

MGW32 : Mesugraf pour Windows 32 bits

Pierre DIEUMEGARD
professeur de SVT
Lycée Pothier
F45044 Orléans

courriel : pierre.dieumegard@free.fr

1 Place de MGW32 dans la chaîne de mesure

Comme les autres membres de la famille Mesugraf, MGW32 est un logiciel d'EXAO, c'est à dire d'acquisition et de visualisation de données expérimentales.

Il est conçu pour être compatible avec (potentiellement) toutes les interfaces et appareils de mesure connectables à l'ordinateur, grâce au système de pilotes (« drivers ») d'appareils de mesure, de type « bibliothèques dynamiques » (DLL). Il est aussi compatible avec les tableurs-grapheurs et logiciels de modélisation et d'analyse de données, d'une part par son format de fichiers, d'autre part par copier/coller.

Pour l'instant (mai 2010), certaines fonctions de MGW (Mesugraf pour Windows 16 bits) n'ont pas encore été implémentées.

MGW32 est librement utilisable et copiable, mais il n'est aucunement garanti : soyez prudent dans son utilisation lorsque l'ordinateur est connecté à du matériel dangereux.

Sachez en particulier que les pilotes d'appareils de mesures reliés à des cartes d'extension font des lectures/écritures à diverses adresses-mémoires.

Si par exemple vous choisissez l'utilisation d'un appareil A de mesure dans les options (« matériel principal », ou « matériel auxiliaire »....), qui lit et écrit certaines adresses-mémoires, alors qu'un appareil B (dangereux) est commandé par des valeurs de ces mêmes adresses-mémoire, il est possible que l'appareil B se mette à fonctionner de façon imprévisible.

1.1 Principe d'une chaîne de mesure informatisée

Les mesures physiques permettent d'obtenir une valeur numérique d'une grandeur quelconque à l'aide d'un appareil de mesure. Par exemple le thermomètre permet d'obtenir une valeur numérique de température, un pHmètre une valeur numérique de pH, une balance une valeur numérique de masse...

Dans leur emploi scientifique, les ordinateurs permettent de traiter des valeurs numériques, de les représenter graphiquement à l'écran et sur une imprimante, de les sauvegarder sur un fichier, de calculer des moyennes, des régressions, des dérivées, etc.

Une chaîne de mesure informatisée permet le lien entre les deux paragraphes précédents. L'ordinateur enregistre automatiquement les valeurs mesurées, ce qui permet des mesures qui seraient impossibles ou ennuyeuses autrement. Par exemple, on peut faire plusieurs milliers de mesures en une seconde, puis les analyser tranquillement ensuite. On peut aussi faire des mesures régulièrement pendant une semaine, en vaquant à d'autres occupations, et venir ensuite récupérer les résultats pour les analyser.

Le monde "physique" contient une grande variété de paramètres à étudier, variant de façon continue, alors que le monde "informatique" ne contient que des signaux électroniques de type "bit".

L'interface de mesure (et les capteurs et actionneurs qui y sont associés) permet le lien entre ces deux mondes.

Les interfaces ont fondamentalement quatre types d'entrées et de sorties de signaux "physiques".

Les "entrées analogiques" permettent les mesures de signaux variant de façon continue, par exemple une température entre 0 et 100°C, ou une tension électrique entre - 5 et + 5 volts.

Les "entrées logiques" ou "entrées binaires" font des mesures par tout ou rien : contact ou non contact, sous tension ou hors tension...

Les "sorties analogiques" permettent à l'ordinateur d'influer sur le monde extérieur, en modulant une grandeur continue, par exemple fixer une tension électrique entre 0 et 12 volts, ou une fréquence entre 0 et 1000 hertz..

Les "sorties logiques" (ou "sorties binaires") agissent sur le monde extérieur selon deux valeurs seulement : lampe allumée ou éteinte, moteur en route ou arrêté...

Pour MGW32, qui est un logiciel de mesure, il faut obligatoirement une "Entrée Analogique" (EA), qui donne une valeur numérique à partir de la grandeur physique mesurée. Ceci est possible grâce au Convertisseur Analogique Numérique (CAN), qui convertit un signal électrique continu ("analogique") en signal numérique traitable par l'ordinateur. Selon la qualité du CAN les mesures pourront être plus ou moins précises, plus ou moins rapides...

Éventuellement, MGW32 peut commander des actionneurs du milieu extérieur, soit par une "Sortie Analogique" (SA), qui peut prendre un grand nombre de valeurs, par exemple varier entre 0 et 5 volts par pas de 10 mV, soit par des "Sorties Binaires" (SB) ou "Sorties Logiques" qui déclenchent un relais ou allument une lampe.

Les "Entrées Binaires" (EB) ou "Entrées logiques" ne sont pas utilisées par directement par MGW32.

1.2 MGW32 fait des mesures en fonction du temps

Le temps est une grandeur physique comme les autres, à ceci près que le temps s'écoule sans arrêt, dans un seul sens.

1.2.a Mesure du temps par l'horloge de l'ordinateur

Il est possible de mesurer le temps directement par l'horloge de l'ordinateur, ou bien par l'intermédiaire d'un chronomètre ("timer") présent dans certaines interfaces. MGW32 mesure le temps à partir de l'horloge interne de l'ordinateur et n'a donc pas besoin de circuits électroniques spécialisés dans l'interface. Cette horloge interne permet de mesurer des temps courts, jusqu'à environ 1 microseconde, ce qui suffit amplement pour les usages normaux.

Fondamentalement, MGW32 représente les grandeurs mesurées dans un repère cartésien, avec le temps en abscisse (axe des X) et la ou les autres grandeurs en ordonnée (axe(s) des Y). Il est néanmoins possible de représenter ensuite les autres grandeurs de façon différente, par exemple l'une en abscisse et l'autre en ordonnée,

La succession [prise de mesure/affichage du point à l'écran] est faite de façon différente selon que les mesures doivent être très rapides ou non.

1.2.b Mesures lentes

Avec des mesures lentes (par exemple toutes les secondes), c'est très simple : à chaque mesure, MGW32 affiche le point à l'écran. Entre les mesures, l'ordinateur est disponible pour les autres applications Windows. Par exemple, vous pouvez très bien utiliser un programme de traitement de texte en même temps que MGW32 fait ses mesures.

1.2.c Mesures rapides

Par contre, avec des mesures rapides (par exemple toutes les millisecondes), c'est différent. Mêmes rapides, les ordinateurs ont besoin de temps pour calculer la position d'un point sur le graphique et pour le tracer. Il n'y a donc pas le temps de tracer les points au fur et à mesure, et MGW32 attend d'avoir fini la série de mesures pour afficher tous les points. Dans ce mode, appelé mode "Oscilloscope", l'ordinateur travaille en permanence pour MGW32, et il ne faut pas essayer de vouloir lui faire effectuer une autre tâche en même temps.

1.3 Mesugraf pour Windows peut transmettre les valeurs à d'autres logiciels.

Néanmoins, MGW n'est pas un logiciel de modélisation. Le mieux est d'utiliser pour cela des logiciels spécialisés : tableaux comme OpenOfficeCalc, Gnumeric ou Excel, petits programmes spécialisés comme Curve Expert, ou autres.

1.3.a Sauvegarde des mesures dans un fichier

MGW32 peut utiliser :

- les fichiers-textes, où chaque ligne correspond à une mesure, et où les valeurs numériques (temps, voie principale, et éventuellement voies auxiliaire et supplémentaire) sont séparées par une virgule, un point-virgule ou une tabulation
- les fichiers DIF ("Data Interchange Format")

Ces fichiers sont lisibles par tous les bons tableurs, y compris sur d'autres ordinateurs que les PC. En particulier on peut charger les fichiers issus de MGW32 directement dans OpenOffice, Lotus 1-2-3, QuattroPro pour Windows, Excel, GNumeric.

1.3.b copier/coller dans le presse-papier

Dans la fenêtre de mesure, les options « Copier » du menu « Editer » envoient les mesures dans le "presse-papier", ce qui permet ensuite, dans le tableur, d'utiliser l'option "coller" pour envoyer directement ces mesures dans les cases du tableur.

On peut copier les données en utilisant soit le point soit la virgule comme signe décimal.

En conclusion, MGW32 doit être mis à la place centrale. Il fait les mesures sur l'appareillage physique, mais ne lui demandez pas de faire ce dont votre appareillage n'est pas capable. Il envoie ces mesures vers les logiciels de modélisation, et la qualité de la modélisation dépend de ces logiciels.

2 Installation et lancement de MGW32

2.1 Matériel nécessaire

Bien sûr, il faut obligatoirement un ordinateur "compatible PC", avec l'environnement graphique Microsoft-Windows (en abrégé "Windows") de type Win95, Win98, Win XP ou similaires.

L'ordinateur devra être suffisamment rapide et avoir de la mémoire vive pour que l'utilisation de MGW soit normale : dans la pratique, il faut un processeur au moins de type Pentium à 75 MHz. Si votre ordinateur a des capacités inférieures, MGW devrait fonctionner quand même, mais de façon trop lente pour être agréable. Pour les ordinateurs lents, il vaut mieux utiliser les vieux logiciels :

- MGW (Mesugraf pour Windows 16 bits) peut fonctionner avec Windows 3, et des ordinateurs de type 386 ou 486.
- "Mesugraf" et "Oscillo" pour DOS, qui fonctionnent à partir des processeurs 8088 à 4 MHz, et 512 ko de mémoire vive.

Les systèmes de type WinNT ou WinXP, qui limitent l'accès aux ports de mesure, posent problème pour les cartes à enficher dans un connecteur d'extension (mais les appareils à connexion série, comme Orphy, fonctionnent correctement)

Bien sûr, pour faire des mesures scientifiques, il vous faudra une interface ou un appareil de mesure. Pour l'installation de MGW32, il n'y a pas de nécessité spéciale d'un appareil ou d'un autre : le choix devra être effectué seulement ensuite, au moment de la réalisation des mesures.

2.2 Copie des fichiers dans le répertoire de votre choix

Pour installer MGW32, il faudra un peu de place sur le disque dur, environ 2 méga-octets : environ 700 ko pour le programme mgw32.exe, mais il faut aussi un peu de place pour les fichiers d'aide, les fichiers de pilotes d'appareils de mesure, et, bien sûr, les fichiers de mesures que vous allez réaliser.

Copiez le fichier mgw32.exe dans le répertoire de votre choix, par exemple C:\mesures_scientifiques\.

Lorsque vous désirerez désinstaller MGW32, il suffira d'effacer ces fichiers ou ces répertoires. Pour l'instant, il n'y a pas de procédure automatique de désinstallation.

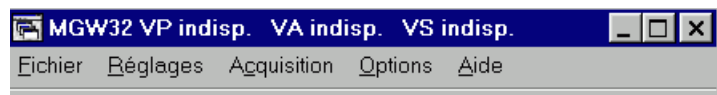
Pour réellement pouvoir faire des mesures, il faut aussi des pilotes d'acquisition de mesure, caractéristiques des appareils de mesure. Leur nom commence par x et se termine par l'extension .dll (par exemple Xcapman.dll). Le mieux est de les recopier soit dans le même répertoire que mgw32.exe, soit dans un sous-répertoire nommé « pilotes » (car MGW32 ira automatiquement explorer ce répertoire).

Copiez les fichiers de pilotes dans un sous-répertoire /pilotes, ou bien dans le même répertoire.

3 Tutoriel : paramétrage pour l'acquisition de données, et réalisation d'une série de mesures

Par défaut, MGW32 va faire 200 mesures, espacées de 1000 millisecondes (c'est à dire une seconde). Si vous voulez changer ces paramètres, utilisez le menu Réglages|Temps

Pour lancer le logiciel, il suffit de double-cliquer sur mgw32.exe.



Il apparaît alors une nouvelle fenêtre, toute petite, en haut de votre écran. C'est le "menu principal" de MGW32.

Ce menu principal est minuscule, c'est vrai, mais il reste visible en permanence à l'écran, c'est à dire que les autres fenêtres de Windows ne le masquent pas. Il vous servira à régler les paramètres de l'acquisition des mesures, et à surveiller l'état des voies de mesures.

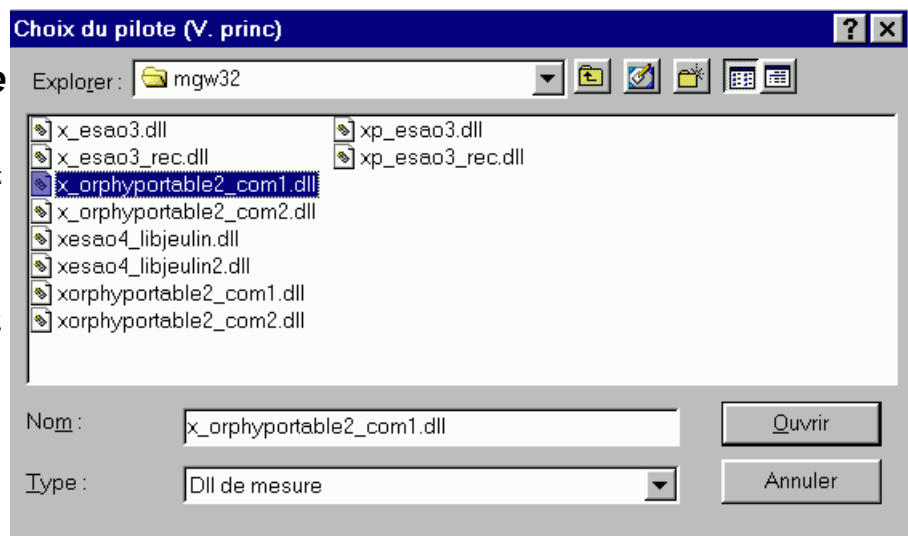
Ne vous inquiétez pas de sa petite taille ! Quand vous lancerez l'acquisition d'une série de mesures, une nouvelle fenêtre apparaîtra, beaucoup plus grande, pour que les points s'affichent de façon claire.

3.1 Réglage de l'interface pour la voie principale

Dans le menu Réglages, choisissez Matériel Voie P.

Il apparaît une boîte avec une liste vide : c'est la liste des voies de mesures possibles sur votre pilote ; comme vous n'avez pas encore chargé de pilote de mesure, c'est normal que cette liste soit vide.

Cliquez sur "Choix pilote" : à ce moment apparaît une nouvelle boîte de dialogue, avec la liste des DLL de mesure présentes dans le sous-répertoire PILOTES : choisissez celle qui correspond à votre matériel.



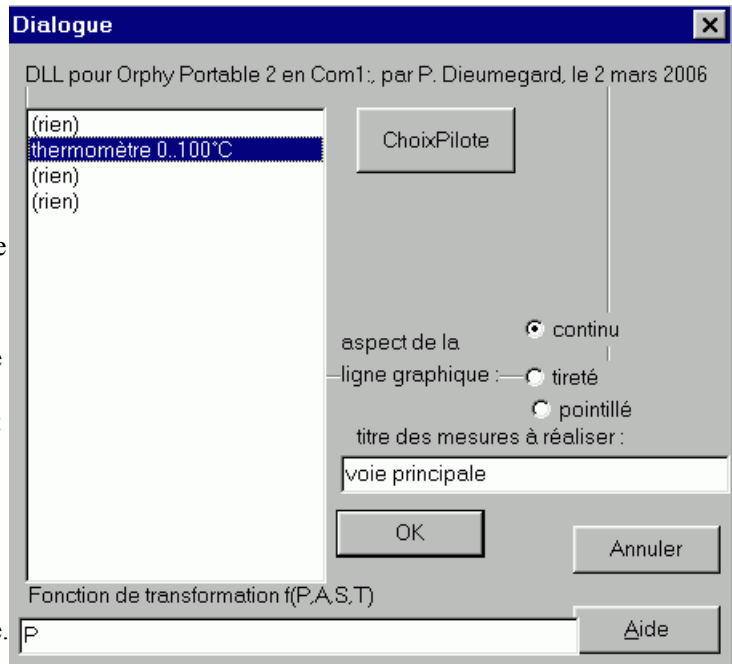
(Si vous êtes simplement en train de tester le logiciel, sans avoir de matériel particulier, vous pouvez prendre QCAPMAN.DLL. C'est le pilote de la prise "manettes de jeu". Si vous avez une vraie manette de jeu, le fait de bouger la manette fera varier les valeurs numériques affichées. Si vous mettez une photorésistance entre les trous 1 et 3 de la prise, ce sera la voie 0, et la variation de lumière fera varier les valeurs numériques affichées...).

Après avoir validé cette boîte, vous constatez que la liste des pilotes s'est remplie. En fonction de votre appareil de mesure, il peut exister entre 1 et plusieurs dizaines de voies de mesure possibles.

Choisissez une voie, puis validez par le bouton OK. A ce moment, le bandeau de la barre de menu change, et au lieu d'afficher VP indisponible, il affiche VP: et une valeur numérique, qui correspond à la grandeur mesurée.

Si vous voulez mesurer sur une deuxième et éventuellement une troisième voie, il faut répéter l'opération en activant le(s) menu(s) Matériel voie Aux, et Matériel voie Supp.

Attention ! pour ces voies, la boîte de dialogue comporte une case à cocher "Prise de mesure ?". Il faut cocher cette case pour que la mesure soit réellement faite sur cette voie. En d'autres termes, vous pouvez choisir un pilote et une voie pour les voies auxiliaires et supplémentaires, sans faire les mesures sur ces voies.



3.2 Réglage des paramètres temporels

Le menu Réglages/temps vous permet de régler l'intervalle entre les mesures qui est par défaut de 1000 millisecondes. Pour un premier essai, laissez-le tel quel.

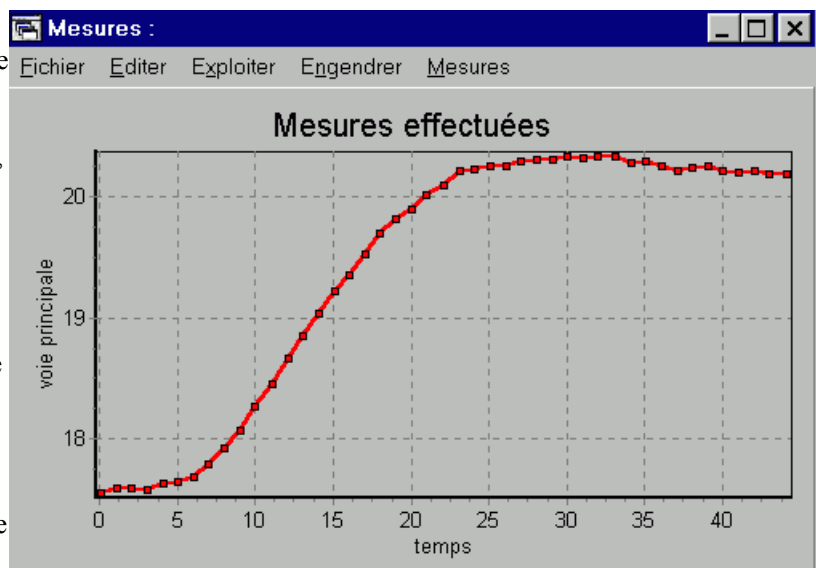
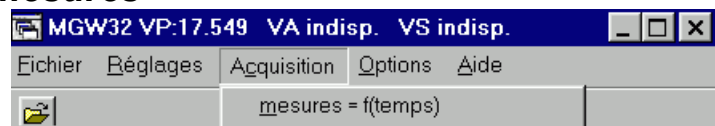
3.3 Acquisition d'une série de mesures

Ouvrez le menu Acquisition, et choisissez mesures=f(temps).

Une fenêtre s'ouvre, et vous pouvez y voir les points de mesure qui se tracent. Cette fenêtre a elle-même son petit menu, qui sera désigné par la suite par "Menu de la fenêtre".

Les points de mesure sont tracés en rouge pour la voie principale. Si vous avez choisi aussi une voie Auxiliaire, les points correspondants sont tracés en vert, et si vous avez choisi une voie Supplémentaire, les points sont tracés en bleu.

Faites varier la grandeur physique mesurée, et constatez que la forme de la courbe tracée varie. Par défaut, la fenêtre est en mode "Autocadrage", aussi bien pour les abscisses que pour les ordonnées, c'est à dire que l'échelle est calculée automatiquement pour que les points soient étalés sur toute la surface de la fenêtre.



Comment arrêter la prise de mesure ? La première façon, bien sûr, est d'attendre que le nombre maximal de mesures soit atteint (ce nombre est fixable dans le menu Réglage/temps). Une autre façon est d'appuyer simultanément sur les touches Contrôle et S (S comme Stop). Une dernière façon est de cliquer à la souris sur le menu de la fenêtre, et de choisir l'option Mesures/Stop.

3.4 Exploitation des données obtenues

Pour mieux voir la série de points, vous pouvez agrandir la fenêtre en cliquant en haut et à droite de celle-ci, sur le triangle ayant la pointe vers le haut. Inversement, si cette fenêtre vous gêne, vous pouvez la réduire ("l'icôner") en cliquant sur le triangle ayant la pointe vers le bas.

Si vous êtes satisfait de votre série de mesure, vous pouvez la sauvegarder sur disque ou sur disquette, par Fichier/sauvegarder. Attention ! il s'agit ici du menu Fichier de la fenêtre, qui permet de sauvegarder les mesures de la fenêtre, et non du menu Fichiers du menu général, qui permet de lire sur disque un fichier, de façon à ouvrir une nouvelle fenêtre.

Le menu Editer permet, grâce aux options Copier d'expédier les données vers le presse-papier, pour les récupérer ensuite par un tableur ou un autre logiciel d'analyse de données.

L'option Enjoliver permet l'ouverture d'une fenêtre de dialogue pour régler les détails du graphique.

Le menu Exploiter n'est pas encore fonctionnel. Il permettra d'une part de faire l'analyse statistique de la (ou des) série(s) de valeurs, et d'autre part la transformation des variables pour la manipulation des données.

Le menu Engendrer n'est pas encore fonctionnel.

Le menu Mesures n'a qu'une option fonctionnelle, « stop », pour arrêter précocement la prise de mesures.

Pour faire un "zoom" sur une portion de l'écran, sélectionnez une portion de l'écran, en déplaçant la souris dans le sens (haut-gauche) vers (bas-droite), tout en appuyant sur le bouton gauche.

Pour revenir à la fenêtre initiale, sélectionnez une portion quelconque de l'écran dans le sens inverse.

4 Les différentes options de menu de MGW32

Il s'agit de la petite fenêtre, réduite pratiquement à son bandeau et à sa barre de menu, qui apparaît lorsqu'on lance MGW32.

Cette fenêtre reste toujours visible, quelle que soit l'application Windows que l'on lance. Elle a comme rôle d'une part d'afficher l'état actuel des voies de mesure, d'autre part de commander les fonctions de MGW32.

Vous pouvez la déplacer avec la souris, en cliquant sur le bandeau et en déplaçant la souris. Vous pouvez l'agrandir ou la rapetisser en cliquant à la souris sur ses limites, et en déplaçant la souris.

4.1 Menu Fichiers

4.1.a Ouvrir

Il permet d'ouvrir un fichier de mesures existant sur disque ou sur disquette. Par défaut, MGW32 sauvegarde et relit ses fichiers dans le sous-répertoire FICHIERS.

Les fichiers de MGW32 sont des fichiers d'extension .CSV, lisibles par les tableurs-grapheurs, mais MGW32 peut aussi lire et écrire des fichiers de type DIF (avec quelques limitations...).

MGW32 peut aussi relire les fichiers de MGW, Mesugraf et TG-Mesugraf pour DOS.

4.1.b Quitter

A effectuer pour quitter MGW32 : Le menu général disparaît alors.

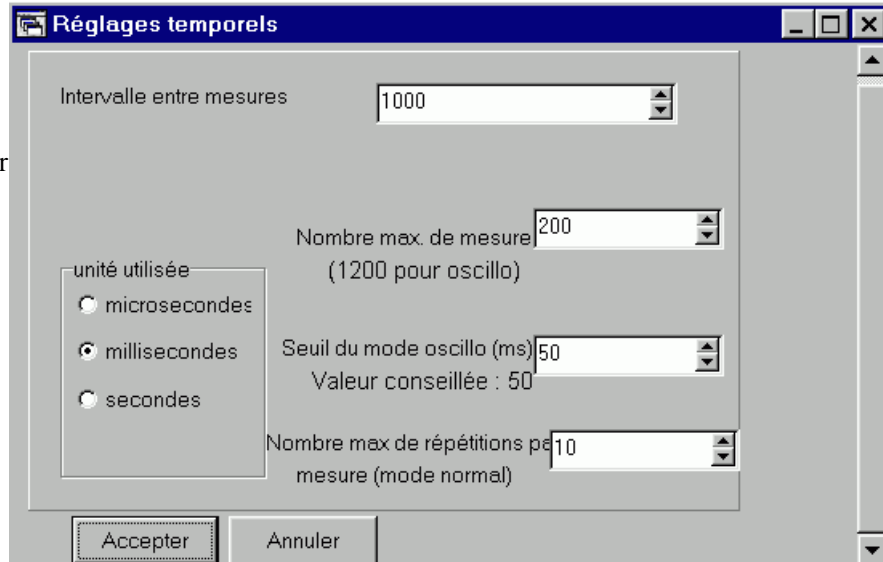
4.2 Menu Réglages

4.2.a Temps

La ligne d'édition "Intervalle entre mesures" permet de choisir une valeur numérique. Comme cet intervalle peut être compris entre une microseconde et plusieurs centaines de secondes, il faut cocher l'une des options situées sous elle pour régler convenablement l'intervalle.

Le nombre maximal de mesures (200 par défaut) peut aller jusqu'à 3000.

Le seuil du mode Oscillo est par défaut de 50 millisecondes. Par expérience, c'est une valeur convenable, et il est déconseillé de le changer si vous n'avez pas de raison valable.



En mode "normal", c'est à dire en mesures lentes, MGW32 rend la main à Windows lorsqu'il ne fait pas de mesure. Ceci permet de bénéficier du mode "multitâche" de Windows, en frappant du texte dans la fenêtre de votre logiciel de traitement de texte pendant que MGW32 fait ses mesures de son côté : le fonctionnement de votre logiciel est à peine ralenti par les quelques instants où MGW32 fait ses mesures.

Si l'intervalle entre les mesures est de 1 seconde (1000 ms), et s'il y a 10 répétitions par mesure, cela signifie que tous les dixièmes de seconde, MGW32 va faire une mesure, et qu'au bout de 1 seconde, il fera la moyenne de ces 10 mesures pour afficher un nouveau point.

4.2.b Matériel voie principale

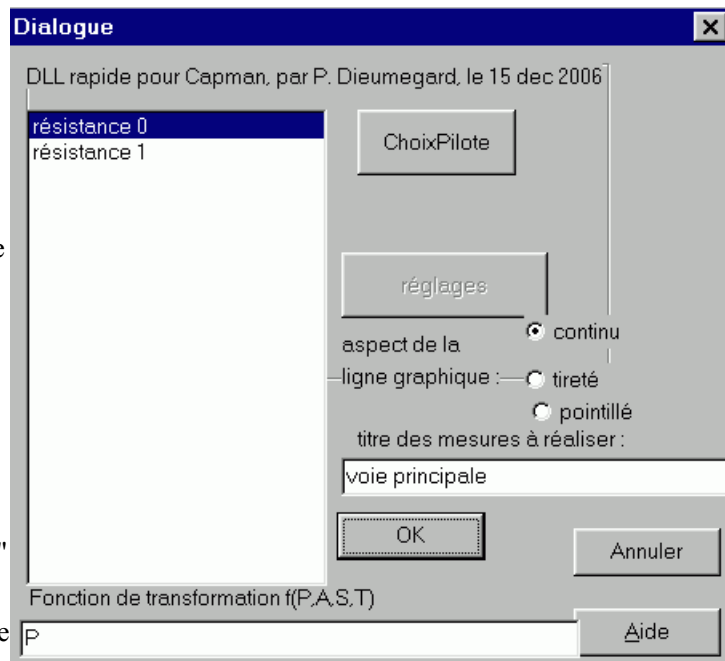
C'est une boîte de dialogue très importante.

Elle comporte une liste des fonctions (entrées analogiques) disponibles dans la DLL de mesure choisie comme pilote pour la voie principale ; bien sûr, tant qu'un pilote n'a pas été choisi, cette liste est vide. Au dessus de cette liste est affiché le détail de l'intitulé du pilote.

Le bouton ChoixPilote permet de choisir le pilote (DLL) correspondant à votre appareil de mesure. Normalement, ces pilotes sont dans le sous-répertoire PILOTES, mais ce n'est pas obligatoire.

Si le pilote choisi comporte une fonction CALIBRATION, un bouton "réglages" permet d'activer cette fonction, ce qui permet des réglages supplémentaires. Si le pilote choisi ne comporte pas cette fonction, ce bouton "réglages" n'est pas activable.

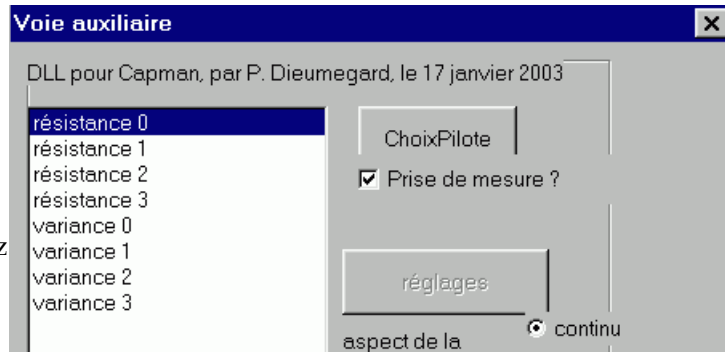
La dernière ligne de la fenêtre de dialogue est une ligne de saisie, qui est la fonction de transformation des valeurs brutes obtenues de la DLL. Vous pouvez l'exprimer en fonction de quatre paramètres :
- P qui est la valeur lue à la voie principale



- A qui est la valeur lue à la voie auxiliaire
- S qui est la valeur lue à la voie supplémentaire
- T qui est la valeur du temps.

Par défaut, c'est simplement P, ce qui signifie qu'il n'y a pas de transformation complexe.

Si le pilote de la voie P vous donne une température en degrés Celsius, et que vous voulez afficher des degrés Fahrenheit (tout est possible !), il faut simplement y entrer la formule de transformation $9 \cdot P / 5 + 32$.



Si vous voulez entrer une formule complexe, pour tenir compte d'une influence du pH lu sur la voie auxiliaire et de la température lue sur la voie supplémentaire, ainsi que d'une dérive progressive dépendant du temps, vous pouvez y entrer une formule du style $P - 0.3 \cdot A + 1.004 \cdot S - T / 4.5$...

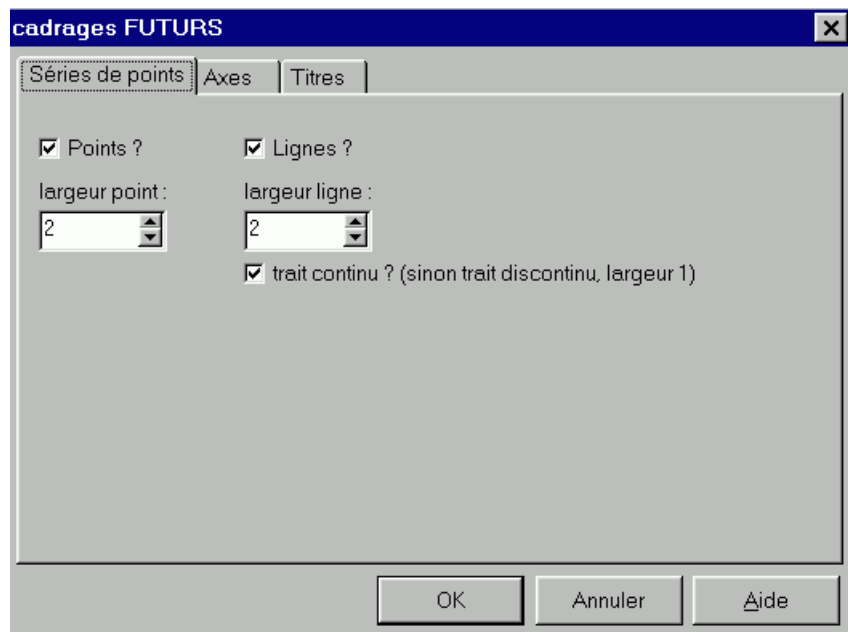
Les détails des formules de transformations sont donnés dans l'annexe IV.

4.2.c Matériel voie Auxiliaire et Matériel voie Supplémentaire

Le principe est le même que pour le choix du matériel en voie principale.

N'oubliez pas de cocher la case "Prise de mesure ?" si vous voulez que ce réglage soit pris en compte.

Les fonctions de transformation par défaut sont simplement "A" pour la voie Auxiliaire, et "S" pour la voie supplémentaire.



4.2.d Cadrages futurs

Cette boîte de dialogue permet de régler l'étendue des abscisses et des ordonnées.

L'axe "Y principal" est tracé à gauche du graphique ; normalement, il correspond à la voie principale.

4.2.e Sorties logiques

Cette option provoque l'apparition d'une boîte de dialogue, où l'on peut choisir un pilote pour les sorties logiques.

Il apparaît une liste de sorties logiques, sous la forme d'un ensemble de cases à cocher. Lorsqu'une case est cochée, l'appui sur "Ok" provoque l'activation de la sortie logique du pilote.

Ceci peut vous permettre de piloter complètement un montage expérimental, par exemple en activant ou inactivant un relais commandant un moteur ou une lampe.

4.2.f Sorties analogiques fixes

Dans le même esprit que les sorties logiques, vous pouvez fixer la valeur de sorties analogiques.

Les sorties analogiques habituelles fixent des tensions de l'ordre de quelques volts. La boîte de dialogue n'autorise qu'au maximum 4 voies, ce qui est amplement suffisant pour les interfaces normales, qui n'ont qu'une ou deux sorties analogiques en général.

4.2.g Sorties analogiques variables

On peut piloter une sortie analogique lors de l'acquisition d'une série de mesures.

Comme d'habitude, il faut choisir le pilote de l'appareil pouvant avoir la sortie analogique variable, ainsi que la voie de cette sortie. Il faut aussi indiquer la formule de transformation dans la ligne de saisie, en fonction des variables P, A, S et T, qui sera prise en compte si on coche la case "Efficace J".

Si la formule ne comporte aucune variable, cela donne une valeur fixe. Par exemple, mettre 3 provoque le passage à 3 volts lors du déclenchement de la série de mesures, alors que mettre 3 pour les sorties analogiques fixes provoque le passage à 3 volts dès que l'on valide la boîte de dialogue.

Si la formule est en fonction de T, la tension variera en fonction du temps. Par exemple mettre $3 + T \cdot 0.1$ fera augmenter la tension de 0,1 V à chaque seconde, à partir d'une tension de 3 volts. Avec des formules plus compliquées, on peut provoquer des variations de tension en créneaux, ou en dents de scie...

Si la formule est en fonction de P, et/ou de A, et/ou de S, on pourra obtenir une régulation. Par exemple si A est une mesure de température et si la sortie analogique commande l'intensité qui passe dans une résistance chauffante, mettre $(20 - T) \cdot 10$ fera augmenter le chauffage d'autant plus que la température est inférieure à 20 degrés, ce qui aboutira à une régulation de la température.

La syntaxe des formules de transformation est la même que celle pour les voies d'entrée (voir paragraphe 6.4).

4.3 Acquisition de données

C'est l'action fondamentale de Mesugraf pour Windows : toutes les autres opérations ne sont faites que dans ce but.

4.3.a Mesures = f(temps)

C'est le mode normal de MGW32, qui enregistre simplement le temps et la valeur lue sur les interfaces de mesures (sur une, deux ou trois voies).

Comme il a été expliqué précédemment, les mesures lentes sont faites en traçant chaque point dès qu'il a été mesuré, et en redonnant la main à Windows entre chaque mesure. Au contraire, les mesures rapides de type oscilloscope sont faites rapidement, et ne sont tracées sur le graphique que lorsque toutes les mesures de la série ont été faites.

Les données transmises par le pilote de mesure sont transformées selon les indications de la fonction données dans le choix de la voie principale, dans le choix de la voie auxiliaire, et dans le choix de la voie supplémentaire.

Attention au mode oscilloscope (obtenu automatiquement lorsque l'intervalle entre mesures est inférieur à 50 ms : voir réglages « temps ») !

- Il ne faut pas choisir trop de mesures par série, sinon l'ordinateur n'aura pas le temps de faire tous ses calculs et de les afficher entre deux séries de mesures. De même, si le produit (nombre de mesures) * (intervalle entre mesures) est trop long, vous vous lasserez d'attendre l'affichage des séries de points.

- pour arrêter le mode oscilloscope : appuyez sur Ctrl-S

(les autres options de ce menu ne sont pas encore fonctionnelles)

4.3.b Prestissimo

Cette option fait les mesures le plus rapidement possible, avec une mesure du temps au début et une mesure du temps à la fin ; les autres valeurs du temps sont estimées par interpolation entre ces deux mesures.

4.4 Options

Ce menu correspond à divers réglages qui ne sont pas nécessaires pour faire de bonnes séries de mesures scientifiques, mais qui peuvent améliorer la facilité d'utilisation de MGW32.

4.4.a Fichier-mémo

Cette option est utile pour les mesures lentes, surtout celles qui durent plusieurs jours, et où l'on serait très ennuyé si une coupure de courant provoquait la perte des données mesurées.

Lorsque cette ligne est cochée, le déclenchement d'une série de mesures lentes provoquera l'ouverture d'un fichier spécial, nommé MEMO.CSV, où les différentes mesures seront enregistrées l'une après l'autre.

L'intérêt essentiel de ce fichier est de ne pas perdre les données en cas d'accident (coupure de courant, par exemple, qui arrive plus souvent qu'on ne pense).

Ce type de fichier .CSV est du type "fichier ASCII", très facile à lire par un logiciel de traitement de texte et les tableurs. La première colonne correspond au temps, comme d'habitude, mais enregistrée au format « date » des tableurs. Il suffira ensuite de charger ce fichier dans votre tableur (OpenOffice, Excel...) : pour avoir tous les points de mesures ; formatez la colonne de gauche au format « date-heure » pour avoir la date et l'heure des différentes mesures.

N'oubliez pas que ce fichier MEMO.CSV sera remplacé par un nouveau dès que vous redéclencherez une série de mesures. Par conséquent, si vous désirez le garder, il faudra le renommer avant de refaire une série de mesures.

4.4.b Sauver les réglages

Cette options sauve les réglages dans un fichier, de façon à pouvoir les recharger automatiquement. Par exemple, si vous avez trouvé les bons réglages pour une expérience de cinétique enzymatique, vous pouvez sauvegarder ces réglages dans le fichier "enzyme.cfg".

Normalement, l'option "lire des réglages" devrait permettre de recharger automatiquement ces réglages, mais elle n'est pas (encore) fonctionnelle.

4.4.c gros chiffres

Cette option provoque l'apparition d'une fenêtre avec les valeurs numériques actuelles en gros chiffres, visibles du fond de la salle. C'est pratique pour des activités d'enseignement, pour montrer la valeur mesurée à l'ensemble des personnes qui assistent à la leçon.



4.5 Aide

"A propos" donne des indications sur la version de MGW32 que vous êtes en train d'utiliser.

"Mode d'emploi F1 " devrait permettre d'afficher de l'aide, mais ce n'est pas (encore ?) fonctionnel.

5 Exploitation de la série de mesures, et menu spécifique des fenêtres de mesures

5.1 Fichier

5.1.a Lire Fichier

Cette option permet de lire un fichier texte de type csv ou tab, et d'en ajouter les valeurs à celle déjà en mémoire dans la fenêtre concernée.

5.1.b Imprimer graphe

Ceci permet l'impression du graphe sur l'imprimante par défaut de Windows.

5.1.c Sauvegarder le graphique au format .bmp ou .wmf.

Ces fichiers seront ensuite utilisables par des logiciels de dessin et incorporables dans des documents de traitements de texte.

5.1.d Sauvegarder sous différentes formes

Bien sûr, ces options provoquent l'ouverture d'une boîte de dialogue où vous pourrez choisir le nom du fichier où vous sauvegarderez les valeurs numériques de la série de mesures.

Par défaut, les fichiers sont sauvegardés dans le sous-répertoire FICHIERS.

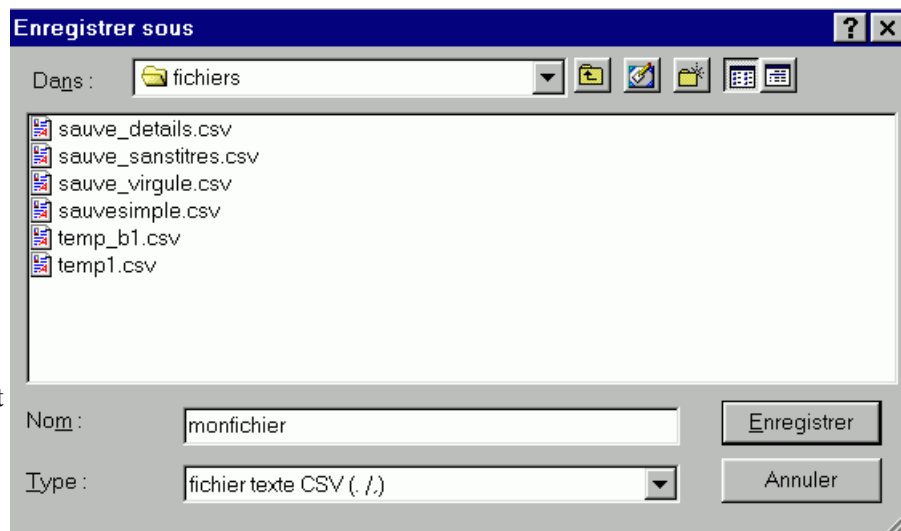
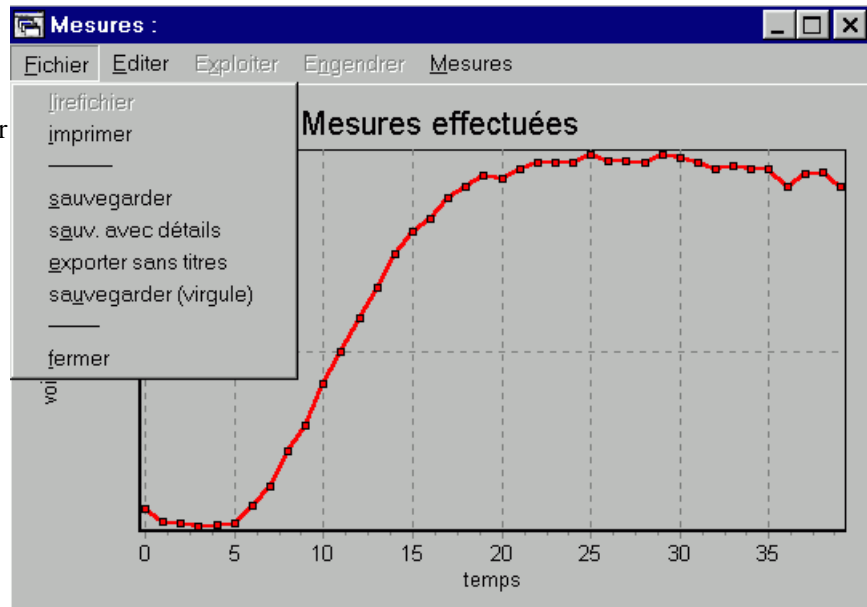
Sauvegarder

Ils peuvent être de trois types :

- CSV : « Champs Séparés par des Virgules » = « Comma Separated Values » : chaque mesure donne une ligne, et les différents champs (temps, voie principale, et éventuellement voies auxiliaire et supplémentaire) sont séparés par des virgules. Le signe décimal est le point.
- TAB : C'est le même format, mais les champs sont séparés par des tabulations

- DIF : Format DIF « Data Interchange Format », un très vieux format en mode texte.

La première ligne comporte le titre des différentes colonnes (temps, voie principale, voie auxiliaire, voie supplémentaire).



Sauvegarder avec détails

Ce sont les mêmes formats, mais à la fin du fichier sont ajoutés divers détails, comme le titre de la fenêtre...

Ces fichiers risquent de poser des problèmes avec certains logiciels de modélisation, qui ne veulent pas de tels détails.

Exporter sans titres

Ce sont les mêmes formats, mais il n'y a pas de première ligne avec les titres des colonnes : le fichier commence directement par les valeurs numériques.

Sauvegarder (virgules)

Ce sont les mêmes formats que l'option Sauvegarder, mais avec une ponctuation différente : le signe décimal est une virgule, et les champs sont séparés par des points-virgules et non par des virgules.

Ce dernier format est davantage compatible avec Excel version française.

Fermer

Cette option provoque la disparition de la fenêtre, et la perte des informations qu'elle contenait. Pensez à sauvegarder les mesures avant de fermer la fenêtre !

5.2 Éditer

5.2.a Copier les données vers le presse-papier

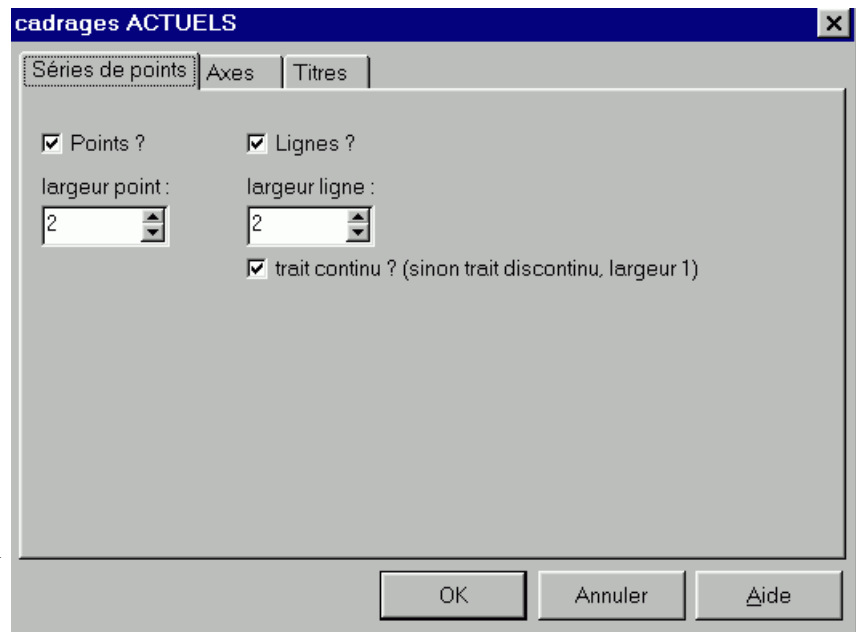
Il suffira ensuite de les coller dans le tableur (OpenOffice, Excel, etc). Deux options sont possibles : soit le signe décimal est le point, soit le signe décimal est la virgule.

5.2.b Enjoliver-Cadrages

Cette option ouvre une boîte de dialogue qui permet de modifier l'affichage de la fenêtre graphique, en particulier en vue d'une impression.

On peut régler la largeur des points et des lignes les joignant. Par contre, on ne peut pas changer les couleurs : la voie principale doit être en rouge, la voie auxiliaire en vert, et la voie supplémentaire en bleu, pour qu'avec l'habitude, l'utilisateur sache immédiatement quelles voies sont figurées par les diverses courbes.

On peut aussi régler les axes verticaux (automatiques ou non-automatiques), régler les titres de la fenêtre, des abscisses et ordonnées...



5.3 Exploiter

5.3.a Statistiques

Cette option donne les paramètres statistiques : moyenne, nombre de points, écart-type.

5.3.b Transformer $P = f(\text{PAST})$

On peut modifier les valeurs de la série de mesures sur la voie principale par une formule en fonction de 4 variables, P (la voie principale), A (la voie auxiliaire), S (la voie supplémentaire) et T (le temps).

5.3.c Lissage P

Cette option permet de lisser sur quelques points les valeurs de la voie principale.

Les options Dériver P/X et intégrer P/X (options non fonctionnelles) : voir menu "Engendrer".

5.4 Engendrer

Ce menu permet d'engendrer de nouvelles fenêtres, avec des valeurs numériques provenant de la fenêtre actuelle, avec plus ou moins de modifications.

5.4.a Tel quel

Comme son nom l'indique, cette option ne fait que dupliquer la fenêtre actuelle.

5.4.b Dériver P/X et Intégrer P/X

La première option permet de calculer la courbe dérivée, et la seconde option provoque au contraire l'intégration, mais seulement pour la voie principale.

5.4.c $P=f(X)$, $P=f(A)$, $P=f(S)$, $A=f(X)$, $A=f(P)$, $A=f(X)$

Ces options permettent de séparer les voies et de les figurer l'une en fonction de l'autre.

Par exemple $P=f(X)$ trace une nouvelle fenêtre, semblable à la première, mais avec uniquement la voie principale. De même $A=f(X)$ trace une fenêtre avec en voie principale les valeurs de la voie auxiliaire de la première fenêtre.

$P=f(A)$ trace une nouvelle fenêtre avec la voie principale en fonction de la voie auxiliaire ; $P=f(S)$ fait la même chose, mais en fonction de la voie supplémentaire ; $A=f(P)$ trace la voie auxiliaire en fonction de la voie principale, et $A=f(S)$ trace la voie auxiliaire en fonction de la voie supplémentaire.

5.5 Mesures

Il n'y a qu'une seule option fonctionnelle (pour l'instant), celle de stopper prématurément une série de mesures. Cette option devient inutile ensuite.

6 Annexes

6.1 Que faire si MGW32 ne fonctionne pas aussi bien que vous le souhaitez ?

MGW32 se lance correctement, mais lorsque je veux charger un pilote (dll) d'une interface, rien ne se passe, ou bien il apparaît une succession de fenêtres avec un message d'erreur... C'est que votre ordinateur est trop puissant.

Trop puissant, ce n'est pas tout à fait vrai. C'est peut-être simplement que votre système d'exploitation est Windows XP (ou un autre Windows NT), qui empêche l'accès des programmes à l'électronique de l'appareil, et en particulier empêche la lecture et l'écriture sur les ports. Donc il est impossible d'utiliser les cartes à enficher dans un port d'extension (Candibus, Jeulin ESAO, PMB, Leybold-Cassy) mais aussi certains appareils à brancher sur un autre connecteur (Capman sur la prise manettes de jeux, Cappar ou ADES sur la prise parallèle...). Par contre, les appareils à brancher sur une prise série (RS232) doivent fonctionner correctement (Orphy, multimètres, pHmètres, etc).

Les solutions sont connues : il faut obliger WinXP à obéir.

- Des pilotes spécifiques pour ces systèmes WinNT ont été conçus. Il faut d'abord mettre le fichiers giveio.sys dans le répertoire de Windows (en général [C:\Windows](#)), puis ensuite utiliser ces pilotes, dont le nom commence par xp (par exemple xpcapman.dll).
- En consultant Internet, on peut trouver d'autres solutions, mais qui utilisent toutes des fichiers spéciaux.
- ou bien changer de système d'exploitation...

Il peut exister aussi des problèmes de pilotes, qui ont été mal programmés. Un certain nombre de pilotes ont été programmés « en aveugle », c'est à dire uniquement avec la documentation, mais sans réellement avoir été essayés pour faire de vraies mesures avec le vrai appareillage, et il est possible que la documentation ait mal été comprise.

Deux solutions possibles :

- contacter l'auteur de MGW pour vous plaindre, et espérer qu'il aura le temps et les compétences pour faire un meilleur pilote.
- programmer vous-même un pilote convenable, grâce aux indications données dans l'annexe 6.3, puis contacter l'auteur pour lui envoyer ce bon pilote, afin qu'il le mette dans la liste des pilotes disponibles. Cette solution est bien meilleure que la précédente !

J'ai un super appareil de mesure, mais qui n'est pas dans la liste des pilotes d'appareils de mesure. Comment faire pour l'utiliser avec les logiciels de la famille MGW32 ?

Les solutions sont les mêmes que pour le problème précédent : soit demander poliment à l'auteur de MGW32 d'en faire un, soit programmer vous-même un pilote (ce doit être assez facile si vous avez la documentation complète, et que vous savez programmer en Pascal (Delphi, FreePascal, Lazarus...), ou en Basic (PureBasic, FreeBasic...), ou en diverses versions de C++, et peut-être dans d'autres langages.

6.2 Matériel de mesure fonctionnant avec MGW32

En principe, MGW32 peut fonctionner avec tous les appareils de mesure connectable à l'ordinateur, à condition qu'il existe une DLL de mesure pour cet appareil. Certains de ces appareils sont anciens, et ne sont plus vendus dans le commerce, mais si vous les avez, il serait dommage de les mettre à la poubelle sans les utiliser. MGW32 peut leur donner une deuxième vie !

Si vous pouvez consulter Internet, des pilotes récents sont disponibles sur le site de l'auteur :
<http://sciencexp.free.fr>

XADES101.DLL

L'interface ADES est conçue par ITC et était distribuée par la Maison des Enseignants de Provence. C'est un boîtier à connecter sur la prise parallèle (prise imprimante), et qui a 2 entrées analogiques (et éventuellement 4 relais). Il est à brancher sur la prise parallèle pour l'imprimante (LPT1:). Dans la pratique, certains ordinateurs utilisent pour leur prise d'imprimante normale les adresses correspondant à LPT2: ou LPT3:, par exemple les ordinateurs IBM PC300 ; il faut donc utiliser xades102.dll, ou xades103.dll. Ces fichiers sont aussi utilisables si vous avez plusieurs prises parallèles sur votre ordinateur. Ce n'est pas très fréquent, mais ça peut exister...

XCANDI.DLL

La carte Candibus de Langage et Informatique, et Candiplus qui lui succède depuis quelques années.

XCAPMAN.DLL

Interface "Capman", c'est à dire "CAPteurs à la place des MANnettes de jeu" : c'est la prise "joystick", qui permet de très bonnes mesures de résistance, donc de lumière par les photorésistances et phototransistors, de température par les thermistances, de longueur et d'angle par les potentiomètres, etc.

Tel quel, ce pilote prend beaucoup de temps pour mesurer les très grosses résistances, en particulier les résistances infinies lorsque rien n'est connecté à cette prise. C'est pourquoi xcapmanrapide_2voies.dll est conçu pour ne pas bloquer le logiciel : il ne fait les mesures que sur les deux premières voies, et arrête le comptage plus rapidement.

XCAPPAR1.DLL, XCAPPAR2.DLL et XCAPPAR3.DLL

Interface CAPPAR, c'est à dire CAPteurs sur la prise PARallèle (prise imprimante). Ces fichiers correspondent aux branchements respectivement en LPT1: (prise imprimante normale), LPT2: et LPT3 (prises imprimantes supplémentaires). Dans la pratique, certains ordinateurs utilisent pour leur prise d'imprimante normale les adresses correspondant à LPT2: ou LPT3:, par exemple les ordinateurs IBM PC300...

XCARTESON.DLL

Pour mesurer à partir de l'entrée micro d'une carte-son.

XELECTROME1.DLL, et XELECTROME2.DLL

L'interface Electrome PC101 (et quelques autres interfaces voisines) est faite par Electrome, de Bordeaux. Les DLL précédente sont respectivement pour un branchement sur leport série n°1 (Com1:), Com2::

XESAO3.DLL et XESAO3REC.DLL

L'interface ESAO3 de Jeulin (et ESAO4, qui lui est compatible) est à brancher dans un slot de l'ordinateur ; XESAO3REC.DLL doit permettre une reconnaissance automatique des capteurs (à vérifier). Vers 1990, Jeulin publiait les adresses de la mémoire utilisées par ESAO3, mais cette bonne pratique n'a pas continué pour ESAO4, c'est pourquoi un bon nombre de fonctionnalités d'ESAO4 ne sont pas disponibles.

XHLAB21.DLL et XHLAB22.DLL

Le pHmètre Handylab2, de Schott, est à brancher soit en Com1: soit en Com2:.

XLX105_1.DLL et XLX105_2.DLL

Ce sont les luxmètres diffusés par la Maison des Enseignants de Provence, à brancher respectivement en Com1: et Com2:

XMETEX1.DLL et XMETEX2.DLL

Ce sont les multimètres METEX, à brancher respectivement en Com1: et Com2:

XMMTES1.DLL et XMMTES2.DLL

Ce sont les multimètres TES2730, diffusés par la Maison des Enseignants de Provence, à brancher respectivement en Com1: et Com2:

XORPHY1.DLL et XORPHY2.DLL

Ces fichiers sont pour l'excellente interface Orphy GTS de Micrelec, à brancher respectivement en Com1: et Com2:

Normalement, ils doivent permettre d'utiliser aussi l'interface Orphy GTI et Orphy Portable.

Pour l'interface Orphy portable 1, qui fonctionne avec des capteurs spécifiques, et qui reconnaît automatiquement les capteurs ("capsules"), il existe aussi les pilotes **XORPHYPORTABLE1.DLL** et

XORPHYPORTABLE2.DLL

Pour Orphyportable2, qui a un petit écran de visualisation, est plus précise qu'OrphyPortable1, et n'a pas les mêmes commandes, il existe **x_orphyportable2_com1.dll** et **x_orphyportable2_com2.dll**

XPCMES.DLL

Interface Eurosmart PCMES2, à mettre dans un slot de l'ordinateur.

XPHM210_1.DLL et XPHM210_2.DLL

pour pHmètre Radiometer PHM210 en com1: et Com2:

XPHMEP1.DLL, XPHMEP2.DLL, XPHMEP3.DLL, XPHMEP4.DLL

Le pHmètre de la Maison des Enseignants de Provence, à brancher sur une prise RS232, respectivement Com1:, Com2:, Com3:, Com4:. Attention ! bien qu'à brancher sur une prise RS232, ces pHmètres n'utilisent pas la communication série, et ils ne fonctionnent pas sous Windows XP.

QPMB.DLL

interface PMB de la Maison des Enseignants de Provence (8 entrées et 8 sorties logiques, 4 entrées analogiques 0 à 2,46 V, 1 sortie analogique 0 à 9 V, conversion sur 8 bits en 5µs)

XPMBSUP.DLL

Interface PMB> (lire PMB supérieur), de la Maison des Enseignants de Provence (sur deux cartes différentes : 8 entrées et 8 sorties logiques, 2 entrées analogiques 0 à 2,5 V, 2 entrées analogiques -5 V à + 5 V, 2 sorties analogiques 0 à 10 V et -5 V à + 5 V, timer pour générer des fréquences ou les impulsions, conversion sur 8 bits en 2 µs)

XSARTOR1.DLL et XSARTOR2.DLL

Ces fichiers sont pour les balances Sartorius PT600, à brancher respectivement en Com1: et Com2:

XTM906A1 DLL et XTM906A2.DLL

Thermomètres TM906 (diffusés par la Maison des Enseignants de Provence), à brancher respectivement en Com1: et Com2:.

6.3 Programmation des pilotes (DLL de mesure) pour divers appareils

Syntaxe proposée pour les DLL :

Fonctions de base

La numérotation des voies commencera à zéro, au moins pour ce qui est de la programmation. Ceci n'empêche pas éventuellement que la première voie (numérotée 0 pour la programmation) puisse porter un nom tel que "Voltmètre 1".

La syntaxe proposée ici est voisine de celle commandant la centrale de mesure Orphy-GTS de Microlec ; si l'opération qu'elles doivent faire est incorrecte (voie inexistante, par exemple) elles renvoient une valeur fixe (par exemple -777 pour ce qui est des fonctions numériques). Pour qu'on sache bien qu'il faut utiliser la convention d'appel stdcall, le nom des fonctions commence par std.

Bien sûr, il est possible de rajouter des fonctions d'autres types, par exemple, pascarea, pascalnea, pascarea, pascalnea, etc.

Pour les réels (de type "double"), la fonction sera précédée de P (comme "Pointeur") lorsqu'elle renverra l'adresse du résultat et non celui-ci directement.

Lors du chargement de la DLL, chaque fonction est identifiée par son nom, qui est une chaîne de caractères. Il y a une différence entre les majuscules et les minuscules ; il vaut mieux convenir que tous les noms de fonctions seront en minuscule.

Chaque variante de fonction de mesure sera associée à un nom particulier. Un premier intérêt en est d'indiquer clairement à quoi correspond la variante. Par exemple, pour les interfaces ayant plusieurs calibres sélectionnables logiciellement (Leybold-Cassy...), chaque voie de mesure correspondra à un calibre, qui sera indiqué dans ce nom, ou bien pour les interfaces ayant plusieurs voies, spécialisées chacune dans un type de mesure (Orphy-GTS, qui a des calibres différents selon les connecteurs, ou bien Pierron SMF10, qui a des entrées spécialisées pour le pH ou le magnétisme...), ce nom indiquera ces spécialités.

Un deuxième avantage, plus universel, est que ce nom permettra au programme appelant de savoir quelles sont les voies réellement appelables. Toutes les voies ayant un nom (non nul) seront appelables, et les voies qui n'existent pas réellement auront un nom de longueur nulle.

Le plus simple est de faire commencer le nom de la fonction qui renvoie le nom de la voie par la lettre N, comme "Nom". Le format de ces nom sera le type "chaîne à zéro terminal" ou "AsciiZ", déclaré dans le langage Pascal comme Pchar ; ce format est le format de chaînes standard dans les langages C et C++.

```
function stdea(n:integer) : integer ;stdcall;export;
nième entrée analogique (résultat directement renvoyé par le convertisseur analogique-numérique)
```

```
function stdnea(n:integer):pchar ;stdcall;export;
nom de la nième entrée analogique
```

```
function stdead(n:integer):double ;stdcall;export;
nième entrée analogique (résultat converti en unité SI, le plus souvent en Volts)
```

```
function stdnead(n:integer):pchar ;stdcall;export;
nom de la nième entrée analogique renvoyant le résultat sous forme de "double"
```

```
function stdsa(n:integer;valeur:integer):integer;stdcall;export;
envoi de valeur au convertisseur numérique analogique sur la nième sortie analogique ; si tout s'est bien passé, elle renvoie valeur.
```

```
function stdnsa(n:integer):pchar;stdcall;export;
nom de la nième sortie analogique fixant directement la valeur du CNA.
```

```
function stdsad(n:integer; valeur:double):double;stdcall;export;
Elle permet la fixation de la nième sortie analogique à valeur. Pour la plupart des interface, la valeur sera en volts, mais on peut imaginer des systèmes plus complexes, où une sortie puisse commander une température, une vitesse de rotation, une intensité lumineuse, ou d'autres grandeurs pouvant varier.
```

```
function stdnsad(n:integer):pchar;stdcall;export;
```

C'est le nom de la fonction précédente

```
function stdeb(n:integer):integer ;stdcall;export;
```

nième entrée binaire (ou entrée logique). Le résultat "vrai" correspondra à 1, et le résultat "faux" correspondra à zéro.

```
function stdneb(n:integer):pchar;stdcall;export;
```

C'est le nom de la nième entrée binaire

```
function stdsb(n:integer;valeur:integer):integer;stdcall;export;
```

fixation de la nième sortie binaire à "vrai" si valeur vaut 1, et à "faux" si valeur vaut zéro ; si tout s'est bien passé, elle renvoie valeur

```
function stdnsb(n:integer):pchar;stdcall;export;
```

nom de la nième sortie binaire.

```
function stdtitre : pchar;stdcall;export;
```

renvoie le titre de la DLL, que l'on pourra utiliser dans des boîtes de dialogue.

```
function stddetail : pchar ;stdcall;export;
```

renvoie le nom détaillé de la DLL, avec par exemple le nom de l'auteur, la date de révision, etc.

Fonctions permettant d'utiliser plusieurs logiciels d'application simultanément, avec la même interface et la même DLL

Ces fonctions de base sont déjà définies depuis plusieurs années. Je me suis aperçu au fil du temps que d'autres fonctions étaient utiles, en particulier pour Mesugraf pour Windows, qui commande les sorties logiques et analogiques, et qui doit donc mémoriser leur état dans la DLL elle-même, de façon à ce qu'une même DLL puisse être appelée par plusieurs logiciels différents.

J'ai donc introduit les fonctions suivantes, qui commencent par R, comme "réponse" :

```
function stdrsa(n:integer):integer;stdcall;export;
```

renvoie l'état de la nième sortie analogique

```
function stdrsad(n:integer):double;stdcall;export;
```

renvoie l'état de la nième sortie analogique, convertie en volts ou autre unité appropriée

```
function stdrsb(n:integer):integer;stdcall;export;
```

renvoie l'état de la nième sortie binaire, 0 pour hors-tension, 1 pour sous-tension.

Fonction permettant d'ouvrir une fenêtre de réglage

```
function stdcalibration(pchar) : pchar ;stdcall;export;
```

Cette fonction facultative provoque l'ouverture d'une boîte de dialogue qui permet d'afficher et de faire les réglages.

Bien sûr, le résultat est que cette DLL est plus grosse que si elle ne contenait pas de boîte de dialogue. A titre d'exemple, une DLL simple fait environ 30 ko, alors qu'une DLL avec une boîte de dialogue fait 250 ko.

Par exemple pour Mesugraf pour Windows 16 bits, il a été fait une DLL pour ADES et une sonde à oxygène. Pour que la valeur renvoyée soit effectivement une concentration en oxygène, il faut calibrer cette sonde, en indiquant les valeurs correspondant au zéro en oxygène, et à la teneur atmosphérique en oxygène. Cette fenêtre de calibration s'obtient par l'appel à la fonction «calibration».

Exemples de bibliothèques dynamiques programmées en Pascal (Delphi 5) : DLL pour ADES connecté à la prise imprimante (1e prise parallèle)

```
library xadestar;
```

```
uses
    adesunit,sysutils,wintypes;
```

```

(* l'unité adesunit contient les routine spéciales pour ADES ;
   pour d'autres appareils, utiliser leurs routines spécifiques*)
type array100= array[0..100] of char;
type pdouble=^double;
var memosb:array[0..3]of boolean;
    memosad:double;

function stdea(n:integer):integer;stdcall;export;
var o,n1,n2:word;
begin
  o:=n;
  stdea:=-777;
  if (n=0)or(n=1)then lit_niveau(n1,n2);
  case o of 0: stdea:=n1;
            1: stdea:=n2;
            2: ades_off;
            3: ades_on;
            end;
end;

function stdnea(n:integer):pchar;stdcall;export;
begin
  if (n=0)
    then stdnea:='E. analogique 0'+chr(0)
    else if n=1 then stdnea:='E. analogique 1'+chr(0)
    else if n=2 then stdnea:='RAZ pour impression'+chr(0)
    else if n=3 then stdnea:='Réinitialisation'+chr(0)
    else stdnea:=chr(0);
end;

function stdead(n:word):double;stdcall;export;
var aux:integer;
begin
  if (n=0) or (n=1)or (n=2) or (n=3)
    then begin aux:=stdea(n); stdead:=(5.0*aux)/1024; end
    else stdead:=-777;
end;

function stdnead(n:integer):pchar;stdcall;export;
begin
  if (n=0)
    then stdnead:='E. analogique 0'+chr(0)
    else if n=1 then stdnead:='E. analogique 1'+chr(0)
    else if n=2 then stdnead:='RAZ pour impression'+chr(0)
    else if n=3 then stdnead:='Réinitialisation'+chr(0)
    else stdnead:=chr(0);
end;

function stdsa(n:integer; valeur:integer):integer;stdcall;export;
begin
  stdsa:=-777; (*Ades n'a pas de sortie analogique*)
end;

function stdnsa(n:word):pchar;stdcall;export;
begin
  if n=0 then stdnsa:='SA fictive'+chr(0) else stdnsa:='' +chr(0);
end;

function stdsad(n:integer ; valeur:double):double;stdcall;export;
begin
  MEMOSad:=valeur; stdsad:=0;
end;

function stdnsad(n:integer):pchar;stdcall;export;
begin
  if n=0 then stdnsad:='SA fictive'+chr(0)else stdnsad:='' +chr(0);

```

```

end;

function stdrsa(n:integer):integer;stdcall;export;
begin (*état de la sortie analogique fictive*)
if n=0 then stdrsa:=2 else stdrsa:=-777;
end;

function stdrsad(n:integer):double;stdcall;export;
begin
if n=0 then stdrsad:=2 else stdrsad:=-777;
end;

function stdeb(n:integer):integer;stdcall;export;
begin (*entrées logiques fictives, inexistantes sur ADES*)
if n=0 then stdeb:=0 else if n=1 then stdeb:=1 else stdeb:=-777;
end;

function stdneb(n:integer):pchar;stdcall;export;
begin
if n=0 then stdneb:='EB fictive 0'+chr(0)
else if n=1 then stdneb:='EB fictive 1'+chr(0)
else stdneb:=chr(0);
end;

function stdsb(n:integer; etat:integer):integer;stdcall;export;
begin
case n of
0..3 : begin relais[n+1]:=(etat=1);
stdsb:=n*100+etat;
pilote_relais(0);
memosb[n]:=(etat=1);
end;
else stdsb:=-777;
end;
end;

function stdnsb(n:integer):pchar;stdcall;export;
begin
result:=chr(0);
if n=0 then result:='relais A'+chr(0);
if n=1 then result:='relais B'+chr(0);
if n=2 then result:='relais C'+chr(0);
if n=3 then result:='relais D'+chr(0);
end;

function stdrsb(n:integer):integer;stdcall;export;
begin
if n in [0..3] then if memosb[n] then stdrsb:=1 else stdrsb:=0
else stdrsb:=-777;
end;

function stdtitre : pchar;stdcall;export;
begin stdtitre:='ADES en LPT1:'+chr(0);end;
function detail : pchar;
begin stddetail:='essai de DLL (32 bits) pour ADES 10 bits en LPT1., par P.
Dieumegard, le 10 août 2002'+chr(0);end;

(*les deux fonctions qui suivent sont pour EXCEL ??*)
function pead(n:integer):pdouble;export;
var dloc:double;
begin dloc:=eaD(n);pead:=@dloc;end;

```

```

function PSAD(n:integer;valeur:pdouble):pdouble;export;
var dloc1,dloc2:double;
begin dloc1:=valeur^;dloc2:=SAD(n,dloc1);psad:=@dloc2;end;

(*les fonctions " double " qui suivent sont pour StarOffice ;
en fait, il est possible que les fonctions stdxxx soient aussi possibles*)
function eadouble(x:double):double;
var nloc:integer;
begin nloc:=round(x); eadouble:=ead(nloc);end;

function neadouble(x:double):pchar;
begin
  neadouble:=nead(round(x));
end;

function sadouble(x:double;xval:double):double;
begin sadouble:=sad(round(x),xval);end;

function nsadouble(x:double):pchar;
begin nsadouble:=nsad(round(x));end;

function rsadouble(x:double):double;
begin rsadouble:=rsad(round(x));end;

function ebdouble(x:double):double;
begin ebdouble:=eb(round(x));end;

function nebdouble(x:double):pchar;
begin nebdouble:=neb(round(x));end;

function sbdouble(n:double;etat:double):double;
begin sbdouble:=sb(round(n),round(etat));end;

function nsbdouble(n:double):pchar;
begin nsbdouble:=nsb(round(n));end;

function rsbdouble(n:double):double;
begin rsbdouble:=rsb(round(n));end;

function ea(n:integer):integer; begin ea:=stdea(n);end;
function nea(n:integer):pchar; begin nea:=stdnea(n);end;
function ead(n:integer):double; begin ead:=stdead(n);end;
function nead(n:integer):pchar; begin nead:=stdnead(n);end;
function sa(n:integer ; valeur:integer):integer ; begin sa:=stdsa(n,valeur);end;
function nsa(n:integer):pchar; begin nsa:=stdnsa(n);end;
function rsa(n:integer):integer; begin rsa:=stdrsa(n);end;
function sad(n:integer; valeur:double):double; begin sad:=stdsad(n,valeur);end;
function rsad(n:integer):double; begin rsad:=stdrsad(n);end;
function eb(n:integer):integer; begin eb:=stdeb(n);end;
function neb(n:integer):pchar; begin neb:=stdneb(n);end;
function sb(n:integer;etat:integer):integer; begin sb:=stdsb(n,etat);end;
function nsb(n:integer):pchar; begin nsb:=stdnsb(n);end;
function rsb(n:integer):integer; begin rsb:=stdrsb(n);end;
function titre:pchar; begin titre:=stdtitre;end;
function detail:pchar; begin detail:=stddetail;end;

exports
  pead ,
  psad ,

  eadouble ,
  neadouble ,
  sadouble,nsadouble,rsadouble,
  ebdouble,nebdouble,
  sbdouble, nsbdouble,rsbdouble,
  stdea, stdnea,stdsa,stdnsad,stdtitre,stddetail,

```

```
stdead, stdnead, stdsad, stdnsad, stdeb, stdneb, stdsb, stdnsb, stdrsad, stdrsb,  
ea, nea, ead, nead, sa, nsa, rsa, sad, nsad, rsad,  
eb, neb, sb, nsb, rsb, detail, titre;
```

```
var i:word;  
begin  
  nbitades:=10;  
  initialise_ades(888);  
  ades_on;  
  for i:=0 to 3 do memosb[i]:=false;  
end.
```


6.4 Syntaxe des fonctions de transformation

Les fonctions de transformation, que ce soit pour les entrées analogiques, les sorties analogiques ou la transformation après la fin de l'acquisition, sont basées sur la bibliothèque Modulo.

Vous pouvez employer :

- Opérateurs habituels : +, -, *, / ;

l'exponentiation est possible par ^ (par exemple 2^p)

- Fonctions mathématiques unaires :

ABS = valeur absolue

SIN, COS, TAN = sinus, cosinus, tangente

ASIN, ACOS, ATAN = arc sinus, arc cosinus, arc tangente

SH, CH, TH = sinus, cosinus, tangente hyperbolique

ASH, ACH, ATH = fonctions inverses des précédentes

LN, LOG = logarithmes népérien et décimal

EXP = exponentielle

INT, FRAC = partie entière et partie fractionnaire

ROUND = arrondi

- Fonctions mathématiques binaires :

Min(X,Y) renvoie le minimum de X et de Y

Max(X,Y) renvoie le maximum de X et de Y

- les constantes E et PI sont acceptées.

6.5 La famille Mesugraf, son histoire et l'origine de divers composants du programme

Dans le monde des "Compatibles PC", depuis 1992 existe Mesugraf, avec différentes versions du domaine public. Il a été le premier logiciel d'EXAO à pouvoir fonctionner avec de très nombreuses interfaces, grâce au système des "pilotes d'interfaces", faciles à programmer et faciles à utiliser. Mesugraf pour Dos pouvait fonctionner même sur de très petites configurations, qui paraissent risibles aujourd'hui : écran CGA, ordinateur sans disque dur ni mémoire étendue...

Un programme-frère de Mesugraf pour DOS est Oscillo, qui fonctionne en tant qu'oscilloscope, .c'est à dire que toutes les mesures d'une série sont réalisées avant d'être affichées.

En 1996 est apparu TG-Mesugraf, toujours sous DOS, mais qui est beaucoup plus puissant. TG signifie "Tout Graphique", mais aussi TEGl, car il a été programmé grâce à l'interface graphique TEGl, de TEGl Systems Corporation, de Vancouver au Canada.

En 2000, est apparu Mesugraf pour Windows, fonctionnant avec les DLL de mesure, avec des graphiques dans les fenêtres par XYGRAPH, de Grahame Grieve (Australie). C'est le logiciel ayant le plus de possibilités.

MGW32 est développé peu à peu, mais il n'a pas encore toutes les possibilités de Mesugraf pour Windows.

Toujours dans le souci de compatibilité maximale, il est souhaitable de ne pas se limiter au système d'exploitation Microsoft DOS-Windows, et j'essaie de programmer l'équivalent pour Linux. Malheureusement le langage Kylix, qui devrait être équivalent pour Linux de Delphi pour Windows est moins pratique à utiliser (les programmes exécutables nécessitent des fichiers supplémentaires de type .so, qui ne sont pas toujours disponibles).

Depuis 2011, une nouvelle famille de logiciels est apparue, la famille Mensurasoft. Les membres de cette famille sont des logiciels libres (au sens GPL). Mensurasoft-PB a été développé en PureBasic, et Mensurasoft-LZ a été développé en Lazarus. Ils utilisent les mêmes types de pilotes que MGW32, mais de type "cdecl" au lieu de "stdcall".

=====

La programmation de ces logiciels a été faite en langage Pascal. Les versions sous Dos ont été programmées en TurboPascal 6 puis en Borland Pascal 7, aussi bien pour les logiciels principaux que pour les pilotes d'interfaces. MGW a été programmé en Borland Delphi 1, et les DLL de mesure ont été programmées soit en Borland Pascal 7, soit en Delphi 1.

MGW32 a été programmé en Delphi5

=====

La souplesse d'emploi des fonctions de transformation vient de la puissance de la bibliothèque MODULOG, de l'Alesup (Luminy).

La mesure du temps à l'échelle de la microseconde a été possible en fouillant dans les archives SWAG qui sont un ensemble extraordinaire de sources de programmation pour TurboPascal pour DOS.

Table des matières

1	Place de MGW32 dans la chaîne de mesure.....	1
1.1	Principe d'une chaîne de mesure informatisée.....	1
1.2	MGW32 fait des mesures en fonction du temps.....	2
1.3	Mesugraf pour Windows peut transmettre les valeurs à d'autres logiciels.....	2
2	Installation et lancement de MGW32.....	3
2.1	Matériel nécessaire.....	3
2.2	Copie des fichiers dans le répertoire de votre choix.....	3
3	Tutoriel : paramétrage pour l'acquisition de données, et réalisation d'une série de mesures.....	4
3.1	Réglage de l'interface pour la voie principale.....	4
3.2	Réglage des paramètres temporels.....	5
3.3	Acquisition d'une série de mesures.....	5
3.4	Exploitation des données obtenues.....	6
4	Les différentes options de menu de MGW32.....	6
4.1	Menu Fichiers.....	6
4.2	Menu Réglages.....	7
4.3	Acquisition de données.....	9
4.4	Options.....	9
4.5	Aide.....	11
5	Exploitation de la série de mesures, et menu spécifique des fenêtres de mesures.....	12
5.1	Fichier.....	12
5.2	Éditer.....	13
5.3	Exploiter.....	14
5.4	Engendrer.....	14
5.5	Mesures.....	15
6	Annexes.....	16
6.1	Que faire si MGW32 ne fonctionne pas aussi bien que vous le souhaitez ?.....	16
6.2	Matériel de mesure fonctionnant avec MGW32.....	17
6.3	Programmation des pilotes (DLL de mesure) pour divers appareils.....	19
6.4	Syntaxe des fonctions de transformation.....	25
6.5	La famille Mesugraf, son histoire et l'origine de divers composants du programme.....	26