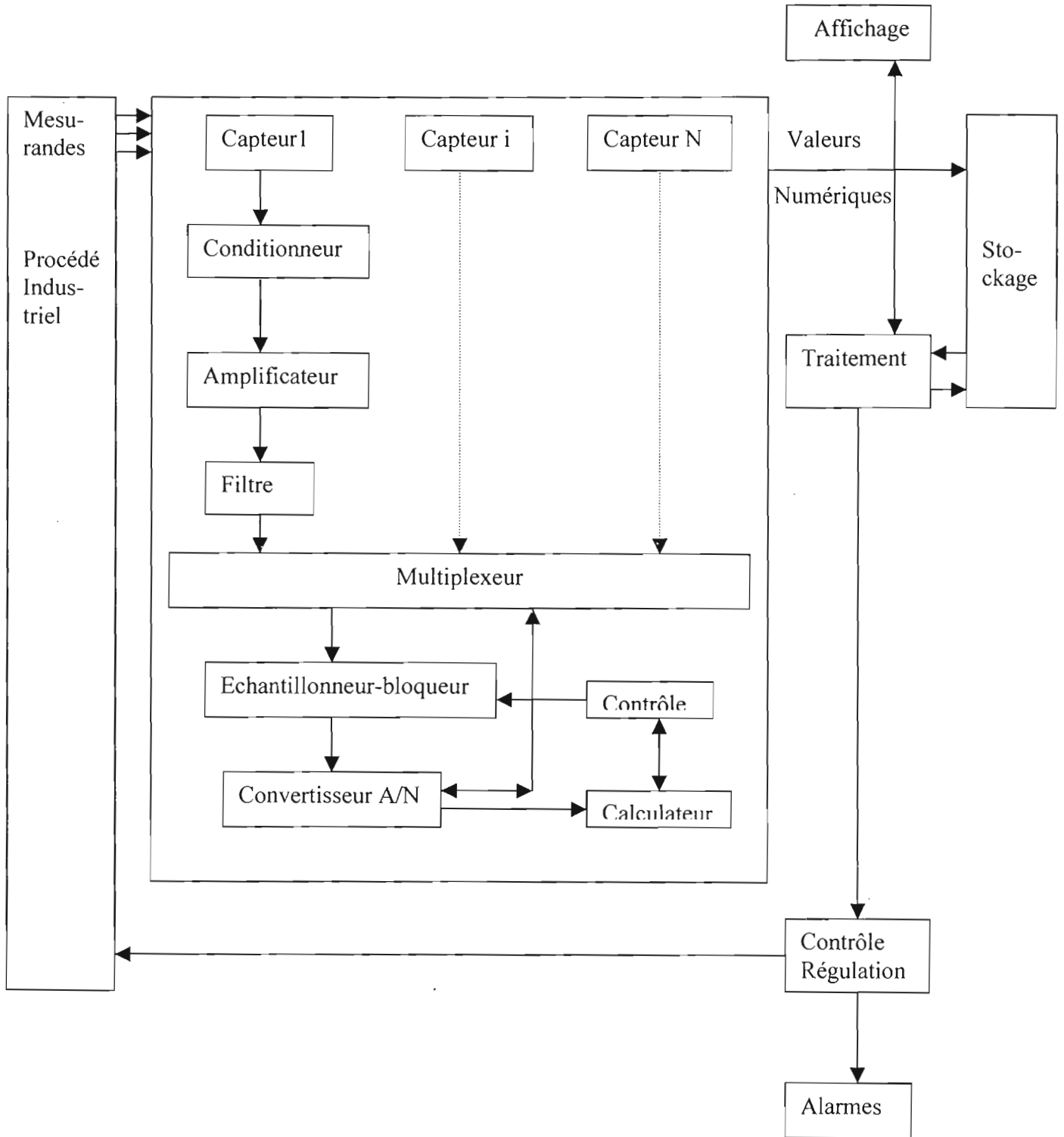


Figure I.1 : Exemple détaillé d'un ensemble mesure-contrôle de procédé.

I.2 Les capteurs et leurs conditionneurs

Les capteurs sont les premiers éléments d'une chaîne d'acquisition de données. A chaque mesurande est associé un capteur dont le rôle est de traduire ce mesurande (m) en une grandeur électrique $s(m)$, fonction connue du mesurande, de telle sorte qu'à chaque valeur s de cette grandeur électrique puisse être liée la valeur m du mesurande. Cette grandeur électrique $s(m)$ est à l'origine du signal électrique que doit traiter le système d'acquisition.

Mais le problème à ce stade est que la plupart des dispositifs ne sont aptes à traiter le signal électrique que sous la forme de tension. L'un des premiers rôles donc, de ce que l'on appelle le conditionneur du capteur, est de convertir, lorsqu'elle n'est pas une tension, la grandeur électrique de sortie du capteur, ou ses variations par rapport à un état d'origine, en une tension dont l'amplitude ou / (et) la fréquence sont déterminées par la sortie du capteur.

Les autres rôles des conditionneurs sont leur aptitude à corriger le signal qu'ils délivrent, à savoir la possibilité de la compensation des grandeurs d'influence (parasites, bruit) et aussi leur capacité à linéariser le signal et à l'amplifier.

Les capteurs sont caractérisés principalement par leur étendue de mesure (E.M) qui représente la plage de valeurs du mesurande pour laquelle le capteur répond aux spécifications du constructeur. Les capteurs sont également définis par leur sensibilité $S(m)$ qui, pour une valeur m du mesurande, est égale au quotient de la variation de la sortie électrique par la variation correspondante du mesurande.

Un capteur est dit linéaire si, dans l'étendue de mesure, $s(m)$ est une fonction linéaire de m ; sa sensibilité est alors constante dans cette plage de fonctionnement.

Les grandeurs d'influence sont les grandeurs physiques autres que le mesurande et qui sont susceptibles d'affecter la sortie électrique du capteur. L'action d'une grandeur d'influence se traduit généralement par le décalage du zéro et une dérive de sensibilité. La grandeur d'influence la plus générale est constituée par la température du capteur, les

vibrations, la pression et l'humidité. Les champs magnétiques peuvent aussi exercer une influence sur la réponse du capteur.

Suivant le type de signal électrique $s(m)$ qui est délivré à leur sortie, les capteurs peuvent être divisés en quatre grandes familles que sont :

I.2.1 Les capteurs analogiques

Ils fournissent un signal analogique en fonction d'une grandeur physique, électrique, mécanique, ... C'est la catégorie la plus importante. On peut décomposer cette famille en deux grands groupes que sont :

→ les capteurs physiques (capteurs optiques, capteurs de déformation par force, pression, accélération, capteurs de température, capteurs magnétiques, capteurs acoustiques, rayonnements nucléaires) caractérisés par l'existence d'une relation permanente entre la grandeur à mesurer et le signal de sortie du capteur.

→ les capteurs chimiques (capteurs de gaz, capteurs d'humidité, capteurs ioniques, capteurs biochimiques) qui ne possèdent pas la caractéristique des capteurs physiques et vont donc impliquer un processus d'utilisation séquentiel beaucoup plus complexe et beaucoup plus susceptible de conduire à des erreurs d'interprétations. En effet, les capteurs chimiques nécessitent une réaction physico-chimique avec l'environnement qu'ils sont chargés d'appréhender, réaction mettant en jeu une surface d'échange dont, par essence, la taille est limitée. Il en résulte le risque de saturation et donc d'inefficacité du capteur, risque que l'on devra apprécier et éviter en procédant très régulièrement à un reconditionnement du capteur. Ce processus sera généralement automatisé selon une fréquence adaptée (toutes les secondes parfois) et, pendant sa durée, le capteur ne sera évidemment pas disponible pour une mesure.

I.2.2 Les capteurs logiques

Les capteurs logiques sont des capteurs présentant deux états (0 et 1) dont le modèle est le contact ouvert ou fermé (fin de course, capteur de niveau...). C'est la catégorie appelée «key sensor» en anglais. Leur importance est notoire dans de nombreux domaines de la technique. Ils interviennent le plus souvent en tant qu'éléments de sécurité.

I.2.3 Les capteurs numériques

Ils fournissent un signal fréquentiel, c'est-à-dire un certain nombre d'impulsions par unité de temps en fonction de la grandeur mesurée. Le capteur type de cette catégorie est le tachymètre à opto-coupleur ou encore le pluviomètre à auget basculant. Leur intérêt majeur réside dans leur facilité d'emploi puisqu'ils peuvent être directement couplés à un compteur et ne nécessitent donc qu'une électronique associée réduite.

I.2.4 Les capteurs digitaux

Il s'agit de dispositifs qui donnent directement des informations binaires combinatoires. Ils sont parfaitement représentés par la famille des codeurs optiques. Leur concept est très intéressant puisqu'il s'agit de dispositifs qui peuvent être directement couplés au système informatique sans la nécessité d'un système d'amplification et de conversion analogique numérique. Ce sont des dispositifs qui possèdent une grande immunité au bruit et sont peu influencés par des grandeurs parasites.

I.3 Les convertisseurs Analogiques/Numériques (CAN)

Une fois mesuré et conditionné, le signal est présenté à la carte qui convertit la tension ou le courant conditionné en un format numérique exploitable par le PC.

En effet, un signal analogique est un paramètre physique défini, fonction continue du temps. Et puisque l'ordinateur ne traite les informations que sous un format numérique, alors il devient impératif de convertir le signal analogique obtenu en un signal numérique. Ce rôle est tenu par la carte d'acquisition de données. Durant cette conversion, le signal d'entrée est comparé à une référence représentant une tension de pleine échelle. Suivant la fraction à laquelle est converti le signal, il lui est associé, en sortie, un nombre codé en numérique.

Une chaîne d'acquisition est représentée de façon schématique ci-dessous.

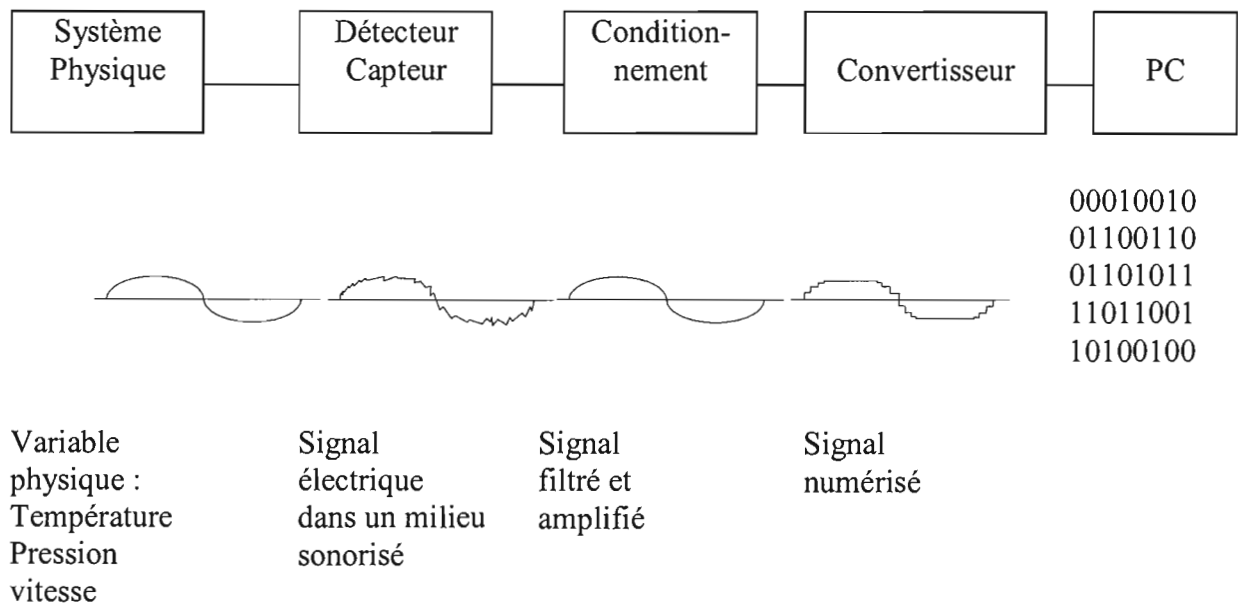


Figure I.2 : Structure générale d'une chaîne d'acquisition de données

Plusieurs méthodes de conversion analogique-numérique sont utilisées dont les méthodes par approximations successives, les méthodes Sigma-Delta, les méthodes Flash.

Les cartes incorporent également plusieurs des capacités suivantes :

- ➔ grandes vitesses DMA de transfert des données vers le PC ;
- ➔ FIFO (First In First Out) et mémoire tampon;

- filtrage contre le bruit de fond ;
- amplificateur à gain programmable.

I.3.1 La période d'échantillonnage

La période d'échantillonnage représente, quant à elle, la rapidité avec laquelle la carte peut scanner le canal d'entrée et identifier la valeur discrète du signal présent. Pour acquérir correctement les données, il faut évidemment que la fréquence d'échantillonnage soit supérieure à celle du signal analogique lui-même (d'au moins le double).

I.3.2 La résolution

Elle représente le nombre de divisions d'une portée de pleine échelle et seules les tensions d'entrée analogiques ayant une valeur supérieure ou égale à la valeur d'une de ces divisions pourront générer une sortie numérique. A chaque niveau de division est également associé un nombre numérique. Plus il y a de divisions, plus la sortie numérique générée est fidèle au format de l'entrée analogique.

I.3.3 Les entrées et sorties digitales

Les cartes d'entrées analogiques disposent généralement de canaux d'entrées/sorties digitales utilisables dans certaines fonctions comme :

- ✓ la mesure et le contrôle des tensions de grandes puissances à partir de relais ;
- ✓ l'utilisation de signaux de courant faible TTL (Transistor Transistor Logic) ;
- ✓ la génération des signaux de taux comme les pulsations, les horloges...

I.3.4 Les sorties analogiques

Les sorties analogiques sont générées en utilisant des procédures réciproques de celles utilisées pour les entrées analogiques à savoir des convertisseurs numériques analogiques. Une sortie analogique est souvent requise pour une application impliquant un appareil commandé par une variable de contrôle telle qu'un servomoteur. La sortie pourra alors être configurée en tension ou en source de courant.

II. L'INSTALLATION HYDRAULIQUE

II.1 Description et Principe de fonctionnement

Les installations FM3 Armfield, d'hydro-puissance à petite échelle, consistent en quatre prototypes de machines industrielles conçues pour mettre en évidence le principe de fabrication et d'opération des turbines.

Trois de ces machines, à savoir la turbine à impulsion FM30 (écoulement axial du fluide), la turbine à réaction FM31 (écoulement radial du fluide) et la turbine à action Pelton FM32, peuvent être reçues séparément par la FM3SU.

Cette dernière unité est constituée par :

- ✓ un support recevant une pompe de circulation centrifuge au débit maximal de 0.33l/s à une hauteur de charge de 29 m,
 - ✓ un réservoir d'eau,
- l'ensemble étant supporté par un banc.

Une fois la FM3SU montée avec l'une des trois turbines, l'installation devient complète et est prête à fonctionner en circuit fermé avec la pompe qui sera chargée de la circulation en recueillant l'eau du réservoir pour l'envoyer vers la turbine.

Les paramètres de fonctionnement sont alors :

- ✓ la pression différentielle entre l'orifice d'entrée de la pompe et les rebords du réservoir,
- ✓ la pression en aval de la pompe (ou encore pression d'entrée de la turbine) contrôlée dans la tuyauterie par une vanne d'étranglement,
- ✓ la vitesse de rotation du rotor de la turbine
- ✓ la force exercée sur la gaine de freinage du rotor et donnant le couple développé par le rotor.

A chacun de ces paramètres, est associé un capteur chargé de recueillir les données et d'envoyer les signaux conditionnés à une carte d'acquisition de données, le DT2811-PGH, via une console électrique, le IFD1. Les données acquises et présentées à l'ordinateur sont exploitées grâce à un programme sous l'environnement DOS qui permet, en servant de pilote à la carte, de recueillir et de stocker les données en vue de leur éventuelle utilisation par l'opérateur.

Une schématisation de l'installation est donnée ci-dessous :

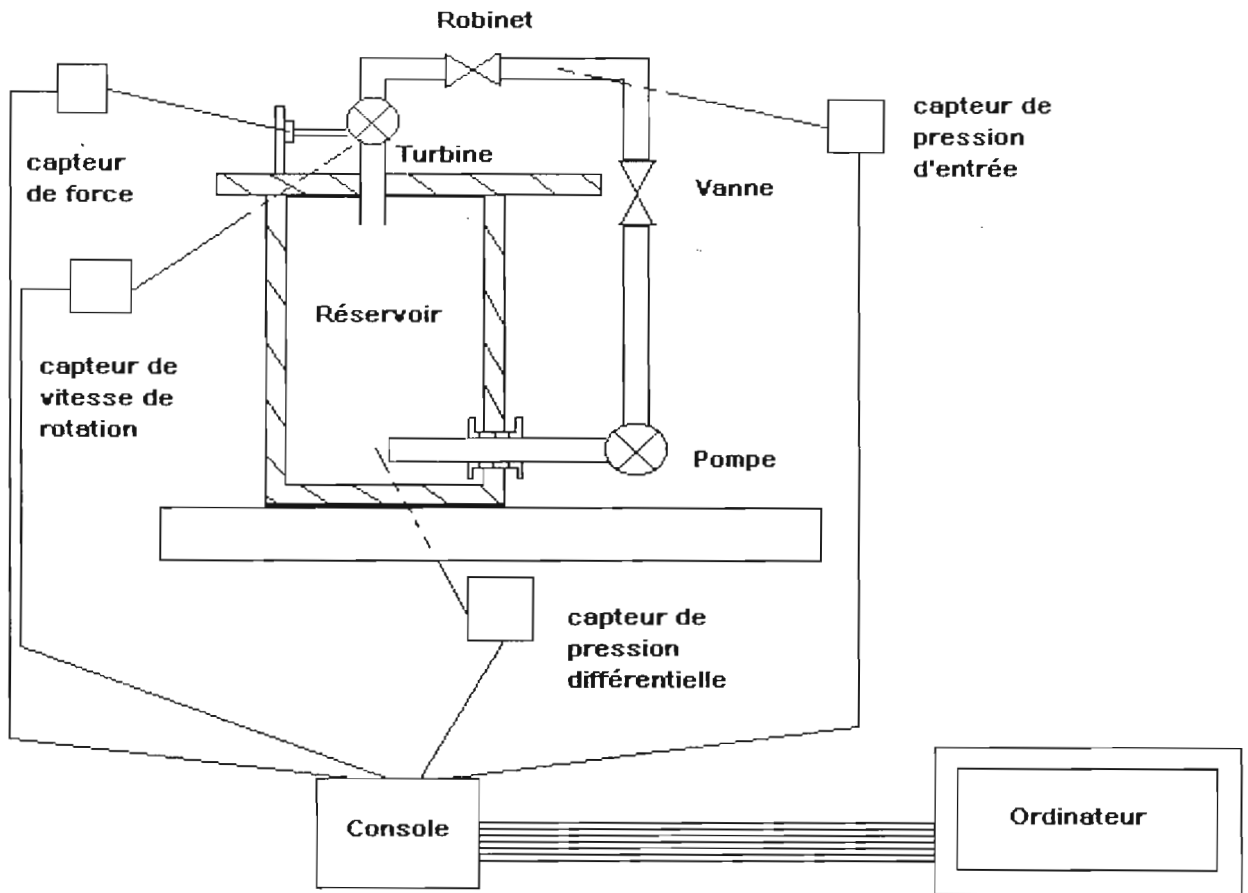


Figure II.1 : Schématisation de l'installation hydraulique

II.2 Les capteurs et leurs conditionneurs

Au niveau de l'installation, sont montés des capteurs analogiques avec leurs conditionneurs respectifs. Ainsi, on a :

⊙ un capteur de pression piézoélectrique dont le fonctionnement repose sur le principe de la piézoélectricité.

En effet, tout matériau piézoélectrique qui subit une contrainte voit ses ions varier de position ce qui induit un déséquilibre électrostatique et un champ électrique est ainsi créé. La charge électrique développée résultante du champ électrique est proportionnelle à la contrainte appliquée. C'est l'effet direct.

Ce capteur de pression, monté dans la tuyauterie en aval de la pompe, subit donc la pression de la veine liquide refoulé par la pompe. Il est ensuite couplé à un conditionneur de signal approprié, protégé par un boîtier, et délivre ainsi un signal dont la tension est comprise entre $\pm 5V$.

⊙ un capteur de vitesse optique à réflexion infra-rouge utilisé pour recueillir la vitesse de rotation de la turbine, et dont le signal, est conditionné dans un boîtier de protection. Le capteur optique est monté adjacent au moyeu du rotor qui incorpore une languette réflective. Le capteur induit donc, à ses bornes, une tension proportionnelle à la vitesse avec laquelle le signal infra-rouge émis est réfléchi.

⊙ un capteur de force qui consiste en une jauge de contrainte incorporée dans la poutre de charge et dont le signal est envoyé dans un boîtier conditionneur. Les capteurs à jauges de contraintes reposent sur l'effet piézorésistif qui est l'effet d'une pression sur les caractéristiques d'une résistance soumise à la même déformation que la membrane qui la supporte. En effet, sachant que la résistance est une fonction linéaire de la longueur, la

variation géométrique d'une jauge de contrainte se traduira par une variation de sa résistivité ce qui pourra permettre d'identifier l'amplitude de la contrainte exercée.

III. LIBELLE DU CAHIER DES CHARGES

Les travaux pratiques de Machines Hydrauliques sont normalement destinés à servir de support au cours du même nom.

Grâce à la reconstitution, en échelle réduite, d'un dispositif de turbines, il est mis à la disposition de l'opérateur un ensemble d'outils permettant de reconstituer, à petite échelle, les conditions de fonctionnement d'une unité de turbines avec tout ce que cela comporte comme interprétations de résultats. En effet, en faisant varier les paramètres de fonctionnement de l'installation, l'opérateur peut constater la manière dont évoluent les caractéristiques de débit, de couple, de puissance, de rendement et de hauteur de charge en fonction de la vitesse de rotation de la turbine, dans les conditions réduites de fonctionnement.

Cela revêt une grande importance pédagogique, en ajoutant le fait que, sur le marché, on ne trouve jamais de turbines toutes faites, prêtes à fonctionner étant entendu qu'il n'y a jamais deux conditions de fonctionnement totalement identiques. Il apparaît, dès lors, que l'installation de toute unité de turbine, dans un environnement bien déterminé, nécessite tout d'abord la réalisation d'un prototype reproduisant exactement toutes les conditions de fonctionnement. C'est à ce stade seulement que les paramètres nécessaires au bon dimensionnement de l'unité sont enregistrés.

De la facilité d'exploitation de l'unité de turbine disposée, dépend aussi une meilleure compréhension et une plus grande rapidité d'exécution des travaux pratiques.

Le programme actuel d'exploitation de l'unité de turbines se présente sous forme de menus. Pour accéder à chaque fonctionnalité, depuis l'échantillonnage jusqu'au tracé des graphiques caractéristiques, en passant par le calcul des paramètres de fonctionnement, il est nécessaire, tout le temps, d'appuyer sur le clavier.

Pour chaque débit de fluide considéré, un maximum de quinze échantillonnages seulement peut être réalisé.

Avant toute exploitation de données, il faut un stockage préalable dans un support magnétique. Il existe alors un grand décalage entre la période d'échantillonnage et celui de la visualisation des résultats sous forme graphique.

Le programme fait alors penser à une énorme toile d'araignée.

La nouvelle chaîne d'acquisition et de traitement de données à réaliser devra alors permettre d'enlever toute lourdeur au déroulement des T.P en garantissant une automatisation maximale des tâches. Et c'est ainsi que le problème se pose.

Une étude plus minutieuse du système déjà en place a tout de même révélé un autre problème et non des moindres. En effet, seul le paramètre rendant compte de la vitesse de rotation varie sur le logiciel déjà sur place. Ceci implique que le but primordial des T.P à savoir les calculs, le tracé des graphes et l'interprétation des résultats ne pourra être atteint s'il se révèle que la défaillance constatée provient du matériel lui-même. Mais, à ce stade du projet, rien ne permet de faire une telle affirmation. Seule la mise en place du nouveau système pourra nous édifier sur la situation.

En définitive, le travail ci-après consistera, non seulement en la conception d'un nouveau système d'acquisition et de traitement de données pérenne, convivial et facile d'exploitation mais aussi, si la défection de l'installation se confirme, à trouver les causes possibles du problème en vue de proposer d'éventuelles solutions.

IV. EXPLICATIONS SUR LE TYPE DE SOLUTION ADOPTÉE POUR LA RESOLUTION PRATIQUE DU PROBLEME

La résolution du problème offre, dès le début, la possibilité de concevoir un nouveau système qui, de par sa structure, annule carrément la possibilité d'une quelconque conservation du système déjà installé. C'est le fait de partir directement de la sortie des capteurs.

Une autre possibilité est de concevoir un nouveau système en parallèle avec l'actuel. Et c'est cette dernière solution qui a été choisie pour les raisons suivantes :

- elle garantira la conservation par défaut du système actuel ;
- elle donnera, à chaque phase de la réalisation du nouveau système, un élément sur lequel l'on pourrait se baser pour faire une comparaison des données affichées par chacun des deux systèmes ;
- elle permettra la réalisation et le suivi des essais de réparation relatifs à la défection du système : montages-tests des différentes turbines, essais de recalibration...

La réalisation du système parallèle intègre l'utilisation d'une nouvelle carte à installer sur un ordinateur parallèle. Le CIO DAS08 PGH convient parfaitement car il est disponible au laboratoire d'électrotechnique et reste compatible avec le logiciel d'acquisition de données choisi à savoir le « Labtech ».

Pour faire les câblages, trois nouvelles solutions se présentent à savoir partir:

- de la sortie de la console électrique ;
- ou de la sortie des conditionneurs ;
- ou remonter plus loin en sortie des différents capteurs.

La première solution est à écarter puisqu'il n'y a aucune information sur les câblages réalisés à l'intérieur de la console, concernant les voies des signaux à acquérir, et de toute façon, il apparaît clair que les câblages ont été réalisés pour la carte DT2811-PGH.

L'adoption de la troisième solution, comme spécifié précédemment, rendrait impossible la mise en place du système parallèle pour cause de branchements, sans compter la nécessité alors de réaliser un long travail de conditionnement dans un environnement de données pas évidemment accessible.

Il reste alors la possibilité de partir de la sortie des conditionneurs, ce qui écarte toutes les difficultés citées ci-dessus. Il faut ensuite relier, par un travail de câblage approprié, les différents capteurs à la carte.

Cette solution est d'autant plus acceptée que les tensions conditionnées de sortie restent comprises dans la plage ± 5 V alors que la carte peut recevoir des tensions jusqu'à ± 10 V sans subir de dommages.

V. SOLUTION DE CABLAGE DES APPAREILS ELECTRONIQUES

D'une part, on a des capteurs, d'autre part, on a une carte à installer dans un ordinateur et entre les deux sera mis en place une pannelle qui servira de tampon. Cette pannelle permettra de faire correspondre, à chaque capteur, des voies d'entrées et de sorties de données bien déterminées et servira de substitut à la console dans le nouveau système. Néanmoins, la console gardera une de ses fonctions à savoir l'alimentation électrique de la pompe de circulation.

Le premier travail consiste, à ce stade, à déterminer les valeurs des signaux délivrés, en sortie, par les capteurs.

Le dégarnissage des câbles, en sortie des conditionneurs, montre la présence de six fils dans chaque câble. A l'ouverture des conditionneurs, l'analyse des circuits intégrés a pu faire correspondre, à chaque type de fil, le type et la valeur du signal y affairant.

C'est ainsi que le tableau ci-dessous a été obtenu :

COULEURS DE FILS	TYPE ET VALEUR DU SIGNAL
Jaune	DAC0
Rouge	+ 12 V
Bleu	SIG
Vert	0 V
Blanc	- 12 V
Noir	DAC1

Tableau V.1: Signal délivré par chaque fil dans un câble

- Les signaux +12 V et -12 V servent à l'alimentation des circuits de conditionnement des capteurs.
- Le SIG correspond au signal délivré par les capteurs.
- Le 0 V correspond à la masse .
- Le DAC0 et le DAC1 correspondent aux éventuels signaux de commande auxquels l'utilisateur peut faire recours pour commander, en sortie de la carte, un paramètre désiré de l'installation. On ne les utilisera pas.

Une fois les capteurs reliés correctement à la carte, cette dernière sera utilisée pour délivrer les tensions de +12 V et - 12 V et recevra en retour les signaux conditionnés délivrés par les capteurs.

VI. LA CARTE D'ACQUISITION DE DONNEES CIO-ADO8-PGH

VI.1 Introduction

Le CIO-AD08-PGH est une extension de la carte CIO-AD08, qui est très utilisée, du point de vue architectural. Les deux cartes sont identiques au niveau des registres, des performances avec les exceptions suivantes :

1. Les gains sont programmables : les gains disponibles sont de 0,5, 1, 5, 10, 50, 100, 500, et 1000.
2. Les entrées analogiques sont des entrées différentielles. Cependant, la carte est conçue pour pouvoir accueillir des résistances pour convertir les entrées en mode pseudo-différentiel.
3. Un oscillateur stable (« XTAL ») délivre un signal d'horloge. Pour que les programmes écrits pour le CIO-AD08 soient compatibles avec le CIO-AD08-PGH, un sélecteur permet de choisir soit l'oscillateur, soit l'horloge du bus.
4. Un convertisseur cc/cc délivre une tension stable de +/-15V aux circuits analogiques.
5. Il y a un seul connecteur sur la carte.

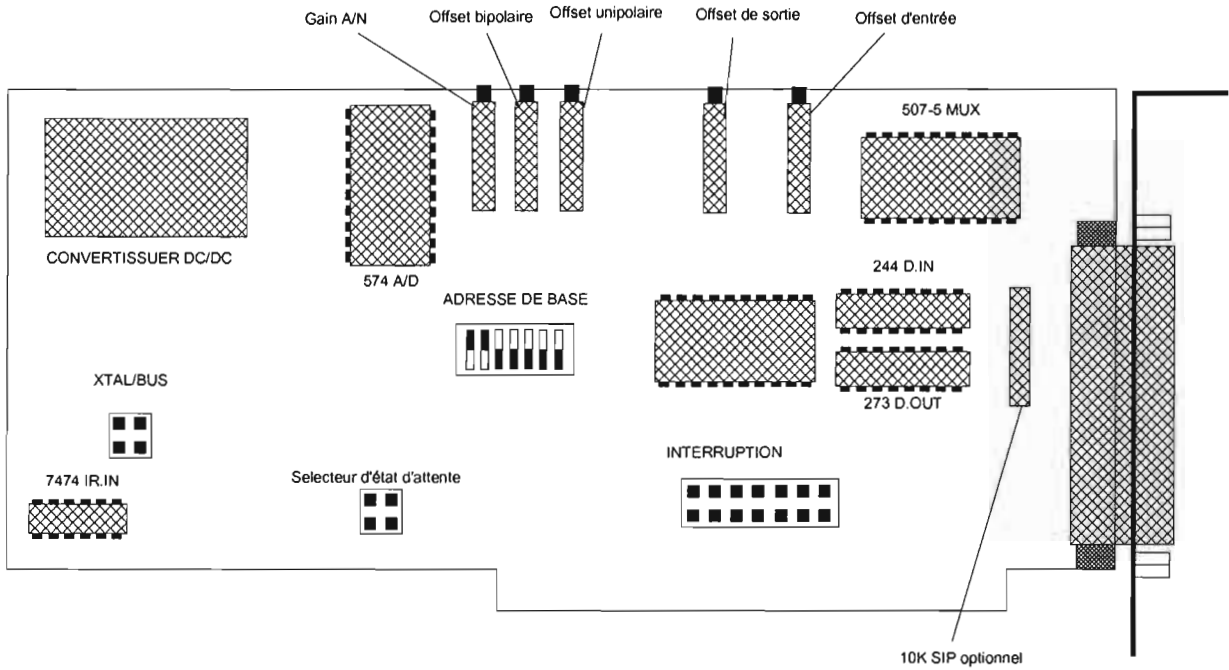


Figure VI-1 : La carte CIO-AD08-PGH et ses composants

VI.2 Connexions des signaux

VI.2.1 Le diagramme de connexions

Le connecteur est à 37 pins, c'est un connecteur de type D accessible à l'arrière du PC par le bornier d'expansion (figure VI.2). Ce connecteur accepte des connecteurs femelles de type D tel que le C3FF-2 présentant deux connecteurs aux extrémités du câble.

S'il y a des changements fréquents au niveau des connexions, les cartes CIO-TERMINAL et CIO-MINI37 pourront faciliter l'utilisation de la carte.

Nous avons donc utilisé le **CIO-MINI37** pour faire les connexions à la sortie du PC.

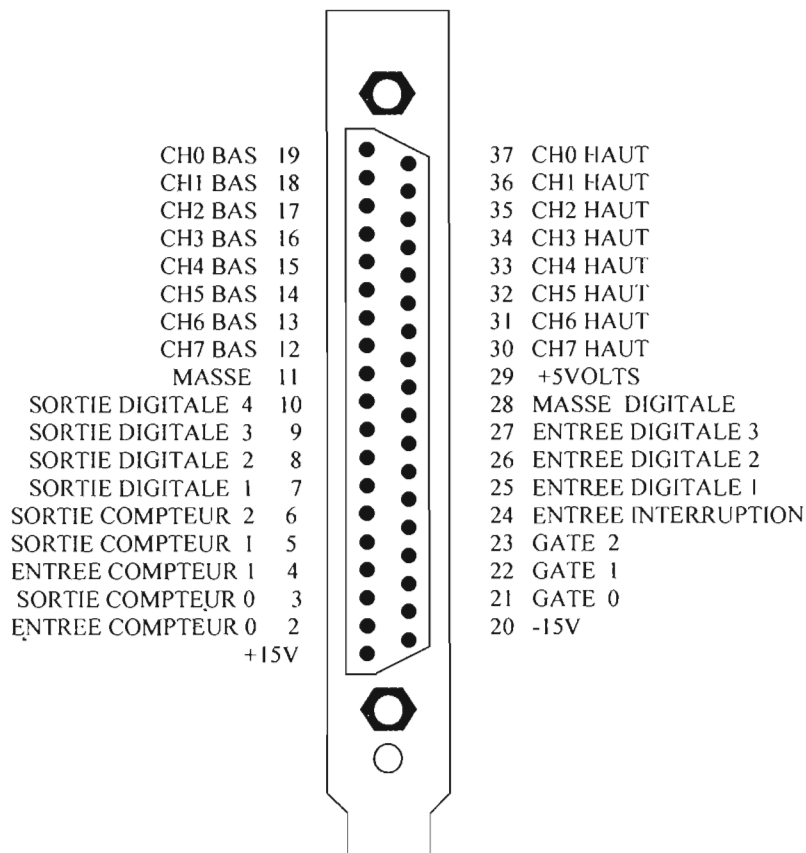


Figure VI.2: Diagramme de connexions du CIO-AD08-PGH

VI.2.2 Entrées différentielles

Les entrées différentielles sont des jonctions analogiques à 3 fils constituées d'un signal haut, d'un signal bas et de la masse. L'avantage dans l'utilisation des entrées différentielles est la capacité de rejet du bruit qui affecte tant le signal haut que le signal bas et la capacité de compenser les différences de potentiel entre le signal bas et la masse.

Bien que les entrées différentielles soient souvent préférables aux entrées simples, il y a des cas où la mesure d'une entrée différentielle peut produire des erreurs lors des essais de lecture. Dans ces cas, les entrées peuvent être converties en différentiel modifié.

VI.2.3 Entrées et sorties digitales

Toutes les entrées digitales sur le CIO-DAS08-PGH sont de niveau TTL.

TTL est un terme d'industrie d'électronique (Transistor Transistor Logic) caractérisé par la valeur de la tension des signaux digitaux qui sont de 0 ou 5V.

VI.3 Installation de la carte dans l'ordinateur

- Éteindre la machine.
- Enlever la couverture de l'ordinateur en évitant de déplacer n'importe lequel des câbles installés sur les cartes dans l'ordinateur.
- Localiser un connecteur d'expansion vide dans l'ordinateur.
- Pousser la carte fermement en bas dans le connecteur du bus d'expansion. S'il n'est pas assis entièrement il ne pourra pas fonctionner correctement et pourrait court-circuiter la puissance du bus du PC. Cela pourrait endommager la carte mère dans le PC aussi bien que la carte.

VI.4 Installation de la carte point de vue logiciel

Les cartes d'acquisitions sont des circuits électroniques (hard) pilotées par des programmes (soft) qui gèrent l'exécution des différentes fonctions d'acquisitions de données et assurent l'affichage, sur le moniteur, des données ainsi que leur stockage sur disque dur ou tout autre support magnétique.

Le logiciel choisi est le Labtech Notebook et il faut installer la carte dans le logiciel pour que ce dernier puisse la reconnaître et aller chercher, à tout moment, les informations nécessaires.

Le logiciel Labtech Notebook intègre ainsi un programme nommé INSTACAL pour l'installation et le test des cartes d'acquisition de données de plusieurs marques dont le OMEGA CIO series, famille à laquelle appartient la carte CIO-AD08-PGH.

Il est à noter que INSTACAL est un programme développé sous DOS intégrant des menus d'installation, de calibrage, de tests, d'options et de fermeture de programme.

Il est représenté ci-dessous :

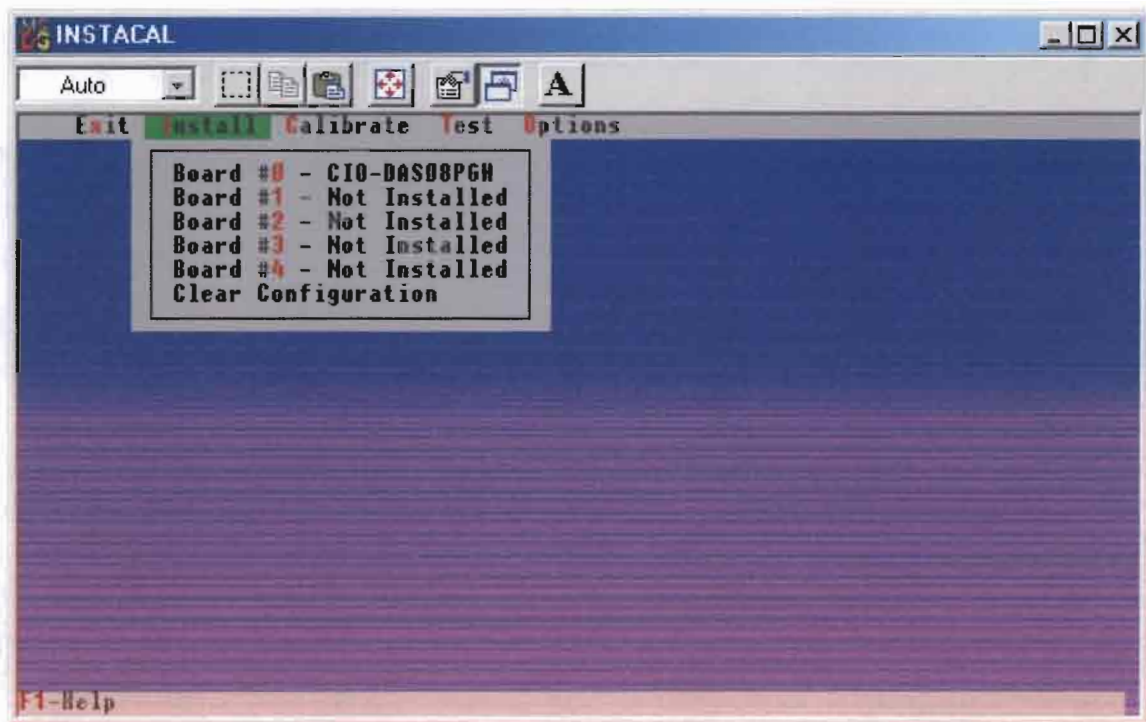


Figure VI.3: Les menus du programme « Instacal »

L'installation de la carte demande la saisie de diverses informations à savoir la valeur de l'adresse de base, le niveau d'interruption, la valeur de l'horloge, le type d'état d'attente.

VI.4.1 L'adresse de base

VI.4.1 L'adresse de base

L'adresse de base du CIO-AD08-PGH est choisie en faisant commuter une banque de basculeurs sur la carte (figure VI.4). Cette banque de commutateurs est libellée ADDRESS et numérotée de 9 à 3.

Pour sélectionner une adresse de base, la banque fonctionne en additionnant les poids individuels actifs des commutateurs. Un poids est actif quand le commutateur est en bas. Comme illustré ci-dessus, les commutateurs 8 et 9 sont en bas pendant que tous les autres occupent la position haute. Les poids 200H et 100H sont ainsi actifs ce qui donne, par addition, l'adresse de base 300. Elle constitue l'adresse choisie par défaut pour la carte à la sortie de l'usine. Si cette adresse est déjà utilisée dans l'ordinateur, une adresse différente doit être choisie pour éviter les conflits entre les périphériques.

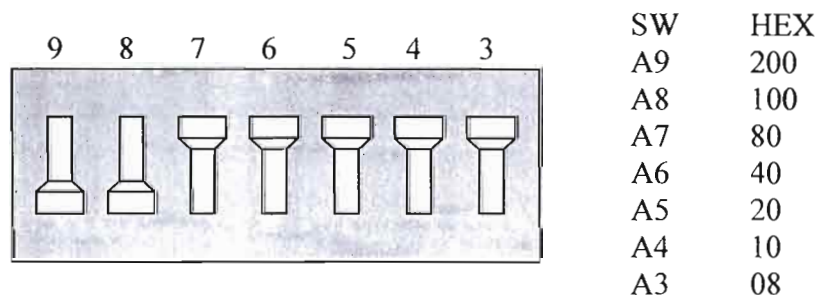


Figure VI.4 : Commutateur d'adresses de base

Dans la configuration de l'ordinateur, les plages disponibles s'étendent de 300 à 31F.

La carte installée a été **adressée à 300**.

VI.4.2 Sélection du niveau d'interruption

Un niveau d'interruption est choisi seulement si le logiciel employé le demande. Il faudra alors tenir compte de la configuration établie de l'ordinateur afin d'éviter les conflits d'interruption.

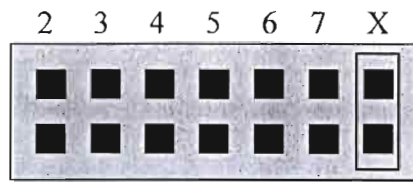


Figure VI.5 : Sélecteur de niveau d'interruption

Il y a un bloc d'interruption, sur le CIO-AD08-PGH, placé juste au-dessus de l'interface du bus de PC (épingles d'or). Le réglage par défaut de l'usine est tel qu'aucun niveau d'interruption n'est sélectionné. La broche est dans la position « X » (figure VI.5).

Le CIO-AD08-PGH utilise les niveaux d'interruption 5 et 7 qui sont le plus souvent disponibles.

5 a donc été pris comme **niveau d'interruption**.

VI.4.3 Sélecteur d'horloge

La source de l'horloge pour le CIO-DAS08-PGH est fixée à 1 MHz pendant que le CIO-DAS08 utilise la source du bus du PC.

La position CLK BUS est choisie pour être compatible avec les programmes écrits pour le CIO-AD08 qui est plus ancien que la carte que nous aurons à utiliser. La position choisie par défaut est de 1mhz.

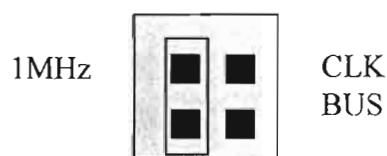


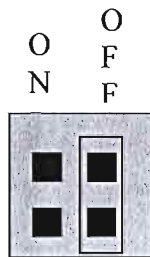
Figure VI.6: Sélecteur d'horloge

VI.4.4 Etat d'attente

L'état d'attente peut être validé sur la carte en sélectionnant, sur le label WAIT STATE, la position ON. La validation de l'état d'attente cause le ralentissement du taux de transfert de données entre la carte et le bus de l'ordinateur.

L'état d'attente est validé dans le cas où l'ordinateur aurait une vitesse de transfert des entrées/sorties de son bus très rapide par rapport au CIO-DASO8-PGH.

Si la carte se plante souvent, la validation du WAIT STATE devient alors nécessaire.



FigureVI.7 : Sélecteur d'état d'attente

La carte entièrement configurée se présente alors, sur le moniteur, sous la forme donnée à la figure VI.8.

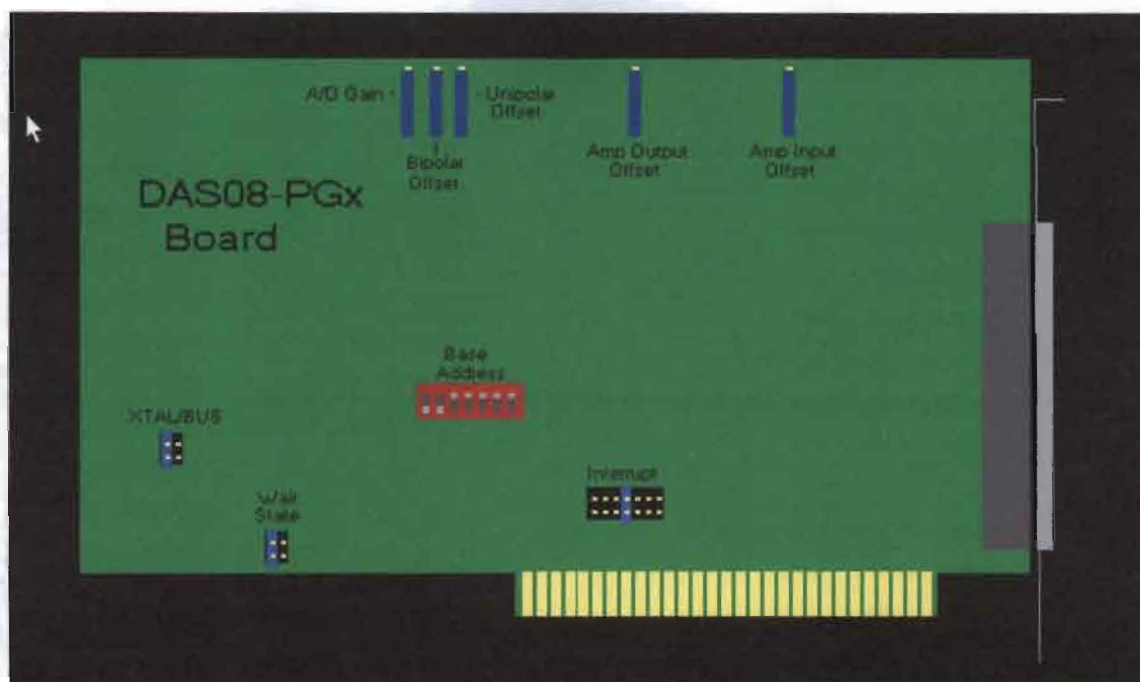


Figure VI.8 : Configuration graphique de la carte après son installation

VI.5 Tests de vérification

Après l'installation de la carte, il faut lui faire subir des tests pour s'assurer de son bon état de marche. Ces tests réalisés sur les différents canaux d'entrées/sorties, se sont déroulés en utilisant une tension externe, à savoir un voltmètre, pour délivrer une tension inférieure à 5 V. Les tests se sont révélés concluants indiquant le bon état de la carte.

A noter que le programme nous guide pour tester les canaux d'entrées /sorties de la carte. Une illustration en est faite à la figure VI.9.

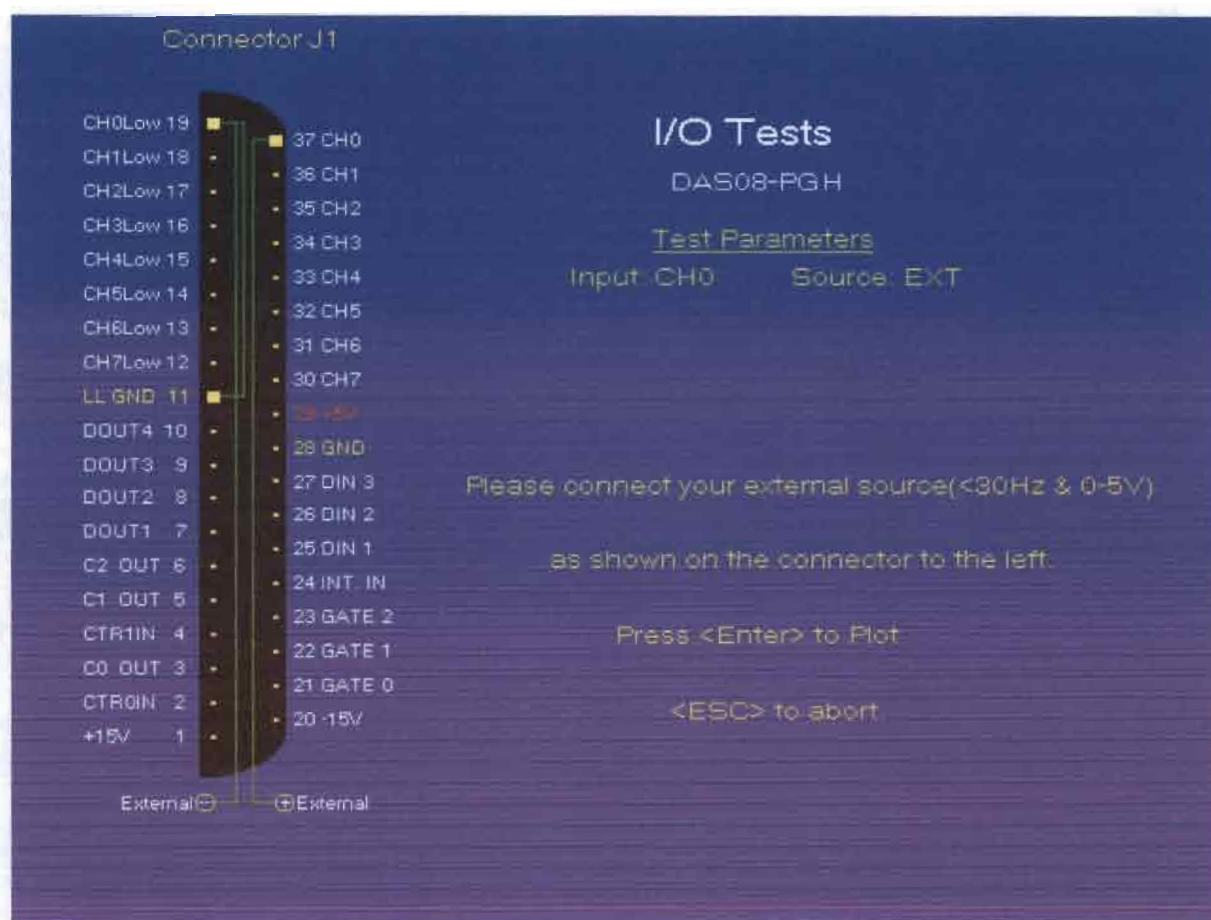


Figure VI.9 : Ecran montrant la connexion pour le test de la voie d'entrée 0

VII. CABLAGE DES CAPTEURS

Comme énoncé dans la solution de câblage à la section V, les sorties des conditionneurs vont être reliées à la carte par l'intermédiaire d'une pannelle. La carte se chargera alors de recevoir les signaux analogiques délivrés par les capteurs tout en fournissant les tensions de +12 V et de - 12 V destinées à l'alimentation des circuits de conditionnement.

Mais à ce niveau un problème survient. En effet, la carte ne peut délivrer que des tensions de +15 V et de - 15 V. Il faut alors intercaler, entre les circuits conditionneurs et la carte, un dispositif électronique permettant de rabaisser la tension de la valeur 15 V à la valeur 12 V stabilisée.

Trois montages peuvent être utilisés :

- ➔ un montage diviseur de tension ;
- ➔ un montage amplificateur opérationnel abaisseur de tension ;
- ➔ un montage régulateur de tension.

L'étude de chacun de ces montages permettra de conclure quant à la meilleure solution à adopter.

VII.1 Principe du montage diviseur de tension

Le diviseur de tension est un dispositif électrique fonctionnant sur le principe suivant : quand une tension est appliquée aux bornes d'une série de résistances parcourue par un même courant, la tension recueillie aux bornes de chaque résistance est proportionnelle à la valeur de la résistance suivant la loi d'Ohm :

$$U = RI \quad \text{avec} \quad U = \text{tension appliquée en Volts (V)}$$

$$R = \text{résistance en Ohm } (\Omega)$$

$$I = \text{courant en Ampères (A)}$$

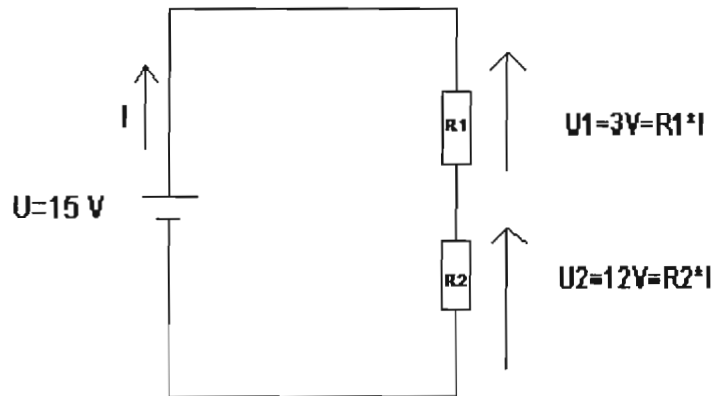


Figure VII.1 : schématisation du circuit diviseur de tension

En posant $\frac{U2}{R2} = \frac{U1}{R1}$ sachant que $U2 = 12\text{ V}$ et $U1 = 3\text{ V}$, on obtient alors $R2 = 4 \times R1$.

Il faudra alors réaliser deux circuits diviseurs, l'un destiné à une tension d'entrée de $+15\text{ V}$ et l'autre à une tension de -15 V .

VII.2 Principe du montage amplificateur opérationnel

Les amplificateurs opérationnels sont des circuits intégrés très utilisés en électronique analogique linéaire et qui permettent la réalisation de plusieurs opérations mathématiques telles l'addition, la soustraction, la différentiation (dérivation), l'intégration... de tensions et de courants.

Actuellement, l'amplificateur opérationnel est utilisé très fréquemment dans la multiplication ou élévation de tension et dans la division ou abaissement de tension.

A cet effet, un montage amplificateur opérationnel, non-inverseur de tension, peut être visualisé sur la figure VII.2.

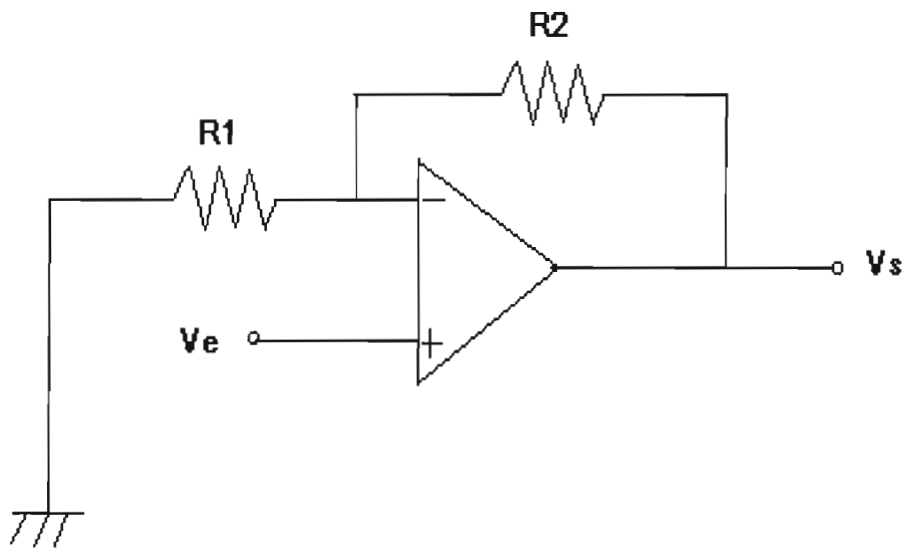


Figure VII.2 : montage non-inverseur

L'expression de la tension de sortie en fonction de la tension d'entrée donne :

$$V_s = \left(1 + \frac{R_2}{R_1}\right) \times V_e$$

La solution de l'utilisation d'un amplificateur opérationnel, dans la résolution de notre problème, consisterait à prendre la tension disponible de +15 V et à l'abaisser avec un gain de 0,8, grâce à un amplificateur non-inverseur de tension, pour obtenir une tension de sortie de +12 V. La tension de -12V s'obtiendra en utilisant un montage amplificateur inverseur.

VII.3 Principe du montage régulateur de tension

Le régulateur de tension est un montage électronique qui permet, en appliquant une tension d'entrée fixe ou variable en son entrée, de recueillir une tension de sortie abaissée et stabilisée qui peut être fixe ou variable.

Il existe ainsi des régulateurs de tension fixe ou variable et, dans le cas présent, seul le régulateur de tension fixe sera utilisé. Il se présente sous la forme d'une puce à trois pattes permettant de disposer d'une tension régulée à + ou - 5 % sous 1 A (en boîtier TO3), ceci en positif ou en négatif.

La puce régulateur de tension fixe dispose de nombreux avantages :

- ◇ Limitation interne du courant de sortie, évitant la destruction par court-circuit ;
- ◇ Limitation, en fonction de la température, du circuit en cas par exemple de mauvais Refroidissement ;
- ◇ Facilité d'emploi simplifiée à l'extrême.

Il faudra simplement prévoir en entrée une tension U_0 tel que :

U_0 supérieur ou égal à $U_{\text{régulateur}} + 2 \text{ à } 3 \text{ V}$

en ne dépassant pas 35 V, pour $U_{\text{régulateur}} < 18 \text{ V}$;

en ne dépassant pas 40V, pour $U_{\text{régulateur}} > 20\text{V}$.

Ces conditions énumérées ci-dessus correspondent exactement au problème à savoir que la tension d'entrée du régulateur est de $\pm 15 \text{ V}$ et celle de sortie devra se limiter à $\pm 12\text{V}$. Les circuits 7812 et 7912 seront alors utilisés, les premiers fournissant la tension positive et les seconds la tension négative pour l'alimentation symétrique des conditionneurs.

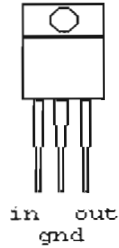
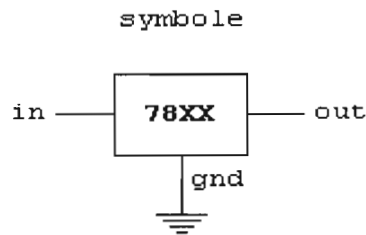
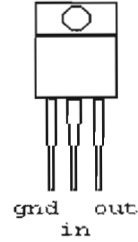
Régulateurs fixes positifs**Régulateurs fixes négatifs**

Figure VII.3 : circuits et symbolisation des régulateurs de tension

VII.4 Solution adoptée

La première solution, à savoir l'utilisation du pont diviseur de tension, est rendue impossible par l'instabilité issue de la trop grande chute de tension créée par les charges constituées par les capteurs.

La deuxième solution nécessite un montage plus compliqué avec l'utilisation de beaucoup plus de composants électroniques notamment des résistances et des amplificateurs opérationnels. Et d'ailleurs, l'amplificateur inverseur ne pourra pas être utilisé comme abaisseur de tension puisque son gain est toujours supérieur à l'unité.

Il reste enfin la troisième solution qui a été retenue grâce à sa simplicité (mise en œuvre très facile), sa grande fiabilité et son caractère peu onéreux.

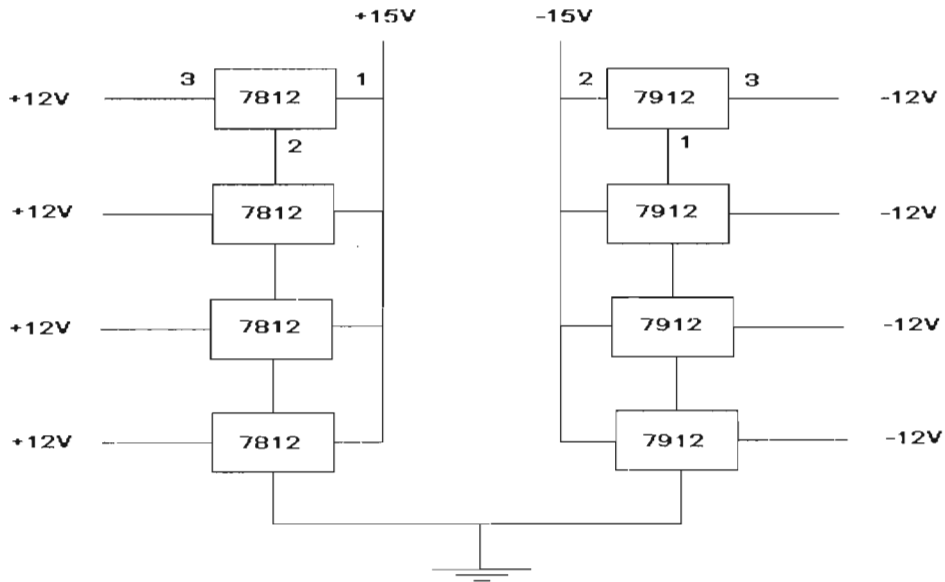


Figure VII.4 : schéma de câblage des régulateurs de tensions

Une fois les circuits conditionneurs alimentés, les signaux délivrés par les capteurs peuvent être recueillis et affichés sur l'écran du logiciel labtech. Il faut préciser que les connexions se feront de façon différentielle.

Voir le schéma de câblage général à la figure VII.5.

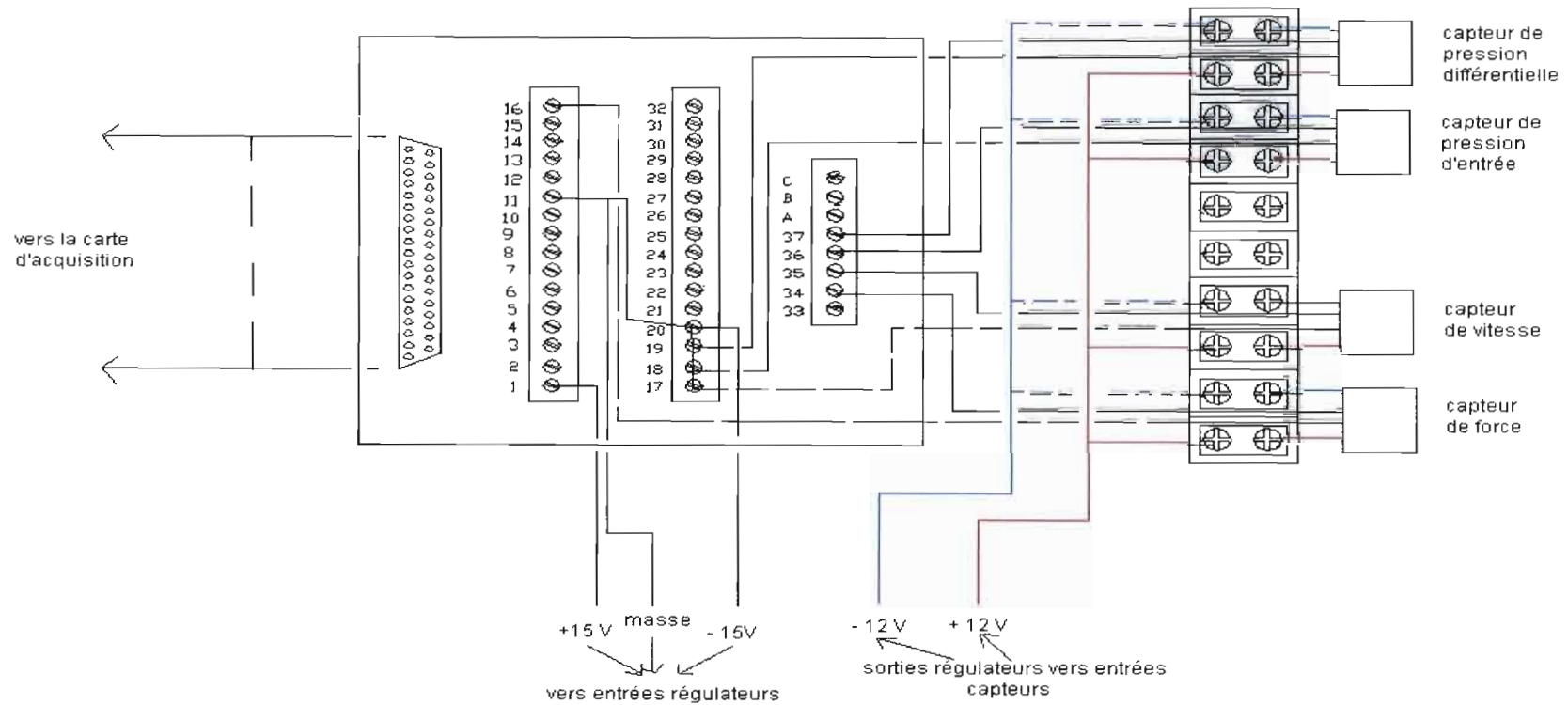


Figure VII.5 : schéma général de câblage des capteurs

VIII. LE LOGICIEL D'ACQUISITION ET D'AFFICHAGE :LE LABTECH

Les logiciels constituent des interfaces entre l'opérateur et le système d'instrumentations, garantissant l'acquisition et l'affichage de paramètres bien définis. Une donnée lue par le microprocesseur doit pouvoir être traitée, soit par un programme soit par un logiciel, et présentée sous une manière conviviale à l'exploitant. Le logiciel « Labtech Notebook » en intégrant le logiciel « Real Time Vision » permet d'assurer ces deux tâches.

VIII.1 Le logiciel Labtech Notebook

Le Labtech est un logiciel facile d'utilisation développé sous l'environnement Windows. Il utilise les mêmes conventions que celles de Microsoft Windows à savoir le système de menus, les boîtes de dialogue, les fenêtres, les icônes . . . Son environnement se compose de deux parties : le système d'icônes et le système de menus.

VIII.1.1 Le système de menus

Le système de menus nous donne accès à toutes les possibilités offertes par le Labtech. Le menu éclaté est présenté à la figure VIII.1.

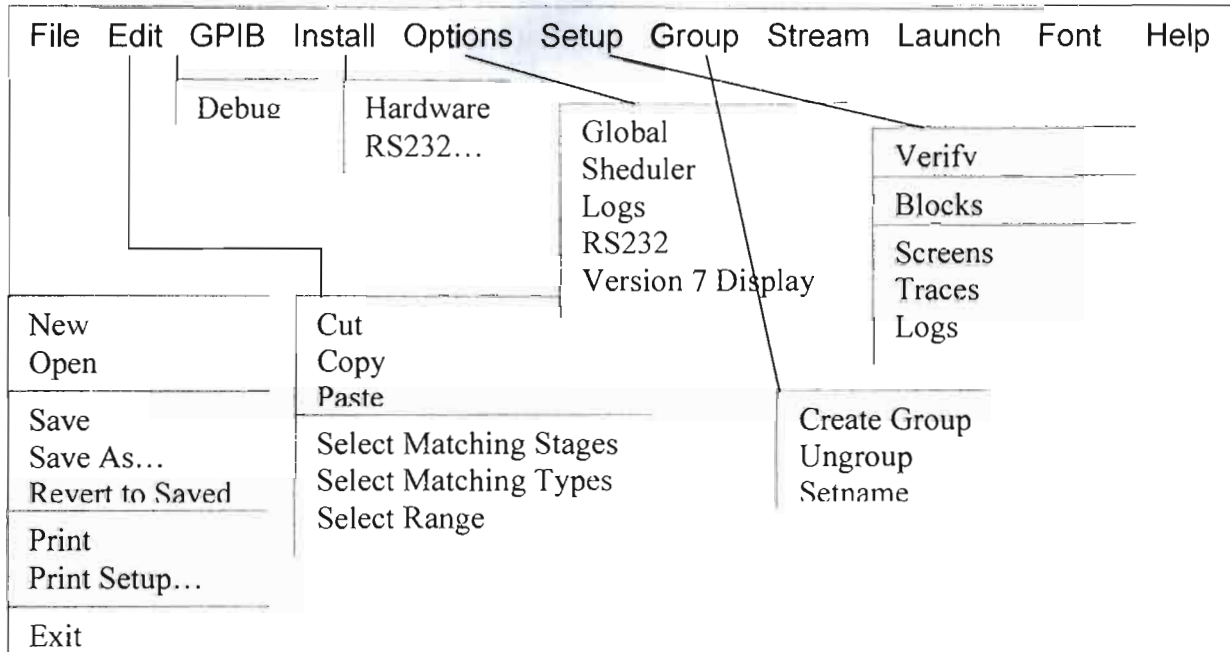


Figure VIII.1 : Atlas du menu de Notebook

VIII.1.2 Le système d'icônes

Le système d'icônes du Labtech est une interface graphique qui, grâce au déplacement et à la connexion des icônes, permet de créer facilement un programme d'acquisition et de contrôle. L'acquisition de données par ordinateur est ainsi passée des programmes, écrits dans un langage, au système d'icônes en passant par le système de menus. L'interface basée sur les icônes procure un aspect visuel de l'application contrairement au système de menus orienté vers un contexte textuel.

Grâce au système d'icônes, l'accès est donné à toutes les fonctionnalités offertes par le logiciel en une place. Ce système nous donne une image de notre application, montrant les relations et les transferts de données entre les blocs.

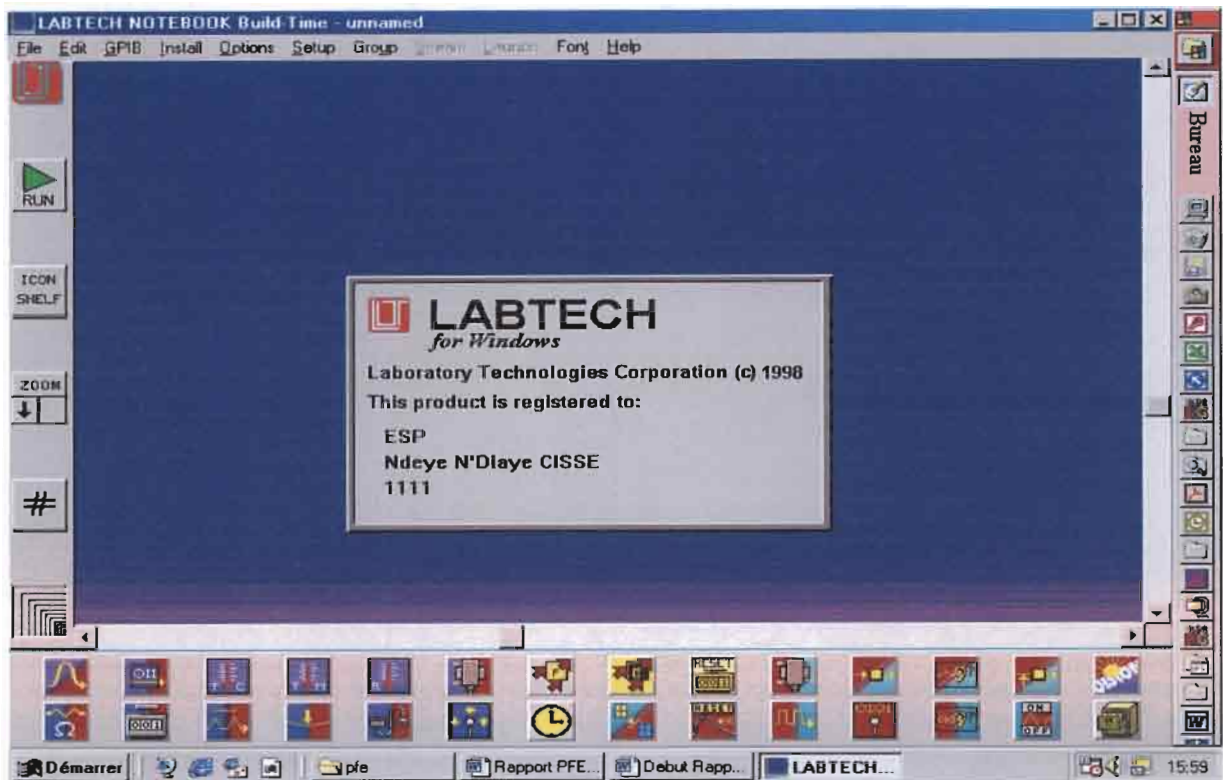


Figure VIII.2 : l'interface graphique de Labtech Notebook

- Les icônes

Une icône est une représentation symbolique d'un bloc, d'une visualisation ou d'un fichier. Un exemple de bloc que nous aurons à utiliser dans notre application est le bloc d'entrée analogique (Analog Input).



Ce bloc représente, dans une application, une donnée analogique collectée depuis la carte d'acquisition de données. A chaque icône, est associée une boîte de dialogue pour la configuration de la fonction qu'elle représente. Ainsi, en double cliquant sur le bloc Analog Input, la boîte de dialogue de la figure VIII.2 apparaît.

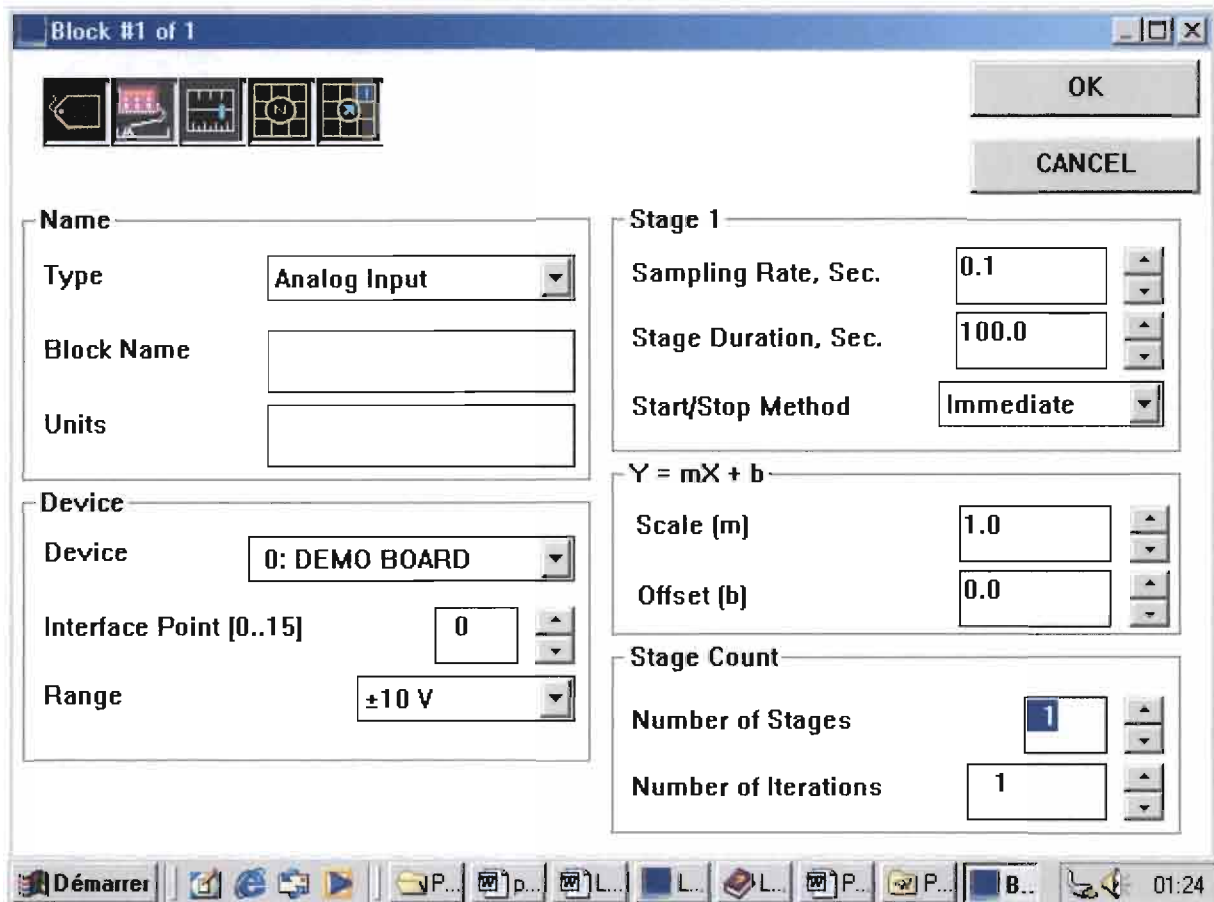


Figure VIII.3 : Fenêtre de configuration d'un bloc

❖ **Le sous-menu Name :**

Le sous-menu Name permet de choisir le type du bloc en question, de donner son nom dans l'interface de traçage et de préciser l'unité avec laquelle les données seront affichées.

❖ **Le sous-menu Device :**

Cette rubrique permet de choisir la carte concernée par l'acquisition, le canal de la carte qui correspond à la mesure ainsi que la plage de tension dans laquelle les mesures seront comprises.

❖ **Le sous-menu Stage 1 :**

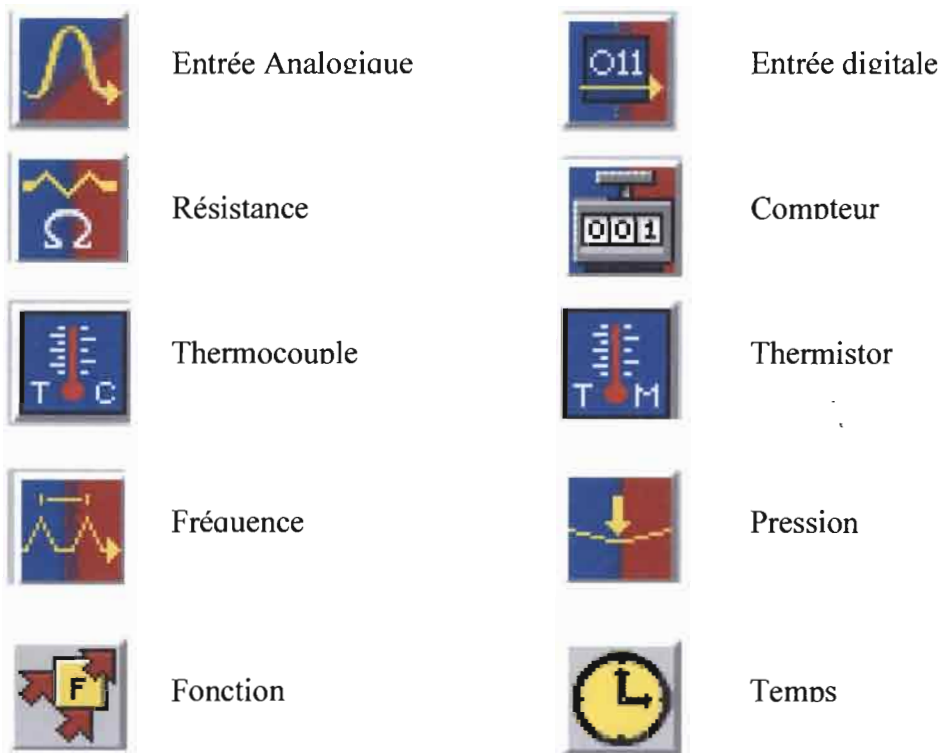
Ce menu sert à définir le taux de l'échantillonnage, la durée totale de la mesure des données correspondantes au bloc en question et enfin les modalités de marche et d'arrêt de la collecte.

❖ **Le sous-menu Stage Count :**

Ce menu permet de définir le nombre de campagne de collecte à faire tout en précisant le nombre d'itérations pour chaque collecte.

▪ Les types d'icônes

En plus de l'icône analogique étudiée ci-dessus, il existe plusieurs autres types d'icônes montrés ci-dessous :



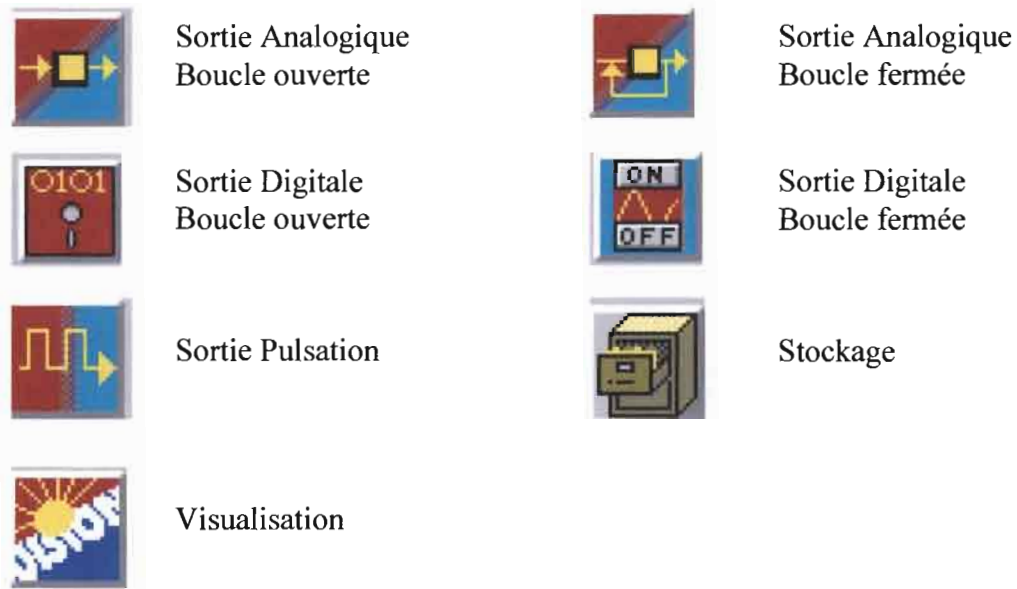


Figure VIII.4 : Les différents types de blocs existant dans le Labtech

VIII.2 Le logiciel Real Time Vision

Le Real Time Vision pour les systèmes d'exploitation de Microsoft Windows permet de :

- ❖ Visualiser graphiquement, à temps réel et de manière dynamique, les informations telles qu'elles évoluent dans une autre application Windows,
- ❖ Créer des synoptiques de visualisation en utilisant une variété d'outils de dessin et d'animation,
- ❖ Changer les informations dans d'autres applications Windows.

VIII.2.1 Système requis

Le Real Time Vision fonctionne dans le Microsoft Windows 3.1 ou un environnement plus récent. Il est recommandé un processeur 486DX, 8 Mo de RAM et un accélérateur graphique de Windows.

VIII.2.2 La fenêtre de Vision

Cette fenêtre est constituée de quatre différentes parties : la barre de titres, la barre de menus, la barre d'outils et la partie pour le design du procédé.

A l'instar du Labtech, il utilise les mêmes conventions que celles de Microsoft Windows à savoir le système de menus, les boîtes de dialogues, les fenêtres, les icônes . . .

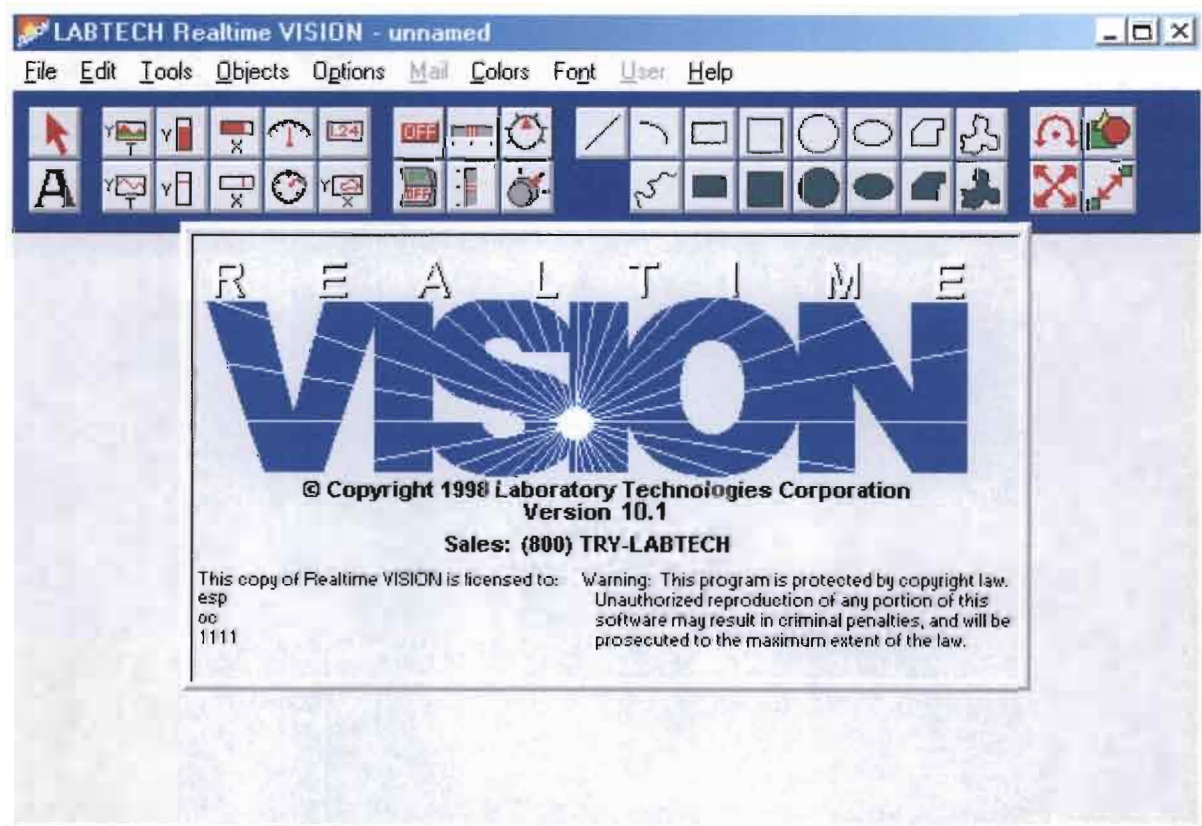


Figure VIII.5 : Fenêtre de visualisation de Real Time Vision

VIII.2.3 Les objets de Vision

Concernant le « design » dans le « Vision », cinq types d'objets peuvent être utilisés à savoir les objets de visualisation, les objets de contrôle, les objets textes, les objets de dessin et les objets d'animation.

- Les objets de visualisation

Les objets de visualisation sont des graphes, des mètres et des cadrans. Ces objets permettent de voir, à temps réel, les changements de paramètres.

La figure ci-dessous montre les différents objets de visualisation. Chaque objet a un format par défaut. Ainsi, en double cliquant sur un objet, on obtient une boîte de dialogue pour changer la configuration de l'objet en question.

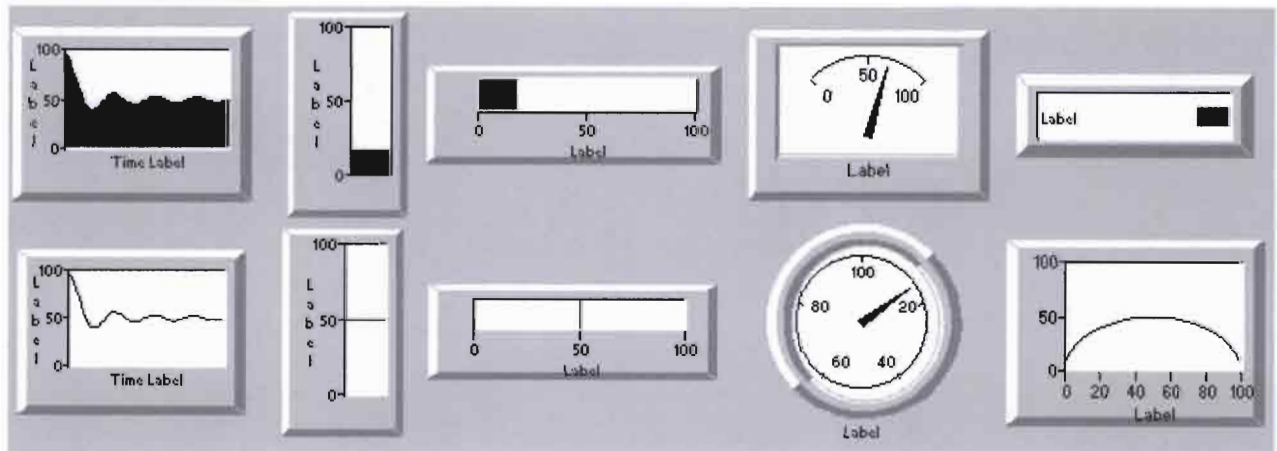


Figure VIII.6 : Les objets de visualisation

- Les objets de contrôle

Les objets de contrôle sont des boutons poussoirs, des cadrans, des sélecteurs, des interrupteurs, des boutons ON/OFF ... Chaque objet a un format par défaut. Ainsi, en double cliquant sur un objet, une boîte de dialogue apparaît et autorise le changement de configuration de l'objet.



Figure VIII.7: Les objets de contrôle

- Les objets texte

Les objets texte sont des chaînes de caractères q'on insère dans le design du procédé. Les objets texte sont créés en utilisant l'outil texte sélectionné depuis le menu outil.

- Les objets dessin

Les objets dessin sont des lignes, carrés, rectangles, cercles, ellipses et polygones. Les objets dessin disponibles dans la barre d'outil sont représentés sur la figure ci-dessous :

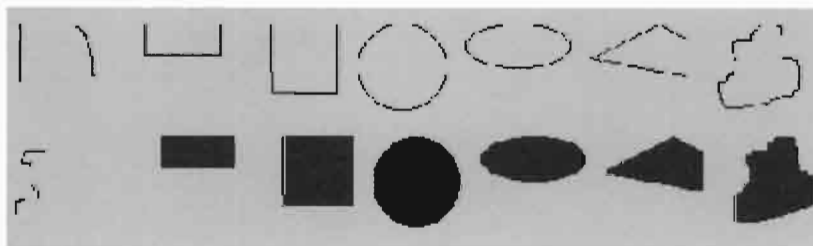


Figure VIII.8 : Les objets dessin

Ainsi avec ces formes de base, des dessins complexes peuvent être réalisés en groupant des composants individuels grâce à la commande « Group » dans le menu « Options ».

- Les objets d'animation

Les objets d'animation sont des objets dessins qui bougent, tournent, changent de couleurs, se réduisent et augmentent. Les objets dessins se transforment en objet d'animation grâce à l'utilisation l'outil « Animation » de la barre d'outil ou en sélectionnant « Animation tools » dans le menu « Objects ».

IX. PROGRAMMATION GRAPHIQUE DE L'INSTALLATION ET TESTS SUR LES DONNEES A ACQUERIR

IX.1 Programmation graphique pour l'acquisition de données

La programmation graphique permet de configurer, par l'intermédiaire de blocs, les paramètres qui assurent l'acquisition et l'affichage de données. Elle permet également, grâce au logiciel « Vision », de schématiser l'installation de sorte qu'elle puisse être visualisée simultanément avec les données lors du déclenchement du procédé d'acquisition et d'affichage de ces dernières.

Les fenêtres créées pour notre application sont ainsi représentées ci-dessous :

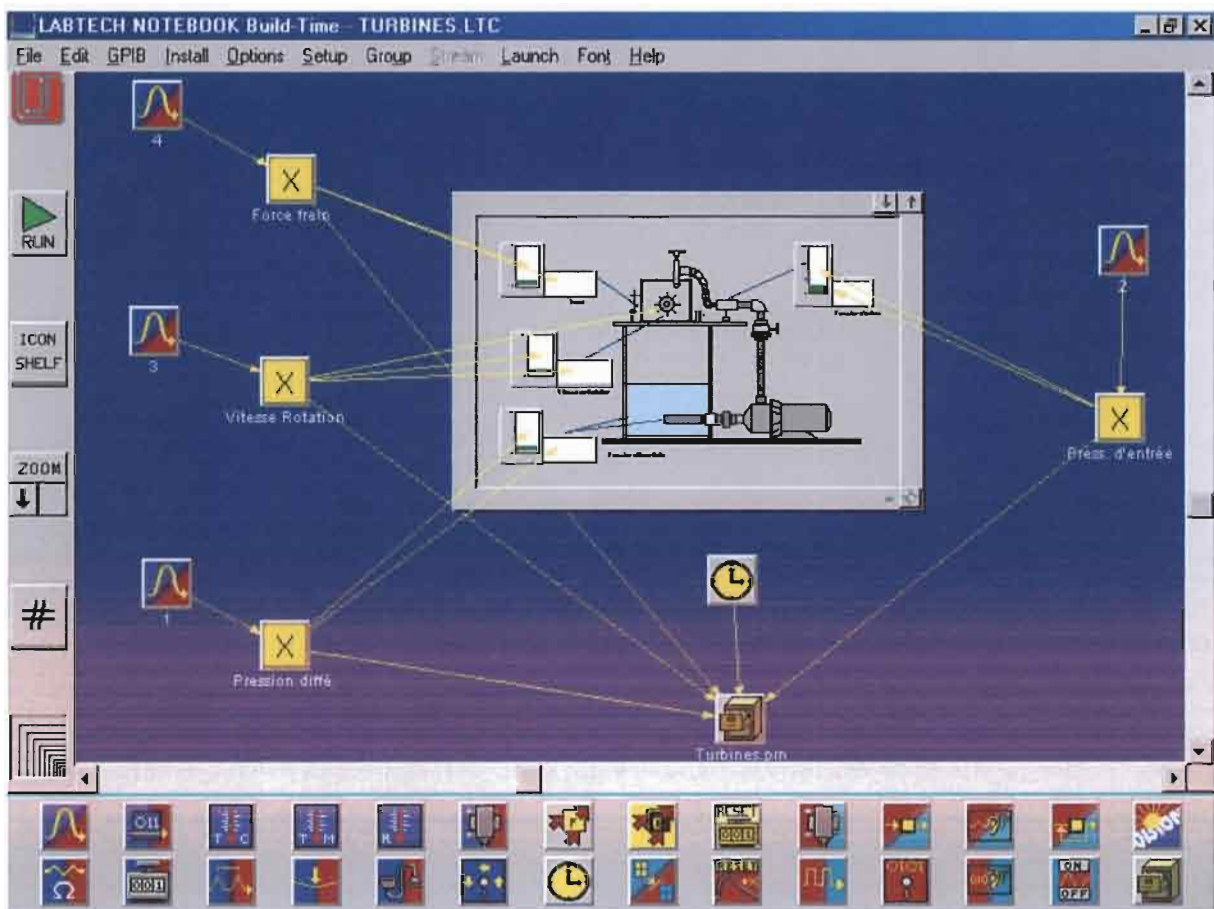


Figure IX.1 : Résultats de la programmation graphique sur labtech
avec en arrière-plan le schéma réalisé sur le « Vision »

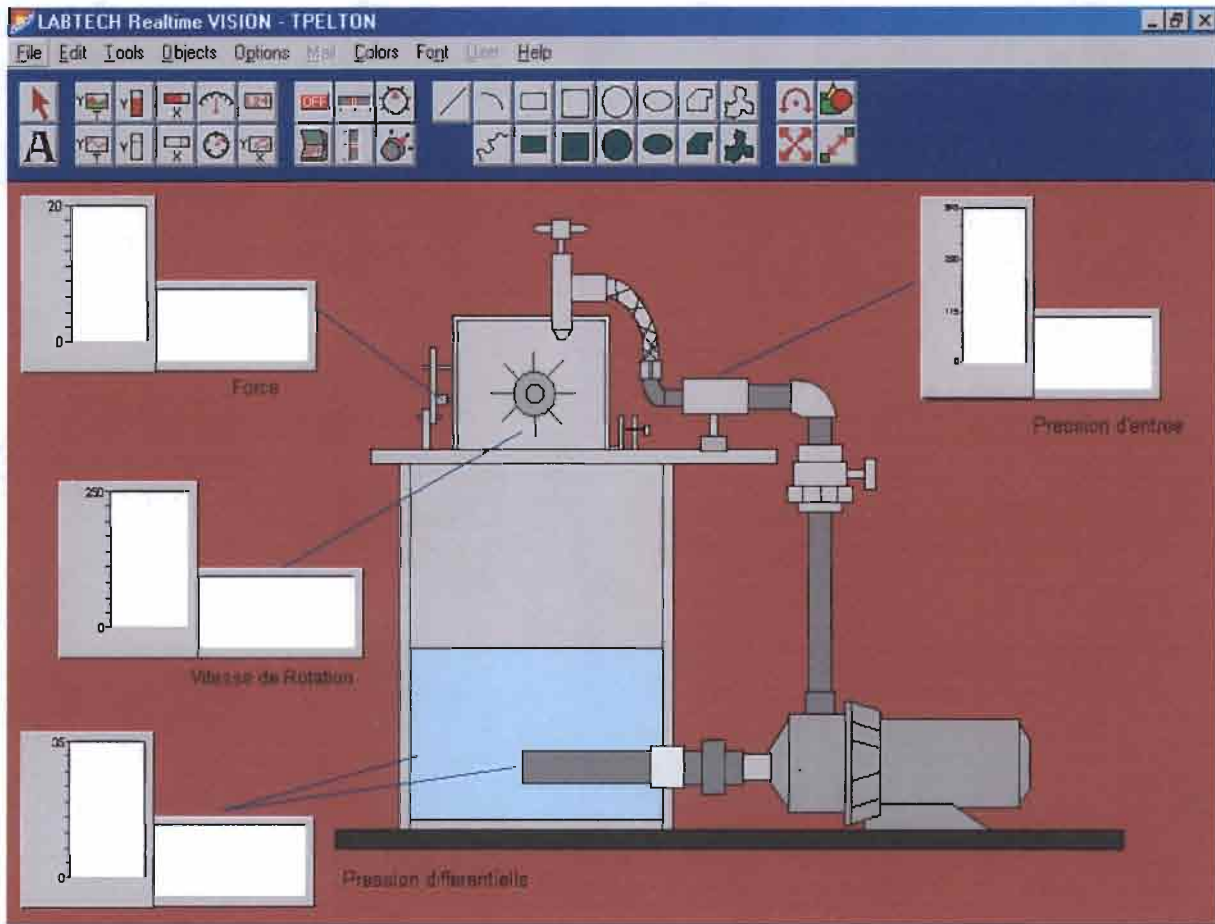


Figure IX.2 : Fenêtre affichée lors de l'acquisition

IX.2 Tests

Ils ont été réalisés séparément sur chaque paramètre en considérant d'abord la vitesse de rotation qui utilise le canal CH2. Cette dernière, à l'arrêt quand le rotor est fixe, prend une valeur d'offset de 2 Hz. En marche, elle évolue de façon égale à la vitesse réelle du système avec un décalage égal à l'offset (vérification par un stroboscope). Ce qui démontre que notre système d'acquisition mise en place fonctionne.

Le deuxième test réalisé sur le canal CH1, donnant la pression d'entrée, révèle une variation faible de cette dernière.

L'ensemble de ces tests réalisés débouche sur les premières conclusions suivantes :

- ➔ le nouveau système d'acquisition mis en place fonctionne correctement ;
- ➔ les données recueillies semblent indiquer les mêmes résultats que l'ancien logiciel, ce qui corroborerait l'idée que la défection du système proviendrait du matériel (conditionneurs ou (et) capteurs) ;
- ➔ néanmoins, les données font des pulsations et sont ainsi difficiles à appréhender de visu.

Toute tentative d'interprétation est alors prématurée .

Il faudrait donc passer à l'étape d'enregistrement, d'affichage et de traitement des données en vue de visionner directement les données recueillies sous forme de tableaux et éventuellement les graphes relatifs au système de turbines (couple, puissance mécanique et rendement en fonction de la vitesse de rotation).

Ensuite, en comparant les courbes normales qui devraient être obtenues et les courbes réellement enregistrées, une conclusion pourra être tirée sur l'état du matériel du système.

X. DEVELOPPEMENTS SUR EXCEL

Il va s'agir, à cette étape, d'élaborer un programme qui va permettre :

- de recueillir automatiquement les données obtenues lors du déclenchement du système d'acquisition ;
- d'enregistrer ces données sous un fichier Excel avec une bonne mise en forme ;
- de calculer, pour chaque série de données enregistrées, les valeurs correspondantes de débit massique, de débit volumique, de hauteur de charge, de couple, de puissance hydraulique, de puissance mécanique et de rendement ;
- de tracer les différents graphiques donnant les paramètres du couple, de la puissance mécanique et du rendement en fonction des différentes vitesses de rotation enregistrées.

Le programme dont il est question va être élaboré grâce au langage « Visual Basic For Applications (VBA)» qui est un langage de programmation inclus sous l'environnement Excel. Le programme écrit est donné intégralement à l'annexe AP1.

Chaque phase du programme élaboré est explicitée ci-dessous :

X.1 Enregistrement des données sous Excel et mises en forme

Lorsque le « Runtime » du labtech est lancé, les données sont enregistrées dans le fichier Notepad nommé Turbines.prn. La première action du programme élaboré est alors d'ouvrir ce fichier, de le copier, d'ouvrir un nouveau classeur Microsoft Excel et de l'y enregistrer.

Ensuite, le programme ferme le fichier Notepad pour permettre à Labtech d'y écrire de nouvelles séries de données.

Cette partie du programme permettant l'archivage des résultats de l'acquisition est donnée ci-dessous :

```
Workbooks.OpenText Filename:="C:\NB\program\Turbines.PRN",  
Origin:=xlWindows , StartRow:=1, DataType:=xlDelimited, TextQualifier:=xlNone,  
ConsecutiveDelimiter:=False, Tab:=True, Semicolon:=False, Comma:=False , Space:=False,  
Other:=False, FieldInfo:=Array(Array(1, 1), Array(2, 1), Array(3, 1), Array(4, 1), Array(5, 1),  
Array(6, 1)  
Range("A1:E100").Select  
Selection.Copy  
Workbooks.Add  
ActiveSheet.Paste  
Application.CutCopyMode = False  
Windows("Turbines.PRN").Close
```

Le programme passe ensuite à la mise en forme des résultats et les présente sous la forme illustrée à la figure X.1.

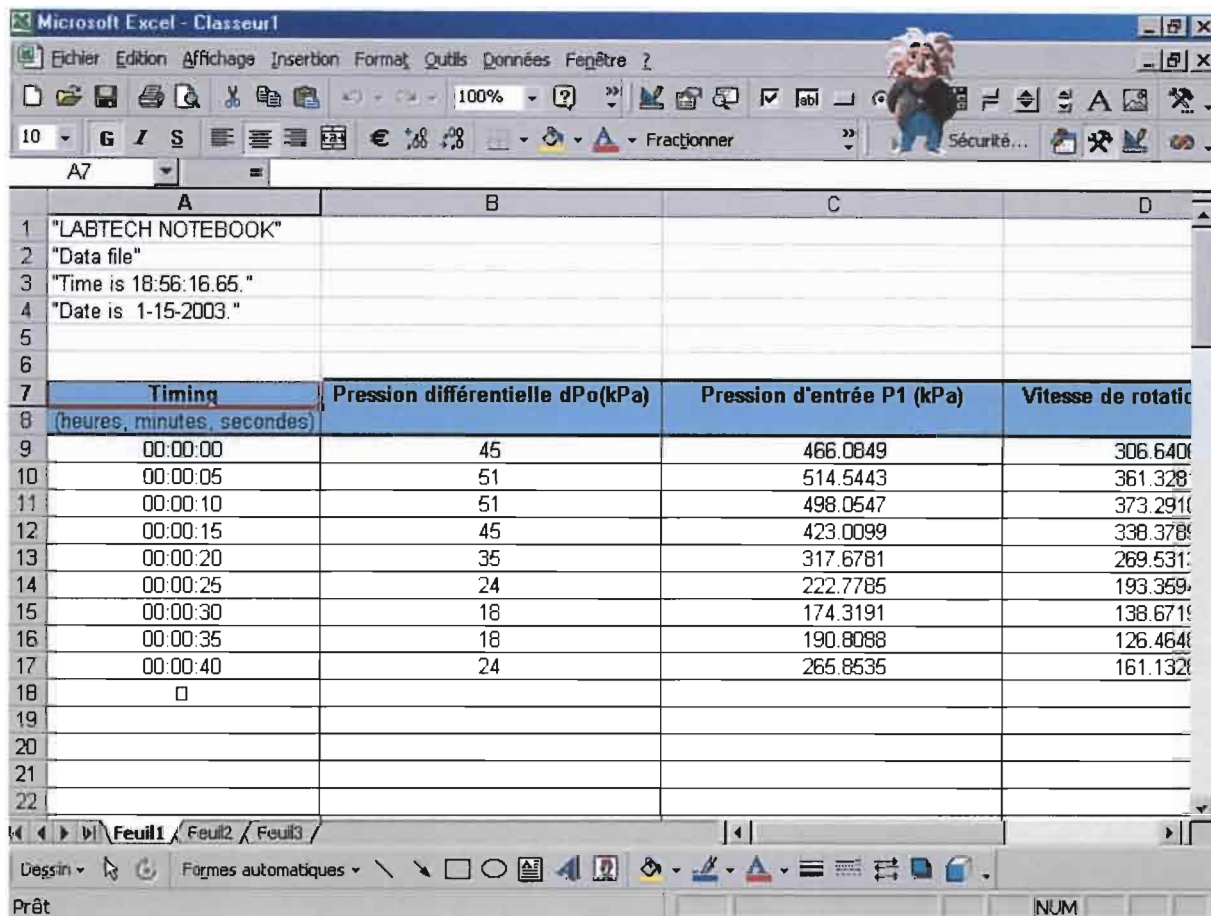


Figure X.1: Fenêtre d'affichage des données enregistrées depuis le labtech et mises en forme (simulation à partir de la carte de démonstration)

Remarque : Le maximum de données à copier, lors d'un échantillonnage, a été fixé à 100 avec une fréquence d'acquisition de 0.2 Hz soit une période d'acquisition de 5 secondes (choisie dans Labtech). Cette fréquence donne au manipulateur le temps de changer les paramètres de fonctionnement de la turbine grâce à l'action sur le robinet et évite des prises de valeurs identiques.

X.2 Calcul des paramètres permettant le tracé des graphiques

Une fois les données enregistrées sur un classeur Excel, le programme doit pouvoir calculer automatiquement les valeurs de débits, de puissances, de couple, de rendement (...), liées aux données acquises par le biais des formules données ci-dessous :

$$Q_m = \frac{\alpha \times 3.14159 \times d^2 \times \sqrt{(2 \times R_{how} \times dPo)}}{4}$$

$$Q_v = \frac{Q_m}{R_{how}}$$

$$H_i = \frac{P_l}{998.2000 \times g}$$

$$P_h = R_{how} \times g \times Q_v \times H_i$$

$$T = F_b \times r$$

$$P_b = 2 \times 3.14159 \times N \times T$$

$$ETA_t = \frac{P_b}{P_h} \times 100$$

Le sous-programme réalisant le calcul automatique possède la structure suivante :

```
Range("F9").Select
```

```
ActiveCell.FormulaR1C1 = "=0.63"
```

```
Range("F9").Select
```

```
Selection.AutoFill
```

```
Destination:=Range("F9:F104"), Type:=xlFillDefault
```

```
Range("G9").Select
```

```
ActiveCell.FormulaR1C1 = "=0.009"
```

```
Range("G9").Select
```

```
Selection.AutoFill
```

```
Destination:=Range("G9:G104"), Type:=xlFillDefault
```

```
Range("H9").Select
```

```
ActiveCell.FormulaR1C1 = "=998.2"
```

Range("H9").Select

Selection.AutoFill

Destination:=Range("H9:H104"), Type:=xlFillDefault

Range("I9").Select

ActiveCell.FormulaR1C1 = "=(RC[-3]*3.14159*RC[-2]*RC[-2]*SQRT((2*RC[-1]*RC[-7]*1000)))/4"

Range("I9").Select

Selection.AutoFill

Destination:=Range("I9:I104"), Type:=xlFillDefault

Range("J9").Select

ActiveCell.FormulaR1C1 = "=RC[-1]/RC[-2]"

Range("J9").Select

Selection.AutoFill

Destination:=Range("J9:J104"), Type:=xlFillDefault

Range("K9").Select

ActiveCell.FormulaR1C1 = "=9.81"

Selection.AutoFill

Destination:=Range("K9:K104"), Type:=xlFillDefault

Range("L9").Select

ActiveCell.FormulaR1C1 = "=(RC[-9]*1000)/(RC[-4]*RC[-1])"

Range("L9").Select

Selection.AutoFill

Destination:=Range("L9:L104"), Type:=xlFillDefault

Columns("L:L").ColumnWidth = 12.14

Range("M9").Select

ActiveCell.FormulaR1C1 = "=RC[-5]*RC[-2]*RC[-3]*RC[-1]"

Range("M9").Select

Selection.AutoFill

Destination:=Range("M9:M104"), Type:=xlFillDefault

Range("N9").Select

ActiveCell.FormulaR1C1 = "=0.024"

Range("N9").Select

Selection.AutoFill

Destination:=Range("N9:N104"), Type:=xlFillDefault

Range("O9").Select

ActiveCell.FormulaR1C1 = "=RC[-10]*RC[-1]*1000"

Range("O9").Select

Selection.AutoFill

Destination:=Range("O9:O104"), Type:=xlFillDefault

Range("P9").Select

ActiveCell.FormulaR1C1 = "=2*3.14159*RC[-12]*RC[-1]/1000"

Range("P9").Select

Selection.AutoFill

Destination:=Range("P9:P104"), Type:=xlFillDefault

Range("Q9").Select

ActiveCell.FormulaR1C1 = "=RC[-1]*100/RC[-4]"

Range("Q9").Select

Selection.AutoFill

Destination:=Range("Q9:Q104"), Type:=xlFillDefault

Mais un problème se pose : Excel utilise le caractère « virgule » pour exprimer les décimaux alors que Labtech utilise le caractère « point ». Donc, l'automatisation des calculs nécessite, en plus, la conversion des caractères « virgule » de Labtech en caractères « point » dans Excel. Le sous-programme ci-dessous le permet :

```
Range("B9:E104").Select
Selection.Replace What=".", Replacement=".", LookAt:=xlPart, _
SearchOrder:=xlByRows, MatchCase:=False
```

La fenêtre d'affichage des données, calculées et mises en forme par le programme, est représentée ci-dessous:

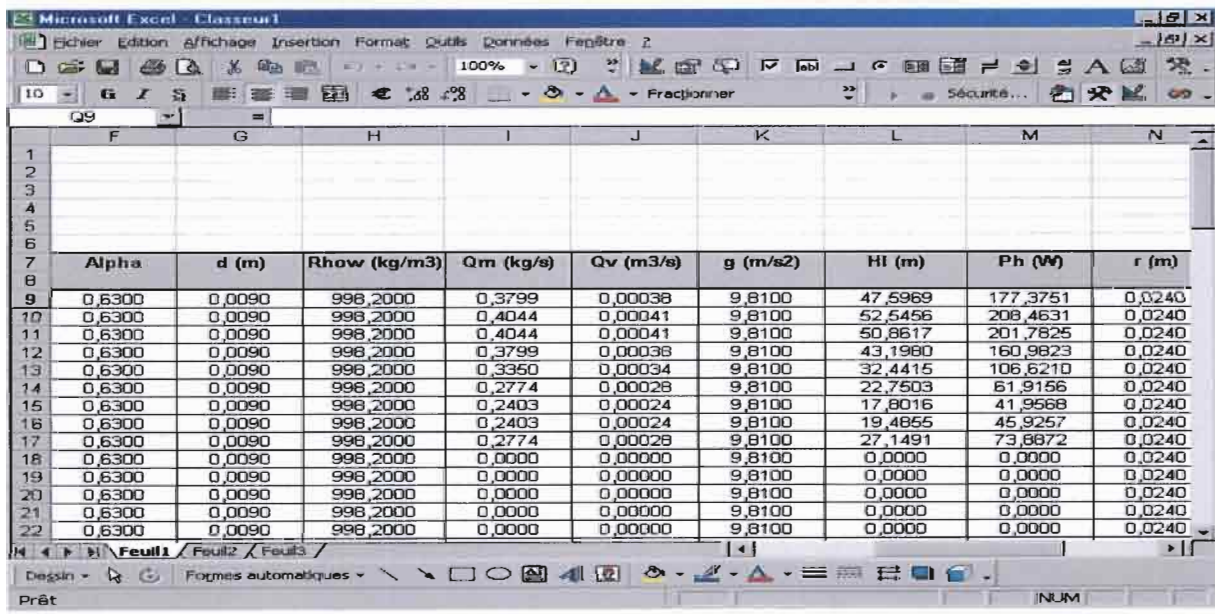


Figure X.2: Fenêtre d'affichage des données calculées

(réalisée à partir d'une simulation)

X.3 Tracé des graphiques

Une fois les paramètres servant au tracé des graphes calculés et présentés sous forme de tableaux, le programme peut passer aux représentations graphiques.

Pour cela, il va afficher une boîte de dialogue pour demander au manipulateur de sélectionner la plage de données d'une colonne bien spécifiée. Cette procédure est nécessaire car le nombre de données maximal est de 100 mais, d'un échantillonnage à l'autre, le nombre de données enregistré change. La boîte de dialogue garantit donc que des cases vides ne sont jamais sélectionnées car le contraire risquerait de donner des courbes erronées.

Des exemples de graphiques, tracés par le programme grâce à des données simulées, sont donnés à l'annexe AP2.

Le programme, ainsi élaboré et prêt à fonctionner, permet de passer à la dernière étape du projet, à savoir les tests sur les éventuelles défaillances de fonctionnement du matériel du système de turbines.

XI. DETERMINATION DE L'ETAT DE MARCHE DU MATERIEL COMPOSANT L'UNITE DE TURBINES

Le programme élaboré va nous guider dans cette partie.

La démarche retenue est la suivante :

- monter une turbine et lancer l'acquisition ;
- interpréter les données d'acquisition enregistrées sous Excel ainsi que les éventuelles courbes tracées ;
- procéder de même pour chaque turbine montée.

Les résultats des données acquises, pour chacune des turbines, sont répertoriés sur le tableau ci-dessous (voir un exemple de données enregistrées, lors des tests sur l'unité de turbines, à l'annexe AP3):

Turbines	valeur dPo	valeur P1	valeur N	valeur F
Pelton	fixe (presque nulle)	fixe	varie (normale)	Rien
Impulsion	fixe	fixe	varie (normale)	Rien
Réaction	fixe (maximale)	varie (très légèrement)	varie (normale)	Rien

Tableau XI.1: Résultats des tests réalisés sur les turbines

Vu ces résultats, aucun graphique n'a été tracé par le programme car cela nécessite, pour chaque turbine, l'enregistrement de tous les paramètres caractéristiques du fonctionnement. Ce test révèle un problème du matériel.

La première solution de réparation envisagée a été de faire une recalibration du système en jouant sur les potentiomètres des conditionneurs des capteurs. Mais aucun résultat satisfaisant n'a pu être obtenu (sauf pour le potentiomètre de réglage de la vitesse de rotation). Ce qui pourrait avoir pour origine une détérioration des résistances constituant les potentiomètres.

Le remplacement de ces dernières peut alors être envisagé. Mais cela nécessite du matériel et une manipulation très délicate car les circuits se présentent sous forme de circuits électroniques intégrés, d'un montage délicat, qui exige une minutie garantissant la non-détérioration des composants.

Une fois ce remplacement possible et la recalibration réussie, il faudrait vérifier la bonne marche du système. Si la recalibration n'a toujours pas résolu le problème, il faudrait alors passer à l'étape supérieure, à savoir étudier les moyens possibles pour une intervention sur les autres composants des conditionneurs et éventuellement sur les capteurs eux-mêmes en se munissant de tout le matériel nécessaire.

Il est à préciser que si les matériaux des capteurs, par sollicitation sévère, fonctionnent au-delà de leur domaine de non-destruction alors ils sont endommagés. En effet, le domaine de non-destruction précise les limites que pourront prendre les grandeurs à mesurer et les grandeurs d'influences sans destruction du capteur, mais avec une détérioration certaine et permanente de ses caractéristiques métrologiques. Quand, par accident, un capteur fait une excursion, même de courte durée, dans ce domaine, il est indispensable de procéder ensuite à un réétalonnage complet du capteur. Si l'excursion se produit hors des limites du domaine de non-destruction, l'altération deviendra une destruction pure et simple.

CONCLUSION

Ce projet de fin d'études avait pour objet d'étudier toutes les possibilités offertes à la mise en place d'un système d'acquisition et de traitement de données, en remplacement de celui qui était utilisé comme guide des travaux pratiques sur machines hydrauliques. Cet objectif a été atteint grâce à un câblage électronique, à l'utilisation du logiciel Labtech, à la réalisation d'une programmation graphique animée, à un recours au VBA pour une automatisation des tâches d'archivages, de calculs des données acquises et de tracé des graphiques caractérisant le fonctionnement des turbines.

Mais le travail devait également ressortir l'état de marche de l'unité de turbines du point de vue matériel. Ainsi, l'hypothèse, selon laquelle l'unité présentait une défection, a été confirmée. Il reste alors à déterminer les voies et moyens de recourir à d'éventuelles réparations pour pouvoir utiliser la chaîne d'acquisition déjà mise en place, en s'appuyant notamment sur le schéma de réparation donnée dans la dernière partie du projet.

Un manque de temps, et surtout de moyens, a empêché de mener à terme ce dernier travail qui ne faisait d'ailleurs pas parti du cahier de charge initial. Mais une fois cette tâche rendue possible et des résultats positifs obtenus, il ne restera plus qu'à perfectionner le programme écrit sur le VBA et à élaborer un nouveau guide des travaux pratiques sur machines hydrauliques.

Il reste à noter que le projet a quelque peu été rendu difficile par un manque de moyens (travail sur une version téléchargée du Labtech, problèmes de disponibilités d'un ordinateur performant).

Les réalisations pratiques se sont également révélées assez difficiles car chaque solution envisagée devait être testée sur le terrain et, seulement alors, une conclusion pouvait être tirée quant à sa validité et à son applicabilité.

En finalité, le projet s'est révélé très enrichissant parce qu'il aura donné l'occasion d'une combinaison soutenue entre la théorie et les réalisations pratiques.

Pour terminer, on peut dire que les travaux pratiques sont incontournables dans une formation d'ingénieurs, qui se veut efficace et efficiente. Il faudrait donc trouver les voies et moyens nécessaires pour réhabiliter ou acquérir des bancs de travaux pratiques, notamment au laboratoire d'hydraulique. Il faudrait également disposer de logiciels d'acquisition et de traitement de données comme le labtech car cela faciliterait, non seulement les projets de fin d'études des étudiants, mais aussi permettrait de renforcer les travaux pratiques réalisés en informatique industrielle, avec notamment une utilisation réajustée des programmes déjà réalisés lors des anciens projets de fin d'études.

BIBLIOGRAPHIE

- [1] ASCH, Georges et collaborateurs. Acquisition de données du capteur à l'ordinateur,
Paris, DUNOD, 1999, 478 p.
catalogue sur Internet
- [2] MARCHAIS, J.-C. L'amplificateur opérationnel et ses applications,
Paris, 4^e édition révisée et complétée, MASSON, 1986, 273 p.
- [3] COMPUTER'S BOARD, inc. CIO-ADO8 & CIO-ADO8-PGA USER'S MANUAL
Edition 1989, 101p.
- [4] Data Translation, Inc. DT2811 USER MANUAL,
Edition 1985, 108 p.
- [5] Labtech Notebook,
Reference Manual, Manual Rev.-5/91
- [6] CISSE, Ousmane. Implantation d'un système d'acquisition et de traitement de données
pour le suivi d'un rehausseur thermique,
Projet de fin d'études
ESP Centre de Thiès, juillet 2002, 95 p.
- [7] Armfield limited.
PC-interfaced training equipment for students of engineering technology,
Bridge House, West Street, Ringwood, Hampshire, England BH24 1 DY, 33 p.
- [8] Armfield limited. Operating instructions and experiments Capture FM3SU - Water
Turbine Demonstration Unit,
33 p.
- [9] Tamba, Séni. LY, Alioune. Manuel de laboratoire pour Machines Hydrauliques,
Ecole Supérieure polytechnique de Thiès, mars 1994, 84 p.

- [10] Jacobson, Reed. Visual Basic Edition Applications Etape par Etape,
Microsoft Press (la maison d'édition de Microsoft), Microsoft Excel2000, 360 p.
- [11] Cours Machines Hydrauliques,
Ecole Supérieure Polytechnique Centre de Thiès.
- [12] Cours Informatique Industrielle,
Ecole Supérieure Polytechnique Centre de Thiès.
- [13] Cours Electronique analogique,
Ecole Supérieure Polytechnique Centre de Thiès.
- [14] **Sites Internet :**
<http://kudelsko.free.fr>
<http://etronics.fr>

ANNEXES

API : Programme écrit dans le langage VBA

Le programme écrit, permettant l'automatisation de l'archivage et du traitement de données, est exposé dans ce qui va suivre.

```

Sub TurbinesHydrauliques()
Attribute TurbinesHydrauliques.VB_Description = "Macro enregistrée le 20/01/2003 par
Ndeye N'Diaye CISSE"
Attribute TurbinesHydrauliques.VB_ProcData.VB_Invoke_Func = " \n14"
'
' Macro enregistrée le 20/01/2003 par Ndeye N'Diaye CISSE
'
Workbooks.OpenText Filename:="C:\NB\program\Turbines.PRN", Origin:=xlWindows _
, StartRow:=1, DataType:=xlDelimited, TextQualifier:=xlNone, _
ConsecutiveDelimiter:=False, Tab:=True, Semicolon:=False, Comma:=False _
, Space:=False, Other:=False, FieldInfo:=Array(Array(1, 1), Array(2, 1), _
Array(3, 1), Array(4, 1), Array(5, 1), Array(6, 1))
Range("A1:E100").Select
Selection.Copy
Workbooks.Add
ActiveSheet.Paste
Application.CutCopyMode = False
Windows("Turbines.PRN").Close

Range("A5:E100").Select
Selection.Cut Destination:=Range("A9:E104")
Range("A7").Select
ActiveCell.FormulaR1C1 = "Timing"
Range("A8").Select
ActiveCell.FormulaR1C1 = "(heures, minutes, secondes)"
Selection.ColumnWidth = 24
Range("A7:E104").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

```

```
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Range("A7:E8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
Selection.Borders(xlEdgeLeft).LineStyle = xlNone
Selection.Borders(xlEdgeTop).LineStyle = xlNone
Selection.Borders(xlEdgeBottom).LineStyle = xlNone
Selection.Borders(xlEdgeRight).LineStyle = xlNone
Selection.Borders(xlInsideVertical).LineStyle = xlNone
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("B7:B8").Select
```

```

Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("D7:D8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("A7:E8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone

```

```
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("A7:E104").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
```

```
.ShrinkToFit = False
.MergeCells = False
End With
Range("A7").Select
Selection.Font.Bold = True
Range("A7:E8").Select
With Selection.Interior
    .ColorIndex = 37
    .Pattern = xlSolid
End With
Range("B7").Select
ActiveCell.FormulaR1C1 = "Pression différentielle dPo(kPa)"
Range("C7").Select
Columns("B:B").ColumnWidth = 30
ActiveCell.FormulaR1C1 = "Pression d'entrée P1 (kPa)"
Columns("C:C").Select
Selection.ColumnWidth = 30
Range("D7").Select
ActiveCell.FormulaR1C1 = "Vitesse de rotation N(Hz)"
Columns("D:D").Select
Selection.ColumnWidth = 25
Range("E7").Select
ActiveCell.FormulaR1C1 = "Force Fb(N)"
Range("E8").Select
Columns("E:E").ColumnWidth = 15
Range("B7:E7").Select
Selection.Font.Bold = True
Range("A7:E8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
```

```
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
```

```
Range("A7:E104").Select
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
Range("F9:Q104").Select
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Range("F7:F8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("H7:H8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
```

```
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("J7:J8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("L7:L8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
```



```
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("N7:N8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("O7:O8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
```

```
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("P7:P8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("Q7:Q8").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
```

```
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Range("F7:Q8").Select
Range("Q7").Activate
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
ActiveWindow.SmallScroll ToRight:=-8
Range("F7:Q104").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
```

```

.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With

```

```

Range("F7").Select
ActiveCell.FormulaR1C1 = "Alpha"
Range("G7").Select
ActiveCell.FormulaR1C1 = "d (m)"
Range("H7").Select
ActiveCell.FormulaR1C1 = "Rhow (kg/m3)"
Range("I7").Select
ActiveCell.FormulaR1C1 = "Qm (kg/s)"
Range("J7").Select
ActiveCell.FormulaR1C1 = "Qv (m3/s)"
Range("K7").Select
ActiveCell.FormulaR1C1 = "g (m/s2)"
Range("L7").Select
ActiveCell.FormulaR1C1 = "Hi (m)"
Range("M7").Select
ActiveCell.FormulaR1C1 = "Ph (W)"
Range("N7").Select
ActiveCell.FormulaR1C1 = "r (m)"
Range("O7").Select
ActiveCell.FormulaR1C1 = "T (mNm)"
Range("P7").Select
ActiveCell.FormulaR1C1 = "Pb (W)"
Range("Q7").Select
ActiveCell.FormulaR1C1 = "ETA %"
Range("F7:Q104").Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .ShrinkToFit = False
    .MergeCells = False
End With
Range("F7:Q8").Select
Selection.Font.Bold = True
With Selection.Interior
    .ColorIndex = 15
    .Pattern = xlSolid
End With
Columns("H:H").ColumnWidth = 12.14

```

```

Range("F9:I104").Select

```

```

Selection.NumberFormat = "0.0000"
Range("J9:J104").Select
Selection.NumberFormat = "0.00000"
Range("K9:Q104").Select
Selection.NumberFormat = "0.0000"
Range("F7").Select

```

```

Range("B9:E104").Select
Selection.Replace What:=".", Replacement:=".", LookAt:=xlPart, _
    SearchOrder:=xlByRows, MatchCase:=False

```

```

Range("F9").Select
ActiveCell.FormulaR1C1 = "=0.63"
Range("F9").Select
Selection.AutoFill Destination:=Range("F9:F104"), Type:=xlFillDefault

```

```

Range("G9").Select
ActiveCell.FormulaR1C1 = "=0.009"
Range("G9").Select
Selection.AutoFill Destination:=Range("G9:G104"), Type:=xlFillDefault

```

```

Range("H9").Select
ActiveCell.FormulaR1C1 = "=998.2"
Range("H9").Select
Selection.AutoFill Destination:=Range("H9:H104"), Type:=xlFillDefault

```

```

Range("I9").Select
ActiveCell.FormulaR1C1 = _
    "=(RC[-3]*3.14159*RC[-2]*RC[-2]*SQRT((2*RC[-1]*RC[-7]*1000)))/4"
Range("I9").Select
Selection.AutoFill Destination:=Range("I9:I104"), Type:=xlFillDefault

```

```

Range("J9").Select
ActiveCell.FormulaR1C1 = "=RC[-1]/RC[-2]"
Range("J9").Select
Selection.AutoFill Destination:=Range("J9:J104"), Type:=xlFillDefault

```

```

Range("K9").Select
ActiveCell.FormulaR1C1 = "=9.81"
Selection.AutoFill Destination:=Range("K9:K104"), Type:=xlFillDefault

```

```

Range("L9").Select
ActiveCell.FormulaR1C1 = "=(RC[-9]*1000)/(RC[-4]*RC[-1])"
Range("L9").Select
Selection.AutoFill Destination:=Range("L9:L104"), Type:=xlFillDefault
Columns("L:L").ColumnWidth = 12.14

```

```
Range("M9").Select
ActiveCell.FormulaR1C1 = "=RC[-5]*RC[-2]*RC[-3]*RC[-1]"
Range("M9").Select
Selection.AutoFill Destination:=Range("M9:M104"), Type:=xlFillDefault
```

```
Range("N9").Select
ActiveCell.FormulaR1C1 = "=0.024"
Range("N9").Select
Selection.AutoFill Destination:=Range("N9:N104"), Type:=xlFillDefault
```

```
Range("O9").Select
ActiveCell.FormulaR1C1 = "=RC[-10]*RC[-1]*1000"
Range("O9").Select
Selection.AutoFill Destination:=Range("O9:O104"), Type:=xlFillDefault
```

```
Range("P9").Select
ActiveCell.FormulaR1C1 = "=2*3.14159*RC[-12]*RC[-1]/1000"
Range("P9").Select
Selection.AutoFill Destination:=Range("P9:P104"), Type:=xlFillDefault
```

```
Range("Q9").Select
ActiveCell.FormulaR1C1 = "=RC[-1]*100/RC[-4]"
Range("Q9").Select
Selection.AutoFill Destination:=Range("Q9:Q104"), Type:=xlFillDefault
```

```
Range("F7").Select
courbe1
courbe2
courbe3
courbe12
courbe13
```

```
fermerWindows
```

```
End Sub
```

```
Sub fermerWindows()
On Error Resume Next
Windows("TurbinePelton1.xls").Close
End Sub
```

```
Sub courbe1()
'
' courbe1 Macro
' Macro enregistrée le 24/01/2003 par Ndeye N'Diaye CISSE
'
Charts.Add
ActiveChart.ChartType = xlXYScatterSmooth
```

```

ActiveChart.SeriesCollection.NewSeries
ActiveChart.SeriesCollection(1).XValues = Application.InputBox( _
prompt:=" Sélectionnez la feuille1, Entrez (S.V.P) la plage de données de la vitesse de
rotation N (Hz), Validez", Type:=64)
ActiveChart.SeriesCollection(1).Values = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P)la plage de données du couple T
(mNm), Validez", Type:=64)
ActiveChart.Location Where:=xlLocationAsNewSheet

```

```

With ActiveChart

```

```

.HasTitle = True

```

```

.ChartTitle.Characters.Text = "Turbine Pelton"

```

```

.Axes(xlCategory, xlPrimary).HasTitle = True

```

```

.Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = _
"Vitesse de rotation N(Hz)"

```

```

.Axes(xlValue, xlPrimary).HasTitle = True

```

```

.Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "Couple T(mNm)"

```

```

End With

```

```

ActiveChart.Axes(xlValue).Select

```

```

With Selection.Border

```

```

.Weight = xlHairline

```

```

.LineStyle = xlAutomatic

```

```

End With

```

```

With Selection

```

```

.MajorTickMark = xlOutside

```

```

.MinorTickMark = xlOutside

```

```

.TickLabelPosition = xlNextToAxis

```

```

End With

```

```

With ActiveChart.Axes(xlValue)

```

```

.MinimumScaleIsAuto = True

```

```

.MaximumScaleIsAuto = True

```

```

.MinorUnitIsAuto = True

```

```

.MajorUnitIsAuto = True

```

```

.Crosses = xlAutomatic

```

```

.ReversePlotOrder = False

```

```

.ScaleType = xlLinear

```

```

.DisplayUnit = xlNone

```

```

End With

```

```

ActiveChart.Legend.Select

```

```

Selection.Delete

```

```

End Sub

```

```

Sub courbe2()

```

```

' courbe2 Macro

```

```

' Macro enregistrée le 24/01/2003 par Ndeye N'Diaye CISSE

```

```

'Sub courbe1()

```

```

'
```

```
Charts.Add
ActiveChart.ChartType = xlXYScatterSmooth

ActiveChart.SeriesCollection.NewSeries
ActiveChart.SeriesCollection(1).XValues = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données de la vitesse de
rotation N (Hz), Validez", Type:=64)
ActiveChart.SeriesCollection(1).Values = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données de la puissance
mécanique Pb(W), Validez", Type:=64)
ActiveChart.Location Where:=xlLocationAsNewSheet

With ActiveChart
.HasTitle = True
.ChartTitle.Characters.Text = "Turbine Pelton"
.Axes(xlCategory, xlPrimary).HasTitle = True
.Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = _
"Vitesse de rotation N(Hz)"
.Axes(xlValue, xlPrimary).HasTitle = True
.Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "Puissance mécanique Pb(W)"
End With
ActiveChart.Axes(xlValue).Select
With Selection.Border
.Weight = xlHairline
.LineStyle = xlAutomatic
End With
With Selection
.MajorTickMark = xlOutside
.MinorTickMark = xlOutside
.TickLabelPosition = xlNextToAxis
End With
With ActiveChart.Axes(xlValue)
.MinimumScaleIsAuto = True
.MaximumScaleIsAuto = True
.MinorUnitIsAuto = True
.MajorUnitIsAuto = True
.Crosses = xlAutomatic
.ReversePlotOrder = False
.ScaleType = xlLinear
.DisplayUnit = xlNone
End With

ActiveChart.Legend.Select
Selection.Delete

End Sub
```

```
Sub courbc3()
Charts.Add
ActiveChart.ChartType = xlXYScatterSmooth
```



```

ActiveChart.SeriesCollection.NewSeries
ActiveChart.SeriesCollection(1).XValues = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données de la vitesse de
rotation N (Hz), Validez", Type:=64)
ActiveChart.SeriesCollection(1).Values = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données du rendement
ETA (%), Validez", Type:=64)
ActiveChart.Location Where:=xlLocationAsNewSheet

```

```

With ActiveChart

```

```

    .HasTitle = True

```

```

    .ChartTitle.Characters.Text = "Turbine Pelton"

```

```

    .Axes(xlCategory, xlPrimary).HasTitle = True

```

```

    .Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = _

```

```

    "Vitesse de rotation N(Hz)"

```

```

    .Axes(xlValue, xlPrimary).HasTitle = True

```

```

    .Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "Rendement ETAT (%)"

```

```

End With

```

```

ActiveChart.Axes(xlValue).Select

```

```

With Selection.Border

```

```

    .Weight = xlHairline

```

```

    .LineStyle = xlAutomatic

```

```

End With

```

```

With Selection

```

```

    .MajorTickMark = xlOutside

```

```

    .MinorTickMark = xlOutside

```

```

    .TickLabelPosition = xlNextToAxis

```

```

End With

```

```

With ActiveChart.Axes(xlValue)

```

```

    .MinimumScaleIsAuto = True

```

```

    .MaximumScaleIsAuto = True

```

```

    .MinorUnitIsAuto = True

```

```

    .MajorUnitIsAuto = True

```

```

    .Crosses = xlAutomatic

```

```

    .ReversePlotOrder = False

```

```

    .ScaleType = xlLinear

```

```

    .DisplayUnit = xlNone

```

```

End With

```

```

ActiveChart.Lcgnd.Select

```

```

Selection.Delete

```

```

End Sub

```

```

Sub courbel2()

```

```


```

```

' courbel1 Macro

```

```

' Macro enregistrée le 24/01/2003 par Ndeye N'Diaye CISSE

```

```


```

```

Charts.Add
ActiveChart.ChartType = xlXYScatterSmooth

ActiveChart.SeriesCollection.NewSeries
ActiveChart.SeriesCollection(1).XValues = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données de la vitesse de
rotation N (Hz), Validez", Type:=64)
ActiveChart.SeriesCollection(1).Values = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P)la plage de données du couple T
(mNm), Validez", Type:=64)
ActiveChart.Location Where:=xlLocationAsNewSheet

```

```

With ActiveChart
    .HasTitle = True
    .ChartTitle.Characters.Text = "Turbine Pelton"
    .Axes(xlCategory, xlPrimary).HasTitle = True
    .Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = _
    "Vitesse de rotation N(Hz)"
    .Axes(xlValue, xlPrimary).HasTitle = True
    .Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "Couple T(mNm)"
End With
ActiveChart.Axes(xlValue).Select
With Selection.Border
    .Weight = xlHairline
    .LineStyle = xlAutomatic
End With
With Selection
    .MajorTickMark = xlOutside
    .MinorTickMark = xlOutside
    .TickLabelPosition = xlNextToAxis
End With
With ActiveChart.Axes(xlValue)
    .MinimumScaleIsAuto = True
    .MaximumScaleIsAuto = True
    .MinorUnitIsAuto = True
    .MajorUnitIsAuto = True
    .Crosses = xlAutomatic
    .ReversePlotOrder = False
    .ScaleType = xlLinear
    .DisplayUnit = xlNone
End With

```

```
ActiveChart.SeriesCollection.NewSeries
```

```

ActiveChart.SeriesCollection.NewSeries
ActiveChart.SeriesCollection(1).XValues = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données de la vitesse de
rotation N (Hz), Validez", Type:=64)

```

```

ActiveChart.SeriesCollection(1).Values = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données de la
puissance mécanique Pb(W), Validez", Type:=64)
ActiveChart.Location Where:=xlLocationAsNewSheet

```

```

ActiveChart.SeriesCollection(2).Select
ActiveChart.SeriesCollection(2).AxisGroup = 2
ActiveChart.Legend.Select
Selection.Delete
With ActiveChart
    .Axes(xlValue, xlSecondary).HasTitle = True
    .Axes(xlValue, xlSecondary).AxisTitle.Characters.Text = _
    "Puissance mécanique Pb(W)"
End With
ActiveChart.ChartArea.Select

```

End Sub

Sub courbe13()

```

Charts.Add
ActiveChart.ChartType = xlXYScatterSmooth

ActiveChart.SeriesCollection.NewSeries
ActiveChart.SeriesCollection(1).XValues = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données de la vitesse de
rotation N (Hz), Validez", Type:=64)
ActiveChart.SeriesCollection(1).Values = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données du couple T
(mNm), Validez", Type:=64)
ActiveChart.Location Where:=xlLocationAsNewSheet

```

```

With ActiveChart
    .HasTitle = True
    .ChartTitle.Characters.Text = "Turbine Pelton"
    .Axes(xlCategory, xlPrimary).HasTitle = True
    .Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = _
    "Vitesse de rotation N(Hz)"
    .Axes(xlValue, xlPrimary).HasTitle = True
    .Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "Couple T(mNm)"
End With
ActiveChart.Axes(xlValue).Select
With Selection.Border
    .Weight = xlHairline
    .LineStyle = xlAutomatic
End With
With Selection

```

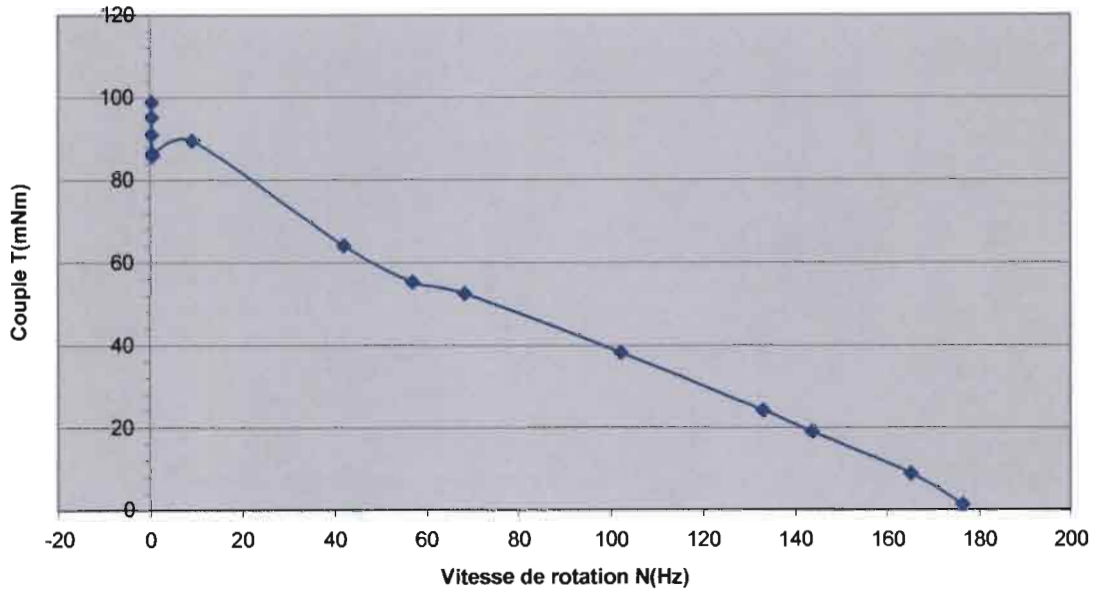
```
.MajorTickMark = xlOutside
.MinorTickMark = xlOutside
.TickLabelPosition = xlNextToAxis
End With
With ActiveChart.Axes(xlValue)
.MinorScaleIsAuto = True
.MaximumScaleIsAuto = True
.MinorUnitsAuto = True
.MajorUnitsAuto = True
.Crosses = xlAutomatic
.ReversePlotOrder = False
.ScaleType = xlLinear
.DisplayUnit = xlNone
End With
.
ActiveChart.SeriesCollection.NewSeries
ActiveChart.SeriesCollection(1).XValues = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données de la vitesse
de rotation N (Hz), Validez", Type:=64)
ActiveChart.SeriesCollection(1).Values = Application.InputBox( _
prompt:=" Sélectionnez la feuille1 , Entrez (S.V.P) la plage de données du rendement
ETA (%), Validez", Type:=64)
ActiveChart.Location Where:=xlLocationAsNewSheet

ActiveChart.SeriesCollection(2).Select
ActiveChart.SeriesCollection(2).AxisGroup = 2
ActiveChart.Legend.Select
Selection.Delete
With ActiveChart
.Axes(xlValue, xlSecondary).HasTitle = True
.Axes(xlValue, xlSecondary).AxisTitle.Characters.Text = _
"Rendement ETAT(%)"
End With
ActiveChart.ChartArea.Select

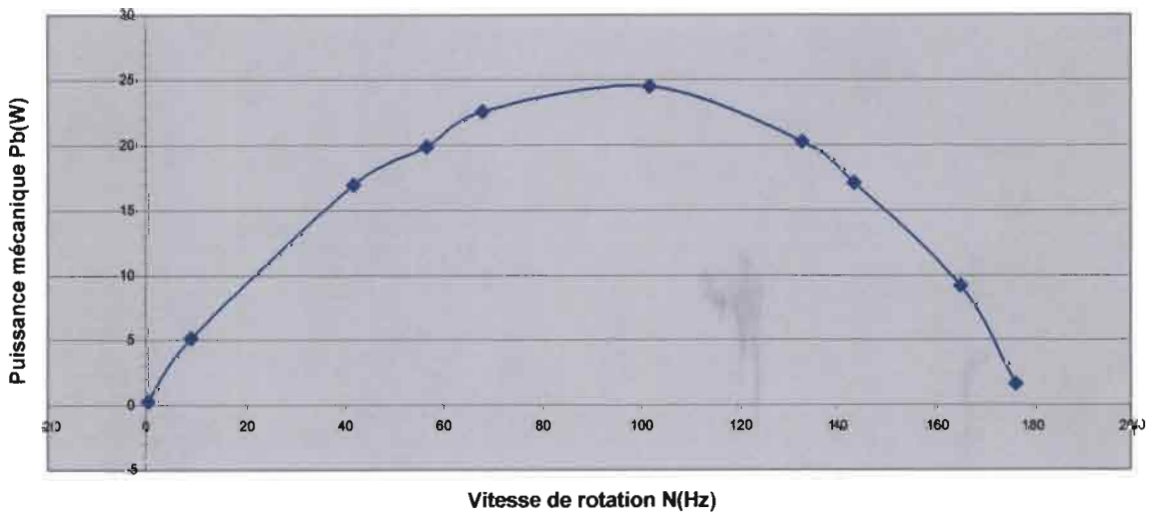
End Sub
```

AP2 : Exemples de graphiques tracés par le programme grâce à l'utilisation des données d'anciens T.P

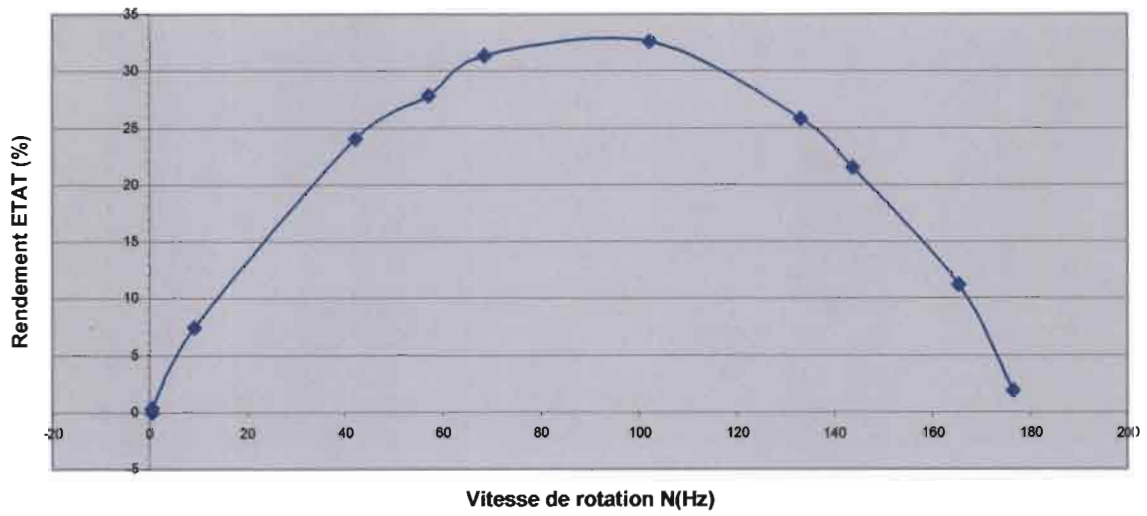
Turbine A Réaction



Turbine A Réaction



Turbine A Réaction



ANNEXE AP3 : EXEMPLE DE DONNEES ENREGISTREES LORS DES TESTS SUR L'UNITE DE TURBINES

"LABTECH NOTEBOOK"

"Data file"

"Time is 11:21:57.89."

"Date is 2-05-2003."

Timing (heures, minutes, secondes)	Pression différentielle dPo(kPa)	Pression d'entrée P1 (kPa)	Vitesse de rotation N(Hz)	Force Fb(N)
00:00:00	69	-33,9889	11,4746	1,6211
00:00:05	69	-34,3254	12,6953	-2,5391
00:00:10	69	-33,9889	12,207	1,8555
00:00:15	69	-34,3254	11,9629	4,2578
00:00:20	69	-34,3254	10,9863	-2,5391
00:00:25	69	-34,3254	10,2539	-2,1484
00:00:30	69	-34,3254	10,2539	1,875
00:00:35	69	-33,9889	10,9863	2,1484
00:00:40	69	-34,3254	12,4512	4,707
00:00:45	69	-34,3254	11,7188	5,5664
00:00:50	69	-34,6619	11,4746	4,9805
00:00:55	69	-34,3254	11,4746	5,3906
00:01:00	69	-34,3254	12,207	4,9609
00:01:05	69	-34,3254	11,7188	6,3086
00:01:10	69	-34,3254	10,2539	6,6992
□				