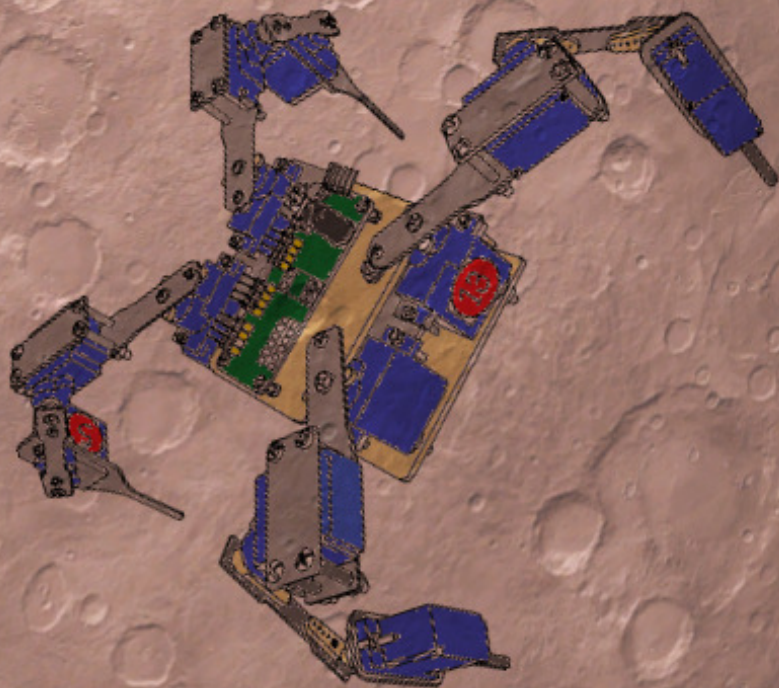


Sonde martienne JEKERT Pilotage autonome avec un PUPITRE de télécommande.



Landing Site



Par Nulentout : Samedi 29 Septembre 2018.

Développement du projet Sonde martienne JEKERT

Piloter JEKERT à partir de la ligne USB et le moniteur de l'**IDE** est parfait pour développer le logiciel au moindre frais et surtout avec un maximum de facilité. Ce n'est toutefois pas la solution la plus conviviale car à proximité de l'ordinateur ne se trouve pas forcément une table suffisamment dégagée pour faire évoluer la petite machine. Par ailleurs, quand on désire présenter l'insecte robotisé à des amis, lors d'une soirée d'hiver, c'est au salon qu'ils sont reçus. Ce n'est pas l'endroit où habituellement réside le P.C. Aussi, si l'investissement matériel est envisageable, télécommander JEKERT avec un petit pupitre autonome est très alléchant. Sur le plan informatique, cette facette à débroussailler n'est pas élémentaire du tout car il faut faire communiquer deux machines. Du coup, ce sont deux logiciels qui doivent cohabiter en parallèle : Celui sur JEKERT et celui du *Pupitre de commande*. Il importe également de définir entièrement l'aspect opérationnel, esthétique, convivial du dispositif à agencer. Autant dire que l'éventail d'investigations se montre merveilleusement vertigineux ...

PRÉSENTATION SOMMAIRE DU PUPITRE



Logiquement, si vous consultez ce document, c'est que vous avez la chance de pouvoir financer ce complément de luxe pour JEKERT, et surtout le privilège de disposer d'assez de temps de liberté pour envisager sereinement la suite. Pour financer, il est relativement facile de réunir les fonds nécessaires : On arrête définitivement de fumer, on supprime les bringues du samedi soir, ce qui du coup nous dégage plein plein de temps de loisir, on abandonne l'achat du coupé sport tout neuf prévu pour frimer le samedi soir, et on demande au patron de faire plein plein plein d'heures supplémentaire. Non, cette dernière option n'est pas viable vu que l'on n'aurait plus de temps pour concrétiser le pupitre ! Bref, après divers sacrifices, le bilan financier rend possible l'achat des divers composants. Alors trêve de bavardages inutiles, passons immédiatement au vif du sujet. Comme ce document n'est qu'une suite intimement liée à celui sur la réalisation de JEKERT, la numérotation des figures et des divers chapitres poursuit linéairement celle du didacticiel initial. Naturellement, on suppose ici que JEKERT est totalement opérationnelle et qu'elle donne pleinement satisfaction quand elle est vérifiée et validée avec le Moniteur USB de l'**IDE**.

Fiche signalétique du pupitre

➤ Encombrement.

- **Largeur** environ 170mm.
- **Profondeur** environ 140mm sans les fiches des cordons ombilicaux.
- **Profondeur** environ 210mm avec les cordons ombilicaux branchés à l'arrière.
- **Hauteur** environ 105mm.

➤ Autonomie énergétique.

- **Alimentation des deux cartes Arduino :**
Utilisation d'un petit module du commerce montré en Fig.84 qui est vendu pour servir de réservoir d'énergie USB quand on est en voyage. Cette source est suffisante pour autoriser au moins 10H d'autonomie aux deux cartes Arduino NANO.

Fig.84



Fig.85

- **Alimentation de la motorisation :**

Bien qu'il sera parfaitement possible de n'en installer qu'un seul, en version de luxe le pupitre intègre deux accumulateurs 6Vcc tel que celui de la Fig.85 pouvant débiter des courants de pointe très importants. Dans la pratique chaque élément ne fait réellement que 1300 à 1600mAh de capacité, raison pour laquelle deux sont prévus dans le coffret. (*Un seul serait suffisant.*)

➤ Gestion opérationnelle.

Fig.86

- **Écran graphique bicolore :** Présentant deux zones de couleurs différentes, (*Voir la Fig.86*) l'écran permet de distinguer facilement la nature des fenêtres contextuelles dans la zone jaune. La zone bleue sera réservée aux divers menus, aux données et surtout aux divers modes graphiques.
- **Clavier de maîtrise :** Utilisant des touches colorées leur répartition par zones bien repérées facilite considérablement le pilotage de la sonde de façon "instinctive". Plusieurs LEDs d'état permettent à l'opérateur d'avoir des informations d'état ou des avertissements. Le visuel est complété par un transducteur sonore générant des BIPs d'alerte.
- **Codeur rotatif incrémental :** Particulièrement convivial pour "se déplacer" dans les MENUS d'exploitation ou pour saisir des valeurs numériques, il ajoute un bouton poussoir central qui globalement sera dédié à la **Sécurité**. (*Voir la Fig.83*)
- **Zone technique :** Montrée en Fig.87 elle comporte cinq LEDs d'état qui peuvent être neutralisées en mode exploitation. (*Appréciables en maintenance du système.*)
- **Gestion des énergies :** Située à gauche du clavier pour la version "droitiers" elle permet librement de mettre en service ou d'isoler les diverses sources d'énergie interne au pupitre. En particulier cette zone inclut la gestion du rechargement des accumulateurs de puissance.
- **Droitiers / Gauchers :** Le pupitre tel qu'il est présenté sur la Fig.83 est optimisé en convivialité pour des droitiers. (*Droitiers au sens manuel du terme, car on est aussi différencié pour l'œil directeur, le pied d'appel etc.*) Les documents fournis avec ce didacticiel, et en particuliers ceux qui seront imprimés pour les façades du coffret sont prévus également pour des gauchers qui ainsi pourront réaliser un coffret "symétrique".



Fig.87

Masse totale du coffret avec deux accumulateurs de puissance : 1170g.

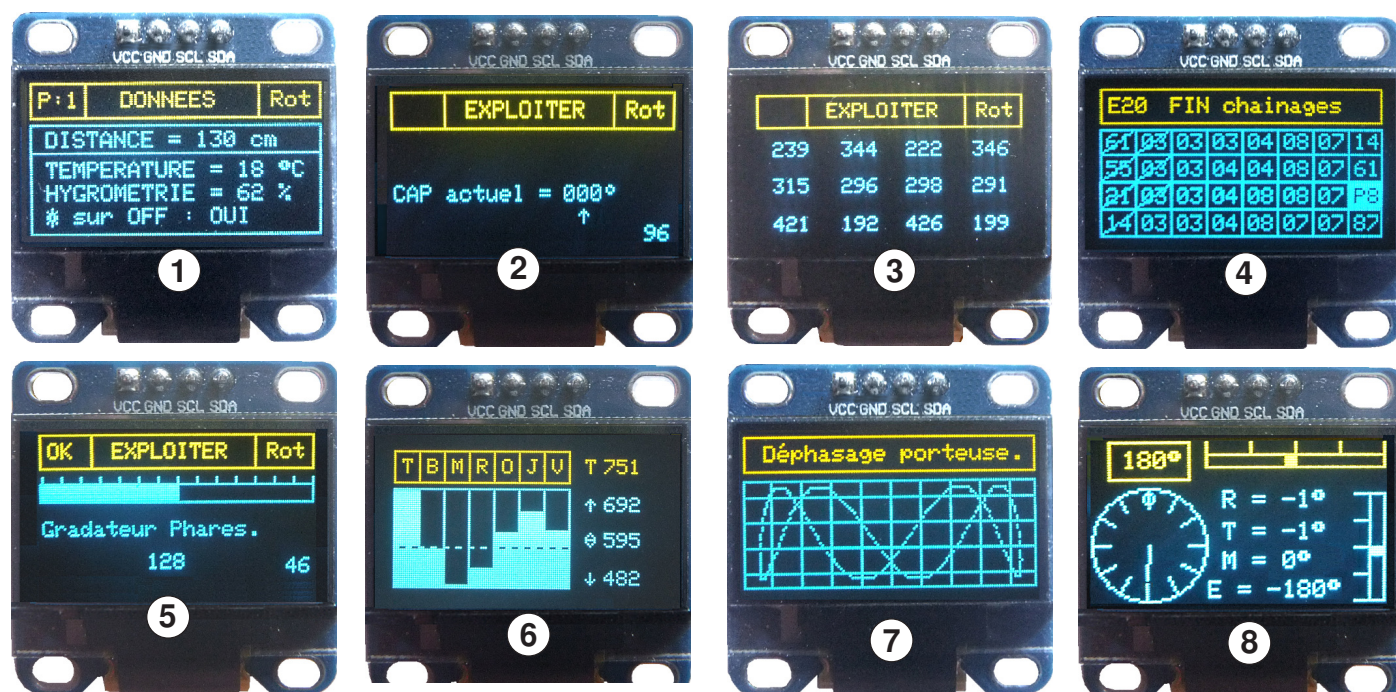
29) Présentation du pupitre.

A vous que c'est un chapitre "publicitaire" pour vous donner envie de réaliser le coffret. À l'instar du didacticiel sur la réalisation de la sonde, ce nouveau "paquet" contiendra également des fiches, mais ces dernières ne seront utiles que pour la "culture technique". Pour exploiter la sonde, c'est le petit livret [Manuel d'UTILISATION.pdf](#) qui contiendra toutes les procédures et protocoles relatifs au nouveau système. Aussi, si vraiment vous allez vous engager dans la réalisation du coffret de pilotage, je ne peux que vous inviter à imprimer et assembler ce livret. Consultez le chapitre 44 de la page 38 pour disposer des informations utiles. Pour la maintenance, [DOSSIER TECHNIQUE.pdf](#) sera assurément très utile, en particulier pour le protocole de rechargement des accumulateurs de puissance qui est assez particulier. Puisque nous en sommes à imprimer et assembler les petits livres qui accompagnent désormais JEKERT, créez également [VALIDER JEKERT.pdf](#) qui sera pratiquement incontournable quand on va procéder aux vérifications complètes des systèmes.

Outre une convivialité incomparable en exploitation, la version "PUPITRE" a débarrassé la sonde de certaines fonctions considérées comme secondaires pour dégager de la place en zone programme, et apporté des améliorations fonctionnelles majeures. On peut citer par exemple :

- * Un mode de stabilisation gyroscopique automatique particulièrement séduisant,
- * Un mode de pilotage manuel des moteurs infiniment plus sécuritaire,
- * Un mode APPRENTISSAGE considérablement développé avec un vrai éditeur de programme,
(On peut créer jusqu'à neuf programmes indépendants et les chaîner à convenance.)
- * Un vrai tableau de bord pour la navigation affiche les paramètres de base,
- * Possibilité de mémoriser jusqu'à huit postures personnelles en mémoire non volatile EEPROM,
- * La correction automatique du compas de route, (Boussole.)
- * Affichage graphique des données les plus pertinentes. (Panoramique ultrasons, luxmètre ...)

Pour achever ce "spot publicitaire", voici "en vrac" quelques copies d'écran pour vraiment vous faire envie ... but fondamental de toute publicité, fut-elle mensongère :



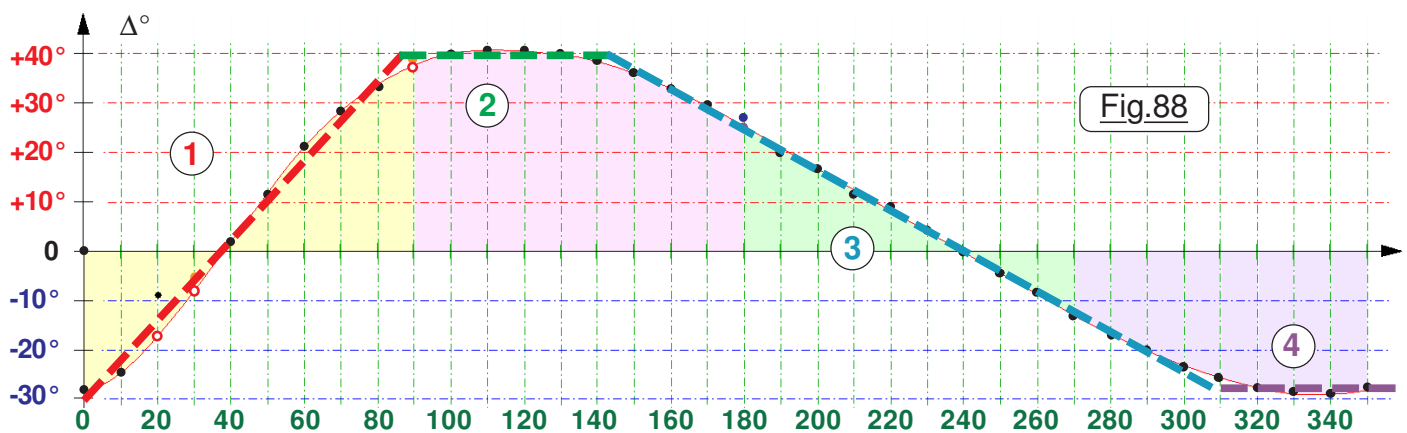
Ces quelques exemple "pour vous faire baver" sont issus du manuel d'exploitation qui en détaille les significations et symboles. En 1 nous avons l'une des pages écran des **DONNEES**. En 2 il y a saisie en cours du CAP magnétique vers lequel on désire naviguer. L'écran 3 affiche les consignes d'une posture alors que sur 4 le programme **P8** a été déclenché. Durant son déroulement les instructions sont cochées une fois réalisées. Mais ici le dernier "03" a provoqué une erreur **E20**. En 5 on visualise l'ajustement du gradateur alors qu'en 6 est affiché le dernier spectre colorimétrique qui a été enregistré. Sur 7 on simule un test des télétransmissions. Enfin en 8 on trouve le tableau de bord de navigation. **Et encore, ce déballage n'est qu'un tout petit échantillonnage !**

30) Correction automatique du compas de route.

Avant de commencer la description du pupitre, nous allons passer en revue les améliorations fonctionnelles ajoutées sur la sonde, et qui imposera une mise à jour du logiciel figé dans son l'ATmega328. Utiliser la belle [Fiche n°12](#) n'a rien de convivial en exploitation. Si l'on désire orienter la sonde à un CAP magnétique particulier, il faudra constamment interpréter les données de correction. Aussi, envisager une correction automatique par le calculateur de bord est séduisant. La courbe de correction affichée en rouge sur la Fig.150 en page 30 du **TOME 3** mis en ligne sur [ROBOT MAKERS](#) fait un peu peur, car pour la reproduire avec des équations trigonométriques on devine confusément qu'il va falloir introduire des harmoniques et de la développée de Fourier. BEURRRRKKKKK !

➤ **Retrouver le Nord.**

Outre le fait que ce ne sera pas facile de trouver les équations qui seront capables de traduire la courbe diabolique, leur traduction en langage C++ va engloutir tout le capital restant pour le code objet. Ce n'est évidemment pas acceptable. Toutefois, si l'on regarde un peu mieux le graphe de la Fig.150 et si l'on accepte une correction avec une imprécision pouvant aller jusqu'à $\pm 5^\circ$, la situation n'est pas aussi dramatique qu'il n'y paraît. Sur la Fig.88 on a superposé des droites à la



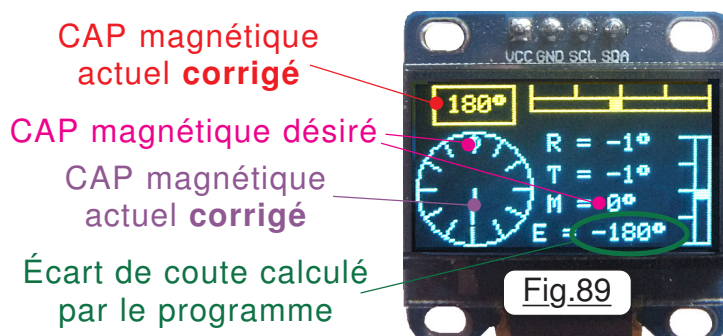
courbe rouge, et l'on peut vérifier que ces dernières masquent relativement bien l'ancienne trace. On peut alors considérer que la réalité sera proche des quatre cas **1**, **2**, **3** et **4**. Pour traiter les cas **2** et **4** la solution est immédiate, il suffit de retrancher 40° ou d'ajouter 29° au cap mesuré pour avoir l'orientation réelle. Pour les cas **1** et **3** il faut établir une **correction linéaire**.

- **1** : De 0 à 85° : **Interpoler** soit $0/+30$ à $85/-40$.
- **2** : De 86° à 143° : Enlever 40° .
- **3** : De 144° à 307° : **Interpoler** soit $144/-40$ à $307/+28$.
- **4** : De 308 à 359° : Ajouter 29° . (*Corriger si valeur >359*)

Il se trouve que **linéariser une valeur entre deux limites** est très facile avec le langage C++ de l'**IDE**, l'instruction `map()` est conçue dans ce but. Elle peut manipuler librement pour les limites des valeurs positives ou négatives et dans un ordre quelconque.

Noter que mathématiquement cette fonction `map()` revient à coder l'équation d'une droite soit une fonction dite du premier degré pour les lycéens qui ont bien appris. La procédure de servitude `Mesure_orientation_magnetique()` se terminera maintenant par la séquence de correction du compas de route, et le CAP magnétique affiché sera précis à quelques degrés ce qui est tout à fait correct. La Fig.89 situe ces paramètres de navigation sur le tableau de bord. **Pour conclure, ce qui**

semble le plus intéressant à souligner, c'est la façon dont on a traité par approximation un problème qui mathématiquement s'avérerait assez indigeste. Aussi, bien plus souvent qu'on n'y pense, ce type de transposition par linéarisation peut nous sortir élégamment de sévères impasses, et ce d'autant plus que les précisions que l'on programme dans nos applications sont souvent exagérées.



31) Stabilisation gyroscopique automatique de JEKERT.

C'est la cerise sur le gâteau. Imaginez le film d'animation suivant : JEKERT est posée sur un plateau dont on peut orienter la surface dans toutes les directions. On transmet l'ordre à cette dernière d'adopter la posture *Stable Transversal* disponible dans l'une des options du menu des **POSTURES**. On dispose des cales sous le plateau pour qu'il conserve une assiette inclinée en Tangage et en Roulis. Naturellement la sonde étant posée sur ce plateau en subit les inclinaisons. Puis, par l'entremise du menu des **OPTIONS** on active la stabilisation automatique. Immédiatement les servomoteurs s'animent et en quelques secondes la sonde s'est tortillée. Le châssis qui porte la centrale MPU6050 s'est positionné parfaitement horizontal. Suite à ce premier test, on prend en main le plateau et on change constamment les inclinaisons en Tangage et en Roulis. Les moteurs réagissent, les membres tremblent, frétilent et se déforment dans tous les sens, et le capteur gyroscopique se maintient en permanence à l'horizontale. C'est absolument magique ...

➤ **Principe de la stabilisation en ROULIS.**

Partant d'une posture qui généralement ressemblera à *Stable Transversal*, on comprend fort bien que la surface de sustentation étant rectangulaire, les comportements en Roulis et en Tangage seront bien différents. Le traitement qui sera effectué devra donc s'adapter à cette caractéristique morphologique de JEKERT. Dans le fonctionnement de la boucle de base, les deux axes sont traités simultanément. Néanmoins nous allons étudier séparément les deux directions.

Sur la Fig.90 la sonde est vue de l'arrière en regardant vers l'avant, et l'on suppose qu'elle est inclinée vers bâbord, le coté le plus bas étant donc à gauche. Dans ces conditions, la valeur retournée par le MPU6050 est positive pour le roulis. Pour compenser l'inclinaison,

on va se contenter de faire tourner les Tibias dans le

sens des flèches roses, ce qui va faire descendre les chaussettes des griffes de bâbord et monter celle de tribord. (*Flèches bleues.*)

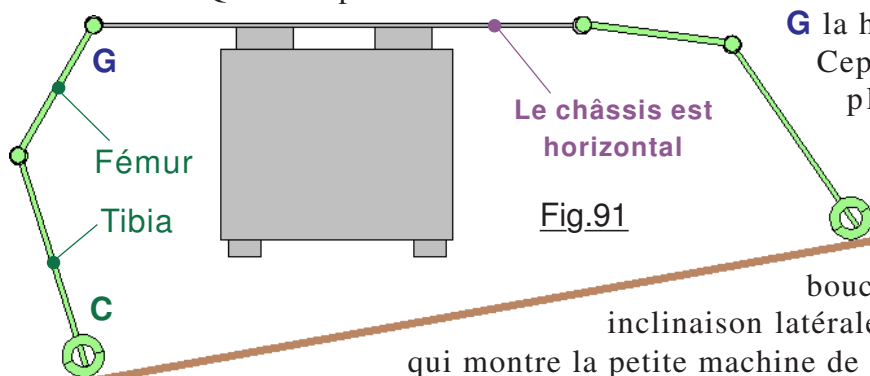
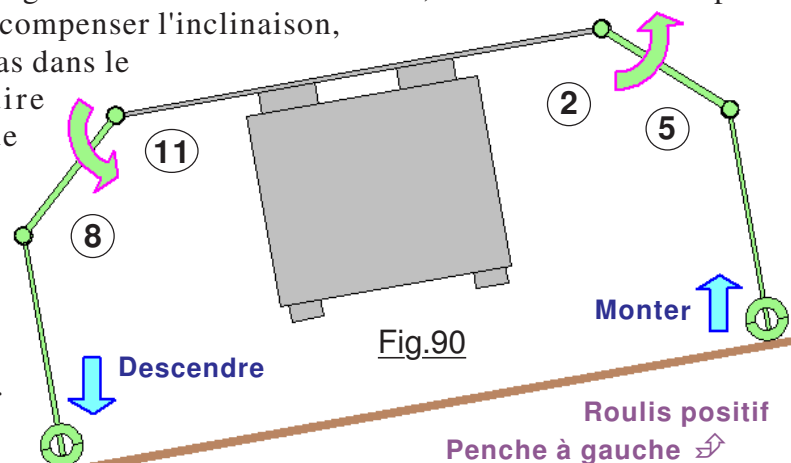
Sur la Fig.90 on a repéré les numéros des moteurs concernés. Ce comportement va se continuer jusqu'à ce que la centrale gyroscopique mesure un Roulis nul. La petite machine se retrouvera dans la configuration de la Fig.91 sur laquelle on constate que la largeur entre les divers points d'appui a tendance à légèrement augmenter. Ce phénomène génèrera un peu de frottement, mais surtout est favorable à la stabilité. La machine voit également sa hauteur diminuer, facteur également favorable au non basculement. Quand le point de contact **C** entre la chaussette et le sol dépasse la verticale du genou

G la hauteur à bâbord se remet à diminuer. Cependant à tribord la hauteur diminue plus rapidement, et la correction continue normalement. La limite de l'inclinaison correspond à la posture pour laquelle la chaussette de la

Jambe A vient en contact avec le bouclier. Ce cas limite correspond à une inclinaison latérale d'environ 9°. Profitons de la Fig.92

qui montre la petite machine de face posée sur une pente α inclinée à 9° sur tribord pour faire le point. Pour cette photographie c'est l'inverse de l'inclinaison étudiée sur les dessins. Issu du connecteur HE14 situé dans la région cachée **R** sur le dessus du circuit imprimé du condensateur de 470µF, la ligne d'alimentation de puissance traverse la machine pour ressortir à bâbord en **9**. Puis ce toron part en **6** passant vers l'arrière entre le bouclier et le circuit imprimé principal. On reconnaît facilement en **4** le LASER alimenté par la petite ligne **1**. En **7** on repère l'un des sabots du bouclier avec en **8** l'un des deux sabots pour l'anti-basculement. Pour

maintenir bien à l'horizontale le circuit imprimé **15**, donc la centrale gyroscopique,



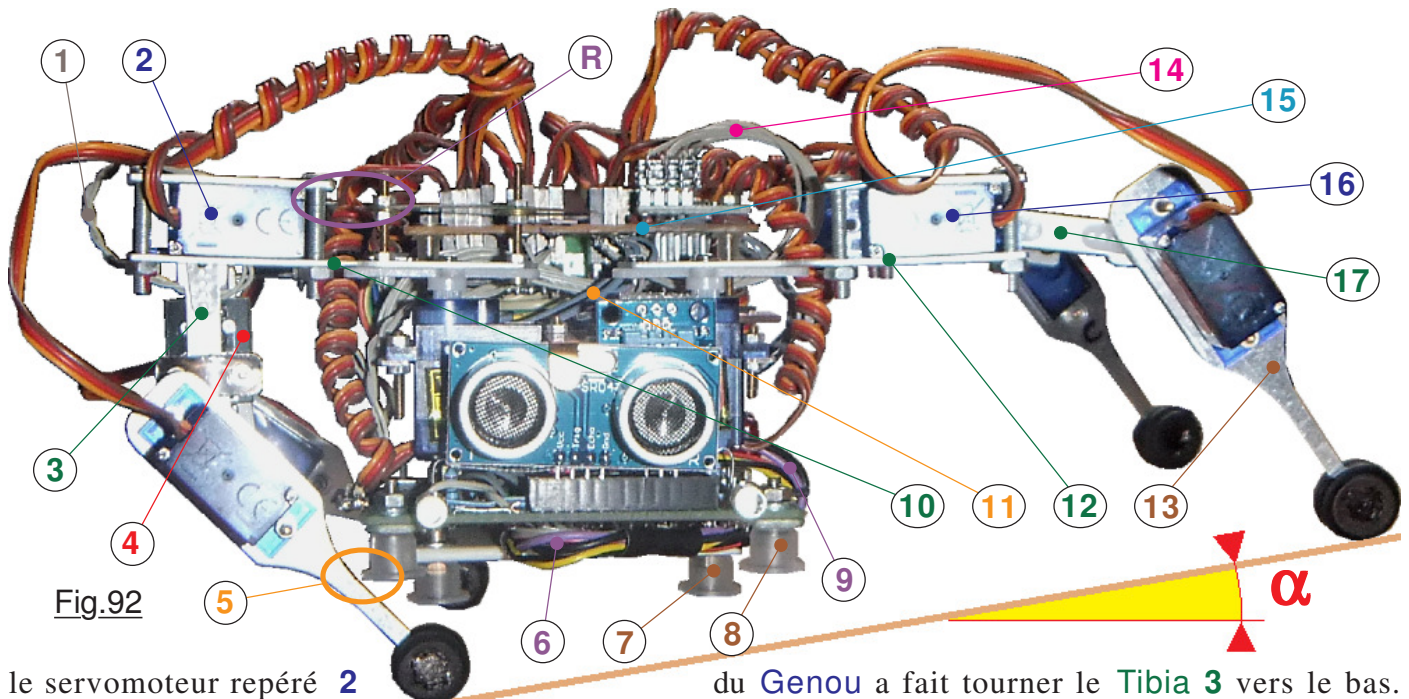
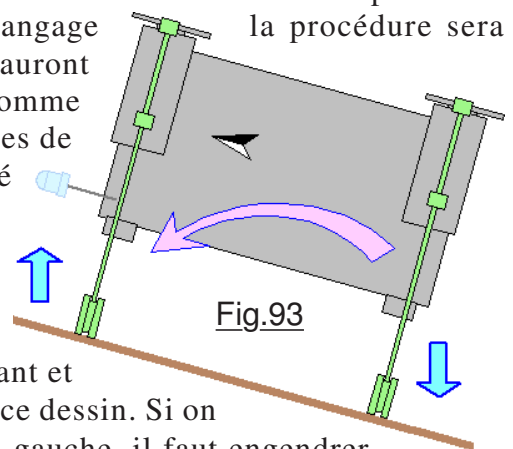


Fig.92

le servomoteur repéré **2** du **Genou** a fait tourner le **Tibia 3** vers le bas. Au contraire, le **Genou 16** a orienté le **Tibia 17** vers le haut. Seuls les **Genoux** tournent sur les servomoteurs, ce qui fait que les angles d'orientation entre les **Tibias** tels que **17** et les **Griffes** telles que **13** ne changent pas. Sur cette photographie la posture est presque en limite. Dans la zone **5** le contact entre la **Griffe** et le sabot du bouclier est presque établi, le moteur du **Pied** serait alors en situation de forçage. **(1)** Quand on incline de plus en plus le support sur lequel repose la sonde, il est assez amusant de voir JEKERT se déformer latéralement vers le bas de la pente et se tasser vers le sol. Le circuit imprimé **15** reste parfaitement à l'horizontal. En **11** l'on observe le toron torsadé de la ligne qui réunit la centrale inertielle au circuit imprimé principal. **Grande faiblesse de cette petite machine : Le manque manifeste de rigidité de certaines pièces mécaniques**, tout particulièrement les **Hanches** qui supportent dans de mauvaises conditions le poids de JEKERT. On voit très bien en **10** et **12** que les deux **Fémurs** fléchissent comme les ailes d'un avion.

➤ Principe de la stabilisation en TANGAGE.

Contrairement au Roulis, pour le Tangage il n'a pas été possible d'éviter des mouvements coordonnés, car tenter de n'agir que sur les **Fémurs** manque totalement d'efficacité, et pour arriver à compenser une faible inclinaison la machine se trouve complètement "écartelée". Du coup on ne peut corriger que de faibles inclinaisons, et surtout accompagnées de frottement de glissement inacceptable. Géométriquement, pour du roulis on monte ou l'on descend les chaussettes "en parallèle" de chaque côté. Si gauche monte, alors droite descend. Pour du tangage la procédure sera analogue, mais c'est devant et derrière que les déplacements auront des sens identiques. Par exemple si l'animal robotisé est cabré comme sur la Fig.93, les deux **Griffes** de devant doivent monter et celles de derrière descendre pour faire tourner le châssis comme montré par la flèche rose. Tous les dessins proposés dans ce chapitre sur le **Mode stabilisation gyroscopique** sont très schématisés mais **réalisés à l'échelle. Les proportions des membres sont respectées, ainsi que la position des articulations.**



Considérons la Fig.94 sur laquelle la sonde est vue de devant et nous "regarde". C'est le train avant qui est représenté sur ce dessin. Si on désire baisser les deux chaussettes simultanément à droite et à gauche, il faut engendrer les rotations **R** sur les **Genoux**. Les extrémités des **Tibias** qui matérialisent les **Pieds** montent en **M**. Par déplacement réciproque on fait **Descendre** l'avant de la sonde, mouvement constituant précisément le but de cette manœuvre. La rotation **R** provoque une translation latérale **T** du **Pied** qui déporte aussi l'extrémité du **Tibia** vers l'extérieur. En soit ce petit déplacement n'est pas

(1) Actuellement les sabots anti-basculement sont remplacés par un Pare-chocs.

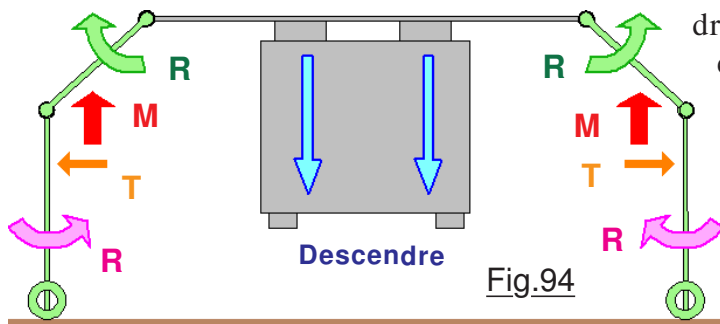


Fig.94

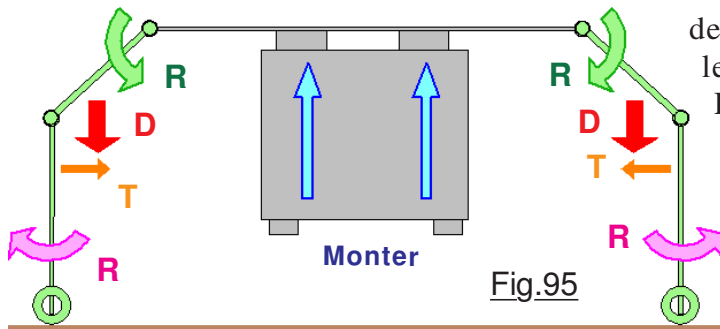


Fig.95

dramatique. Par contre, la rotation **R** ramenée en extrémité de **Griffe** engendre un écart latéral considérable. La machine est "écartelée et peut s'effondrer sous l'influence des efforts de flexion. C'est pour parer ce problème, que nous allons simultanément coordonner une rotation inverse **R** de la **Griffe**. L'idéal pour éviter tout frottement, et donc de l'usure par abrasion, consisterait à annuler tout déplacement au sol de la chaussette. Dans la pratique, pour minimiser le code objet on se contente d'une approximation. Du coup, au cours de la stabilisation en Tangage on observera de petites variations de l'empattement, ces dernières restant tout à fait acceptables. Pour maintenir parfaitement constante la distance qui sépare la chaussette du plan médian de la machine il faudrait coordonner **R** et **R** par des amplitudes de

consignes angulaires qui relèvent d'un calcul trigonométrique goinfre en octets de code machine. On se contente de faire varier linéairement les consignes qui traitent les rotations **R** et **R**. Avec la Fig.94 nous avons vu comment faire **Descendre** le nez de la petite machine. Si on doit le faire **Monter**, car la machine au lieu d'être cabrée s'est engagée dans une pente descendante, on fait l'inverse sur les rotations et provoquer de ce fait les descentes **D**. Naturellement, si au lieu de vouloir faire monter ou descendre l'avant de la machine on veut obtenir cet effet sur l'arrière, on procède exactement pareil mais sur les servomoteurs des **Jambe B** et **C** au lieu des membres sur **A** et **D**. Sur la photographie de la Fig.96 la sonde repose sur le plateau **P** dont l'inclinaison est maintenue au moyen de la cale **X**. On constate que l'amplitude de la stabilisation est considérable puisque sur cette vue la pente est d'environ 30°. Cette efficacité résulte de la distance relativement faible qui sépare le train avant du train arrière.

D évoilant une foule de petits détails, la Fig.97 montre la sonde vue de dessus quand elle est en posture de stabilisation gyroscopique. Conformément aux explications relatives à la correction d'assiette en Tangage, pour **Descendre** les **Griffes** à l'arrière les **Fémurs** tournent vers le bas. On remarque bien sur la photographie Fig.97 que les **Fémurs** sont pratiquement verticaux. La rotation **R** compense un peu trop **R** ce qui engendre un glissement de la chaussette **Gint** vers l'intérieur. Si l'on adopte un support pas trop rugueux, ce glissement reste toutefois acceptable. Du reste, lorsque la sonde s'est stabilisée en attitude horizontale, on peut déclencher "**p14***" pour libérer les efforts. Pour monter les **Griffes** à l'avant les **Fémurs** tournent vers le haut écartant les **Genoux** vers l'extérieur. La rotation **R** ne compense pas assez **R** ce qui engendre un glissement de la chaussette **Gext** vers l'extérieur. On voit parfaitement sur cette image que l'animal mécanique est plus écartelé à l'avant qu'à l'arrière. (*Observation de la position des appuis au sol.*)

Pour réduire ces déplacements parasites, il a été tenté d'établir une meilleure proportionnalité dans les deux mouvements coordonnés. La technique a consisté à doubler la variation de consigne sur l'élément trop "lent". Les dérapages au sol étaient effectivement bien moindres.

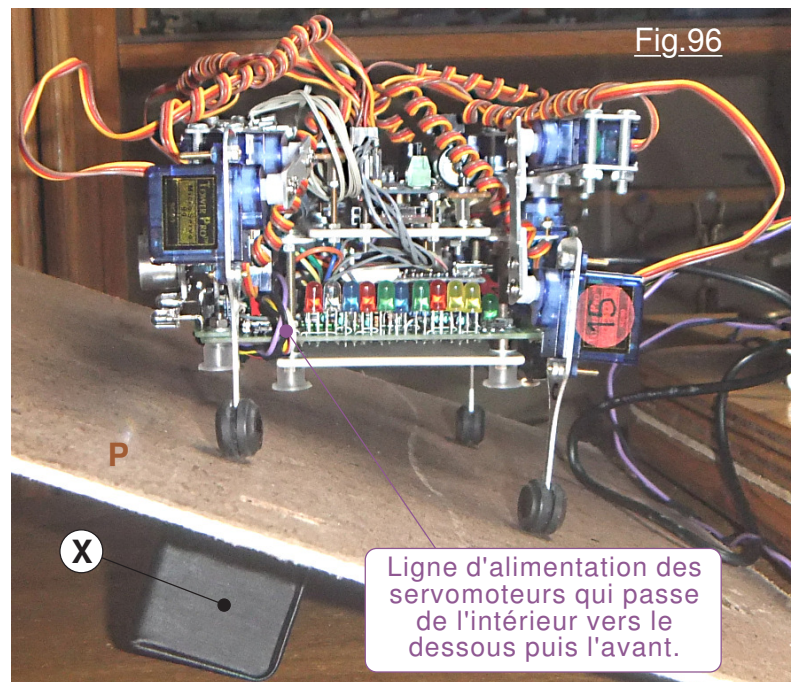
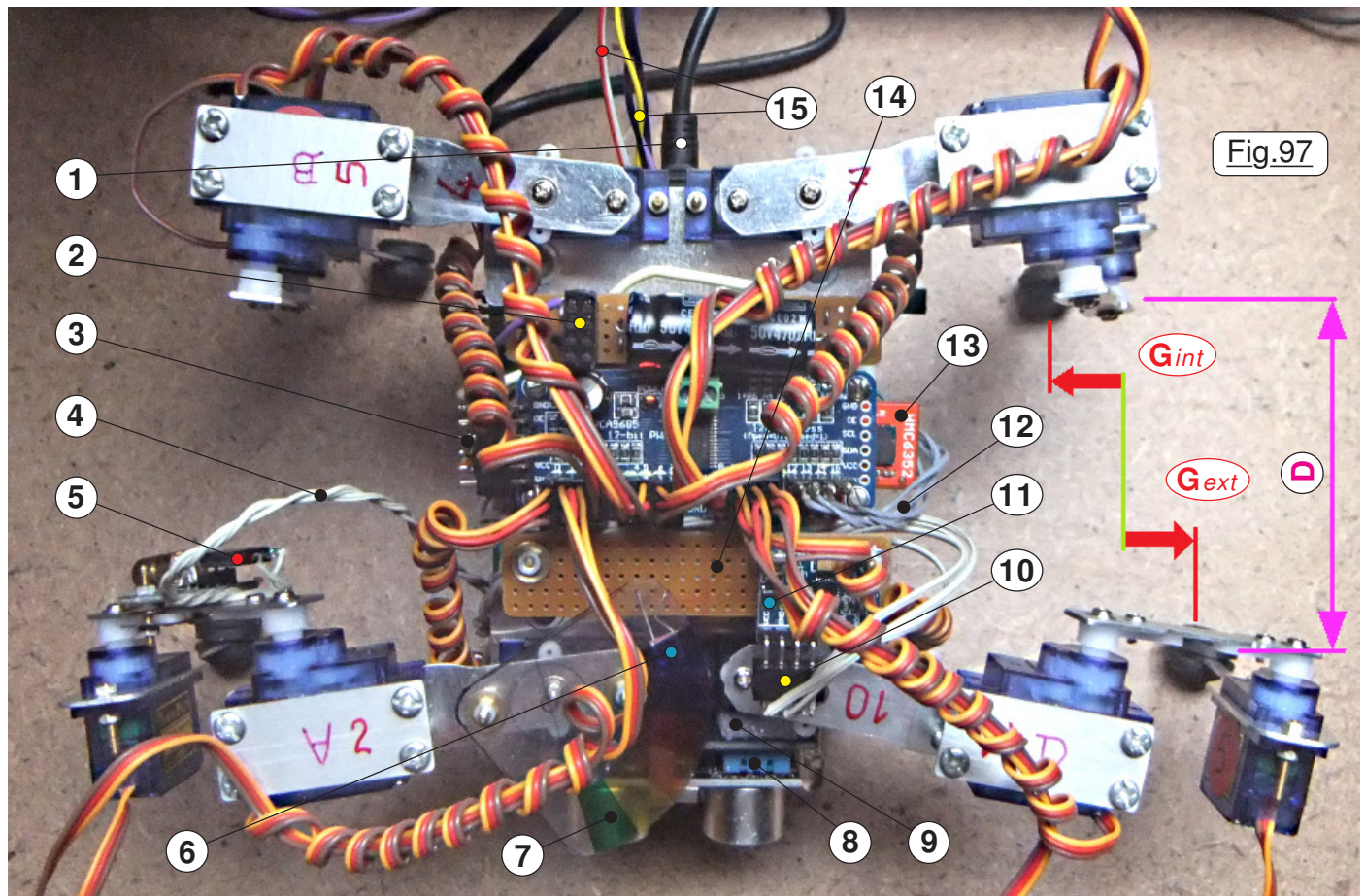


Fig.96

En contre partie, la flexion des membres associée à des accélérations plus importantes génèrent des surcompensations se traduisant par une oscillation entretenue infiniment plus néfaste que le petit inconvénient du frottement au sol des chaussettes. La photographie de la Fig.96 démontre que l'efficacité de la correction en Tangage est impressionnante puisqu'autant en cabrage qu'en piqué on peut compenser facilement des pentes de 30°. Cette particularité s'explique par le fait que la distance **D** qui sépare les appuis avant des appuis arrière est relativement faible. Du coup un différentiel de hauteur entre les deux trains se traduit par une variation en tangage importante.

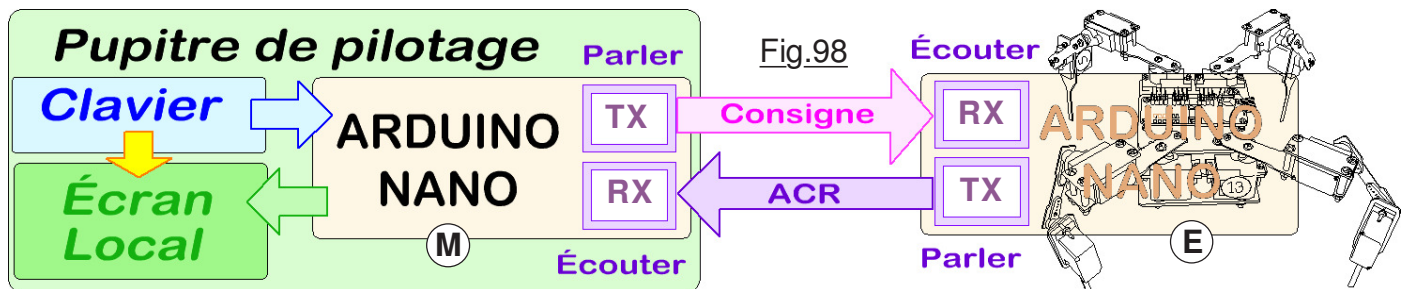


Revenons à cette photographie pour observer une fois de plus la petite machine dans sa version complète. On va se faire une petite révision, ce n'est jamais redondant. Chaque image apporte une foule de détails. Lorsque vous aurez un quelconque doute, les photographies seront vos meilleures alliées. En **1** le cordon USB passe entre les deux moteurs arrière et va vers le P.C. pour un pilotage par le moniteur de l'**IDE**. (Ici la machine est en cours de développement. C'est un programme démonstrateur qui sur **JEKERT** permet d'effectuer la mise au point des techniques. Le pupitre n'est pas encore réalisé, donc le pilotage se fait par des commandes spécifiques envoyées à partir du moniteur de l'**IDE**. En ce qui vous concerne, cette étape n'existera pas. Vous utiliserez directement le pupitre pour effectuer la manipulation qui montre le comportement de l'insecte robotisé quand on prend un malin plaisir à le déstabiliser.)

En **2** on distingue le connecteur HE14 double qui branche la ligne d'alimentation des moteurs sur le circuit imprimé du condensateur de 470 μ F avec en **3** le connecteur HE14 reliant le multiplexeur au circuit imprimé principal. La ligne **4** alimente le LASER **5**. Discrète, en **6** la cellule photorésistante se cache sous le filtre coloré **7**. Le capteur d'hygrométrie **8** est poussé vers l'arrière par le module du télémètre et s'appuie sur le petit bloc de mousse synthétique **9**. Le connecteur HE14 en **10** est branché sur la centrale gyroscopique **11**, sa ligne de fils torsadés gris arrive du dessous du circuit imprimé **14**. Venant du circuit imprimé principal, la ligne **12** relie les sorties S12 à S15 du multiplexeur. (Ces quatre sorties commandent quatre témoins lumineux sur le circuit imprimé principal.) Bien visible en **13** le module de la boussole statique dépasse un peu du multiplexeur. Enfin en **15** s'éloigne de la sonde par l'arrière le cordon ombilical de pilotage qui ira au pupitre de commande ainsi que la ligne d'alimentation en puissance des servomoteurs.

32) Dialogue entre machines.

Faire dialoguer deux machines n'est absolument pas élémentaire. Contrairement aux bavardages humains pour lesquels les messages sont transportés par l'air ambiant, sur des machines ce sont des impulsions binaires qui se succèdent à cadences élevées sur les lignes filaires de communication. Dans notre cas ce seront **TX** et **RX** sur **D0** et **D1** qui seront affectées à cette fonction palabresque. Non seulement pour se comprendre les deux machines devront échanger des propos




par le biais de protocoles très précis, mais surtout des problèmes de synchronisation sérieux pourront rompre et bloquer "les discussions". La Fig.98 résume l'architecture globale qui présidera la conduite des études indigestes à venir. Le principe retenu pour dialoguer est celui du *"chacun son tour"*.

➤ Dialogue Homme / Machine : Le retour.

Globalement nous allons retrouver les préceptes qui dominaient le dialogue entre le P.C. et JEKERT. La grande différence réside dans le fait que maintenant c'est une carte Arduino NANO qui sera chargée de gérer un Clavier et un *Écran local* intégrés dans un Pupitre de commande que l'on nommera aussi *Raquette de commande*, vocable suggéré par ses dimensions les plus réduites possibles. Décortiquons le principe d'un échange "verbal" entre l'Humain et les deux Machines :

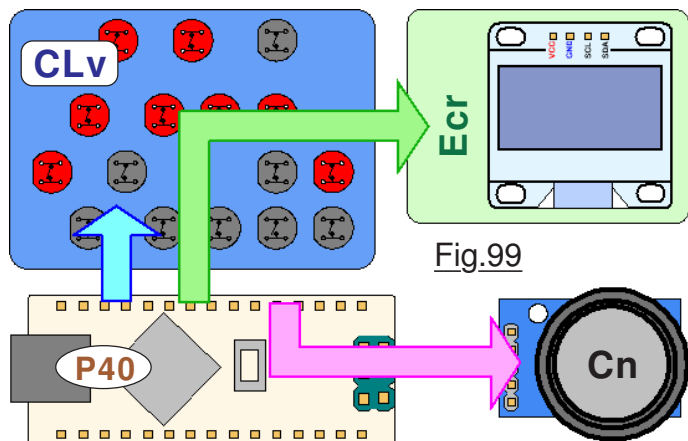
- 1) Le logiciel *Maître* du pupitre **M** est en attente d'une directive issue du *Clavier*, l'information qui en résulte transitant comme l'indique la flèche bleue. (Par exemple on cliquera sur un bouton poussoir.)
- 2) Pour aider le technicien qui exploite la sonde, **M** affiche (*Répétiteur*.) sur l'*Écran local* des informations traduisant les actions de l'opérateur sur le *Clavier*. (Flèche symbole jaune.)
- 3) La directive de l'opérateur est alors traduite en *Consigne* et **M** va *Parler* sur **TX**. Puis, comme on privilégie ici un *mode alternat*, **M** va immédiatement *Écouter* sur sa broche **RX**.
- 4) Le *logiciel Esclave* **E** étant à l'*Écoute* sur sa broche **RX** enregistre le message reçu.
- 5) L'*analyseur syntaxique* décode cette *Consigne*. Si elle est valide, immédiatement elle est exécutée. Après une éventuelle action valide, **E** ensuite *Parle* sur **TX** pour *AC*cuser *R*éception.
- 6) Le *logiciel Maître* du pupitre **M** décortique l'*ACR* reçu sur **RX** et rend compte sur l'*Écran local* après une mise en forme, puis immédiatement se replace en surveillance du *Clavier*.

 **NOTE** : À partir d'ici, le mot *Consigne* ne désignera plus une valeur de positionnement angulaire pour les servomoteurs, mais un *ordre à exécuter* par le *logiciel Esclave*.

Étant donné que les affichages sur l'*Écran local* sont gérés par la carte **M**, il devient totalement parasite d'encombrer les échanges *Consigne* / *ACR* par du "bavardage" de présentation claire pour l'humain. Rendre ces "données Machine" aussi compactes que possible aura deux conséquences positives. D'une part les dialogues binaires prendront moins de temps en ligne **TX/RX**. D'autres parts le *logiciel Esclave* va considérablement maigrir, libérant ainsi beaucoup de place, ce qui a servi à intégrer les nouvelles fonctionnalités. Il faut donc concevoir entièrement les protocoles de dialogue entre les deux ATmega328. Il importe aussi de déterminer quel type de clavier sera utilisé, et choisir un module électronique pour équiper l'*Écran local* de la Raquette de commande. Clavier, *Écran local*, diodes électroluminescentes et codeur rotatif incrémental constituent un tout intimement lié qui a conduit au pupitre actuel. N' imaginez-pas qu'un simple coup de baguette magique a dévoilé la solution actuelle. Cette dernière a émergé lentement, suite à de longues études qui progressivement ont élaboré les concepts et les solutions actuelles. Sur le site de ROBOT MAKERS les sept TOMES décrivent en détail la longue route pour aboutir au prototype actuel. Dans le cadre de ce didacticiel simplifié, on va se contenter de matérialiser le PUPITRE sans chercher spécialement à justifier ou comprendre les solutions adoptées. C'est parti pour la pratique ...

33) Structure matérielle et logicielle du pupitre de commande.

Globalement, le pupitre de commande sera équipé d'un clavier **CLv** à seize touches. *La répartition géométrique sur le coffret est définie soigneusement en vue d'une convivialité d'utilisation maximale.* L'avantage à créer entièrement un clavier personnalisé, c'est précisément de pouvoir "dispenser" géométriquement les touches à notre convenance. On se doute que leur répartition spatiale sera compactée au maximum pour des raisons d'encombrement. Enfin nous pouvons librement intercaler entre les touches des témoins lumineux pour assister l'opérateur, et



l'on ne va pas s'en priver. Le besoin montre que seize boutons poussoir ne seront pas de trop. Ce nombre n'est pas le fruit du hasard. Il correspond à une optimisation fonctionnelle.

Pour piloter le LASER, la TORSION, le niveau des éclairages, nous utilisons sur le prototype un potentiomètre. Aussi, personnellement je n'aime pas trop ces dispositifs qui dans le temps se mettent à "cracher". Par ailleurs, balader des signaux analogiques de tension faible sur de longues lignes filaires les rend très sensibles aux parasites électromagnétiques environnants. Aussi

ce dernier sera remplacé par un Codeur Numérique **Cn**. Enfin nous visualiserons les données sur un écran graphique **Ecr**. La Fig.99 résume assez bien les divers périphériques qui devront être gérés par le logiciel **P40_PGM_MAITRE_RAQUETTE.ino**.

➤ **La gestion d'un clavier multiplexé.**

Prendre en compte un clavier à seize touches en n'utilisant que quatre sorties binaires et une entrée analogique sur l'ATmega328 n'a strictement rien de magique. Le schéma électronique et la technique de programmation relèvent d'une optimisation à outrance des ressources du microcontrôleur. La [Fiche n°1](#) dévoile les secrets de cette optimisation. Seule différence par rapport avec la réalité, sur la Fig.1 le clavier est représenté avec des touches disposées géométriquement en ligne/colonnes alors que dans la réalité leur répartition "géographique" est plus subtile. L'entrée analogique **A6** a été sélectionnée, car elle ne peut pas fonctionner en sortie binaire. Autant la réserver à cette application en libérant de ce fait les broches les plus universelles.

➤ **Changement de stratégie pour optimisation.**

Anciennement la commande "**b***" était réservée pour figer les lumières quand le potentiomètre servait à pointer le LASER, ou à déplacer en TORSION. Pour son compte, la commande "**n***" neutralisait le potentiomètre quand on changeait de moteur cible en mode manuel. Il sera plus logique d'affecter un indicateur pour préciser sur qui on agit quand le codeur rotatif est en cours d'utilisation. Les commandes "**b***" et "**n***" devenues inutiles libèrent leurs LEDs respectives. En revanche, il devient utile de savoir quand le potentiomètre virtuel est arrivé en butée logique puisque le codeur incrémental n'a pas de limite angulaire.

La LED jaune de l'ancien "**n***" signalera la saturation dans le sens Positif. La LED blanche de l'ancien "**b***" signalera la saturation dans le sens Négatif. Émuler un potentiomètre virtuel libère l'entrée **A0** qui deviendra une sortie pilotant une LED rouge "**BUZZER**" montrée sur le Fig.100 soudée sur un petit adaptateur HE14 à trois broches se branchant à la place de l'ancien potentiomètre. C'est un témoin visuel "système" servant au développement "silencieux" du logiciel. Le **DOSSIER TECHNIQUE** rend compte de ce changement dans les pages P2 et P3. Par ailleurs, en page P36 on trouve la nouvelle affectation des LEDs d'état de la sonde. Anticipant sur la suite du didacticiel, vous trouverez en pages P4 et P5 le schéma de l'électronique du pupitre ainsi que l'affectation fonctionnelle des broches du microcontrôleur. On notera au passage que le **DOSSIER TECHNIQUE** prend en compte la sonde et le pupitre, avec des développements électroniques, logiciel et matériel. Tout particulièrement en page P18 se trouve le *Protocole pour recharger les accumulateurs.*



34) Gestion d'un codeur rotatif indexé.

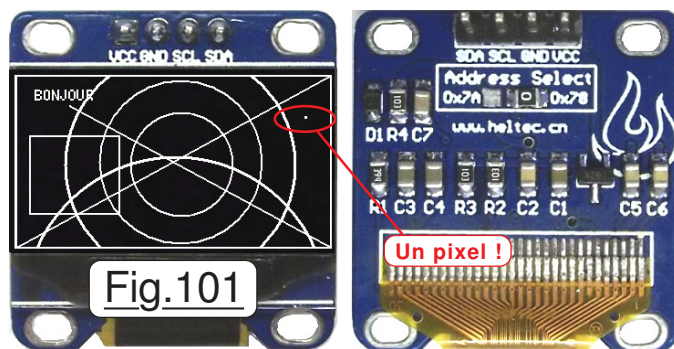
Exclure le potentiomètre qui servait à pointer le LASER, à augmenter ou diminuer la puissance lumineuse ou diriger la TORSION, a été justifié pour des raisons techniques. Composant peu fiable par nature, le remplacer par des technologies plus robustes s'impose naturellement. Ce n'est toutefois pas la seule raison qui incite fortement à le remplacer par un codeur rotatif incrémental. C'est surtout un critère de qualité opérationnelle d'utilisation de notre pupitre. En effet, le nombre de **Consignes** dépassait les cinquante dans l'ancienne version. (23 programmes et environ 31 consignes sous forme d'un seul caractère.) Il n'est pas question d'envisager un clavier qui comme celui d'un P.C. permet d'envoyer tous ces "textos". Nous allons forcément procéder par des sélections sous formes de "MENUs Déroulants". Hors la rotation d'un potentiomètre fait au maximum 270° et surtout on domine mal sa "progressivité". Alors qu'**avec un codeur incrémental on sait immédiatement que chaque clic d'indexation engendre un pas**. Clic clic clic : On s'est déplacé de trois items dans la liste, en avant ou en arrière en fonction du sens de rotation. Coté convivialité on ne peut pas trouver mieux. Du coup, ce qui est très pratique pour déplacer un index sur un menu écran l'est aussi pour diriger un LASER ou doser un éclairage. Le pupitre de JEKERT sera donc muni d'un tel dispositif si agréable à employer, surtout sur un petit pupitre où il y a peu de place.

➤ Mise en œuvre d'un codeur rotatif incrémental.

Qualifié d'incrémental car bien qu'il puisse tourner sans butée autant que désiré, il ne peut occuper de manière stable que des positions bien précises. Il y a quarante indexations par tour pour le modèle utilisé. Généralement les codeurs Angulaires fournissent l'information de position sur plusieurs sorties binaires. Pour résoudre le problème des fausses numérisations par décalages temporels subtils sur les changements d'états de plusieurs sorties, on utilise des codes binaires intelligents tel que le code Gray par exemple. Grosse différence, **dans notre cas ce n'est pas une position angulaire que l'on désire prendre en compte, mais un incrément de rotation**. Du coup deux sorties sont suffisantes pour pouvoir effectuer du "comptage" et déterminer le sens de rotation. Peu importe la gestion logicielle qui est minutieusement explicitée dans le **TOME 5** mis en ligne sur **ROBOT MAKERS**, on va se contenter dans ces lignes du minimum. Toutefois, la technologie de ces composants ainsi que le principe d'utilisation sont précisés sur la **Fiche n°2**. C'est surtout le branchement donné en page 4 du **DOSSIER TECHNIQUE.pdf** qui nous concerne. Notons que pour gérer ce composant seules deux broches sont suffisantes, connectées respectivement sur **D2** et **D3** de l'ATmega328. Ces broches sont également en relation avec deux diodes électroluminescentes jaunes qui peuvent informer le programmeur de l'état logique de ces dernières. Le "simple utilisateur" ne sera pas vraiment concerné par ces informations techniques. Il suffira de ne pas alimenter la zone de **Gestion technique du dialogue** en coupant **INV**. (Voir la Fig.83)

35) Gestion d'un petit écran graphique.

Quand à la réception du coli postal on ouvre le paquet et que l'on sort le minuscule module OLED Adafruit SSD1306, on reste dubitatif en considérant ses dimensions vraiment discrètes. L'écran en verre ne mesure que 27 x 15mm. Tant que l'on n'a pas vu ce que donne une si petite image, on ne peut que douter de l'efficacité de ce composant. Rapidement on "triturer" un programme pour afficher le plus grand cadre possible, pour se rendre compte que ses dimensions ne font plus que 22mm x 16mm. Le doute devient encore plus sérieux. Et pourtant, dès que l'on arrive à afficher du texte, la plus petite police reste parfaitement lisible. "Traçons" ensuite des traits, des rectangles, des cercles, une sinusoïde, tout ce petit monde graphique s'avère parfaitement visible, et immédiatement le doute s'évapore. Un si petit écran remplit totalement sa mission, comme vous pouvez le vérifier sur la Fig.101, le plus petit texte en taille ainsi que les graphes sont très facilement "lisibles". Sur cette photographie, le texte BONJOUR utilise la plus petite police de caractères soit des individus de 5 x 7 pixels. (Soit une matrice de texte bien classique.) La netteté est



en outre assez dégradée par le manque de définition de cette photographie, sur laquelle figure en haut à droite un pixel isolé. Il n'y a plus qu'à le mettre en service ce qui oblige à :

➤ Installer la bibliothèque indispensable.

Programmer un afficheur graphique est particulièrement laborieux, car il faut définir les caractères textuels, tracer des droites, des cercles, limiter ces derniers à convenance, gérer les débordements de définition de l'écran. Heureusement pour nous, quand un composant du commerce s'avère particulièrement séduisant, on trouve généralement une bibliothèque bien pensée créée par un passionné, ou souvent directement par le concepteur du produit commercialisé. Le programme du pupitre n'est pas dans vos priorités, puisqu'il est entièrement livré "clef en main". Toutefois, pour le que compilateur puisse téléverser ce dernier sur l'ATmega328, il devra disposer de la bibliothèque spécifique qu'il faut lui mettre à disposition. Dans notre cas il suffit de télécharger pour l'**IDE** la bibliothèque **Adafruit_ssd1306sy.h** particulièrement efficiente. Il nous faut installer cette "library", rassurez-vous, ce n'est pas bien compliqué :

- 1) Téléchargez la bibliothèque dans un répertoire quelconque, (*Voir le lien chapitre suivant.*)
- 2) Décompresser le fichier ZIP dans ce répertoire.
- 3) Placer le dossier de la bibliothèque décompressé dans un dossier de votre choix, celui dans lequel vous préserverez toutes les "library" que vous ajouterez à votre environnement **IDE**.
- 4) Déclarer la nouvelle bibliothèque dans la liste disponible :

Croquis > **Importer bibliothèque..** > **Add Library...** >

Indiquer le chemin de votre dossier personnel sur le H.D. (*Précisé en étape 3.*) La bibliothèque est alors ajoutée dans un dossier personnel :

<C:\Users\VotreOrd\Documents\Arduino\libraries>

ou éventuellement dans

<C:\Users\user\Documents\Arduino\libraries>.

- 5) Recharger un programme quelconque. Cliquer sur **Croquis** puis placer le curseur de la souris sur **Importer bibliothèque..** : En **1** on découvre les bibliothèques disponibles par défaut et en **2** les bibliothèques importées dans l'**IDE** par vos soins. Vous devez y trouver **Adafruit_ssd1306sy.h** comme mis en évidence sur la Fig.102 dans l'encadré rouge.

Ouf, **P40_PGM_MAÎTRE_RAQUETTE.ino** pourra être compilé puis téléversé.

Pour les plus curieux d'entre vous, la **Fiche n°3** résume les instructions de la bibliothèque.

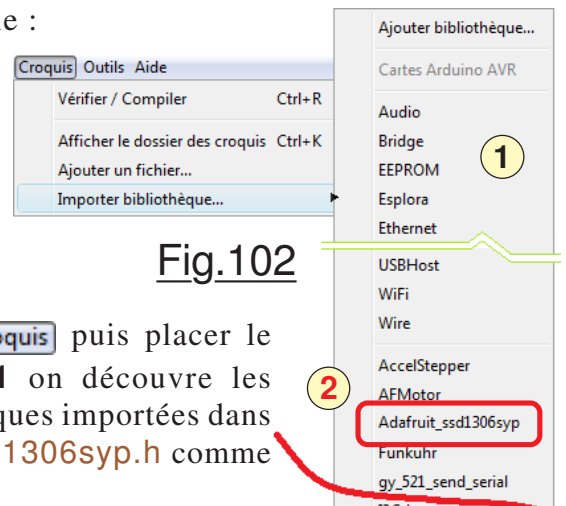


Fig.102

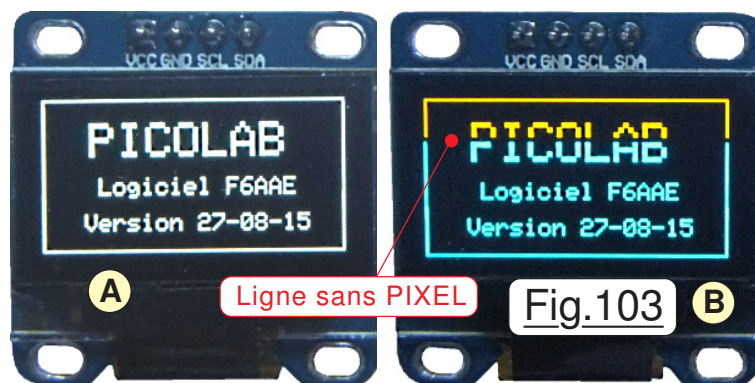
➤ Faisons connaissance avec l'afficheur OLED SSD1306.

Plusieurs bibliothèques personnalisées sont disponibles sur la toile. Il faut bien faire un choix. Celle utilisée dans les programmes proposés dans ce didacticiel m'a semblé la plus "populaire". Il s'agit de la "library de référence" **Adafruit_ssd1306sy.h** disponible sur de nombreux sites. Par exemple on peut aller sur : http://www.codeforge.com/read/242813/Adafruit_ssd1306sy.h__html Disposant de ses fonctionnalités, programmer le SSD1306 devient un jeu d'enfant ... Encore que ! En effet, pour tracer tout ce que l'on désire, le code est immédiat pour peu que l'on s'imprime une grille de 64 x 128 carreaux pour repérer les coordonnées. Mais la petite puce fait de la résistance. Le petit écran ne se laisse pas faire et nous oppose dès le départ deux difficultés.

Première embuche : L'approvisionnement peut nous réserver une petite surprise. Ayant mis en service le tout premier de ces composants dans le PICOLABORATOIRE décrit sur : <http://www.robot-maker.com/ouvrages/apprendre-a-programmer-arduino-en-samusant/> c'était un écran monochrome qui sur la Fig. 103 est présenté en **A**. Ayant commandé la même référence pour avoir un exemplaire de rechange en cas de "pépin", (*Et oui, ça arrive !*) j'ai reçu un modèle un peu différent. Comme montré sur la Fig.103 en **B** il est bicolore, avec une zone orange de 16 lignes en haut, une ligne de séparation sans pixel (*Repérée en rouge.*) et le reste inférieur en bleu. Du coup, avec le même programme on constate qu'en blanc sur la photographie **A** il y a continuité du tracé, alors qu'en bicolore sur l'image **B** il y a une discontinuité entre les deux couleurs.

Renvoyer le produit au fournisseur était une option, mais tout compte fait, la couleur

rend l'écran bien plus attrayant. En tenant compte dans les programmes de la ligne sans pixels, on peut concevoir un logiciel qui favorise l'esthétique, les deux couleurs mettant en évidence des informations de type différent. La version en blanc est bien moins visuelle. Personnellement je trouve que la couleur est très séduisante, et l'inconvénient de discontinuité faible puisqu'il n'est présent qu'en mode "dessin plein écran". Pour que chacun puisse choisir à sa guise le modèle d'afficheur utilisé, tous les programmes seront conçus sur l'afficheur bicolore avec prise en compte de la discontinuité. Ainsi les deux versions du composant seront totalement compatibles.



(ATTENTION : Les deux photographies **A** et **B** dégradent considérablement la finesse réelle de l'afficheur, car l'appareil de prise de vues n'est pas prévu pour saisir des clichés en objectif rapproché.)

Deuxième difficulté : Fonctionnant en allumant et en éteignant des pixels, l'effacement de l'afficheur total ou partiel prend du temps. Quand on recherche de la rapidité de rafraîchissement sur l'écran, il faut faire un gros effort logiciel pour que le code soit optimisé en rapidité d'exécution. Y compris dans ces conditions il ne sera pas possible d'obtenir la "fluidité" d'autres écrans du commerce. Rassurez-vous, mis à part quelques cas particuliers, et d'un usage marginal, dans l'ensemble, les performances de l'affichage au final restent très "compétitives".

Choix de l'afficheur : Il existe dans le commerce une quantité impressionnante de composants pour afficher sur des matrices textuelles ou graphiques. Des monochromes, des multicolores, sous toutes les tailles et définitions, dont certains sont tactiles. Ayant expérimenté pas mal de ces périphériques, mon choix c'est porté sur l'afficheur **graphique** OLED pour plusieurs raisons. En particulier il est minuscule et convient à merveille pour le petit pupitre de commande. J'affectionne tout particulièrement le bicolore jaune et bleu d'où son adoption.

36) Protocole d'exploitation de la sonde JEKERT.

Particulièrement délicat à élaborer, l'agencement d'un contexte d'utilisation d'un système quel qu'il soit se heurte à une difficulté particulière liée au fait que les divers MENUS qui permettront le pilotage sont liés à la présentation du pupitre. Hors pour disposer géométriquement ce dernier il faut savoir à quoi serviront les touches, donc avoir présent à l'esprit les protocoles envisagés. Bref, c'est le serpent qui se mord la queue. Dans la conception d'une Raquette de commande c'est de loin la phase la plus délicate car elle sera suivie de longues études logicielles qui seront remises en cause si ce que l'on a imaginé à ce stade aboutit à une NON convivialité et oblige à mettre à la poubelle des heures d'études logicielles. La gestation définitive du pupitre n'a pas été un long fleuve tranquille, loin s'en faut. Nous allons dans le cadre simplifié de ce didacticiel aller directement à l'implantation finale, celle qui réalisée donne pleinement satisfaction.

Globalement, l'exploitation rationnelle de JEKERT impose les modes fonctionnels suivants :

- 1) **Consulter la sonde :** Configuration système / Données scientifiques. (*Consignes passives.*)
- 2) **Configurer les OPTIONS :** Rapide/Lent, Moteurs OFF/Actifs, VEILLE etc.
- 3) **Imposer une POSTURE.** (*Peut obliger à vérifier la configuration initiale.*)
- 4) **MOUVEMENTS :** Déplacements de base translations/Rotations.
- 5) **Mode EXPLOITATION des expériences scientifiques et des facultés d'apprentissage.**

Fort de cette répartition "opérationnelle", en résulte la liste des MENUS suivante, liste qui imposera sur le clavier une touche pour l'appel individuel à chaque fonction :

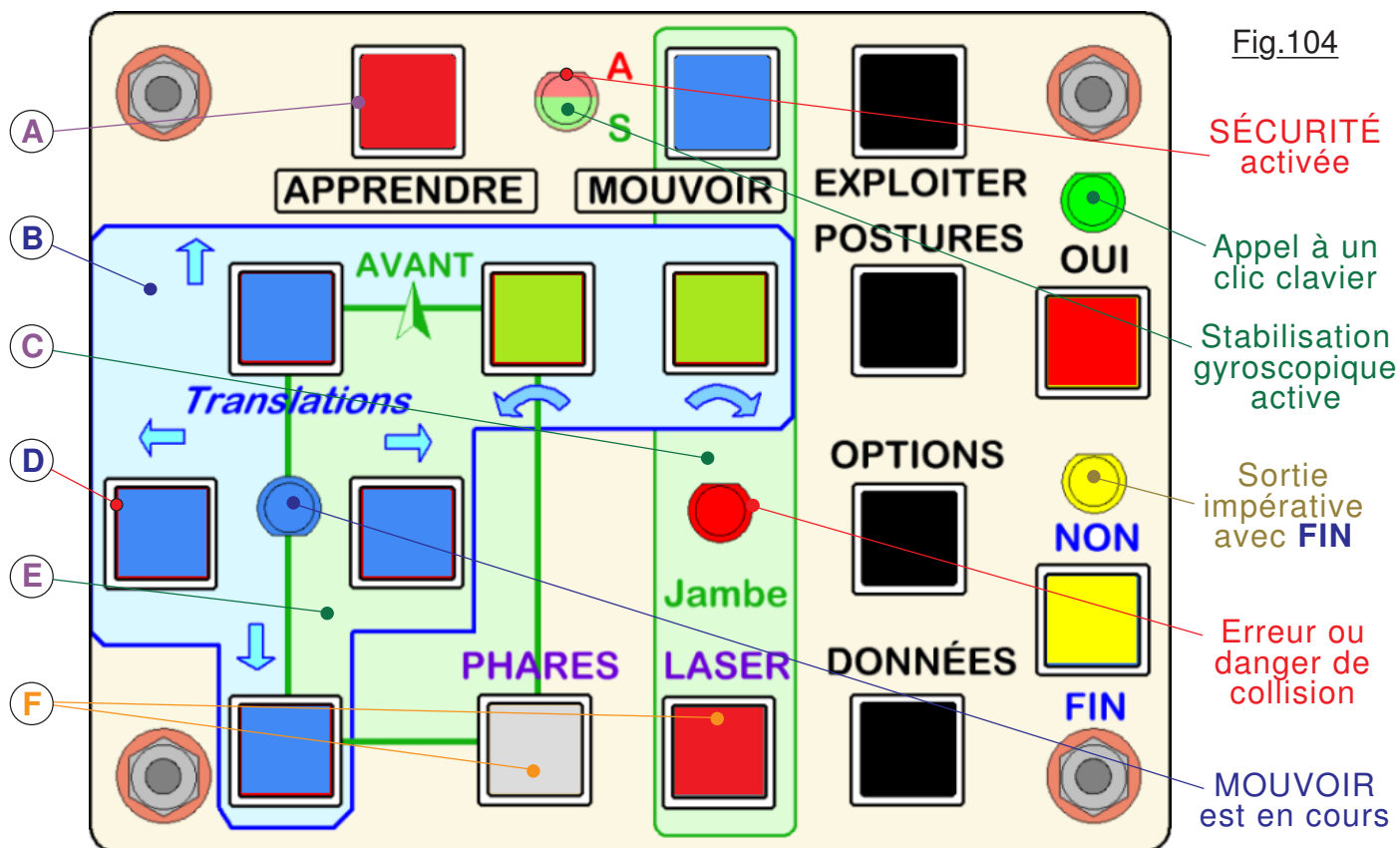
- EXPLOITER,
- POSTURES,
- OPTIONS,
- DONNEES,
- MOUVOIR,
- APPRENDRE.



Un menu étant activé par l'un des boutons poussoir dédié, c'est alors le codeur rotatif qui sera chargé de faire défiler les items le concernant sur le petit écran de maîtrise.

➤ Agencement du clavier.

Protocoles d'exploitation, répartition des touches du clavier et des divers autres éléments de maîtrise comme les témoins lumineux et les divers inverseurs et commutateurs sont intimement liés. Le problème étant résolu, passons directement à la présentation du clavier. Restant tributaires des positions des trous disponibles sur les plaques de circuit imprimés préperçées, les positions des divers boutons poussoir du clavier ne pourront pas se voir librement réparties. Les positionnements horizontaux et verticaux seront sur des alignements forcément "au dixième de pouce". Les études ont globalement abouti à ce qui ressemble à la solution présentée ci-dessous :



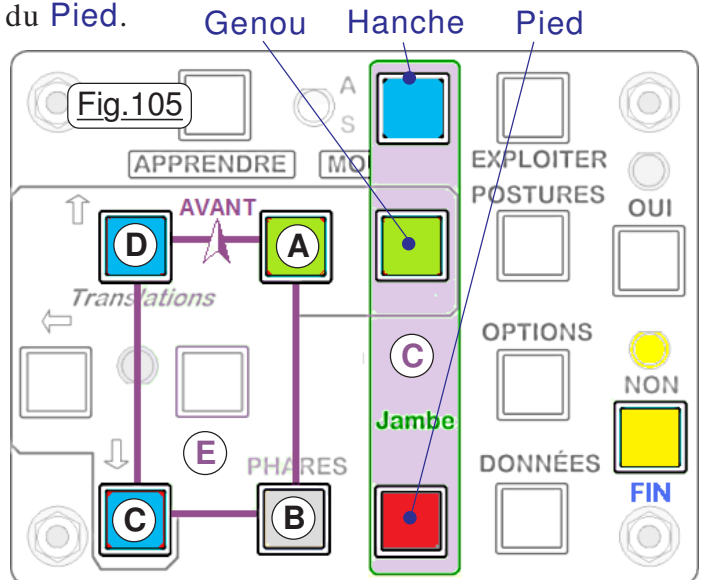
Pour faire court, nous allons utiliser des touches de couleur avec des choix qui faciliteront l'utilisation du clavier. Celles en bleu et vert sont dédiées aux mouvements du petit robot. Celle en rouge sont "potentiellement dangereuses" ou imposent une attention particulière de l'opérateur. (Par exemple répondre OUI à un item engendrera une action, la touche est rouge, alors que NON étant sans effet, on a utilisé une touche jaune.) Celles en noir sont les **quatre menus de base**. Cliquer sur l'une de ces touches ouvre le menu correspondant dont les items seront balayés par le codeur rotatif. La touche **APPRENDRE** en A présente exactement un effet analogue, mais sa couleur est rouge car les actions y sont potentiellement "dangereuses". Quand ce mode est actif la LED rouge du petit tableau qui porte la fenêtre de l'afficheur OLED s'allume en rouge. Vous avez certainement compris que si la stabilisation gyroscopique est active, la LED verte S s'éclaire. En réalité, cette LED est tricolore. Elle peut aussi s'allumer en bleu. Aussi, lorsque on entend un BIP sonore avec alerte sur erreur E12, la LED bleue s'allume, incitant l'opérateur à activer **SÉCURITÉ** sans avoir à consulter le manuel pour rechercher la nature de cette alerte. Toute action sur le pupitre éteint ce témoin lumineux. (Les nombreuses commandes sont détaillées plus avant, rassurez-vous. Ici on ne fait qu'explicitement "la répartition géographique" des touches.) **PHARES** et **LASER** en F ne sont fonctionnelles que dans certains menus. **LASER** est rouge car on ne doit l'allumer que s'il n'est pas dirigé vers une personne ou un animal quelconque. Quand on valide les déplacements par l'appel à **MOUVOIR**, la LED bleue **MOUVOIR est en cours** s'illumine et les touches de déplacement bleue telles que D et vertes de la zone B sont alors fonctionnelles. Au dessus de **OUI** la LED verte signale qu'une touche clavier est souhaitée. De même que la LED jaune au dessus de **NON** impose de cliquer sur **FIN** pour sortir de l'option en cours. (**FIN** et **NON** sont en réalité sur la même touche prévue pour deux actions sémantiquement différentes.) Vous avez forcément remarqué en

zone **C** la LED rouge qui s'illuminera un court instant sur une "erreur" lors des manipulations, et surtout lors d'un danger possible de collision entre les membres de la sonde et sa structure.

➤ Pilotage manuel des moteurs.

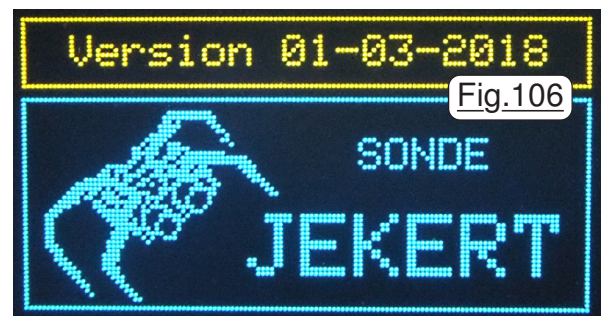
Quand on pilote un moteur en mode manuel, (*Développé en page 25 ...*) il devient impératif de savoir instinctivement sur quelle touche agir pour changer d'articulation. Dans ce but la sérigraphie comporte deux zones **C** et **E**. (*En version définitive, pour des raisons esthétiques ces zones sont en violet.*) Quand on est en mode pilotage manuel des servomoteurs, il faut oublier l'usage standard des diverses touches du clavier, seules celles coloriées sur la Fig.105 auront un effet. En particulier les deux boutons **F** n'agissent plus sur les phares et sur le LASER. La zone verticale **C** doit faire penser à une **Jambe** dont les boutons en partant du haut vers le bas représentent respectivement la **Hanche**, le **Genou** et le pivot du **Pied**.

La zone rectangulaire **E** (*En vert sur la Fig.104*) représente la sonde vue par dessus, les touches situées dans les angles symbolisant les **Jambes**. Dans cette symbolique, l'**AVANT** de la petite machine est indiquée par la flèche violette. Quand on désire changer de servomoteur, on désignera dans un ordre arbitraire l'articulation concernée, c'est à dire avec **A**, **B**, **C** ou **D** la **Jambe** sur laquelle on désire agir. Puis on indiquera à tout moment l'articulation à animer. Lorsque l'on a abouti à une configuration intéressante, on peut à tout moment sauvegarder les valeurs de la posture actuelle en EEPROM avec **Sauver la POSTURE ?** la commande dédiée si la configuration est séduisante ...



37) Les MENUS de base de la sonde JEKERT.

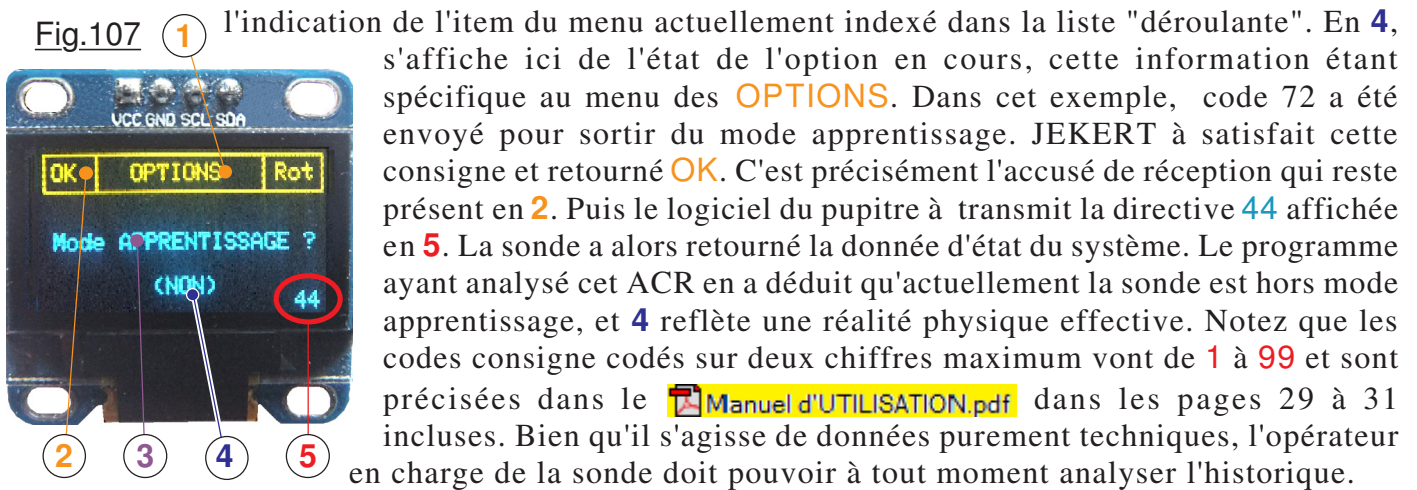
Intimement liés à l'agencement du clavier et des autres dispositifs du pupitre de pilotage, les divers MENUS participent intensément à la qualité opérationnelle de l'ensemble en exploitation de la sonde sur le terrain. Il importe de savoir que le codeur incrémental possède un bouton poussoir noté **BP_{CCR}** que l'on peut cliquer quand on appuie sur son gros bouton rotatif. Si vous consultez le livret **Manuel d'UTILISATION.pdf** en pages 16 à 18 vous constaterez qu'il permet dans plusieurs menus d'allumer la LED de **SÉCURITÉ**. À la mise sous tension du pupitre, l'écran d'accueil de la Fig.106 s'affiche et la LED verte du clavier clignote rapidement nous invitant à cliquer sur une touche pour passer à la suite. Dans le cadre jaune est précisée la version du logiciel qui fonctionne actuellement sur le pupitre. (*En réalité P40_PGM_MAÎTRE_RAQUETTE.ino est plus récent que cette copie d'écran et la date affichée sera 28-04-2018.*) Bien que l'on puisse cliquer sur n'importe quelle touche pour passer à la suite, une bonne technique consiste à utiliser la touche la plus proche, soit sur **OUI**.



➤ La répartition des informations sur l'écran OLED.

Avant d'aborder chaque MENU fonctionnel dans le détail, ouvrons un chapitre pour décrire la façon dont globalement les informations sont réparties sur le petit écran. Dans les stations de poursuite ou de gestion de satellites, (*Matérialisées ici par notre petit pupitre.*) les opérateurs ont des écrans pour toutes les informations descendantes, mais également vers celles qui sont envoyées aux lointains vaisseaux. La répartition des informations sur la surface utile est représentée sur la Fig.107 avec en haut à droite un petit cadre qui ne contiendra que **Rot** ou **CLV** précisant que dans la fonction active on doit choisir une action en tournant le codeur **Rotatif** ou sur l'une des touches du **CLavier**. En **1** on trouve le titre du menu actuellement exploré. Sur la petite zone située à gauche du cadre jaune en **2** seront affichés les accusés de réception. On observera en **3**

ATTENTION : Le pupitre et la sonde dialoguent en permanence. Le programme maître envoie une consigne. La sonde traite cette dernière puis accuse réception. (**ACR**) La raquette est en attente de cet ACR. Si la sonde n'est pas branchée, le programme du pupitre va se bloquer. Il importe donc que les programmes soient bien logés sur les ATmega 328 des deux entités pour se synchroniser et que nous puissions ainsi tester les divers comportements logiciels. (Voir dans le **tutoriel sur la réalisation du coffret** le résumé des actions à conduire pour figer dans les deux microcontrôleurs les logiciels et les données indispensables.)



➤ Le MENU des **DONNÉES**.

L'aptitude à interroger la petite machine sur son état système ou sur les valeurs mesurées par ses différents capteurs est fondamental en exploitation. Outre les données scientifiques mesurées, on doit pouvoir à tout moment avoir une vue précise de la configuration de la machine et de l'état de ses divers systèmes fonctionnels. L'agencement d'un menu ne doit pas être considéré à la légère. De sa convivialité dépend directement la qualité opérationnelle "du produit". Chaque fois que l'on valide la fonction **DONNEES** il importe d'ouvrir la page la plus "utile". Dans cette étude c'est celle qui correspond à la gestion des moteurs et de l'énergie. Ensuite, entrant dans ce menu, les items qui sont voisins doivent privilégier des pages "importantes en exploitation". Si on tourne dans le sens horaire on ouvre la page des mesures scientifiques. Une rotation dans le sens antihoraire ouvre les informations des périphériques consommant (*Virtuellement, car sur JEKERT ces périphériques ne sont absolument pas énergivores.*) beaucoup d'énergie, c'est à dire les Phares et le LASER. C'est donc celle des options et informations sur la stabilisation et l'APPRENTISSAGE qui se trouve la plus "éloignée en rotation" de la fenêtre d'entrée. Notez au passage que les affichages ralentissent considérablement le processeur, car le composant OLED étant graphique doit entièrement générer des dessins qui pour vous sont des caractères de texte. Il faut aussi effacer des zones ce qui impose de blanchir beaucoup de points. Aussi, quand vous tournez le codeur ne soyez pas trop rapides, laissez le temps à l'affichage de finir de tracer les informations concernées avant de tourner fébrilement d'un nouveau cran. Notez au passage que le symbole ☀ visible en bas de la Fig.109 et qui sera rencontré sur plusieurs écrans représente un Soleil. Il symbolise l'alimentation en énergie de la ligne qui peut être disjonctée et qui allume en particulier les phares, le LASER et les nombreuses LEDs situées sur la sonde. Ouvrant sur la page de la Fig.108, en 1 est précisé le bloc programme actuellement indexé en EEPROM. (Voir le menu des apprentissages et des options.) Si en 2 le booléen est **OUI** la sonde a été configurée avec **QUITTER**. Ne pas confondre avec ☀ sur OFF en 7 sur la page écran suivante Fig.109 qui traduit l'état actuel du **disjoncteur des énergies** "lumineuses". (LEDs / Phares / LASER.) En 3 est précisée la version du logiciel qui anime la sonde. En version ultime, **Page 17**

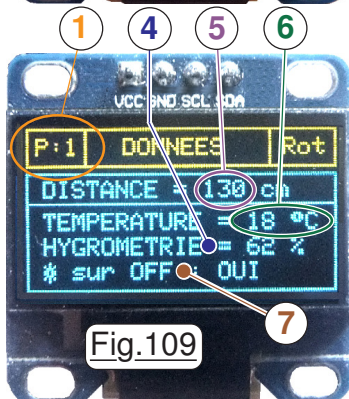
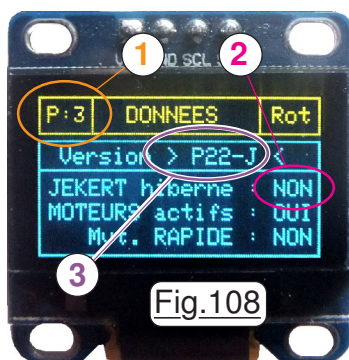


Fig.108

Fig.109

celle figée dans les microcontrôleurs est > P50 <, information qui sera affichée. En **5** est affichée la distance qui sépare JEKERT de l'obstacle le plus proche situé devant les capteurs ultrasonores. En **4** on trouve la valeur de l'hygrométrie relative. La température en **6** est celle de l'atmosphère ambiante. **NOTE** : Si c'est la page de la Fig.109 qui est présente à l'écran, le **BPccr** fait afficher la température interne du circuit MPU-6050 (Voir pages 19 à 22.) de la centrale gyroscopique. Dans ce cas le texte du paramètre est un peu changé car la précision de la donnée retournée au centième de °C.

Considérons la page écran de la Fig.110 sur laquelle en **A** est indiquée la tension d'alimentation des servomoteurs avec en **B** trois options système. Si l'énergie de puissance n'est pas branchée sur le cordon ombilical, les deux inverseurs étant coupés, la tension affichée sera de 1.47 v environ. (Ce "résidu" résulte de la présence du circuit de rechargement, ne pas s'en préoccuper il est normal.)



Fig.111

Fig.110

Le dernier écran réparti de façon différente ses données. Pour des facilités de lecture les quatre informations utiles sont regroupées par deux en **D** dans le rectangle du bas. Du coup la zone d'information annexe **C** étant libre, on y précise la plage de variations possibles pour la valeur des énergies que l'on peut



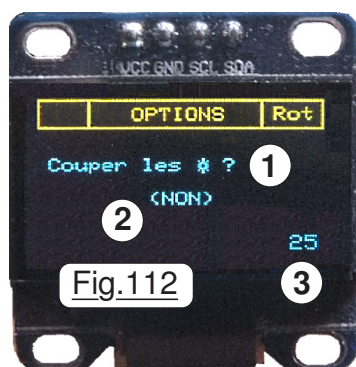
consigner aux deux périphériques à l'aide du gradateur. Il importe de noter qu'il aurait été possible de mémoriser les valeurs des booléens d'état dans la Raquette, évitant ainsi d'interroger la sonde ce qui diminuerait le nombre des codes et allègerait d'autant le programme objet de la sonde. Cette approche ne serait pas fiable, car on peut envoyer un ordre à la sonde, et que suite à un incident quelconque il ne soit pas exécuté. (Problème de transmission ...) **Il est donc impératif que ce soit la sonde qui fournisse les états effectifs des divers paramètres** et non ceux qui sont désirés.

➤ Quelques éclaircissements sur le logiciel.

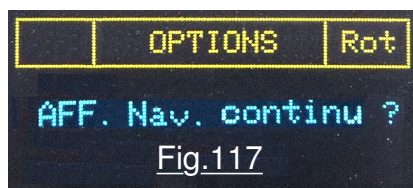
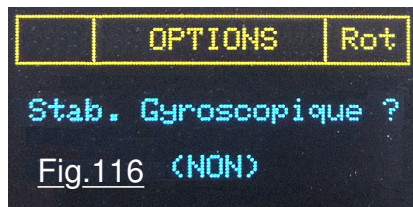
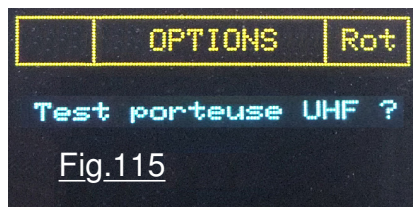
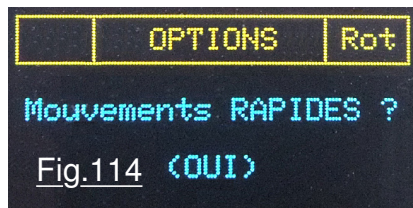
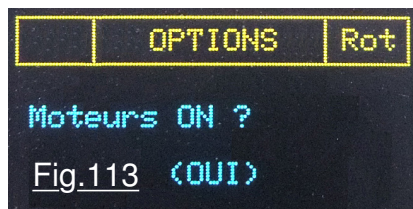
Contrairement à ce laisserait entendre le chapitre sur L'ordre des pages écran du menu des **DONNEES**, l'afficheur OLED n'est pas le seul à ralentir la vitesse d'exécution du programme. N'oubliez-pas que tourner le codeur rotatif ou cliquer sur **OUI** ou **NON** impose au programme de dialoguer avec le calculateur de bord. Par exemple **OUI** ou **NON** impose d'envoyer le code **Consigne** correspondant à l'item d'option affiché et d'attendre l'accusé de réception. Pour vous en rendre compte, il suffit lors de la rotation du codeur incrémental de valider les LEDs systèmes. Les deux LEDs blanches **TX** et **RX** s'illuminent à la cadence des impulsions envoyées sur la ligne de dialogue série. Visiblement quand on change de page écran une foule d'échanges se font entre les deux machines qui bavardent. Vous comprenez pourquoi la plus grande vitesse de transmission compatible avec une bonne fiabilité a été adoptée et pourquoi le rafraichissement écran présente un petit délai. Enfin, si vous tournez trop rapidement le codeur, sachez que traitées par interruptions deux requêtes sont prises en compte. C'est la raison pour laquelle quand on tourne trop rapidement, une deuxième page s'affiche, un peu "en retard".

➤ Le MENU des OPTIONS.

Concrètement, certaines sont vérifiables par deux possibilités. Par exemple l'allumage des Phares et du LASER sont observables visuellement et par interrogation de la sonde. Moteurs figés la LED rouge confirme l'information disponible avec le menu des **DONNEES**. Quand le programme est au point, le réalisme consiste à enlever le petit "strap" à languette pour éteindre toutes les LEDs système qui sont sur le circuit imprimé principal de la sonde, car supposée posée sur Mars elle n'est



plus visible et seule la télémétrie peut nous renseigner. Il faudra alors faire confiance aux télémétries, le réalisme est à ce prix. Beaucoup plus épurés que pour le menu des données, les écrans de base pour les **OPTIONS** ne présentent qu'un seul item à la fois. Dans le rectangle jaune, **Rot** précise que l'on changera d'option avec le codeur rotatif. Sur la Fig.112 est précisé en **1** l'option en cours de saisie et en **2** son état actuel. La logique veut que l'on clique sur la touche **OUI** ou sur le bouton poussoir **NON** si l'on désire inverser l'état de l'option. Le code de consigne est alors affiché en **3**. Puis la sonde inverse son état local et retourne l'ACR **OK**. En **2** l'information est alors mise à jour.



Tourner le codeur incrémental dans un sens ou dans l'autre fait changer d'item dans la liste des options. La permutation circulaire est naturellement fonction du sens de rotation. Les cinq items suivants du menu des options sont présentés par ordre croissant sur les Fig.113 à Fig.117 dont la qualité des images laisse parfois à désirer. (*Problèmes inhérents à la prise de vue avec du matériel inadapté.*)

Moteurs ON en Fig.113 se passe presque de commentaires. Avec l'option **(NON)** ils restent inertes alors que des consignes de mouvements sont envoyées. On peut ainsi effectuer des manipulations "à vide", en toute sécurité pour tester le logiciel etc. Si l'option **Mouvements RAPIDES** en Fig.114 est forcée à **(NON)**, les animations de type "coordonné" sont plus lentes ce qui permet à l'observateur de mieux comprendre la façon dont sont organisés certains *mouvements coordonnés* complexes. En Fig.116 on peut activer la stabilisation gyroscopique. Cliquer directement sur **OUI** déclenchera une erreur **E12** et la LED tricolore s'allumera en bleu comme déjà mentionné dans ce didacticiel. Cliquer sur le **BP_{CCR}** changera la couleur en rouge précisant que **Sécurité** est activée. Un nouveau clic sur **OUI** active la stabilisation gyroscopique et la LED tricolore s'illumine en vert pour en témoigner. ATTENTION : pour tester cette fonctionnalité alors que les moteurs sont actifs bien respecter le protocole de la page 11 de [Manuel d'UTILISATION.pdf](#). Écran de la Fig.115 affichée, cliquer sur **OUI** provoquera également une erreur **E12**. Donc, un petit clic sur le **BP_{CCR}** suivi de **OUI** transformera l'écran en celui d'un oscilloscope cathodique tels que ceux que l'on observe souvent dans les reportages scientifiques. Cette fonction est totalement virtuelle, uniquement pour "créer l'ambiance". Reportez-vous en page 27 du manuel pour en découvrir les subtilités.

38) L'OPTION AFF. Nav continu.

Dernière option dans la liste des fonctions étudiées, elle est mise à part, car sa "richesse fonctionnelle" est telle qu'il faudra un chapitre entier pour la décrire. Vous avez certainement remarqué que, comme pour l'option de la Fig.115 l'état n'est pas affiché à l'ouverture de la page écran. Cette information est inutile, car dans les deux cas valider l'option engendre un changement complet de l'affichage rendant inutile le **(OUI)**. Comme toute fonction qui pourrait engendrer des difficultés particulières, **SÉCURITÉ** est exigée. Valider l'option fait afficher sur l'écran OLED un véritable tableau de bord qui visualise "en temps réel" les paramètres d'attitude et de navigation de la sonde. La sortie de ce mode doit impérativement se faire avec **FIN**, donc la LED jaune clignote.

- **CAP** : Le cap sera mesuré par rapport au champ magnétique, (*Terrestre en l'occurrence !*) donc influencé par la déclinaison locale. Quelle que soit la façon dont sera affichée la valeur du CAP Magnétique, cette dernière est perturbée par les servomoteurs s'ils sont en service. La seule façon d'obtenir une valeur stable et fiable consiste à couper l'alimentation de puissance 6Vcc. **Le compas de route** sert à initialiser la **centrale gyroscopique**. Cette dernière enregistrera les moindres variations d'attitude et de LACET. Elle doit être initialisée pour pouvoir "naviguer géographiquement".
- **ATTITUDE** de JEKERT : Elle sera précisée par les angles de ROULIS et de TANGAGE. Ces inclinaisons sont mesurées par rapport au vecteur pesanteur local par la centrale gyroscopique qui intègre également trois accéléromètres trirectangles. (*Donc par rapport à un plan horizontal.*)
- **Référence gyroscopique** : Le cap indiqué sur le tableau de bord est issu d'une centrale gyroscopique MPU-6050 qui n'est plus influencée par le magnétisme, mais uniquement par les mouvements de JEKERT par rapport à l'Univers. On commence par désigner le CAP magnétique souhaité, ce qui permet d'orienter géographiquement la sonde. Puis on impose cette direction comme référence à la centrale gyroscopique. À partir de maintenant, tout changement d'orientation sera capté par le MPU-6050 et traduit numériquement et graphiquement sur le petit tableau de bord de JEKERT.


Donnée à visualiser	Valeur numérique	Symbole graphique
Cap magnétique actuel	OUI	OUI
Cap magnétique désiré	OUI	OUI
Référence gyroscopique	OUI	NON
Angle de ROULIS	OUI	OUI
Angle de TANGAGE	OUI	OUI
Ecart de route	OUI	NON

En rose des données liées.

➤ Utilité pratique de la référence Gyroscopique.

Puisque par nature la référence magnétique est plus facile à prendre en compte par le pilote de la sonde, pourquoi se préoccuper de cette référence interne qui s'oriente par rapport à l'Univers et ignore le lieu sur lequel on se trouve ? Plusieurs raisons techniques viennent plaider sa cause. Initialement, le Gyroscopie vient remplacer le compas magnétique quand ce dernier n'est plus crédible. Par exemple, quand sur un petit avion on engage un virage standard, on incline l'appareil en roulis. Si à cette inclinaison acquise relativement vite on ajoute de plus des turbulences, la boussole s'affole. Un navire, même de taille imposante, peut se faire chahuter considérablement en roulis et en tangage par une mer forte. (*Rien n'est grand par rapport à la mer.*) Le compas de route dans ces conditions se dandine avec frénésie. Dans ces exemples, le gyroscope reste de "marbre". C'est l'outil idéal pour assurer la navigation ... sauf qu'il dérive car la Terre tourne par rapport à sa référence universelle. C'est alors le compas de route, quand il n'est pas perturbé, qui sert pour le pilote à recalibrer le gyroscope de bord. Il est clair qu'avec l'avènement du GPS, surtout comparé au coût important d'un gyroscope mécanique, les conservateurs de cap "à l'ancienne" sont en voie de disparition ...

Reste que la centrale gyroscopique est bien plus précise que la **boussole**. Nous savons que **le compas de route** utilisé à bord de JEKERT **est corrigé**, le calcul assurant cette fonction conduit à une imprécision angulaire de plusieurs degrés. Enfin, notre petit robot est aussi un vaisseau spatial. Ces machines que l'on éjecte à des distances phénoménales se déplacent loin loin loin de tout astre. Pour les orienter en vue d'effectuer les corrections de trajectoire ou pour le freinage de capture gravitationnelle pour se mettre en orbite, elles n'ont que les étoiles pour faire le point. La centrale gyroscopique est alors la technologie incontournable pour assurer la navigation et ce d'autant plus qu'il n'y a plus présence de champs magnétiques pour orienter une boussole. Par défaut, sur un RESET de l'ATmega328, c'est le Cap Magnétique qui est pris en référence, avec une route souhaitée plein Nord, c'est à dire au CAP magnétique 0°.

 **Pour comprendre les différentes informations qui seront affichées sur le tableau de bord, il importe d'avoir une idée minimale de ce que l'on nomme "la dérive gyroscopique". Un chapitre à part est logé dans l'encadré de la page 22 et je vous invite fortement à le parcourir.**

➤ Présentation du tableau de bord de JEKERT.

Optimiser l'affichage sur le minuscule écran impose de répartir judicieusement un maximum d'informations sans toutefois nuire à la lisibilité. Pour satisfaire ce critère, quatre options seront possibles. On change la nature de certaines informations en cliquant sur le **BPccr**. La valeur désirée pour le CAP à suivre sera initialisée dans le menu **EXPLOITER**, le protocole étant donné en page 23 du manuel d'utilisation de la sonde. La page d'ouverture du mode de visualisation du tableau de bord est représenté sur la Fig.118 avec en

4 la valeur du **Cap Magnétique** actuel quel que soit le mode d'affichage adopté. En **1** et **2** on trouve en permanence l'inclinaison en ROULIS, de même qu'en **3** et **6** celle du TANGAGE. En **5** le petit cercle précise la direction de la Route désirée. Ce "Bug" est ajouté sur la rose des caps uniquement en mode d'affichage **M**.

7 : Route magnétique actuelle si option **M**. (*Le Nord est en haut.*)

9 : Route Magnétique désirée.

8 : Ecart de route entre le Cap actuel et la référence.

Dans cet exemple on désire un cap plein Nord : Le "Bug" est en haut et $M = 0^\circ$. Le vaisseau est orienté au 180° . L'écart de route **E** est de -180° , des explications d'interprétation s'imposent. (*Voir paragraphe plus avant.*)

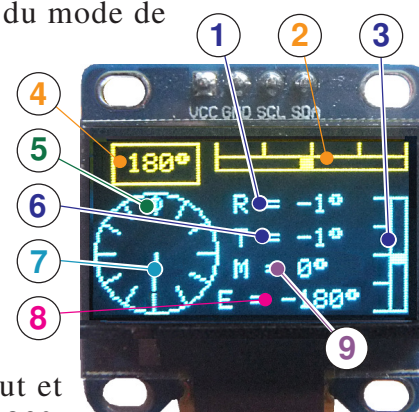


Fig.118

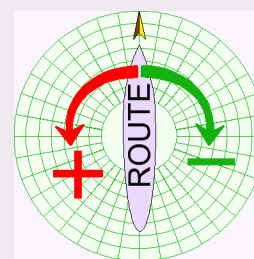
Cliquer sur le **BPCCR** fait passer à l'option suivante. La lettre **M** est remplacée par **D** qui symbolise la route **Désirée**. (*L'AVANT est en haut.*) L'aiguille pointe le **Gisement** de la cible. Le **Gisement** est l'angle mesuré entre 0° et $\pm 180^\circ$ compris entre l'avant du vaisseau et la cible. Si le Cap désiré est à droite la valeur est positive, s'il est négatif la direction souhaitée est à gauche. *Ce type de visualisation est commode, car il montre de quel coté se trouve la cible par rapport à l'axe longitudinal de la sonde.* Il suffit pour le navigateur de faire tourner sa machine (*Satellite, avion, navire, mobile quelconque ...*) vers l'aiguille pour corriger la route. L'aiguille tourne alors vers le haut. Quand elle est verticale le Cap suivi est correct et il faut stopper l'évolution en LACET.

Nouveau clic sur le **BPCCR** et l'affichage passe en référence gyroscopique. La lettre **D** se change en **G** et la valeur ne présente plus de signification pour le navigateur, raison pour laquelle elle est remplacée par "////". L'écart de route devient égal à zéro car la direction actuelle dans l'Univers est prise en référence. L'aiguille pointe sur la rose des Caps la direction souhaitée. Grosse différence par rapport à la référence **Magnétique**, à partir de la validation de ce mode c'est une référence gyroscopique interne au MPU-6050 qui est prise en compte.

Dernière option obtenue avec le **BPCCR** : Le mode **C**. L'effet est strictement identique à celui de l'entrée dans le mode **G** sauf qu'à partir d'ici la dérive gyroscopique mentionnée en page 22 est Compensée automatiquement et en permanence par le calculateur de bord. L'opérateur n'a plus à recalculer la centrale toutes les cinq minutes environ. Logiquement, à un ou deux degrés près, l'écart **E** ne doit plus évoluer si on laisse la sonde immobile sur une longue période.

RÉSUMÉ : En référence Magnétique **E** représente l'écart de route entre le Cap actuel et la valeur imposée avec **CAP magnétique**. (*Pour le signe voir la Fig.119*) Si **C** = //// ou **G** = //// c'est la rotation de la sonde par rapport à l'Univers entre l'instant où le **BPCCR** valide l'option et l'instant de l'affichage des données. En mode Directionnel, (*Le plus naturel à interpréter*) la flèche indique le gisement de l'orientation désirée, l'avant de la sonde étant situé en haut de la rose des caps. **Il faut orienter le nez du vaisseau vers la flèche pour corriger la route.** (*"Courir après l'aiguille."*)

Fig.119



➤ Interprétation du ROULIS et du TANGAGE.

Observant la Fig.119 on peut se demander pourquoi le sens positif ne serait pas celui du sens horaire. Fondamentalement, la centrale gyroscopique MPU-6050 calcule en tenant compte de ce que les spécialistes nomment **"les Angles d'Euler" qui orientent les axes avec le sens trigonométrique**, du coup c'est celui qui est adopté pour le LACET, le ROULIS et le TANGAGE.

Lorsque l'on pilote un mobile, sauf cas très particulier, on regarde de l'arrière vers l'avant. (*C'est tout de même mieux pour observer la route !*) La Fig.120 nous place dans ces conditions. Notre regard va de l'arrière de la sonde vers l'avant. Si elle s'incline à gauche, la rotation sera considérée comme positive puisqu'elle se fait dans le sens trigonométrique direct. De l'autre côté, l'inclinaison sera affectée du signe moins. Sur la Fig.121 JEKERT est vue depuis bâbord, elle "pique du nez" et les signes à prendre en compte y sont précisés. En **3** et **6** de la Fig.118

les inclinaisons sont indiquées sous forme numérique dans la plage $\pm 180^\circ$ alors que sur les échelles linéaires **2** et **3** la pleine déviation est obtenue pour $\pm 10^\circ$ avec effet de butées. De loin le plus simple consiste à se dire que le ruban analogique se décale vers le coté où la machine s'incline. Si la bande lumineuse est tracée vers le bas, la sonde est en train de piquer. Vers le haut, c'est que JEKERT est cabrée. Pour le ROULIS on penche vers le ruban.

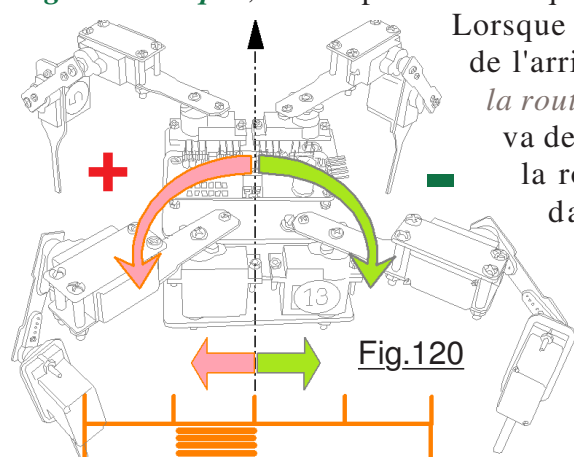


Fig.120

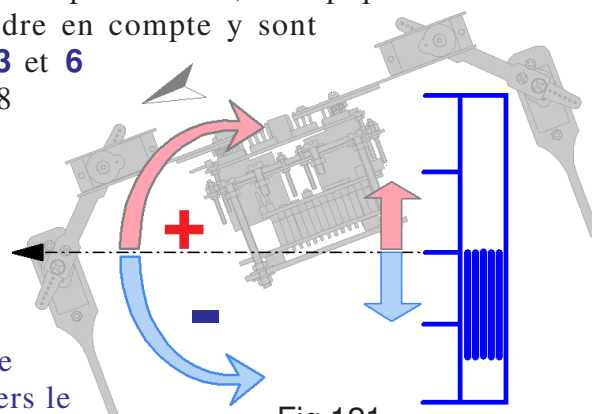


Fig.121

39) La dérive gyroscopique.

Toutes celles et tous ceux qui ont eu le privilège de piloter un avion muni d'un conservateur de cap gyroscopique connaissent ce phénomène. De quoi s'agit-il ?

Si vous utilisez une boussole magnétique, cette dernière s'oriente par rapport au champ magnétique terrestre. Posez cette dernière sur une table. Elle indique une direction. Revenez plusieurs heures ou plusieurs jours plus tard. Si elle n'a pas été déplacée, elle indique toujours la même direction.

- Normal direz-vous, elle s'oriente par rapport aux lignes du champ magnétique local qui est fixe par rapport au géoïde. Si la boussole n'est pas réorientée, elle indiquera immuablement un CAP magnétique invariable.

Laissez JEKERT bien tranquille, pénarde en hibernation par exemple. À bord les systèmes de navigation, les dispositifs expérimentaux restent opérationnels. Par les télémesures on continue à recevoir les valeurs météorologiques, la valeur du CAP magnétique ainsi que les données gyroscopiques de la centrale inertielle. Tangage et Roulis ne varient pas au cours du temps. Mais la valeur **Ecart route** évolue en permanence dans le sens négatif, comme si on tournait en LACET. Cette variation augmente régulièrement d'environ 15° par heure. Curieux non ?

L'explication réside dans la nature même des capteurs gyroscopiques. Par conception ils conservent "une orientation par rapport à l'Univers", ou si vous préférez par rapport aux belles étoiles qui illuminent merveilleusement nos cieux nocturnes.

-Et alors ?

-Ben ... certains affirment avec conviction que la Terre tourne.

Elle entraîne dans ce tourbillon circadien tout ce qui est immobile par rapport au sol. Du coup, bien qu'immobile sur le bureau, par rapport aux étoiles JEKERT effectue un tour en 24H, soit 15° par heure. C'est cette rotation à laquelle nous ne sommes pas sensibles qui inexorablement est enregistrée par les dispositifs inertiels gyroscopiques.

CONCLUSION : Contrairement au GPS qui calcule votre déplacement par enregistrement des points successifs occupés sur votre trajectoire, un conservateur de CAP gyroscopique sera inexorablement influencé par la rotation de l'astre sur lequel on se trouve *et que l'on désire prendre en référence pour indiquer une direction*. C'est la raison pour laquelle sur les petits aéronefs le "GYRO" est toujours complété par un compas magnétique qui permet de recalibrer régulièrement le conservateur de CAP pour tenir compte de ce phénomène. Le premier qui a mis ce phénomène en évidence est Foucaud, savant illustre et présent dans tous les ouvrages de physique par son expérience célèbre sur le pendule qui porte son nom.

La centrale inertielle MPU-6050 intégrée à bord de la petite sonde est réellement un dispositif gyroscopique. Il est donc influencé par la rotation de la Terre. Avec 360° en 24 heures, on doit s'attendre à une dérive gyroscopique de 15° par heure ce qui n'a vraiment rien de négligeable. N'oubliez-donc jamais ce phénomène. Il vous arrivera de faire des manipulations qui peuvent durer facilement trente minutes à une heure. Le CAP magnétique continuera à indiquer une direction crédible. En revanche, l'écart de route pourra vous interpellier. Par exemple vous remplacez la Sonde plein Nord, direction pour laquelle vous aviez procédé au calage gyroscopique. Vous vous attendez à un **Ecart route** nul puisque l'orientation de calibrage est retrouvée.

- Mazette, la télémesure indique -43°, vas pas bien l'truc !

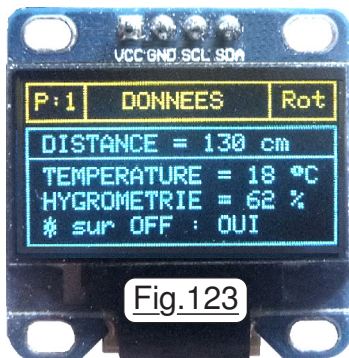
- Ben si Dudule, fonctionne bien la centrale, mais pendant que t'as siroté ton Pastis la Terre a tourné. Le GYRO c'est comme l'heure, son "aiguille" tourne en permanence.

La course du Soleil dans notre ciel correspond exactement à la réciproque. Ce n'est pas lui qui se déplace, mais la Terre qui change d'orientation par rapport à l'Univers. Comme le Soleil fait partie intégrante de l'Univers, on le voit tourner, comme pour toutes les autres étoiles. Il devient possible de déterminer l'heure en observant sa position, (*Les cadrans solaires pour ne pas les nommer ...*) De façon analogue, il serait possible d'utiliser le module MPU-6050 comme chronomètre en utilisant l'évolution du Lacet. Cette possibilité impliquerait on s'en doute de maintenir JEKERT strictement immobile durant le chronométrage. C'est une hypothèse amusante pour illustrer le propos, mais scientifiquement ajouter un chronomètre informatique s'imposerait.

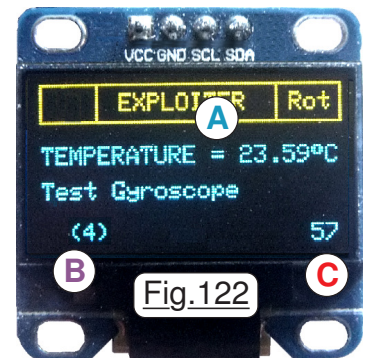
voir <https://www.lavionnaire.fr/InstVolDirect.php>

40) Le menu EXPLOITER.

C'est forcément le plus copieux, il suffit d'ouvrir [Manuel d'UTILISATION.pdf](#) en son centre pour immédiatement le comparer aux autres. Il est conçu en supposant que l'on vient de mettre sous tension le pupitre engendrant un RESET sur les deux microcontrôleurs. Sous cette hypothèse la sonde est en **mode hibernation**. Tous ses systèmes sont en sommeil. Il est donc assez logique que la première action consistera à **Réveiller JEKERT**. (Voir l'encadré en haut de la page 24.) Cliquer sur **OUI** transmet immédiatement le code **55**, mais il faut un petit délai pour que JEKERT retourne le **OK**, preuve qu'à bord plusieurs actions sont déclenchées. On tourne le codeur incrémental dans le sens horaire. Comme le protocole de la page 3 du manuel n'est pas respecté, il faut revenir "en arrière" d'un pas pour avoir **Test Gyroscope**. On pourrait se passer de cette procédure un peu "laborieuse", mais pour le réalisme l'opérateur doit toujours engager une réinitialisation de la centrale gyroscopique MPU-6050. Le pupitre transmet le code **58** puis un dialogue s'instaure entre les deux machines. La Fig.122 présente la procédure qui en **A** précise la température du circuit électronique interne au MPU-6050. On peut remarquer la précision avec laquelle la sonde retourne cette information. On constate sur la Fig.123 que la température extérieure mesurée par le petit capteur météorologique est nettement inférieure. Pour s'assurer de l'initialisation correcte de la centrale gyroscopique, la procédure consiste à déclencher un compteur

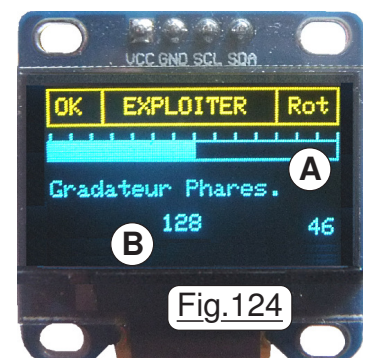


B. Toutes les secondes on interroge la sonde sur la valeur de sa référence gyroscopique par la commande **C**. Quand le compteur **B** est initialisé à 20, on mémorise la valeur retournée par **C**. Chaque fois que la valeur retournée toutes



les secondes est identiques à celle mémorisée, on décrémente le compteur **B**. Si la valeur change, on la mémorise à nouveau et l'on recommence le décomptage à vingt. Quand la centrale gyroscopique est en régime stabilisé, le compteur va diminuer lentement sans être recyclé à son maximum. Si vingt mesures prouvent la stabilité, la procédure s'achève et rend la main au pupitre en retournant l'accusé de réception attendu, c'est à dire **OK**.

A ce stade, si vous consultez l'état des phares et du LASER dans le menu des **DONNEES**, vous allez constater qu'ils sont au niveau minimal de 1, valeur la plus faible qui est imposée en mode hibernation. Du reste, quand on ouvre le menu **EXPLOITER**, un pas dans le sens antihoraire propose la fonction **QUITTER**. C'est elle qui permet en exploitation de faire passer JEKERT en hibernation, configuration qui la place en veille quand elle va devoir rester durant une longue période en sommeil. (Par exemple quand la station météorologique prédit l'arrivée d'une tempête de sable.) Aussi, ajuster les phares et le LASER à des niveaux notables avec le gradateur s'avère un peu laborieux. (14 pas sont nécessaires pour couvrir l'amplitude totale de la variation.) C'est la raison pour laquelle le troisième item du menu **Phare & LASER = 128** force l'initialisation des deux niveaux à la valeur moyenne de 128. Un pas de plus avec le codeur rotatif et l'on trouve l'item **Gradateur Phares**. Valider avec **OUI**



ouvre la fenêtre de la Fig.124 qui visualise l'état actuel en numérique en **B** et sur une échelle linéaire en **A**. La pleine déviation du ruban couvre la plage entre **1** et **254**. En tournant le gros bouton on augmente ou l'on diminue le niveau d'une graduation sur l'échelle linéaire soit un incrément ou un décrétement numérique de dix huit unités. Si on insiste à tourner le bouton quand la valeur est maximale ou minimale, une butée virtuelle fige la valeur, une alerte sonore est générée, associée à **E1** ou **E2**. La sortie de la fonction doit impérativement se faire avec **FIN** pour effacer correctement l'écran.



NOTE IMPORTANTE : Arrivera forcément un moment où l'opérateur engagera une mauvaise manipulation qui "brouillera" les affichages ou bloquera la synchronisation dans les dialogues. Les divers cas sont répertoriés en page 5 du manuel et surtout les procédures pour se sortir des situations critiques.



QUITTER réalise les actions suivantes : (*Actions listées en page 28 du manuel.*)

- Active les moteurs si ils sont sur OFF. (*Pour pouvoir "poser" la sonde au sol.*)
- Force **Rapide** à "faux" puis place la Sonde au sol sur son bouclier protecteur.
- Force les moteurs sur OFF, et la puissance à 1 sur les phares et sur le LASER.
- Coupe le disjoncteur des énergies, ainsi que les phares et le LASER.
- Désactive le mode APPRENTISSAGE si actif et éteint la LED rouge sur la sonde.
- La "cible" du codeur incrémental devient "personne".
- Désactive le mode TORSION manuelle ainsi que la stabilisation gyroscopique.
- L'option **Affichage en continu** est forcée à "faux".

Reveiller JEKERT n'est pas la réciproque et ne réalise que les actions suivantes :

- Réarme le disjoncteur pour rétablir les énergies.
- Réactive la motorisation sur ON.
- Impose la configuration **Stable_Transversal** et **Libère les efforts**.

Encore une rotation d'un pas dans le sens horaire, et c'est le **Gradateur LASER** de la Fig.125 qui est proposé, l'écran étant strictement analogue à celui des phares quand on valide l'option. ATTENTION : Les phares et le LASER sont coupés par défaut quand on passe en hibernation. Pour observer visuellement l'effet du gradateur il importe de les activer avec leurs boutons poussoir respectifs qui sont valides durant le menu **EXPLOITER**. *Notez au passage que si vous cliquez sur une touche non valide dans un menu particulier, un BIP sonore vous avertit de l'erreur, et la DEL Erreur s'illumine durant la tonalité du BIP.* Ce comportement se retrouve globalement dans tous les menus disponibles.



➤ Utilisation du LASER.



Première fonction rencontrée dans le menu **EXPLOITER** qui avertit que **la sortie devra IMPÉRATIVEMENT se faire avec la touche FIN.** (*Voir la Fig.126*) Quand la rotation du codeur incrémental visualise la commande **Utiliser le LASER ?** un BIP sonore attire l'attention de l'opérateur sur l'information **➤ Retour par FIN.** En effet, certaines fonctions complexes imposent quand on les quitte à replacer la sonde dans une configuration propre, généralement en posture **Stable Transversal**. Ce n'est qu'un avertissement, un rappel "fort" pour l'opérateur. Cliquer sur **OUI** déclenche le processus qui impose un

protocole rigoureux listé en page 6 du **Manuel d'UTILISATION.pdf**, car il y a risque potentiel pour la sonde et pour les personnes présentes dans le local. (*Pour les personnes c'est assez exagéré, mais la rigueur fait partie intégrante de l'astronautique ...*) La LED jaune clignote : Rappel qu'il faut terminer par **FIN**. Surtout la LED rouge nommée **Risque** sur le pupitre en fait autant. Elle signale que nous sommes en pilotage manuel des servomoteurs et que l'on risque la divergence des asservissements si on oriente les articulations hors limites des moteurs SG90. Si vous consultez la page 21 du **DOSSIER TECHNIQUE.pdf** vous y trouverez les angles d'orientation à ne pas dépasser. Si l'énergie de puissance est distribuée et que la motorisation est active, la sonde adopte une configuration particulière pour se sustenter sur trois **Jambes**, celle supportant le LASER servant à le diriger à convenance. Notez au passage que cliquer sur la touche **PHARES** durant ce mode déclenche un Tir LASER dont il est question plus avant en page 27. L'idée directrice de cette fonction consiste à pointer le LASER sur une roche, puis d'effectuer un tir pour simuler le spectromètre de masse intégré dans la sonde Curiosity. Cliquer sur **FIN** restitue la posture "standard" **Stable Transversal**.

➤ Le spectre colorimétrique en mode graphique.

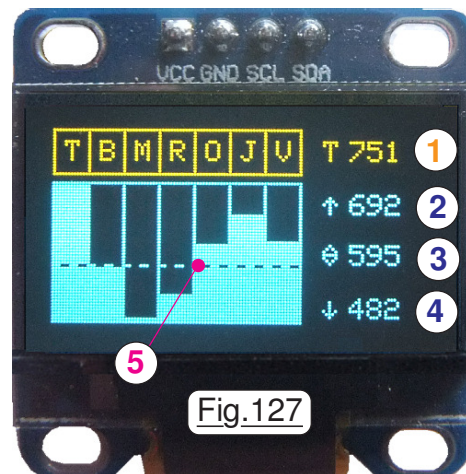
Évolution particulièrement agréable, le pupitre grâce à son afficheur graphique autorise la présentation des données du luxmètre sous forme graphique. Quand on fait appel à la fonction **Aff spectre couleur ?** il importe de savoir que c'est un enregistrement figé en mémoire EEPROM non volatile qui est représenté, et non les caractéristiques de l'atmosphère en temps réel. Ce choix est très avantageux, car il permet de visualiser en différé et à convenance un spectre

préalablement enregistré. Dans la liste du menu **EXPLOITER** on rencontre en premier la fonction **SAV spectre couleur ?** qui exige que **SÉCURITÉ** soit armée. Cette obligation prévient un "écrasement" involontaire des données actuellement inscrites en EEPROM. Si on valide la fonction en respectant le protocole donné en haut de la page 21 du **Manuel d'UTILISATION.pdf**, un processus analogue à celui de l'utilisation du LASER va se déclencher. La sonde va être configurée en posture stable sur trois **Jambes**. Puis le filtre coloré va se voir déplacé au dessus de la cellule photorésistante et les valeurs enregistrées. Le programme replace la sonde en posture **Stable Transversal** puis affiche les données spectrales. Il faut ressortir de la fonction par **FIN** pour "nettoyer" l'écran.

Un pas de plus avec le codeur rotatif et c'est l'option **AFF spectre couleur ?** qui est proposée. Cliquer sur **OUI** engendre l'affichage montré sur la Fig.127 qui présente une copie d'écran de la page écran dédiée. Chaque barre verticale présente une hauteur proportionnelle à son pourcentage énergétique. Les lettres dans la zone jaune précisent la couleur :

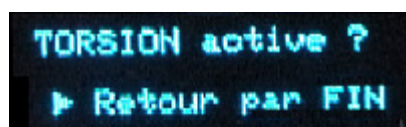
T : Luminosité Totale **B** : Bleu **M** : Mauve
R : Rouge **O** : Orange **J** : Jaune **V** : Vert

La barre pour **T** occupera systématiquement l'intégralité de la hauteur du cadre bleu pour "dilater" au maximum l'amplitude verticale du graphe. En **1** est indiquée la valeur totale mesurée sans la filtration. Elle correspond à la hauteur du cadre bleu. En **2** comme l'indique le symbole \uparrow est précisée la valeur pour la couleur la plus énergétique. Dans cet exemple c'est le jaune. En **4** nous trouvons la valeur numérique de la couleur la moins énergétique symbolisée par \downarrow . La ligne pointillée **5** représente la valeur moyenne de toutes les couleurs. ATTENTION, ce n'est pas la valeur moyenne entre le maximum et le minimum (*Qui ici vaudrait $(692 + 482) / 2 = 587$.*) mais bien la moyenne des six valeurs énergétiques du spectre. Dans l'exemple présenté sur la Fig.127 elle vaut : $\text{moyenne} = (594 + 482 + 531 + 632 + 692 + 642) / 6 = 3573 / 6 = 595,5$. Il est donc normal que la ligne pointillée **5** ne soit pas exactement à mi-hauteur entre la barre représentative de la couleur **Mauve** et celle proportionnelle à l'énergie du **Jaune**.



➤ Le mouvement de torsion.

Lorsque l'on utilise les mouvements de base pour déplacer la sonde sur le sol martien, les amplitudes sont figées. En particulier il n'est pas possible avec les rotations en LACET d'orienter avec précision JEKERT. On peut avoir envie d'une orientation fine de la plateforme instrumentale. Par exemple orienter finement le télémètre à ultrasons vers un obstacle situé vers l'avant mais un peu sur les cotés et non exactement dans l'axe longitudinal. L'option **TORSION active ?** est prévue dans ce but. Elle est du type **Retour par FIN**, car en sortie de cette fonction il faut ramener la structure de JEKERT dans "l'axe morphologique" du petit insecte mécanique.



Le codeur incrémental agit alors pour engendrer des déviations en LACET de petites amplitudes à notre guise vers bâbord ou tribord. Le protocole est précisé en page 7 du manuel. Notez que la LED rouge **Risque** clignote car nous sommes dans un mode de pilotage manuel des servomoteurs. Si on les amènes hors limites, les automatismes peuvent diverger.

➤ Pilotage manuel et individuel des moteurs.

Cette fonction est conçue pour créer des postures particulières pouvant être mémorisée dans la mémoire non volatile EEPROM du microcontrôleur. Le nombre de formes possibles n'est limité que par notre imagination, nous avons toute liberté d'en agencer des très originales. Il est alors possible de les sauvegarder en EEPROM, huit emplacements leurs sont réservés. Éventuellement cette fonction permettrait de sortir la sonde d'une situation critique, par exemple un atterrissage sur un rocher qui la place dans une posture à la limite d'un basculement latéral. Les techniciens au sol définiraient alors une stratégie pour la dégager, en agissant moteur après moteur. Pour le plaisir, vous pourrez simuler une telle situation et trouver une manœuvre de sauvegarde. Si elle échoue et que JEKERT bascule dans une posture sans issue ... fin de la mission ! La **SÉCURITÉ** n'est pas exigée pour ouvrir cette fonction, car la valider ne provoque aucun mouvement sur la sonde.

En revanche, il faudra sortir par **FIN** car il faut purger l'écran de toutes les informations spécifiques.



La stabilisation gyroscopique incompatible est coupée automatiquement quand on ouvre ce mode.

L'apprentissage s'il est actif est également interrompu quand on construit une posture particulière.



NOTE : Pour des raisons de sécurité des personnes présentes dans le local, le LASER sera automatiquement coupé pour ces manipulations. En toute logique, construire une posture n'a rien à voir avec les fonctions de pointage et de tir LASER. Il sera toujours possible par contre d'allumer ce dernier lorsque l'empreinte sauvegardée en EEPROM aura été invoquée pour configurer la posture de la machine.

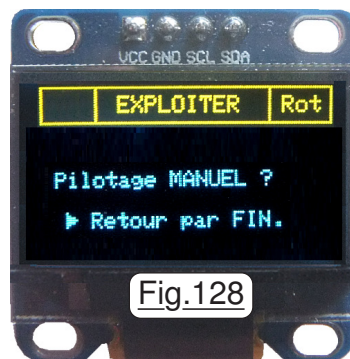


Fig.128

Quand on valide l'option de la Fig.128 l'écran de la Fig.129 s'affiche. La LED jaune clignote, mais on s'en doutait un peu. Comme pour tout pilotage manuel, il est possible d'amener les déviations



Fig.129

angulaires des servomoteurs hors limite avec à la clef la divergence des automatismes. (Le moteur part en rotation continue jusqu'à ce que la mécanique se bloque sur la structure de la sonde. Il faut immédiatement couper les énergies, car le servomoteur va surchauffer et se détruire si on le laisse trop longtemps dans cette situation.) **L'avantage considérable d'un potentiomètre numérique, c'est de pouvoir l'initialiser à une valeur quelconque.** Donc, chaque fois que l'on change de servomoteur, le programme embarqué va chercher sa valeur actuelle de consigne et la recopie dans la variable gérée par le codeur incrémental. L'ensemble des moteurs reste alors totalement inerte, et seule une action sur le codeur incrémental engendrera des mouvements. En 4 de la Fig.129 est indiqué le dernier code envoyé par la raquette avec en 1 l'accusé de réception. En

2 sera indiqué la **Jambe** sur laquelle on agit actuellement, et en 3 quel servomoteur est sollicité. Pour des raisons d'agrément, dans ce mode le **BPCCR** permet d'alterner entre des incréments angulaires importants ou faibles. Il est ainsi facile à tout moment d'adapter la finesse avec laquelle on engendre les mouvements individuels. Pour mémoire, la disposition des touches du clavier favorisant ces manipulations est décrite sur la Fig.105 de la page 16. La page 8 du [Manuel d'UTILISATION.pdf](#) propose le protocole précis à respecter pour piloter en manuel les moteurs, et la page 9 pour sauvegarder en mémoire non volatile EEPROM la configuration particulière construite .

➤ Affichage des consignes d'une posture.

Particulièrement utile pour effectuer des analyses diverses, on peut par les deux items du menu **EXPLOITER** qui se suivent quand on tourne le codeur incrémental dans le sens horaire, se faire afficher à convenance les douze consignes de motorisation correspondant soit à la configuration actuelle, soit à l'une des huit postures sauvegardées en EEPROM. La première s'obtient avec la commande **X** de la Fig.130 alors que la posture logée en EEPROM sera listée par l'item **Y**. Vous remarquerez qu'en **Y** est précisé le **Numéro** de la posture EEPROM actuellement indexée, et que naturellement on peut choisir à notre convenance. (Voir plus avant comment procéder.) Notez au

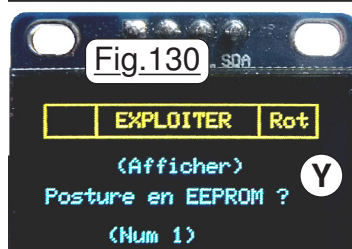


Fig.130

passage le texte **(Afficher)** qui aide à ne pas confondre cette fonction avec celle qui permet d'imposer à JEKERT une posture mémorisée sur la machine. Quand on valide avec la touche **OUI** on obtient un écran comme celui représenté sur la Fig.131 et la LED jaune se met à clignoter annonçant qu'il sera important de sortir par usage de la touche **FIN**. (Cette touche aura pour effet d'effacer correctement l'écran et de ne pas laisser des résidus, ce qui se produirait si on tourne le codeur incrémental sans avoir quitté la page écran des consignes moteur. Si vous passez outre, pour retrouver un écran "propre" il suffit par exemple d'invoquer un menu quelconque qui rafraîchi systématiquement entièrement l'écran.) La copie d'écran montre que les valeurs des consignes sont organisées sous forme d'une matrice dont les lignes sont concernées par les articulations et les colonnes sont représentatives des

Jambes de l'insecte mécanique. Pour information, quand on désire que l'un des servomoteurs adopte une orientation particulière, on transmet au circuit électronique qui le pilote une valeur qui représente l'angle la position que l'on désire imposer à ce dernier. Cette valeur correspond à un consigne que pour des raisons purement informatique on délivre sous forme d'un nombre dont la valeur sera globalement comprise entre 145 et 500. Peu importe la technologie et les raisons qui aboutissent à ces valeurs. Toutefois, l'utilisateur de JEKERT désirera parfois connaître ces valeurs, par exemple pour les comparer à celles qui correspondent par exemple à celles de la posture actuelle. Avouons que c'est surtout le programmeur qui sera concerné. Reste que : Qui peut le plus assurera le moins. Libre à vous d'ignorer cette fonction.

Fig.131

Hanche
Genou
Pied.

UCC GND SCL SDA			
	EXPLOITER		Rot
239	344	222	346
315	296	298	291
421	192	426	199
A	B	C	D
Jambe			

► Tir LASER.

Expérience scientifique qui a été mise au point par nos chercheurs et nos techniciens toulousains, cette technologie (*Cocorico !*) a été sélectionnée par les Américains pour se voir embarquer à bord du plus gros "rover" posé sur Mars : L'explorateur Curiosity, une machine absolument hors du commun. Le fondement de cette manipulation consistait à bombarder un rocher avec un laser très puissant pour faire fondre le matériau. L'analyse spectrale de la couleur des gaz

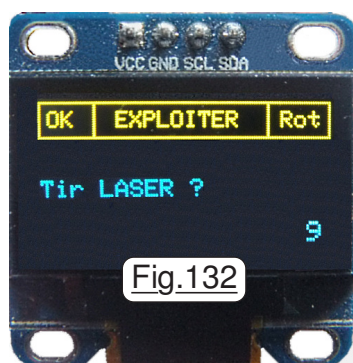


Fig.132

qui en résulte permet aux savants d'en déduire une foule d'informations relatives à la planète rouge. La petite simulation présentée ici consiste à réaliser une rafale d'éclairs de quelques secondes à la puissance maximale quelle que soit l'état actuel du LASER. Seule limite à cette action : Le système électrique de bord ne peut en aucun cas passer outre le disjoncteur s'il est déclenché. (*Mode hibernation.*) Dans ce cas il n'y aura aucun effet mis à part l'accusé de réception **OK** qui termine l'action issue de l'instruction ayant pour code **9**. Pas besoin de se torturer les méninges pour comprendre que c'est l'item de la Fig.132 qui permet de déclencher un tir LASER. **ATTENTION** : Le "rayon de la mort" sera émis inconditionnellement et sans aucune vérification, y compris si le LASER est actuellement éteint ou à son niveau énergétique minimal. **N'activez cette fonction** avec la touche **OUI** que si la direction du canon à photons est dirigée loin du regard de toute personne présente dans le local. Après avoir déclenché la simulation d'un tir au LASER, le système est reconfiguré à son état initial. À la fin du tir, on retrouve le niveau énergétique et l'état allumé ou éteint du dispositif rayonnant.

► Imposer numériquement la valeur du Cap désiré.

Décrit en détail en page 20 le tableau de bord du pupitre affiche les données de navigation qui prennent en compte la valeur du **CAP désiré**, ce dernier définissant la route à suivre. Pour **choisir la direction magnétique que devra prendre en compte le calculateur de bord**, encore faut-il définir sa valeur. C'est l'objet de la fonction que l'on trouve quand on effectue un pas de plus dans le sens horaire, étant dans le menu **EXPLOITER**. (*Voir l'encadré situé en haut de la page 28 pour exploiter de façon conviviale le manuel.*) **Imposer un Cap Magnétique désiré** peut se faire agréablement par utilisation du codeur incrémental. Dans les items du menu **EXPLOITER**, avec cinq pas dans le sens négatif à son appel, on fait afficher la fenêtre de la Fig.133 qui validée avec la touche **OUI** ouvre la page de saisie. Par défaut sur un RESET la route désirée est plein Nord, donc un Cap Magnétique nul. C'est ce que montre la Fig.134, le code **96** servant à récupérer la valeur initialisée sur la sonde. Comme abordé déjà plusieurs fois, nous savons que pour des raisons de fiabilités des informations affichées sur la console, **les valeurs sont préservées sur la sonde et c'est par interrogation de cette dernière que l'on rafraîchit l'écran OLED**. La valeur actuellement consignée dans la sonde est donc récupérée par dialogue sur la ligne série, puis affichée sur le petit écran OLED.



Fig.133

USAGE RATIONNEL DES DIVERS MENUS DU PUPITRE :

Ouvrez le [Manuel d'UTILISATION.pdf](#) en son centre pour y observer les commandes relatives aux trois menus les plus utilisés. Pour effectuer un **Tir LASER** par exemple, il faut effectuer 12 pas dans le sens horaire à partir de l'item disponible en accueil. Hors en rotation inverse, six pas sont suffisants. *Pour minimiser les manipulations*, chaque menu est séparé en deux zones. Celle avec les symboles \downarrow et celle avec en tête d'item le caractère \uparrow . À simple vue du tableau, vous saurez ainsi pour atteindre l'item de votre choix dans quel sens tourner le codeur incrémental et combien de pas sont nécessaires. Vos manipulations seront ainsi optimisées.

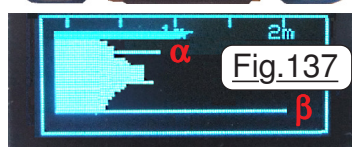
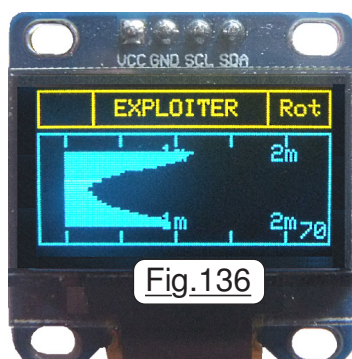


Les deux touches de Translations latérales \leftarrow et \rightarrow servent à déplacer en permutation circulaire un curseur d'écriture sous les trois chiffres du Cap. (Trois chiffres significatifs sont affichés systématiquement avec éventuellement des zéros en tête.) Ce curseur symbolisé par une flèche verticale indique le chiffre qui sera modifié si on tourne le codeur incrémental quand on est en mode de saisie de la route souhaitée. Quand le gros bouton rotatif est manipulé dans un sens ou dans l'autre, on a l'impression visuelle que seul le chiffre pointé est modifié. En réalité, on agit sur un nombre, c'est à dire que l'entité complète est augmentée ou diminuée. Par exemple faire un pas négatif d'une unité quand le cap vaut

100° fait passer l'affichage à 099°. En négatif une butée virtuelle bloque la valeur à zéro. En positif, si arrivé à 359° on tourne le capteur dans le sens horaire, quel que soit le chiffre indexé la valeur repasse à 000°. La sortie du mode "Saisie du cap pour la route souhaitée" se fait impérativement avec **FIN**. Pour des raisons de convivialité la saisie du cap s'ouvre avec l'index sur le poids significatif des dizaines. Toute touche du clavier autre que **FIN**, \leftarrow et \rightarrow génère un BIP sonore d'erreur.

➤ Le panoramique télémétrique.

Assimilable à un mode torsion qui serait automatique, cette fonction enrichi les possibilités du télémètre. L'idée de base consiste à effectuer un balayage frontal sur un angle suffisant, et à enregistrer simultanément les retours du télémètre pour construire un spectre représentant les formes des obstacles situés devant la sonde. Un peu comme un radar à ultrasons. Ne rêvons pas, la définition angulaire des capteurs ultrasons utilisés est franchement médiocre et le spectre obtenu sera "très moyen", on n'y reconnaîtra pas le chat Pignouf ou le beau visage d'Alice. Reste que la manipulation en soit reste très séduisante. Les commandes pour réaliser un panoramique télémétrique du menu **EXPLOITER** sont présentées sur la Fig.135 respectivement en **A** pour l'enregistrement en EEPROM, et en **B** pour sa visualisation. Le processus est tout à fait comparable à celui de l'enregistrement ou de la restitution d'un spectre colorimétrique, et présente des contraintes analogues. (Armer **SÉCURITÉ**



pour pouvoir enregistrer etc.) En observant finement l'affichage sous forme graphique montré sur la Fig.136 on peut déduire que la définition verticale du tracé se fait sur trente deux enregistrements pour laisser une place suffisante aux graduations de distance, la portée maximale du dispositif étant de deux mètres. Cette définition est largement suffisante vu la médiocre ouverture angulaire des transducteurs ultrasons et la présence de parasites comme ceux présents sur la Fig.137 en α et β par exemple. Les protocoles pour ces deux fonctions sont décrits en détails dans la page 21 du [Manuel d'UTILISATION.pdf](#). Quand on enregistre un panoramique, ce dernier est inscrit dans la mémoire non volatile EEPROM du microcontrôleur, et la fonction se termine par la visualisation de ce dernier. Il sera possible de visualiser ce graphe à n'importe quel moment en faisant appel à la commande **AFF un balayage ?** dont la sortie se fait avec **FIN**, comme précisé par la LED jaune clignotante.

➤ Sauvegarder la posture actuelle.

Page 26 de ce didacticiel, nous avons abordé la possibilité de visualiser les douze consignes des servomoteurs SG90, mais également celles de postures préenregistrées en EEPROM. Il restait à expliciter comment sauvegarder les données correspondant à la configuration actuelle de la sonde. Pour mémoire, cette fonction est utile lorsque l'on a passé pas mal de temps à créer une configuration originale et que l'on désire la retrouver par la suite à tout moment. C'est précisément l'objet de l'avant dernière option du menu **EXPLOITER**. La page écran montré sur la Fig.138 montre que le numéro de l'empreinte qui sera modifiée est dans notre exemple le (Num 4). Naturellement une procédure nous octroie la faculté à tout moment de changer la zone mémoire EEPROM qui sera affectée à l'écriture des données de posture. Comme il n'est pas question "d'écraser" par erreur une configuration déjà présente à l'emplacement visé, **SÉCURITÉ** est exigée. Le protocole n'est pas immédiat, car pour armer **SÉCURITÉ** nous sommes obligés de passer provisoirement dans le menu des **OPTIONS** ou le menu **APPRENDRE**. C'est du reste dans ce dernier que l'on débute le protocole de désignation de la posture indexée en EEPROM, technique détaillée en page 10 du [Manuel d'UTILISATION.pdf](#).



Fig.138

41) Le menu des déplacements "élémentaires".

Impératif pour pouvoir exploiter sur le terrain la petite machine, le menu pour télécommander des déplacements est assez particulier. À franchement parler, ce n'est pas un menu mais une fonction purement opérationnelle. Du reste, quand on tourne le codeur rotatif il n'y a pas de changement d'item sur l'écran de maitrise de la lointaine exploratrice. C'est la seule option de pilotage pour laquelle dans le petit cadre jaune il y a affichage en 1 de **CLV** au lieu de **Rot**. Pour bien nous informer que dans ce mode c'est le clavier qu'il faut utiliser, et la LED bleue s'illumine. Considérons sur la Fig.139 la page écran relative à cette fonction. Le titre **MOUVEMENTS** en 2 précise sans ambiguïté que l'on va imposer à la sonde de se déplacer. Pour le cas où vous n'auriez pas totalement pressenti le danger, le logiciel insiste en 3 avec l'avertissement **ATTENTION : - DEPLACEMENTS -**. Pourquoi une telle obstination ? *On ne pilote pas du tout une sonde comme une automobile. Entre le moment où l'on télécommande une consigne et celui où la sonde la réalise puis rend compte, il peut se produire un délai jusqu'à une heure pour recevoir l'ACR. C'est un peu comme si vous conduisiez votre automobile en ouvrant un court instant les yeux, en analysant la route, en corrigeant au volant puis en fermant les mirettes durant une heure sans savoir ce qui se passe !* Aussi, dans la réalité astronautiques, pour ne pas risquer de faire tomber la machine dans un ravin, on analyse avec finesse les images issues de la caméra de bord. Puis on décide soit de changer de CAP, soit de faire un déplacement élémentaire. Si le terrain est plat, on peut se risquer à enchaîner plusieurs étapes, mais on ne cherchera jamais à couvrir une longue distance d'un coup, c'est contraire aux protocoles astronautiques. Pour des raisons de réalisme, on s'en tiendra à de telles procédures. Néanmoins, il nous sera facile de titiller frénétiquement les boutons de déplacement pour bouger de façon notable et à vue directe le petit insecte mécanique si on le désire ... Quand on tourne le codeur incrémental le nombre indiqué en 4 par **Repetier N fois** change entre 1 et 9 dans un sens qui dépend de celui de la rotation. Lorsque vous déclenchez un déplacement élémentaire, en n'appuyant qu'une seule fois sur la touche désirée, la machine va automatiquement enchaîner N fois le déplacement demandé au robot. Aussi, vous allez rapidement constater que 9 fois c'est beaucoup et vous prendrez assez vite la bonne habitude de replacer à 1 le facteur de répétition, sauf cas particulier. Par exemple j'utilise la répétition de 9 pour effectuer les test d'autonomie sur un accumulateur, ce genre de chose ... encore que dans ce cas il vaut mieux faire apprendre à la machine un long programme comportant tout une suite de mouvements qui se succèdent. La sonde étant alors sur un berceau on déclenche le programme qui anime les moteurs sans avoir à surveiller JEKERT. Notez au passage que sur un RESET le facteur de répétition est forcé à 1. En revanche, si vous modifiez cette valeur, elle sera conservée dans la mémoire du système quand

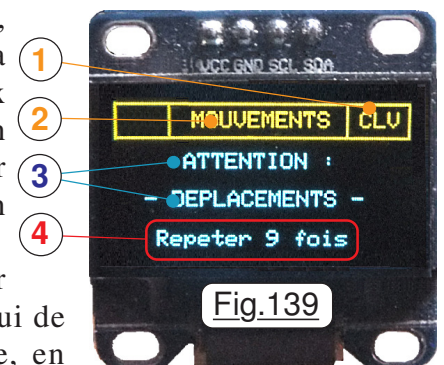
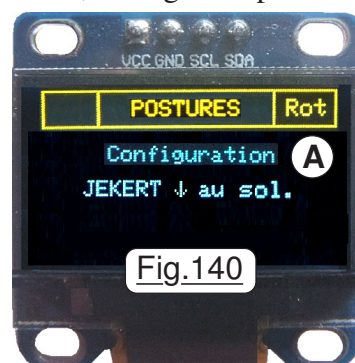


Fig.139

vous quittez les déplacements. Lorsque vous revenez dans la fonction **MOUVOIR**, l'ancien facteur ajusté pour la répétition est retrouvé. Quand on impose un déplacement, (*Avec répétition ou non.*) on a l'impression que le système s'est bloqué. (*Surtout si les moteurs sont sur OFF ou que l'on n'a pas encore distribué l'énergie de puissance.*) Ce n'est en rien un aléa de programmation. Le pupitre a envoyé à la sonde une directive. Après avoir parlé, n'oublions pas que nous sommes en alternat, le logiciel passe à l'écoute et attend un accusé de réception issu venant de Mars. Tant que la ligne **RX** reste muette, le programme espère avec patience. Lorsque JEKERT a entièrement effectué le ou les déplacements élémentaires, alors seulement elle transmet sur **TX** son accusé de réception. La raquette de commande redevient disponible ce qui visuellement se voit par l'affichage de l'ACR attendu **OK**. Aussi, il peut se montrer agassif de déclencher un déplacement alors que l'on a oublié de replacer à **1** le facteur de répétition. C'est la raison pour laquelle personnellement je ne l'utilise que très peu. Dans ce mode on peut librement allumer ou éteindre les PHARES et le LASER, mais pas en changer la puissance avec le gradateur. (*N'oubliez-pas non plus que piloter un allumage ne sera effectif que si le disjoncteur n'est pas déclenché.*) La seule façon de sortir consiste à appuyer sur l'une des touches de MENU ... par exemple celle des **POSTURES** que l'on va aborder maintenant.

42) Le menu des POSTURES.

Nombreuses sont les postures que doit pouvoir adopter une sonde d'exploration planétaire lors de son exploitation sur le terrain. Certaines servent de base comme **Stable transversal** pour effectuer les déplacements élémentaires. Chaque pas en translation et en orientation se termine par un retour à cette configuration. Ainsi, *cet invariant* facilite le développement logiciel pour élaborer la gestuelle du quadropède. Bien qu'engendrant des mouvements sur la machine, changer de posture ne génère pas de risque de voir JEKERT tomber dans un ravin ou heurter un obstacle. Les changements de configuration ne provoquent pas de déplacement à proprement parler, ils se font "sur place". Ces adoptions d'attitudes ne constituent pas des déplacements d'exploitation, ce sont des configurations particulières qui ne sont justifiées que dans des cas singuliers peu fréquents. C'est la raison pour laquelle ces commandes




sont regroupées à part dans le menu des **POSTURES** pour ne pas trop encombrer la fonction **MOUVOIR** déjà bien chargée. La Fig.140 présente l'écran d'accueil du menu qui propose en premier la configuration qui permet de placer la sonde en posture de **VEILLE**. Dans la pratique c'est la consigne qui impose à JEKERT de poser son bouclier sur le sol pour libérer les efforts dans les **jambes** et présenter une stabilité maximale en cas de vents forts. (*Tempête hypothétique.*) L'information **Configuration** en **A** est une constante textuelle sur les écrans du menu des postures.

L'écran suivant, quand on tourne le codeur rotatif dans le sens horaire, et visible sur la Fig.141, est plus "dynamique" en ce sens qu'il s'accompagne d'un BIP sonore. Cette alerte a pour but d'attirer l'attention de l'opérateur sur **Retour par FIN.** qui prévient qu'il nous faudra quitter par **OUI** si on valide cette option. *En effet, certaines postures particulières imposent un retour à une configuration de base par un mouvement coordonné spécifique.* La posture **Stable Raisonnable** correspond à la plus grande surface de sustentation possible, attitude que l'on pourra adopter lorsque la sonde est sur un terrain en forte pente pour en assurer une meilleure stabilité. L'écran **2** correspond à l'adoption de la **Hauteur Maximale** pour laquelle les **Tibias** et les **Griffes** sont en orientations verticales. Il s'agit également d'une posture qui doit être suivie d'une touche **FIN**. Elle permet aux caméras supposées se trouver à bord de voir le plus loin possible pour analyser la situation présente. Puis, tournant le bouton toujours dans le sens horaire, les écrans vont se succéder

dans l'ordre des copies d'écran montrées entre **3** et **9**.

La configuration **3** place tous les moteurs à leur origine zéro, c'est à dire au centre des plages angulaires de leurs rotations possibles. Après avoir déposé JEKERT au sol c'est une posture qui peut s'avérer bien utile, et en particulier s'avérer une bonne base de départ quand on veut créer de nouvelles configurations originales. Faire attention toutefois à **Configuration Apprise en EEPROM** qui suppose que dans la mémoire EEPROM du microcontrôleur (*Dont le numéro actuellement pointé est indiqué entre parenthèses car nous disposons de huit empreintes possibles.*) on a logé au moins une fois une configuration cohérente. Si l'EEPROM est vierge et n'a été chargées que par les postures de base et les textes, la zone réservée à tout ce qui concerne l'apprentissage ne contient que des \$FF. Si vous déclenchez l'option **4** les servomoteurs vont diverger.

Les trois postures qui suivent sont placées dans l'ordre logique de leur utilisation dans l'hypothèse du début de la mission qui commence forcément par le lancement. Quand on affiche l'écran **5** un BIP d'alerte se fait entendre pour attirer notre attention sur le fait que la sonde doit impérativement se trouver sur son (**BERCEAU !**). Si elle est au sol quand vous validez cette configuration ... les moteurs ne vont pas apprécier. Le programme teste pour vérifier que le micro-contacteur de présence "du sol" est bien activé. Si ce n'est pas le cas un **E7** va se déclencher, accompagné d'un autre BIP d'alerte. Même punition en **7** si **Deposer JEKERT** est validée et que le contact avec le sol n'est pas effectif. On suppose que la sonde s'est posée. Pour la protéger de cette phase critique dans la mission, on désire que l'impact se fasse sur le bouclier. C'est la raison pour laquelle il faut ramener les quatre **Jambes** vers le haut au dessus de la structure. Pour atteindre ce but on utilise la fonction **6**.

La fonction **7** vérifie que le bouclier est bien en contact avec le sol. Puis la sonde est déployée en configuration de base **Stable Transversal**. Enfin on fait "trépigner" l'insecte mécanique pour libérer les efforts dans les articulations. *Notez au passage que dans ce menu, cliquer sur le **BPccr** permet à tout moment à l'opérateur de déclencher ce processus de décharge des efforts mécaniques parasites.* Vous avez compris qu'à tout moment la parsonne aux commandes peut invoquer **Stable Transversal** avec la dernière option de ce menu présentée en **9**. En observant l'option **4** vous avez remarqué que l'empreinte actuellement pointée en EEPROM est la **n°1**, c'est celle qui sera adoptée si on valide la fonction. L'option **8** sera celle que l'on va invoquer pour indexer une autre posture en EEPROM. Vous constatez que c'est la **n°4**. La procédure pour désigner une cible est décrite en bas de la page 10 du  et impose au préalable de valider le menu **APPRENDRE**, d'y sélectionner une valeur (*Ici le PGM n°4.*) puis d'invoquer la fonction **8** du menu des **POSTURES**. Valider la fonction indexera alors la posture désirée, et son n° sera désormais précisé en option **4**.

43) JEKERT retourne à l'école en apprentissage.

Spéciale au spatial, la gestion des personnels est très particulière, notamment pour les ingénieurs en génie logiciel. Il faut savoir qu'une sonde qui va vers une planète du système solaire doit voyager durant des mois avant de parvenir dans la banlieue de sa cible. Par exemple pour Mars, dans les fenêtres de lancement les plus avantageuses la machine scientifique va mettre plus de six mois terrestres pour arriver dans la sphère d'influence du corps céleste.

Imaginons que le nombre d'ingénieurs informaticiens recrutés soit suffisant pour développer entièrement les logiciels d'exploitation embarqués entre le début du projet et le jour du lancement. Quand la fusée quitterait le pas de tir, ils se trouveraient au chômage. Aussi, pour "réguler" leur charge de travail, on recrute deux fois moins de personnes. Du coup, quand on allume les moteurs, la sonde est totalement inutilisable. Seules les séquences qui vont permettre d'effectuer le décollage, le voyage, la mise en sommeil, les télémesures et l'atterrissage sont disponibles. Pendant les six mois durant lesquels la sonde est en sommeil, à Terre les ingénieurs continuent à créer les programmes d'exploitation. Et, régulièrement, le vaisseau étant perdu dans l'immensité du vide sidéral, on réveille l'insecte et on complète ses performances. Cette méthode propre au spatial qui induit **des aptitudes à l'apprentissage** impose que le programme de base résident qui anime la machine soit, lors du lancement vers la planète à explorer, au minimum capable de :

- Débuter un apprentissage. (*Et naturellement suspendre ce mode.*)
- Lister le programme actuellement mémorisé. (*Il faut le vérifier avant de s'en servir.*)
- Sauvegarder le programme en mémoire non volatile EEPROM.
- Utiliser à convenance le programme actuellement sauvegardé.
- Effacer un programme devenu caduque qui encombre la mémoire.

➤ La fonction temporisation d'une seconde.

Ouvrons une petite parenthèse qui n'est qu'indirectement liée aux techniques d'apprentissage de JEKERT. Imaginons que lors d'une démonstration, on désire faire clignoter les phares par exemple. Manuellement les allumer et les éteindre en respectant une certaine cadence n'est pas impossible, au prix d'une "galère bien indigeste" c'est faisable. En revanche, enregistrer un programme qui réalise une telle action sera facile, si toutefois la commande "temporisation" est prévue dans le logiciel d'exploitation de la sonde. Le code **90** a été réservé dans ce but et impose au programme de JEKERT de réaliser une temporisation d'une seconde. Il sera intégrable à convenance dans un bloc de programme d'apprentissage, à condition toutefois de pouvoir déclencher cette commande dans l'un quelconque des menus du pupitre.

Étant en train de faire apprendre un programme à la sonde, nous pouvons être amenés à jongler entre les instructions de déplacements et celles d'exploitation. Intercaler des temporisations d'une seconde entre les instructions deviendrait un enfer si cette fonction était disponible sous forme d'un item noyé dans un quelconque menu. C'est la raison pour laquelle, dans la version actuelle du logiciel, le code **90** sera généré par le **BPccr** dans le menu **EXPLOITER**. Cette commande ne sera effective que si l'on n'est pas en mode **Utilisation du LASER** ou en **Pilotage MANUEL** des servomoteurs, car pour ces actions le **BPccr** n'est pas disponible.

Pière maitresse du mécano logiciel embarqué par la sonde : Sa faculté d'apprendre, de pouvoir installer dans sa mémoire non volatile de nouveaux programmes qui seront développés en toute sérénité à Terre. (*Donc à partir du pupitre de pilotage.*) L'expérience a montré depuis des décennies que cette possibilité de télécharger du code à bord des vaisseaux spatiaux a sauvé plus d'une mission, les exemples astronautiques foisonnent. Outre les impératifs listés en bas de la page 31, si l'on désire que la programmation opérationnelle de JEKERT soit performante, il faut ajouter :

- Éditer le programme, c'est à dire pouvoir y ajouter des instructions,
- Corriger le programme, c'est à dire pouvoir effacer sa (*Ou ses*) dernière(s) instruction(s),
- Créer non pas un seul programme, mais jusqu'à **neuf blocs logiciels indépendants**,
- Pouvoir chaîner librement plusieurs de ces blocs programmes dans une séquence unique,
- Pouvoir déplacer l'un des blocs programmes dans une séquences.

Nous allons pas à pas passer en revue ces différentes fonctions, et décrire leurs spécificités ainsi que les écrans dédiés sur l'afficheur OLED. C'est l'objet des chapitres qui vont se succéder.

➤ Indexer le programme cible.

Indexer signifie : Montrer du doigt. Instinctivement, quand on désire désigner une direction à notre interlocuteur, on utilise le doigt le plus long de notre main, c'est à dire l'index. Le vocable "indexer" en dérive. Dans notre cas il signifie que dans les neuf candidats "nominés", on devra systématiquement préciser au logiciel lequel on veut effacer, lister, modifier etc. Que l'opérateur soit en permanence informé de la cible est vital. Aussi, quand on consulte les pages du menu des **DONNEES**, l'emplacement réservé aux ACR dans les autres menus indiquera quel est actuellement le programme indexé par la sonde. Par exemple sur la Fig.142 en **X** on note qu'actuellement la sonde pointe le bloc programme n°3 dans la

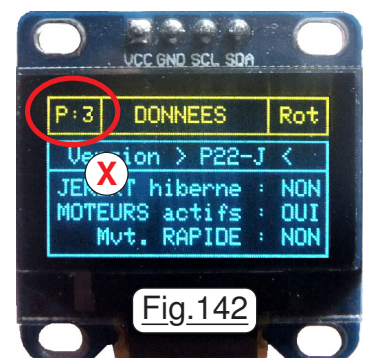


Fig.142

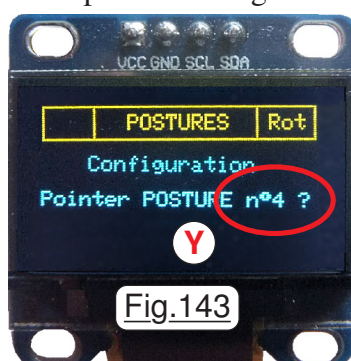


Fig.143

liste, c'est lui qui sera l'objet des traitements relatifs à l'apprentissage. C'est dans le menu des **POSTURES** que l'on trouve l'item de la Fig.143 qui, si on le valide avec **OUI**, imposera comme empreinte indexée celle précisé en **Y**. On débute le protocole en ouvrant le menu **APPRENDRE**, qui débute sur l'écran de la Fig.144 la zone des ACR en **Z** étant "vierge". Comme vous vous en doutez, si l'on valide l'option, on passe dans un mode de saisie d'un chiffre qui ne pourra être sélectionné par rotation du gros bouton qu'entre 1 et 9, la référence désignée pour les blocs de programmes étant mémorisée en EEPROM. Durant cette saisie, comme montré en **Z** la petite zone réservée à l'affichage des ACR

précise la valeur qui sera retenue comme index quand on sortira de la saisie du chiffre à enregistrer par la touche **FIN**. Du reste, pour ne pas oublier cette contrainte, la LED jaune du clavier clignote à une cadence relativement rapide. À partir de ce stade, c'est maintenant le bloc **ProGraMme** qui sera traité comme cible pour les diverses fonctions d'apprentissage. (Ce sera également cette valeur qui sera prise en compte quand on désirera désigner une empreinte de posture en EEPROM dont la Fig.143 présente la page écran.)

➤ Lister le contenu du bloc programme indexé.

Fonction absolument indispensable, que ce soit en cours d'écriture du programme ou avant d'en déclencher l'exécution, il importe de pouvoir à tout moment en consulter le contenu. Ce dernier est constitué d'une liste d'instructions reconnues par la sonde sous forme des consignes dont les



Fig.145

codes sont proposés dans les pages 29 à 31 de [Manuel d'UTILISATION.pdf](#). Notez au passage que toutes les consignes ne sont pas recevables. Seront ignorées en enregistrement celles précédées de "*" dans le livret.

Il est évident que suite à la validation de l'écran Fig.144 sur cet exemple c'est le programme n°2 qui est indexé, donc listé. Toutefois, nous serons pénalisés d'une erreur **E15** si le bloc programme est vide. (Ou est effacé.)

Testons par exemple le bloc n°6 ce qui impose la procédure de désignation explicitée en bas de la page 32. On peut alors valider l'affichage des données, ce qui ouvre l'écran de la Fig.146 relatif à la cible désignée. En

surbrillance en **A** est précisé le numéro du bloc programme. La cellule juste en dessous contient les consignes envoyées par le pupitre à la sonde. (Ici le 86 pour demander de lister.) Chaque autre cellule de la grille pourra contenir un code consigne. Dans cet exemple le bloc comporte 21 instructions. Le nombre maximum par bloc de programme sera donc de 30 instructions. Si tout va bien, nous ne devrions jamais voir figurer les codes non valides repérés par les symboles "*"/. Si d'aventure l'un de ces interdits figurait, il faudrait impérativement analyser le problème informatique potentiel. Examinons à titre d'exemple le contenu du bloc programme **P6** : Le **55** réveille JEKERT. (Fait sortir la sonde du mode hibernation.) **54** force PHARES et LASER au niveau d'énergie 128. La suite fait clignoter les phares à 1Hz et s'arrête en configuration phares éteints. (27 allume les phares, 90 réalise une temporisation d'une seconde et 28 éteint les deux phares.) Notez que le [Manuel d'UTILISATION.pdf](#) en page 26 décortique les blocs programmes n°4 à n°9 qui sont disponibles en EEPROM quand on a logé les données en EEPROM avec **P60_Clone_EEPROM_sonde.ino** mentionné sur le troisième tutoriel. Notez que pour effacer l'écran, ce n'est plus la touche **FIN** qui sera à cliquer, mais tourner le codeur rotatif dans l'un des deux sens possibles. L'écran est alors effacé et la commande potentielle affichée.

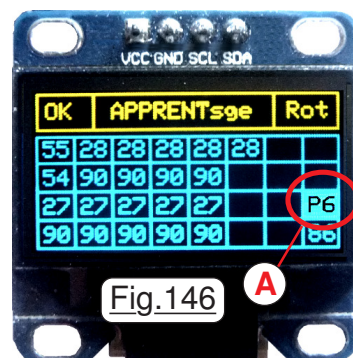


Fig.146

➤ Corriger le bloc programme indexé.

Fonction affichée quand à partir de l'item de la Fig.145 on tourne le codeur incrémental d'un pas dans le sens horaire. Cette fonction qui consiste à effacer la dernière instruction d'un bloc n'est possible que durant une session en **mode apprentissage**. Quand on ferme le mode enregistrement des instructions, la seule modification possible reste l'effacement total pour recommencer. Tenter la manipulation hors **mode apprentissage** vous gratifiera d'une erreur **E16** signalant l'obligation d'avoir la LED rouge "apprentissage en cours" rouge allumée. Sécurité activée et mode apprentissage ouvert, **OUI** effacera, comme visible sur la Fig.147, le dernier code du programme actuellement indexé. On peut réitérer cette commande à notre guise, les effacements "remontant" vers le début. Les codes **89** se soldent par un **OK** tant que c'est possible. Quand il ne reste plus rien à enlever, **E17** nous précise que ce n'est plus la peine d'insister ... **circulez ya rien à voir !** Notez que cette

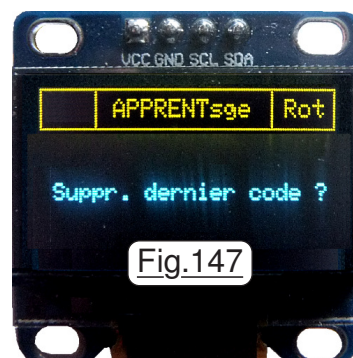


Fig.147

commande n'est logique que dans la mesure où quelques codes sont à enlever en vue de corriger un listage comportant un grand nombre d'instructions et que l'on ne désire modifier que les dernières. Si c'est pour tout effacer autant faire appel à la commande de la Fig.148 qui élimine la totalité d'un coup :

➤ **Effacer un bloc programme.**

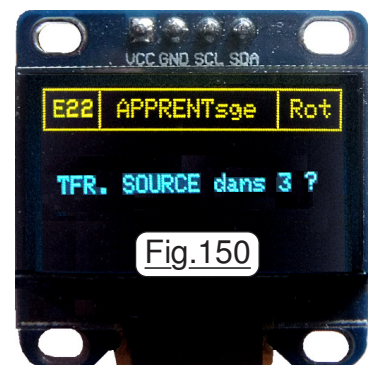
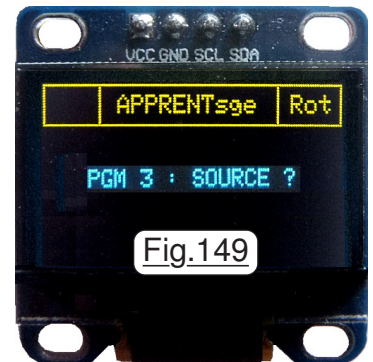
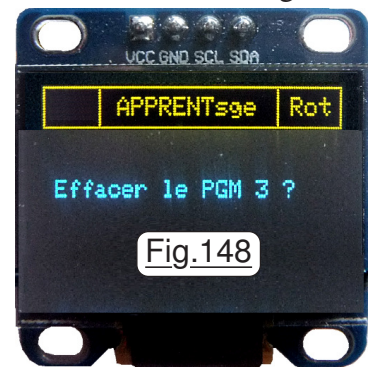
Comme on s'en doute, la commande spécifique qui permet d'effacer à convenance le bloc programme actuellement indexé suppose au préalable d'avoir armé **SÉCURITÉ**. L'item qui suit celui de la Fig.147 dans le menu affiche l'écran de la Fig.148 qui précise le numéro du bloc concerné. On pourra changer cette référence si nécessaire. (*Voir la procédure en page 32.*) Tenter d'effacer un bloc actuellement vide déclenchera une erreur **E13** associé à son bip sonore. Notez au passage que le dernier code transmit à la sonde est souvent **44**, c'est celui qui achève une commande pour en vérifier les effets réels sur la sonde.

➤ **Déplacer le contenu d'un bloc programme.**

Plusieurs raisons peuvent nous amener à déplacer un programme. Neuf zones possibles constituent une manne qui sera probablement largement mise à contribution. Chaque utilisateur va se concocter des programmes "meumeu" que l'on sera ravi de déclencher pour faire une petite présentation de JEKERT à des amis. Aussi, au cours du temps certains vont ressortir comme méritant une résidence pérenne en EEPROM alors que d'autres moins ludiques seront effacés. On aura probablement envie de placer à la fin les incontournables, et mettre au début les expérimentations nouvelles. Enfin, comme il est possible de chaîner en automatique plusieurs blocs programmes, on pourra de ce fait réaliser des modules qui une fois mis au point devront se succéder dans l'ordre souhaité en EEPROM. Cinq étapes seront nécessaires pour déplacer un programme d'une zone à une autre :

- Écrire dans le bloc qui servira de source. (*Voui voui voui, on s'en doutait un tantinet !*)
- Préciser à la sonde quel est le bloc ciblé pour effectuer un transfert. La commande sur le pupitre est visualisée sur la Fig.149 qui n'impose pas la **SÉCURITÉ**, car ne présente pas de risque particulier.
- **Indexer** le programme qui servira de réception du code **SOURCE** transféré. (*Voir en page 32.*)
- Effacer la zone de réception. En effet, pour ne pas courir le risque d'écraser un module, (*C'est la terminologie consacrée en informatique.*) le transfert ne peut se faire que sur une zone vide. Comme l'effacement impose d'armer la **SÉCURITÉ**, on ne perdra pas un listage par erreur ... tout au moins en principe mais alors c'est que vraiment vous n'aurez pas fait bien attention !
- **Transférer le programme source dans la zone de réception** supposée préalablement vidée, ou vous serez "punionnés" par un **E24**. Transférer un programme sur lui même est illégal, car par logiciel le transfert se termine par l'effacement de la zone **SOURCE** pour qu'elle soit disponible à l'édition. De ce fait, un transfert gigogne effacerait le programme source. (*Dommage !*) Pour éviter cet aléa une telle tentative se solde par une erreur **E23**. **Pour procéder à un transfert, il faut que le mode apprentissage soit fermé.** Comme montré sur la Fig.150 si ce n'est pas le cas une erreur **E22** sera générée. Enfin, vous vous en doutez probablement, la **SÉCURITÉ** sera armée pour que le **OUI** ne se solde pas par **E12** dont on finit par bien cerner la signification. Quand toutes les conditions sont respectées, enfin le code **13** se termine victorieux par le **OK** tant attendu. Certains vont trouver que la procédure complète est un peu compliquée. Concrètement, elle est bien plus facile à pratiquer qu'à décrire.

RESUMÉ : Les deux commandes qui suivent **Effacer le PGM N ?** quand on tourne le codeur incrémental dans le sens horaire sont associées pour pouvoir transférer un bloc quelconque dans un bloc préalablement effacé. En particulier transférer un bloc permet de construire une séquence ordonnée de plusieurs blocs pour former un "macro-programme". On peut ainsi mettre chaque bloc au point, puis les placer dans l'ordre convenable en vue de former une séquence chaînée dont il est question plus avant.

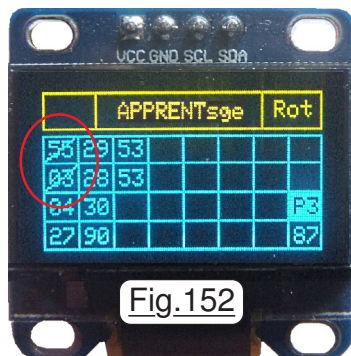


➤ Exécuter le programme indexé.

Raison d'être de toutes les commandes d'apprentissage qui ont occupé les chapitres précédents, déclencher l'exécution d'un programme enregistré n'a rien d'anodin. La logique de base semble simple et se résume à une boucle du genre : *"tant que l'on n'a pas terminé, on analyse le code pointé, on réalise la fonction qui le concerne, on passe au code suivant"*. Sous cette forme, l'opération globale semble assez simple. La difficulté consiste à continuer le dialogue entre la sonde et le pupitre, hors perdre la synchronisation est particulièrement risqué dans une telle séquence. C'est du reste pour cette raison que certains codes ne sont pas autorisés, le chapitre suivant va décrire la boucle et son fonctionnement. Si tout se passe bien, quand la raquette aura donné l'ordre **87**, le programme esclave va dérouler les instructions. Quand toutes auront été réalisées, il retournera le **OK** qui sur le pupitre rendra la main à l'opérateur. Plusieurs problèmes peuvent se présenter. Vous avez compris que pour JEKERT cette fonction est l'une des plus dangereuses. Par exemple vous avez aligné un nombre exagéré de mouvements en avant, puis sans trop réfléchir déclenché le processus :

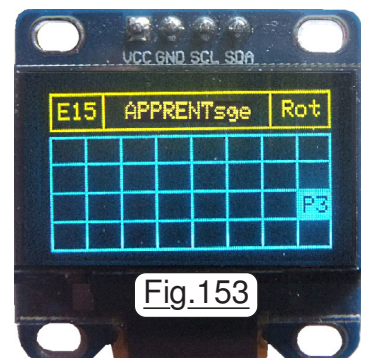
- Et Totoche, ya plus la sonde, c'est normal ?
- Ben oui Dudule, regardes en bas du ravin !
- Et Totoche, la caméra montre le ciel de Mars, c'est prévu ça ?
- Ben non Dudule, mais sur la droite il y avait un obstacle et JEKERT a basculé sur son flanc ...

D'accord, pour déclencher l'exécution du programme indexé il sera sage de bien analyser le listage pour valider la procédure. Ensuite, on devra impérativement armer **SÉCURITÉ** pour ne pas croiser l'inévitable **E12**. Imaginez également ce qui se passerait si la fonction était invoquée alors que le mode apprentissage est en cours. Il y aurait de l'auto-apprentissage en boucle avec à la clef une saturation à trente, suivie de plus rien ... car le dialogue et la synchronisation seraient bloqués. Aussi pour vous éviter cette fin tragique de la mission, la Fig.151 montre que tout au plus un **E19**



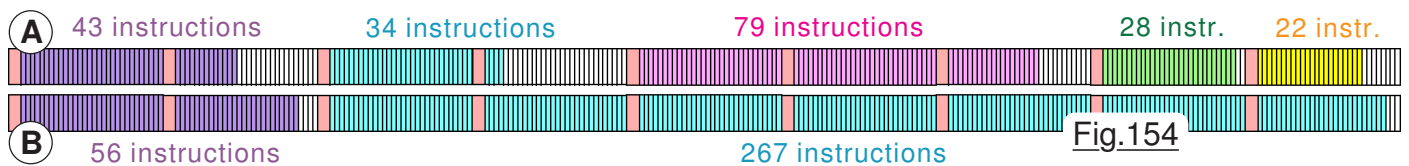
pénalisera notre amour propre, on s'en remettra plus rapidement que les scientifiques qui ont investi dans JEKERT tous leurs espoirs. Par ailleurs, si vous tentez de déclencher un programme vide, **E18** se rappellera à votre bon souvenir. Si tout va bien, **OUI** sera accepté par le logiciel esclave qui affichera la grille, listera les codes du programme pointé du doigt, ainsi que dans la case à droite en bas visualisera le **87**. À partir d'ici, le pupitre n'est plus opérationnel, inutile de titiller les touches ou le codeur rotatif. Sur l'écran OLED, un dialogue de type alternat étant établi entre pupitre et sonde, chaque fois qu'une instruction est achevée on voit, comme sur la Fig.152, la cellule la concernant être biffée. Ainsi

nous savons exactement ce qui se passe. (*Frimeur !*) Puis, si toutes les instructions sont réalisables, un **OK** final achèvera la séquence, le programme maître rendant alors la main à l'opérateur. Outre l'accusé de réception qui signale que le processus a été jusqu'à son terme, la LED **SÉCURITÉ** s'éteint sur le pupitre. L'écran reste affiché jusqu'à ce que l'on clique sur l'un des boutons permanent de menu ou que l'on tourne le codeur incrémental. Reste qu'il n'est jamais exclus que l'une des instructions soit irréalisable. Par exemple le code **3** est puisé dans le listage pour faire avancer la petite machine d'un pas. Si un obstacle se trouve à moins de 8cm, le programme esclave retournera le code d'erreur **E20**, c'est à l'opérateur de déterminer la cause du problème rencontré. Ce ne sera probablement pas trop délicat d'en diagnostiquer l'origine, car l'écran reste figé et *la case de l'instruction qui a engendré une impossibilité opérationnelle n'est pas encore cochée*. Avec douze messages d'erreur typés spécifiquement réservés au mode apprentissage, on peut dire que le logiciel pare déjà pas mal de cas illogiques. Par exemple, et j'ai oublié de le mentionner sur le chapitre du listage d'un programme indexé, si vous tentez la fonction sur un module vide, **E15** vous informe pourquoi, comme visible sur la Fig.153, les cases de la grille réservées aux codes des instructions restent toutes sans contenu. Nous allons pouvoir passer au chaînage des blocs de programme.



➤ Macroprogramme par chaînage de blocs.

Pouvoir chaîner automatiquement plusieurs blocs programmes chamboule fortement les possibilités opérationnelles. Il devient facile de réaliser toutes les combinaisons possibles en taille de programme entre quelques commandes et 270 instructions. Un programme étant indexé, quand on va valider son exécution, on aura au préalable initialisé une variable indiquant combien de blocs successifs devront être parcourus. La Fig.154 présente un exemple de ce qu'il est possible d'agencer dans l'éventail de la combinatoire offerte par cette dernière option. Dans l'exemple **A** nous avons un premier programme de 43 instructions, un qui suit de 34 instructions etc. Dans l'exemple **B** on trouve un premier programme de 56 instructions suivi d'un gros "paquet" de 267 codes obtenu par le chaînage de sept modules. La plus grande séquence possible sera forcément de neuf fois 30 codes, soit les 270 annoncés ci-avant. Pour construire un gros programme, on ne peut que procéder par des apprentissages bloc par bloc. Sur les exemples **A** et **B** les blocs qui débutent les séquences sont saturés à 30, ce qui va dans la logique d'une optimisation. Cependant, rien n'interdit de chaîner des paquets incomplets, l'exécution n'utilisant que les codes enregistrés. C'est dans ce



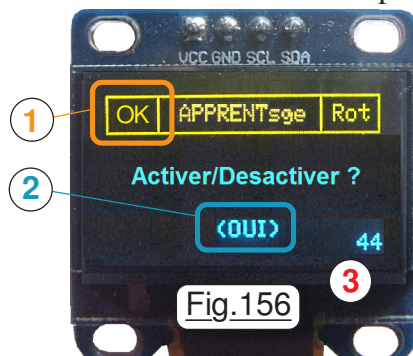
contexte de "panachage" qu'il sera séduisant de pouvoir transférer un paquet à convenance. Ainsi on va mettre dans l'ordre des séquences déjà au point mais réparties un peu au hasard en EEPROM.



Ouvrir le menu **APPRENDRE**, puis faire tourner le codeur incrémental de trois pas en sens antihoraire fait afficher l'item de la Fig.155 qui validé par **OUI** indique en **1** le nombre mémorisé. (Par défaut il est initialisé à un.) La **SÉCURITÉ** doit être activée. La saisie se fait avec le codeur incrémental dans un sens ou son contraire et s'achève impérativement avec la touche **FIN**. La LED jaune qui clignote nous évite de l'oublier. Chaque fois qu'un chaînage est déclenché, les **N** blocs vont être déroulés. À la fin, qu'il se produise une erreur ou pas, la valeur du nombre de chaînages est forcée à **1** et **SÉCURITÉ** est ensuite éteinte. Dans la pratique, il y aura systématiquement chaînage, c'est à dire que le logiciel va déclencher l'exécution du bloc indexé, puis continuer autant de fois que **N** le précise. Comme la fin d'un chaînage force **N** à **1**, par défaut c'est un bloc élémentaire qui sera invoqué. Pour résumer, si vous désirez chaîner plusieurs blocs, il faut au préalable traiter l'option de la Fig.155, puis valider la commande **Activer le PGM** qui précise entre parenthèses le bloc cible. (Voir la Fig.151)

➤ Activer ou suspendre le mode APPRENTISSAGE.

Puisque débiter un apprentissage ne risque pas de détruire des données, tous au plus on va ajouter des instructions au bloc actuellement indexé, **SÉCURITÉ**. n'est pas exigée par la commande de la Fig.156 qui est la dernière du menu **APPRENDRE**. La touche **OUI** peut générer une erreur **E14** qui signale que l'on veut tenter un apprentissage sur un bloc déjà saturé à 30 instructions. Si l'ordre ne pose pas de problème, le programme maître transmet l'ordre **71** suivi de



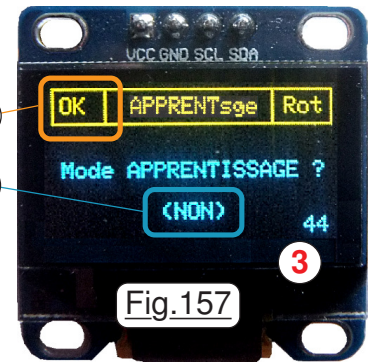
44 comme montré en **3**. Le programme esclave renvoie alors l'état des booléens de la sonde, information qui décortiquée est alors visualisée en **2** complétée par l'ACR **OK**. Pour que l'opérateur puisse savoir en permanence que le mode apprentissage est en cours, la LED rouge **APPRENDRE** s'allume sur le tableau de maitrise du pupitre. (Effectif y compris si l'inverseur des LEDs techniques est sur **OFF**.) Si les LEDs de servitude sur la sonde sont actives, celle qui en rouge témoigne du mode apprentissage est alors allumée. Si on tente une deuxième ouverture, le code **71** ne provoquera aucun effet. Pour mettre fin au mode apprentissage, il faut revenir sur l'item

de la Fig.156 et frapper sur la touche **NON**. La sécurité n'est pas du tout exigée car l'action est neutre. Le résultat est montré sur la Fig.157 sur laquelle on retrouve une procédure

analogue à celle de l'ouverture du mode, le résultat final en 2 étant cette fois négatif. La LED **APPRENDRE** est alors éteinte.

➤ Enregistrement des instructions dans un programme.

Naturellement, vous avez compris que chaque commande envoyée à la sonde verra son code s'inscrire en EEPROM à partir de la première cellule disponible dans le bloc actuellement indexé en EEPROM, puis les unes à la suite des autres viendront encombrer la zone réservée dans la mémoire non volatile du microcontrôleur. Ce naturel admet toutefois des limites, car il faut parer un certain nombre de problèmes. Le premier qui vient à l'esprit, c'est la saturation des trente cellules. L'opérateur portant son attention sur les instructions qu'il impose, et sur les conséquences possibles n'a pas le loisir de compter. Aussi, quand toutes les cellules sont remplies, la sonde n'inscrira plus rien dans l'EEPROM et retournera le code d'erreur **E14**. L'attention étant attirée par le BIP sonore, l'opérateur sera alors averti et pourra prendre les mesures adéquates. La deuxième pierre d'achoppement réside dans certaines fonctions qui bloqueraient l'exécution du programme enregistré. Le deuxième problème a déjà été évoqué, il s'agit des instructions non valides en enregistrement "*" / " qui sont exécutées, mais ignorées pour leur enregistrement en EEPROM.

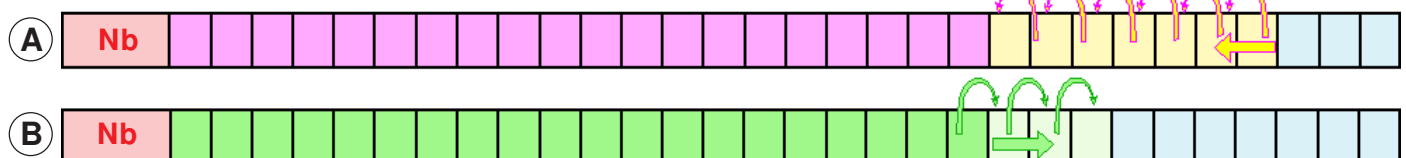


➤ Éditer un bloc programme.

Éditer un bloc programme consiste à ajouter ou retrancher librement des instructions. Dans le cas de JEKERT, nous nous limiterons à ces deux possibilités, mais **uniquement à la fin du bloc indexé**. Pour insérer une ou des instructions "dans le bloc", il faudra effacer toutes celles qui suivent la position désirée, puis ajouter les codes de fin du programme. Cette restriction simplifie le logiciel. Pour effacer les derniers codes la procédure a été décrite en page 33. Pour ajouter des instructions, c'est la procédure décrite ci-dessus, car débiter un apprentissage ne fait qu'ajouter du code à ce qui existe déjà dans le bloc actuellement indexé. La commande se contente d'envoyer l'ordre **20** si le bloc n'est pas vide, et de faire positionner sur la sonde le pointeur correctement en EEPROM à la suite du dernier code inscrit pour le programme indexé. En résumé cette commande est disponible inconditionnellement et sert à ouvrir le mode apprentissage. On peut ainsi éditer directement à partir d'un programme vide, ou le compléter à convenance.

Corriger le programme indexé.

Fig.158



La Fig.158 résume les deux méthodes qui nous octroient la possibilité de modifier le contenu d'un bloc programme. En **A** on procède avec **Suppr. dernier code**. Chaque validation avec **OUI** décale "vers la gauche" l'emplacement de la dernière "case disponible" dans le bloc des trente codes possibles. La valeur de **Nb** est décrémentée conjointement avec le pointeur d'écriture en EEPROM. En **B** la fonction **Activer/Desactiver ?** est Active. Chaque commande valide est inscrite en EEPROM à l'emplacement du pointeur qui est alors incrémenté conjointement avec la valeur **Nb**.

Ce dernier chapitre achève la description des très nombreuses possibilités offertes par le pupitre pour gérer notre petite exploratrice mécanique. Il vous sera parfaitement possible de tester toutes ces fonctionnalités sans avoir réalisé le coffret proprement dit. Cependant, pour être en mesure d'effectuer les premiers essais, il faut au minimum disposer des circuits imprimés et avoir téléversé toutes les données sur les deux microcontrôleurs. La réalisation pratique des divers modules électroniques et celle du coffret proprement dit seront regroupées dans un didacticiel à part. Dans ce dernier seront également détaillées les programmes à téléverser sur les deux petites cartes Arduino NANO pour y loger toutes les données binaires. Néanmoins, avant de clore ce tutoriel, consacrons deux pages supplémentaires :

La première sera dédiée à la réalisation des petits manuels.

La deuxième précisera la façon la plus rationnelle d'utiliser le manuel d'exploitation.

44) Un petit manuel d'utilisation de la sonde JEKERT.

Compte tenu du nombre considérable de fonctions et options disponibles pour piloter JEKERT, il me semble impossible de se remémorer toutes les commandes sur le long terme, ainsi que le comportement attendu des programmes. Par ailleurs, certaines consignes peuvent engager des risques matériels si des conditions initiales ne sont pas respectées. Enfin, certaines manipulations imposent une suite notable d'instructions qui doivent s'enchaîner dans un ordre strict. Avoir des protocoles optimisés sera forcément un atout important. Encore faut-il que le manuel d'exploitation de la petite machine soit commode et convivial. Ce chapitre est prévu pour vous donner quelques détails sur la conception du livret et la façon rationnelle de s'en servir.

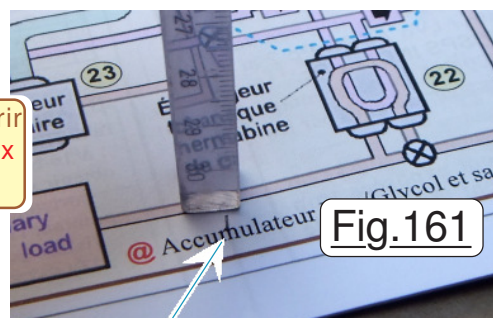
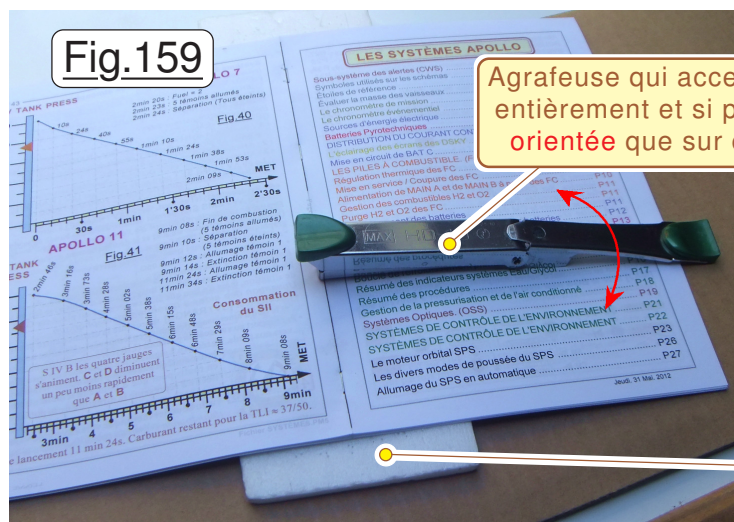
➤ **Réalisation matérielle du livret d'exploitation de JEKERT.**

Agencé à un format A5, les faibles dimensions de ce manuel en font un document parfaitement adapté à son usage. Pas trop petit, les dessins et schémas sont de dimensions suffisantes, pas trop gros, il trouve facilement sa place dans le coffret de rangement du petit robot et n'encombre pas exagérément la table lors des démonstrations. Le fichier **Manuel d'UTILISATION.pdf** est prévu pour imprimer RECTO/VERSO. Il importe donc de choisir du papier d'épaisseur "normale" pour ne pas que l'encre ne traverse. **Papier recyclé méga écolo** s'abstenir ! Personnellement je commence par imprimer toutes les pages impaires. Puis, paquet de feuilles replacé sur le bac à papier de la machine **CORRECTEMENT ORIENTÉ ET DANS LE BON ORDRE** je fais imprimer toutes les pages paires. Pour cette phase il me semble moins risqué d'opérer page par page, et vérifier à chaque tentative que deux A4 n'ont pas été "aspirés" par le mécanisme qui tracte les feuilles sous les têtes d'impression. Vous ne perdrez ainsi qu'une seule épreuve, alors que si vous engagez l'opération pour les huit feuilles ... c'est tout le paquet qu'il faudra entièrement réimprimer. Bien réfléchir quand on replace le paquet dans le bac de la machine, car les pièges sont nombreux. (*Inverser le haut et le bas, face sur le dessus qui n'est pas la bonne, pages entassées dans l'ordre incorrect ...*)

Puis, quand tout est imprimé, réaliser l'assemblage est relativement élémentaire :

- 1) Commencez par plier toutes les pages en deux. (*Et si possible du bon coté !*)
- 2) Trouvez une plaque de polystyrène ou du carton bien classique. (*Voir Fig.159*)
- 3) Positionnez les pages bien à plat et surtout bien les unes cadrées sur les autres.
- 4) Avec une petite agrafeuse qui accepte de s'ouvrir complètement mettre en place quatre "crochets".

ATTENTION : Quand on appuie sur l'agrafeuse il faut bien la tenir latéralement car elle veut se décaler sur les cotés. Du coup comme montré en Fig.160 l'agrafe est mal enfoncée et se plie. Quand une agrafe s'est tordue, la retirer avec un cutter et en placer une deuxième exactement au même endroit. Le deuxième essai sera le bon ...



- 5) Retourner le livret **sur un support rigide** et fermer les agrafes à la main avec un outil quelconque. Dans mon cas j'utilise une règle de section carrée comme montré sur la Fig.361 sur laquelle à peine visible on voit un coté de l'agrafe non encore replié.

Notez que pour vous faciliter la tâche les pages sont numérotées verticalement au centre pour repérer plus facilement l'ordre d'assemblage. Gutembérisez bien les amis ...

➤ Utilisation rationnelle du livret d'exploitation de JEKERT.

Après quelques tâtonnements et pas mal de "gros mots", vous avez enfin réalisé avec amour le petit manuel. Votre satisfaction est grande et c'est parfaitement mérité. Bien que se servir d'un document technique reste globalement évident, quelques petits détails peuvent vous faciliter les manipulations. Pour en tirer le meilleur parti, il me semble utile d'avoir à l'esprit certaines informations clef sur son agencement.

- Systématiquement le livret a tendance à s'ouvrir en son milieu. C'est la raison pour laquelle on trouve sur les pages 16 et 17 la liste des commandes pour les menus principaux. Tournez la page 17 et l'on trouve les menus principaux, et page 18 les **POSTURES** et les **DONNEES**.
- Chaque menu est présenté sous la forme d'un tableau séparé en deux zones. La partie colorée en jaune correspond aux commandes "de sens positif" (*Flèches dirigées vers le bas*) alors que la zone du bas concerne les commandes de sens rétrograde. (*Flèches orientées vers le haut*). Quand on ouvre un menu avec la touche clavier qui lui est affectée, on arrive à la première commande, celle qui est tout en haut dans la liste. Si la commande que vous désirez est en zone jaune, le plus rapide pour y arriver consiste à tourner le codeur rotatif dans le sens horaire. Si la commande est dans la zone des flèches orientées vers le haut il sera plus astucieux de tourner le bouton en sens antihoraire. Par exemple vous désirez l'item **Tir LASER ?** du menu **EXPLOITER**. Le plus rapide consiste à effectuer six indexages en sens antihoraire pour aboutir à cette fonction.
- S'ils sont concernés, les items sont complétés par des repères du genre **(S)**, **(F)** etc. Ces indications sont précieuses car elles précisent si la commande exige la **SÉCURITÉ**, s'achève par **FIN** ...
- Les divers protocoles sont optimisés. Ils tiennent compte des informations qui précèdent, assurent la sécurité pour la machine et peuvent vous éviter bien des erreurs de manipulation. Je ne peux que vous conseiller fortement de les prendre en considération.
- La couverture arrière du livret précise la signification des codes d'erreur. Pas besoin de chercher dans le livret pour savoir quel est l'origine d'un BIP d'alerte, il suffit de tourner le manuel. Quand on transmet des commandes, on peut avoir envie de savoir à quoi correspond un code de consigne. C'est également le cas lorsque l'on désire relire un programme enregistré. Faciles à trouver, tous les codes de consignes sont placés à la fin du livret, pas besoin de consulter la table des matières.
- La page 2 regroupe un certain nombre d'informations à ne pas oublier. Surtout, la page 3 est relative à la mise en service de la sonde et aux procédures de fin d'activité. Pour l'exploitation globale de la petite machine, la table des matières est placée en couverture pour pouvoir la consulter directement sans avoir à feuilleter.

Enfin, cette possibilité n'a pas été abordée, mais si vous consultez la page 19 explicite le protocole qui permet à tout moment de consulter l'historique des dix dernières commandes soumises au pupitre.

Chères lectrices, chers lecteurs, avec ce didacticiel, nous avons exploré les multiples possibilités offertes par l'exploitation de la petite sonde à partir d'un pupitre. C'était l'aspect opérationnel. Pour concrétiser ces manipulations il nous reste encore à matérialiser les circuits électroniques et construire un beau coffret. Cette phase créative va nous consommer du temps, certainement trop par rapport à nos espaces de loisir. Reste que c'est une activité valorisante avec à la clef beaucoup de satisfaction ...

*Je vous souhaite à toutes et à tous
agréable lecture et ... à bientôt dans le
troisième tutoriel.*

Amicalement : Nulentout.

Ben Mâmôa je ne trouve que
des pages paires et en triple dans
mon manuel. C'est normal ça ?