

## UNE NANO STATION MÉTÉO.

L'apprentissage du C++ d'Arduino à travers [la saga sur le mesurage](#) est ENFIN terminée. Le didacticiel est mis en ligne. Après deux années consacrées à ce projet, le moral est au beau fixe, et je vais enfin pouvoir faire autre chose. Par exemple reprendre le pilotage virtuel du B737, ou réembaucher à la NASA pour repartir dans l'espace, en Navette ou à bord d'Apollo. Bref, c'est la jubilation complète. Et en pleine sérénité joviale, **PAAAF** c'est le pépin, la tuile, la cata : Figurez-vous qu'une publicité automatique intrusive et non invitée arrive d'Internet et me fait découvrir par hasard la minuscule carte NANO Arduino.

- *C'est quoi ça la NANOtruc PAPPY ?*

- *Ben fiston, c'est comme une carte Arduino, sauf que c'est une Arduino !*

Aucune raison objective d'ajouter ce petit module à ma collection et de vous en parler, ce n'est qu'une version de plus. On a chevauché la UNO qui trône sur tous les établis, on a glissé vers le module fréquencemètre qui est un produit dérivé. On a côtoyé la BOARDUINO avec PICOLAB. Alors franchement pourquoi remettre le couvert, il y en a déjà assez. J'allais changer de page Internet, mais un petit détail turlupinait les neurones : Cette petite chose est réputée pouvoir remplacer une UNO tout en étant vraiment très petite et surtout ... on la trouve ce jour pour 3,67€ port compris et livrée à domicile avec son cordon de raccordement USB. À ce tarif, elle est moins onéreuse qu'un ATmega328 seul, et permet si l'on en croise la documentation, absolument tout ce que l'on peut faire avec une playine Arduino UNO.

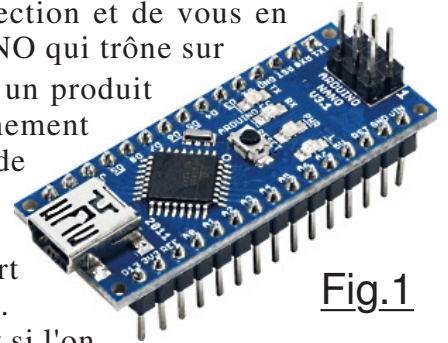


Fig.1

Compte tenu de son coût, il devenait particulièrement tentant de vérifier les affirmations relatives à ses performances. Des recherches sur la toile préviennent que des clones de cette carte issus de Chine ne sont pas compatibles pour dialoguer avec l'IDE. D'autres signalent que le cordon de liaison entre la prise miniUSB et le P.C. est pourrie. Ces deux points se sont avérés en partie exacts. Sur quatre exemplaires commandés, tous à la même adresse, deux cordons USB étaient défectueux ... poubelle. Effectivement, la ligne USB n'était pas reconnue par WINDOWS. Mais sur Internet, les fournisseur de ces clones mettent à disposition le DRIVER propre à leur circuit intégré de dialogue USB logé sur leur version. Dès que j'ai installé dans WINDOWS ce pilote, une pure formalité, immédiatement l'IDE a établi le contact et les téléversements ont été immédiats. Premier test : Un gros programme qui saturait PICOLABO avec texte et LOGO en EEPROM. Pas de problème, le fonctionnement sur OLED a été immédiat. Alors, vu le prix d'achat de cette toute petit chose, qui est plus que compatible avec une Arduino Uno, franchement, si ce n'est pas pour enficher en gigogne un quelconque SHIELD, à hésiter, et ce d'autant plus qu'elle est encore plus petite que BOARDUINO, alors pour "miniaturiser" elle confine au parfait. mon sens il ne faut pas



Fig.2

Crise sur la gâteau, non seulement elle met à notre disposition l'intégralité des ressources logicielles et matérielles d'une carte Arduino UNO, mais le boîtier de son processeur permet de récupérer deux entrées analogiques de plus. OUI, oui, vous avez bien lu. Non seulement elle est plus petite, elle fait tout ce que peut proposer UNO, mais en promotion *on dispose de deux lignes d'entrées de plus*. Ces deux lignes ne peuvent pas fonctionner en sorties, ni en entrées binaires. Elles sont uniquement configurées en entrées analogiques, mais rien n'interdit de réserver les six entrées classiques pour du fonctionnement binaire si on le désire.

J'ai donc estimé que ne pas vous en parler était comme botter en touche, alors on recommence. Le but de ce dernier volet n'est plus d'apprendre le langage C++, mais uniquement de mettre en service cette petite carte dans une application ludique pour en démontrer les aptitudes opérationnelles.

### Un peu de palable préambulesque.

Ajouté à la saga sur le mesurage, alors qu'il n'était absolument pas prévu, ce dernier volet ne vise plus à apprendre certaines finesses du langage C++ d'Arduino. Son seul but : Vous fournir un prétexte "grand public" pour concrétiser une application ludique utilisant un clone de ces minuscules cartes Nano Arduino. Résolument tourné vers la simplicité, nous allons réinvestir les développements relatifs à la mesure de la température et de l'hygrométrie abordés en détails lors de l'utilisation de l'ENREGISTREUR À DEUX VOIES.

Tous les marins savent qu'à bord de leur bateau il vaut mieux avoir un baromètre qu'un thermomètre pour anticiper l'arrivée d'un "coup de tabac". Si vous préférez, une installation qui affiche se targue du titre "Météo" doit au minimum être capable de fournir la valeur de la pression atmosphérique. Outre la mesure de la température et celle de l'humidité de l'air, notre petit appareil va donc inclure un BAROMÈTRE et bien entendu UNE SENTINELLE pour nous réveiller la nuit !

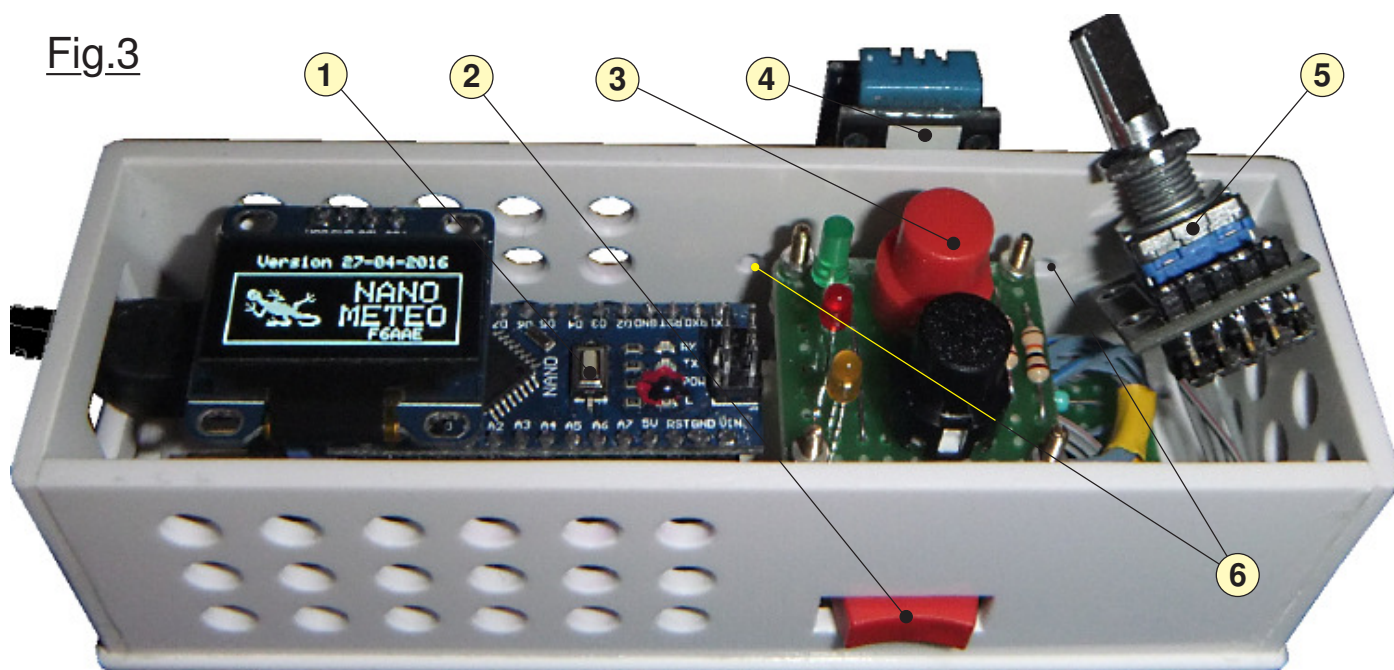
Savez-vous que depuis les débuts de l'aviation, pour mesurer leur altitude les pilotes utilisent un ... baromètre ? Le concept est assez simple. Tous les physiciens savent que la pression de l'air résulte du poids de l'atmosphère située au dessus. Donc, quand on s'élève, "au dessus" il y en a de moins en moins. Par voie de conséquences, la pression environnante diminue. Si l'on établit la courbe d'évolution de la pression en fonction de la hauteur, réciproquement, avec la valeur barométrique on peut en déduire l'altitude. *Génial de simplicité non ?* Comme la petite station météo sera conçue pour présenter de toutes petites dimensions, pourquoi ne pas prévoir la fonction ALTIMÈTRE pour agrémenter vos promenades pédestres et ainsi évaluer les dénivelés gravis avec courage quand il fait très très chaud et qu'il n'y a pas d'arbre pour vous protéger ? C'est décidé, notre NANO STATION MÉTÉO sera également un altimètre.

**Certainement pas !** Contrairement à l'exclamation "*Génial de simplicité non ?*", mesurer une altitude reste d'une complexité phénoménale. Vous savez tous que le Mont Blanc culmine à 4807m. Ben oui, c'est à l'école primaire que l'on apprend ça. Mais ... mesurée par rapport à quoi ? L'Instituteur se garde bien de le préciser. Sans entrer dans les détails, c'est par rapport à la mer la plus proche qui sert de référence, donc la Méditerranée lorsqu'elle est au niveau moyen de la marée et que l'atmosphère est standard etc etc etc. Et oui, la pression atmosphérique pèse sur l'eau. Plus la pression est élevée, plus elle fait descendre le niveau. S'il y a une dépression, c'est l'inverse, le niveau de la mer monte. Bref, mesurer une altitude n'a rien d'anodin. Par ailleurs, un pilote en fonction des phases du vol changera le mode d'utilisation de son altimètre. Pour se poser, il préférera connaître la HAUTEUR par rapport à la piste, alors que durant le vol près du relief il vaudra mieux connaître l'altitude, pour la comparer à celle des sommets situés dans les environs. *(Non, pas question ici d'ajouter le mode STANDARD, ce n'est pas un cours de pilotage !)*

Bien que notre "jouet" soit capable d'afficher correctement une altitude, **il n'est absolument pas agréé pour le pilotage d'un aéronef**. Donc pas question de vous en servir pour faire du vol sans visibilité ou poser la Navette. Considérez-le que comme **un gadget pour randonneur**, rien de plus.

Transocéanic 28 Alpha tango, Passons moins 5000 pieds QFE et demandons à faire surface pour l'atterrissage.

Fig.3



### Présentation sommaire.

**R**egardant la Fig.2 sans attention particulière, on a la furieuse impression d'observer PICOLAB. Le coffret est réalisé avec des dimensions strictement identiques à celle du petit laboratoire pour des raisons esthétiques et d'homogénéité. On retrouve le petit clavier en **3** (Fig.3) qui supporte également trois petites LED de service. (Ces dernières ne protègent plus des entrées, mais complètent l'exploitation du clavier et celle du centre en rouge est relative à la SENTINELLE.)

- Hé bé Totoche, c'est vraiment pas la peine d'utiliser un MICRONANOPICOtrucmachin qui oblige à lire la notice en chinois pour finalement faire aussi grand !
- Aussi grand mais ... avec un truc qui tourne en plus, comme les nuages autour d'une dépression, que le baromètre il en prend le tournis.

**E**ffectivement, c'est précisément la taille réduite du NANO Arduino en **1** qui permet de "tenir" les dimensions tout en intégrant le capteur rotatif **5**, qui, nous l'avons constaté avec PICOSYNTHE, octroie à l'appareil qui l'utilise un agrément d'utilisation incontestable. Du reste, à l'usage vous allez découvrir qu'arriver à conserver des dimensions totalement équivalentes à celle du laboratoire confine à un réel défi. Avant de vous engager à faire aussi réduit, vérifiez bien la faisabilité de l'entreprise par rapport à vos méthodes habituelles de "fabrication".

En **4** sont visibles les deux capteurs amovibles dont le support est enfiché sur un connecteur orienté vers l'arrière de l'appareil. Contrairement à ce que l'on observe sur la Fig.2 le capot de protection de ces capteurs n'est pas en place, la station météo étant en cours de réalisation. Nous verrons plus avant que ces capteurs peuvent être déportés loin de la station quand cette dernière est utilisée pour mesurer les paramètres météorologiques. Par contre, emportée dans le sac à dos pour mesurer des dénivelés, le petit appareil doit être compact. Dans ce cas, les capteurs sont enfichés directement en face arrière comme sur la Fig.3 sur laquelle en **6** on distingue les deux trous de fixation du petit protecteur. Notez pour conclure cette présentation rapide, qu'en **2** se trouve un inverseur à bascule à deux positions et rappel central. Sa présence facilite considérablement l'utilisation, car avec les seuls boutons poussoir du clavier, l'organisation des protocoles d'exploitation aurait été scabreuse et configurer ou interroger notre petite grenouille informatique infiniment moins agréable. Loger cet inverseur en "face avant" est un impératif pour conserver les dimensions du boîtier.

Hé Pépé, normalement, PICO c'est plus petit que NANO en principe ?

Ben pas forcément fiston, PICO et NANO ont exactement le même nombre de lettres par exemple.



## Le circuit imprimé principal.

Autant arriver à concevoir un petit circuit utilisant les techniques largement décrites pour concrétiser PICOLAB est indigeste, autant se simplifier cette activité par logiciel peut s'avérer salvateur. Tout particulièrement, l'affectation des diverses broches d'Entrées / Sorties permet d'éviter de trop nombreux croisements de pistes cuivrées, des ponts de câblage etc. Il suffit dans le programme source de changer la valeur d'une directive **#define** et l'entrée ou la sortie désirée "saute" immédiatement d'une position "géographique" sur le circuit à une autre infiniment plus facile à exploiter.

Quand NANO MÉTÉO a été développée, la logique a consisté à reprendre "intégralement" les affectations des E/S qui définissaient l'organisation de PICOLAB. Ainsi toutes les séquences de programme relatives à OLED, à la mise en œuvre des capteurs étaient directement "copiables". Puis, pour le plaisir d'utiliser les deux nouvelles entrées analogiques **A6** et **A7**, et vérifier leur comportement analogique banal, les deux contacts de l'inverseur à bascule et rappel central y étaient branchés.

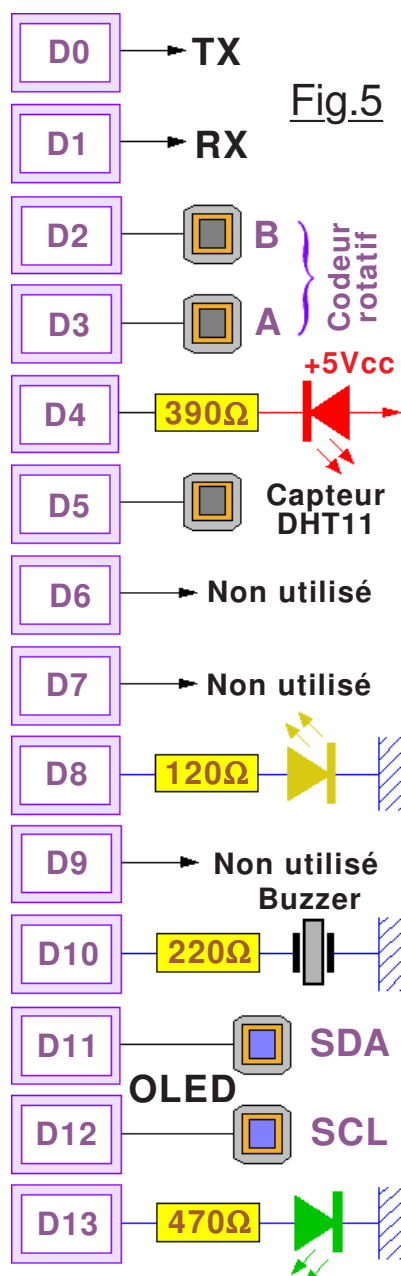


Fig.5

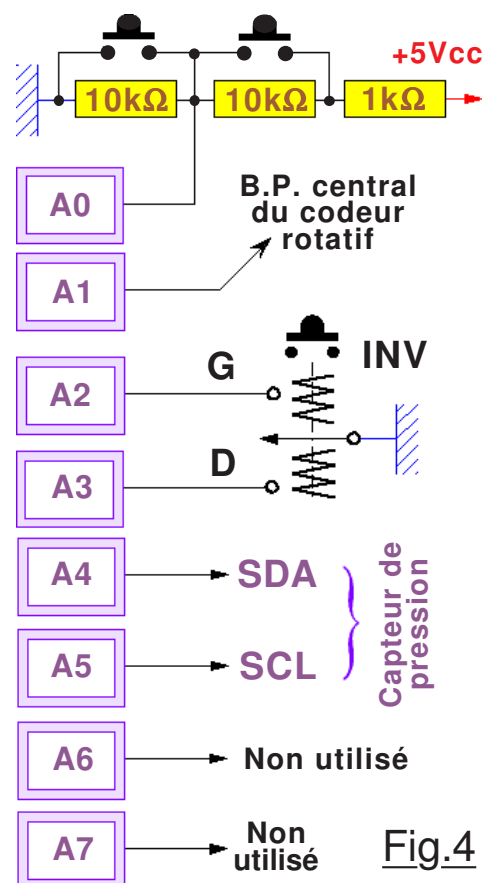


Fig.4

Programme complet globalement au point, le moment de passer à l'étude du circuit imprimé principal est arrivé. Cette phase est particulièrement délicate à conduire, car de nombreuses contraintes viennent compliquer à foison l'élaboration du projet. Sur l'ordinateur, on développe simultanément le dessin du circuit imprimé principal, celui du clavier pour les deux boutons poussoir, **et surtout le dessin en trois vues et à l'échelle du coffret avec les divers éléments qu'il devra contenir**. Sur le bureau on dispose également de PICOLAB et d'un réglet. On mesure, on tergiverse, on découpe du carton pour voir si l'intégration sera possible. L'INTÉGRATION : Le maître mot dont dépend l'issue de sortie de notre projet. Cet aspect d'un développement a été largement débattu dans le didacticiel, inutile d'alourdir le propos. Sachez que c'est durant cette phase que pour des raisons de miniaturisation, il a été décidé de placer l'inverseur à bascule sur "la face avant", vocable qui désigne celle qui est dirigée vers nous.

Cette décision oblige à approvisionner un inverseur "spécial", mais s'impose si l'on veut respecter le volume "du cahier des charges". L'emploi de ce petit appareil de mesures en altimètre autonome impose de placer les capteurs sur la face arrière. Toutes ces contraintes ont abouti à changer l'affectation des Entrées/Sorties pour faciliter l'élaboration du circuit imprimé. On tasse, on empile, on compacte ... sans pour autant rendre impossible le soudage des composants, l'extraction des circuits pour MAINTENANCE ou évolution. Bref, cette étude engloutit des heures, mais en sortir victorieux n'a que plus de charme. Aussi, si je vous assure que le prototype a confirmé le bienfondé des solutions proposées dans ce volet de la saga, vérifiez qu'elles restent compatibles avec "vos savoirs faire". Quand à imaginer plus petit ... franchement j'en doute !



Levons le voile sur le schéma électronique qui complète la mise en œuvre de la petite carte NANO Arduino. Les Fig.4 et Fig.5 ne sont pas autre chose que des copies éhontées du schéma général de PICOLAB à peine modifiées pour traduire les petites variantes propres à notre nouvelle application ludique. Notez au passage que l'utilisation d'un écran OLED bicolore qui permet de se passer d'un opérateur SN7400 est impératif, il n'y a vraiment pas la place pour loger ce composant complémentaire dans notre réalisation. Il importe donc durant les essais "sur plaque de prototypage" de vérifier la compatibilité du modèle approvisionné. La Fig.4 détaille l'affectation des entrées.

Pour **A4** et **A5** nous n'avons pas le choix. La bibliothèque [Adafruit\\_BMP085.h](#) utilise de façon standard ces deux broches pour piloter la ligne I2C du capteur de pression atmosphérique. (*Cette bibliothèque ne nous laisse pas la possibilité de réaffecter les broches.*)

Répartir les sorties, comme indiqué sur la Fig 5 ne présente que très peu d'originalité. **D13** reste affectée à la LED Arduino qui sur le circuit du clavier est de couleur verte. Idéalement positionnées sur la carte NANO Arduino, les sorties **D11** et **D12** continuent à piloter la ligne I2C de l'afficheur OLED. Complété par la Fig.6 on constate que l'alimentation des deux picots **ALIM** sur le connecteur HE14 n'est pas reliée directement à **GND** et **+5Vcc**. Sur le circuit imprimé principal ces broches vont sur deux petits connecteurs supportant chacun un "strap". Ainsi il sera facile d'orienter les polarités en fonction de l'afficheur OLED approvisionné.

Comme on dispose d'un nombre de sorties dépassant largement nos besoins, et que la conception du circuit imprimé principal ne l'exige pas, **D0** et **D1** ne sont pas utilisées, laissant totalement libre le téléportage des programmes sans avoir à isoler ces lignes des circuits de l'application.

On peut s'étonner d'avoir choisi pour la LED de la sentinelle **D4** une logique négative. La facilité d'implantation des composants pour le circuit imprimé du petit clavier a présidé ce choix. Pour les autres broches, la répartition est dictée pour faciliter au maximum l'étude du circuit imprimé principal.

### Affectation des Entrées / Sorties.

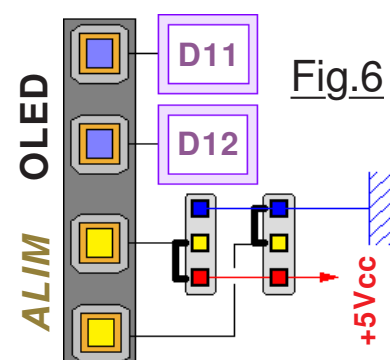


Fig.6

AFFECTATION DES ENTRÉES / SORTIES	
Broche	Utilisation
D0	Réservée en standard pour la liaison série USB. (RX)
D1	Réservée en standard pour la liaison série USB. (TX)
D2	Sortie <b>B</b> du codeur incrémental rotatif.
D3	Sortie <b>A</b> du codeur incrémental rotatif.
D4	Pilotage de la LED et de l'ALARME DE TEMPÉRATURE.
D5	Pilotage sur ligne série "bilatérale" du capteur d'humidité DHT11.
D6	Broche non utilisée.
D7	Broche non utilisée.
D8	Pilotage de la LED de service jaune.
D9	Broche non utilisée.
D10	Pilotage du petit BUZZER.
D11	Dialogue I2C avec l'afficheur OLED : Ligne SDA.
D12	Dialogue I2C avec l'afficheur OLED : Ligne SCL.
D13	Pilotage de la LED Arduino 13.
A0	Gestion des deux B.P. de changement de fonction et de choix des options.
A1	Bouton poussoir central du codeur incrémental.
A2	Contact latéral <b>Gauche</b> de l'inverseur à bascule à rappel mécanique au neutre.
A3	Contact latéral <b>Droit</b> de l'inverseur à bascule à rappel mécanique au neutre.
A4	Dialogue I2C avec le capteur de pression atmosphérique : Ligne SDA.
A5	Dialogue I2C avec le capteur de pression atmosphérique : Ligne SCL.

Fig.7

Pour des raisons de mise en page de ce document les entrées analogiques **A6** et **A7** non utilisées ne sont pas listées.

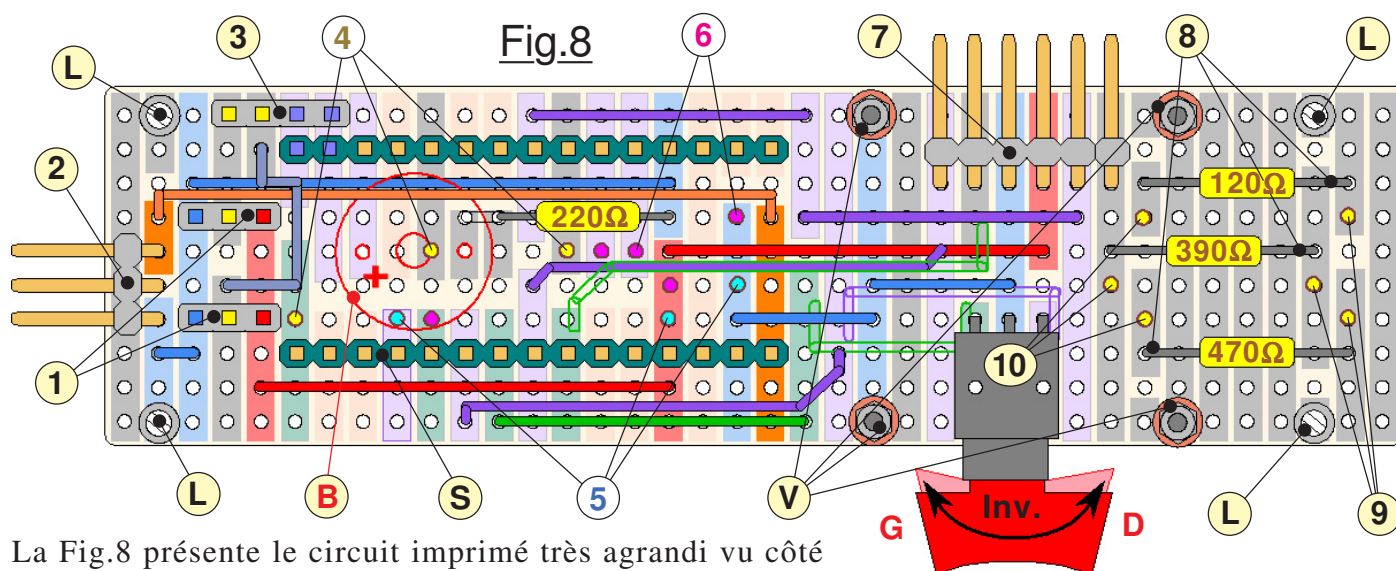
### La carte électronique qui supporte le microcontrôleur.

Contrairement à un développement à partir de zéro où il vaut mieux commencer par placer des composants sur une plaquette de connections rapides pour élaborer progressivement les circuits qui seconderont NANO Arduino, vous pouvez opter pour commencer directement par la réalisation matérielle, motivation d'autant plus justifiée que l'on peut très facilement téléverser les programmes directement sur site. Toutefois, avant d'engager la concrétisation matérielle, prenez au moins la précaution de vérifier que votre afficheur OLED est compatible et n'imposera pas l'intervention d'un opérateur NAND sous forme d'un circuit SN7400.

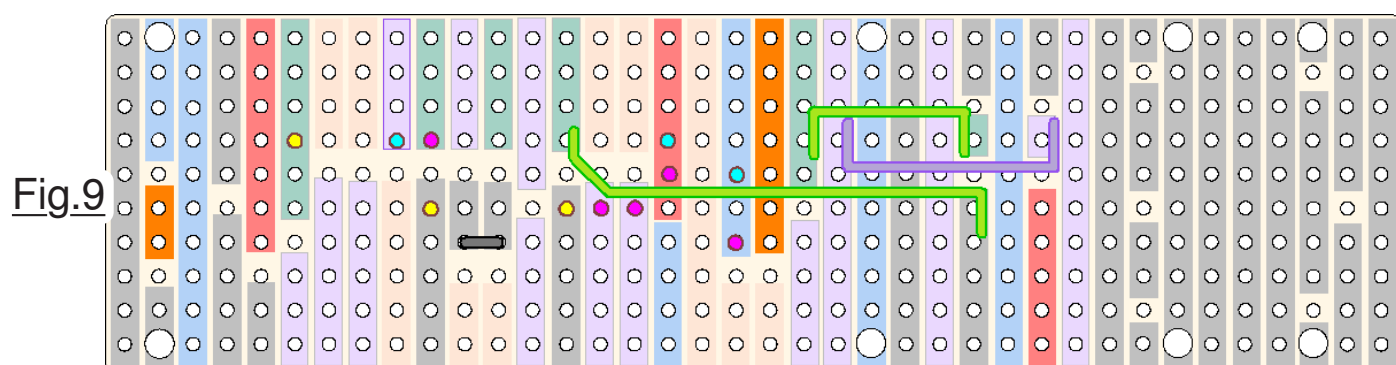
Au risque de rabâcher, surtout n'abusez pas de la miniaturisation à outrance. Ne présumez-pas de vos aptitudes d'horlogers. Le plus important, c'est d'arriver à finaliser votre petite station météo sans exaspérations ni énervements. Il s'agit d'un loisir de détente, il faut se faire plaisir.

Dans ce qui suit, je vais vous présenter au mieux les petits détails du prototype qui, une fois n'est pas coutume, n'a révélé aucune mauvaise surprise durant sa réalisation.

Naturellement, nous allons commencer par le circuit imprimé principal. Comme pour tous les croquis et les images qui concernaient les variantes des petits laboratoires de mesure, les macrophotographies et les dessins des circuits imprimés sont trompeurs. Ils donnent des fausses impressions de grandeur. Tous les éléments sont petits, presque minuscules pour certains. Dans le boîtier les espaces sont calculés au plus juste. La solution globale toutefois ne rend aucune manipulation scabreuse. Par contre, un minimum de méthode s'impose. C'est parti, concrétisons :



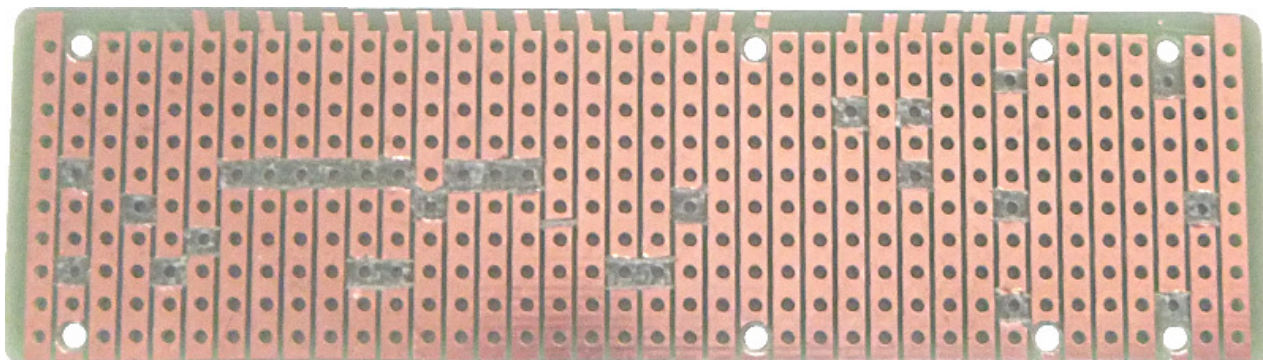
La Fig.8 présente le circuit imprimé très agrandi vu côté composants avec les conventions déjà adoptées dans les chapitres précédents. Pour ceux qui prendraient le train en marche, c'est à dire que non directement concernés par les petits laboratoires sont passé directement à ce volet, le dessin montre les pistes cuivrées comme si le support était transparent. Les bandes de cuivre sont coloriées facilitant l'interprétation de leur affectation et mettant en évidences certaines coupures de pistes peu repérables. Les fils rigides servant de ponts sont coloriés en fonction de leur utilisation. Bleu pour **GND**, rouge pour le **+5Vcc**, orange pour l'alimentation extérieure sur le petit connecteur HE14 repéré **2**.





Comportant trois broches pour mieux supporter l'élément femelle qui y sera branché, sa broche centrale n'est pas soudée pour "augmenter l'isolement électrique" entre les deux extrémités. Les ponts violets sont relatifs à des entrées ou des sorties. Si des ponts de liaison sont tracés en mode filaire, c'est qu'ils sont placés sur le dessous coté cuivre. En **L** nous avons les quatre trous de liaison

Fig.10



entre le circuit et le coffret. Comme pour PICOLAB, les petits boulons sont de diamètre  $\Phi 2$  mm. En **V** on trouve les vis qui supportent le petit clavier. C'est entre ces boulons solidaires du circuit imprimé et isolés coté pistes cuivrées par les rondelles colorées en rouge, que l'on intercale l'Inverseur à bascule et le petit connecteur HE14 à six broches en **7** qui va assurer la liaison électrique amovible avec les capteurs. En **S** se trouve le connecteur femelle HE14 qui supporte le petit module NANO Arduino. Le type utilisé est un peu haut pour laisser entre les deux lignes de contacts assez d'espace verticalement pour loger le Buzzer **B** représenté en mode filaire. En **1** nous avons les deux

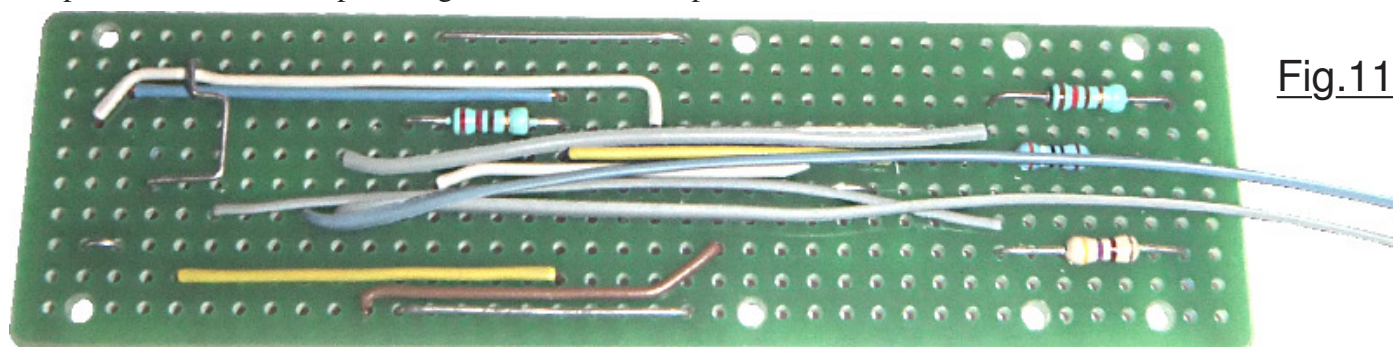
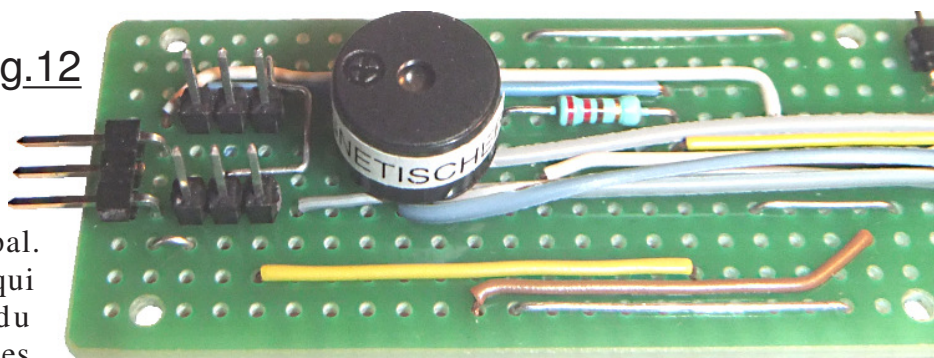


Fig.11

petits connecteurs HE14 mâles recevant les straps qui répartissent sur le connecteur **3** de l'afficheur OLED les polarités de l'alimentation. Les couleurs affectées aux broches sont les mêmes que celles utilisées sur la Fig.6 avec en jaune celles pour **ALIM**. Comme on peut le constater sur la Fig.8 des liaisons sont soudées sous le Buzzer **B**. Il est un peu surélevé du circuit imprimé et n'est mis en place que lorsque les fils souples tels que ceux repérés **4** sont soudés coté pistes. Les trous d'implantation des liaisons filaires sont représentés par les petits cercles sur le dessins. Les trois colorés en jaune en **4** relient les trois trous également jaunes en **10**. Ces trois liaisons assurent le pilotage des trois petites

LED de  $\Phi 3$  mm du clavier. Comme ce dernier est trop petit pour contenir les trois résistances de limitation de courant **8**, ces dernières sont "déportées" sur le circuit principal. Se sont les trois fils souples **9** qui vont vers le petit module du clavier. Les autres liaisons souples

Fig.12



qui vont jusqu'au clavier sont repérées par les petits cercles bleus **5**. Les cinq petits fils souples soudés tels que **6** vont vers le connecteur HE14 du codeur incrémental rotatif. La Fig.9 expose les particularités du coté cuivre, à comparer avec la Fig.10 qui présente la plaquette coté cuivre quand les bandes conductrices ont été correctement séparées. La photographie a été un peu retouchée pour mettre plus en évidence les trous de passage des diverses vis.



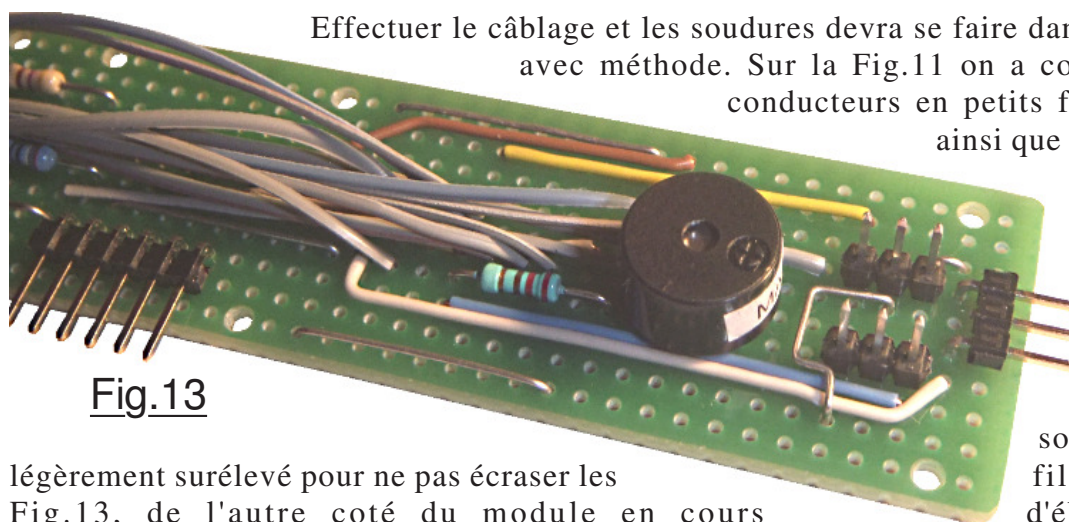


Fig.13

légèrement surélevé pour ne pas écraser les Fig.13, de l'autre côté du module en cours

nettement que le buzzer est situé au-dessus des deux petits fils rigides. Quand vous assemblez celui qui est coudé cinq fois et qui saute les deux fils bleu et blanc, intercalez un petit carton rigide entre les trois conducteurs pour ne pas risquer de faire fondre l'isolant au moment de la soudure.

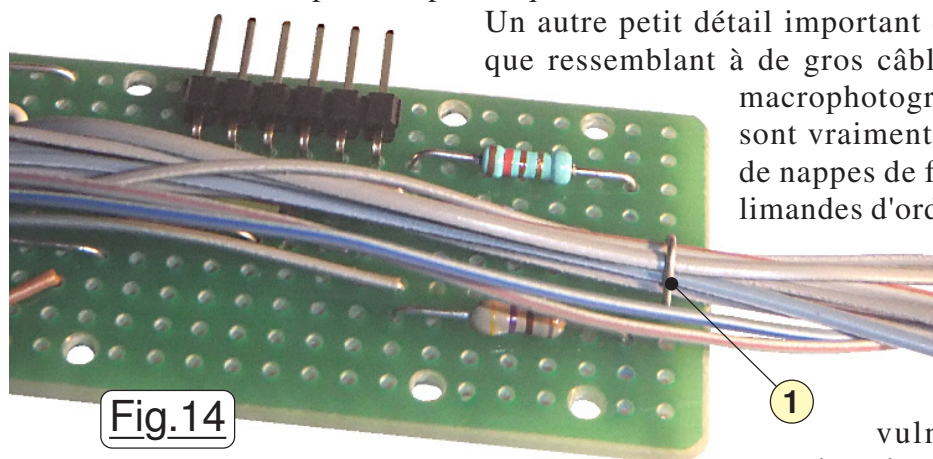


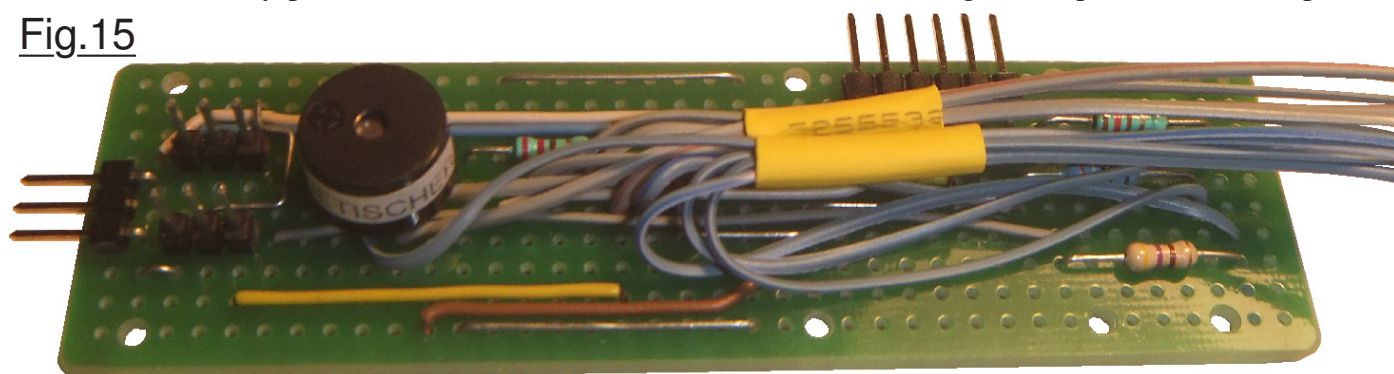
Fig.14

Un autre petit détail important doit attirer votre attention : Bien que ressemblant à de gros câbles EDF véhiculant 500A sur les macrophotographies, les fils de liaison souples sont vraiment très fins puisqu'ils sont détachés de nappes de fils employées pour constituer des limandes d'ordinateurs. On passe son temps coté pile pour insérer les fils et les composants, puis coté face pour réaliser les soudures. Au raz des implantations, ces longs fils souples sont particulièrement vulnérables à la torsion. Pour les protéger durant ces nombreuses manipulations,

au fur et à mesure de l'avancement des opérations, les torons sont provisoirement tenus à l'extrémité droite en 1 par du petit fil rigide replié coté cuivre pour l'immobiliser en place. (Voir la Fig.14)

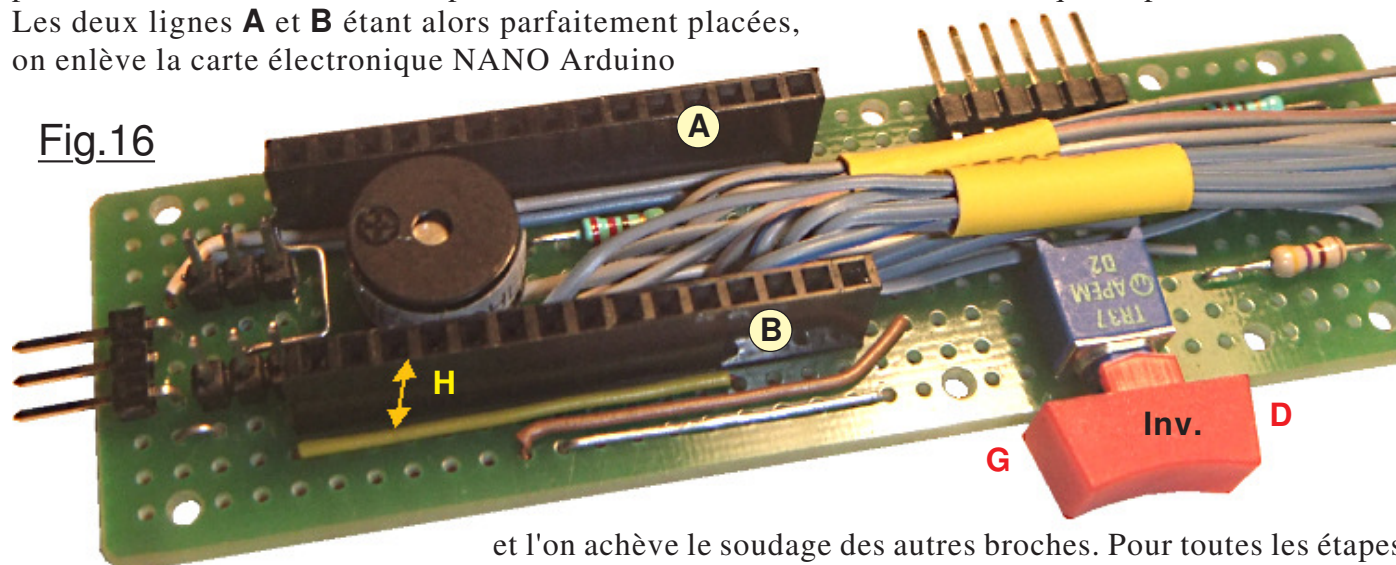
**C**ourage, le bébé se porte bien. On continue résolument sans s'énervier, avec calme et bonne humeur. Quand tous les fils souples sont soudés, qu'il ne manque plus que les connecteurs HE14 supports de l'afficheur OLED et du microcontrôleur, ainsi que le petit inverseur latéral, on prend une quelconque sonnette ou un ohmmètre et l'on réuni en deux torons bien distincts ceux qui iront vers le clavier et ceux qui seront branchés sur le codeur incrémental rotatif. Puis, comme le montre la Fig.15 on les sépare mécaniquement en les "compactant" par de la gaine thermo-rétractable. Notez que ces lignes sont coudées assez proche du buzzer pour leur octroyer de la souplesse ce qui sera impératif lors des opérations de maintenance ou d'améliorations éventuelles. Souder l'inverseur latéral ne présente strictement aucune difficulté. Pour les connecteurs restant non plus du reste, en revanche il faut s'y prendre avec méthode car les trois éléments exigent un positionnement précis.

Fig.15



L'assemblage des deux lignes de contacts **A** et **B** qui supportent la carte NANO Arduino doit présenter une orientation verticale soignée. Pour assurer la correspondance parfaite avec les picots du connecteur de la carte électronique, personnellement je commence par en vérifier l'alignement rigoureux des trente broches. Puis j'insère les deux connecteurs **A** et **B** qui sont ensuite positionnés sur notre circuit imprimé. On retourne le tout et l'on soude les quatre picots d'extrémité. Les deux lignes **A** et **B** étant alors parfaitement placées, on enlève la carte électronique NANO Arduino

Fig.16

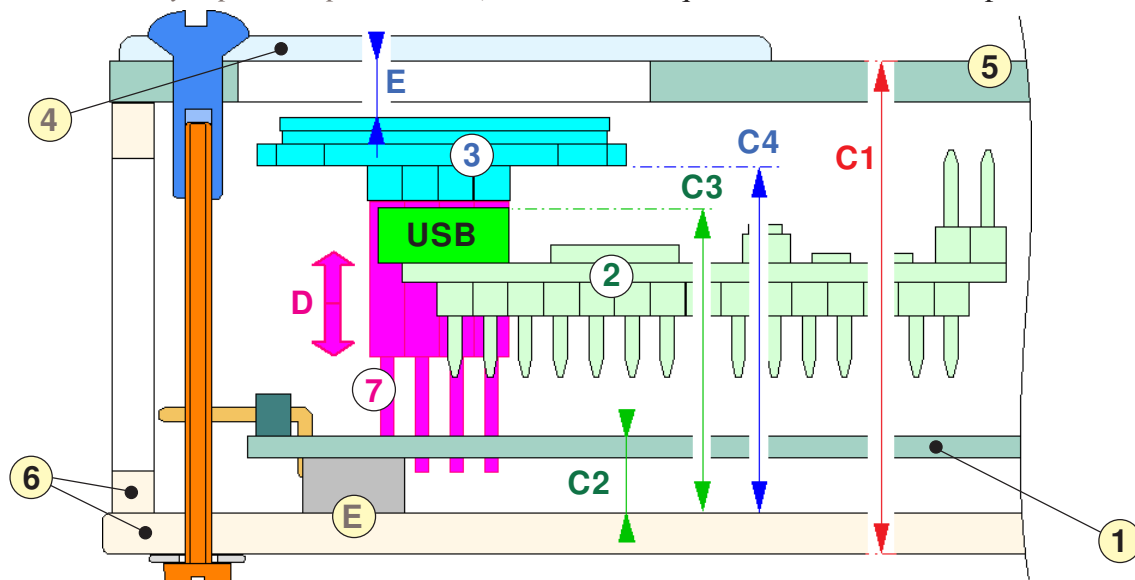


et l'on achève le soudage des autres broches. Pour toutes les étapes de la réalisation de ce circuit, chaque soudure est vérifiée avec une loupe à fort pouvoir grossissant pour s'assurer qu'il n'y a pas de contact interdit avec les éléments voisins, et que la soudure présente un aspect visuel correct. Un contrôle permanent à chaque étape est le gage d'un fonctionnement "immédiat" et fiable. Les deux lignes **A** et **B** de connecteurs femelle HE14 présentent une hauteur **H** moyenne. Il faut impérativement que la petite carte électronique entièrement enfichée, il reste entre ses composants situés sur le dessous et le Buzzer une place suffisante. Les torons de fils aussi doivent bénéficier d'un certain confort. Pour vous donner une idée plus précise, la hauteur **H** des éléments sélectionnés pour équiper le prototype fait presque 9mm.

#### Le connecteur HE14 qui supporte l'afficheur OLED.

Comme c'est la hauteur à laquelle devra se trouver l'écran qui conditionne les dimensions et la position verticale du petit connecteur à quatre contact, il faut au préalable avoir une idée assez précise des dimensions du coffret et de la façon dont sera supporté le circuit imprimé principal. (Et encore, quand j'écris *Idée assez précise*, ce n'est absolument pas suffisant.) Avec la Fig.17 nous allons pouvoir énumérer les critères qui permettent d'aboutir à un résultat totalement satisfaisant. Supposons le problème comme étant résolu. (C'est mon Prof de Math qui utilisait ce vocable étrange, car si c'est résolu ... il n'y a plus de problème !) C'est à dire, que suite à des études particulièrement

Fig.17

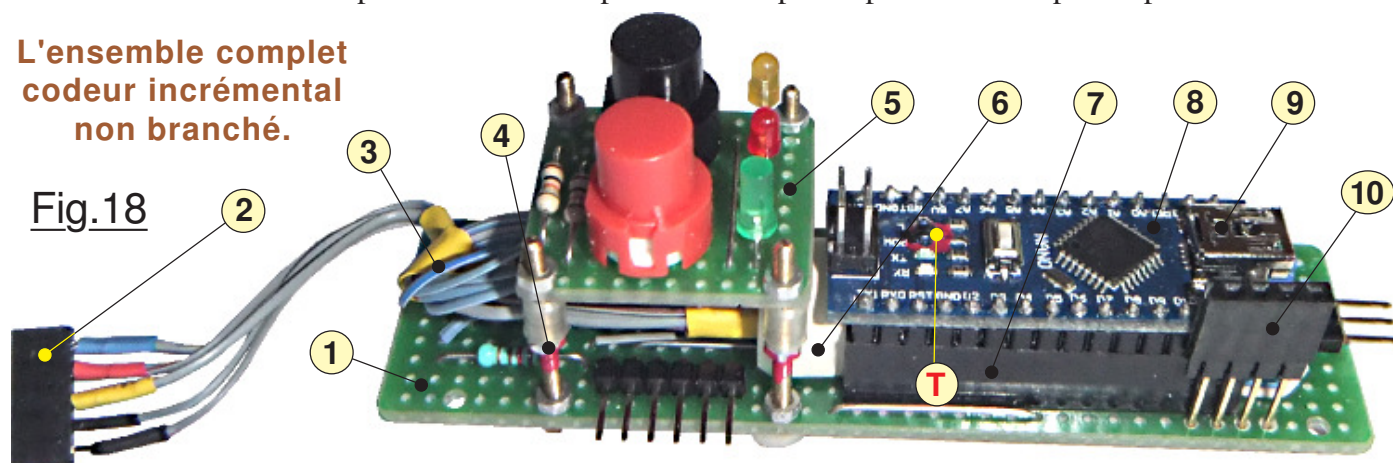




fouillées, nous avons déterminé toutes les dimensions de NANO MÉTÉO. Le coffret **6** est alors entièrement déterminé, avec en **1** le circuit imprimé principal, et en **4** la petite vitre constituée d'une plaque en matière thermoplastique totalement translucide. Pour pouvoir observer l'afficheur OLED **3** colorié en bleu clair dans les meilleures conditions possibles, on désire que l'écart **E** soit le plus faible possible sans pour autant que **3** ne touche le couvercle **5**.

C'est la hauteur de l'entretoise utilisée **E** qui conditionne la cote **C2**. La cote **C3** est celle que l'on mesure entre l'appui des entretoises **E** et le dessus de la minuscule prise **USB** coloriée en vert foncé de la carte NANO Arduino **2**, représentée elle en vert pastel. Sur ce dessin, **2** semble léviter comme par magie. Pour des raisons de clarté, les connecteurs HE14 **A** et **B** qui supportent **2** ne sont pas représentés. Leurs hauteurs **H** conditionnent la valeur de **C3**. Il ne faut surtout pas que les éléments situés sur le dessous de l'afficheur **3** ne touchent la prise **USB**. Un écart suffisant conditionne alors la cote **C4**. Ce sont tous ces critères qui sur un dessin effectué à l'échelle des dimensions permettent de trouver le meilleur compromis pour la cote **C1**. *(En pratique, il ne faut surtout pas oublier la présence des boutons poussoir du clavier dont la hauteur devra pouvoir s'ajuster correctement, mais ... oublions pour le moment.)* Quand la solution définitive est trouvée, on constate qu'il faut utiliser un connecteur HE14 **7** à broches longues. Personnellement j'ai réalisé les éléments latéraux et le dessous du coffret. Puis, circuit **1** immobilisé sur ces entretoises **E**, il devenait assez facile de positionner avec précision **7** par déplacement **D** pour optimiser l'écart **E**.

### L'ensemble complet codeur incrémental non branché.



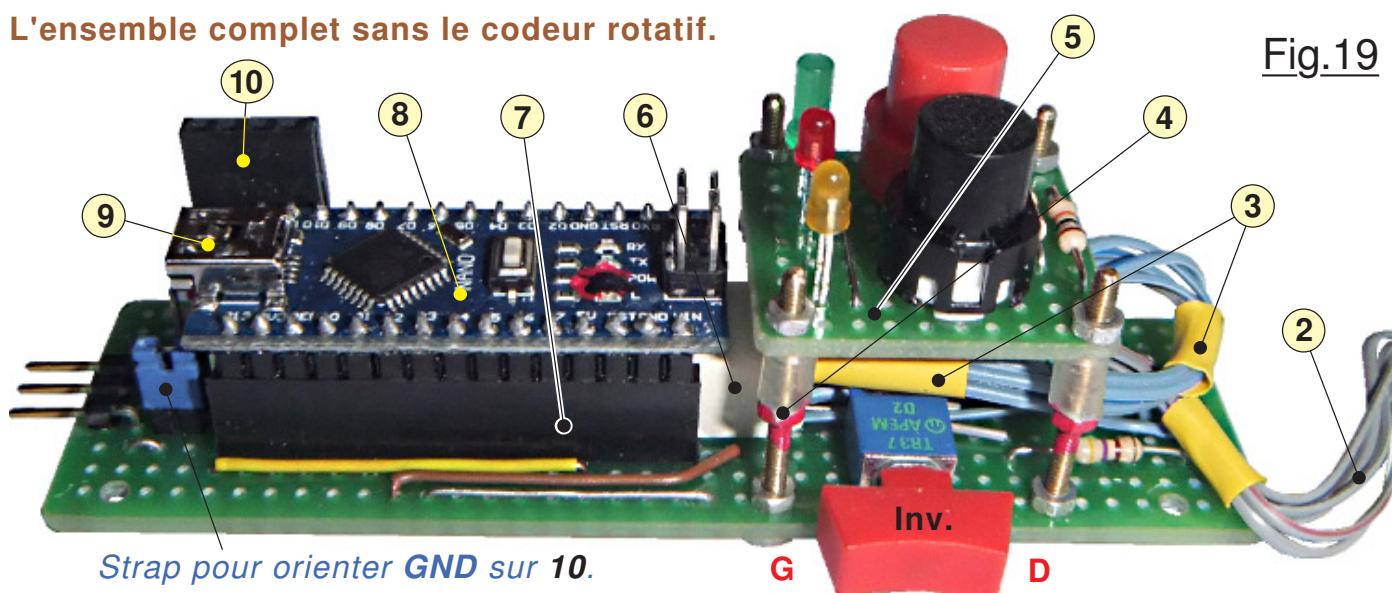
**CONCLUSION :** Pour pouvoir positionner parfaitement le petit connecteur qui supporte l'afficheur OLED, il faut avoir en partie réalisé le coffret, ou tout au moins disposer d'un dessin précis de ce dernier pour pouvoir y mesurer les dimensions critiques. Pour vous aider à franchir cette étape du projet, vous trouverez les dessins rigoureux du prototype. Il vous suffira de transposer quelques cotes pour rendre compatible ces dernières avec les contraintes issues de "vos technologies".

Anticipant un peu sur les descriptions qui vont suivre, la Fig.18 présente l'électronique principale entièrement terminée avec en **1** le circuit imprimé de base et en **2** le connecteur qui se branche sur le codeur rotatif incrémental. En **3** on distingue le toron de fils souples qui est soudé au circuit imprimé **5** du petit clavier. Les trois petites LED sont bien visibles. En **4** les écrous qui permettent de positionner avec précision **5** en hauteur. Ces écrous sont freinés une fois ajustés par du vernis à ongle rose. Les écrous **4** ne sont pas directement en contact avec **5**, mais séparés par les petites entretoises isolantes transparentes. En **6** on observe un petit morceau de carton qui s'enroule au dessus de tous les fils souples pour les protéger de tout pincement quand on enfiche la carte NANO Arduino **8** sur son connecteur **7**. En **9** est bien visible la prise **USB** qui ne doit pas toucher au dessous de l'afficheur OLED. Enfin en **10** le connecteur HE14 qui reçoit l'afficheur OLED. Il est manifeste que ce composant est de grande hauteur. Ses broches ont été écourtées une fois qu'il est soudé exactement à la bonne hauteur sur **1**, car ces broches sont initialement deux fois plus longues. En **T** on note la présence d'un "barbouillage rose et bleu foncé". C'est une "peinturlure" de vernis à ongle pour masquer la LED rouge de présence d'alimentation. Comme elle est bien trop lumineuse et interfère avec l'écran d'affichage, elle a été entièrement cachée. Le vernis à ongles rose n'était pas assez efficace, il a été recouvert d'un autre gribouillis en vernis bleu foncé.



## L'ensemble complet sans le codeur rotatif.

Fig.19



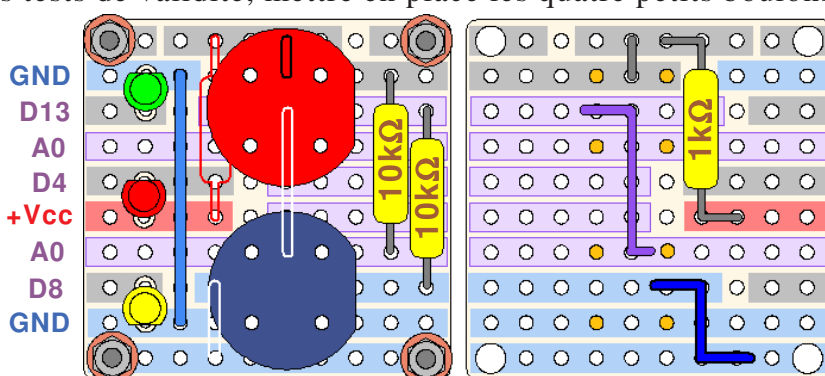
La Fig.19 présente l'ensemble coté **Inverseur** à bascule. Les repères sont identiques à ceux répartis sur la Fig.18 pour pouvoir comparer plus aisément. Avant de placer **8** sur le connecteur **7** il faut mettre en place les deux petits straps qui orientent l'**ALIM**entation sur **10**. Cette vue montre clairement que la hauteur de passage pour les deux torons sous le circuit imprimé **5** du clavier est très confortable. Le toron **2** est de longueur idoine pour permettre d'écarter suffisamment le couvercle et ainsi accéder au petit connecteur pour le débrancher ou le rebrancher. Enfin, s'il vous semble que la zone du haut de la petite LED verte est plate, vous n'avez pas la berlue. C'est uniquement parce qu'en cette couleur et en diamètre de 3mm je n'avais pas de type classique terminé en demi-sphère.

## Le petit circuit imprimé du clavier.

**C**opieusement inspiré de celui de PICOLAB il hérite de ses faibles dimensions. Comme vous pouvez le vérifier sur la Fig.19 sa surface est à peine suffisante pour y répartir les deux boutons poussoir et les trois LED. On arrive à "caser" les deux résistances de **10kΩ** sur le dessus, mais celle de **1kΩ** est reléguée coté pistes cuivrées. Les quatre trous de passage des vis support sont centrés sur les trous angulaires de la plaquette préperçée. Ainsi il devient plus commode d'agrandir ces trous en assurant la coïncidence entre les deux éléments. On remarque sur la Fig.18 et sur la Fig.19 que l'espace situé sous le clavier est bien encombré par le connecteur des capteurs et par l'inverseur à bascule. Il reste cependant assez de place pour le libre passage des torons de fils souples.

**T**oujours à grande échelle, la Fig.20 présente le dessin du circuit imprimé caractéristique d'une forte "occupation des sols". Il faudra se montrer attentif lors de la réalisation des soudures proches des angles. Par exemple la cathode de la LED verte. Il faut éviter une soudure trop "volumineuse" qui engendrera une difficulté pour faire porter correctement l'entretoise isolante. Donc pas d'excès d'étain dans "les coins". Les pistes recevant les fils souples de liaison avec l'électronique principale sont doublées pour certaines. Autant choisir les plus longues ou les plus dégagées pour y souder le fil de liaison. Par nature, les liaisons filaires sont fragiles. Aussi, dès que l'assemblage aura été vérifié et passé les tests de validité, mettre en place les quatre petits boulons support et immobiliser grossièrement le circuit clavier sur la plaque principale. De nombreuses manipulations vont accompagner la réalisation du coffret, les tests du programme sur site etc. Ce serait vraiment dommage d'altérer ces liaisons filaires réalisées avec attention, patience et soin.

Fig.20



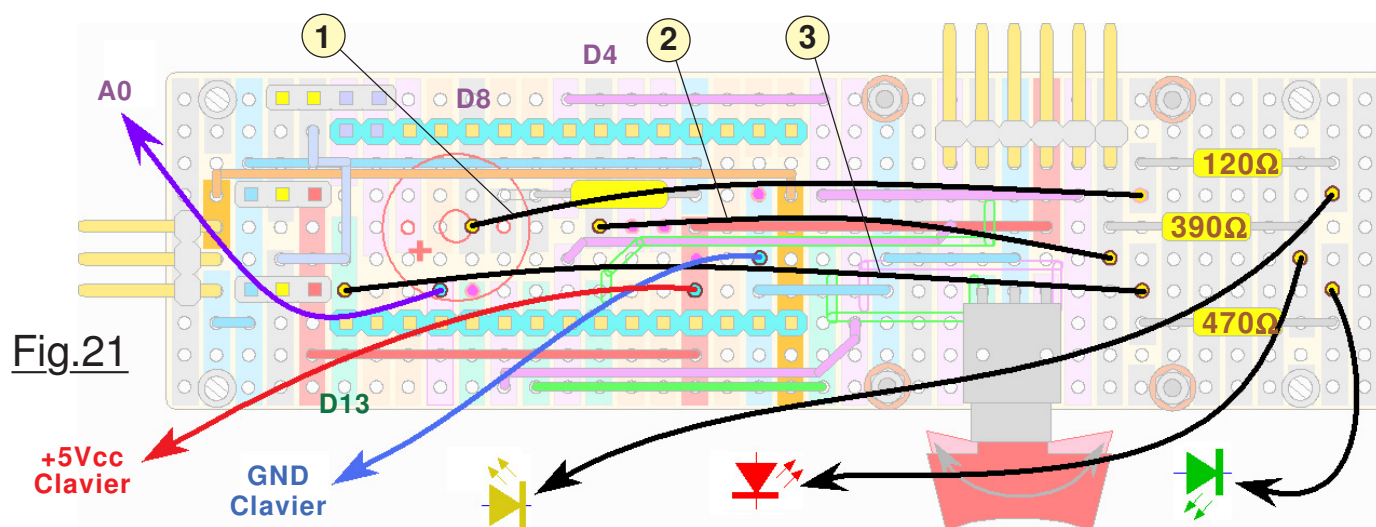
**Tous ces petits fils, on va s'y perdre !**

Souvent, la différence entre grogne agressive énervementale et "Fastoche finalement" se joue à peu de chose. Une petite idée de rien du tout vient considérablement faciliter une opération qui semblait particulièrement indigeste. À commencer par les branchements. Quand on arrive à la phase d'ajout des liaisons filaires souples, avec tous ces petits conducteurs gris la confusion est au rendez-vous, sans pour autant avoir été spécialement invitée. C'est bien le schéma de la Fig.4 et celui de la Fig.5 mais quand on compare aux dessins des figures 8 et 9 on y perd son latin.

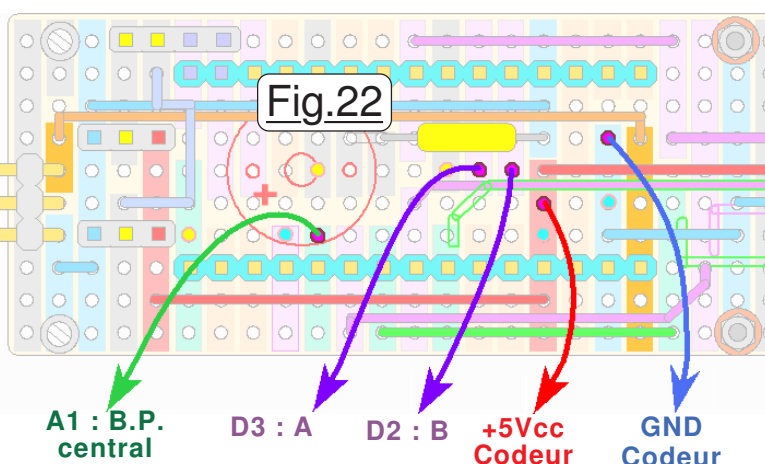
- *Vantard Totoche, t'as jamais pigé le plus petit mont latin carapé diem de carapé diem !*

- *Patienssus et calmus omnibus et circuitus imprimus soudus ... HAaaa mais alors !*

Un petit dessin de plus sera certainement une aide précieuse quand ce bon vieux frère à souder sera à la bonne température. La Fig.21 devrait apporter une aide significative pour vous y retrouver.



En premier on s'occupe des diodes électroluminescentes dont les cercles d'implantation des fils de liaison sont jaunes. Vous aurez mis en place le conducteur **1** qui va de **D8** à la **120Ω**. Puis, en **2** la liaison entre **D4** à la **390Ω**. Enfin le troisième pont en **3** relie **D13** à la résistance de **470Ω**. Pour les liaisons du toron qui va jusqu'au clavier il y a les six fils. (*Dont trois cercles d'implantation coloriés en bleu ciel.*) Trois "jaunes" sont soudés à droite pour piloter les LEDs. Les trois autres vont à **A0** et aux pistes **GND** et **+5Vcc**. Rassemblez ces six liaisons en un toron unique et compact avec un petit tronçon de gaine thermo-rétractable et c'est fini pour le clavier. Pour le codeur incrémental c'est aussi simple, car il n'y a que cinq liaisons dont les implantations sont repérées par les petits cercles rose bonbon. (*Miam miam.*) On retrouve deux connections pour **GND** et **+5Vcc**. Surtout n'oubliez pas que tous doivent être soudés **AVANT** le Buzzer.



**Le connecteur du codeur rotatif incrémental.**

Immobilisé sur le couvercle, le codeur rotatif incrémental en fait partie intégrante. Si l'on veut pouvoir intervenir aisément dans le coffret, il faut absolument pouvoir le débrancher. Ce n'est pas compliqué du tout. Comme le capteur est fourni dans le commerce avec un petit connecteur HE14 à cinq picots, on termine le toron concerné par un équivalent femelle. On affecte au toron une longueur minimale pour pouvoir le "rétracter" dans le coffret quand on referme. Sa longueur sera toutefois suffisante pour permettre un dégagement suffisant du couvercle assurant un débranchement et rebranchement commode lors des opérations de maintenance ou lors de l'étape d'intégration.



Compte tenu des caractéristiques d'encombrement du capteur fourni dans le commerce et des impératifs d'exigüité du coffret, tel qu'il se présente le petit module ne convient pas. La cote d'encombrement **Ce** est trop importante, il serait impossible d'adopter la largeur désirée sans réduire cette dernière. Un peu de chirurgie s'impose sur ce codeur. La solution est simple. Il suffit de "faire passer" le petit connecteur en **B** comme montré par la flèche **F**. De loin, le plus facile et le plus propre consiste à couper les broches en **A** et à souder directement sur les anciennes en **B** un petit connecteur HE14 coudé. Sur la Fig.24 qui présente le couvercle vu de l'intérieur, le petit connecteur coudé se trouve en **X**. En faisant

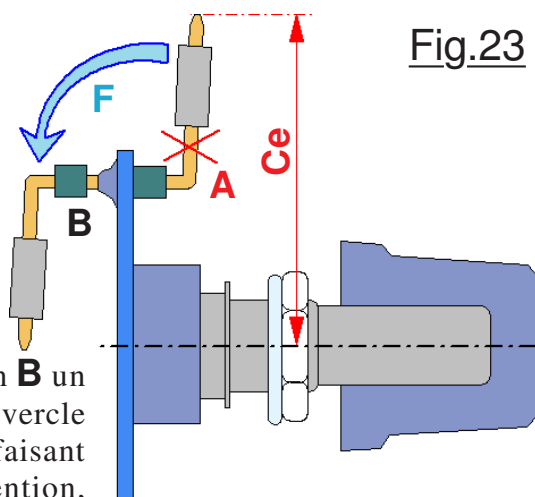
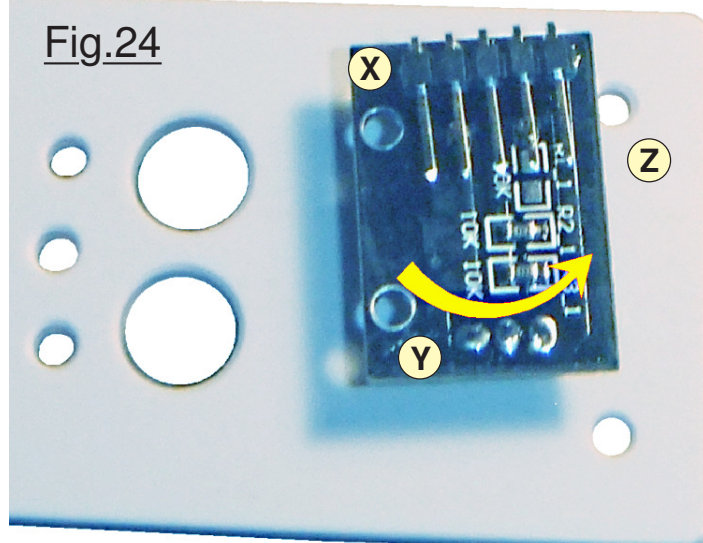


Fig.23

plus attention, on voit bien que l'orientation en rotation semble un peu bâclée. Les bords du petit circuit imprimé ne sont pas du tout parallèles à ceux du couvercle. C'est volontaire. Des critères esthétiques ont influencé la distance qui sépare les trous de passage des vis de fixation du couvercle par rapport aux bords de ce dernier. Dans ces conditions, si l'on veut respecter le faible encombrement du coffret et placer le codeur rotatif assez espacé des deux boutons poussoir du clavier, il faut accepter la déviation angulaire **Y** pour permettre le libre passage de la vis à travers le trou **Z**. Cette particularité est strictement sans inconvénient pratique.

Fig.24



**Les petits riens qui font la différence.**

Parfois, la dichotomie entre une galère interminable accompagnée d'un agacement peu propice à de la belle ouvrage, et une progression calme et régulière ne tient qu'à deux ou trois petits détails qui ne sont pas forcément évidents sur les photographies. La zone encombrée du minuscule clavier à deux touches mérite certaines précisions. Sur la Fig.25 on retrouve en **1** le faisceau de fils souples le plus court possible mais assez long pour pouvoir débrancher le codeur rotatif et déposer le couvercle. Le toron **2** pour sa part doit rester à l'intérieur du coffret, mais autoriser l'éloignement suffisant du circuit imprimé **5** pour permettre de souder coté piste, et par interdire l'introduction du circuit principal **13** dans le coffret. Le circuit imprimé **5** sera lors de l'intégration finale ajusté finement en hauteur pour placer les collerettes de la base des deux boutons poussoir juste en dessous du couvercle, et obtenir ainsi le dépassement maximal de leur partie cylindrique activable. La hauteur des trois petites LEDs dont la verte **6** est "plate" sera ajustée avec attention au moment du soudage pour qu'elles puissent dépasser légèrement du couvercle. C'est la hauteur des deux boutons poussoirs qui conditionne leur positionnement.

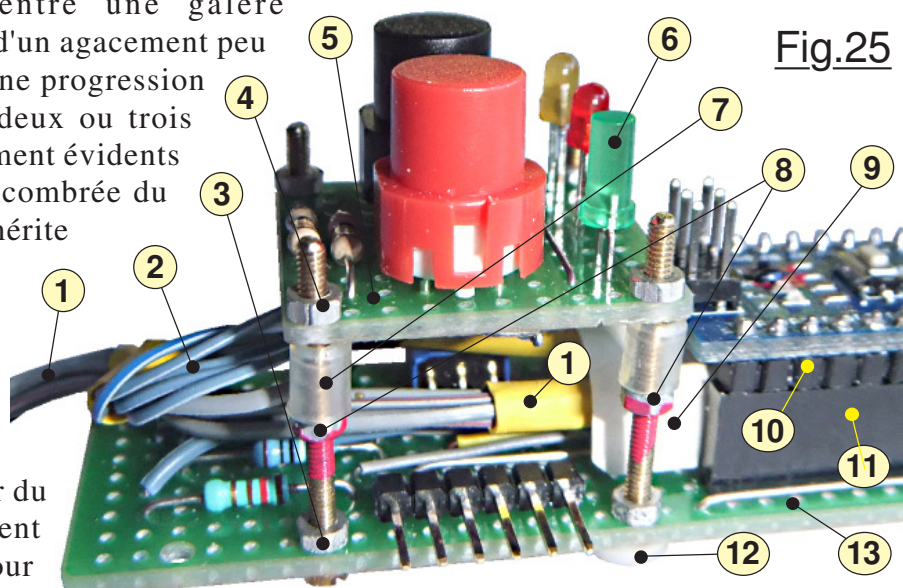


Fig.25

nous le verrons plus avant ne



Revenons à l'immobilisation du clavier **5** sur le circuit imprimé principal **13**. On commence par bloquer provisoirement les quatre boulons sur **13** au moyen des écrous inférieurs **3**. Les deux vis de gauche sont isolées des pistes cuivrées par les rondelles en nylon **12**. On insère à hauteur approximative les quatre écrous intermédiaires **8** ainsi que les entretoises en nylon **7**. Ces dernières de faible diamètre portent correctement sur **5** sans être gênées par les soudures voisines. On place ensuite les contre-écrous **4** et l'on recherche la position précise en hauteur des écrous **8**. Cet est en partie réalisé pour disposer au mesurer avec précision la position du des écrous **8** est déterminée, on colle ces derniers en position avec du vernis à ongles qui ici est rose bonbonmiamiam. Inutile de serrer fortement les écrous **4** lors de l'assemblage final. Les entretoises **7** présentant une élasticité relative il ne faut pas les écraser. Cette souplesse assurera le "freinage" des écrous **4** car le nylon se comporte comme une "rondelle" élastique.

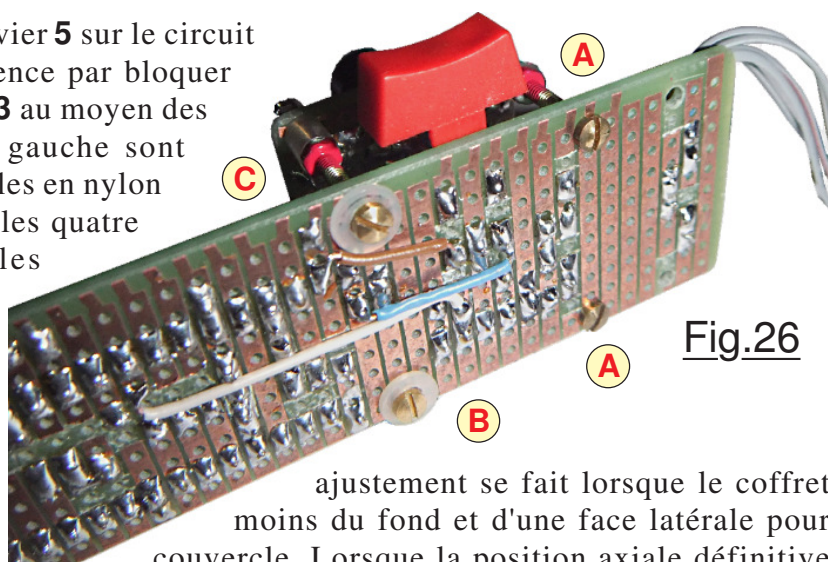


Fig.26

ajustement se fait lorsque le coffret moins du fond et d'une face latérale pour couvercle. Lorsque la position axiale définitive

Autre petite tracasserie facile à contourner qui se présente quand on introduit la petite électronique Nano Arduino dont en **10** on voit le connecteur mâle, sur le support HE14 femelle **11**. Les fils des torons **1** et **2** font preuve d'une tendance systématique à faire "gonfler" les deux faisceaux car les petites bagues de serrage jaunes ne réunissent que localement les fils souples épris de liberté. Vous pouvez être certains qu'au moment où vous allez effectuer une pression sur la carte électronique, un ou deux individus seront pincés entre **10** et **11**. La résolution de cette petite rébellion est élémentaire. On intercale un petit protecteur en carton **9** cintré en une "gouttière" en U inversée qui coiffe tous les fils situés entre les deux rangées **A** et **B** des deux lignes de contacts **11**.

D'une qualité médiocre, la Fig.26 sur laquelle on remarque nettement le freinage des écrous intermédiaires et l'inverseur à bascule latéral, montre toutefois que les têtes des deux vis **A** provoquent un court circuit entre trois pistes cuivrées voisines. La Fig.8 démontre que ces trois pistes restent inutilisées, donc leurs union électrique est sans importance. En revanche, en **B** et **C** la liaison intempestive est carrément interdite d'où l'obligation d'intercaler une rondelle isolante. Les deux trous de passage des vis sont chanfreinés pour interdire tout contact électrique entre le filetage et la piste cuivrée centrale. Ainsi aucun risque de conduction prohibée avec le circuit imprimé du clavier. Pas très visible sur la Fig.26 la rondelle **C** passe un peu en dessous du pont filaire marron. Oui, je sais que ce n'est pas avec la Fig.27 que je vais remporter le premier prix de photographie artistique. On y remarque toutefois que les rondelles isolantes débordent un peu de la plaque cuivrée.

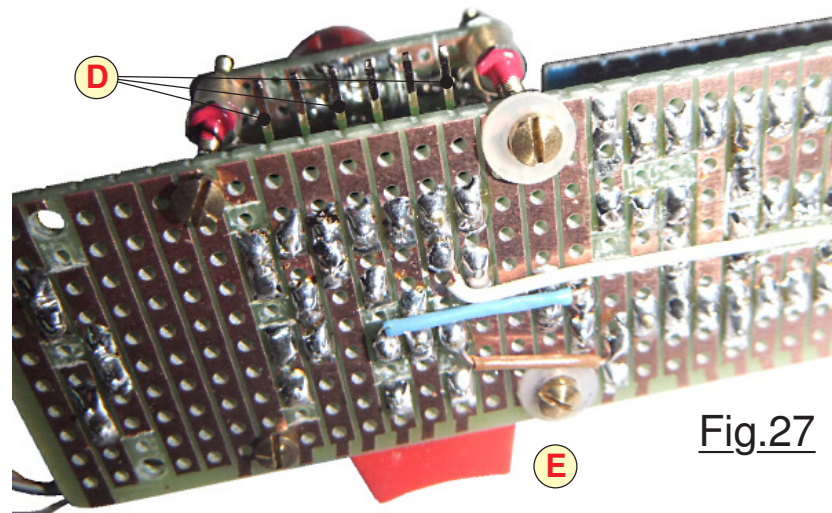
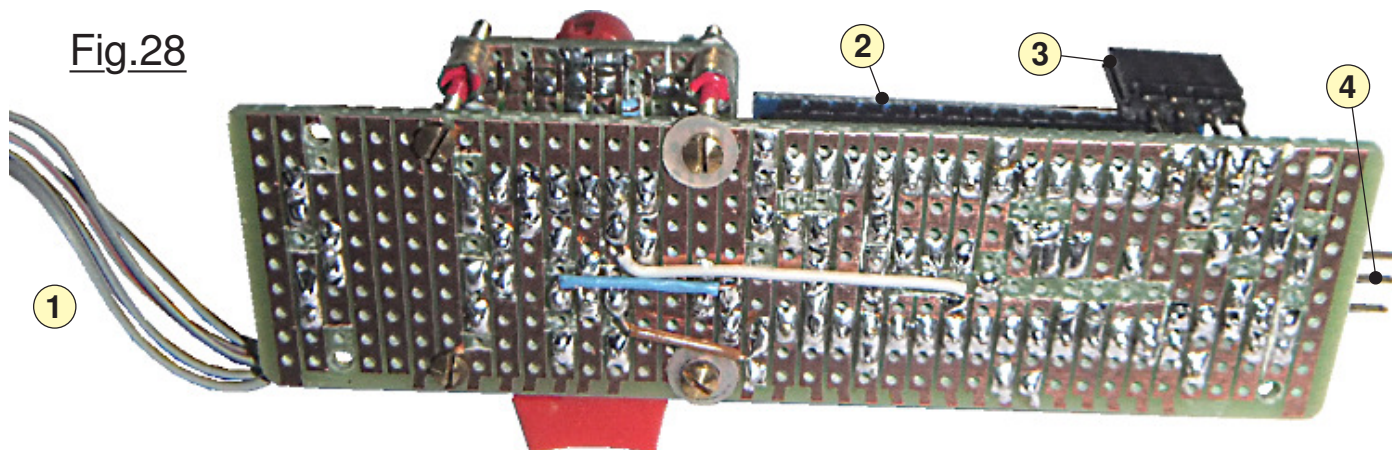


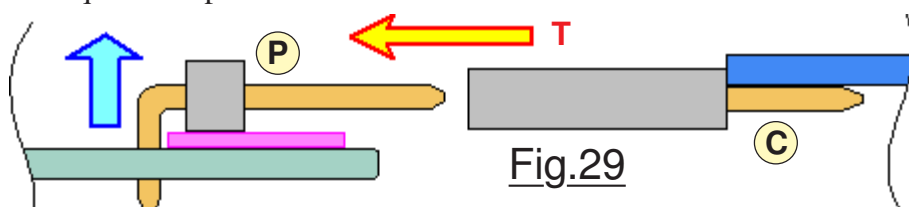
Fig.27

Ce petit détail est strictement sans importance, car ce qui complique la conception du coffret pour pouvoir y insérer sans trop torturer l'électronique, c'est le débordement en **D** (*Assez peu visibles sur l'image.*) des picots du connecteur des deux capteurs météorologiques, et en **E** de la partie mobile de l'inverseur à bascule. Les grosses barres blanche, bleue et marron sont en réalité de tout petits fils de câblage électriques. Je vous conseille de souder le marron avant le bleu pour ne pas altérer son isolant par la chaleur.

Fig.28



Une dernière petite image en Fig.28 pour vous présenter l'électronique principale entièrement terminée. En **1** le toron de fils n'est pas encore "compacté" par une bague en gaine thermo-rétractable. En **2**, pas tellement visible, la carte NANO enfichée sur son support. En **3** le connecteur à quatre broches qui supporte l'afficheur OLED qui n'est pas encore en place. Enfin en **4** le petit connecteur HE14 coudé sur lequel il sera possible de brancher, si on désire l'autonomie, une pile de 9V dont il sera question plus avant. Deux connecteurs coudés sont soudés sur le circuit imprimé principal, ce



coudé. Pour pouvoir engager facilement **C** sur **P** par la translation **T** il faut qu'entre le HE14 femelle et le circuit imprimé subsiste un petit jeu. Pour aménager un petit écart suffisant, la technique consiste à surélever le connecteur coudé (*Flèche bleue.*) au moyen d'un petit carton provisoire (*En rose sur le dessin.*) au moment de la soudure. L'électronique de base étant fonctionnelle, nous pouvons passer à la réalisation du petit circuit qui supporte les deux capteurs météorologiques.

### Le circuit imprimé de support des deux capteurs.

Disposé à l'arrière du coffret et solidaire de ce dernier si notre petite réalisation est anoblée à la fonction d'altimètre, il faut concevoir un assemblage aussi petit que possible. Le petit circuit imprimé qui sera chargé de cette mission devra placer les capteurs pratiquement contre le coffret pour que l'ensemble avec le protecteur approprié soit aussi compact que possible. (Voir la Fig.30) C'est ce critère de miniaturisation qui a présidé la conception de la minuscule plaquette cuivrée **4**. En **1** on voit le trou de passage de l'une des deux vis de liaison sur le coffret du protecteur mécanique. En **2** on reconnaît le capteur d'humidité et en **3** celui de la pression atmosphérique. Le module **2** est pratiquement en contact avec la face arrière du coffret.

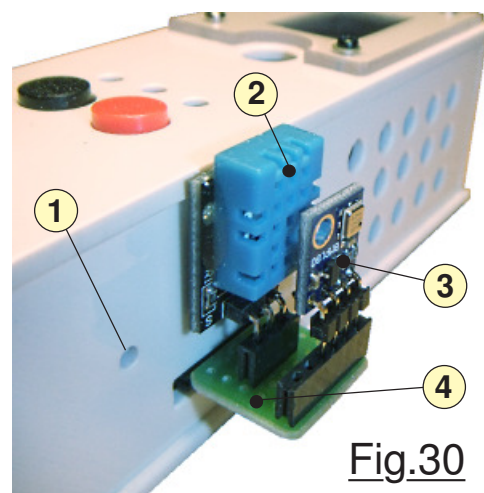
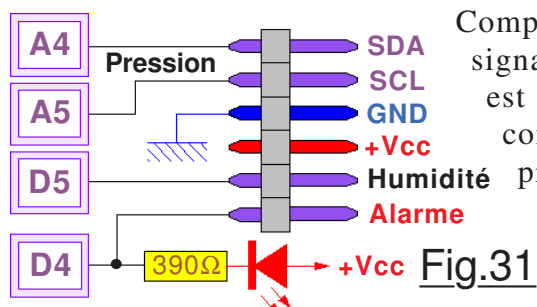


Fig.30



Complétant les Fig.4 et Fig.5 le schéma de la Fig.31 résume les signaux disponible sur le connecteur HE14 coudé qui sur le dessin est vu de dessus. Les deux petits capteurs seront insérés sur des connecteurs HE14 car ils sont amovibles. On peut à convenance préférer une station météo compacte et "monolithique", ou au contraire préserver le coffret dans un local douillet, et placer les capteurs loin dans la zone intérieure ou extérieure à surveiller. Dans ce cas une longue ligne décrite plus avant



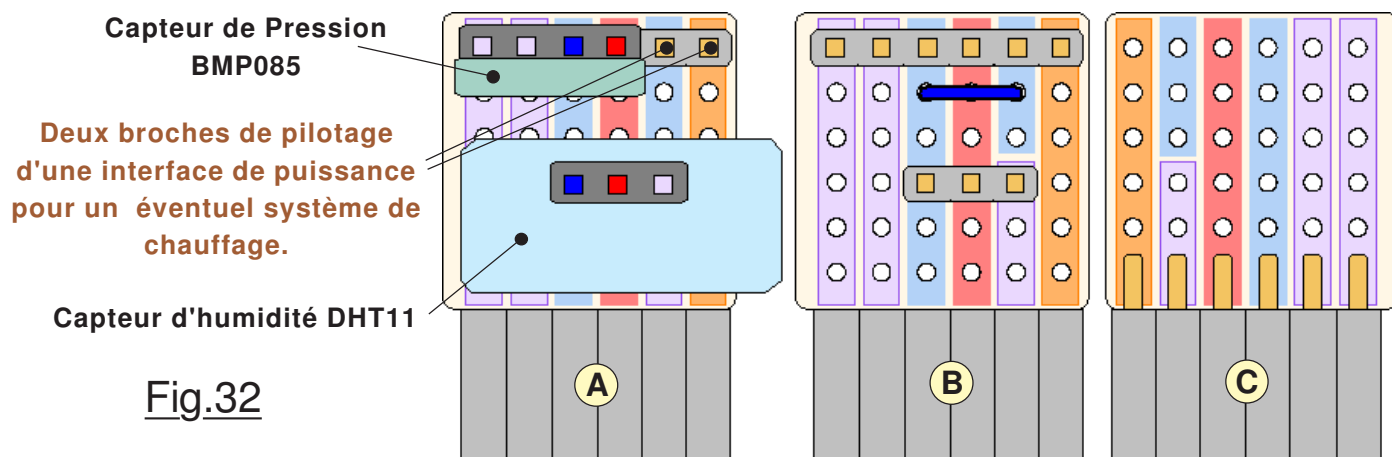
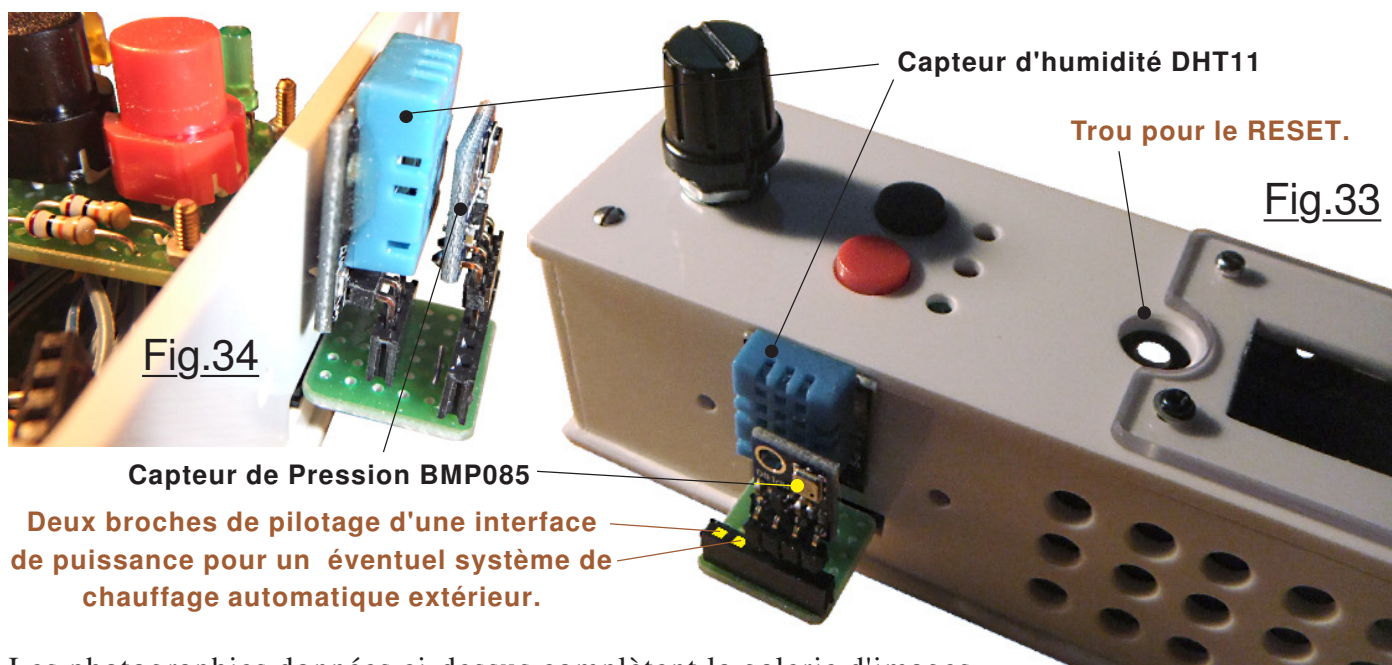


Fig.32

sera mise à contribution. Avec les conventions habituelles, la Fig.32 propose le dessin du circuit imprimé à grande échelle. En **B** le circuit est vu coté composant. Les pistes cuivrées situées sur le dessous sont vues "par transparence". En **C** le module est vu coté pistes cuivrées. Enfin en **A** le circuit est vu de dessus, avec les deux capteurs insérés sur les supports.



Les photographies données ci-dessus complètent la galerie d'images avec en Fig.34 une vue proche de la zone du petit clavier à deux touches. On y voit nettement que le module du capteur d'humidité est très proche du coffret. Sur la Fig.33 la petite plaque qui sert de "vitre" est échancrée pour dégager le trou de passage d'un stylet pour provoquer un RESET de l'extérieur si nécessaire. Par l'orifice RESET du couvercle on aperçoit l'un des petits trous d'aération sur la paroi latérale. Enfin, sur la Fig.35 sont proposées deux vues du petit capot de protection des capteurs qui se fixe sur la face arrière du coffret de la NANO STATION MÉTÉO.

Trou pour laisser passer vers le dessus une petite ligne électrique à deux fils pour éventuellement piloter une interface de puissance extérieure

Fig.35

Trous de passage pour les deux vis de fixation avec le coffret

Protecteur des capteurs en vue intérieure.



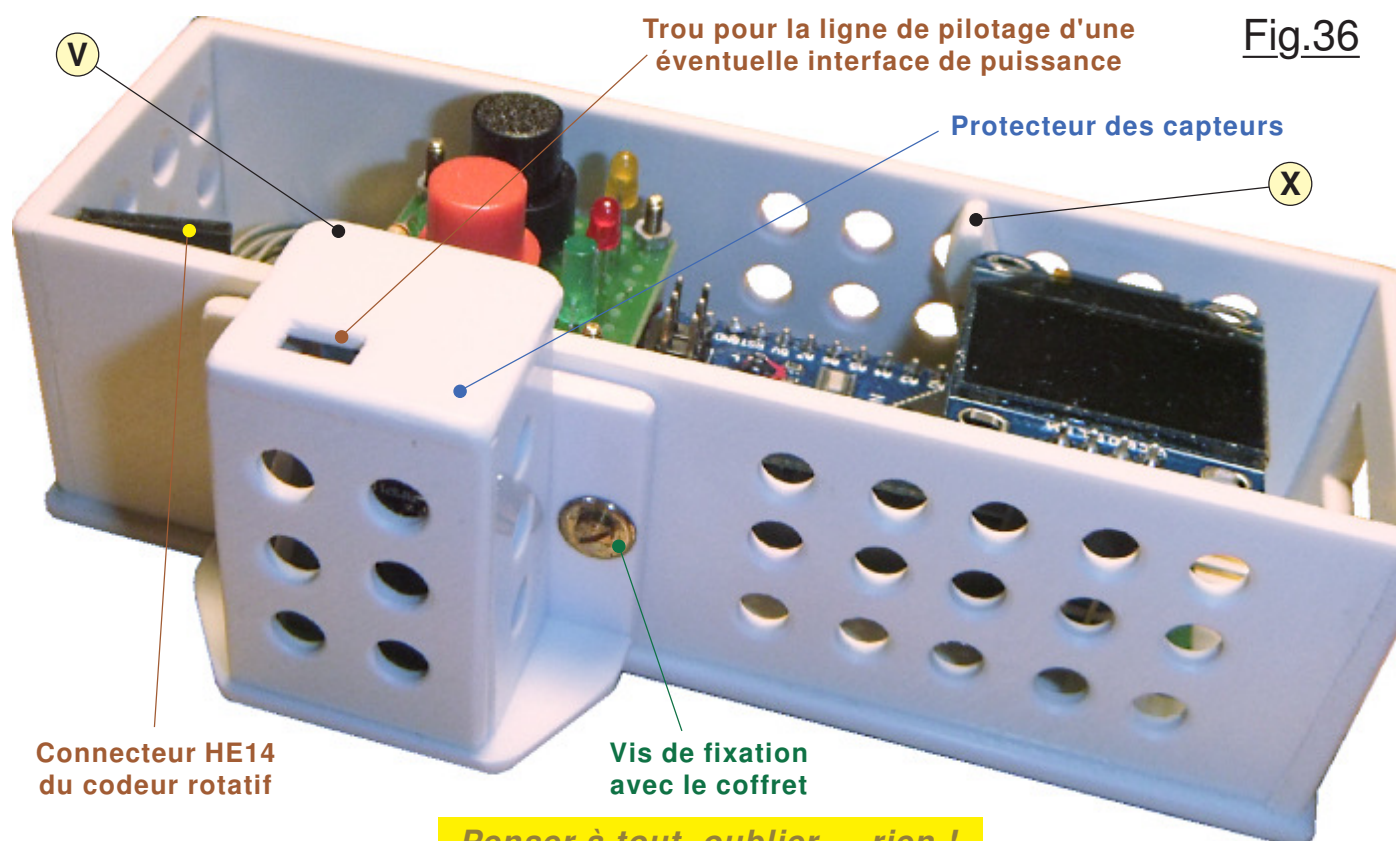


Fig.36

Étape essentielle d'un projet tel que celui-ci, **la phase de conception ne sera jamais élémentaire**. L'agencement final résulte forcément d'un compromis entre nos souhaits, et les difficultés inhérentes à la concrétisation limitée par nos moyens techniques. L'utilisation des matériaux de base et les méthodes pour les assembler vont influencer inexorablement l'encombrement du "produit fini". C'est durant l'étude du dessin qu'il faudra soumettre notre sagacité à un effort maximal, car à ce stade il faut penser à tout. La Fig.36 présente le prototype dont l'échelle se devine par la présence du minuscule afficheur OLED. Notez en **X** l'existence d'une petite pièce mécanique qui supporte l'écran LCD en face du connecteur HE14 sur lequel il est inséré. Un piège potentiel est à l'affût. Ce que ne montre pas de façon évidente la photographie ci-dessus, c'est que la "Visière" du protecteur dépasse sur le dessus de NANO MÉTÉO. Il est évident que les écrous de fixation pour immobiliser le petit capot seront serrés avant la mise en place du couvercle. Hors, pour positionner le couvercle il faudra l'introduire en biais pour le décaler des deux boutons poussoir. Cette manipulation ne sera possible que si la visière **V** ménager un espace d'environ **3mm** au dessus du couvercle lorsque cet élément est en position définitive. Les dimensions du protecteur doivent tenir compte de cet impératif. (Voir la Fig.37 ci-dessus.)

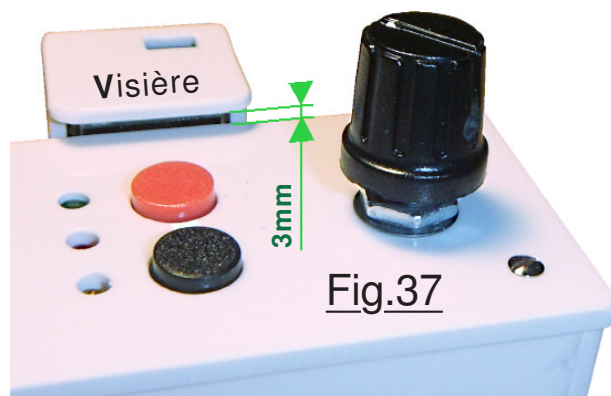
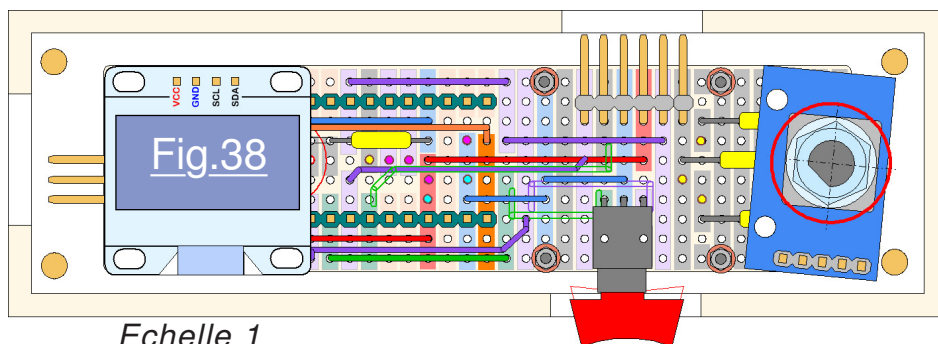


Fig.37

### Les dessins à l'échelle du coffret.

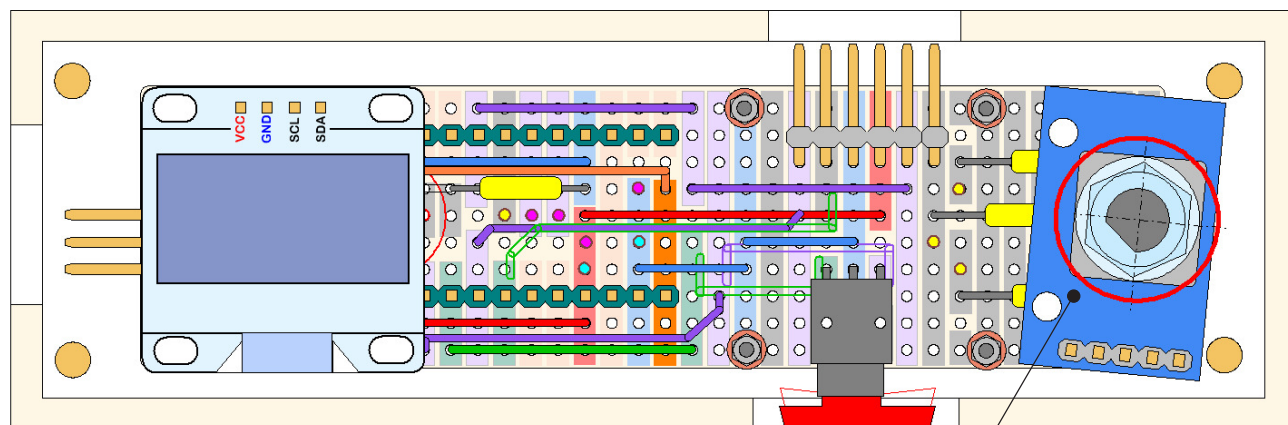
Surtout s'il est tracé en respectant avec rigueur les dimensions de l'intégralité des éléments, un dessin en plusieurs vues sera faussement rassurant. **Il ne peut garantir à lui seul la faisabilité matérielle**. Vous livrant "en bloc promotionnel" mes dessins, mes méthodes et éventuellement certains avertissements pour vous éviter de vous fourvoyer, vous risquez fortement d'imaginer que la réalisation du prototype a été linéaire. Le dessin terminé le coffret a été réalisé, puis l'assemblage, puis la mise en service. Et bien non, pas aussi directement. Par exemple le protecteur a été étudié quand le coffret était achevé ... mais dès le début la présence des boulons de liaison était prise en compte. Dessins terminés, un vague clone en carton épais a été découpé, collé, pour vérifier que



Echelle 1

l'introduction des divers modules était réaliste. Ce n'est absolument pas anodin, car la recherche d'un boîtier le plus petit possible rend l'intégration assez "technique" nous y reviendrons. Par exemple la Fig.38 est réputée représentée à l'échelle unitaire. Ce ne sera

vrai que si vous imprimez le document avec une machine qui ne changera pas les attributs du format A4 de la page virtuelle. Dans ce cas, on réalise à quel point les modules à l'intérieur du coffret ne laissent que peu de place pour le passage des outils. (*Et encore moins pour les doigts !*)



Vue de dessus.

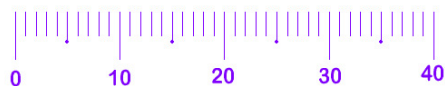
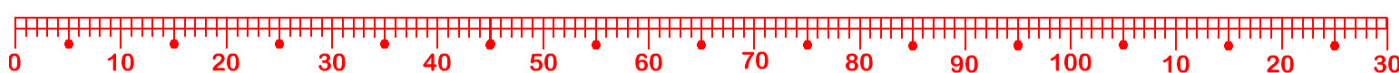
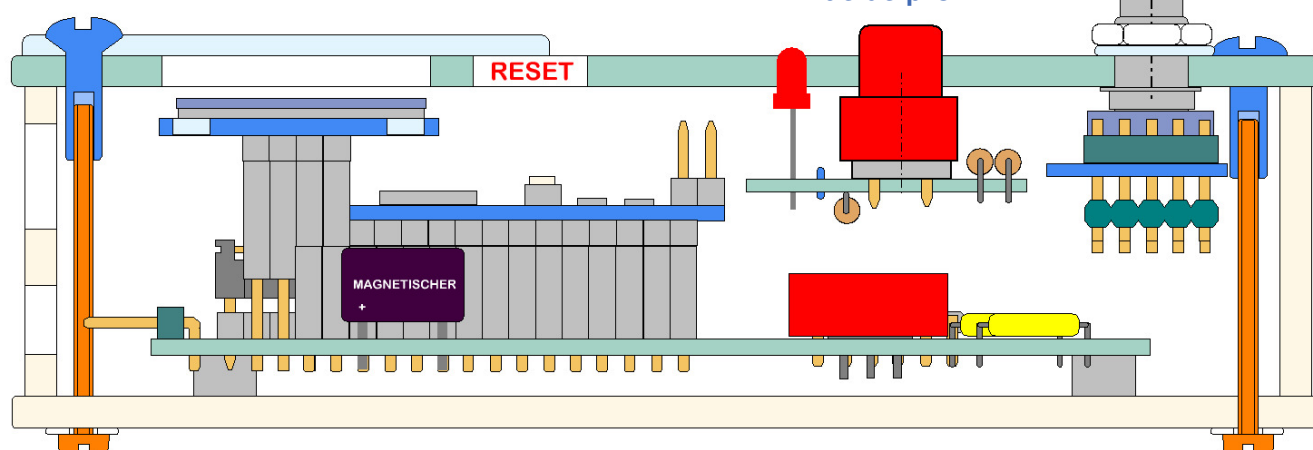


Fig.39

(Voir la Fig.24.)

Vue de profil.



L'impression de confort qui se dégage de la Fig.39 est totalement injustifiée. Le dessin est grand, beau, en couleurs. Si partout les jeux impératifs au montage / démontage sont suffisants, il n'en reste pas moins qu'ils sont calculés au plus juste, et que des petits pinces précelles seront indispensables ... plus quelques petits aménagements particuliers. Il est manifeste sur la vue de dessus, par exemple, que la largeur de l'électronique entre le connecteur pour les capteurs et la partie active de l'inverseur à bascule est supérieure à la largeur intérieure du coffret. Donc "ça ne rentre pas" ... sauf en biais et avec astuce dont les détails seront précisés plus avant. Les échelles graduées sont ajoutées pour pouvoir déterminer toutes les dimensions critiques du prototype



### Du carton à la matière thermoplastique.

Il n'est pas inutile d'en rajouter une couche sur le polystyrène choc et mes techniques de collage au trichloréthylène, méthodes personnelles qui ne seront certainement pas les vôtres. Je vais me contenter de souligner certains aspects pratiques pour vous aider, ensuite, ce sera à vous à adapter ces observations pragmatiques à vos contraintes locales. Diverses photographies réalisées durant le façonnage des éléments ainsi que leur assemblage aideront probablement certains lecteurs "débutant". Le premier sous-but consiste à créer le coffret proprement dit visible en Fig.40 en espérant "avoir pensé à tout" et que l'on parvienne à la victoire de la Fig.41 démontrant la faisabilité d'une intégration réussie. En **1** se trouve l'ouverture de passage du circuit imprimé qui supporte les deux capteurs météorologiques ainsi que les deux trous de fixation du protecteur. En face, en **2**, bien observable, le trou pour la partie active de l'inverseur à bascule. En **3** l'un des nombreux trous d'aération qui ne sont certainement pas justifiables par une quelconque évacuation calorifique. Ne consommant que 30mA sous 9Vcc (*Quand les LEDs sont éteintes.*) ce n'est pas les 0,3W de puissance dégagée qui vont faire chauffer l'électronique interne. Ces nombreux trous sont purement esthétiques. Si vous ne partagez pas cette approche artistique avec l'auteur de ces lignes, surtout ne vous croyez pas obligé de copier. En **4** le trou de fixation du petit support de l'écran OLED n'est pas encore percé. C'est en **5** que l'on introduit la ligne USB qui permet l'alimentation en 5Vcc à partir d'un petit convertisseur secteur ou les téléversements de programmes depuis le P.C. En **6** on peut brancher une petite pile 9V pour l'alimentation en autonome.

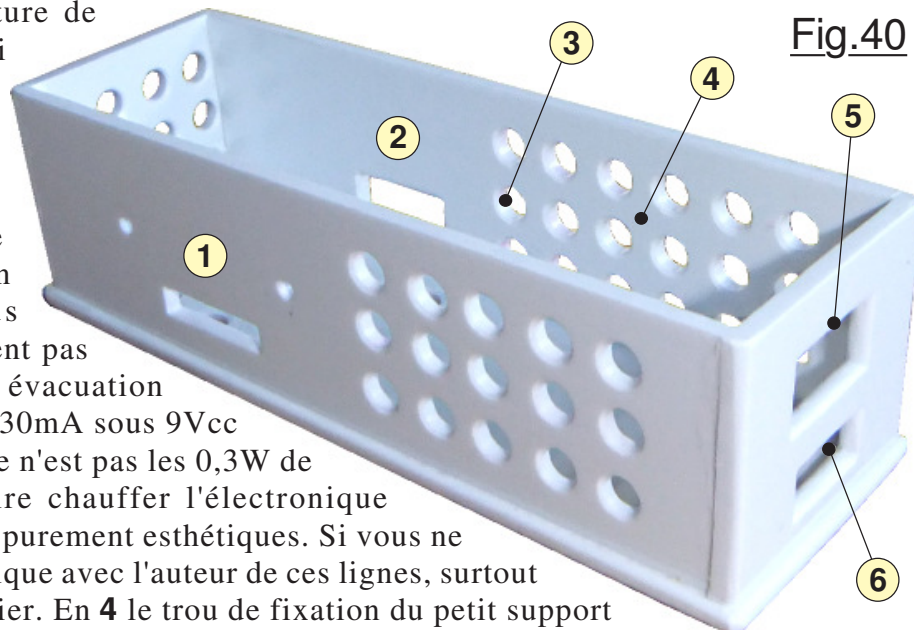


Fig.40

Photographiée lorsque l'appareil était en cours de réalisation, la vue de la Fig.41 prouve que tout peut se positionner à l'intérieur de l'étroit coffret, mais au prix que quelques petits astuces décrites dans le chapitre suivant. En **7** se trouve la seule zone située entre l'afficheur graphique et le petit clavier qui "gaspille de la place, c'est à dire environ trois centimètres cubes de non occupés. Ce vide vertigineux laisse libre le passage d'un éventuel stylet vers le bouton de RESET de la petite carte NANO Arduino. En **8** on devine les trous de liaison du protecteur qui à ce stade du projet n'est pas encore étudié. Seuls les deux trous de passage des vis de fixation ont été percés, leur position étant évaluée par les dimensions estimées du protecteur et surtout la possibilité de passer une pince précelle pour enfiler les rondelles d'appui et tenir les écrous. En **9** le codeur rotatif arrive bien à s'insérer dans l'emplacement qui lui est réservé ... sans plus !

### Trous de passage des vis pour l'immobilisation du couvercle

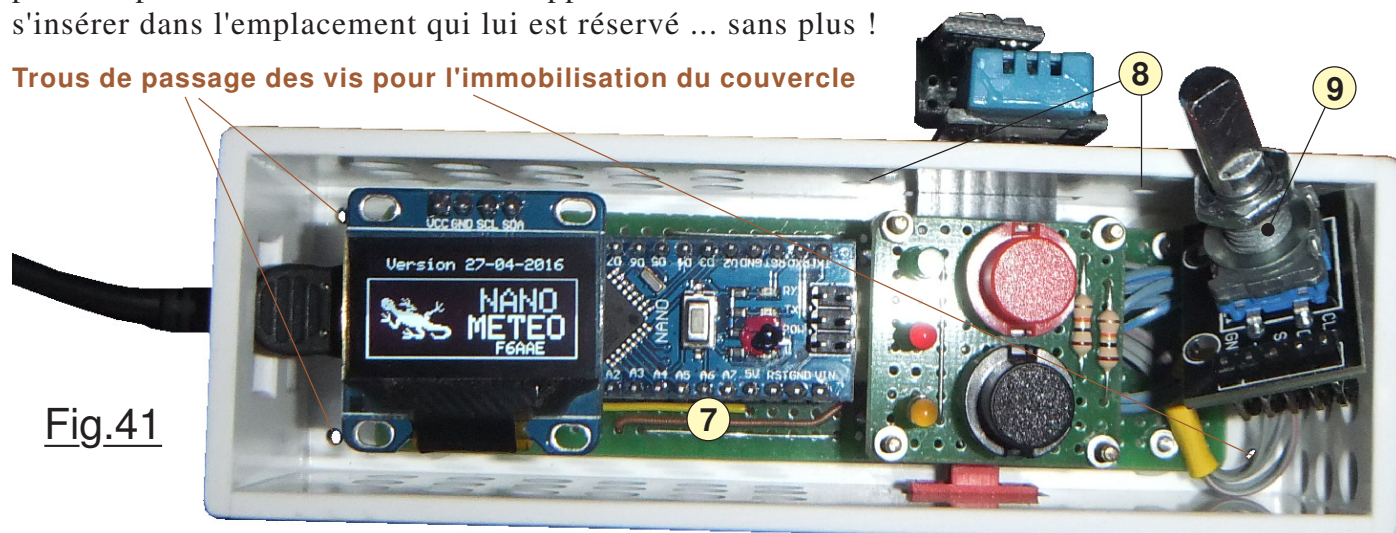
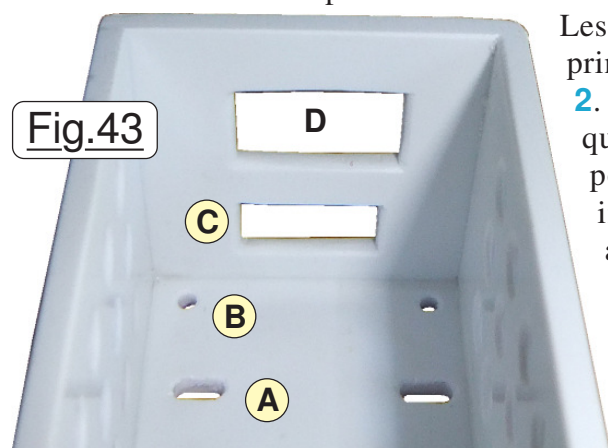
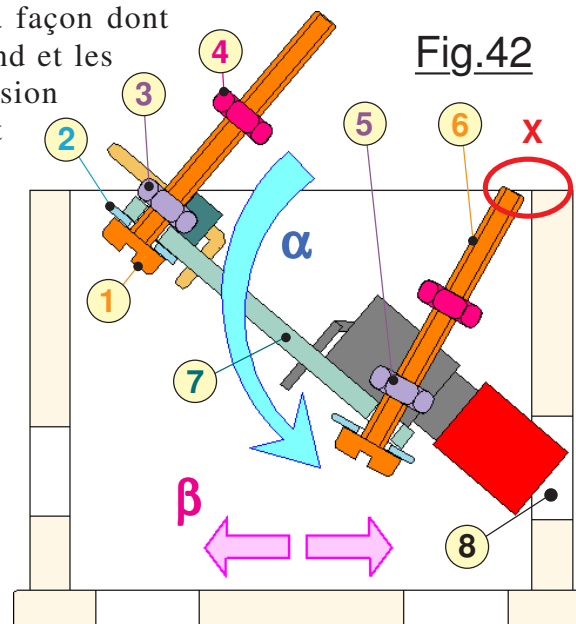


Fig.41

**Patience et longueur de temps font plus que force ni que rage !**

Difficile de comprendre comment on peut arriver sereinement à tout loger et assembler dans si peu de place, sans se prendre la tête ou proférer des gros mots qu'il serait politiquement incorrect de mentionner dans le texte. Par ailleurs, le résultat final ne sera satisfaisant que si les orifices de passage des boutons poussoir, de l'inverseur, des connecteurs se retrouvent avec le minimum de jeu et "bien en face". J'ai déjà souligné la façon dont j'arrive assez facilement à ce résultat. En premier, le fond et les quatre flancs latéraux sont découpés et ajustés avec précision à leurs dimensions d'encombrement définitives. Durant cette étape on fait bien attention à la perpendicularité des arêtes. Puis les trous de fixation du circuit imprimé principal sont percés sur la plaque du fond. Ce module y est immobilisé provisoirement à sa place avec les petits boulons et les entretoises. On peut alors tracer avec précision la position des divers orifices à ajourer sur les faces latérales. On assure ainsi une parfaite coïncidence entre les trous et les éléments qui devront les traverser. Ce n'est que lorsque tous les trous sont achevés que l'on procède à la solidarisation finale.

Considérons le dessin de la Fig.42 qui permet d'expliquer les différentes phases d'intégration du circuit imprimé principal, suivi de l'introduction du clavier. Le module principal étant plus large que la cote intérieure du coffret, la seule façon de pouvoir le mettre en place consiste à l'incliner inverseur vers le bas. Initialement, durant les essais nous avons finement positionné en hauteur le clavier et collé les écrous 4 à leur place définitive.

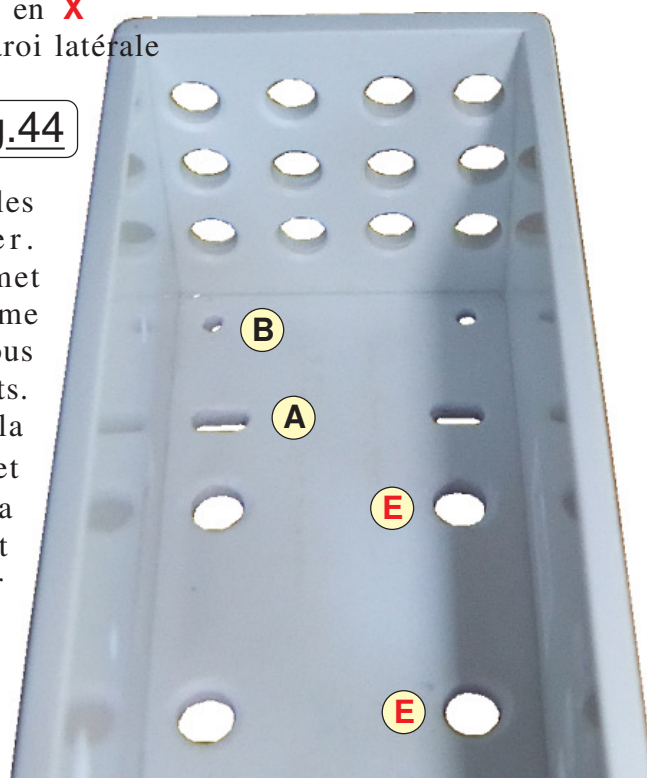


Les vis 1 et les écrous 3 étaient immobilisés sur le circuit principal sans oublier les deux rondelles d'isolation électrique 2. Pour effectuer la manipulation, on enlève le clavier ainsi que ses quatre entretoises. Mais ce n'est pas suffisant pour pouvoir introduire le total au fond du coffret. Les vis 6 interfèrent en X avec la paroi latérale

**Fig.44**

Il faut les dégager.

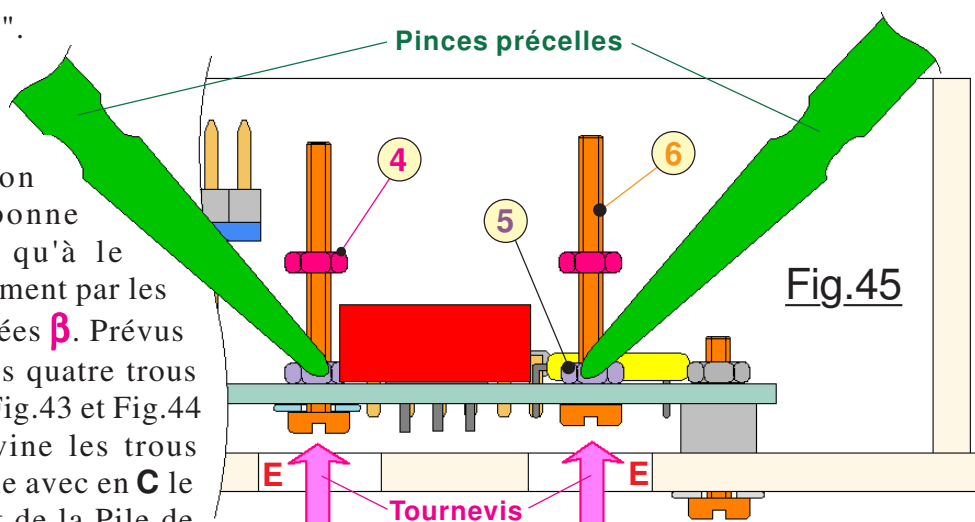
Dans ce but, on libère les écrous 5 ce qui permet d'incliner les longues vis 6 vers l'intérieur comme montré sur la Fig.42 à condition toutefois que les trous de passage à travers 7 soient de diamètres suffisants. Lors de la descente de l'ensemble on coordonne la rotation α pour que l'inverseur puisse pénétrer et traverser l'orifice latéral. Cette opération ne sera possible que si l'ouverture 8 est de taille suffisante et positionnée correctement en hauteur. C'est pour affiner ces éléments critiques que le "mammoth" réalisé préalablement en carton sera d'une aide précieuse. On peut ainsi optimiser les dimensions de la lucarne et la disposer avec rigueur. Au moment d'assembler le total, la manipulation coule de source





et notre satisfaction "culmine".

Quand le circuit imprimé principal "touche le fond", on introduit les entretoises et l'on peut procéder à son immobilisation. Il est à bonne hauteur, il ne reste plus qu'à le positionner au mieux latéralement par les translations rectilignes repérées  $\beta$ . Prévus dès la réalisation du fond, les quatre trous repérés **A** sur les dessins des Fig.43 et Fig.44 sont allongés. En **B** on devine les trous d'immobilisation du couvercle avec en **C** le passage pour le branchement de la Pile de 9V assurant l'autonomie électrique. En **D** l'orifice pour le cordon USB n'est pas curviligne, c'est la distorsion sphérique de la macro photographie qui donne cette impression d'arêtes cintrées.



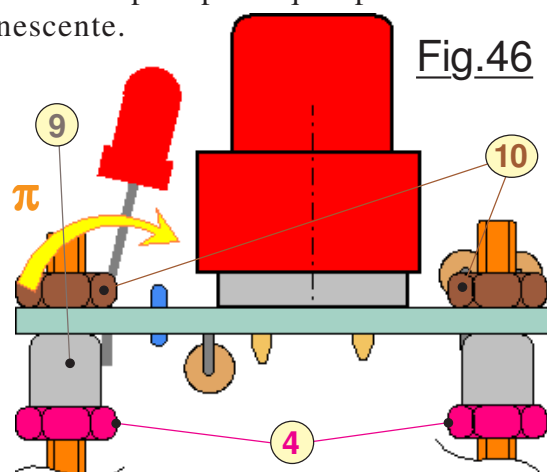
Lorsque le circuit imprimé est "définitivement" à sa place, parfaitement positionné en latéral, il faut immédiatement s'occuper du petit clavier, car il est "pendu" par son toron de raccordement, en configuration vulnérable. Ce n'est pas spécialement compliqué. On commence par redresser à la verticale les deux vis 6 et on les immobilise en serrant les écrous 5. Cette opération est facilitée par le fait que l'on peut facilement engager sur les cotés **une pince précelle** pour tenir l'écrou 5. Surtout, dans le fond du coffret nous avons pris soin de pratiquer quatre trous **E** de servitude en face des vis 6. Ainsi on tient l'écrou avec la pince précelle et on tourne 6 avec un tournevis qui traverse librement le fond. Vous observerez que pour les deux vis 1 ce n'était pas la peine, mais au moment de la conception rien ne prouvait qu'il ne serait pas nécessaire d'incliner les deux vis proches du connecteur HE14. Ces deux trous **E** pratiqués dans le fond font passer "d'impossible" à facile.

#### Mise en place du clavier.

L'opération est pratiquement élémentaire. On commence par introduire les quatre entretoises en nylon 9. Sur la Fig.46 les écrous 4 collés au vernis à ongles sont toujours de couleur identique à celle sélectionnée sur les autres schémas. Puis en prenant garde de ne pas tirer sur les fils souples du toron, on introduit à son tour le petit circuit imprimé du clavier. C'est enfin au tour des écrous 10 placés sur le dessus. Comme déjà précisé dans le didacticiel, il ne sert à rien de les serrer fortement. Il suffit de les mettre en légère pression, et affiner notre gestuelle pour que la plaquette support soit bien horizontale. Deux petites difficultés peuvent toutefois compliquer légèrement l'opération. La première résulte du manque de place latéralement pour tourner la clef. Une pince précelle auto serrante est bien commode pour l'insertion initiale. Puis avec une petite clef adaptée on termine la manipulation. La deuxième difficulté résulte de la proximité entre les deux LED très proches de 10. Durant le serrage de ce dernier la clef peut provoquer par un contact intempestif le pliage  $\pi$  des broches de la diode électroluminescente.

#### Intégration de l'afficheur OLED.

Insérer le petit écran sur le connecteur HE14 prévu à cet effet est un jeu d'enfant. Naturellement, on commence par protéger les torons de fils avec la "gouttière" en carton. Puis on place le module NANO Arduino sur les deux lignes de contact. En principe lors des essais les deux straps qui "polarisent" l'alimentation de l'écran LCD avaient été correctement répartis, et aucune raison ne justifie de les avoir retiré. Par contre, je sais que vous y avez pensé immédiatement, pour introduire le circuit imprimé principal



dans le coffret ainsi que le clavier, NANO Arduino avait été extrait des deux lignes de contacts. Afficheur OLED placé sur son support, il vaut mieux à ce stade vérifier le fonctionnement parfait de l'ensemble. Vous constaterez immédiatement en voulant brancher la minuscule fiche USB qui alimente l'appareil, que le connecteur est assez en retrait à l'intérieur. Comme le cordon masque globalement son trou de passage, placer la fiche exactement dans l'axe de la petite prise est assez délicat. C'est durant cette opération courante que vous allez apprécier la présence des trous latéraux "d'aération" qui permettent de voir l'intérieur de l'appareil.

**CONCLUSION :** *Si réaliser tous ces petits trous esthétiques n'emporte pas votre adhésion, prévoyez quand même une petite "lucarne" dans la zone latérale de la prise USB pour la voir quand vous désirerez brancher le cordon ; activité bien trop fréquente pour accepter une manipulation pénible.*

Force est de constater que les lyres du connecteur ne sont pas suffisantes pour soutenir l'afficheur qui se trouve en porte à faux. Outre que l'écran ne reste pas horizontal, il peut venir en contact avec la partie externe de la petite prise USB qui est au potentiel de la masse **GND**. De plus, penchant vers le bas à l'opposé du connecteur, elle peut venir interférer avec la fiche quand on cherche à insérer le cordon USB avec un risque notable d'endommager ce composant relativement fragile. Il faut donc impérativement supporter l'afficheur OLED du côté opposé au connecteur HE14.

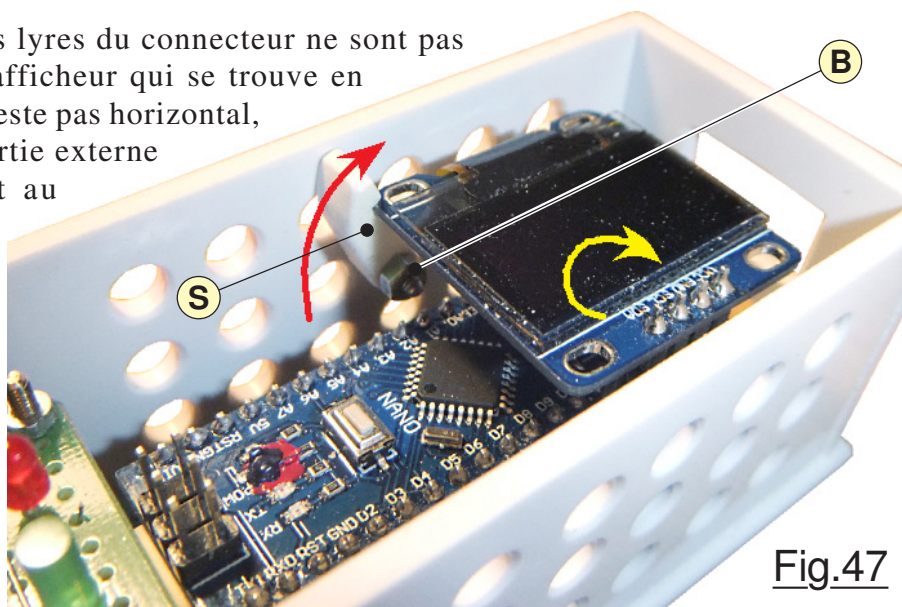


Fig.47

Saisie en gros plan, la Fig.47 peut faire croire que le support **S** confine à de la grosse mécanique. En comparant avec le boulon **B** qui ne fait que 2mm de diamètre nominal, on relativise. Cette petite pièce est plus proche de l'horlogerie que de la travée d'un pont routier basculant à six voies. Ce petit élément présente coté afficheur une fente dans laquelle le circuit imprimé de l'écran s'insère. Par la rotation de la flèche rouge on peut ajuster avec précision la hauteur, mais également le gauchissement par rotation symbolisée par la flèche jaune. Ainsi les cotés de l'écran seront parfaitement parallèles à ceux de la fenêtre du couvercle.

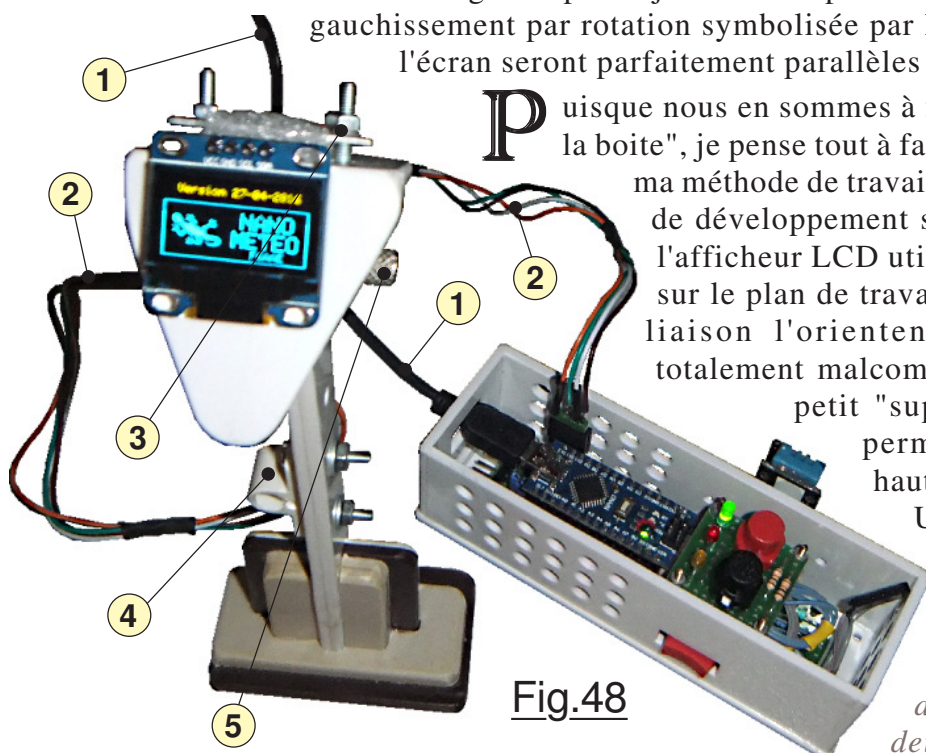


Fig.48

Puisque nous en sommes à faire les essais avant de "refermer la boîte", je pense tout à fait utile de vous présenter en Fig.48 ma méthode de travail quand un programme est en cours de développement sur site. Quel que soit la taille de l'afficheur LCD utilisé, il se penche toujours en biais sur le plan de travail. Quoi que l'on fasse, les fils de liaison l'orientent inexorablement en rendant totalement malcommode son observation. Aussi, un petit "support universel" à tout faire me permet de l'immobiliser à la bonne hauteur. On retrouve en **1** le cordon USB. Une liaison filaire souple longue **2** permet de relier n'importe quel élément placé loin de mes cartes Arduino. (Plusieurs lignes de ce type sont disponibles. Par exemple une deuxième ligne permet de brancher



le codeur rotatif et de le placer à portée de la main à proximité du clavier de l'ordinateur.) Si le développement risque de prendre plusieurs jours, la ligne **2** passe dans le pontet **4** pour avoir une préhension proche de l'afficheur directement ancrée sur la colonne du dispositif. La vis à tête moletée **5** constitue un petit système de serrage qui permet d'ajuster à convenance l'inclinaison de l'écran. Ce dernier est immobilisé sur le support par son connecteur HE14 avec une bride dont on peaufine le serrage avec les écrous **3**. Entre le connecteur HE14 et la bride est intercalé une mousse à bulles en protection. (Plusieurs plateaux orientables de dimensions différentes sont disponibles ...)

### Au tour du chapeau et l'habillage sera complet.

**S**imple à créer puisqu'il s'agit d'une plaque plate, sa finalisation reste délicate par nature. En effet, le résultat final ne sera beau que si tous les trous percés dans le couvercle coïncident parfaitement avec les éléments situés dessous ou qui le traversent. Arriver à ouvrir la fenêtre d'écran bien centrée sur la partie active de l'afficheur LCD n'est pas foncièrement délicat. Percer le trou juste au dessus du bouton de RESET est tout aussi aisé. De simples mesures entre les éléments concernés et les bords du coffret permettent de centrer correctement les orifices.

La vraie difficulté va se rencontrer avec les deux boutons poussoir et les trois LEDs. Pour **FC+** et **FC-** il faut que le jeu autour de la partie activable soit faible, trop d'espace nuit à l'esthétique. C'est d'autant plus vrai que la distance entre les deux boutons est faible. Pour les LEDs, c'est plus problématique. Considérons la Fig.49 pour laquelle on suppose que dans les deux exemples le trou percé dans le couvercle est un peu à coté. Dans le cas **B** les deux broches sont l'une derrière l'autre sur ce dessin. La diode jaune accepte de fléchir et peut traverser le couvercle. Par contre, dans le cas **A** la rigidité est totale. Impossible de faire dévier le corps rouge comme montré avec la flèche orange. Seul remède si le trou n'est pas parfaitement dans l'axe : l'allonger, artifice qui donnera à l'ensemble un aspect "bâclé". Dommage, il est trop tard. Vous l'avez compris, il faut absolument réaliser tous les orifices exactement "dans l'axe".

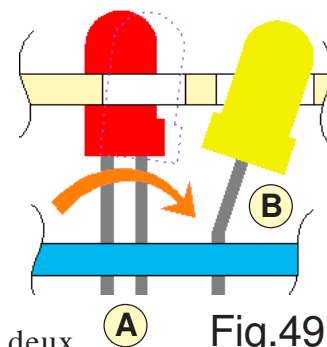


Fig.49

### Un carton pour un patron !

**R**ien à voir avec une quelconque lutte des classes en usine. Chaque fois que l'on est confronté à des activités où la coïncidence de divers trous se doit d'être parfaite, soit je procède par de l'usinage en contre-perçage, soit du carton est mis à contribution pour réaliser un patron. Lorsqu'un modèle précis est disponible et copieusement vérifié, il n'y a plus qu'à en faire une copie conforme avec le matériau privilégié pour matérialiser le coffret. Quand on en sera au stade de la Fig.50 et que tous les trous seront parfaitement "en face", de loin l'étape la plus délicate du projet sera dans notre sillage. On pourra souffler et contempler le résultat avec "fierté". Sur cette photographie, la petite plaque transparente qui fait office de vitre est invisible. C'est assez normal puisque la photographie, comme toutes les autres du reste, a été retouchée pour des raisons tant artistiques que pédagogiques. Ce détail anodin étant précisé, dans la pratique c'est aussi le cas, car le matériau utilisé est totalement translucide. Les deux petits boulons qui solidarisent la plaque transparente avec le couvercle quand on dépose ce dernier sont, cet pratiquement un tic, collées avec l'inévitable vernis à ongles. Les trois trous pour le passage des LEDs sont petits écarts imposés par le soudage sur le circuit imprimé du clavier, d'où le "contre" positionnement des trous avec précision.

Tous les orifices sont chanfreinés, facilitant ainsi considérablement la mise en place du couvercle sur le coffret. Sur cette image le non alignement angulaire du capteur rotatif est bien visible, le trou de passage de la vis de liaison Couvercle/Boitier est alors correctement dégagé.



Fig.50

**R**iche d'enseignements, la photographie de la Fig.51 livre une foule de petits secrets de fabrication. On se doute que réaliser le patron en carton semi rigide est presque aussi délicat que de tracer sur une plaque rigide de 3mm d'épaisseur. Percer et découper avec une paire de ciseaux en revanche est bien plus facile. Il n'en reste pas moins que l'on ne sera pas plus précis, certains trous seront inexorablement décalés. Notez au passage que le patron **P** est vu de dessus, il représente le couvercle quand on regarde du haut vers le bas. Le couvercle **C** est photographié en montrant la face intérieure. Le trou ovalisé **10** ne traverse pas. Il sert à loger l'ergot de positionnement angulaire du codeur rotatif. **Voici comment je m'y prends pour réaliser ce type d'élément mécanique :**

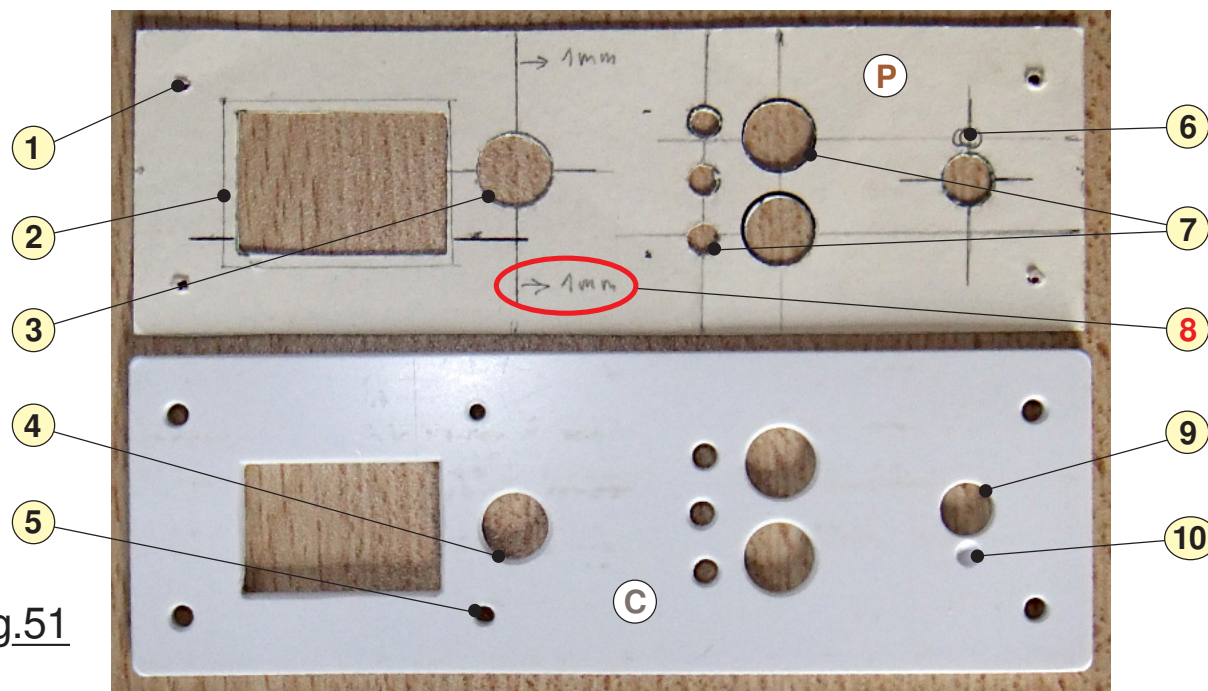


Fig.51

La première action consiste à découper un rectangle en carton dont les dimensions sont bien supérieures à celles de la partie supérieure du coffret. On commence par tracer et percer les trous de passage des LEDs et des deux boutons poussoir **7**. Il faut commencer par eux, car d'une part c'est la phase la plus minutieuse, d'autre part, la suite impose de placer le carton sur le coffret. Il faut que ces éléments de la façade puissent le traverser. Les trous seront découpés sans jeu, ainsi ils obligeront le carton à occuper une position précise sur son assise. On peut alors tracer tout le contour sans dépassement. La pièce définitive sera confectionnée avec un débordement périphérique de 0,5mm.

**A** partir d'ici, on peut disposer le prototype en carton exactement à sa place définitive sur le coffret. Dans un premier temps on peut le centrer sur la semelle, et avec un poinçon on perce les quatre trous **1** de passage des "vis cloche" qui maintiendront le couvercle. Ensuite, on découpe approximativement la fenêtre d'observation de l'afficheur. Électronique sous tension, par exemple avec l'écran d'accueil, le couvercle en carton étant bien en place, on corrige la position et les dimensions de la lucarne pour en optimiser les contours. En **2** se trouve le tracé optimisé. Du reste, le trou de passage du stylet en **3** n'est pas parfaitement au dessus du bouton de RESET. L'indication **8** précise la correction à effectuer. En **4** on voit nettement le chanfrein réalisé avec une petite fraise conique à 90°. En **9** le trou de centrage est placé au milieu "verticalement" pour des raisons esthétiques. Il est déporté au maximum à droite pour augmenter l'écart avec les deux boutons du clavier. En **6** l'ergot est repéré, et le tracé allongé pour prévoir l'éventuel jeu angulaire. La profondeur avoisine 1,5mm (Moitié de l'épaisseur de la plaque.) ce qui est largement suffisant. On voit bien en **10** que cette alvéole un peu allongée ne traverse pas la plaque.

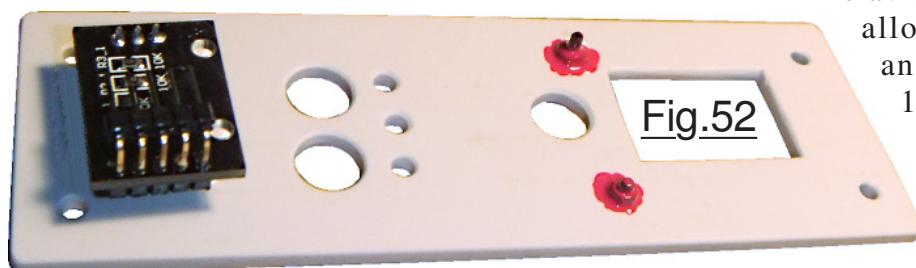


Fig.52



Traînés d'accessoires ces petits objets n'en sont pas pour autant des arguments de vente. Ils sont pratiquement indispensables. Si la petite station météo est emportée pour la promenade du dimanche en étant configurée en altimètre, il sera indispensable de lui fournir une énergie électrique sous la forme d'une petite pile de 9V par exemple. Vous pouvez aussi la brancher sur la

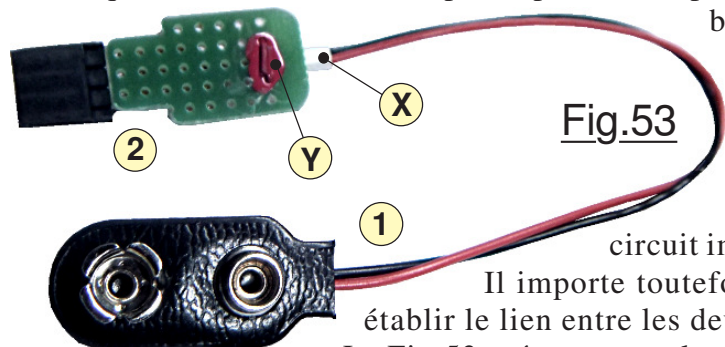


Fig.53

batterie au plomb de votre automobile qui pour la circonstance sera déposée du moteur et rangée dans le sac à dos, mais certains esprits chagrins vont objecter que c'est un tantinet pesant.

Dans l'optique d'une autonomie complète, il suffit d'utiliser le petit connecteur situé sur le circuit imprimé principal sous le module NANO Arduino.

Il importe toutefois de se confectionner un petit adaptateur pour établir le lien entre les deux picots HE14 et les deux bornes de la pile 9V. La Fig.53 présente un tel cordon réalisé à partir d'un coupleur de pile filaire

du commerce 1, complété par la fiche 2. Il saute aux yeux que la fiche 2 est un composite à base d'un connecteur HE14 femelle à trois broches soudé sur une minuscule chute de plaque cuivrée préperçée. La ligne filaire issue de 1 est protégée par un petit tronçon de gaine en X. Cette dernière est immobilisée coté cuivre par un petit pont en fil rigide replié en zone Y sur le dessus comme une agrafe. Les deux parties pliées sont agglutinées en Y avec une colle de nature "TOP SECRET DÉFENSE". Cette attache est parfaitement visible sur la Fig.54 la fiche étant branchée sur le connecteur. On distingue parfaitement la petite gaine de protection au départ du fil. Marquée sur l'élément 8,2V en cours d'usage, il s'agit d'un petit accumulateur rechargeable qui accuse les cycles, il ne lui reste plus que 41mAh de capacité soit environ une heure et vingt minutes d'utilisation : Pas assez pour assurer une grande randonnée.

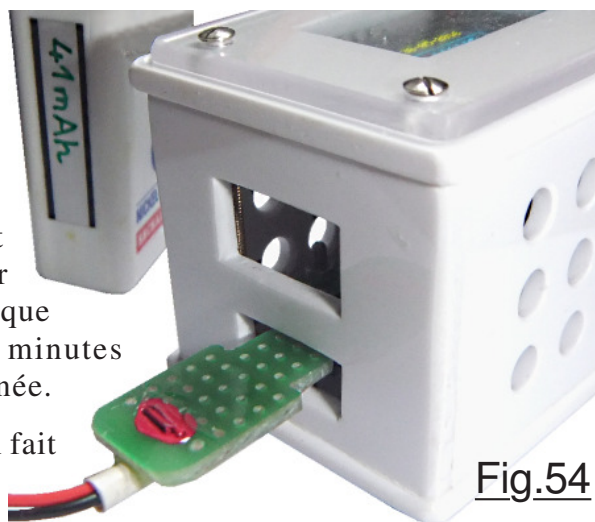


Fig.54

L'accessoire de branchement en cours de description fait partie intégrante de ces petits dispositifs qui sont entassés dans les tiroirs pour effectuer diverses expériences avec Arduino. Ils sont manipulés à tout va et de se fait doivent se montrer robustes. Dans cet esprit, on retourne la petite station météo pour l'observer Fig.55 coté semelle. Les deux fils du coupleur de pile sont soudés sur les picots du connecteur HE14. Puis la petite gaine de protection à la sortie est assurée par le petit pont en fil rigide. Notez que ce fil rigide ne fait pas de

court-circuit entre les deux pistes cuivrées soudées sur les fils car les bandes conductrices sont séparées. Quasiment ritournelle, tout le dessous est ensuite complètement noyé dans de la colle Araldite qui en assure la protection mécanique, et fait office d'isolant électrique pour éviter tout contact intempestif avec un élément quelconque lors des manipulations expérimentales ou de mise au point.

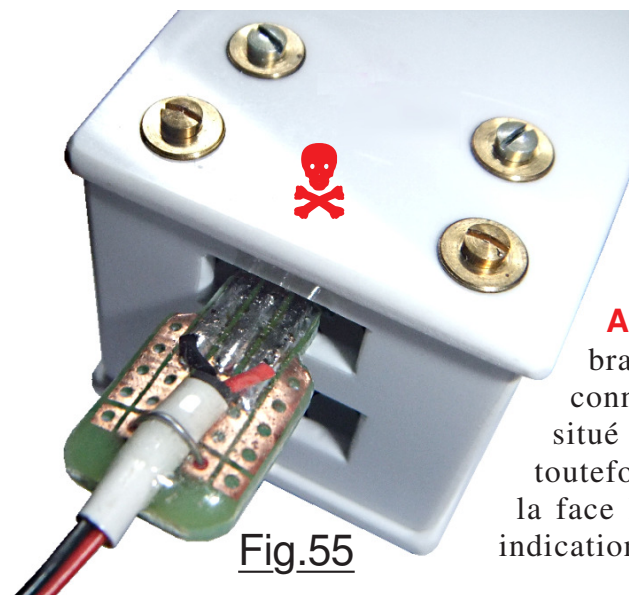


Fig.55

**ATTENTION :** Rien n'empêche d'effectuer un branchement inversé de la fiche quand on l'insère sur le connecteur interne. Bien que le régulateur tripolaire intégré situé en aval accepte en principe ce type de bêtise, il ne faut toutefois pas tenter le diable. Il serait plus sage de compléter la face latérale du coffret de la petite station météo par une indication de polarité, surtout si elle sert souvent d'altimètre.

Vraiment petite cette NANO station météo tenue en main sur la Fig.56 et je vous assure que je n'ai pas une "paluche" qui pourrait me servir de raquette. La photographie montre assez bien la fiche de liaison avec l'accumulateur qui assure l'autonomie électrique de l'appareil. Cette fiche dépasse exagérément et dégrade de façon significative la miniaturisation de l'ensemble. Par ailleurs, cette verrue qui "pendouille" n'en fait pas un dispositif particulièrement "portable". Il serait infiniment plus judicieux de prévoir sous la semelle un compartiment à piles. Des types AAA confèreraient une plus grande autonomie, et le tout, entièrement fermé pourrait se targuer sans abuser, du qualificatif de

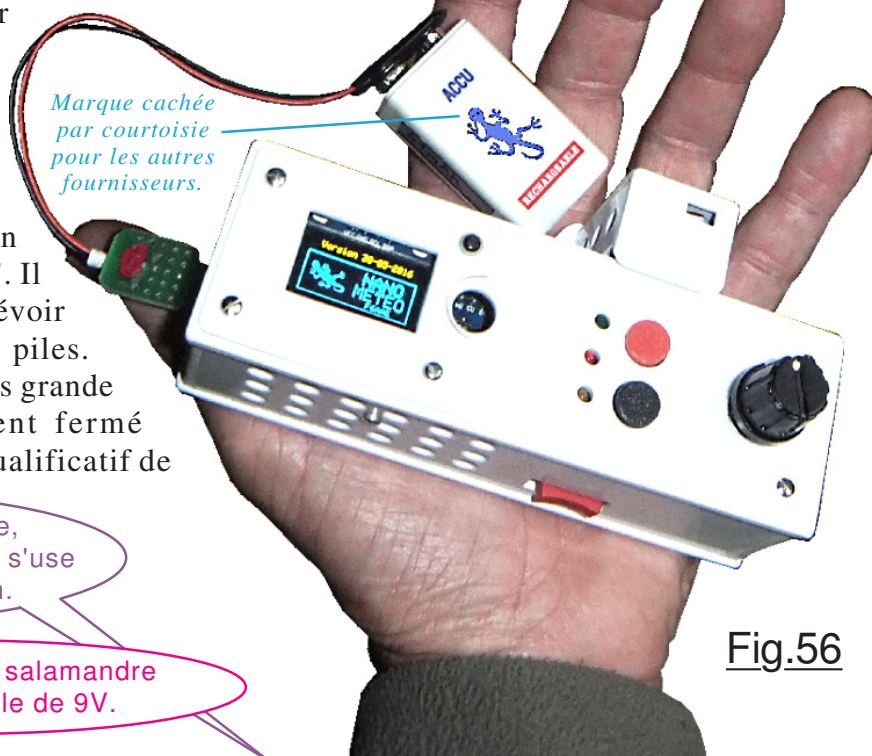


Fig.56

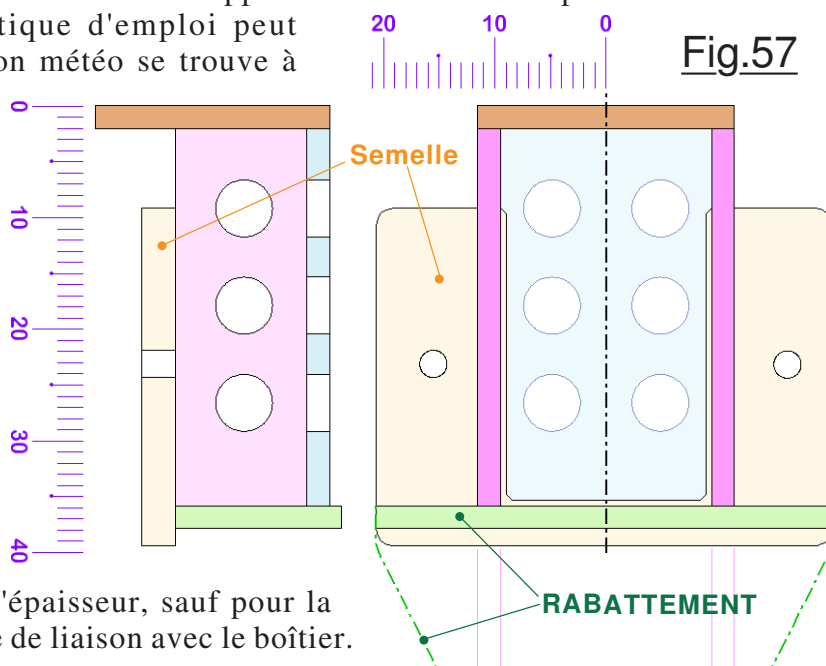
Ben c'est une nouvelle marque, l'ÉNERGIZALAMANDRYNE qui ne s'use que s'il pleut à plus de 5000m.

Hé Pépé, Ya une salamandre sur la petite pile de 9V.

nomade. N'ayant pas pour vocation la fonction d'altimètre podométrique et autres variantes cardiotruc, mon prototype ne sera sauf exception utilisé que sous la forme d'une station météo à enregistrement sur long intervalle de temps. (*Je dispose déjà d'un appareil mémorisant sur 24h.*) Du coup, je me suis contenté de prévoir les essais sans pour autant augmenter le volume et la complexité par ajout d'un compartiment à piles. À chacun de choisir sa version, mais ajouter un coupleur de piles spécifique (*Ou le fabriquer.*) et l'enfermer n'a vraiment rien de sorcier, c'est du petit bricolage ...

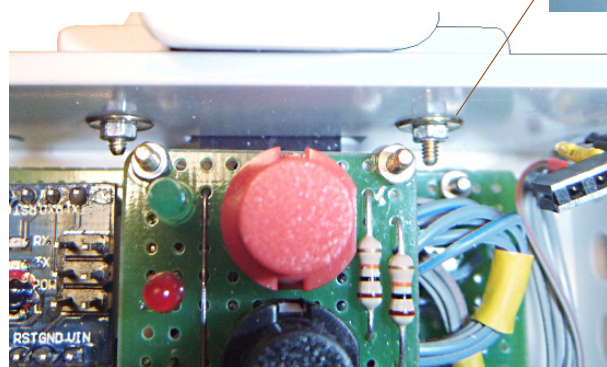
### Le petit capot de protection des capteurs.

Honnêtement, c'est plus une option qu'un accessoire. Pour le mettre en place il faut déposer le couvercle. Une fois avoir refermé le coffret, on peut considérer qu'il restera définitivement sur le boîtier. Dans cette configuration, on obtient un appareil totalement transportable à vocation migratoire. Cependant une autre optique d'emploi peut emporter le suffrage. Celle où la station météo se trouve à poste fixe dans une pièce "confortable" pour sa consultation périodique. Les capteurs sont déportés à l'extérieur ou dans un local éloigné. Dans ce cas, le protecteur n'est plus très utile, par contre il faut prévoir un moyen de télémesures filaire ou par ondes hertziennes. Largement détaillé en Fig.35 inutile d'en tartiner encore des lignes et des lignes. Un simple petit dessin Fig.57 en deux vues **réalisées à l'échelle des dimensions** suffira pour vous inspirer à créer le votre. Contrairement au coffret, ce petit élément est réalisé en plaques de polystyrène choc de 2mm d'épaisseur, sauf pour la plaque de base orange servant de semelle de liaison avec le boîtier.





Une dernière petite image en Fig.58 lorsque le protecteur est immobilisé sur le coffret. Les deux petits boulons de liaison de diamètre nominal  $\phi$  2mm sont bien visibles. L'espace entre la cloison et les circuits électroniques est largement suffisante pour pouvoir utiliser une



Liaison entre le protecteur et le coffret

Fig.58

Connecteur du codeur rotatif

pince précelle. Cette dernière aide à insérer les rondelles plates sur les vis et à tenir les écrous au moment du serrage. Le connecteur du codeur rotatif peut facilement être dégagé du boîtier pour le brancher.

### **Liaison avec des capteurs situés à distance.**

Envisager de mesurer les paramètres à grande distance, par exemple dans un local situé à cinquante mètres de NANO MÉTÉO imposera vraisemblablement de réaliser un dialogue par voie hertzienne en VHF ou UHF. De nombreux modules commerciaux prévus pour des échanges de type série sont disponibles en ligne. Ils sont très simples à mettre en œuvre, obligeant toutefois à ajouter une petite couche de programme dans le croquis de NANO MÉTÉO. Comme ce volet n'a pas pour but de développer une telle application, les dispositifs hautes fréquences seront passés sous silence dans ce tutoriel. En contre partie, si l'on peut se contenter d'un éloignement de quelques mètres, un petit câble filaire est largement suffisant. C'est l'option adoptée pour mon usage personnel. Comme c'est le cas pour l'adaptateur de la Fig.53 cette ligne sera aussi considérée comme un outil disponible pour les opérations expérimentales avec Arduino. Donc ... il faut du solide !

Elle a fière allure la limande de la Fig.59 constituée d'une nappe plate de fils pour ordinateurs et de deux prises FEMELLES ! C'est une question de sécurité. Quand on a branché d'un côté, surtout dans un contexte expérimental où dans le laboratoire cohabitent une foule de systèmes électriques "à l'air libre", vous pouvez être certains que si à l'autre extrémité on a une prise mâle, ses broches vont entrer en contact sur le module qui engendrera le maximum de dégâts. C'est OBLIGÉ !



Fig.59

Donc, c'est pratiquement un principe, une ligne électrique doit présenter à ces deux extrémités des prises femelles. Faire transiter des signaux binaires à fréquence élevée sur des lignes très longues et non blindées n'est pas très conseillé. Un câble blindé avec masse, plus au moins cinq lignes et d'un encombrement réduit, n'est probablement pas très commun, sans compter son prix d'achat. La tentation de tester avec de la nappe d'ordinateur très courante est forte, d'autant plus que la ligne de dialogue est concrétisée par des sorties d'Arduino à niveaux TTL et sous basse impédance. Dans ces conditions on peut espérer que les capacités réparties sur une ligne étendue ne dégraderont pas trop les signaux échangés et que l'immunité aux parasites environnants sera suffisante.

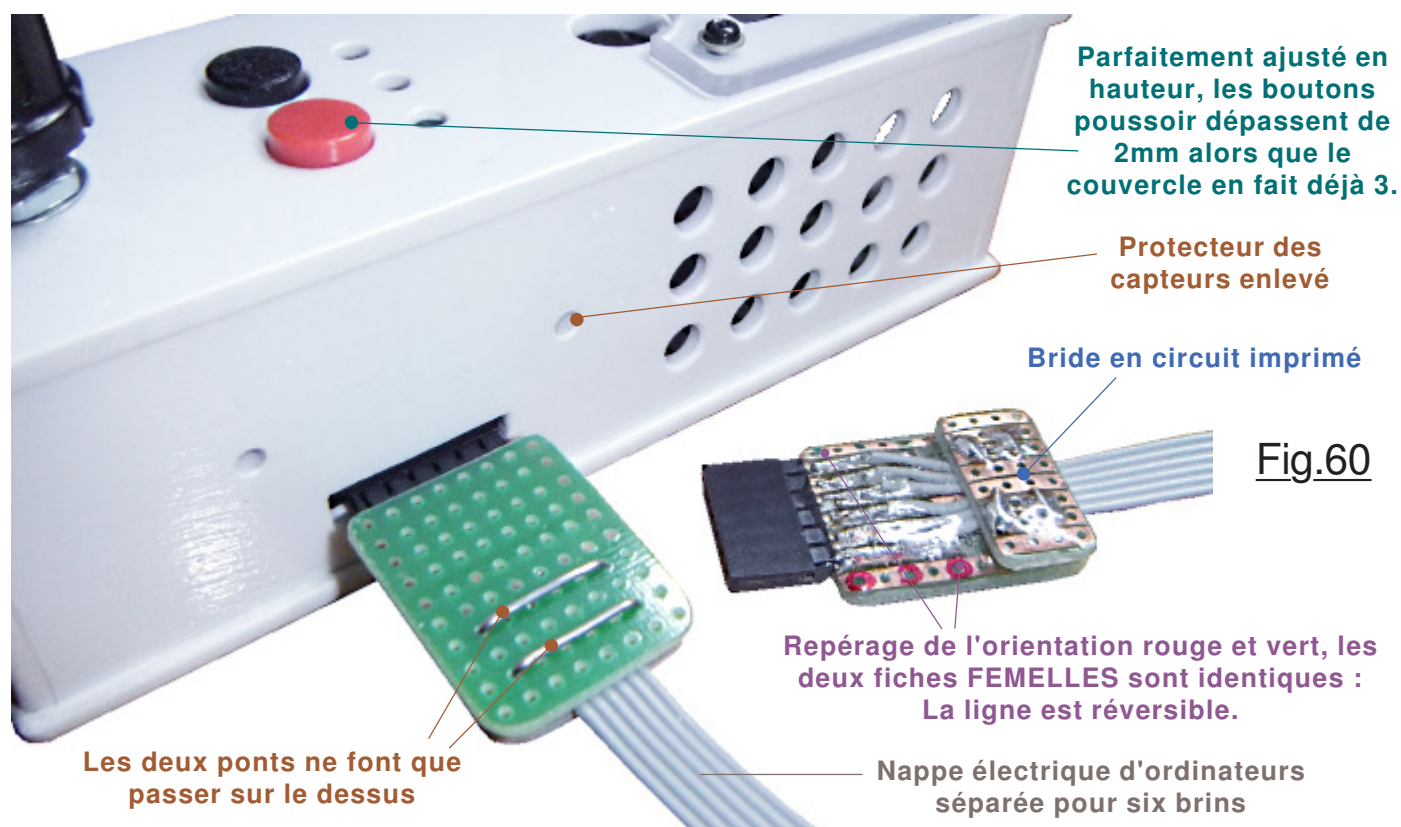


Fig.60

Toutefois, engager les lecteurs dans cette voie sans l'avoir testée sévèrement ne serait absolument pas sérieux. C'est la raison pour laquelle, bien que non indispensable pour mon usage personnel, la ligne présentée dans ce chapitre a été réalisée. Elle permettra bien d'autres expérimentations avec Arduino, indispensable à coup sûr pour sur certaines études futures.

Technique désormais bien au point, on prépare un petit morceau de circuit imprimé prépercé. On coupe les pistes cuivrées aux endroits stratégiques. On soude le connecteur par ses broches longues. Sur ces dernières on soude les fils de la nappe plate. Dans une minuscule chute de plaque imprimée on taille une bride.

Comme la limande est plate, dans ce cas particulier une gaine cylindrique de protection ne convient pas. La bride rigide est plus appropriée. Par rapport à la solution de la Fig.53 on adopte l'inverse. Les fils rigides servant de pont passent librement sur le dessus de la petite plaque et sont repliés sur le dessous coté bride. La vue agrandie de la

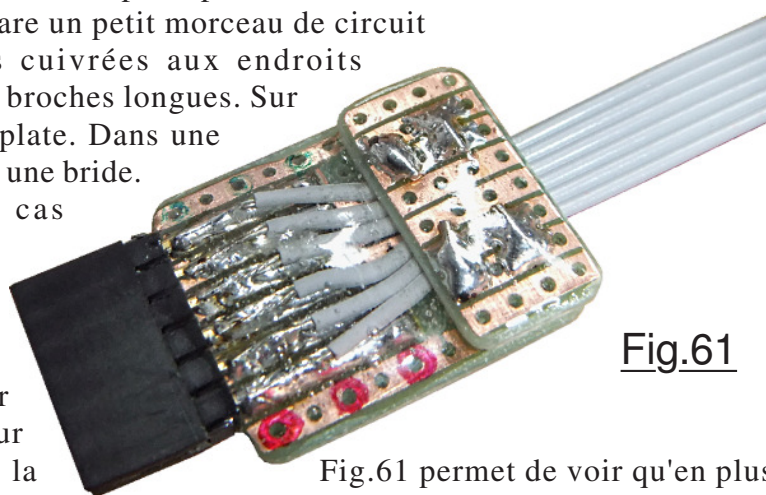


Fig.61

les deux agrafes sont soudées sur les pistes que les deux pistes latérales ont été repérées par des cercles colorés. Il sera ainsi facile de trouver la correspondance des fils lors du branchement. Enfin, ces marques seront indélébiles, car la fiche et la bride sont copieusement incluses dans une gangue protectrice réalisée ... avec de la colle Araldite. C'est solide, c'est costaud, c'est isolant, c'est fonctionnel !

Fonctionnel, est asséné un peu trop rapidement il me semble. Avant de valider cette belle bretelle de raccordement, faut-il qu'elle passe victorieusement les tests. Dans le cahier des charges il est stipulé qu'**elle ne doit pas dégrader les signaux binaires du dialogue série**. Hors une capacité parasite répartie sur une grande longueur peut fort bien diminuer les taux de transition entre le "0" et le "1". La rapidité de changement d'état conditionne la vitesse de transmission possible. Par ailleurs, cette capacité parasite répartie tout le long de la ligne peut engendrer de la **diaphonie** : Le signal d'une ligne filaire se retrouve plus ou moins et se superpose sur celui d'une autre ligne.



Ce phénomène existe forcément. Si l'onde parasite est très inférieure à celle qui est valide, surtout dans un contexte binaire de tout ou rien, elle passera totalement inaperçue. Reste à le vérifier. Autre particularité des lignes longues et plates : L'**induction parasite**. Si les fils du toron sont torsadés, on augmente très légèrement la capacité répartie. Par contre, on élimine radicalement le problème de l'induction parasite. De quoi s'agit-il ?

**U**ne ligne de dialogue série comporte forcément deux fils, celui qui véhicule le signal et celui qui sert de référence zéro pour le potentiel, c'est à dire la masse **GND**. Hors ces deux fils sont obligatoirement "fermés" aux deux extrémités par l'électronique qui parle, et celle qui écoute. On obtient une spire magnétique très plate, mais comme sa longueur est significative, sa "surface" n'est plus négligeable. Nous savons tous qu'**une spire magnétique va transformer en courants induits toute variation de champ magnétique dans laquelle elle se trouve**. Les pics des tensions induites seront d'autant plus virulents que les variations de magnétisme seront brutales. L'orage, le démarrage de moteurs électriques, l'allumage d'un spot lumineux puissant sont des sources diaboliques pour torturer l'électronicien. *(Et encore, je n'ose même pas aborder le problème des ondes stationnaires qui peuvent se révéler redoutables avec des signaux de type "tout ou rien" !)*

**S**eule façon de tester la validité d'une solution quelconque : Caricaturer l'agression et observer le comportement de l'éprouvette. La première approche consiste à construire une ligne longue, plus étendue que celle dont on aura besoin, **pour amplifier volontairement les problèmes potentiels**. Ce n'est pas spécialement évident sur la Fig.59 car la ligne est enroulée, mais sa longueur fait quatre mètres. La station météo est configurée en enregistrements rapides : **Aucun point aberrant** n'apparaît durant plusieurs heures de fonctionnement. C'est bon signe, le dialogue entre NANOMÉTÉO et les capteur s'organise correctement, tout semble aller pour le mieux dans le meilleur des mondes. Ce n'est toutefois pas suffisant pour prouver la fiabilité de cette ligne de transmission. Probablement que les fronts "verticaux" doivent être altérés par la constante de temps résultant des capacités réparties, que les transitoires doivent générer des oscillations amorties induite par les fréquences de résonance issues des inductions parasites. Ces dégradations ne semblent pas altérer le dialogue série. C'est un peu normal, car l'on ne fait pas transiter des signaux analogiques faibles, mais **des "0" et des "1" précisément inventés pour palier ce type de difficulté**. Reste à prouver que "le bruit de fond" est bien en dessous des marges de perturbations tolérées par de l'I2C.

**Abuser, exagérer, c'est faire preuve de sérieux !**

**P**rovocation ! vont hurler les personnes vertueuses ... ce titre est un scandale !

Ben oui, un chtipeu je l'avoue. Encouragé par ces premiers résultats, la sagesse pousse à dépasser les bornes. OUI, abuser pour une fois peut s'avérer prudent ! L'idée consiste à pousser le bouchon volontairement trop loin. En fait de bouchon, c'est des capteurs dont il est question ici. Encore plus, jusqu'à faire déborder le verre, c'est à dire augmenter encore la longueur de la ligne de dialogue de 4,8 mètres. Une deuxième rallonge copiée sur la première a été réalisée pour compléter le matériel expérimental de torture électronique-robotique. Par rallonge, il faut comprendre que l'une des fiches est de type mâle pour s'insérer directement sur la ligne initiale. C'est long, outrageusement long. Un rapide calcul du genre  $4 + 4,8 = 8,8$  mètres de distance n'est pas très parlant. Pourtant, grâce à cette étendue filaire, on peut placer la station météo dans la cuisine pour la consulter au "pti deuje", et les capteurs trois étages plus haut sur la toiture.

**- Vont avoir le vertige les captulateurs !**

Et bien pas de problème, le dialogue s'initialise immédiatement et dans le plus grand silence malgré la distance, on entend rien, mais ça cause du pays en binaire. C'est magique le blablabla série. Pour valider avec crédibilité une telle longueur de ligne, des essais poussés doivent impérativement compléter la simple mise en service de cette longue limande. NANO MÉTÉO sur le bureau, ligne branchée qui fait tout le tour du laboratoire, on peut commencer la campagne de validation.



Hé Papy, on est encerclé par une longue limande électrifiée !

C'est pour empêcher la salamandre de se sauver fiston !

*Jamais assez prudent quand on va rendre public.*

Entre réaliser un petit appareil pour son compte personnel, et le proposer en ligne pour encourager des lecteurs à s'engager dans la même voie il existe un gouffre. L'internaute peut être séduit par le descriptif, et faisant confiance à l'auteur va investir du temps et des finances. **La confiance se mérite**, et moralement l'auteur doit mettre en œuvre tous ses moyens techniques pour traquer impitoyablement toutes les faiblesses de l'objet technique dont il propose la description, car **le bon fonctionnement d'un prototype ne prouve absolument pas sa reproductibilité**. Nous savons que les composants qui entrent dans la composition d'un ensemble sont tous affectés d'une dispersion de caractéristiques inhérente aux processus industriels. Imaginez que deux ou trois composants sur le prototype ont été dimensionnés avec des valeurs critiques. La chance a favorisé le concepteur, et les "plus" de l'un compensent les "moins de l'autre". Le bébé fonctionne à merveille.

Statistiquement, il n'y a aucune raison pour qu'il en soit ainsi chez les nombreux lecteurs, avec pour conséquences des dysfonctionnements et des déconvenues. Hors celui qui a été tenté par la réalisation décrite n'a pas forcément les compétences pour trouver l'origine du problème. Il ne dispose pas obligatoirement non plus des moyens d'analyse qui sont à la disposition du "créateur". C'est la raison pour laquelle quand on décide de publier une description, moralement on s'engage à pousser le plus loin possible la recherche d'éventuels problèmes pour s'assurer d'une grande tolérance dans le comportement du système proposé, et ce aussi bien logiciel que matériel. C'est dans cette optique que la ligne a été portée à presque neuf mètres de long et qu'elle va être soumise à un interrogatoire "virulent". Ce chapitre n'est qu'une façon de vous présenter quelques façons de procéder quand on aborde le chapitre de la **fiabilité**. D'une façon très générale :

- **On exagère techniquement le prototype pour en caricaturer le comportement,**
- **On injecte un signal "nerveux" à son entrée.** (*Ici le début de la ligne de transmission.*)
- **On observe à la loupe ce que devient le signal en sortie.** (*L'extrémité avec les capteurs.*)
- **On évalue le taux de dégradation du signal et on compare à la tolérance acceptable.**

Clef de voûte de tout laboratoire d'électronique, que ce soit en loisir ou en professionnel, l'oscilloscope est de loin le stéthoscope le plus efficace pour "voir" ce qui se passe. Bien que la majorité du temps un simple multimètre bas de gamme se montre largement suffisant, dès que l'on monte un peu en technicité, cet appareil s'avère indispensable. Le mien accuse plus de quarante années de bons et loyaux services. Quand j'en ai fait l'acquisition, son prix d'achat était très déraisonnable au regard de mes maigres revenus à cette époque. Si j'ai consenti ce sacrifice, c'est que moralement c'était acceptable, n'ayant pas encore de charge de famille. Je n'ai JAMAIS REGRETTÉ CET INVESTISSEMENT. Quarante années, c'est une éternité par rapport à l'évolution des techniques et des besoins de diagnostic ... et bien il reste d'actualité en performances malgré son volume et son tube cathodique ringard. Si je me suis fourvoyé à ajouter ce petit chapitre, c'est pour attirer votre attention sur un point que je crois très important. Le jour où vous déciderez de faire l'acquisition d'un tel appareil, consacrez vraiment du temps aux choix de celui qui sera l'élu ...

**Admission du patient dans la clinique.**

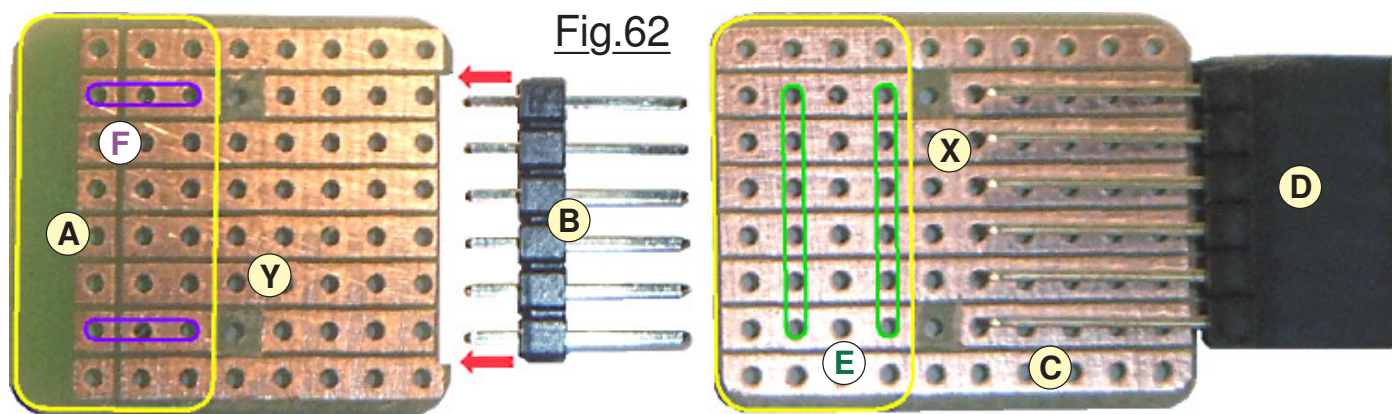
Avant de soumettre notre longue longue ligne de télémesures à des tortures électrisantes, commençons par faire connaissance. En effet, bien que copiée sur sa sœur Femelle/Femelle la rallonge présente quelques petits détails de fabrication qui peuvent vous aider. Par ailleurs, **il faut effectuer la campagne de validation en situation réelle**, c'est à dire avec en bout de ligne les capteurs tels qu'ils seront utilisés dans la pratique. Comme il s'agit d'une application "sédentaire", on peut se permettre un encombrement plus important. Alors autant vous proposer une variante qui intègre une LED de sentinelle "déportée", mais également une interface de pilotage d'un éventuel système de chauffage fonctionnant en logique positive. Le frère à souder reprend du service ...



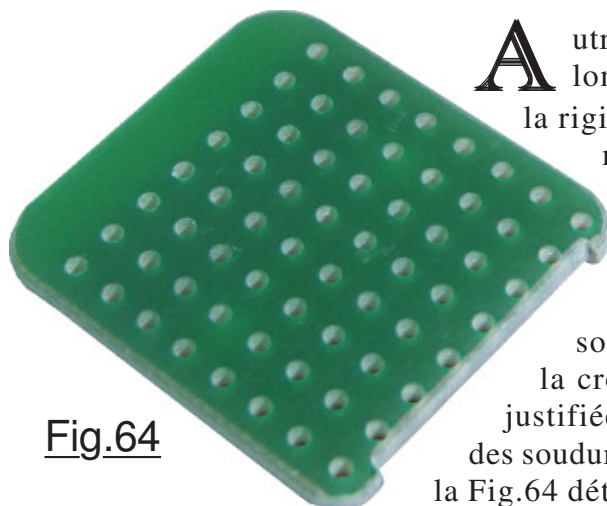
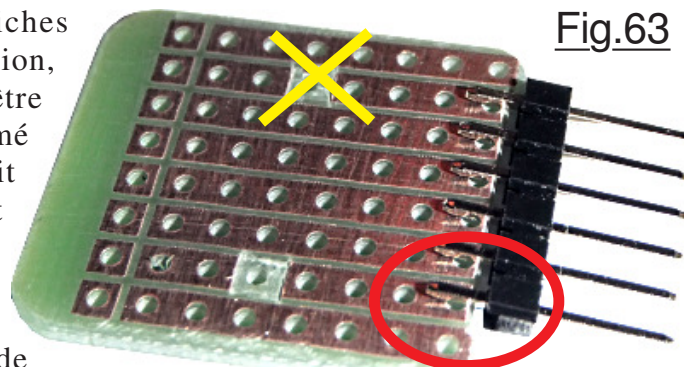
Hé Papy, ya la salamandre qui s'est sauvée de la limande électrifiée !

Ben oui fiston, mais avec tout ce bla bla bla elle a fini par trouver un passage !



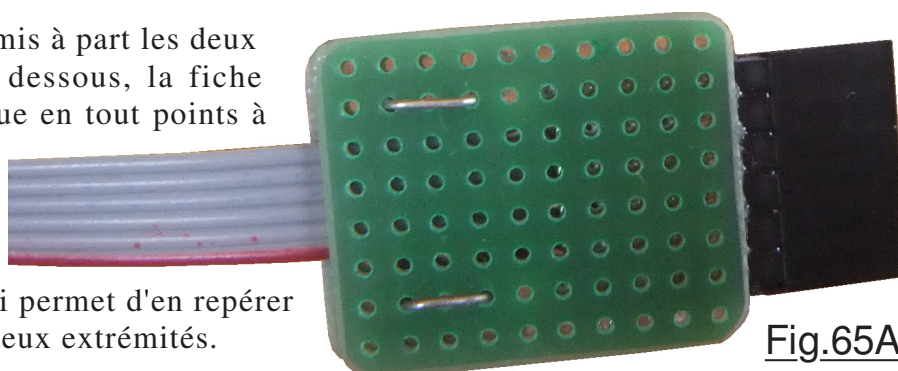


**B**ien qu'entièrement copié sur les deux fiches femelle de la première ligne de transmission, quelques subtilités dans la confection méritent d'être soulignées. En **A** nous avons le petit circuit imprimé pour la fiche mâle de la rallonge avec en **B** le petit connecteur à y souder. En **C** le circuit imprimé est copié intégralement sur celui de la ligne Femelle/Femelle. Pour mémoire, sur ces deux circuits est ajouté en surimpression jaune le contour de la petite plaque de circuit préperçé qui va servir de bride à la limande. Cette bride est immobilisée sur **C** par les deux ponts **E** dessinés en vert et soudés sur les pistes cuivrées de la petite plaque à 4 x 8 trous. Si ces deux ponts en fil de câblage rigide touchent les pistes cuivrées en traversant **C** ils peuvent créer des courts-circuits entre les broches extrêmes de **D**. C'est la raison pour laquelle les pistes cuivrées ont été coupées en **X** pour isoler. On peut toutefois simplifier. Au lieu de faire des ponts "verticaux", on peut les faire "horizontaux" comme en **F**. Du coup ils sont plus courts, et surtout les isolements tels que ceux en **Y** ne sont plus justifiés. On gagne en simplicité. (*Comme quoi copier n'est pas malin !*)

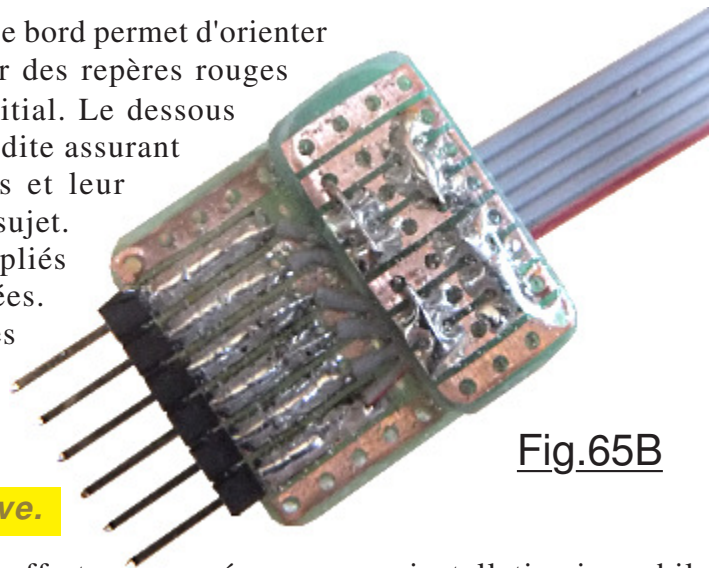


**A**utre différence, le connecteur femelle **D** présente des broches longues. Soudées sur une longueur de presque cinq trous, la rigidité et la solidité sont évidentes. Par contre, les broches mâles en **B** sont bien plus courtes. Leur encastrement fait à peine la longueur d'un trou. Pour améliorer la rigidité une petite échancre est pratiquée sur la plaque **A** permettant un encastrement symbolisé par les deux flèches rouges. Il faudra particulièrement soigner les soudures. La Fig.63 présente la fiche mâle en préparation, la croix jaune précisant que la coupure des pistes n'est plus justifiée. Dans le médaillon rouge on voit bien que la longueur des soudures sera faible et l'on devine l'échancre. Pour son compte la Fig.64 détaille le dégagement, la petite pièce étant vue par dessus.


Il est manifeste sur la Fig.65 que mis à part les deux ponts pour tenir la bride située dessous, la fiche femelle de la rallonge est identique en tout points à celles de la ligne de 4 mètres. Elle en adopte exactement les mêmes dimensions. Notez que la limande est séparée en conservant sur l'un des fils extrêmes le bord rouge qui permet d'en repérer immédiatement l'orientation aux deux extrémités.

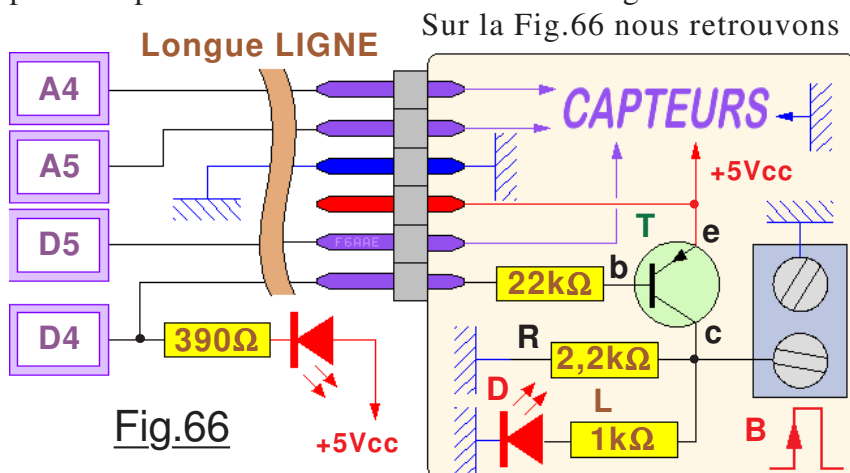


Colorié en rouge le fil de la nappe situé sur le bord permet d'orienter correctement la ligne sans avoir à tracer des repères rouges comme c'était le cas sur la Fig.61 du cordon initial. Le dessous est toujours englué dans sa gangue de colle Araldite assurant la protection mécanique des petits fils soudés et leur isolement électrique. Rien de vraiment neuf à ce sujet. Les petits ponts de fil pour serrer la bride sont pliés (*Tels des agrafes.*) et soudés sur les pistes cuivrées. C'est fait, notre "exubérante" ligne de télémessures est complète. Il ne nous reste plus qu'à agencer le module qui sera branché à son extrémité et l'on va pouvoir passer aux essais.



### **Support des capteurs et logique positive.**

Variante du petit module qui se fixe sur le coffret,  prévu pour une installation immobile les contraintes d'encombrement sont moins restrictives que celle de l'option autonome. On en profite pour expérimenter une version dont le signal pour l'alarme sera en logique positive. Cette petite interface supplémentaire autorisera le pilotage direct de relais statiques dont il était question à la fin du didacticiel relatif à PICOSYNTHE. Cette option n'a strictement rien d'obligatoire. Nous pouvons parfaitement réutiliser en bout de ligne le circuit de la Fig.32 strictement sans modification.



état logique "0". Quand **D4** passe en alarme à l'état "0", le courant de base à travers la résistance de **22kΩ** sature **T** dont le collecteur **C** peut alors fournir un courant important depuis le **+5Vcc**.

- *Publicité mensongère, ce n'est pas Musclor ton transistor !*
- *Ha bon pourquoi tu t'excite d'un coup, c'est la sale Amandre qui t'a électrisé ?*
- *Hé bé non, mais oser parler de courant important c'est de l'abus.*



Mince alors, le complot est découvert, il faut me justifier : Le transistor **T** qui sur le prototype est un 2N2907 pourrait drainer un courant collecteur allant jusqu'à 600mA dans un fonctionnement en mode saturé / bloqué. Compte tenu de son faible gain en courant, la résistance de base a été sélectionnée à **22kΩ** pour volontairement limiter à 50mA environ son courant de collecteur quand **D4** "monte à" +5V. Ce choix est justifié pour deux raisons. La première résulte du courant maximal que l'on ne doit pas dépasser tant sur les connecteurs HE14 que dans les petits fils de la nappe de liaison. Et surtout il ne faut pas oublier que le courant consommé par le module des capteurs est entièrement fourni par le minuscule connecteur USB de la carte NANO si vous avez opté pour un bloc secteur de ce type. Un court-circuit en sortie sur le bornier **B** sera donc sans conséquence sur le matériel. Du reste vous pouvez observer sur la Fig.66 que pour limiter autant que faire ce peut le courant consommé par la LED témoin, sa résistance de limitation du courant **L** est augmentée à **1kΩ**. (Par rapport à celle branchée sur **D4** qui ne fait que 390Ω.) La diode électroluminescente sélectionnée est donc un modèle à bon rendement pour conserver un éclairage important.



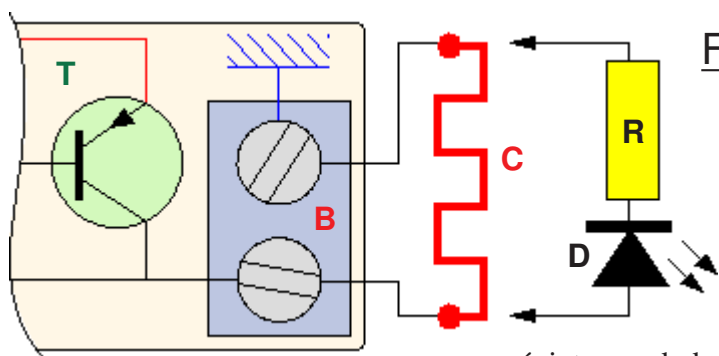


Fig.67

Le transistor **T** soudé sur le circuit présente, à un débit collecteur de 50mA, un gain en courant  $\beta$  qui avoisine 240 ce qui est déjà considérable. Pour ne pas dépasser ce courant, il faut que dans la base on limite le débit à :  $50\text{mA} / 240 = 0,2\text{mA}$  soit environ  $200\mu\text{A}$ . La sortie **D4** passant à l'état "0" donc pratiquement au potentiel de **GND**, on a une tension d'environ 5V aux bornes de la

résistance de base. (*Oublions la tension Émetteur/Base.*) On peut

calculer la valeur de cette résistance :  $R = 5 / 0.0002 = 25000$ . On adopte pour cet échantillon une valeur standard de  $22\text{k}\Omega$ . En fonction du gain en courant  $\beta$  que présentera l'échantillon PNP que vous aurez intégré dans votre interface, il sera probablement obligatoire de modifier la valeur de la résistance de base. Pour en déterminer la valeur, la technique est simple. Vous branchez sur le bornier **B** un ampèremètre sur le calibre 100mA. Il se comporte comme un court circuit. Puis en résistance de base vous débutez la manipulation avec une valeur très exagérée, par exemple  $150\text{k}\Omega$ . Le courant dans l'ampèremètre sera très faible. Il suffira ensuite de diminuer progressivement la valeur de la résistance jusqu'à avoisiner 50mA à 60mA environ. C'est enfantin et sans danger.

**P**assons à l'exploitation automatique de la sentinelle, c'est à dire utiliser le bornier **B** pour piloter une charge extérieure qui se chargera du réchauffage d'un local quand la température devient inférieure au seuil d'alerte. Le principe d'utilisation représenté sur le schéma partiel de la Fig.67 consiste à fournir un petit courant à l'interface de puissance. Quand l'Alarme se déclenche, **T** devient conducteur et débite tout ce qu'il peut dans la résistance de Charge **C**. Maximum 48mA dans

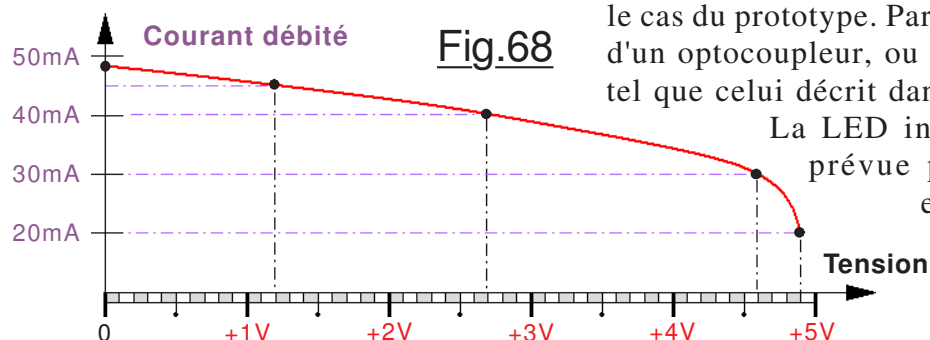


Fig.68

le cas du prototype. Par exemple la charge sera constituée d'un optocoupleur, ou d'un relais statique de puissance tel que celui décrit dans le tutoriel sur PICOSYNTHÉ.

La LED interne **D** d'un tel dispositif est prévue pour un courant nominal. Par exemple 20mA. C'est à vous de déterminer la valeur de la résistance **R** qui limitera le courant dans **D**. La Fig.68 présente la variation de tension sur le

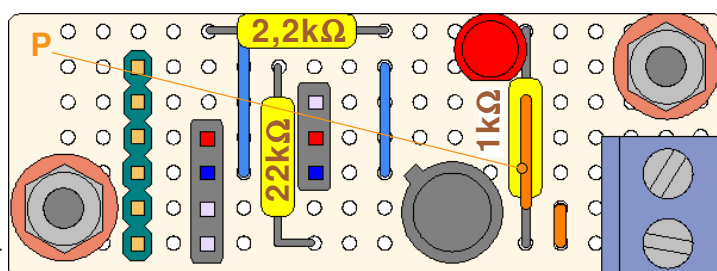
bornier **B** en fonction du courant débité dans la charge extérieure **C**. À 40mA la tension chute à +2,7V. Cette valeur commence à s'approcher de la tension de conduction de la LED rouge **D**. Cette dernière voit sa luminosité diminuer notablement. Vers 45mA elle s'éteint complètement. On est ainsi averti quand l'interface externe consomme un courant exagéré par rapport aux possibilités du module. Naturellement si une interface de puissance se montre plus exigeante, **T** ne fournira que le signal logique, ce sera à l'électronique extérieure de piloter ses divers effecteurs avec les courants idoines.

### Aspect pratique de l'interface et des supports des deux capteurs.

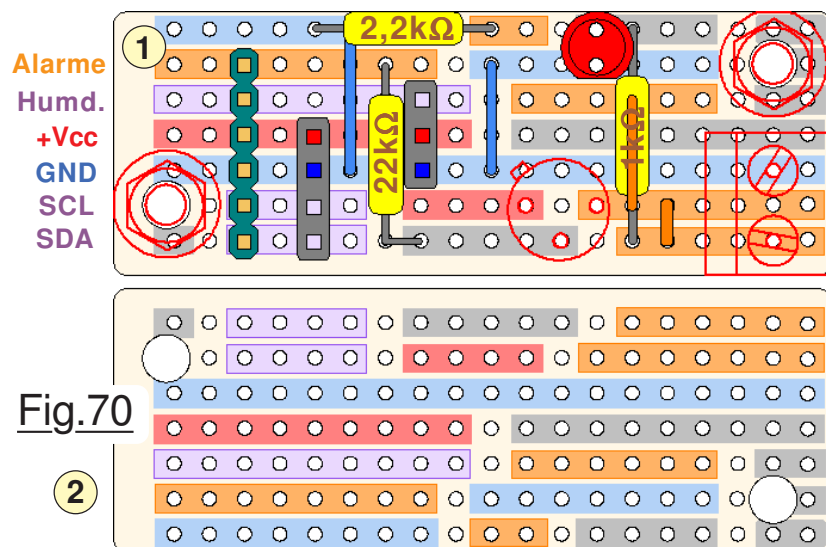
**L'**avantage d'un didacticiel, c'est que l'auteur s'est torturé les méninges pour concevoir le circuit imprimé. Vous n'avez plus qu'à copier. La technologie ne change pas. Plaque de prototypage préperçée, bandes cuivrées coupées à la demande, on tasse, on compresse pour réduire au maximum l'encombrement tout en conservant une grande facilité d'élaboration. Bref : Une ritournelle ...

La Fig.69 donne une impression de grosseur pour les deux vis qui supportent le circuit dans son boîtier de protection. Utilisant des entretoises du commerce, les deux liaisons placées en diagonale font usage de vis  $\phi 3\text{mm}$ . Le pont **P** de quatre trous de long est en réalité placé sous la résistance de  $1\text{k}\Omega$ .

Fig.69

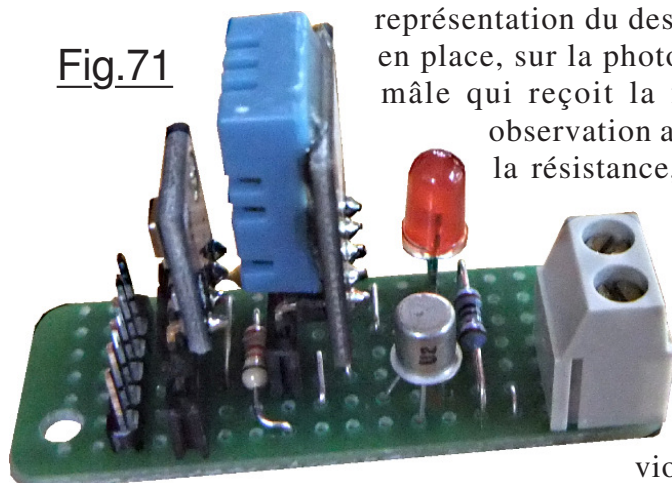


Toujours à grande échelle et avec les conventions habituelles, la Fig.70 présente en **2** le circuit imprimé coté pistes cuivrées, et en **1** la vue de dessus avec certains composants représentés en mode filaire. L'ordre des fonctions sur le connecteur d'arrivée de la ligne de télémessures est bien entendu exactement identique à celui du connecteur HE14 de départ sur la petite station



météorologique. N'oubliez surtout pas au moment de souder les composants que le petit pont orange est sous la résistance de  $1k\Omega$  de limitation de courant de la LED. Si vous avez mangé la consigne il faudra le placer dessous, et ce serait dommage vu que c'est l'un des rares circuits sur lequel le coté cuivre ne comporte rien d'autre que les soudures. Remarque analogue pour le pont bleu qui passe sous la résistance de  $2,2k\Omega$ . Vu que la miniaturisation ne passe plus au premier plan, on peut facilement opter pour une LED de 5mm de diamètre. Contrairement à la

Fig.71



représentation du dessin de la Fig.70 pour lequel les capteurs ne sont pas en place, sur la photographie d'ensemble de la Fig.71 seul le connecteur mâle qui reçoit la fiche de la limande n'est pas encore servi. Une observation attentive ne permet pas de voir le petit pont situé sous la résistance, (*Pas vu, pas pris !*) mais permet en revanche de

remarquer que cette  $1k\Omega$  est un peu surélevée au dessus de la plaque prépercée. C'est plus facile à discerner sur la Fig.72 pour laquelle le circuit est vu "par la tranche". Une particularité mérite d'être soulignée. Pour augmenter l'espace entre le capteur d'humidité DHT11 et le transistor 2N2907, ce dernier est "décalé" dans le sens de la flèche violette  $\pi$ . Dans ce but, les trois fils du transistor ne

sont pas écartés symétriquement. Celui de droite colorié en jaune reste vertical, les deux autres étant inclinés vers la gauche pour se trouver en face des trous de traversée de la plaquette aux pistes rectilignes cuivrées.

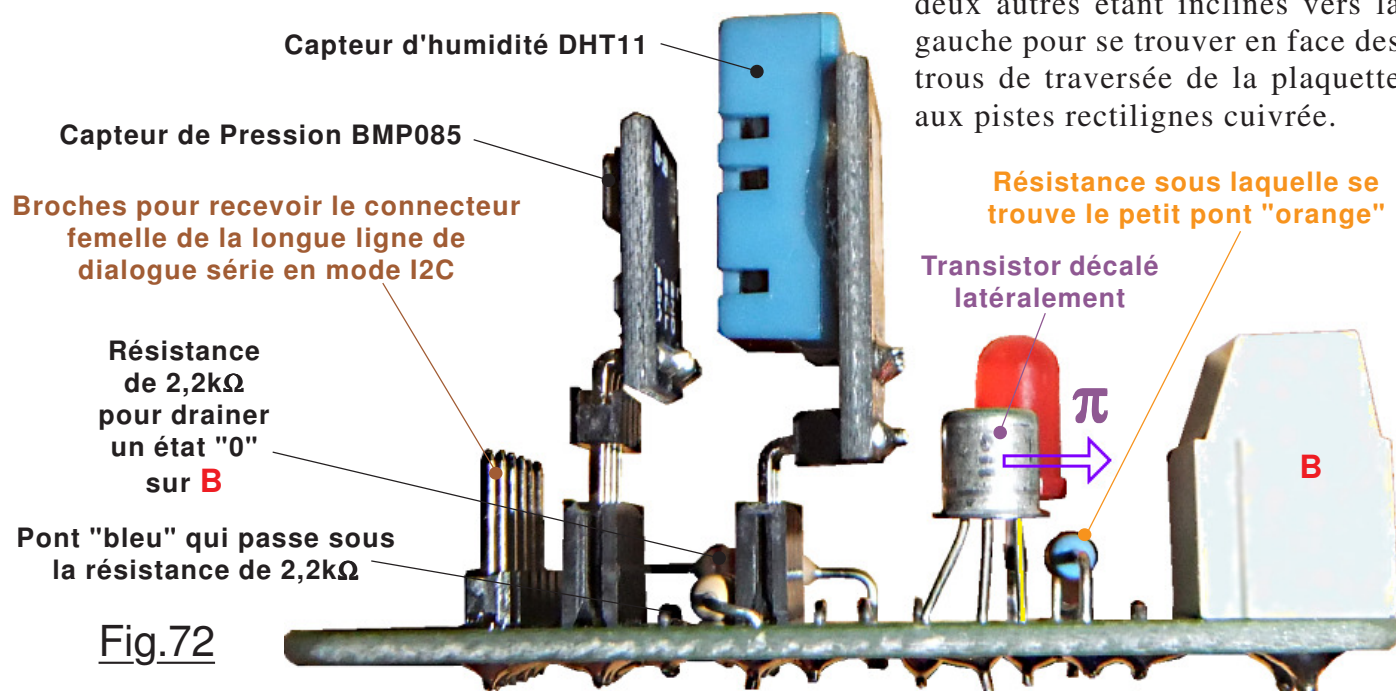


Fig.72



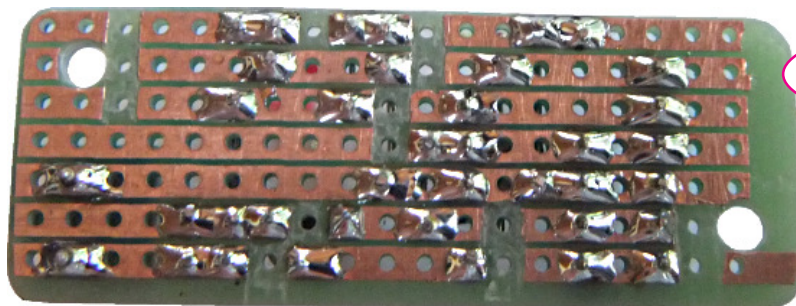


Fig.73

Ben non fiston, sont trop petits.  
Passera pas la salamandre.

Encore deux dernières images pour clore ce chapitre sur la réalisation du petit bloc d'interfaçage et supportant des deux capteurs. Toujours vu par la tranche, de l'autre côté cette fois, la Fig.74 est parfaitement représentative de la façon dont sont conçues les fiches de branchement. La longue nappe plate de six fils arrive en 1. Elle est plaquée sur le petit circuit imprimé qui sert de corps à la fiche par la bride 2. En 3 les deux agrafes, constituées de queues de composants quand on les raccourci après soudure, servent à serrer un peu la bride sur le corps. Ces morceaux de fils rigides sont repliés coté cuivre de la bride 2 et soudés pour pérenniser la liaison. En 4 la colle Araldite se transforme en "inclusion sous plastique". Le capteur d'humidité n'est relié à la plaquette imprimée que par ses trois broches situées vers le bas du petit capot bleu. Détestant les composants qui sont en porte-à-faux, pour rigidifier la liaison mécanique, en 5 la colle Araldite a encore frappé.

(Oui, je sais que j'en tartine allègrement un peu partout, mais figurez-vous que si on les achète par 5000 tubes d'un coup dans le magasin, on a une méga réduction de 0,1% sur la facture !)

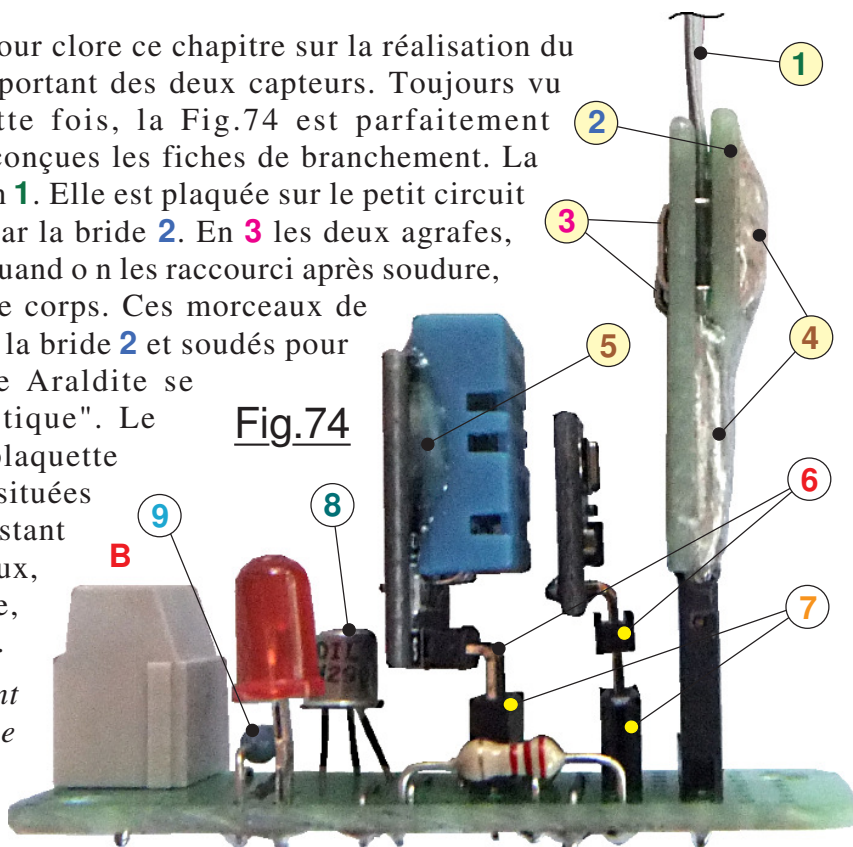


Fig.74

On remarque qu'en 6 les deux modules capteurs sont également en porte-à-faux sur leurs broches HE14. Pour améliorer leur maintien en position, les connecteurs 7 sont des modèles femelles un peu plus hauts que pour les types ordinaires. La hauteur des exemplaires utilisés est d'environ 5,5mm. (Concrètement deux dixièmes de pouces.)

En 8 il est manifeste que le petit transistor T est déporté sur la gauche pour s'écarter plus du capteur d'humidité et ainsi en faciliter la mise en place. Enfin en 9 on retrouve la résistance de 1kΩ un peu surélevée au dessus du petit pont coloré en orange sur le dessin du circuit imprimé. Quand à 10, c'est un repère gratos en promotion !

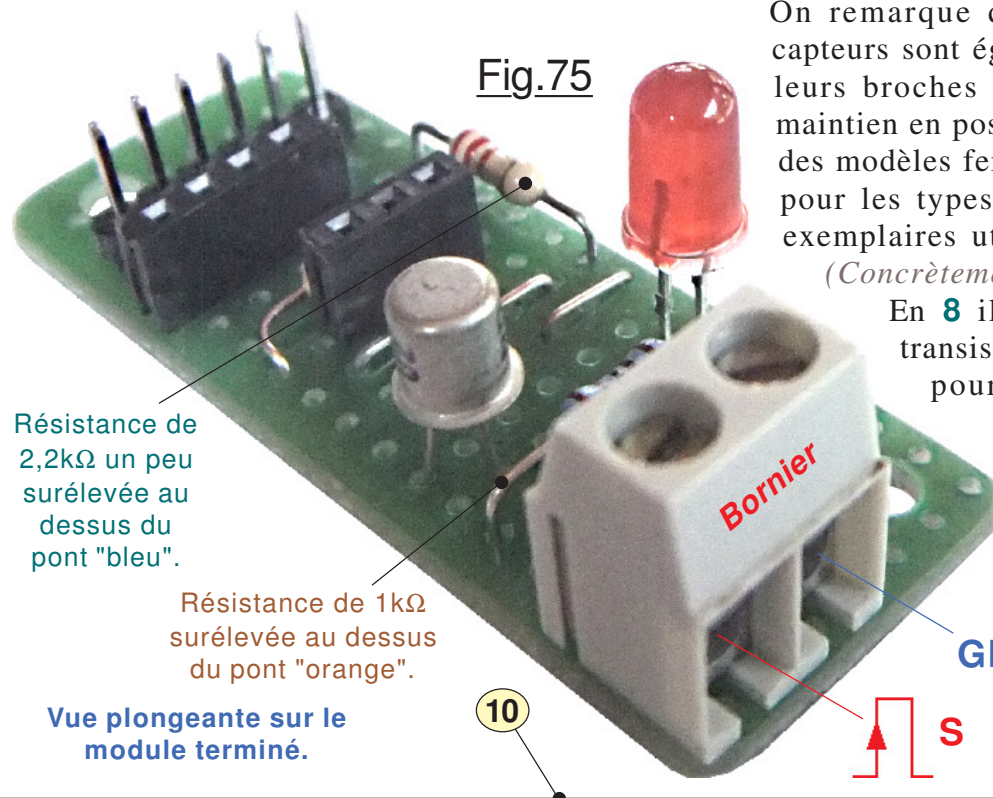


Fig.75

Résistance de 2,2kΩ un peu surélevée au dessus du pont "bleu".

Résistance de 1kΩ surélevée au dessus du pont "orange".

Vue plongeante sur le module terminé.

### Auto oscillations, ondes stationnaires et autres joyeusetés.

**F**ranchement, ce chapitre a autant pour vocation de vous faire toucher du doigt les problèmes que l'on peut rencontrer quand on cherche à transmettre un signal sur une ligne filaire, que pour vérifier un comportement propre et répétitif, si de nombreux exemplaires risquent potentiellement de peupler la planète LudoArduine. Avec la limande étalée linéairement pour former une grande boucle magnétique ou enroulée sur elle même, il ne faut pas constater de dégradation significative du signal qui transite entre ses deux extrémités. Les fronts de commutation binaire doivent rester "verticaux", et si possible sans oscillations amorties sur les transitoires. Pas de diaphonie détectable non plus. La ligne sera alors enroulée autour d'un petit transformateur secteur débitant au nominal sur une charge pour vérifier son immunité au "50Hz". Bref, on va la soumettre à un régime totalement déraisonnable pour en analyser le comportement. On se doute que le signal va forcément se dégrader, reste à voir si dans ces conditions il reste exploitable AVEC FIABILITÉ.

**L'**interprétation de signaux complexes comme ceux issus d'un dialogue I2C n'est pas idéale pour faire émerger des observations les phénomènes critiques. Il est généralement plus probant de soumettre la ligne de transmission à un signal "carré" généré à la fréquence la plus élevée possible avec le microcontrôleur équipant le dispositif en cours de vérifications. Pour la circonstance, dans notre cas c'est **P05\_test\_de\_la\_longue\_ligne\_I2C.ino** qui sera chargé de cette mission. Quatre voies filaires dont trois bilatérales doivent être soumises à la campagne de tests. **SDA** et **SCL** de dialogue avec le capteur de pressions, la ligne pour celui des températures, et la sortie de pilotage de l'Alarme, à sens unique dirigée de l'ATmega328 vers l'interface. On va tester chaque voie séparément, ce qui nous obligera à modifier trois fois ce programme et à le téléverser quatre fois. Pour le jeu d'essais prévu, NANOMÉTÉO est entièrement achevée et bien calfeutrée dans son coffret. Elle est alimentée par la ligne USB qui sert au téléversement des croquis. La ligne de dialogue est également en place dans la configuration décrite dans le chapitre précédent. Enfin, les capteurs sont branchés à l'autre extrémité de la limande, disposés sur leurs supports dédiés.

En résumé : **Le test de validation est effectué en conditions réelles avec le matériel en configuration nominale et "définitive".**

On se doute que les capteurs recevant un signal qui n'a rien à voir avec les protocoles officiels vont ignorer les fausses consignes. Leur présence est impérative, car ils constituent en fin de ligne une charge résistive et capacitive qui influence le comportement de la ligne de transmission.

#### **Un son de cloche : La résonance.**

**F**rappez énergiquement la base d'une cloche avec un marteau. Elle réagit bruyamment et fait entendre sa protestation à des lieux à la ronde. Puis, elle va marmonner de moins en moins fort jusqu'à retrouver son mutisme. Frappée avec énergie, son corps vibre à sa fréquence propre. La note musicale émise dépend un peu du matériau utilisé, (*Avec du caoutchouc mousse la tonalité obtenue ne sera pas géniale !*) et surtout de ses formes et de ses dimensions. Tout élément solide recevant brusquement de l'énergie mécanique se comportera de façon identique. L'énergie emmagasinée se résorbera lentement. Cette diminution progressive est nommée le "**décrément logarithmique**".

- **Hé totoche, avec tout ce baratin tu vas faire rappliquer la Salamandre.**

- **Oups, t'as raison, faisons plus court, mais je crois que c'est trop tard.**

Électriquement on retrouve des comportements absolument analogues. Un conducteur électrique est pénard, au repos et sans électrons sauteurs qui se bousculent dans ses molécules.

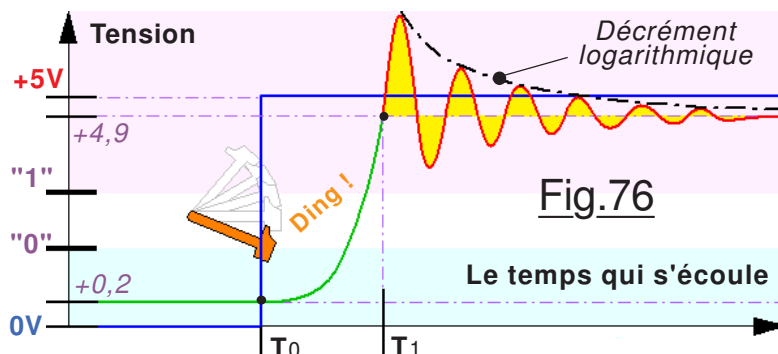
Pafff, brusquement c'est le coup de marteau, un générateur lui cogne +5Vcc d'un seul coup.

Comme la cloche, secoué dans ses entrailles les plus intimes, il va se mettre à vibrer. En réalité ce n'est pas la ligne qui constitue "la caisse de résonance", mais le circuit complet avec capacités parasites, inductance répartie. Bref, le tout matérialise **un circuit LC** qui ne demande qu'à osciller à sa fréquence propre. Puis, le conducteur étant maintenu à une tension constante de +5V, l'oscillation va diminuer d'amplitude car la résistance interne du circuit transforme cette vibration en chaleur. (*Et aussi en ondes dites radio.*) Après quelques oscillations la vibration devient insignifiante. (*En réalité si le circuit est faiblement amorti le nombre d'oscillations peut être notable.*)



### Le marteau électronique.

**I**llusion collective, nous sommes tous persuadés que les machines informatiques fonctionnent en binaire, en tout ou rien, en "0" ou "1". **C'est IMPOSSIBLE.** Aucun mobile ne sera capable quelle que soit la technologie de passer de l'immobilité à une vitesse constante en un intervalle de temps nul. C'est pareil pour les électrons et totalement identique pour la "force qui les pousse", c'est à dire la tension. En réalité, quand la logique électronique passe de "0" à "1" par exemple, la tension passe de 0V à +5V en un temps très court, mais elle emprunte toutes les nuances entre ces deux valeurs. Cette variation est brutale, elle représente pour les éléments du circuit un vrai "coup de marteau électrique". La durée nécessaire pour passer de "rien" à "tout" se nomme le temps de montée. (*Pour le front montant.*) La Fig.76 est propice à mettre en évidence la différence qui sépare la théorie de la pratique. Ce petit dessin envisage le comportement de l'une des sorties binaires d'Arduino. En **bleu** est tracée l'évolution de la tension dans un **circuit hypothétiquement parfait**. Jusqu'à  $T_0$  l'état logique est "0". La broche du microcontrôleur est exactement à 0V. Puis, instantanément, la sortie passe à "1", c'est à dire à exactement +5V.



**M**alheureusement pour nous, la réalité électronique est bien plus compliquée que la logique informatique. En fonction du circuit branché sur l'ATmega328, ce dernier peut avoir du mal à "tirer" le potentiel exactement à zéro. Et l'on observe une petite tension résiduelle de +0,2V par exemple. Puis, électriquement à  $T_0$  c'est le coup de marteau. Les électrons s'affolent, et il faut pousser fort pour les bousculer. La tension peine à augmenter, et ce n'est qu'après le **temps de montée** qu'arrivée à  $T_1$  elle arrive à +5Vcc. (*Du reste le microcontrôleur n'arrivera peut être pas à maintenir exactement +5V. On peut s'attendre à un petit écart, par exemple -0.1V dans cet exemple.*) En fonction de la rapidité des circuits électroniques impliqués, ce temps de montées entre  $T_0$  et  $T_1$  peut être plus ou moins important. Il ne peut faire que quelques nano, pico, femto secondes, de toute façon la transition ne sera jamais "verticale" comme celle du tracé bleu. On ne la croit verticale sur les oscilloscopes que si la base de temps est lente au regard de ce phénomène. Mais si on la fait cavalier ventre à terre, la trace s'incline, la pente devient évidente, comme représenté par la courbe verte.

#### - STOP : On ne bouge plus !

Impossible, maintenant que les électrons ont été brutalisées, que la tension s'est mise à grimper comme une folle, il ne faut pas s'imaginer qu'arrivée exactement à +5Vcc elle va par magie stopper son évolution. Comme un caillou que l'on a lancé vers le haut, ce n'est pas parce que la main s'ouvre et ne pousse plus sur ce dernier qu'il s'arrête pile à la position du moment. Une fois lancé il continue.

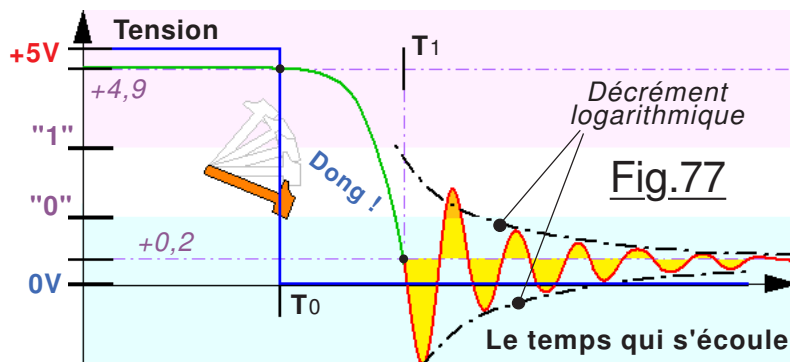
**C'**est exactement ce que traduit la partie tracée en rouge sur la Fig.76 où la tension continue de grimper et dépasse celle de l'alimentation du système. Puis, la gifle étant terminée, le circuit, à l'instar de la cloche, va se mettre à osciller à sa fréquence propre. Comme c'est très souvent le cas dans la nature, une oscillation libre prend l'allure d'une sinusoïde. C'est ce que l'on observe sur la Fig.76 pour le circuit **LC** concerné. Comme déjà précisé, l'énergie de vibration va se résorber, et l'on constate la diminution qui, concrètement, est généralement d'évolution logarithmique.

#### - Houlàla houlà, mais c'est méga la cata ça ! Plus rien ne va fonctionner alors ?

#### - Pas de panique Totoche, on a prévu les pansements pour les bobos électroniques.

Pour parer ces difficultés que l'on rencontrera forcément dans la pratique du binaire électrique, le premier remède va consister à diminuer autant que faire se peut le coefficient de surtension du circuit, le but étant de rendre l'amplitude de l'oscillation amortie la plus faible possible. (*Ce ne sera pas forcément facile ou possible.*) La deuxième parade consiste à fixer des seuils de "décision" les plus larges possibles pour "masquer" ces phénomènes. Sur la Fig.76 toute la zone bleue des tensions inférieures au seuil "0" seront considérées comme un état "RIEN".

Toute les tensions de la zone coloriée en rose au dessus du seuil "1" seront considérées comme un état logique "TOUT". Si les alternances négatives de l'oscillation restent contenues dans la zone rose, alors elles n'auront aucun effet logique. On retrouvera une approche équivalente pour les fronts descendants qui eux aussi vont auto-osciller. On peut observer sur la Fig.77 l'oscillation amortie qui se produit au moment de la transition négative quand la sortie de l'ATmega328 passe de l'état logique "1" à l'état binaire "0". Les alternances positives de ces vibrations devront rester dans la zone coloriée en bleu clair pour "ne pas exister". Il ne semble pas très malin d'avoir placé le seuil "0" de la zone bleue pastel éloigné de la valeur qui caractérise celui du "1" pour la région rose clair. Dans l'exemple de la Fig.77 on constate un léger dépassement de la première alternance positive coloriée en orange. C'est que cet écart entre "0" et "1" tel qu'il est défini dans le standard TTL est imposé par les contraintes de fonctionnement des circuits intégrés logiques. On n'est plus dans la mathématique, il faut "industriellement" concilier de nombreux compromis.



**- Et alors, que se passe-t-il dans la zone interdite blanche ?**

- Ben ... comme dans toute zone exposée au feu de l'ennemi ... il ne faut pas s'y attarder.

Autrement dit, ne pas imposer à un circuit électronique logique une tension stable comprise entre les deux seuils de la norme. Si on persiste dans cette expérience, l'opérateur logique va se comporter comme un amplificateur à grand gain et à tous les coups il risque d'exploser en une virulente auto-oscillation.

EN LOGIQUE ÉLECTRONIQUE ON TRANSITE LE PLUS RAPIDEMENT POSSIBLE ENTRE LES ÉTATS "0" et "1".

Ainsi les circuits n'ont que le temps de "monter" ou de "descendre", pas d'osciller.

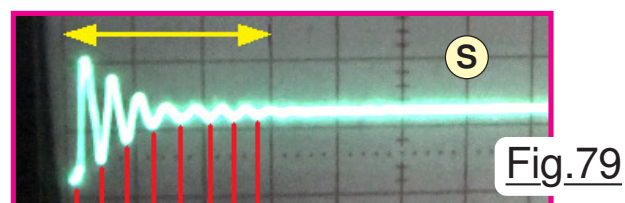
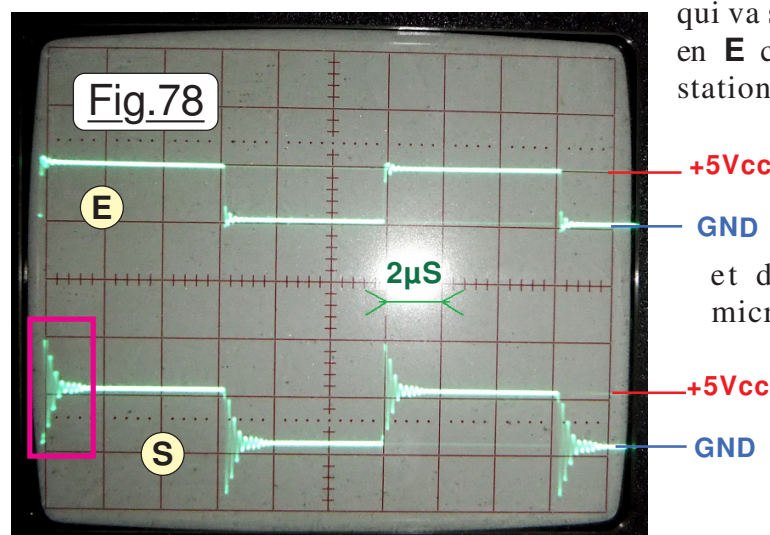
***De la théorie à la pratique.***

Désolé les amis d'avoir été aussi verbeux. Trois chapitres consacrés à de la théorie c'est un peu lourd, carrément indigeste. Mais si l'on veut comprendre un minimum ce que l'on va observer sur notre **longue ligne de dialogue**, autant se perdre dans la théorie du signal serait hors propos, autant faire moins me semble irréaliste. C'est tout bon, on peut passer à la pratique.

*- Et bé, il était temps, je me suis complètement endormi mézigue !*

Pour les lecteurs qui disposent d'un oscilloscope dans leur laboratoire d'électronique de loisir, le croquis [P05\\_test\\_de\\_la\\_longue\\_ligne\\_I2C.ino](#) est téléversé sans modification. Pour les autres, il vous faudra vous contenter des photographies issues de mes manipulations. C'est le test n°1 qui va s'exécuter, fournissant sur la sortie Alarme la cadence la plus effrénée possible. Voyons voir ce

qui va se passer. La photographie de la Fig.78 montre en **E** ce que l'on observe en Entrée de ligne coté station météo, et en **S** ce que devient le signal carré en Sortie sur les broches de l'interface. Avec une base de temps de  $2\mu\text{S}$  par graduation, les transitoires sont visuellement verticaux. On peut affirmer que les temps de montée et de descente sont de loin inférieurs à la microseconde. Les tensions qui déterminent les





états les états "0" et "1" sont pratiquement celles de **GND** et de **+5Vcc**. Ces références sont reproduites sur la Fig.78 en observant que la trace pour **S** n'a pas été tout à fait cadrée sur la ligne du bas. On constate sur la Fig.78 que les oscillations amorties provoquent un dépassement en sortie d'environ 4V, la tension crête arrive à environ +9V. La première alternance négative fait descendre l'état "1" à environ +1V, nous commençons à singulièrement friser la correctionnelle.

**P**as de panique, comme à l'extrémité **S** d'exploitation on trouve un transistor qui fonctionne en Saturé/Bloqué, si la commande met une microseconde à se stabiliser pour le pilotage d'une résistance chauffante, ça ne va pas changer grand chose à la température du local. Quand aux 9V appliqués à la base de **T**, si durant ce court laps de temps on passe à 400µA, le composant s'en fiche royalement. L'observation de la transition négative permet d'affirmer que **S** descend jusqu'à pratiquement -5V. On remarque qu'avec une alimentation de 5Vcc, le circuit en étude reçoit entre -5V et +9V ... vous comprenez pourquoi l'électronique doit intégrer des protections et tolérer des plages fonctionnelles qui dépassent largement les valeurs nominales ?

**A** analysons plus en détail les signaux visualisés sur l'écran gradué et tout particulièrement ceux de la Fig.79 qui constituent un agrandissement qui sur la Fig.78 se trouve dans l'encadré rose. L'amplification est obtenue en accélérant la base de temps. Pour ce cliché, elle est ajustée à 0,5µS par graduation ce qui va nous permettre d'évaluer la fréquence de l'oscillation amortie. Sur la Fig.79 trois graduations correspondant à 1,5µS sont mises en évidence par la flèche jaune. C'est le temps qu'il faut pour que l'amplitude de la vibration devienne négligeable. Durant ce laps de temps les traits rouges mettent huit alternances en évidence. On en déduit que la période de l'onde d'oscillation en résonance est de  $1,5 / 8 = 0,188\mu\text{S}$ . On en déduit facilement l'ordre de grandeur de la fréquence propre de la ligne qui avoisine  $1 / 0,00000188$  soit 5.33MHz.

**E**xaminons maintenant le programme qui permet de générer le signal carré servant à "secouer" la ligne de transmission. Dans le programme **P05\_test\_de\_la\_longue\_ligne\_I2C.ino** on a placé en remarque les trois autres tests pour ne générer qu'un seul signal à la fois. Ainsi on va générer la fréquence la plus rapide possible incluant la présence d'instructions de délais. Si dans la boucle de base on enlève tous les commentaires, elle se résume alors à :

```
void loop() {  
  ① digitalWrite(Sentinelle, HIGH); delayMicroseconds(1);  
  ② digitalWrite(Sentinelle, LOW); delayMicroseconds(1); }  
}
```

Cette boucle sans fin est élémentaire. En ① on fait passer la sortie nommée **Sentinelle** à "1". Puis on utilise l'instruction **delayMicroseconds** pour générer une temporisation d'1µS. En ② la sortie **Sentinelle** repasse à "0" et toujours avec **delayMicroseconds** on y reste durant 1µS. Le délimiteur **"}**" ordonne de reprendre en ① et le cycle recommence indéfiniment. En théorie, la période du signal carré devrait être de 2µS si nous étions dans un monde parfait. Force est de constater en **E** de la Fig.78 qu'une période complète s'étale sur six graduations soit 12µS.

**L'**explication à cette anomalie apparente n'a rien de bien compliqué. Chaque instruction sur l'ATmega328 exige quelques "cycles machine". Il faut donc un peu de temps pour faire passer **Sentinelle** à "0" ou à "1". Il faut aussi quelques "pouillèmes" pour effectuer le branchement de retour **"}**". Enfin, et surtout, **delayMicroseconds** pour se réaliser consomme un minimum de cycles d'horloge. C'est la raison pour laquelle il est bien spécifié dans la documentation du langage C de l'**IDE** qu'en dessous de 3µS l'instruction **delayMicroseconds** n'est plus fiable.

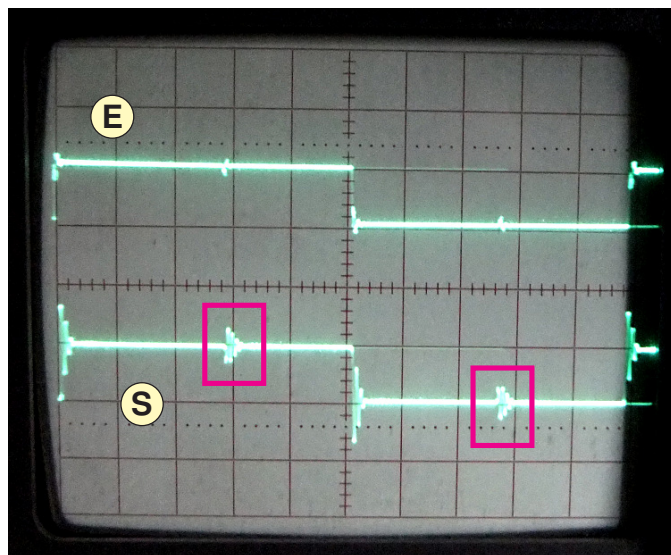
**R**emarquez que pour notre campagne d'essais cette fréquence de l'ordre de 83kHz est largement suffisante pour remuer énergiquement la longue ligne de dialogue. C'est pour aborder avec vous ces petites remarques que les instructions **delayMicroseconds** ont été introduites dans la boucle de base. Si vous désirez vraiment générer la fréquence la plus rapide possible avec ce microcontrôleur cadencé à une fréquence d'horloge nominale de 16MHz il suffit d'enlever les deux temporisations qui ralentissent la boucle de base.

**- Ouaisss, encore un long chapitre sans dessin, et paffff une salamandre de plus !**

### Comportement des lignes pour les capteurs.

Conduire les trois autres tests va nous obliger à adapter à la demande le programme de validation **P05\_test\_de\_la\_longue\_ligne\_I2C.ino** et à le téléverser pour procéder à la génération des signaux binaires. La technique consiste à passer en remarque les deux lignes ① et ② du test précédent, et à valider celles relative au test à animer. Par exemple pour procéder au test suivant dans l'ordre d'écriture du croquis, la boucle de base devient :

```
void loop() { //----Test n°2 de la ligne SCL du capteur de pression BMP085 ----
  digitalWrite(SCL, HIGH); delayMicroseconds(1);
  digitalWrite(SCL, LOW); delayMicroseconds(1); }
```



**Fig.80**

— +5Vcc

— GND

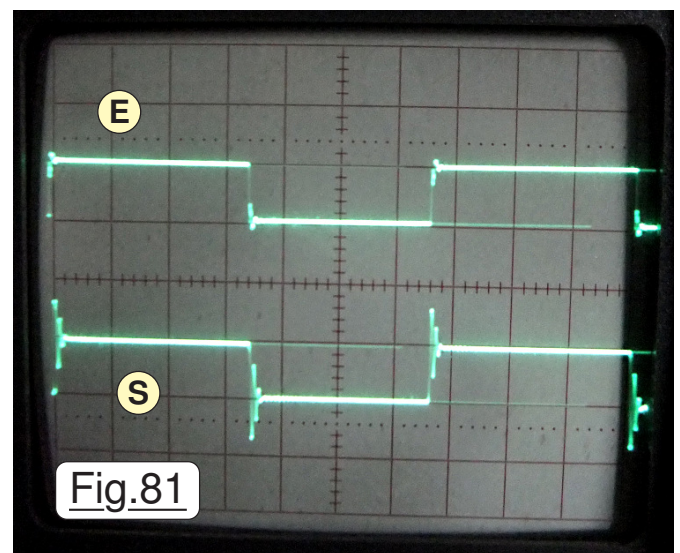
— +5Vcc

— GND

des deux phénomènes. On constate que vers  $2,8\mu\text{S}$  du front, mis en évidence dans l'encadré rose, la présence d'une oscillation amortie. Elle est de plus faible amplitude que celle engendrée sur le front

montant ou descendant, mais bien présente. La nature de cette petite perturbation n'a pas été déterminée par une analyse plus fouillée. (*Onde stationnaire en retour sur la ligne, effet de miroir dû à la présence des capteurs ...*) Une expérience strictement analogue conduite sur la ligne **SDA** révèle un comportement absolument identique, rien ne permet de distinguer de différence dans les oscillogrammes obtenus qui ne sont pas reproduits. Enfin la Fig.81 montre ce que l'on obtient sur la ligne filaire unique qui dialogue avec le capteur d'humidité DHT11. Il n'y a plus présence de la petite perturbation en phase stabilisée de la ligne, mais on retrouve l'oscillation amortie engendrée par les facteurs capacitifs et inductifs des longs conducteurs. Il est normal de constater que la fréquence de résonance soit du même ordre de grandeur. Nous pouvons maintenant proposer un bilan vraisemblable.

**CONCLUSION :** Que ce soit le DHT11 ou le BMP085, la durée de passage d'un état à son complémentaire sera largement supérieure à la durée des oscillations amorties. Ces perturbation inhérentes aux caractéristiques de notre longue ligne de pratiquement 9m passeront inaperçues, tout au moins en théorie. Par ailleurs, si une tentative de dialogue est perturbée, l'écouteur ignorera ce qui se passe en ligne, car il considèrera que la consigne actuellement sur l'I2C ne lui est pas adressée. On peut raisonnablement considérer cette ligne comme fonctionnelle avec une faible probabilité de mesures incorrectes ou perdues. Une mise en fonction sévère durant plusieurs jours semblent confirmer ces estimations. (*La ligne a été testée en plusieurs configurations.*)





**P** ar plusieurs configuration il faut entendre : Fonctionnement avec uniquement la première moitié de 4m, enroulée, déroulée, étalée en boucle fermée, en ligne. Un test a consisté à en faire une sorte de grosse bobine placée devant la porte d'un four à micro-ondes. Rien à signaler.

- Et dans le four à micro-ondes, t'as tenté le coup Totoche ?

- Vi, c'est rigolo, on dirait un orage avec des éclairs, mais NANO ne l'a pas prévu !

NON, on ne fait pas ce genre de plaisanterie, le four serait irrémédiablement détruit vous l'avez bien compris. Toutes ces tentatives avaient pour but de chercher à s'assurer qu'il serait possible et fiable de placer les capteurs loin de la station. Ils ont surtout montré un aspect particulièrement indigeste de cette belle ligne de télémesures :

**F** inalement, la caractéristique la plus critique d'une limande de ce type n'est pas sa capacité répartie, l'induction parasite, la diaphonie ... c'est sont aptitude magique à faire inexorablement des nœuds tortillonnesques chaque fois que l'on désire la dérouler. **AGASSIF !**



Fig.82

*Si la station météo est statique et la liaison passée une fois pour toute "dans les murs", pas de quoi se rouler dans le gazon. Par contre, si la petite merveille de la Fig.59 est anoblie en matériel expérimental manipulé souvent, un "H" en matériau quelconque tel que celui de la Fig.82 deviendra INDISPENSABLE. C'est simple, facile à faire et vous évitera plusieurs visites chez votre médecin de famille ...*

**T** erminer la description matérielle va enfin nous permettre d'aborder l'aspect logiciel. La Fig.83 présente la limande rallonge. Si vous acceptez un conseil, réalisez les deux "H" d'enroulement de la bretelle de raccordement AVANT de réaliser et de souder les fiches. J'ai compris la facilité qu'il y avait à le faire après avoir galéré substantiellement lors de la fabrication du premier prototype. Comme c'était le cas pour les petits laboratoires, une petite étiquette comme celle montrée sur la Fig.84 sera la bienvenue avant de procéder à la finalisation logicielle. De toute façon elle sera indispensable, alors tant qu'à la

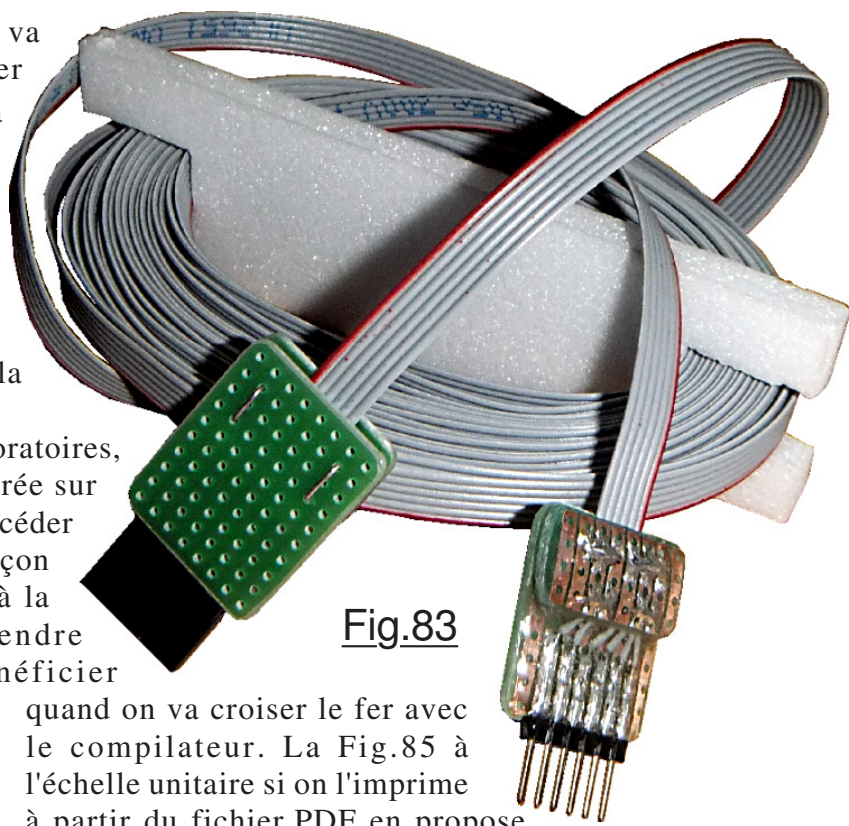
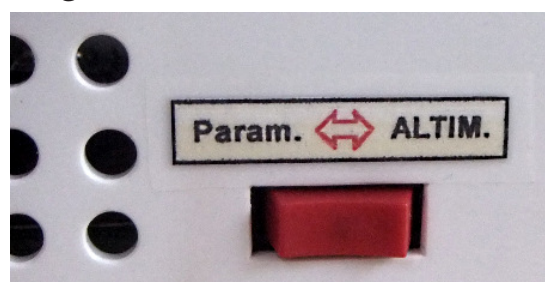


Fig.83

Fig.84

réaliser, autant s'y prendre rapidement pour en bénéficier



quand on va croiser le fer avec le compilateur. La Fig.85 à l'échelle unitaire si on l'imprime à partir du fichier.PDF en propose l'exemplaire adopté sur ma NANOMÉTÉO.

Fig.85



### Le programme qui anime notre petite réalisation.

**S**eulement deux motivations vont certainement vous pousser à faire l'effort d'aller décortiquer les nombreuses lignes abscondes du croquis qui anime NANOMÉTÉO. (*Soit 950 lignes au moment de la rédaction de ce didacticiel sans compter les petits programmes démonstrateurs.*) Le premier cas est relatif à celle ou celui d'entre vous qui s'est promené avec enthousiasme dans le gros didacticiel donné sur :

<http://www.robot-maker.com/ouvrages/apprendre-a-programmer-arduino-en-samusant/>

Dans ce cas, c'est l'envie d'apprendre à programmer Arduino qui vous pousse à papillonner dans ce dernier volet de la saga. Autre cas plausible, vous avez pour le plaisir de la posséder réalisé avec succès ce petit appareil, mais désirez "du personnalisé", avec un comportement correspondant plus à vos aspirations. Devant modifier le programme, il faut en cerner un minimum les subtilités.

**L'**objectif de ce tutoriel ne réside plus dans l'apprentissage poussé du langage C++ de l'**IDE**, mais dans la mise en œuvre "ludique" d'une carte NANO. Aussi, je vais me contenter de détailler uniquement quelques procédures spécifiques, car globalement ce programme s'inspire largement de celui du l'ENREGISTREUR À DEUX VOIES du générateur de signaux. Le croquis du programme complet **P10\_NANO\_METEO.ino** est largement pourvu de commentaires, et certaines séquences ont été détaillées dans le tutoriel sur le générateur de signaux. Inutile d'y revenir. Il va de soi que ceux qui sont émoustillés par la possession d'une telle petite station météo tout en n'ayant aucune envie de comprendre le déroulement du programme qui l'anime sont parfaitement libres d'ignorer les chapitres "informatiques". *Contentez-vous de logger le croquis* et vogue la galère ...

#### Première mise en service du programme.

**E**ncore une "entourloupe", une escroquerie qui fait croire que l'on téléverse le programme complet **P10\_NANO\_METEO.ino** et que notre station météo sera parée pour que l'on puisse envoyer à **Météo France** nos prévisions 100% fiables avec un mois d'avance. En procédant de la sorte, ce premier téléchargement du programme ne servira à rien, et vous n'obtiendrez qu'un écran incohérent ou tout noir. Ces deux comportements seront directement fonction d'un booléen logé en EEPROM et informant le logiciel sur ce qu'il doit faire sur un RESET. Les explications à ce sujet seront développées plus avant. La première mise en service et paramétrage de NANOMÉTÉO suppose que cette dernière soit entièrement réalisée, que tous les périphériques soient branchés et en particulier le petit module des capteurs. (*Au début ne pas intercaler une éventuelle ligne de télémesures qui ne peut que compliquer les premiers essais.*) Si vous ne voulez absolument pas chercher à comprendre les subtilités du programme, voici la procédure commentée pour conditionner une carte NANO qu'elle soit vierge de tout octet ou qu'elle soit habitée par un quelconque croquis y compris avec des données diverses plus ou moins entassées dans la mémoire EEPROM :

- 1) Téléverser **P01\_Ecriture\_LOGO\_en\_EEPROM.ino** qui place en début d'EEPROM le LOGO personnel qui est une petite salamandre. Si le circuit imprimé est globalement correct, l'afficheur OLED rend compte avec un petit texte présenté en zone bleue.
- 2) Téléverser **P02\_Afficher\_LOGO.ino** n'est absolument pas indispensable pour mettre en service la petite station météo. Il ne fait que montrer le petit dessin. La LED verte doit clignoter à environ cinq hertz prouvant qu'elle est correctement câblée.
- 3) Téléverser **P03\_Ecrire\_les\_textes\_en\_EEPROM** va logger dans l'ATmega328 tous les textes affichés par NANOMÉTÉO. Ces données ne sont pas incluses de façon banale dans le programme principal car nous sommes dans l'obligation de libérer de la mémoire dynamique. À la limite, peu importe la raison, ne pas utiliser ce programme va engendrer des textes étranges donnant à penser que le logiciel diverge, ce qui n'est pas forcément le cas. Quand ce croquis a été téléversé, utiliser le Moniteur série pour vérifier le résultat. (**ATTENTION : Vitesse 115200 Bauds sur l'IDE.**)
- 4) Téléverser **P10\_NANO\_METEO** qui au point de vue du chargement de code dans le microcontrôleur constitue la dernière étape. Tout ce qui concerne le logiciel est en place. Toutefois, comme c'est le cas pour tout système informatique, il faut maintenant aller dans le SET UP :



**Conditionnement initial et consignes "au sommet".**

**P**réésentant deux modes de redémarrage sur un RESET et une fonction de Sentinelle surveillant la température, notre petite station météo va à ce stade de sa première mise en service pouvoir potentiellement présenter plusieurs comportements différents que l'on va passer en revue. On suppose dans ce qui suit que le mini clavier et que le capteur rotatif sont correctement branchés et opérationnels. Donc ... avec un multimètre vous avez vérifié qu'électriquement le fonctionnement correspond à celui attendu.

- **Premier cas** : L'écran est tout noir, affiche **Capteurs NON OK** et l'alarme couine en boucle. C'est le matériel qui est en cause. Tant QUE LES DEUX CAPTEURS ne dialogueront pas correctement avec NANOMÉTÉO on en prendra plein les oreilles. Il faut absolument corriger le problème.

- **Deuxième cas** : L'écran affiche un court instant **Capteurs OK** puis devient tout noir. La LED jaune clignote environ toutes les trois secondes, le cœur de notre bébé bat régulièrement :

- *Youpyyyyy, ça marche !*

Cet écran tristounet n'a rien d'anormal, c'est du reste ce qui va probablement se passer si la carte NANO est vierge. Son EEPROM ne comporte que des octets à \$FF, (*Mis à part les textes et le LOGO.*) et la variable "SÉCURITÉ MÉTÉO" est alors validée.

Il est également fort probable que la LED rouge de l'alarme va s'allumer. Rien d'anormal non plus, surtout si la carte NANO est neuve et que son EEPROM n'a jamais été programmée.

- *Lé pas beau cet écran, on peut voir la grenouille ?*

- *Non, désormais à Météo Perso c'est une Salamandre qui prédit le temps.*

- *Comment on fait pour la trouver ?*

- *Fastoche :*

- **Deuxième cas** : Pour conditionner un comportement coutumier avec affichage d'un écran d'accueil sur RESET il faut, **quand l'écran est tout noir**, c'est à dire que l'appareil est en mode VEILLE, cliquer à gauche sur l'inverseur à bascule. L'écran affiche **SECURITE METEO NON** et la LED jaune clignote rapidement invitant à cliquer sur **FC+** par exemple.

- *C'est pas bon, c'est encore tout noir !*

- *Normal Dudule, t'as changé l'option sur RESET, mais la station continue à stationner !*

- *Qué que je dois faire alors Totoche ?*

- *Ben un RESET gros malin !*

- **Troisième cas** : L'écran affiche la version ainsi que le LOGO et la LED jaune clignote rapidement invitant à cliquer sur **FC+**. (*Ou FC-* à ce stade c'est équivalent.) Ce comportement est également celui que l'on obtient sur RESET quand on a commencé avec le deuxième cas et que l'on a utilisé l'inverseur à bascule à gauche. Nous analyserons dans le prochain chapitre toutes les possibilités.

- *C'est beau, mais je déteste le feu rouge !*

- *Ben il te suffit de cliquer sur FC- long et de consigner une température.*

**C**omme pour l'activation de chaque option possible, le programme demande confirmation. **FC- court** annule la requête, **FC+ court** valide la saisie. Cliquez sur **FC+ long**, le programme présente alors le texte **°C Alarme = 0**. Appuyez enfin sur **FC+ court**, le programme revient en EXPLOITATION et au bout de trois secondes au maximum la LED rouge s'éteint. C'est terminé, globalement notre appareil fonctionne correctement. Seule une exploration de toutes les fonctions confirmera dans le détail un comportement 100% conforme à ce que l'on attend, cependant si les premiers essais qui précèdent ont conduit à des manipulations correctes, raisonnablement on peut replacer définitivement le couvercle. Avant de "refermer la porte à clef" vous avez opté pour une utilisation "portative" ou "sédentaire", car le protecteur des capteurs en utilisation mobile doit être mis en place avant de refermer définitivement le coffret.

**Note** : Le programme **P04\_Corriger\_la\_version\_en\_EEPROM.ino** ne sert strictement à rien si vous n'avez pas l'intention de modifier le logiciel à votre guise.

## Les deux fonctions de base.

Considérons que la fonction ALTIMÈTRE de NANOMÉTÉO n'est qu'un petit plus. Avouons qu'avec les minuscules modules du commerce qui sont spécialement adaptés aux randonneurs, notre réalisation fait pâle figure et ne peut concurrencer ces merveilles de technologie dédiées aux "sportifs" et promeneurs expérimentés. Cette possibilité incite toutefois à concevoir vraiment petit, ce qui justifie l'usage d'une carte NANO. Étant donné que l'Altimètre reste rudimentaire, on va en décrire en premier les options et les informations qu'il délivre. Débarrassé de ce qui n'est qu'un accessoire très secondaire, on pourra alors aborder l'aspect météorologique aux très nombreuses options et données présentées.

### **La fonction altimètre pour promeneur.**

Basculer l'inverseur à droite est prévu pour invoquer cette fonction élémentaire. Quel que soit le mode de fonctionnement en cours, comme on peut fort bien provoquer cette séquence par erreur de manipulation, le programme demande confirmation avec le message **Mode ALTIMÈTRE ?**. La touche **FC- court** ou **long** provoque la fuite et le retour en fonction météorologie dans le mode en cours. Si vous validez par erreur, (*Faites un petit effort, car ça fait deux erreurs de suite tout de même !*) le système "pardonne". Il y a retour à la station météo, mais en mode EXPLOITATION ce qui n'est finalement pas tragique. La sortie de la fonction ALTIMÈTRE s'obtient aussi bien avec **FC+ long** que la touche **FC- long** si confirmé.

### **Mode d'affichage Normal.**

C'est celui où les données utiles sont visualisées sur l'écran. La Fig.86 le présente et va nous permettre d'en appréhender les diverses informations. En supposant que notre altimètre soit correctement calibré, c'est en **2** que sera indiquée l'altitude, c'est à dire la hauteur mesurée par rapport au niveau de la mer Méditerranée. (*Cette notion a été légèrement introduite en début de ce didacticiel ...*) Pour ceux qui préfèrent les "unités aviation", en **3** la valeur est convertie en pieds. En revanche, la valeur **4** n'a rien à voir avec une hauteur, c'est à dire la distance verticale qui nous sépare du sol. C'est pour notre appareil le dénivelé que nous aurons franchi en nous déplaçant. Ce dernier est chiffré en mètres sans décimales. Si on se trouve plus haut que la **Référence zero** il n'y aura que la valeur sans le signe. Si au moment du rafraîchissement de l'écran on est plus bas, la valeur sera complétée par le signe moins. Sur la Fig.86 nous sommes légèrement plus bas, mais comme les décimales ne sont pas indiquées, on obtient un zéro avec le signe moins ce qui ne va pas plaire du tout aux mathématiciens. (*Car en principe le zéro n'a pas de signe puisqu'il désigne le néant.*)

**- N'ont qu'à descendre de plus d'un mètre, c'est pas sorcier !**

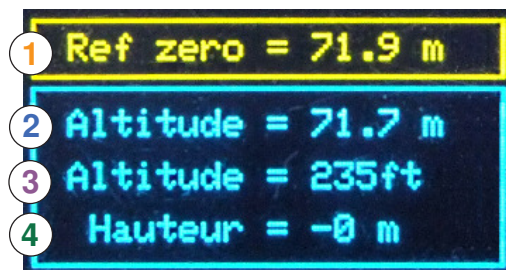
Figurez-vous que cette photographie a été prise dans le laboratoire qui se trouve à une altitude réelle de 161m environ. Pourtant ce n'est pas ce qu'affiche l'altimètre, loin s'en faut. Pour un circuit intégré complexe avec logique et informatique poussée réputé travailler avec une résolution altimétrique de 0,1m le moins que l'on puisse dire c'est que le résultat est médiocre. Et bien ce pauvre capteur de pression atmosphérique n'y est pour rien. Nous avons vu qu'utiliser cette pression impose d'apporter une correction pour tenir compte du fait que l'atmosphère du jour n'est pas "standard". C'est la raison pour laquelle tout altimètre basé sur la pression atmosphérique doit être doté d'un moyen de calibrage. Donc, avant de partir de chez vous il faudra l'ajuster, ou emporter avec vous une carte qui permet d'effectuer cette initialisation avant de vous promener.

- Hé bé Totoche, t'as mesuré la hauteur du labo jusqu'à la mer Méditerranée depuis Cahors ?
- Mais non gros malin, c'est par satellite que l'on procède.
- T'as un satellite perso qui tournicote autour de la Terre ? ? ?
- Mais non béta Dudulesque, c'est sur Internet que tu trouves ça très facilement.

Personnellement pour avoir l'altitude du laboratoire j'ai utilisé la fabuleux "lien" :

[http://www.toutimages.com/google\\_gps/maps\\_altitude.htm](http://www.toutimages.com/google_gps/maps_altitude.htm)

Fig.86





Première étape quand vous avez proposé le lien, vous centrez grossièrement la carte de France sur une grande ville proche de votre habitation. Pour cela il suffit de maintenir activé le **BGS** (*Bouton Gauche de la Souris.*) et centrer le lieu souhaité dans le rectangle réservé à la carte. Avec la molette de la souris on peut changer à convenance le facteur de ZOOM. (*Touchez plus ou moins du pavé numérique si votre souris n'a pas de molette.*) En combinant agrandissement et déplacement vous arriverez à cerner votre lieu d'habitation. Passez ensuite en visualisation **Satellite**. La Fig.87 présente ma maison au plus grand facteur d'agrandissement. Un petit clic avec le **BGS** quand le curseur est exactement à l'endroit désiré et la fenêtre contextuelle **F** donne l'information. ATTENTION, inutile de cliquer sur la toiture de la maison pour en déduire sa hauteur. L'application indique globalement l'altitude du sol. Étant situé à l'étage par rapport à la cour pointée comme référence, en ajoutant 3m j'obtiens les 161m annoncé dans le chapitre précédent. À partir de cette information, il devient possible de calibrer l'appareil de mesure.



Fig.87

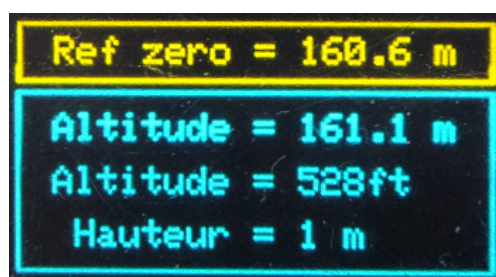
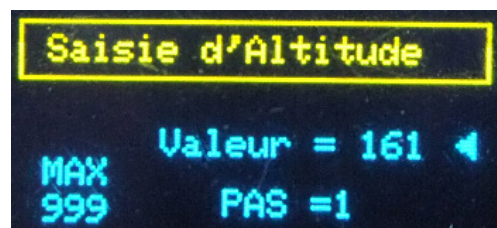
### Mode VEILLE.

Maniaque invétéré des économies d'énergie et de la moindre usure inutile des matériels, le mode veille consiste à éteindre l'écran. Je n'ai pas mesuré l'économie de courant qui en résulte, mais de toute façon étant en usage "autonome" il faut ménager au maximum la pile qui alimente notre joujou portatif. Comme il ne nous en coûte que quelques octets dans la mémoire de programme, ce serait idiot de ne pas le prévoir. Par ailleurs, cette option peut vous sauver la vie. Obnubilé par la hauteur que vous grappillez avec peine sous un soleil de plomb, votre nez reste collé au petit écran magique. Et Paffffffff, ouille, ouille ... tombé dans le profond ravin ! Dans ce mode, l'appareil conserve ses réglages et la LED jaune clignote un court instant une fois toutes les trois secondes. On alterne entre les deux modes d'affichage en cliquant sur le bouton central du capteur rotatif. Ce dernier n'est consulté qu'à la volée dans la boucle de programme de l'altimètre. Un clic court va probablement ne pas être détecté. **L'utilisation rationnelle consiste à enfoncer le bouton poussoir du codeur rotatif et attendre que la LED verte s'illumine** attestant de sa prise en compte.

### Correction barométrique.

Fig.88

N'importe quel altimètre basé sur la mesure de la pression atmosphérique locale, quelle que soit sa technologie, doit être corrigé en fonction des caractéristiques du moment de l'air ambiant dans lequel il se trouve. On peut affirmer que la correction barométrique est en réalité une adaptation météorologique. Cette initialisation peut être calculée en tenant compte des différences de caractéristiques entre l'atmosphère environnante et celle de l'**atmosphère standard**. Les calculs sont assez indigestes, aussi, sur le plan pratique, on se contente de conditionner l'altimètre en l'obligeant à indiquer une valeur correcte au lieu où l'on se trouve. Que ce soit avec Internet, par consultation d'une carte précise avec courbes de niveaux, peu importe la source. Le laboratoire est à 161m, il suffit comme sur la Fig.88 d'imposer cette valeur en correction altimétrique. Une fois avoir effectué cette saisie, l'appareil va fonctionner avec précision. Ce n'est vrai que sur une période



relativement courte. Quelques heures, car le temps change, et par contre coup la pression atmosphérique. Ce n'est également fiable que si l'on ne franchit pas une distance trop importante durant ces quelques heures. Il est évident que la météo à Paris n'est pas identique à celle de Rome, ce serait trop simple. Comme on peut le constater sur la Fig.89 la valeur attendue est maintenant crédible. Voyons maintenant la procédure à suivre pour cette manipulation simple.

Fig.89

Sollicitons la touche **FC+ court** qui ouvre la fenêtre de la Fig.88 laquelle annonce que l'on ne peut proposer que des altitudes inférieures à 1000m. Cette limite n'a rien à voir avec une quelconque logique d'utilisation d'un altimètre, elle résulte d'une contrainte informatique. En effet, de nombreux paramètres de notre appareil peuvent être modifiés à convenance. Les procédures de saisie doivent se contenter des trois boutons poussoir et du capteur rotatif pour présenter des protocoles simples. Associés à ces éléments il faut présenter des fenêtres contextuelles. On se doute que les procédures logicielles complexes sont communes à toutes les sélections d'options. En mode station météo les valeurs à imposer restent faibles : Des durées en heures, minutes et secondes, une température qui restera inférieure à 45°C. Pour des raisons d'optimisation du logiciel et de l'exploitation du petit écran OLED, la procédure **MAJ\_VALEUR\_avec\_limites()** restreint toutes les saisies à maximum trois chiffre soit 999. Comme la fonction Altimètre est estimée marginale, si votre promenade débute à 2834m vous consignez 834. Il suffit à la consultation d'ajouter mentalement 2000m et l'appareil reste fonctionnel. N'oublions pas que le plus utile pour cette application n'est pas la valeur de l'altitude proprement dite, mais sa variation quand on monte ou que l'on redescend.

### Protocole d'une saisie d'altitude.

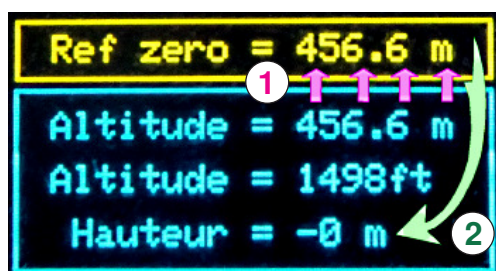
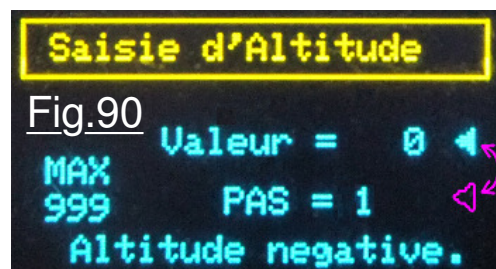
Probablement rarissime, il ne vous arrivera peut être jamais de commencer votre promenade à une altitude négative par rapport à la mer la plus proche. On ne peut toutefois exclure complètement cette hypothèse. Il faut donc avoir la possibilité d'imposer un signe négatif. Par ailleurs, les protocoles de saisie doivent répondre à un standard de comportement quel que soit le paramètre en cours de définition. Globalement une saisie va répondre aux invariants suivants :

- Toute page de saisie résume la nature de la donnée consignée dans le rectangle jaune.
- La touche **FC+ long** ou **FC- long** valide la saisie et fait sortir de la fenêtre contextuelle.
- Généralement à gauche dans le cadre bleu est précisée la valeur maximale autorisée.
- Durant la saisie toute valeur exagérée sera limitée à 999 avec un avertissement sonore.
- En validation toute valeur consignée sera "rabotée" à celle maximale autorisée.
- Le bouton central du codeur rotatif alterne entre une saisie de la **Valeur** ou du **PAS**.
- Le codeur rotatif incrémente ou décrémente la **Valeur** ou le **PAS**. (Indiqué par ◀.)
- La touche **FC- court** force le **PAS** à 1 et **Valeur** à zéro.
- **FC+ court** multiplie par dix **Valeur** avec une limitation à 999. (Pas d'effet visuel si 0.)

C'est l'inverseur ↵ basculé à gauche vers **Param.** qui pour l'altimètre fait alterner entre une valeur positive et une valeur négative. Dans le cas d'une valeur négative, comme montré sur l'écran de la Fig.90 un texte évocateur est ajouté en bas du cadre de la zone bleue de l'afficheur OLED. La **Ref zero** est réinitialisée à cette altitude en validation de la consigne.

### Initialisation de l'altitude de référence.

A tout moment durant la promenade il sera commode de réaffecter la **Référence zero**. Quand on clique sur **FC- court** il ne se passe pas grand chose. Comme montré en 1 sur la Fig.91 la valeur actuelle de l'altitude est recopiée dans la donnée de la zone jaune. Simultanément, la valeur de la **Hauteur** en 2 est remise à zéro. À partir d'ici, tout changement d'altitude sera pris en compte et la différence calculée avec **Ref zero** sera indiquée en 2. La donnée **Hauteur** indique en réalité



le dénivelé entre la position de départ initialisée avec **FC- court** et le lieu actuel. Ce n'est pas vraiment une hauteur, car par définition cette dernière se mesure à la verticale. Ceci étant précisé, aucune loi n'interdit de vous servir de cette fonction pour estimer la hauteur des différents étages de la tour Eiffel quand vous aurez le privilège d'en gravir toutes les marches lors de votre visite à notre belle capitale.

Fig.91



## La petite station météorologique.

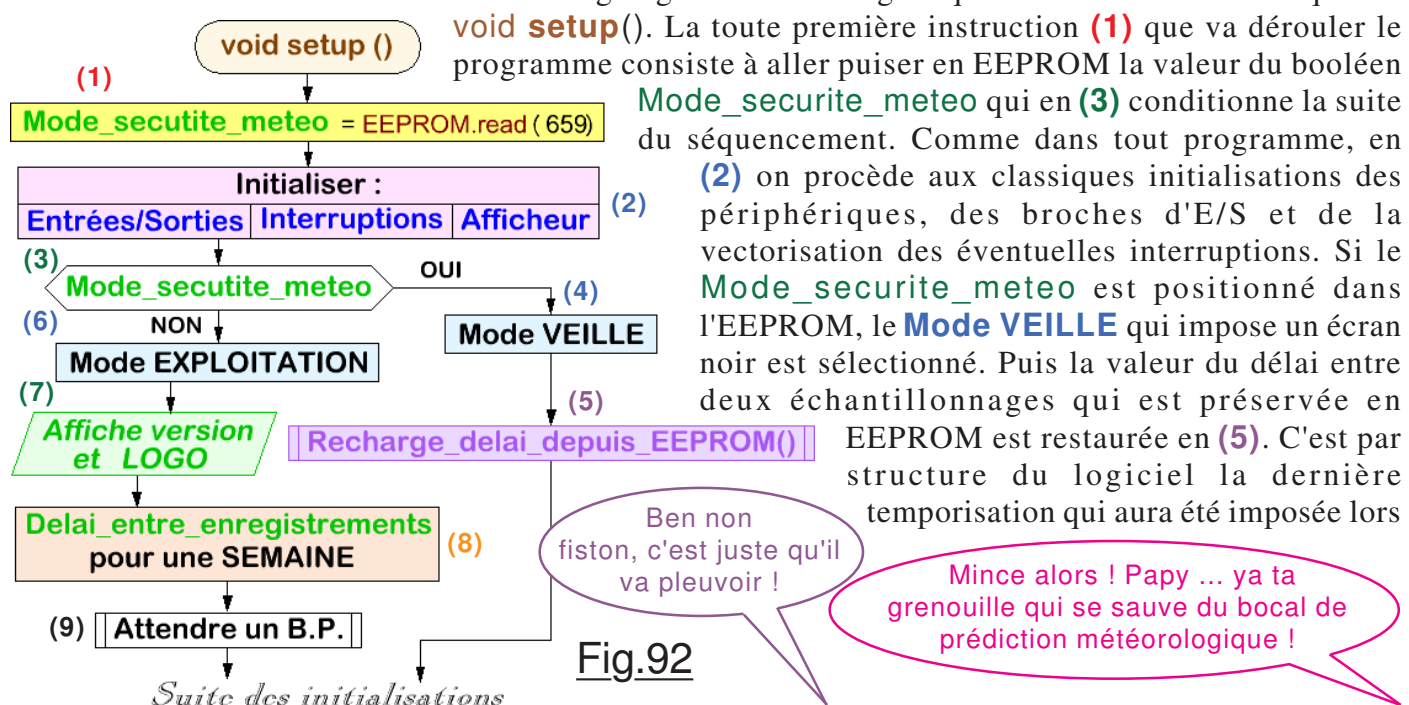
**P**robablement que ce petit instrument coulera une vie calme et sédentaire. Choyé dans le confort d'une pièce chauffée dans la maison, et que seuls les capteurs seront déportés à l'extérieur pour endurer les frimas hivernaux. C'est l'application la plus ludique. Celle qui se montre la moins exigeante en fiabilité. En tant que station météo, il faudrait ajouter un pluviomètre, un anémomètre et pourquoi pas une girouette ? Toutefois, nous ne serions plus tout à fait dans un contexte totalement ludique avec pour seul prétexte celui d'utiliser une petite carte NANO Arduino pour en explorer les possibilités. L'autre utilisation potentielle consiste à placer les capteurs dans une petite véranda qui accueille l'hiver nos belles et nombreuses plantes en les préservant du gel extérieur. Dans ce cas, les trois paramètres enregistrés sont largement suffisant puisqu'intrinsèquement seule la température serait à surveiller. On peut alors se contenter de l'alarme. Si vous estimez que l'allumage de la LED rouge n'est pas suffisant, on peut aussi opter pour l'avertissement sonore complémentaire. (*Un BIP BIP BIP agassif...*) Enfin, la sentinelle thermique peut également piloter une interface de puissance additionnelle qui sera chargée de chauffer le local surveillé dès que le seuil de l'alarme est franchi.

### Comportement de NANOMÉTÉO sur un RESET.

**C**onfier la surveillance de nos "trésors botaniques" à une sentinelle électronique impose ipso facto des critères de fiabilité incontournables, et tout particulièrement la prise en compte d'une coupure intempestive du secteur électrique qui alimente ce petit appareil. Il faut notamment rétablir le contexte lors du retour de l'énergie sur le réseau électrique, donc sur un RESET. Restaurer les paramètres de consigne consiste à retrouver impérativement :

- La température de seuil consignée pour la sentinelle thermique,
- L'écran de veille pour économiser l'écran de l'afficheur LCD,
- Le délai enregistré à respecter entre la saisie de deux échantillonnages.

Un écran noir sur RESET interdit l'affichage de notre LOGO, c'est un peu frustrant. Deux approches sont toutefois possibles. **Première possibilité, une station météo pour le plaisir.** Sur un RESET le programme présente le LOGO ainsi que la version. Dans ce cas, par défaut sur RESET la durée totale d'un écran correspondra à une semaine. (*Un jour par graduation sur "la base de temps".*) **La deuxième possibilité sera celle d'une installation de surveillance** avec reprise des dernières consignes sur un RESET et écran de veille. Dans le but de satisfaire ces deux préférences potentielles, une option de saisie permettra à tout moment d'inverser le comportement qui sera conditionné par la variable spécifique `Mode_securite_meteo`. Le mécanisme de réanimation de l'ATmega328 lorsque se déclenche un RESET est résumé sur l'organigramme de la Fig.92 qui décrit le début de la séquence



d'une manipulation par l'opérateur. Si le booléen en EEPROM(659) est à **false**, en (6) le **Mode EXPLOITATION** est alors validé. *(Ces divers modes d'utilisation de la station météo sont explicités en détail plus avant.)* En (7) est alors présenté l'écran d'accueil avec indication de la version du logiciel et tracé du LOGO. Puis, par défaut en (8) le délai d'échantillonnage est consigné pour un graphe "historique" s'étalant sur exactement une semaine. *(Les trois cas de base de temps particuliers sont naturellement décryptés dans la suite de ce tutoriel.)* Enfin en (9) le programme attend que l'on clique sur une touche du clavier pour nous laisser le temps de contempler l'écran de bienvenue. Chaque fois que le programme sera en attente, pour nous le faire savoir la LED verte de **FC+** clignotera rapidement. Autre point important qui n'est pas mentionné dans l'organigramme : La suite des initialisations procède inconditionnellement au rechargement depuis l'EEPROM de la dernière **température de seuil pour la sentinelle** thermique qui aura été consignée par l'utilisateur.

### Première mise en service du programme.

Déjà abordé en page 42, il n'est pas question d'en reprendre les explications. Maintenant que le processus de réveil du microcontrôleur est explicité sur la Fig.92 on peut préciser quelques points particuliers. Nous savons que l'EEPROM d'un ATmega328 vierge est intégralement remplie d'octets \$FF ou si vous préférez des 255 en décimal. Hors pour le comportement du langage C toute valeur différente de zéro correspond pour un booléen à **true**. Lors de la première activation du programme complet on va donc se trouver dans le **Premier cas** avec un écran est tout noir. La température de consigne sera hors normes à 255°C avec un affichage impropre sur l'écran.

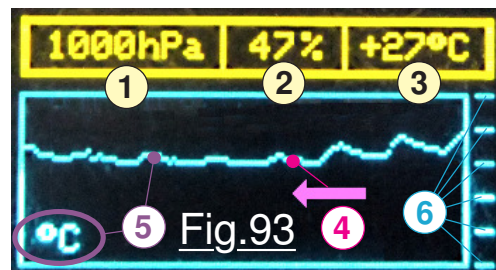
## LES TROIS MODES DE FONCTIONNEMENT DE NANOMÉTÉO.

Conditionnant directement l'agrément que nous éprouverons à utiliser notre petit appareil électronique, de longues heures de programmation ont permis de faire émerger trois "besoins spécifiques". La possession d'une petite station météorologique du commerce depuis des années a également fait naître des envies. Le programme actuel n'a donc strictement rien à voir avec de la génération spontanée. Le Mode EXPLOITATION on s'en doute sera celui à privilégier quand on voudra consulter les valeurs du moment. Ces dernières seront rafraichies toutes les quatre secondes environ. Le Mode VEILLE correspondra plus à un automatisme de confiance, exactement comme le thermostat qui gère le chauffage des pièces de votre maison. On initialise et on oublie. Enfin, le Mode HISTORIQUE pourra être invoqué à tout moment pour étudier en détail la façon dont ont évolué les trois paramètres météorologiques au cours des heures ou des jours qui précèdent.

### Le Mode EXPLOITATION.

Lorsqu'il est sélectionné, c'est "l'instant présent" qui est prioritaire. Toutes les trois à quatre secondes, la LED jaune s'illumine un court instant et les valeurs numériques des trois paramètres météorologiques sont rafraichies. C'est en quelques sortes "du temps réel". L'affichage ressemble à celui de la Fig.93 qui se présente sous sa forme la plus épurée car la base de temps est quelconque et ne donne pas lieu à des graduations pour situer les intervalles de temps. En 1 nous avons la valeur actuelle de la pression exprimée en hectopascals. *(Ou si vous préférez en millibars, unité désuète qui n'a plus cours.)* En 2 la valeur du pourcentage d'humidité dans l'air par rapport à la saturation. Enfin en 3 se trouve la valeur de la température exprimée en degrés Celsius. Ces trois valeurs sont rafraichies environ toutes les trois secondes. *(Éventuellement plus rapidement si la durée consignée pour l'intervalle de temps entre deux échantillonnages est inférieure à la seconde.)*

La zone jaune est donc affectée au "temps réel". Dans le cadre bleu est présenté le graphe de l'historique construit à partir des 120 échantillons précédents. Comme représenté par la flèche rose rajoutée en 4 l'ensemble du graphique se décale d'une position vers la gauche à chaque échantillonnage. Si la base de temps est courte, on observe ce décalage régulier. Pour la température ou pour la pression la courbe prend alors l'allure d'une droite horizontale car ces deux paramètres ne varient que sur le long terme. Pour l'hygrométrie la trace "sautille" d'environ 5 PIXELs verticalement,





car ces mesures par fonctionnement du capteur sont bien plus fluctuantes. Initialement les trois courbes pour les trois paramètres étaient visualisées simultanément. Le résultat restait inexploitable. Le remède consiste à n'en présenter qu'une seule à la fois et d'en changer automatiquement toutes les trois secondes environ. *(Ce délai peut doubler dans le cas d'un intervalle d'échantillonnage très court qui mobilise copieusement le temps processeur pour rafraîchir l'affichage des données.)*

Repéré en **5**, en bas à gauche de l'écran sont recopiées les unités de la donnée en cours de tracé graphique. On sait ainsi sans ambiguïté quelle est la variable dont on observe l'historique. En **6** le coté droit de la fenêtre divise l'amplitude verticale des variations en six niveaux. Comme on le verra plus avant ces jalons facilitent l'interprétation des graphes au point de vue des valeurs des variables. Par exemple le pourcentage d'humidité verticalement va varier entre 0 et 100%. Entre deux graduations la variation sera donc de 20%. Voyons les options possibles dans ce mode :

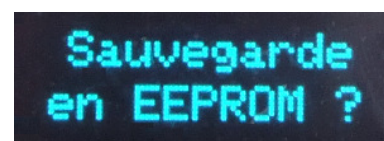
#### Option pour FC+ court.

Le message présenté au centre de l'écran comme photographié en Fig.94 est explicite. **FC- court** ou **long** correspond à la fuite, c'est à dire au retour à la normale sans rien changer au contexte. **FC+ court** ou **long** engendre la sauvegarde des enregistrements, c'est à dire trois fois 120 valeurs. Cette opération prend un court instant, puis il y a retour à l'écran graphique et à la présentation en temps réel des trois variables météorologiques.

#### Option pour FC- court.

C'est l'option réciproque à la sauvegarde. Bien que cette opération ne présente pas de grosse perte si on l'invoquait involontairement, il y a demande de confirmation. *(Il y a "écrasement" du graphe en cours pour restituer un historique ancien.)* **FC- court** ou **long** engendrent la fuite. **FC+ court** ou **long** provoque le remplacement de l'historique actuel par celui "immortalisé" dans l'EEPROM.

Fig.94



Rappel EEPROM  
Enregistrements ?

Fig.95



**RAPPEL :** Pour les trois modes écran, **FC+ long** ouvre le menu de changement de la base de temps. **FC- long** fait passer à la saisie de la température de seuil pour la sentinelle et valide ou non l'alarme sonore. Ces deux fonctions sont étudiées plus avant.

**NOTE :** Pour les trois modes de la station météo, globalement le bouton rotatif sert à obtenir des informations, alors que l'inverseur **Param.** basculé à gauche ouvre une option "marginale".

#### Option pour Param. ←.

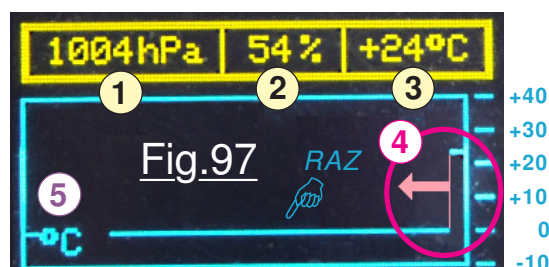
Encore un écran qui se passerait aisément de commentaire. Invoquer cette fonction sera utile

Fig.96

lorsque l'on a changé la valeur de la base de temps, dans ce cas, seules les nouvelles mesures seront crédibles au point de vue évolution temporelle. Effacer le graphe épure l'écran sur toute sa largeur, les nouvelles valeurs arriveront par la droite lors des décalages à gauche sur échantillonnage. L'effacement des échantillons n'est pas aussi trivial que l'on pourrait le croire, car remettre à zéro la valeur de la pression n'est pas réaliste. Voici la stratégie adoptée :

- Pour les valeurs de l'hygrométrie on force toutes les données à 0%. Le graphe représentatif est alors constitué d'une droite horizontale confondu avec le bas du cadre bleu.
- Pour les valeurs de la pression l'intégralité des échantillons est imposée à 940hPa qui est la plus petite valeur numérisée possible. *(Explications plus avant ...)* Le graphe représentatif est alors également une droite horizontale qui "se cache" en bas du cadre bleu.
- Pour les températures, la logique pousse à adopter 0°C pour tous les échantillons. La Fig.97 présente en bleu clair à droite les valeurs des températures correspondantes aux graduations. On voit que le 0°C n'est pas tout en bas du cadre. Sur cette image le décalage en **4** du graphe correspond à la saisie de trois échantillons à 24°C. On retrouve en **1**, **2** et **3** les valeurs "temps réel".

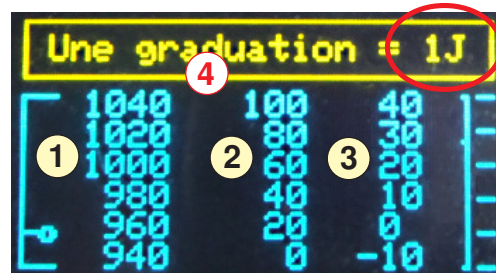
Effacer les graphes ?



### Option pour le capteur rotatif dans le sens ↺.

Comme illustré sur la Fig.98 l'écran donne un rappel des valeurs correspondant aux graduations d'amplitude. En 1 on constate que la pression sera présentée pour des valeurs allant de 940hPa à 1040hPa. Dans les annales mondiales les valeurs barométriques extrêmes enregistrées sur Terre sont respectivement de 868hPa et de 1086hPa. La

Fig.98



restriction imposée sur le graphe n'est pas pénalisante, car à nos latitudes les valeurs adoptées de 940 et de 1040 ne sont que rarement atteintes. De toute façon le graphe "talonnera" sur le cadre bleu, certes, ce qui n'empêchera strictement pas de présenter une valeur numérique correcte dans le cadre jaune. Nous le verrons dans un chapitre postérieur, mais notez dès à présent que trois bases de temps particulières sont gérées par le programme. Ces cas présentent la particularité de présenter une échelle graduée horizontale sur le bord supérieur du cadre bleu pour figurer les écarts temporels et faciliter la lecture de l'historique. Dans ce cas en 4 est précisée la durée qui s'écoule entre deux graduations horizontales. Dans notre exemple chaque intervalle correspondra à une journée soit 24 heures. Si nous avons imposé une durée entre échantillons quelconque, le graphe est épuré comme sur la Fig.93 et le texte Une graduation = est barré dans le cadre jaune.

### Option pour le capteur rotatif dans le sens ↻.

Fig.99

Toujours dans le cadre des rappels mémoriels, cette option va provoquer l'affichage de deux fenêtres d'informations. La première a pour mission d'afficher les paramètres imposés pour l'intervalle de temps entre deux échantillonnages. Par exemple en A sur la Fig.99 on a imposé des valeurs assez quelconques. À partir de ces dernières, le logiciel calcule (*Au dixième de jour près.*) et affiche en B dans le cadre jaune, la durée qui correspondra à un histogramme complet. Pour cette



temporisation la largeur du cadre graphique bleu correspondra à une durée totale de 27,0 jours. Le logiciel est en attente d'un clic sur l'un des deux boutons du clavier pour nous laisser le loisir de lire ces diverses données. Pour nous le faire savoir la LED verte clignote rapidement. Toutefois, durant cette phase la temporisation pour effectuer les échantillonnages n'est plus surveillée, il ne faut pas laisser trop longtemps la station météo dans cette fonction. Pour nous inciter à passer à la suite le

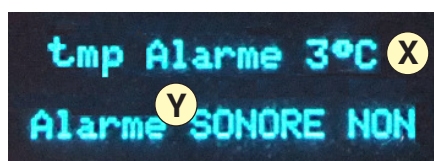
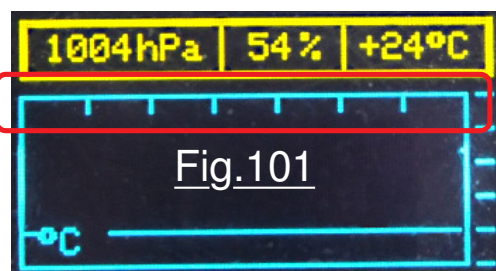


Fig.100

texte Enregistrements en C est barré. Obtempérer présente alors la fenêtre d'informations relatives à la sentinelle thermique. Cette page montrée sur la Fig.100 présente deux données. En X se trouve la recopie de la dernière valeur de consigne pour le seuil de température qui activera la sentinelle. C'est la valeur qui est mémorisée en EEPROM. En Y spécifie si l'on a choisi de compléter l'alerte de la LED rouge par une alarme sonore très intrusive. L'écran est simplifié, le message C n'est pas réitéré, il ne faudra cependant pas laisser ce mode inerte et revenir au fonctionnement normal en cliquant sur l'un des deux boutons poussoir du clavier.

### Les délais d'échantillonnage particuliers.

Anticipant les chapitres qui vont suivre, on va ouvrir une parenthèse. En effet, ce qui suit est strictement indépendant des modes d'utilisation de NANOMÉTÉO et se retrouvera dans les trois types de fonctionnement. Comme les écrans relatifs au chapitre Mode HISTORIQUE sont influencés par cet artifice informatique, autant en "parler" ici, le reste coulera de source. Il s'agit d'un complément d'affichage dans le cadre bleu pour faciliter l'évaluation des intervalles de temps sur le graphe représentatif de l'historique. Considérons par exemple la Fig.101 qui ressemble étrangement à la Fig.97 pour laquelle les valeurs des paramètres sont identiques. Toutefois, des



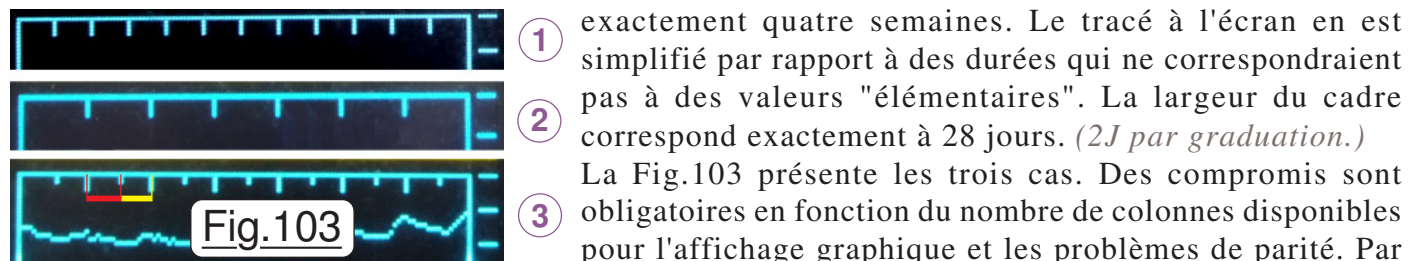


graduations verticales sont ajoutées "dans le cadre rouge". Elles correspondent au résultat de la Fig.98 qui annonce que la durée entre deux graduations est d'une journée. Nous sommes dans le cas particulier où un histogramme complet correspond à une semaine. Chaque "tranche" en largeur correspond à une journée écoulée.

L'opérateur n'a strictement rien de particulier à faire. Le logiciel examine la durée consignée entre deux échantillonnages. Il en déduit la durée totale de l'histogramme. Si ce dernier correspond à l'un des trois cas particuliers envisagés, alors les graduations sont tracées en conséquence avec un nombre de petits traits verticaux visant à faciliter l'observation du dessin. Naturellement, les particularités prises en compte dans ce croquis correspondent à des critères personnels. Vous pouvez fort bien préférer d'autres périodes temporelles, soit par remplacement de l'un des cas, soit en ajouter. Voici mes préférences avec en Fig.102 les valeurs temporelles à consigner :



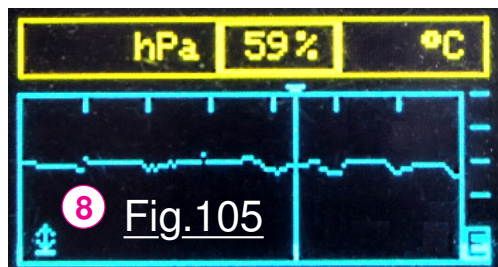
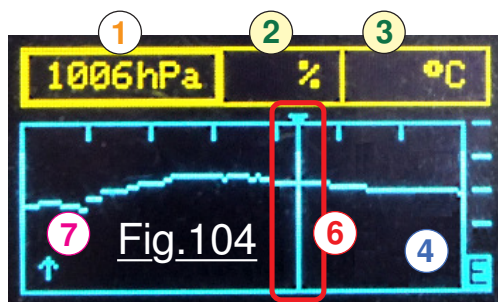
- (1) Pour l'opportunité de pouvoir analyser une variation rapide des paramètres météorologiques, par exemple le passage d'un front froid, **la plus courte durée d'enregistrement est d'une journée**. Le graphe est alors plus "fin". En contrepartie, la période totale mémorisée est plus restreinte.
- (2) Pour ma part en exploitation ou en sentinelle, je trouve qu'**une semaine est un bon compromis** à la fois sur la finesse du tracé et sur la durée d'enregistrement. C'est du reste cette préférence opérationnelle qui justifie dans mon croquis la période par défaut sur un RESET.
- (3) Troisième utilisation séduisante, **enregistrer sur une longue période**. (En espérant qu'il n'y aura pas de coupure secteur !) Par exemple c'est l'été et vous partez durant trois semaines. Veinards ! De retour, vous désirez consulter les évolutions qui ont influencé votre véranda durant votre absence. Sachant qu'une durée maximale de 45 jours est incontournable, j'ai opté pour un mois, ou plus



exactly quatre semaines. Le tracé à l'écran en est simplifié par rapport à des durées qui ne correspondraient pas à des valeurs "élémentaires". La largeur du cadre correspond exactement à 28 jours. (2J par graduation.) La Fig.103 présente les trois cas. Des compromis sont obligatoires en fonction du nombre de colonnes disponibles pour l'affichage graphique et les problèmes de parité. Par exemple une observation attentive du cas 3 montre que les intervalles impairs "rouge" font un PIXEL de plus en largeur que les graduations paires "jaunes". On remarque également que pour le cas 1 nous avons en haut à droite une graduation de plus que pour le choix 2 le "calcul" donnant alors une valeur exacte d'intervalles "dans le cadre". Dans tous les cas les graphes s'interprètent aisément.

### Le Mode HISTORIQUE.

Toujours dans le cas d'une utilisation ludique de notre petit appareil, cette fonction est prévue pour donner à l'opérateur la faculté d'analyser finement les évolutions "du passé". La Fig.104 illustre clairement le résultat à l'écran de cette option. Dans ce mode l'affichage est stable et une seule courbe est tracée à l'écran. Il est ainsi facile de l'observer tout à loisir. Comme le programme ne surveille plus la temporisation d'échantillonnages, pour nous prévenir que l'on risque d'augmenter la durée théorique de saisie, la lettre E est visualisée en bas et à droite, en 4 de la zone bleue. Pour savoir quel est le graphe tracé actuellement, le cadre jaune qui entoure la donnée numérique correspondante est d'une épaisseur double. Les deux autres cadres de données sont vides. Par exemple sur la Fig.104 en 1 on trouve la valeur de la pression, et en 2 et 3 les champs sont noirs, seuls les symboles des unités y figurent. La courbe tracée est donc celle relative à hPa c'est à dire l'évolution des pressions atmosphériques. La particularité notable de ce mode de visualisation, outre la stabilité



d'affichage, réside dans la présence d'un curseur vertical en **6**. En tournant le codeur rotatif dans un sens ou dans l'autre, on balaye à convenance le graphe sur quarante positions possibles, soit une toutes les trois "colonnes PIXEL". La valeur correspondante du paramètre visualisé est indiquée dans le cadre jaune "épais". Il est ainsi facile d'avoir à notre guise les évolutions numériques sur la période enregistrée. Si se sont les **valeurs extrêmes** qui vous intéressent : Passez en **mode VEILLE**, une option y est prévue. (*Explications plus avant.*)

**A**utre information qui peut vous concerner : La tendance symbolisée en zone **7**. Bien qu'il ne s'agisse pas à franchement parler d'historique, mais plus d'une information à consulter en temps réel, cette entité est donnée dans cette configuration, car le Mode EXPLOITATION est déjà encombré en bas à gauche de la zone graphique. Le symbole en **7** présente le sens de **variation entre la valeur actuelle et la dernière qui a été échantillonnée**. Par exemple sur la Fig.104 on déduit que la pression atmosphérique est en train d'augmenter. Sur la Fig.105 qui pour le même groupe de mesures visualise le pourcentage d'humidité relative, l'évolution **8** est à la stabilité. (*Pas de changement entre les deux mesures.*) Assez significatif est le cas de la Fig.106 pour les températures. En **9** on peut observer qu'actuellement elle est en train de diminuer. Dans le

graphe, chaque tranche couvre une journée. On observe assez clairement les baisses de température nocturnes et les remontées diurnes. Globalement la température a diminué entre -7J et -4J et elle remonte significativement durant les trois derniers jours. Il va falloir arroser ! Notez que si l'hygrométrie reste globalement assez constante, c'est que ce graphe a été enregistré dans la maison. Si les capteurs sont placés à l'extérieur, les variations sont plus importantes, et ce pour les trois variables.

#### Option pour FC+ court et pour FC- court.

C'est avec les touches du petit clavier que l'on sélectionne la variable actuellement présente à l'écran. Le changement se fait en permutation circulaire, dans un sens qui sera fonction de la touche utilisée.

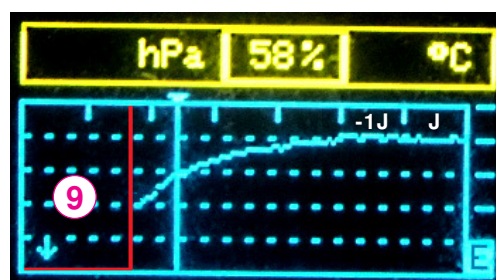
#### Option pour le capteur rotatif dans les sens ↺ et ↻.

Déjà explicité dans ce qui précède, tourner le capteur rotatif aura pour effet de déplacer le curseur **6** vers la gauche ou vers la droite. Quand il arrive "en butée" à gauche ou à droite, si l'on insiste sans changer de sens, l'index vertical "saute" de l'autre côté de l'écran.

#### Option pour Param. ↵.

Quand on fait appel à cette option que l'on peut considérer comme marginale, le graphe est complété par quatre lignes de pointillés qui situent sur toute la largeur du cadre bleu les hauteurs des niveaux intermédiaires. Ce tracé surcharge le graphe, et l'on n'aime pas forcément, raison pour laquelle cette possibilité est placée en option. Sur un RESET c'est l'écran le plus sobre qui sera proposé, la consigne n'étant pas mémorisée en EEPROM. Si vous trouvez intéressant de la pérenniser, la modification à apporter au croquis est élémentaire, et il reste largement de la place pour un octet de plus en EEPROM. L'effet visuel est montré sur la Fig.107 sur laquelle on observe la variation d'humidité de l'air. Une remise à zéro ou un RESET a été effectué il y a environ cinq jours et demi. Toute la zone réinitialisée est repérée par les deux traits rouges sur le graphe. Si l'on amène l'index de consultation des données à gauche du trait vertical rouge, la valeur sera inexorablement égale à zéro. Notez que l'humidité a augmenté régulièrement jusqu'à -1J. Puis elle est en train de diminuer, ce que confirme le petit symbole en **9**.

Fig.107





### Le Mode VEILLE.

Quittons les utilisations ludiques pour plonger sans transition dans une "application sérieuse". Notre petit "thermostat" perfectionné sera configuré avec soin, puis ... on l'oublie somptueusement. Il va alors avec endurance et discrétion s'occuper de surveiller avec une attention totale la température d'un local quelconque, ou de l'extérieur de votre maison, en fonction du lieu où seront disposés les capteurs. C'est exactement comme le congélateur placé dans le garage ou dans la buanderie. On lui fait totalement confiance, inutile d'aller voir son tableau qui en permanence indique les -24°C de la consigne. Ce n'est que lorsqu'il couine avec insistance de l'on réalise brusquement qu'il existe. *(Et aussi quand on cherche dans ses tiroirs les langoustes qui y sont préservées ... miam miam !)*

Adapté à la mission de sentinelle qui va surveiller la température d'hivernage de nos richesses botaniques, quand on aura fait joujou avec toutes les possibilités de NANOMÉTÉO, arrivera forcément le moment où ces petits plaisirs du début vont s'émousser. Titiller les diverses options de notre bébé va devenir aussi banal que de revoir un film *(Surtout si c'est un navet !)* pour la N<sup>ième</sup> fois. Convaincu de la fiabilité de notre appareil, nous n'irons le consulter qu'épisodiquement, juste pour vérifier que tout va bien. Dans ces conditions, il devient totalement inutile d'user 24H/24H les luminophores d'OLED. La première particularité du mode VEILLE, c'est précisément de générer un écran tout noir. En complément, l'utilisation des divers boutons permet de consulter l'appareil soit pour information météorologique, soit pour "vérification technique". La LED jaune flashe régulièrement pour témoigner du déroulement normal de la boucle de base.

#### Option pour FC+ court.

Elle délivre la page d'informations de la Fig.99 pour l'affichage des quatre valeurs temporelles qui gèrent l'échantillonnage. La LED verte clignote rapidement. Cliquer sur **FC- court** ou **long** fait passer à la **page des données relatives à la sentinelle**. **FC- court** ou **long** ramènent à l'écran tout noir.

#### Option pour FC- court.

L'écran affiche les valeurs "temps réel" des trois paramètres météo. Dans la zone du cadre bleu le graphe de la donnée qui était affichée dans le Mode EXPLOITATION au moment de passer en VEILLE est tracé avec l'historique actuel. La LED verte clignote rapidement et **FC- court** ou **long** retournent à l'écran tout noir.

#### Option pour le capteur rotatif dans le sens ↺.

Le programme réaffiche les valeurs numériques actuelles dans la zone jaune, mais surtout effectue une comparaison sur l'intégralité des trois historiques. Sous chaque cadre jaune individuel sont alors présentées la plus petite et la plus grande valeur des 120 échantillons mémorisés. Nous avons ainsi les données des six extrêmes sur la période correspondant à toute la largeur de l'écran. Les deux textes **Min** et **MAX** identifient cette option. Comme pour les autres options, **FC- court** ou **long** font revenir à l'écran tout noir.



Fig.108

#### Option pour le capteur rotatif dans le sens ↻.

Fig.109

C'est une fonction de maintenance qui permet de vérifier que les capteurs sont toujours opérationnels et qu'ils n'ont pas été dégradés par une gentille petite souris qui a grignoté la longue limande ou une grosse grosse araignée velue qui loge dans le protecteur. Le message complet de la Fig.109 est affiché sur l'écran. Le programme réalise alors un test d'état des capteurs. Si le dialogue retourne des informations correctes, alors le mot **NON** est effacé et l'écran rafraîchi. *(L'espace exagéré entre les deux mots du compte rendu final résulte donc d'une optimisation logicielle au prix d'une dégradation esthétique estimée acceptable.)* On se doute que la LED verte clignote avec frénésie, et que **FC- court** ou **long** provoquent un saut dans le néant.

Capteurs NON OK

**Dans le mode VEILLE, et également dans le mode EXPLOITATION, chaque fois que la LED jaune clignote il y a mesure des paramètres météorologiques. Si la température actuelle devient inférieure à celle imposée dans le seuil d'alerte, la LED rouge s'allume, éventuellement des BIPs répétitifs sont générés et le signal de chauffe est envoyé vers une éventuelle interface de puissance externe.**

### Option pour Param. ↩.

Le bouton à bascule cliqué à gauche **inverse l'état du booléen Mode\_securite\_meteo sans demande de confirmation** puis affiche son statu **OUI / NON**. La nouvelle valeur est également sauvegardée en EEPROM pour être systématiquement rechargée en RAM lors d'un RESET. (*Manuel ou sur retour alimentation secteur.*) L'influence de **Mode\_securite\_meteo** a largement été détaillée dans le chapitre *Comportement de NANOMÉTÉO sur un RESET*. Cette variable ne modifie que la façon dont redémarre la station météo quand elle est mise sous tension, et surtout dans notre cas lorsque l'énergie électrique est rétablie, notamment suite à une coupure sur le secteur.

### Initialisation des paramètres temporels et de l'alarme.

Logiquement la valeur du délai entre deux échantillonnages et les paramètres pour la sentinelle thermique seront rarement modifiés. Il importe donc que la procédure soit évidente pour ne pas avoir à chaque fois à se replonger dans le manuel. Le déroulement général de la saisie d'une option est résumé dans l'encadré de la page 46, nous allons détailler ici les diverses pages contextuelles disponibles pour configurer notre petite station météorologique.



**RAPPEL :** Pour les trois modes écran, **FC+ long** ouvre le menu de changement de la base de temps. **FC- long** fait passer à la saisie de la température de seuil pour la sentinelle et valide ou non l'alarme sonore. Ces deux fonctions sont étudiées plus avant.

### Option pour FC+ long.

Fig.110

Si en réponse à la demande de confirmation de la Fig.110 on valide la saisie, trois pages contextuelles vont s'enchaîner dans l'ordre des écrans de la Fig.111 avec pour chacun la valeur, le type d'information

Modifier  
Intervalle ?



Fig.111

en cours de saisie et en bas à gauche la valeur maximale admissible. Par exemple pour les minutes on a appuyé sur le bouton central du codeur rotatif et l'on modifie le **PAS** à 10 incréments. La sortie des secondes génère l'écran de la Fig.112 qui résume les valeurs actuelles. Enfin **FC+ long** provoque le retour en fonctionnement banal.

Nb Jours = 28.4  
Heures = 5  
Minutes = 40  
Secondes = 36  
Incements

Fig.112

### Option pour FC- long.

Le message **Changer Tmp Alarme ?** demande confirmation. Si on accepte d'entrer dans cette option, le premier écran contextuel de la Fig.113 se passe de commentaire.

Quand on sort de la saisie par **FC+ long** on ouvre un écran très sobre :

Alarme SONORE ?



Fig.113

La touche **FC+ long** ou *court* valide l'agaçant BIP BIP de la sentinelle thermique. Au contraire, la touche **FC- long** ou *court* laissera l'appareil silencieux comme une carpe. La sortie de cette ultime page conditionnant la sentinelle provoque un affichage tel que celui de la Fig.100 qui en résume les paramètres de comportement.

Nous avons passé en revue toutes les facettes de notre petite station météo, ainsi que les divers menus d'options qu'elle met à notre disposition. La fin de ce tutoriel sera consacrée au logiciel, pour en détailler quelques petites particularités, facilitant votre intervention si vous désirez l'adapter à vos préférences.



### Le programme complet qui fait vivre notre petite réalisation.

**T**otalisant ce jour 950 lignes comportant généralement plusieurs instructions, le décortiquer intégralement serait stupide. La grande majorité relève d'une programmation banale sans grande originalité. Le listage est amplement saupoudré de commentaires, il se lit presque comme un journal. Les noms utilisés pour les identificateurs sont minutieusement sélectionnés pour contribuer à la lisibilité du total. La capacité d'accueil de l'ATmega est largement consommée puisque le logiciel occupe 82% de l'espace disponible. Avouons qu'une optimisation à outrance n'a pas été jusqu'ici une préoccupation majeure car estimée non indispensable. C'est en réalité la place disponible pour les variables dynamiques qui impose une optimisation poussée pour l'organisation des données.

#### **Organisation des données dynamiques.**

**E**tudié en profondeur dans le didacticiel relatif à la réalisation des appareils de mesures, nous savons qu'une boulimie exagérée en consommation d'espace de mémoire dynamique peut engendrer rapidement une collision entre la PILE et le TAS. Je vous invite fortement à revoir le sujet si vous désirez cerner avec précision les informations qui suivent.

Pour ceux qui se contenteront d'un survol rapide de ce chapitre, contentez-vous de savoir que deux consommateurs de mémoire dynamique sont susceptibles de se goinfrer :

- Les "textes" que l'on insère dans le programme pour les affichages écran.
- Les variables TABLEAUX qui peuvent rapidement se gaver à leur tour.

Le premier remède quand on risque de manquer d'espace entre la PILE et le TAS consiste à placer tous les textes en EEPROM. Tout au moins ceux qui ne changent pas. C'est en général le cas pour la quasi totalité des affichages. C'est exactement la stratégie adoptée dans **P10\_NANO\_METEO.ino**. L'autre "angle d'attaque" consiste à réduire autant que possible la taille des données dynamiques, et en particulier les tableaux. Si ce sont des tableaux de constantes, comme le LOGO par exemple, il faut les logger en EEPROM, comme pour les textes. Si ce sont des tableaux de variables, il faut en minimiser la taille. En particulier les valeurs échantillonnées sont préservées dans trois tableaux.

#### **Traitement des échantillons et de leur mémorisation.**

**C**onceptuellement nous avons trois variables à mémoriser dans un historique. Compte tenu de la définition de l'écran graphique et des dimensions adoptées pour les dessins, nous avons opté pour 120 échantillons par variable. Nous disposons de 2048 octets pour la mémoire dynamique. Les variables globales en consomment déjà 822. Il ne faut pas oublier que le pointeur de PILE va venir consommer copieusement, car il doit logger les adresses de retour lors des appels à procédure, les points de retour quand se produisent des interruptions, ranger les variables passées par valeur etc. Bref, tout ce petit monde vient manifester sa fringale en octets. De rapides estimations ont montré qu'il fallait absolument se montrer prudent pour la taille des espaces affectées à l'historique des données. Décision initiale : Sauvegarder la pression, l'humidité relative et la température sur UN OCTET PAR ÉCHANTILLON. Ce choix initial n'est pas sans conséquence et impose une façon d'enregistrer les valeurs assez particulière. Examinons point par point la structure des données :

**Enregistrement de l'humidité relative.** On commence par l'hygrométrie car sa sauvegarde et sa récupération sont élémentaires. Les valeurs d'humidité relative qui correspondent à un pourcentage ne peuvent s'échelonner au maximum qu'entre 0 et 100. Pas de signe, la variation est largement dans la fourchette disponible sur un OCTET qui va de 0 à 254. La valeur mesurée sera directement stockée dans le tableau **MEMOIRE\_HYGROMETRIE[120]** à l'emplacement "courant".

**Enregistrement de la température.** Bien que les valeurs ne s'étalent pas sur des centaines de degrés, deux difficultés viennent compliquer un peu leur stockage. La première résulte du fait que le capteur fournit ses valeurs au dixième de degré près. Avec une fourchette de possibilité sur un OCTET il est impossible de mémoriser avec cette finesse, la donnée devrait passer à un entier et occuper de ce fait une place deux fois plus importante. Par ailleurs l'affichage en numérique dans la zone jaune prendrait deux caractères de plus, les données y seraient exagérément tassées. Comme le dixième de degré ne changera pas grand chose au ressenti et surtout pour le gel éventuel des plantes dans un local hivernal, décision a été prise de n'afficher qu'en degré, mais à l'arrondi au plus

approchant. La deuxième difficulté qui pose problème, c'est que l'on désire mesurer des températures négatives jusqu'à  $-10^{\circ}\text{C}$ , alors que sur un OCTET on ne peut coder que du positif. En outre, on désire que l'affichage graphique soit facile à interpréter avec cinq niveaux verticaux. D'où la limitation entre  $-10^{\circ}\text{C}$  et  $+40^{\circ}\text{C}$ . Pour satisfaire ces exigences, les températures seront mémorisées en les majorant de **+100**. Ainsi quelle que soit la température mesurée, on restera dans des valeurs positives qui sont acceptables sur un OCTET. Notez au passage que seul le graphe va talonner entre  $-10^{\circ}\text{C}$  et  $+40^{\circ}\text{C}$ , mais les valeurs numériques peuvent potentiellement aller de  $-100^{\circ}\text{C}$  à  $+154^{\circ}\text{C}$  donc ne seront pas affectées numériquement. L'affichage de la température en revanche sera faussé si on dépasse les  $99^{\circ}\text{C}$  car il est prévu sur deux chiffres significatifs. Ceci dit à moins d'une nouvelle glaciation ou de faire tomber les capteurs dans l'eau bouillante prévue pour le thé, ce risque est exclu.

- ① `Temperature = char (Capteur_BMP085.readTemperature()) + 100;`
- ② `// Arrondir au °C le plus proche :`
- ③ `if (abs(Capteur_BMP085.readTemperature()) - abs(int(Capteur_BMP085.readTemperature())) > 0.5)`
- ④ `if (Capteur_BMP085.readTemperature() > 0) Temperature++; else Temperature--;`

La séquence qui se charge d'effectuer une mesure et de sa préparation pour le stockage sur un OCTET est listée ci-dessus. On constate en ligne ① que c'est le capteur de pression qui sera également sollicité pour mesurer la température. La directive **char** en début d'instruction impose au compilateur d'effectuer des calculs en retournant le résultat sur un OCTET. À la valeur mesurée, cette instruction ajoute la constante **100**. Mais ce calcul fait perdre les décimales, car **char** impose en travail sur huit BITS. La ligne ② est un commentaire relatif aux deux lignes qui suivent. La ligne ③ a pour objet d'extraire la valeur de la décimale avec deux fois **abs** qui dans les calculs "élimine le signe". Enfin, si le chiffre des décimales dépasse cinq, en ④ on incrémente la grandeur affichée ou on la décrémente en fonction du signe de la température. On dispose ainsi d'une valeur arrondie avec un OFFSET de **100**.

**Enregistrement de la pression.** La valeur la plus faible jamais enregistrée à l'échelle mondiale avoisine 868hPa qui dépasse de beaucoup la possibilité de codage sur un OCTET si l'on désire afficher la valeur à 1hPa près. Suite à une analyse poussée et à différentes stratégies évaluées lors du développement, il a été décidé de se contenter de sauvegarder la valeur de la pression en relatif par rapport à une référence zéro de 940hPa. L'instruction qui traite la valeur barométrique est élémentaire :

`Pression = (Capteur_BMP085.readPressure() / 100) - 850;`

L'appel à la fonction `Capteur_BMP085.readPressure()` **retourne la valeur de la pression en centième d'hectopascals**. La division par **100** fournit la valeur entière. On retranche ensuite la constante **850**, la transposition 0 à 255 correspondant alors à la fourchette 850hPa à 1104hPa. Cette fourchette de valeurs couvre et dépasse les grandeurs des records jamais enregistrés, donc en affichage numérique NANOMÉTÉO n'a aucune restriction. C'est la présentation graphique qui modère un peu la performance par effet de saturation en haut et en bas du dessin de la courbe de variations. Il est évident que pour afficher les valeurs numériques à l'écran, les données mémorisées subissent des transformations réciproques, c'est à dire enlever **100** ou ajouter **850** ... pas de quoi s'affoler.

### Traitement de l'affichage graphique.

**R**ésultant de compromis pas aisés à formuler, les solutions adoptées résultent de plusieurs essais qui avaient pour but d'aboutir à des graphes pas trop plats, éviter des changements d'échelles pour simplifier le logiciel en le cantonnant dans une taille de code raisonnable. Enfin il a été décidé de n'avoir à droite qu'une division en cinq niveaux pour les trois paramètres, en vue de simplifier l'interprétation des graphes par l'utilisateur. C'est ce critère qui limite la plage de variations graphique des températures entre  $-10^{\circ}\text{C}$  et  $+40^{\circ}\text{C}$  dans la visualisation du cadre bleu. Pour ceux qui habitent des régions très froides, il serait possible de décaler "vers le bas" l'échelle des températures, et les contraindre dans la fourchette allant de  $-20^{\circ}\text{C}$  à  $+30^{\circ}\text{C}$ . Bref, il vous sera possible à votre guise d'adapter les limites à souhait. Quand les divers choix pour les limites imposées dans la fenêtre graphique ont été adoptés, il restait à transposer les coordonnées verticalement pour tenir compte des caractéristiques de l'écran OLED. Verticalement la dalle de luminophores du cadre bleu réservé



au dessin fait quarante cinq PIXELs de haut. De plus, la fonction qui permet de tracer un point présente un axe vertical dirigé du haut vers le bas. Les ordonnées du cadre (*Ordonnées : positions en hauteur.*) vont de **18** à **63**. En fonction de la valeur du paramètre dont on construit la courbe représentative de son historique, il faut transposer entre ces deux limites.

**A**vec toutes ces valeurs qui s'entrecroisent joyeusement, il devient un peu délicat de s'y retrouver et de comprendre clairement tous les choix effectués. Le mieux encore est de consulter la synthèse proposée en Fig.114 qui résume l'intégralité des éléments relatifs au traitement des pressions. En violet nous avons les valeurs entières que retournera le capteur de pression BMP085, les décimales étant éliminée après réception du dialogue série. En réalité le composant peut numériser entre 300hPa et 1100hPa, mais nous ne descendrons jamais aussi bas. (*Sauf si on monte très très très haut en altitude ... Wouaff le jeu de mots !*)

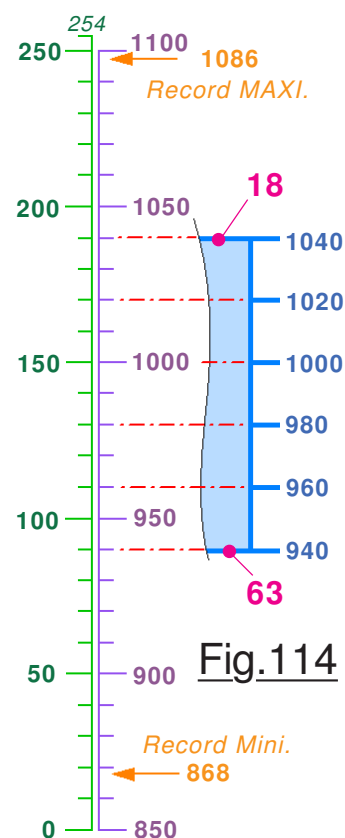
L'échelle verte donne les valeurs transposées par - **850** pour aboutir au codage sur un OCTET. On observe que la plage de mesure effective déborde largement des **records mondiaux jamais enregistrés**.

**T**raçer les variations à pleine échelle aboutirait sur l'écran OLED à des graphes presque sans relief, car dans nos régions les variations restent bien plus faibles. On a donc dans le traitement imposé un effet de dilatation verticale basé sur une valeur moyenne 990hPa. En étalant la plage 940hPa à 1040hPa sur toute la hauteur du rectangle réservé aux historiques, les courbes représentatives de variations présenteront deux fois plus d'amplitude. On retrouve après ce traitement les valeurs colorisées en bleu qui sur la Fig.98 sont rappelées dans la colonne 1. Enfin, en rose sur le dessin, on repère l'ordonnée du haut du cadre en **18** et celle du bas en **63**. Pour construire un histogramme, partant de la gauche vers la droite on va tracer sur une verticale le point représentatif correspondant à l'ordre de l'échantillon, et balayer avec un pointeur **PTR** les 120 valeurs de la gauche vers la droite. C'est la procédure `void Trace_un_triplet_de_mesures()` qui se charge pour chaque position horizontale de tracer le point figuratif correspondant à la variable représentée. Voici le début de cette procédure qui commence par traiter la pression :

```
void Trace_un_triplet_de_mesures() {
  ① OCTET = MEMOIRE_PRESSION[PTR];
  ② if (OCTET < 90) OCTET = 90;
  ③ if (OCTET > 190) OCTET = 190;
  ④ OCTET = map(OCTET, 90,190, 63, 18);
  ⑤ if ((Parametre == 0 && Mode_Ecran == 0) || (Courbe == 0 && Mode_Ecran == 1))
  ⑥   display.drawPixel(PTR, OCTET, WHITE);
```

*Dans le programme source, les lignes comportent plusieurs instructions. Ici le listage est "étalé" pour des raisons purement pédagogiques.*

Nous allons voir que finalement, si les choix opérationnels ont exigé de nombreuses tentatives, le traitement final reste très simple. En ① la variable **OCTET** reçoit la valeur de l'échantillon correspondant à la position latérale de **PTR**. Puis, si la valeur déborde par le "bas" de la fenêtre du ZOOM, en ② on rabote la valeur inférieure à 90. Une valeur de 90 pour l'octet mémorisé représente **940** pour le bas de l'écran graphique. Si le débordement se fait par le haut, en ③ toute valeur supérieure à 190 sera réduite à cette limite. Cette valeur mémorisée de 190 correspond aux **1040** du ZOOM vertical. Puis l'instruction **map** en ④ effectue une transposition qui inverse le sens d'évolution et en change la plage de variation. Cette instruction est assez magique et nous évite plusieurs lignes qui réaliseraient cette proportionnalité inversée. Le test en ⑤ permet de décider si ce paramètre doit être visualisé ou non. L'expression (**Parametre == 0 && Mode\_Ecran == 0**) sera vraie si on est en mode HISTORIQUE et que c'est le paramètre pression qui est sélectionné. Pour son compte l'expression (**Courbe == 0 && Mode\_Ecran == 1**) en mode EXPLOITATION alterne les affichages avec la variable **Courbe**. Enfin si l'un des deux tests donne **true**, en ⑥ on trace le point représentatif.



**ANALE DES RECORDS MONDIAUX.**

*Les valeurs annoncées en fonction des sources d'informations consultées varient très légèrement. On peut considérer comme correctes :*

Température la plus élevée au monde : + 56,7°C dans la Vallée de la Mort (U.S.) en 1913.  
Température la plus élevée en France : + 44,1 °C à Conqueyrac dans le Gard en 2003.  
Pression atmosphérique minimale : 870 hPa dans l'œil du typhon Tip dans le Pacifique en 1979.  
Pression atmosphérique minimale en Atlantique Nord : 882 hPa avec l'ouragan Wilma en 2005.  
La plus haute pression atmosphérique : 1083,8 hPa à Agata en Sibérie en décembre 1968.

**Un programme est toujours perfectible.**

**M**aladie qui frappe grand nombre de programmeurs dans le loisir logiciel, le "toujours plus" sévit avec une âpreté militaire. Le matériel est définitivement figé, le logiciel téléversé pour la dernière fois. OUF, cette fois c'est fini, plus question de modifier quoi que ce soit. On ne va quand même pas se laisser dominer par ce petit lutin malicieux nommé programmation.

**- NON MAIS ALORS !**

Avec autorité on range tous les outils, on clique sur l'icône qui va endormir l'ordinateur et l'on peut enfin passer à autre chose. (*Aller arroser les planplantes par exemple.*)

**- NON, NON et NON, plus de programmation !**

Cette volonté d'acier n'a d'égal que l'épais blindage d'un blockhaus invulnérable.

**- Et, Totoche, elle est si mignonne la petite grenouille du bocal météorologique.****- J'ai dit NON ! NON c'est NON haaaa mais alors !**

Cette fois c'est bien terminé. C'est la fin de la route, il n'y a plus de rails face à nous. Et puis le temps passe, et le mur de l'Atlantique se désagrège lentement. Bientôt seuls quelques panneaux pour les touristes signaleront que juste en face, il y a bien longtemps, existait une forteresse inviolable. La programmation c'est un peu ça, sauf que l'inébranlable résolution s'effrite généralement plus rapidement que le béton. Et PAFFFFfff, on craque et le P.C. reprend vie.

**Préambule au "toujours plus".**

**D**ifficile quand on a planché pendant des lustres sur un logiciel, et que l'on manipule avec satisfaction le nouveau joujou, de ne pas avoir inexorablement de nouvelles idées. Et les "dommage que", et autres "si j'y avais pensé avant" se bousculent. Il faut bien à un moment où à un autre décider que l'instant est venu de considérer notre objet comme totalement abouti. Le dernier programme complet qui intègre toutes les petites modifications d'amélioration frise les 1095 lignes de source. Sa taille présente une masse pondérale de 28122 OCTETS soit 91% de l'espace disponible. Il est grand temps de se calmer, *surtout que la prudence incite à conserver un peu de place pour corriger des éventuelles erreurs*. Il ne reste plus que 3 OCTETS de disponibles en EEPROM. Coté rentabilité on n'a pas fait dans la dentelle. Pour vous faciliter la vie, il ne vous sera pas nécessaire de reprendre le contenu de l'EEPROM sauf si vous voulez remplacer la Salamandre par la *gentille petite grenouille*. Tous les textes sont déjà en place. Voyons maintenant ce qui a été transmutationné.

**Sauvegarder en EEPROM l'option alarme SONORE.**

**C**ommencer par cette broutille coule de source. Sans trop gamberger, on triture quelques lignes de code en plus et le tour est joué. On démontre ainsi, sans y passer des nuits blanches à broyer du noir, qu'il est facile d'améliorer l'existant. C'est un peu de l'hypocrisie, car pousser un Arduinaute à venir titiller les muses du logiciel, c'est un peu comme expliquer à un Écolo invétéré que ce n'est pas citoyen de balancer la fiole vide de filtre solaire en matière thermoplastique sur la magnifique plage de sable fin, d'algues en train de sécher et de plaques de pétrole brut solidifiées. C'est le croquis **P11\_Complet\_version\_plus.ino** qui va se prendre de plein fouet les petites modifications explicitées dans ce chapitre. Dudule va finir par gagner, il l'aura sa petite grenouille. Du coup, tous les programmes relatifs aux perfectionnements apportés à **P10\_NANO\_METEO.ino** sont logés dans le dossier qui répond au nom étrange de **<Grenouille>**.



**Sauvegarder en EEPROM toutes les options.**

**F**inalement, il a été décidé de sauvegarder systématiquement en EEPROM toutes les configurations résultant d'options diverses. Leur nombre va augmenter avec les petits perfectionnements, mais effectuer une écriture en EEPROM et une restitution sur RESET n'exige que peu de code. C'est le développement de chaque option qui consomme, car il faut un texte de présentation, éventuellement donner un rappel de la valeur actuelle. Quand le menu de l'option est en place et ses effets correctement émuloés, deux échanges en EEPROM représentent un travail dérisoire. Par ailleurs, chaque option doit sauvegarder un booléen. Ce dernier ne consommera qu'un seul octet, pas de quoi dramatiser. Par exemple, actuellement on sauvegarde l'option des Lignes de pointillés et on la restitution sur RESET. Ainsi, si se produit une coupure secteur on retrouve l'intégralité du contexte sans avoir à reprogrammer l'appareil. Profitant de cette mutation, dans la version définitive du logiciel, si l'option des repères de niveaux est validée, on affiche les Lignes de pointillés dans les graphes des trois Modes écran. *(Quand on aime, on aime partout ...)*

**Sécurité météo à 45 jours.**

**I**nitialement, la séquence qui gère l'arrivée à 45 jours avait été copiée sur le programme source de PICOSYNTHE. On se contentait de générer des BIPs et un affichage qui incite l'utilisateur à faire un RESET pour réinitialiser le compteur interne au microcontrôleur. La fonction météo avec sentinelle change toutefois les conséquences potentielles. En effet, supposons que c'est l'hiver, il fait très froid et vous partez fêter Noël en famille pendant une semaine. Vous avez autre choses à penser qu'à la station météo qui discrètement et fidèlement gère le chauffage de votre petite serre. À peine parti de la maison : BIP BIP BIP BIP et plus personne pour s'en rendre compte. La température va baisser si l'interface est au repos et les plantes vont geler. Si l'interface de puissance est active, le radiateur va fonctionner en permanence jusqu'à votre retour. Bref, c'est la CATA !

**F**acile à parer ce problème : Il suffit dans la boucle infinie qui incite à faire un RESET de continuer à s'occuper de la sentinelle. La boucle infinie `void Comptage_maximum()` qui est activée quand on arrive au terme des possibilités du compteur électronique est la suivante :

```

1 void Comptage_maximum() {
2   Clic_sur_le_BP = false;
3   while (true) { // Boucle infinie
4     display.clear(); Affiche_cadre_du_haut(8); Aff_TEXTE_EEPROM(259, 19);
5     display.setCursor(42,29); Aff_TEXTE_EEPROM(631, 7);
6     display.setCursor(20,39); Aff_TEXTE_EEPROM(560, 14); display.update();
7     BIP(); digitalWrite(LED_Arduino, HIGH); digitalWrite(LED_Jaune, HIGH);
8     delay(1000); digitalWrite(LED_Arduino, LOW); digitalWrite(LED_Jaune, LOW);
9     display.clear(); display.update();
10    Mesurer_les_trois_parametres(); // Mesurer Humidite, Pression et Temperature.
11    if (Temperature - 100 < T_MINI_autorisee) digitalWrite(Sentinelle, LOW);
12    else digitalWrite(Sentinelle, HIGH); delay(3500);
13    Tester_le_BP_A0();
14    if (Clic_sur_le_BP) Sauvegarde_les_Echantillons_en_EEPROM(); } }
```

En ① on commence par indiquer que pour le moment aucun bouton poussoir n'a été activé. Puis en ② on utilise `while (true)`, et comme la condition est toujours vraie, le microcontrôleur est définitivement prisonnier de cette boucle infinie comprise entre les deux délimiteurs `{ }` et dont on ne peut sortir que sur un RESET. Les lignes ③ effacent l'écran puis présentent la page de la Fig.115 sur l'afficheur OLED. Pui, en ④ on génère un BIP et on allume les deux LED qui sont en vis à vis des deux boutons poussoir du clavier pour inciter l'opérateur à utiliser l'une des deux touches de fonction. Au bout d'une seconde en ⑤ on éteint les deux LEDs. Puis en ⑥ l'afficheur est forcé à "tout noir".

**45 J MAXI : RESET !**

**Si B.P. Fig.115  
SAU historique**

En ⑦ on mesure les trois paramètres, mais seule la **Temperature** sera prise en compte dans cette séquence de programme. La première ligne ⑧ commence par comparer la température mesurée à celle du seuil consignée dans **T\_MINI\_autorisee**. On doit enlever **100** car **Mesurer\_les\_trois\_parametres()** ajoute cette constante pour coder sur un OCTET toujours positif. Si on est en dessous du seuil la sortie **Sentinelle (LED Rouge.)** est activée. Dans le cas contraire la consigne repasse au repos. Puis la deuxième ligne ⑧ s'achève par une temporisation de 3,5 secondes. Ainsi on a un BIP toutes les 4,5 secondes. L'écran est noir une majorité du temps avec un rapport cyclique de 22% toujours dans le but d'économiser ses luminophores. En ⑨ le programme "regarde si l'un des deux B.P. est activé. Si OUI, en ⑩, comme sur la Fig.94 il y a passage au sous-programme qui demande si l'on désire sauvegarder en EEPROM. La sortie du sous-programme fait reboucler en ligne ②, et si c'est la touche **FC+** qui a été actionnée, les paramètres actuels sont inscrits en EEPROM.

### Interroger sur l'écart qui nous sépare des 45 jours.

Considéré comme un paramètre important pour l'utilisateur, il faut pouvoir en consulter la valeur actuelle à tout moment quand on le désire. Ajouter une fonction qui affiche le nombre de jours écoulés depuis le dernier RESET ainsi que la durée restante jusqu'à la limite maximale de 45 jours n'est pas du tout compliqué. Nous sommes dans du traitement banal. La difficulté par contre réside dans les protocoles d'exploitation, car le nombre de boutons disponibles est vraiment limité. On désire de plus rester dans des manipulations qui ne s'écartent pas trop de la philosophie générale d'ouverture des menus d'options. Il semble logique de disposer de cette possibilité en mode



VEILLE. De temps en temps, l'opérateur passe en revue les divers paramètres de "maintenance préventive" ou ceux relatifs à l'exploitation. Comme le mode VEILLE n'est pas logiquement celui où l'on va modifier la valeur de l'Intervalle ou celle du seuil de température sentinelle, **FC+** **long** et **FC- long** peuvent être réaffectées. Consulter l'écart à 45 Jours est assez marginal et sans conséquences. C'est donc

**FC- long** qui servira à ouvrir la fenêtre contextuelle de cette option. (**FC+ long** sert à l'option de sauvegarde automatique dont il est question dans le chapitre qui suit.) Le contenu de l'écran représenté sur la photographie de la Fig.116 se passe aisément de tout commentaire. Le retour à l'écran tout noir se fait librement avec **FC+** ou **FC- long** ou **court**.

### Option de Sauvegarde Automatique des échantillons.

Comme tout automatisme, l'imposer irait forcément à l'encontre de la convivialité. Imaginons que se trouve en EEPROM un historique particulier que vous désirez conserver un certain temps. Par exemple le dernier PROFIL de promenade que vous voulez montrer à des amis, (*Voir plus avant de quoi il retourne.*) ou une variation de pression barométrique assez peu courante dans vos annales locales. Dans ce cas, il faut pouvoir inhiber la fonction de sauvegarde automatique. À contrario, si votre préférence va sur l'utilisation permanente de NANOMÉTÉO en mode VEILLE, l'historique sur la période écoulée revêt à vos yeux de l'importance, surtout si vous avez configuré l'appareil pour une longue durée d'échantillonnage. Si une coupure secteur se produit, vous risquez de perdre plusieurs jours ou semaines d'enregistrements. La sauvegarde automatique est alors la parade à sélectionner. Si cette option est validée, **une sauvegarde dans l'EEPROM** des 120 échantillons pour les trois variables est déclenchée toutes les heures.

- **Quoi, mais téfou Totoche, à ce régime tu vas griller l'EEPROM !**
- **T'a raison Dudule, en principe on ne peut y réinscrire des données que 100000 fois.**
- **Du coup il va falloir changer la carte NANO vachement souvent.**
- **Ben oui Dudule, une fois tous les 11 ans et cinq mois environ.**
- **Elle va nous coûter un bras ta NANOGrenouille !**
- **Ben oui Dudule, à environ quatre euros par carte NANO on en aura pour 36 centimes par an.**

(100000 / 24 / 365 = 11,41 ans ... faut faire des économies !)





État actuel.

Fig.117

Attention : **FC+** ou **FC-** en sortie de cette page d'option conditionneront le nouvel état de ce paramètre.

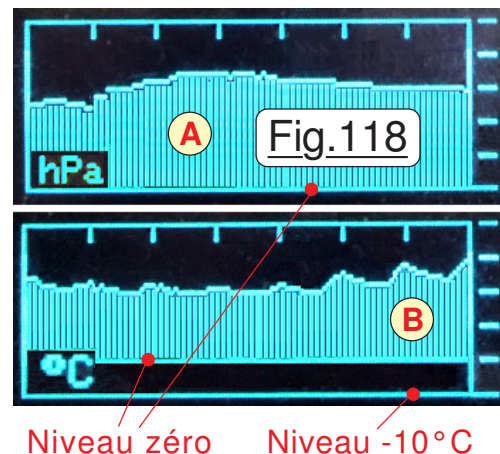
Comme précisé dans le chapitre précédent, c'est **FC+ long** qui servira à ouvrir la fenêtre contextuelle de cette option quand on se trouve en mode VEILLE. L'affichage indique alors l'état ACTUEL de la consigne qui se trouve en mémoire. Bien comprendre qu'en fonction du bouton poussoir actionné pour sortir, le paramètre sera réinscrit en RAM et en EEPROM. Donc bien faire attention au moment de **quitter en cliquant sur la bonne touche** du petit clavier.

### Traces points ou histogrammes type "surface".

Pour la modique somme de 170 octets de programme et de deux emplacements en mémoire dynamique, nous allons bénéficier d'une petite gâterie supplémentaire, à savoir deux présentations possibles pour les graphes historiques. En réalité, c'est pour le tracé du PROFIL de promenade en fonction altimètre que cette idée a émergé des profondeurs barométriques. Comme c'est un petit plus esthétique, autant le généraliser à tous les écrans graphiques. Naturellement, l'opérateur aura le choix entre chaque type de représentation, la variable consignée étant sauvegardée en EEPROM et restituée sur un RESET.

T rève de publicité, l'option est vendue. Pour comprendre la différence entre les deux modes de représentation, considérons la Fig.118 qu'il faut comparer aux Fig.104 et Fig.106 car il s'agit des mêmes historiques, sauf qu'ici on est en représentation "surface". Au lieu de tracer le point figuratif en hauteur, on relie verticalement tous les luminophores entre l'ordonnée zéro et la valeur au point considéré. Il importe dans ce mode de représentation de bien situer la référence zéro. Par exemple sur la Fig.118 **A** le zéro se trouve tout en bas au bord inférieur du cadre. Les barres verticales partent donc du bas jusqu'au pixel figuratif de la valeur de l'échantillon. Par contre, en **B** nous savons que le zéro pour les températures se trouve à la deuxième graduation en hauteur. Les lignes de "remplissage" partent donc de cette horizontale. Si les températures deviennent négatives, les traits seront tracés entre la référence zéro et en dessous pour l'amplitude de la température.

Q uand en mode VEILLE on sollicite **FC- court** il y a affichage temporaire des données du moment. Si vous sortez de cette page d'informations avec **FC+ court** ou **long** comme suggéré par la LED verte qui clignote rapidement, il y a retour banal à l'écran noir. Par contre, si vous quittez l'affichage graphique avec **FC- court** ou **long** il y a ouverture d'une page contextuelle pour définir cette option, avec affichage du texte **Graphes pleins ?**. Vous avez compris que répondre avec **FC+** validera l'option, alors que **FC-** imposera la représentation "filaire". Si l'option des lignes pointillées horizontales pour définir les quatre niveaux intermédiaires est active, elle reste valide. Toutefois, les barres verticales du graphe de "surface" les masquent si elles traversent ces niveaux.



Niveau zéro

Niveau -10°C

**NOTE :** Un autre petit perfectionnement est ajouté en mode VEILLE. Maintenant, chaque appui sur **FC-** en ouvrant la fenêtre graphique temporaire fait changer la nature de la donnée qui est visualisée. Ainsi on peut en trois ouvertures et fermetures successives visualiser les trois histogrammes. Ce n'est qu'une amélioration mineure, mais elle montre à quel point on peut à l'usage d'un appareil quelconque lui trouver de multiples personnalisations.



Ben ça alors PAPY, t'as cassé ta tirelire en forme de grenouille !

Ben oui fiston, pour acheter plein plein plein beaucoup de carte NANO pour les changer tous les onze ans.

### Enregistreur du PROFIL d'une promenade.

**C**rise sur le gâteau, c'est l'option qui fait vendre. C'est aussi indispensable que les super rétroviseurs obus profilés pour des vitesses supersoniques sur une automobile. Totalement inutiles, c'est le genre de détail qui peut faire craquer et entrer chez le concessionnaire. La fonction PROFIL appartient à ce genre d'extravagances, et sera d'autant plus déraisonnable qu'elle va se gloutonner pratiquement 1500 octets pour le code qui l'émule. (*Menus et affichages compris !*) Au diable l'avarice, dans le loisir on fait souvent des folies, la vie est belle, les oiseaux font cuicui et nous on va se cogner des montées et des descentes uniquement pour engranger du profil. Bref, un petit délire et ce d'autant plus irrationnel que l'on va "mettre le paquet" pour que ce soit beau !

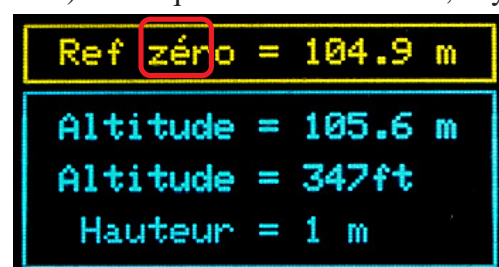
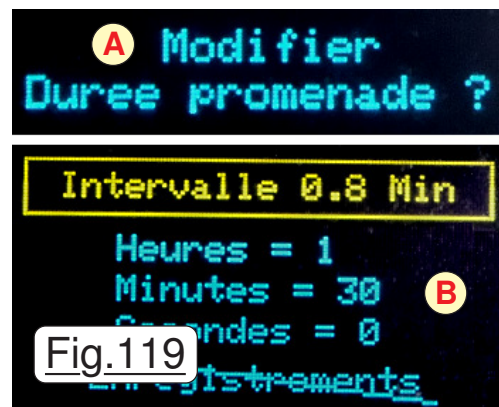
**L**idée concerne l'altimètre pour promeneur, elle consiste à enregistrer les hauteurs durant la ballade à des intervalles réguliers. Ainsi, à la demande, on peut faire afficher l'évolution des dénivelés en fonction du temps. On obtient de la sorte le PROFIL des montées et des descentes parcourues par le marcheur au cours de sa sortie. C'est du reste ce type de représentation qui a fait émerger l'idée des deux modes de représentation : Le filaire et le graphe de type "surface".

#### Définir la durée entre deux échantillonnages.

**A**vant de présenter les écrans d'affichage du PROFIL de la promenade, il me semble utile d'aborder la procédure opérationnelle qui permet de définir la durée entre la saisie de deux échantillons. Attention, ce n'est pas la valeur de l'altitude qui est mémorisée, mais le dénivelé. C'est à dire la différence d'altitude entre la référence de départ et celle du moment. Par rapport à la référence zéro au moment du départ, on peut indifféremment monter ou descendre. Nous aurons ainsi, aussi bien des hauteurs positives que négatives. La représentation du profil devra bien visualiser cette référence zéro correspondant au début de la sortie pédestre.

**E**tant en fonction ALTIMÈTRE, nous savons que **FC+ long** fait revenir au fonctionnement en centrale météorologique. Pour son compte, maintenant **FC- long** ouvre le menu Fig.119 **A** qui permet de saisir la durée qui devra s'écouler entre deux enregistrements de hauteur. Il serait possible comme pour la centrale météorologique de consigner directement ce délai. Ce n'est toutefois pas idéal, car ce que désire au final l'utilisateur, c'est d'obtenir un graphe qui pour la largeur de l'écran couvrira avec les 120 échantillons l'intégralité du parcours. Généralement, le promeneur a une idée assez précise de la durée de sa sortie. C'est donc cette dernière qui sera consignée en heures et minutes. Pour alléger le protocole de saisie, les secondes ne sont pas demandées. Quand on valide la valeur des minutes, on obtient l'écran **B** qui n'est pas sans rappeler celui de la Fig.99 car on se doute que c'est une procédure commune qui est invoquée pour saisir des durées. Les secondes sont forcées à zéro par le logiciel. Notez que la plus grande durée possible pour la sortie pédestre sera de **8H 59 Min** ... ce qui est déjà pas mal. Dans le cadre jaune est affichée la durée calculée de l'intervalle. La valeur maximale arrondie au dixième de minute sera de :  $((8 \times 60) + 59) / 120 = 4,49$  arrondi à 4,5. Donc la valeur dans le texte jaune sera toujours indiquée avec trois caractères dont un point décimal entre les deux chiffres significatifs.

**ATTENTION :** Quand on **consigne l'altitude de départ (Ref zéro)** en cliquant sur **FC- court**, il y a **remise à zéro du graphe**, donc effacement du PROFIL. C'est la raison pour laquelle, maintenant le programme demande confirmation en affichant le message **Changer Ref zéro ?**. Soulignons au passage une autre petite sucrerie informatique. Au prix injustifié d'un gaspillage d'OCTETS, divers textes ont maintenant des accentués comme montré sur l'exemple de la Fig.120 dans l'encadré rouge.





### Sauvegarde en EEPROM et restitution du PROFIL.

Compte tenu de la relative faible dépense en octets qu'exigent ces deux fonctions, il aurait été bien dommage de ne pas pouvoir immortaliser notre fabuleuse sortie au Mont Blanc pour rendre jaloux les copains. L'inverseur à bascule est disponible, il est choisi pour pouvoir faire appel à ces deux fonctions. Dans l'affichage courant des données d'altitude ou en mode veille l'inverseur génère un BIP d'erreur. **C'est uniquement en mode Affichage du PROFIL** que basculé à gauche on ouvre le menu pour l'option de rechargement avec possibilité d'accepter ou de refuser. Basculé à droite, si on valide en sortant, le PROFIL actuel sera "sculpté" en EEPROM.

Supposons que la promenade corresponde à environ trois heures de marche. Il est prévu un petit repas d'environ une heure, voir plus. Enregistrer durant tout ce temps "d'immobilité" engendrera un palier horizontal sur le graphe pas très porteur d'informations. Ce n'est pas génial. Une fonction qui figerait les enregistrements à la demande a été testée, mais elle conduisait à une débauche d'OCTETS avec un rapport qualité prix trop défavorable. L'idée a donc été abandonnée.

### Présentation graphique du PROFIL.

Activé avec le codeur rotatif, le graphe présente plusieurs échelles en hauteur. L'apparence sera directement fonction de l'option de tracé type points ou "graphes pleins" mémorisée dans la fonction station météo, cette variable étant sauvegardée en EEPROM et rétablie sur un RESET.

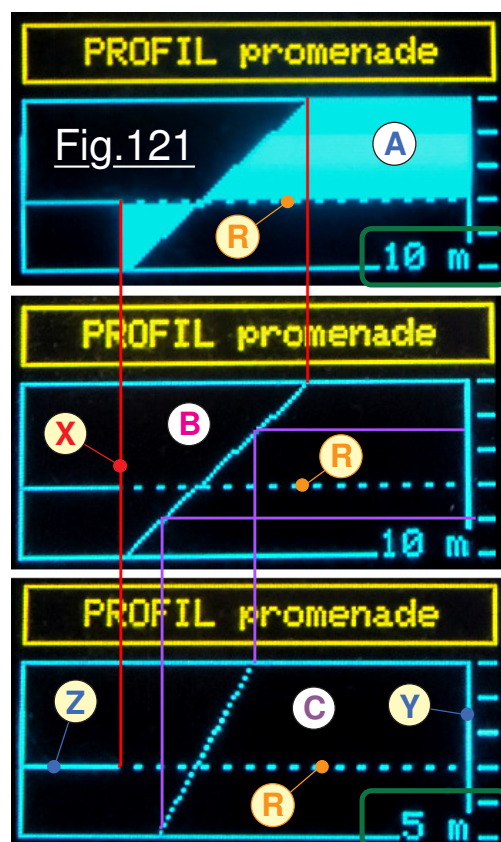
Considérons la Fig.121 relative à un PROFIL qui est en cours d'enregistrement. La promenade a commencé à -20m en **X**. Actuellement on "marche" en **Y** et la hauteur relative actuelle est d'environ 70m. Sur les trois graphes l'échelle sature vers le haut. Toute la zone **Z** à gauche de **X** vaut zéro puisque les enregistrements ont été effacés lors de la saisie de **Ref zéro**.

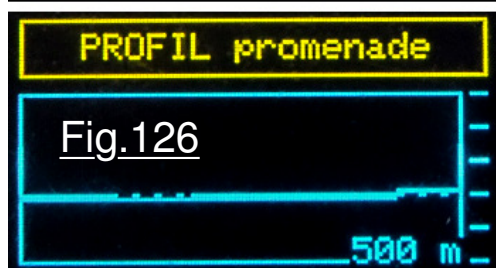
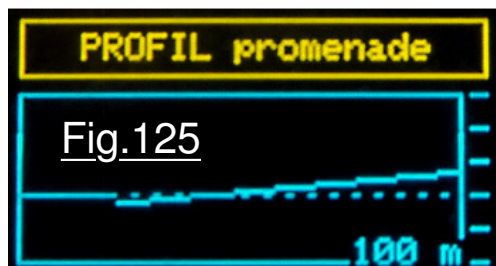
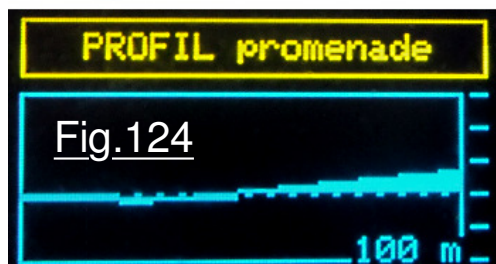
Quel que soit le mode de représentation, surface en **A** ou filaire en **B**, la **Référence zéro** est repérée en **R** par une ligne horizontale pointillée située à deux niveaux en hauteur sur le graphe. La sortie de la représentation type PROFIL se fait librement avec **FC+** ou **FC- long** ou **court**. Durant ce mode d'utilisation de l'altimètre, quand on tourne le codeur rotatif, on change la plage "verticale" des valeurs de dénivelé.

La différence de hauteur entre deux graduations de niveaux est indiquée en bas à droite. (Sur la Fig.121 dans l'encadré vert.) On peut ainsi librement changer l'amplification de représentation, mais la ligne de référence est immuable dans le cadre, il y aura en permanence un rapport 2/5 et 3/5 respectivement pour les descentes et les montées avec effet de saturation sur les deux bords bas et haut de la zone bleue. En **A** et **B** on tronque la représentation à +30m. En **C** avec une déviation de 5m par graduation, la coupure hors cadre se produit à +15m. (Correspondances par les traits violets ajoutés aux photographies.)

On peut ainsi librement changer l'amplification de représentation, mais la ligne de référence est immuable dans le cadre, il y aura en permanence un rapport 2/5 et 3/5 respectivement pour les descentes et les montées avec effet de saturation sur les deux bords bas et haut de la zone bleue. En **A** et **B** on tronque la représentation à +30m. En **C** avec une déviation de 5m par graduation, la coupure hors cadre se produit à +15m. (Correspondances par les traits violets ajoutés aux photographies.)

Actuellement, j'ai estimé que cinq coefficients d'amplification sont bien suffisants, à savoir : 5, 10, 50, 100 et 500 mètres par graduation. L'usage peut nous donner l'envie d'en ajouter d'autres. Par exemple une grande sensibilité de 2m par graduation, un intermédiaire de 20m/Grd ou pourquoi pas un 1000m/Grd ? Il sera facile dans les instructions **switch (Parametre)** de la procédure **void Affiche\_le\_denivele()** d'ajouter autant de coefficients que l'on veut ... mais attention à la boulimie en octets ! Comme on peut le constater sur les Fig.122 à Fig.126, plus l'amplitude verticale est importante, plus le diagramme manquera de relief. Toutes ces photographies correspondent à la même balade représentée à des amplifications différentes. Par exemple les deux affichages Fig.122 et Fig.123 sont analogues, le premier est de type diagramme surface, le deuxième est en représentation filaire. Sur les Fig.124 et Fig.125 la pente "se tasse" car on passe à 100m par graduation.





Subtilité de présentation qui participe "aux gaspillages d'octets, le texte qui indique l'échelle des niveaux en bas et à droite de l'écran est de longueur variable pour empiéter le moins possible sur le graphe car il masque ce dernier. C'est du luxe j'en conviens. Enfin, sur la Fig.126 le diagramme est tout raplapla car la hauteur totale va de -1000m à +1500m.

**TRICHERIE inavouable :** Non les amis, je n'ai pas effectué cette marche. Du reste on pourrait se demander comment on est passé de zéro relatif à -20m en un intervalle de temps nul. Une montée permanente aussi régulière n'est pas réaliste, car d'une part il est peu courant d'avoir une pente aussi linéaire, d'autre part un marcheur qui cadence à ce point son rythme est un "pro" de l'escalade. Ce graphe résulte d'une génération automatique lors des tests. Pour avoir des courbes plus variées, je vous invite fortement à aller au grand air profiter du soleil et d'emporter avec vous le beau joujou tout neuf ... Cette dénonciation anonyme sur ma sombre fourberie m'incite en outre à une transition facile :

#### Trucs et astuces de programmation.

Gouffre à temps insondable, la programmation peut arriver à des temps de développement irréalistes si l'on ne contourne pas certains délais temporels particulièrement consommateurs. Ce chapitre va dévoiler certaines petites astuces simples pour gagner un temps considérable. Elles ne sont pas spécifiques à cette application et peuvent se généraliser à tout projet. Vous avez un premier exemple dans le chapitre précédent pour lequel, la construction artificielle d'un profil a été générée dans une boucle qui calculait les nouvelles hauteurs par incrémentation.

Imaginez que pour tester la boucle infinie qui s'anime à 45 jours après le RESET il m'a fallu environ une cinquantaine d'essais. Si chaque fois on avait attendu 45 jours il aurait fallu six ans rien que pour cette séquence. Pour la mettre au point il suffit de l'invoquer inconditionnellement en tête de la boucle de base. Quand la séquence est bien au point, on enlève cette

instruction surabondante et provisoire. Puis, pour vérifier le déclenchement, on transforme

```
if (millis() > 3900000000) Comptage_maximum(); // Message d'alerte à 45,14 jours.
```

```
en if (millis() > 5000) Comptage_maximum();
```

qui va déclencher le processus cinq secondes après le RESET.

D'une façon générale, il faut en cours de développement "sauter" toutes les séquences qui impliquent de nombreuses manipulations et aller directement aux procédures en cours d'élaboration.

Par exemple dans `void setup()` on ajoute les deux instructions "roses" :

- ① `Heures = 1; Minutes = 24; Secondes = 0; // Valeurs pour une semaine.`
- ② `goto ICI; //xxxxxxxxxx`
- ③ `ATTENDRE_un_BP();}`
- ④ `ICI: //xxxxxxxxxx`
- ⑤ `display.setTextSize(1); Chronometre = 0; Altimetre = false;`

Arrivé en ① le programme a affiché la version et le LOGO. Normalement en ③ il attend de notre part un clic sur le clavier pour passer à la suite. Pour ne pas avoir chaque fois à effectuer cette manipulation, les lignes ② et ③ font oublier cette instruction en passant directement en ⑤. Certains



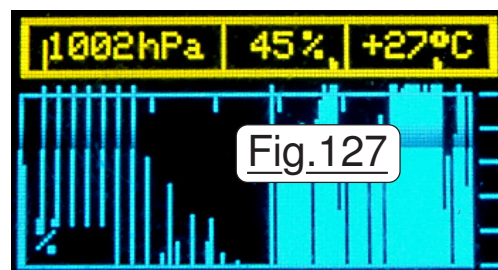
programmeurs, dont je fais partie, s'interdisent le **goto** dans les programmes car ils rendent les logiciels confus et délicat à interpréter. Toutefois, utilisés très provisoirement comme ici, cette instruction bannie reste logique et bienvenue. Autre astuce importante : Le nombre de lignes de code qui sont ainsi disséminées un peu partout dans le listage source peut devenir important. Quand tout est au point, il faut les effacer. Pour que cette opération de purge soit aisée et rapide :

- Chaque ligne de mise au point ne contient QUE du code surabondant pour pouvoir l'effacer entièrement sans avoir à se préoccuper de son contenu. (*Ainsi il n'y a pas de risque de perte d'instructions utiles.*)
- Chaque ligne de mise au point se termine par une remarque **//xxxxxxxxxx**. Quand on veut éliminer toutes ces instructions parasites sans en oublier, il suffit de confier ce texte au mode recherche.

Dans un ordre d'idée tout à fait semblable, le développement des affichages graphiques constitue une tâche longue et fastidieuse. Il faut tester les résultats aussi bien en utilisation Altimètre qu'en fonction station météorologique. Pour ne pas avoir à titiller chaque fois l'inverseur à bascule et d'avoir à confirmer, il suffit de commencer **void loop()** par une instruction du genre :

**Procedure\_ALTITUDE(); //xxxxxxxxxx.**

Autre boulimie temporelle : Obtenir un historique pour travailler les présentations d'écran. C'est long et fastidieux. Il faut imposer un délai entre échantillons d'une seconde. Puis attendre à chaque fois 120 échantillons soit deux minutes à ronger son frein et prendre des boutons sur le visage. INFERNAL ! Pour nous épargner cette corvée, dans la procédure d'initialisations **void setup()** se trouve une séquence de cinq lignes de code surabondant. Elles sont bien délimitées au début en ① et à la fin en ⑥ par deux remarques de mise en évidence globale. Quand les séquences sont au point il vaut mieux transformer ces lignes de source en remarques, ainsi par la suite si l'on veut reprendre le logiciel elles seront disponibles. Notez la remarque **//xxxxxxxxxx** pour respecter la discipline détaillée ci-avant.



```

① //===== Historique quelconque pour tests en développement de PGM. =====
② //for (PTR = 0; PTR < 120; PTR++) { //xxxxxxxxxx
③ //  MEMOIRE_PRESSION[PTR] = EEPROM.read(PTR);
④ //  MEMOIRE_HYGROMETRIE[PTR] = EEPROM.read(PTR+50);
⑤ //  MEMOIRE_TEMPERATURE[PTR] = EEPROM.read(PTR+100);
  //Mode_Ecran = 1;}
⑥ //=====

```

La ligne ② va créer rapidement 120 échantillons sur les trois variables. Pour avoir des valeurs quelconques, on se contente de "pomper" 120 valeurs dans la zone EEPROM du LOGO. En ③ on recopie celles du début, en ④ on lit cinquante emplacements plus loin, et en ⑤ on décale de cent emplacements par rapport au début de l'EEPROM. Par cet artifice nous obtenons trois historiques distincts, ce qui permet d'observer des différences quand on change de variable visualisée. Comme ces valeurs ne sont plus filtrées comme le sont celles issues des mesures météorologiques, nous avons des débordements vers le haut y compris dans le cadre jaune. C'est sans inconvénient pour tester les lignes pointillées, les types de graphes "surfaces" ou "lignes" etc.

Évidemment, il ne s'agit que de quelques exemples pour illustrer le propos, et vous en trouverez bien d'autres. Globalement, l'approche consiste à "sauter" toutes les séquences laborieuses qui demandent de nombreuses interventions du programmeur, et d'éluder par "simulation" toutes celles qui consomment par nature beaucoup de temps. Dernière suggestion lors des mises au point : Tester la place disponible entre la PILE et le TAS, surtout si le logiciel développé implique la création de tableaux. C'est la raison pour laquelle, dans tous mes "gros croquis", se trouvent les deux blocs dans **void setup()** délimités par des **//@@@@@@@@**. Vous pouvez être certains qu'avant de considérer un programme comme bon pour le service, je valide les lignes de ces deux séquences et en étudie avec attention le résultat. Pour NANOMÉTÉO le verdict annonce 183 octets de disponibles ce qui normalement est suffisant. Du reste notre croquis fonctionne correctement.

Des LOGOS À GOGO.C'est petit petit le format 32 x 30 ! 

**F**ranchement, s'il avait fallu que je me "cogne" une fois de plus la saisie OCTET par OCTET des valeurs pour coder le dessin en EEPROM, je me serais largement contenté de la petite salamandre. Bien qu'elle ait fichu une panique sans nom dans mes didacticiels, elle reste bien mignone et tout à fait artistique. Néanmoins, il semblerait plus logique pour une station météo, de remplacer cette signature personnelle par un LOGO typé qui suggère une grenouille.

Se construire un dessin "binaire" quelconque pour le faire afficher sur un écran quelconque peut conduire à des tableaux de valeurs importants. Avoir à transformer une image en nombres décimaux devient rapidement infaisable, surtout si le dessin présente une définition "meumeu".

Toujours dans le cadre de la programmation de loisir, pour automatiser ce genre d'activité, j'ai développé une méthode simple et un programme spécifique proposés sur le lien :

<http://forum.arduino.cc/index.php?topic=394550.0>

Vous y trouverez un fichier de type PDF dont le titre est repris pour ce chapitre. Il n'est pas question de détailler ici toute la procédure. Dans le dossier <Grenouille> se trouvent les croquis qui vont vous permettre d'expérimenter ce didacticiel à part si l'aventure vous tente. Vous y trouverez :

- **IMGtest.bmp** qui constitue le dessin EN NOIR ET BLANC que l'on désire comme LOGO.
- **P00\_Analyseur\_BMP.ino** qui sert à analyser le dessin et créer le texte **TABLEAU.txt**.
- Pour ceux qui n'ont pas de lecteur de carte SD et qui veulent analyser le contenu du croquis **P01\_Ecriture\_LOGO\_en\_EEPROM.ino** vous y trouverez **TABLEAU.txt** pour comparaison.
- Quand j'ai décidé de créer le nouveau dessin, **P02\_Complet\_avec\_grenouille.ino** les nombreuses modifications n'étaient pas encore envisagées. Seul le LOGO est modifié par rapport au programme **P10\_NANO\_METEO.ino** et donnait le résultat de la Fig.128 en **A**.

Si vous observez attentivement la version actuelle du croquis en **B** vous allez certainement constater que la grenouille y est moins "empâtée". Sa tête est plus fine, les deux membres du haut plus dégagés du corps. En résumé, le dessin a été corrigé. Je peux vous certifier que je ne me serais pas donné ce mal s'il avait fallu réécrire les valeurs numériques du modèle une à une dans le tableau. C'est grâce à la technique automatisée que je n'ai pas hésité à peaufiner mon petit dessin. **Fig.128**

### Contraintes pour concevoir le nouveau dessin.

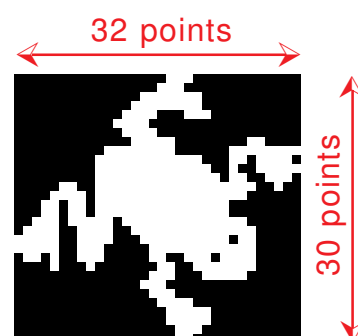
**R**emplacer des PIXELs par d'autres PIXELs dans un programme très avancé et volumineux impose des limites incontournables. La Salamandre est en place. Elle est constituée d'une matrice de 24 lignes de 5 octets soit une taille de 120 octets. Son tableau va de l'indice 0 à 119. Si l'on ne veut pas avoir à changer les textes de place dans l'EEPROM, ce qui impliquerait de revoir toutes les adresses relatives dans les procédures d'affichage, il faut que le nouveau dessin reste dans les limites de 120 OCTETs. De plus, si l'on ne veut pas avoir à développer un code complexe pour faire afficher le dessin, il faut que la taille en largeur soit un multiple de huit.

Définitions possibles (Largeurs x Hauteur) :

5 x 24 octets soit 40 x 24 PIXELs = 120 OCTETs (Salamandre.)

4 x 30 octets soit 32 x 30 PIXELs = 120 OCTETs (Grenouille.)

D'autres définitions seraient possibles mais elles conduisent à des finesses de tracé inférieures ou à des proportions incorrectes. Comme notre batracien est presque carré, le format le plus approprié est celui qui a été adopté. Comme les proportions ne sont plus identiques, les coordonnées d'affichage ont été corrigées. Notez que l'image est inversée verticalement, c'est pour ne pas avoir à corriger la séquence d'affichage sur OLED.





### Protéger les capteurs sédentaires.

Ultime chapitre de ce long cheminement en programmation de loisir sur Arduino, il me semble utile de vous décrire le petit coffret qui préservera les deux capteurs ainsi que l'électronique d'adaptation de niveau logique. Si l'utilisation de NANO MÉTÉO va dans le sens d'une installation fixe, le paramètre de faible encombrement est bien moins pertinent. On peut se permettre un volume du coffret plus grand. Il faut envisager un boîtier qui sera immobilisé sur un mur ou sur une poutre, la semelle de liaison devra tenir compte de cette hypothèse. En volume, mon prototype fait pratiquement la moitié de celui de NANO MÉTÉO complet en configuration autonome, ce qui pourrait sembler exagéré. Il présente les dimensions hors tout de 87mm x 45mm x 83mm ce qui pour un objet de ce type est tout à fait acceptable. On va lui demander une grande robustesse et une facilité d'installation. S'il est placé en hauteur, ce qui risque d'être le cas,

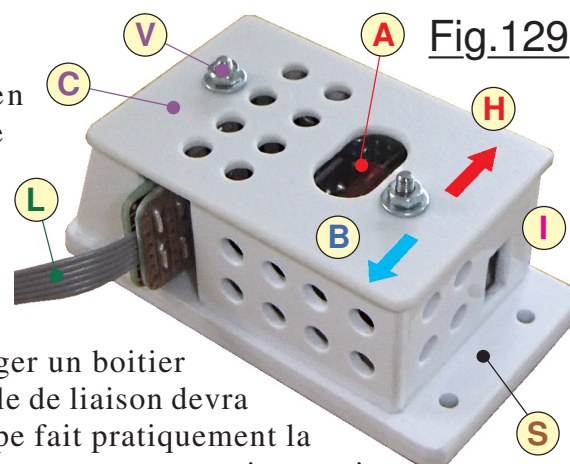


Fig.129

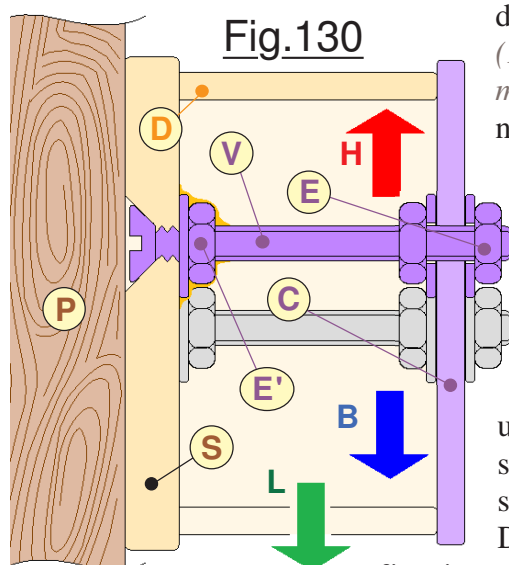


Fig.130

(Par exemple sur le flanc d'une poutre de toiture qui déborde du mur ...) venir y intervenir les bras en l'air avec des pinces à épiler ne sera pas très judicieux. Donc : Du solide facile à installer, facile à ouvrir et refermer. La Fig.129 présente le coffret sachant que **H** sera dirigée vers le haut et **B** vers le bas. La limande de liaison **L** se branche sur le dessous, évitant aux poussières de venir remplir cette petite boîte par l'ouverture de passage. Pour compléter cette protection, le dessus n'est pas percé. En **V** on distingue l'une des deux vis qui permettent la liaison et la

dépose du couvercle **C**. En **A** est pratiquée dans ce couvercle une large ouverture qui permet de bien voir la LED rouge, que l'on soit beaucoup plus bas que le coffret ou très décalé latéralement. La semelle **S** est rigide, constituée de deux plaques collées ensemble. D'épaisseur 6mm cette plaque qui présente quatre trous pour sa

fixation sur un mur ou une solive est très résistante. Notez au passage, en **I** sur le coté qui sera vertical dans la pratique, l'ouverture qui autorise le branchement pour piloter une interface de puissance. Considérons maintenant le dessin de la Fig.130 qui illustre l'une des particularités à ne pas négliger. Correctement orienté verticalement, cette vue transversale montre l'ensemble fixé à la poutre **P**. Le dessus **D** qui du reste sert à immobiliser le petit circuit imprimé de la Fig.72 n'est pas percé pour éviter la pénétration des poussières. On retrouve également en **L** la ligne de dialogue qui sort par le bas **B**. Pour fixer le couvercle il faut pouvoir serrer ou desserrer l'écrou **E** alors que nous ne pouvons saisir la vis **V** pour l'empêcher de tourner. Pour avoir une certitude absolue qu'elle restera totalement solidaire de la semelle **S**, on commence par l'y visser directement. Les filets qui seront en contact avec **S** sont imprégnés de vernis à ongles pour faire office de colle. **V** étant serrée sur **S**, on vient alors ajouter et serrer les écrous **E'**. Enfin, symbolisé par la surcharge orange, l'écrou **E'** et sa rondelle associée sont également copieusement englués dans du vernis à ongles. Si après tout ça **V** se desserre, je change de religion, de nationalité, de genre et pi tout, et pi tout et pi tout.

La fig.131 présente l'aspect trapu du protecteur vu coté semelle. Il importe d'utiliser pour **V** des vis F90 tête fraisées, car elles doivent être noyées dans la semelle et ne pas dépasser de cette dernière pour ne pas qu'elle en gêne une portée correcte. Quatre trous de fixation tels que **F** ont été prévu, mais une liaison en diagonale est plus que suffisante.

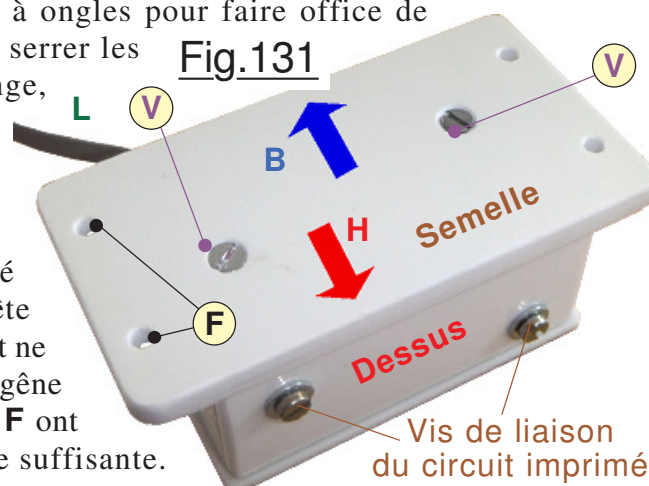


Fig.131

## Galerie d'images du protecteur des capteurs.



Fig.132

Le petit coté dans l'angle fait aussi office de nervure de renfort



Robustesse "militaire".

Le protecteur des capteurs avant intégration des éléments internes.

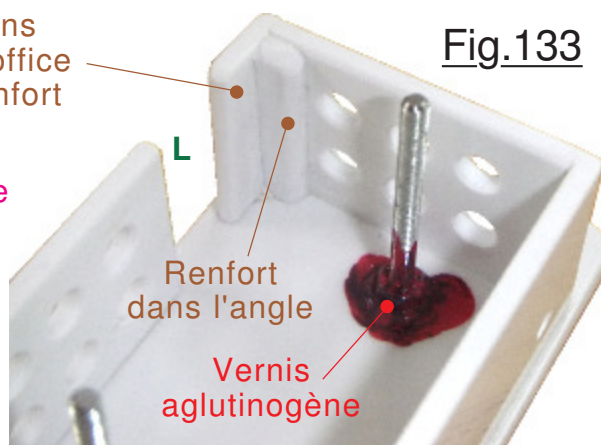


Fig.133

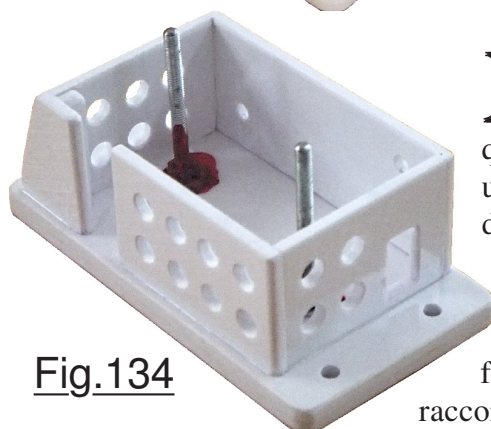


Fig.134

Éparpiller quelques photographies vous permettra de voir à la loupe comment le petit coffret est agencé et de vérifier que son contenu peut y être placé sans trop d'acrobaties. C'est aussi une façon d'achever ce laborieux et long didacticiel avec beaucoup d'images et peu de texte abscond. Tous ces petits trous percés dans le couvercle n'ont rien à voir avec une quelconque ventilation.

Au point de vue fonctionnel ils sont totalement inutiles, l'ouverture pour laisser passer la

fiche **F** de la limande de raccordement **L** étant largement suffisante pour permettre aux divers

capteurs de se trouver correctement assujettis à la température, à l'humidité relative et à la pression atmosphérique. C'est simplement cette maladie à toujours vouloir percer une foultitude de petits trous sous prétexte que ça fait beau. Donc si le gruyère ne vous séduit pas, contentez-vous des trois ouvertures de fonction, au contraire

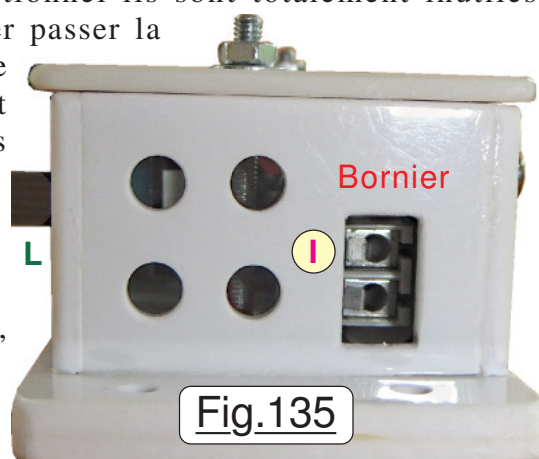


Fig.135

la poussière aura plus de mal à pénétrer.

Sur la Fig.135 le petit bornier **I** qui servira à piloter une éventuelle interface de puissance est bien visible, et centré sur l'ouverture latérale. Notez au passage en **R** sur la photographie de la Fig.136 l'utilisation d'écrous et de rondelles ajustés avec finesse en "hauteur" pour servir d'appui au couvercle et éviter de le déformer au moment du serrage des écrous **E**.

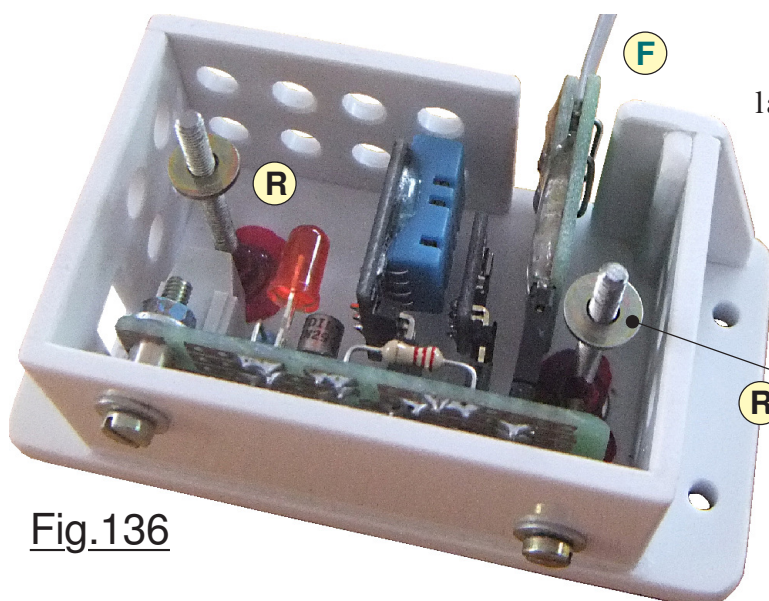
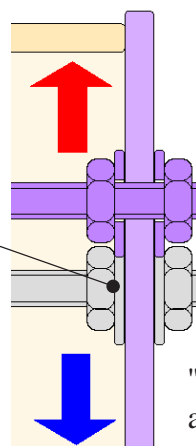


Fig.136



Hé Papy, pourquoi t'as acheté ce gros tanck dans un surplus militaire ?

Ben c'est pour tester la solidité du protecteur de mes capteurs fiston.



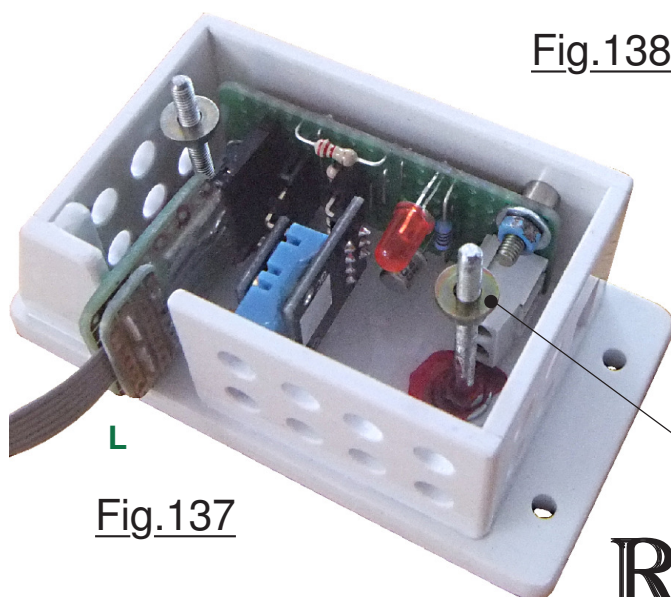
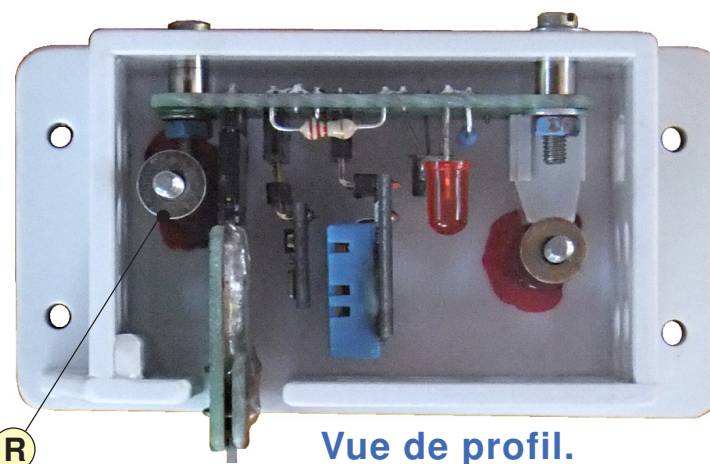


Fig.137



Vue de profil.

Fig.138

Rédiger un didacticiel oblige son auteur à faire des efforts particuliers pour expliciter les divers

aspects du système en cours de description, l'obligeant à relire en détail le programme à la loupe. Surgissent alors des faiblesses ou de nouvelles idées. Le croquis **P11\_Complet\_version\_plus.ino** n'échappe pas à cette assertion et a bénéficié d'une foule de légères améliorations.

Trop tard, le didacticiel "bouclé", et reprendre en détail certains chapitres bousculerait totalement la mise en page. Franchement je n'en ai vraiment pas le courage. Aussi, avec votre permission, (*Formule politiquement correcte d'une hypocrisie totale, puisque vous n'avez pas la possibilité de vous y opposer !*) je vais à peine en signaler un ou deux points.

Par exemple, dans la liste des bases de temps spécifique, le cas particulier de 45 jours a été ajouté et trace sur l'écran des graduations spécifiques.

Également non détaillé dans le didacticiel, mais pris en compte dans la notice d'utilisation, tourner dans le sens horaire le codeur incrémental fait afficher l'état actuel de la Sauvegarde AUTOMATIQUE et de SÉCURITÉ MÉTÉO quand on est en mode EXPLOITATION.

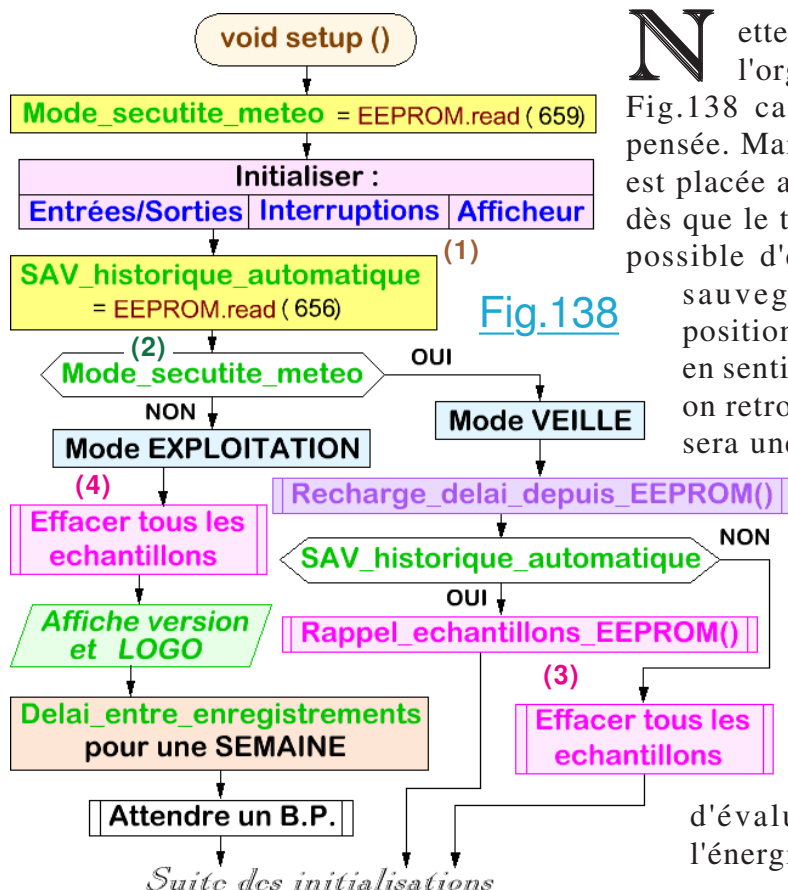


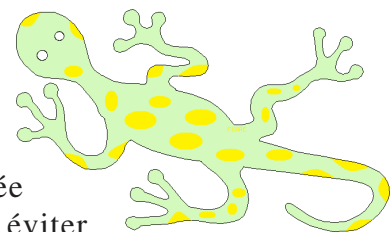
Fig.138

Nettement plus important sur le plan opérationnel, l'organigramme de la Fig.92 devient celui de la Fig.138 car la séquence de redémarrage est mieux pensée. Maintenant, la mise à jour en (1) de la variable est placée au début, pour pouvoir être prise en compte dès que le test (2) aura été effectué. Ainsi, en (3) il est possible d'effectuer une sélection. Si la variable de sauvegarde automatique des échantillons est positionnée à vraie, c'est que le système fonctionne en sentinelle thermique. Dans ce cas, sur un RESET on retrouve l'intégralité du contexte. La seule perte sera une légère perturbation de la "base de temps" car il y a suspension momentanée de l'échantillonnage. Si la coupure de l'énergie sur le secteur est de courte durée, la "dégradation" du graphe sera pratiquement insignifiante. Si on désire savoir qu'un RESET intempestif s'est produit durant notre absence, en (3) le graphe est effacé et en conservera la trace évidente. Il sera de surcroît possible d'évaluer sur les diagrammes à quel moment l'énergie a été rétablie dans l'historique. Quand le



système redémarre avec affichage de la version du programme et présentation du LOGO, en (4) on efface inconditionnellement les échantillons, car il n'y a aucune raison d'afficher un historique ancien qui ne représenterait probablement pas des données récentes.

**B**ien d'autres petites améliorations sont venu se greffer sur le croquis. Certaines on donné lieu à des corrections dans le tutoriel, donc vous n'en saurez jamais rien. Par exemple des améliorations de texte à l'écran, l'échelle graduée pour une journée mieux positionnée latéralement etc. D'autre ne sont pas abordées dans le didacticiel pour éviter de l'encombrer inutilement. Donc, si à l'usage vous constatez des petites différences entre la narration, les images et la réalité du comportement de NANOMÉTÉO, ces observations ne seront absolument pas anormales. Encore une petite recommandation : Avant de refermer définitivement le couvercle, vérifiez ... qu'une scongregneugneu de salamandre ne soit pas cachée dans les circuits !



**C**ette fois, c'est la bonne. Le long cheminement de loisir en programmation sur Arduino touche à sa fin. Au départ il ne s'agissait que d'évaluer la faculté d'une minuscule carte NANO à pouvoir assumer sa mission aussi bien que sa grande sœur l'ARDUINO UNO. Elle a passé les tests avec mention très bien et félicitation du jury, et compte tenu de son prix de vente tout à fait alléchant, à mon avis elle va chez moi supplanter la carte de développement dans toutes les applications où il n'y aura pas à brancher un SHIELD. Initialement j'avais envisagé le minimum vital, et puis, pris au jeu, c'est une véritable petite station météo qui va dans ma petite serre hivernale se charger de la surveillance, sentinelle infatigable. La fonction altimètre également peut se targuer d'émuler une application ludique très sympathique. Enfin, bien qu'il ne s'agisse plus d'un tutoriel sur la programmation en langage C, je ne peux que vous inviter à consulter le listage du programme source qui comporte un nombre assez important de séquences assez élaborée.

este que la prudence est toujours de mise. Évaluez avec attention la faisabilité compte tenu de vos aptitudes,

**R**car l'ensemble est compact et impose au bricoleur des savoirs-faires incontestables. Le pire serait d'aboutir à une déconvenue parce que vous avez sous-estimé les difficultés pratiques relatives à l'aspect réalisation. Pour le logiciel, je crois pouvoir rester serein. On ne peut pas exclure une ou deux petites vermines informatiques bien cachées, mais en l'état il semble donner entièrement satisfaction. **Si vous envisagez de piloter une interface de puissance**, au risque de me répéter, **ne vous engagez pas à la légère**. Cette voie peut vous amener à ouvrir des coffrets dans lesquels sont présentes des tensions dangereuses, ne serait-ce que l'arrivée 220V du réseau électrique.



**Ne jamais ouvrir le boîtier d'un dispositif électrique ou électronique quelconque si votre niveau technique dans les domaines impliqués n'est pas suffisant pour vous préserver de tout risque d'accident, tant matériel qu'humain.** En particulier relisez bien le chapitre relatif aux risques électriques largement détaillé dans le didacticiel sur la réalisation de petits laboratoires de mesures.



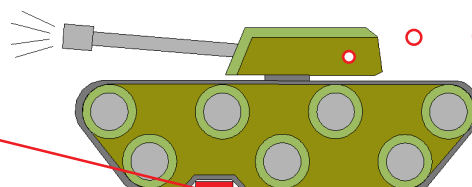
C'est le moment un peu tristounet où l'on doit se quitter.

*Je vous souhaite vraiment autant de plaisir à réaliser cette petite station météo d'un usage simple et de sa fonction altimètre pour promeneur, que j'en ai éprouvé à la créer et à développer le programme qui l'anime. Munie d'une électronique facile à programmer, je ne doute pas un instant, que titillé par les muses du génie logiciel, ces lutins malicieux vont vous pousser à vous faire du "sur mesure", activité d'autant plus gratifiante que voir évoluer "son bébé" procure toujours beaucoup de satisfaction.*

Amicalement : Nulentout

**BOOUUUMMM**

Solide le  
protecteur des  
capteurs !!!



Mince ... il  
était chargé !