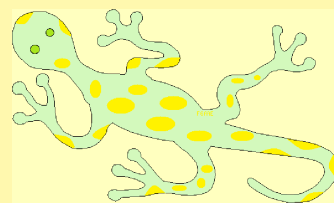


Cube 3D

Par Nulentout : Dimanche 26 Mai 2019.



Purement décoratif, cet objet lumineux constitue une petite folie. Non pas pour son coût qui reste tout à fait raisonnable, mais par le travail d'électronicien conséquent qu'il impose. Disponible en de nombreuses versions "sur les étagères" du commerce en ligne, il s'agit d'un objet lumineux volumique constitué de diodes électroluminescentes disposées de telle sorte à former un cube dont la définition est une matrice 8 x 8 x 8. Ce faible nombre de points confine à une tristounette "pixellisation" des images que l'on va créer, donnant à penser que l'effet visuel sera d'une pauvreté tragique. Pourtant, la troisième dimension change tout. Les formes phosphorescentes que l'on peut imaginer, malgré une si faible densité de points lumineux, restent d'une richesse étonnante, et ce d'autant plus que l'animation des tableaux créés leur donne une réelle dynamique, avec la sensation d'objets visuels particulièrement vivants et séduisants pour fêter le prochain Noël.

Presque utopique, la folie douce de ce projet ludique ne résulte pas de sa complexité technique, les circuits électroniques binaires restent d'une simplicité banalissime et fonctionnent du premier coup si l'on n'a pas commis d'erreur par rapport au schéma logique de base. C'est par le nombre de soudures à effectuer et de composants à préformer avec rigueur que devra traiter le bricoleur, que résulte ce que l'on pourrait presque qualifier d'oxymore. C'est simple, c'est facile ... mais méga rébarbatif. Le dessin de la Fig.1 ainsi que la photographie de la Fig.2 sont assez représentatifs du faible nombre de "PIXELS électroniques intégrés dans cette réalisation. Pourtant

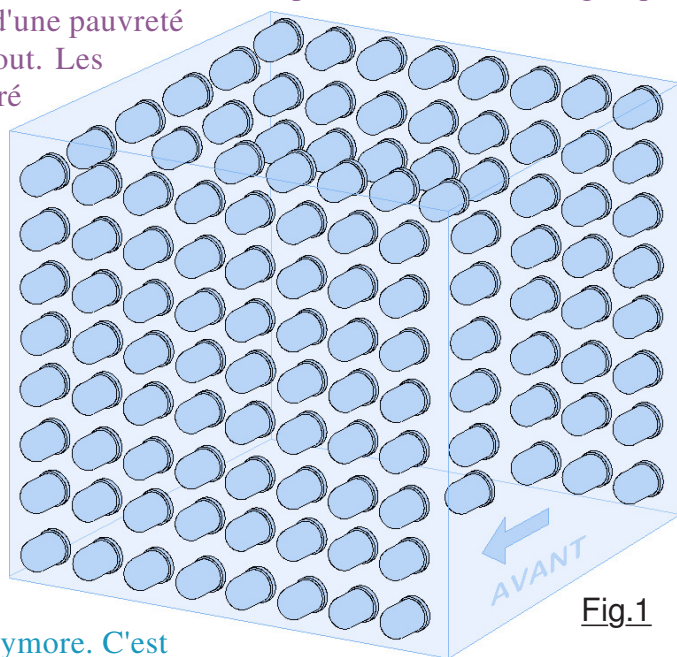


Fig.1



Fig.2

un rapide calcul va nous ramener à la réalité. Nous avons huit LEDs de large sur huit LEDs de haut, soit soixante quatre points lumineux par face. En profondeur on totalise huit tranches, c'est donc huit fois soixante quatre LEDs à assembler. Hé oui chère lectrice et cher lecteur, pas moins de cinq cent douze diodes à préformer. Comme chaque élément présente deux fils, c'est donc 1024 soudures à se coltiner sachant que ces éléments électroniques sont "en l'air".

Rassurez-vous, avec un peu de méthode, finalement la construction du cube lumineux n'est en rien un défi, et avec quelques heures d'assiduité et de patience on en vient à bout. Avec les deux circuits imprimés qui accompagnent cette électronique 3D, au total c'est environ 1500 soudures qu'il faudra soigner. Naturellement cette mise en garde n'a pas pour but de vous décourager, mais uniquement pour que vous mesuriez à l'avance le travail qui vous attend. Pour ma part quarante heures ont été nécessaires pour concrétiser l'intégralité du matériel, cube transparent de protection et châssis compris. **Page 1**

1) L'électronique utilisée et le montage matériel de base.

Contrairement à tous mes autres projets ludiques publiés sur la toile, c'est le seul pour lequel je n'avais aucune intention d'étudier quoi que ce soit. Séduit par l'idée de réaliser un cube lumineux pour les prochaines fêtes de fin d'année, j'ai donné dans la facilité et commandé un KIT. Tout est fourni, il n'y a rien à étudier, juste à souder les nombreux composants. Tombant dans la facilité, j'ai donc approvisionné un produit commercial et assemblé ce dernier qui a fonctionné du premier coup. Comme ce KIT constitue la base de ma réalisation, je fais exception à mes habitudes et vous livre l'adresse à laquelle j'ai commandé ce produit, vous aurez ainsi sa référence :

https://www.amazon.fr/dp/B0769HRN46/ref=pe_3044141_189395771_TE_dp_2

Pour un prix de vente de 30€ ce KIT est très sérieux et fournit l'intégralité des composants nécessaires, visserie comprise. Les circuits intégrés sont placés sur des supports et le circuit imprimé qui réunit le total est superbe. Si vous allez consulter le lien, vous constaterez que j'ai placé un avis ne comportant que trois étoiles. Le matériel fourni n'est pas en cause. Quand au manque de schéma, maintenant le problème est résolu puisque ce didacticiel élimine radicalement le problème auquel je me suis confronté.

Naturellement, vous vous demandez dans ces conditions l'argumentation qui justifie la pertinence de ce tutoriel, et pourquoi j'ai ajouté un circuit imprimé personnel à un ensemble qui d'origine fonctionnait parfaitement. La réponse à ce questionnement légitime tient en deux assertions :

- ***Autant le matériel est irréprochable, autant je trouve que le programme qui anime ce cube lumineux par l'entremise d'un microcontrôleur 80C52 est d'une tristesse à pleurer à mon avis.***
- ***Développer un logiciel pilotant un module Arduino NANO me séduit au plus haut point.***

C'est de loin l'écriture du programme qui a gloutonné une quantité d'heures colossales, mais la programmation étant ma passion, quand la météo incite à rester à l'intérieur, le temps investi ne compte pas. Bien entendu, pour vous cette phase est "gratuite" car mes programmes accompagnent ce tutoriel. Pour vous donner une idée, compte tenu des caractéristiques de la petite carte Arduino NANO, le programme **Cube_3D.ino** anime l'imagerie lumineuse durant plus de quarante minutes sans présenter

deux fois un thème identique. Soit le dessin plan ou volumique change, soit ses déplacements dans l'espace sont différents. L'encadré donné ci-contre résume le travail de soudage à effectuer. Pour le cube de LEDs la méthode utilisée sera à la base de la réussite et surtout est fondamentale pour que ce soit aisé à conduire. Bien entendu vous aurez sur ce point toutes les explications et photographies possibles. Il faut ajouter la réalisation du châssis en matériaux plastiques ainsi que le façonnage du cube de protection transparent en Altuglas. Pour ma part, outre les soudures, deux journées de huit heures ont été suffisantes pour aboutir au prototype entièrement achevé sur le plan matériel. Ajouté au logiciel qui anime le cube lumineux, vous trouverez l'utilitaire **Test_materiel.ino** qui permet de vérifier

entièrement le bon fonctionnement de l'électronique, encore qu'avec le circuit intégré 80C52 déjà pas mal d'observations sont possibles. Toutefois, le petit module de servitude **Test_materiel.ino** est mieux adapté pour une analyse minutieuse, car il met en évidence immédiatement un croisement toujours possible entre les liaisons qui vont de la carte Arduino au support du circuit 80C52 qui sert de connecteur. (*On peut aussi voir facilement des LEDs dont la luminosité est très inférieure à celle de la moyenne etc.*)

Bon, le "frère à souder" ne vous fait pas peur, ce chiffre 1538 n'est qu'une balise excitant notre persévérance. Seule contrainte matérielle : Avoir installé le compilateur et l'**IDE** pour pouvoir téléverser le code OBJET sur l'ATmega328. Mais si vous tentez l'aventure, c'est que probablement c'est déjà le cas. Ceci dit, rien interdit de vous limiter au KIT et dans ce cas seuls les conseils de réalisation seront utiles.

Bilan de soudage :

Circuit imprimé du KIT : (*Total : 346.*)

- * Supports : $(8 \times 20) + 18 + 40$,
- * Résistances, LEDs etc : $16 + 9 + 31$,
- * Picots de liaison avec le cube de LEDs : 72.

Circuit imprimé pour Arduino NANO : (*Total : 136.*)

- * Supports : $20 + 30$,
- * Composants divers : 22,
- * Liaisons locales filaires entre HE14 : 18,
- * Liaison bretelle de raccordement : 20,
- * Ponts électriques rigides : 26.

Cube de LEDs : (*Total : 1056.*)

- * LEDS : 1024,
- * Liaisons filaires avec les multiplexeurs : 16,
- * Fils de rigidification du haut : 16.

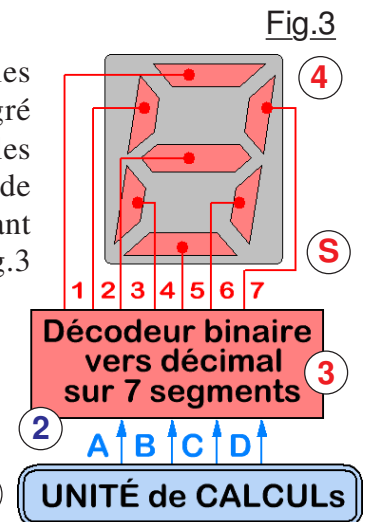
TOTAL : 1538 soudures.

2) L'architecture électronique initiale.

Imposé par l'achat d'un KIT commercial, nous n'avons pas vraiment l'opportunité d'effectuer des choix techniques, nous nous contenterons de ce qui a été élaboré par des professionnels. À analyser de près, et sans vouloir minimiser le mérite des concepteurs, l'ensemble est globalement très classique. En effet, nous sommes dans un domaine où LE MULTIPLEXAGE s'impose de façon évidente. (*Ce qu'est le multiplexage sera précisé dans la suite pour les personnes qui ne connaissent pas cette technique relativement banale dans le monde de la logique binaire.*) Naturellement, les techniciens concernés ont effectué des choix judicieux pour optimiser la mise en œuvre du microcontrôleur 80C52 qu'ils ont sélectionné pour cette application. Reste que pour les habitués de la logique TTL le schéma qu'ils ont peaufiné est clair, bien réfléchi probablement, et je vais dans ce qui suit vous en détailler rapidement le fonctionnement. Ainsi, que ce soit pour vérifier le bon fonctionnement ou pour "greffer" la carte Arduino, vos diagnostics et vos interventions seront plus efficaces. Mais avant ... parlons multiplexage.

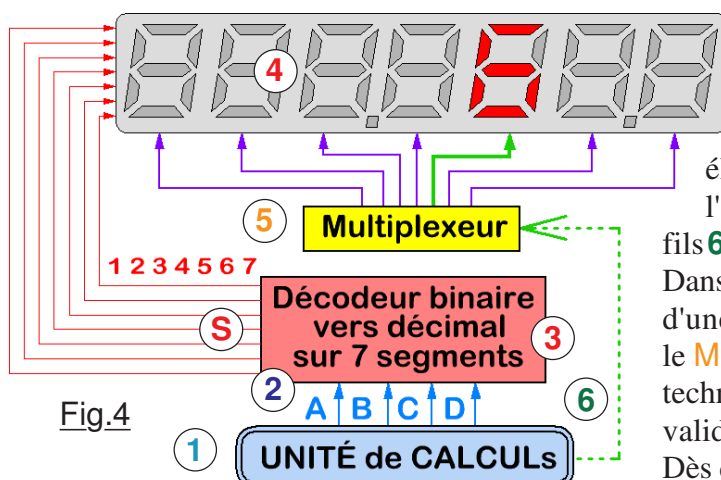
➤ Avant le multiplexage : le décodage.

Analysées avec l'avènement des premiers circuits intégrés logiques, les techniques de multiplexage consistent à utiliser un seul circuit intégré complexe, donc coûteux, pour piloter plusieurs composants. Par exemple, les afficheurs numériques sur des réveils radio, des calculatrices etc ont "abusé" de cette technique, aussi ce type d'application va nous servir d'exemple. Mais avant de "dispatcher" l'information, il faut en premier la décoder. Considérons la Fig.3 qui symbolise une petite zone fonctionnelle d'une calculatrice électronique. Sur son clavier vous avez effectué une multiplication, l'unité de traitement 1 doit alors visualiser le résultat sur son écran composé d'afficheurs à sept segments tels que 4. Hors l'unité de calcul fonctionne en binaire. Chaque chiffre à afficher sera disponible sur sa sortie en 2 sous la forme de quatre BITS notés ici traditionnellement A, B, C et D. (Hé oui, à cette époque on "parlait" en Hexadécimal ...) À l'une des combinaisons binaire telle que 0110, les poids forts étant à gauche, il faut faire correspondre une combinaison en S qui sur l'afficheur à sept segments 4 va allumer les bons éléments avec les sorties 1,2 ... 7. C'est le composant électronique 3 qui se charge de cette transposition. On se doute que la calculatrice va présenter les résultats de ses opérations sur plusieurs "digits". Par exemple, sur un produit travaillant jusqu'à 999.999.9 l'UNITÉ de CALCULS devra piloter sept décodeurs comme celui montré en 4.



➤ Les secrets du MULTIPLEXAGE.

Solution qui semble la plus naturelle, une quasi évidence consiste à doter l'appareil d'autant de décodeurs tels que celui en 3 qu'il y a d'afficheurs à piloter. Vu sa complexité, sur la "puce" en silicium qui le compose il faut agencer plusieurs centaines de transistors, diodes et autres éléments électroniques microscopiques pour un seul de ces décodeurs. Au début des circuits intégrés de technologie TTL, un tel composant présentait une grande complexité relative et restait coûteux. Aussi, pour diminuer le prix de revient du matériel, l'idée consiste à n'en placer qu'un seul, tous les afficheurs 4 étant branchés en parallèle.



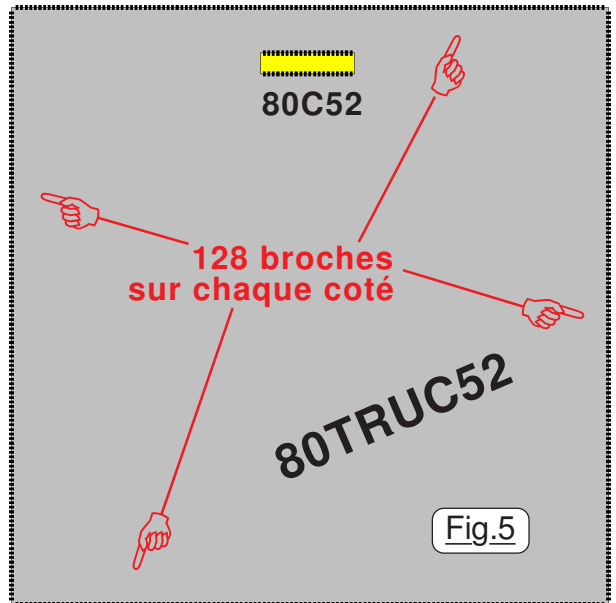
la technique de multiplexage dont la Fig.4 illustre l'architecture. On ajoute un circuit intégré de plus en 5 nommé Multiplexeur. Chacune de ses sorties aura pour mission de valider un afficheur élémentaire. Une seule passera en conduction. C'est l'UNITÉ de CALCULS qui par une ligne à plusieurs fils 6 précisera au Multiplexeur quel afficheur il doit activer. Dans notre exemple c'est celui des dizaines. Hors le résultat d'une opération sera exprimé sur plusieurs chiffres et le Multiplexeur ne peut en valider qu'un seul à la fois. La technique consiste à placer en S la valeur d'un digit, de le valider par 6 un court instant, puis passer au suivant. Dès que le balayage "circulaire"

dépasse une certaine fréquence, l'œil humain ne se rend plus compte qu'il y a clignotement rapide et l'on a la sensation d'un affichage stable et permanent. Du reste, si vous secouez une telle calculatrice équipée de chiffres électroluminescents, par opposition à des afficheurs LCD, vous constaterez que le balayage lumineux est "pointillé". C'est un effet typique d'un système de visualisation multiplexé.

➤ **On change d'époque, mais pas de technologie !**

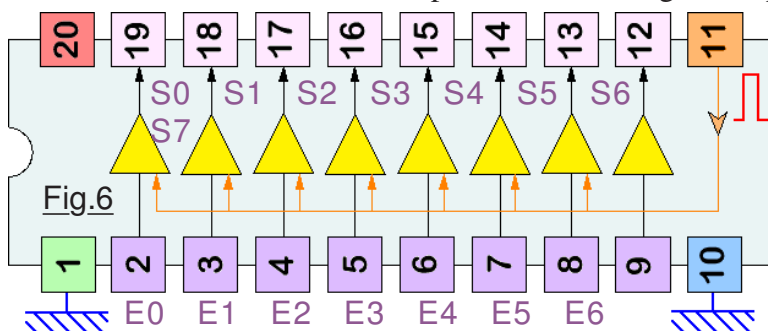
Environ un demi-siècle plus tard, réaliser des centaines de milliers de transistors et de diodes sur une puce est devenu banal, on ne compte plus ... ou plus exactement on change d'échelle. Aussi, le circuit d'un décodeur tel que celui repéré en 3 devient ridiculement insignifiant, on peut sur quelques millimètres carrés de silicium en loger des populations considérables. Aussi, terminés les multiplexeurs ringards de grand-père ... et bien pas tant que ça ! Considérez le microcontrôleur 80C52 qui équipe le KIT dont il est question en page 2. On ne peut pas dire qu'il soit spécialement minuscule. Si vous "explosiez" son encapsulation, vous seriez vraiment étonnés de voir à quel point la "surface électronique" est petite. Ce qui impose les dimensions de ce composant, c'est le nombre de broches situées à sa périphérie, séparées d'un dixième de pouce soit 2,54mm. Avec 40 broches encapsulées, on obtient le gros circuit livré avec le KIT.

Imaginez maintenant, qu'un spécialiste réalise un circuit intégré spécialisé pilotant directement toutes les LEDs du cube. Les cathodes seraient branchées sur une masse commune. Chaque anode serait reliée à une broche de notre 80TRUC52 qui devrait avoir au moins 512 broches. (*Sans compter quelques broches de plus pour alimenter en énergie et le programmer.*) La Fig.5 présente avec un respect total des échelles, en jaune le boîtier 40 broches du 80C52, et en gris un hypothétique 80TRUC52 sur lequel sont répartie à la périphérie les 512 broches pour piloter les LEDs. L'échelle est rigoureusement respectée. (*Pour ceux qui ont de la patience, vous pouvez vérifier avec une loupe que chaque côté du carré comporte 128 broches.*) Comme c'est le cas des circuits logiques "simples" comme le 68HC11 par exemple comportant un nombre de broches significatif, l'encapsulation n'est plus rectangulaire mais carrée. Le produit 80TRUC52 en question serait aussi grand que la table ! Bref, pour ne pas aboutir à du déraisonnable ... on va multiplexer.



➤ **Dominer c'est verrouiller.**

Piloter 512 sorties avec un microcontrôleur tel que l'ATmega328 qui dispose au maximum de 22 broches d'interfaçage est tout à fait possible, voir "banal". Il suffit de multiplexer. Dans notre cas particulier, il faut penser "volume". Nous allons donc multiplexer en largeur, en profondeur et en hauteur. Globalement la technique est assez classique, la particularité dans cette application réside dans la répartition des rôles et la structure géométrique d'organisation des multiplexeurs. Quand on regarde le schéma électronique de la carte du KIT on peut avoir comme un léger vertige, sans compter que sur ce schéma on n'a même pas représenté les 512 LEDs. Pas de panique, vous allez constater que c'est au final d'une simplicité presque évidente. La première étape consiste à comprendre comment fonctionne le multiplexeur de type 74HC573 utilisé dans ce projet. Tous ceux qui se sont fourvoyés dans l'univers binaire et de la LOGIQUE savent que d'une façon générale "tout va par 8" et que le mot le plus souvent rencontré est OCTET. Le circuit 74HC573 représenté sur la Fig.6 comporte assez naturellement huit lignes verrou.



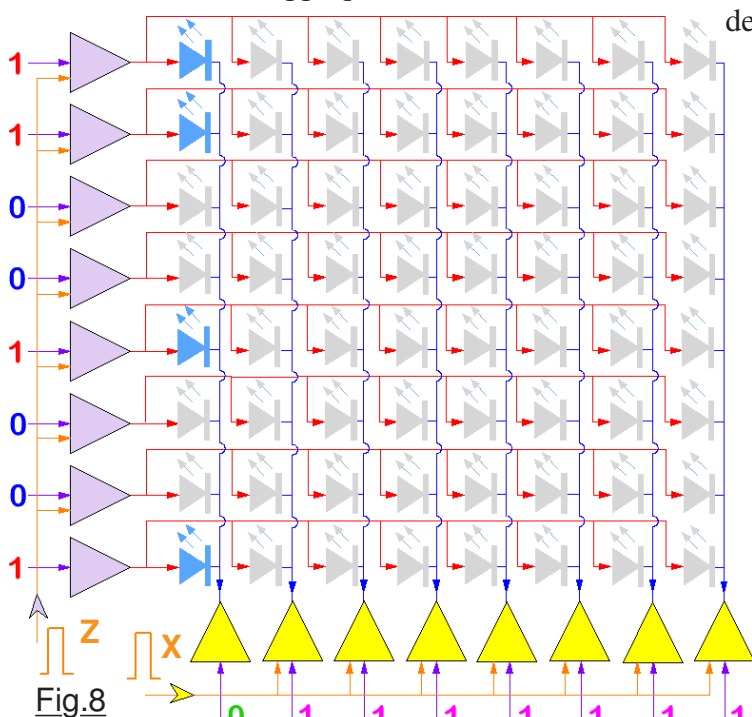
broche 10 est réunie à GND et 20 va au +5Vcc. Quand la broche 1 est soumise au "1" logique, (Donc à +5Vcc.) Les huit sorties S0 à S7 sont en haute impédance, c'est comme si le circuit n'était plus branché. Soudée en permanence à GND, donc en état logique "0", le circuit sera constamment fonctionnel. Comme déjà précisé ci-avant, on rencontre huit "verrous" avec

pour chacun une entrée **En** et une sortie **Sn**. C'est la broche **11** qui sert à piloter ce module. Lorsque cette broche nommée **Latch Enable** sur les documents de caractéristiques est forcée à l'état "1", les huit amplificateurs deviennent transparents. C'est à dire que les états logiques des entrées **En** sont recopiés sur les sorties **Sn**. Quand cette broche **11** repasse à "0", les états présents sur les entrées sont mémorisés et restent figés tant que **Latch Enable** sera maintenue à l'état électrique de **GND**. Pour résumer le comportement du circuit 74HC573 il suffit de retenir que ce dernier mémorise sur ses huit sorties les états logiques de ses huit entrées lorsque la broche **11** reçoit une courte impulsion à l'état logique "1". L'impulsion de verrouillage peut être aussi longue que l'on veut. Toutefois, pour des raisons de rapidité de fonctionnement du programme, on va descendre à la μS , ces circuits pouvant sans problème cadencer à plus de 10MHz.

➤ **Pas de complexe ... on va multiplexer directement "en surface".**

Pour utiliser un vocable populaire ... on va se taper une tranche. (*Tranche verticale dans le cube.*) Avant de chercher à créer une image lumineuse, un petit coup d'œil sur le tableau de la Fig.7 présente les quatre combinaisons possible d'état d'une LED en fonction des tensions appliquées sur ses deux électrodes. Pour les cas 1 et 2 la diode reste sans vie, car la tension aux bornes est nulle. Pour le cas 3 punition analogue, car la différence de potentiel dépasse le seuil de conduction mais se trouve appliquée en inverse. Seule la combinaison 4 engendre l'éclairage du composant car nous avons une tension suffisante et appliquée dans le sens de la conduction. Il importe

Cas	Anode	Cathode	État
1	GNG	GNG	Éteinte
2	+Vcc	+Vcc	Éteinte
3	GNG	+Vcc	Éteinte
4	+Vcc	GNG	Allumée



de noter qu'en inverse il ne faut pas dépasser la tension de claquage sous peine d'endommager le composant. Ici aucun risque car cette tension critique dépasse les 50v. N'oublions pas non plus qu'une résistance sera mise en série pour limiter le courant traversant la diode, car un maximum est spécifié par le fournisseur. Considérons maintenant la Fig.8 qui constitue la base de tout multiplexage croisé. Horizontalement on trouve un 74HC573 piloté en **X**. Verticalement un deuxième octuple verrou est commandé par sa broche **Latch Enable** en **Z**. Par le truchement de ces deux circuits intégrés on devient capable d'allumer individuellement n'importe quelle LED dans cette matrice de 8 fois 8 tout en ne monopolisant que 16 broches d'interfaçage. Examinons la technique utilisée, qui précisément consiste à multiplexer : Dans la structure de ce schéma, **on ne peut piloter individuellement chaque diode électroluminescente que sur une colonne, les autres devant rester éteintes**.

Pour cet exemple, on montre le cas où c'est la colonne la plus à gauche qui est traitée. Le processus est le suivant : 1) On applique des "1" sur les huit entrées du multiplexeur **X** et l'on transfère ces états sur ses sorties par une courte impulsion à +5Vcc sur **X**. Toutes les LEDs sont alors éteintes, car on force leurs cathodes au positif. (*Cas 2 ou cas 3 sur la Fig.7 en fonction de l'état sur leurs anodes.*)

2) On applique une combinaison binaire sur les entrées du multiplexeur **Z**, puis par une courte impulsion à +5Vcc sur **Z** on mémorise ces états sur les sorties horizontales. Pour le moment l'ensemble des LEDs reste sans vie puisque les cathodes sont toutes au positif.

3) On applique un état "0" sur l'opérateur qui pilote la colonne sur laquelle on désire gérer les points lumineux, les sept autres entrées reçoivent un "1" pour rendre inertes les LEDs de leurs colonnes. Immédiatement les LEDs de la colonne de gauche recevant un "1" sur leurs anodes vont s'allumer. Comme tous les états de la matrice sont mémorisés par les octuples verrous, cette situation est statique et durera aussi longtemps que l'on ne modifiera pas l'une des broches de pilotage **X** ou **Z**. Hors on désire que toute l'image soit traitée, que tous les "PIXELS" participent au dessin. C'est ici que le balayage "transversal" devient indispensable. On laisse la colonne de gauche illuminée un court instant, par

exemple 10mS. Puis on recommence le cycle décrit. On reprend intégralement les trois étapes sauf que pour chaque tranche, en phase 3 on change de colonne. Ce balayage peut s'effectuer librement de la gauche vers la droite ou réciproquement. La seule condition à respecter, c'est qu'il soit suffisamment rapide pour ne pas engendrer un phénomène parasite de scintillement ou de clignotement.

➤ **Multiplexons le multiplexage.**

Préserver de belles compositions planes verticales, c'est déjà bien pour agrémenter notre sapin de Noël. Par contre, nous sommes encore loin d'atteindre notre objectif de volume. Et bien pas tant que ça, passer à la troisième dimension reste élémentaire. Il suffit tout simplement de faire huit fois ce que montrait la Fig.8 et encore, j'exagère considérablement, car pour le cube lumineux entier un seul multiplexeur "vertical" est suffisant. Voyons

ensemble la structure que va prendre le

schéma complet, sachant que nous allons nous "contenter" de réaliser huit tranches de LEDs comme celle de la Fig.8 pilotées chacune par un circuit mémoire.

Sur la Fig.9 les "tranches verticales" sont repérées de 0 à 7, seuls les deux plans de LEDs extrêmes sont représentés. À

la base de chaque plan, la direction **X** est commandée par les huit verrous binaires intégrés dans un multiplexeur de type 74HC573. Huit composants de cette référence sont répartis dans la direction **Y**, chacun étant affecté à un plan vertical. Comme le multiplexage consiste intrinsèquement à placer les éléments pilotés en parallèle, c'est ce que l'on fait dans la direction **Z**. Chaque plan horizontal d'orientation XY sera contrôlé par l'une des huit sorties de l'octuple verrou dont l'encapsulation est représentée ici,

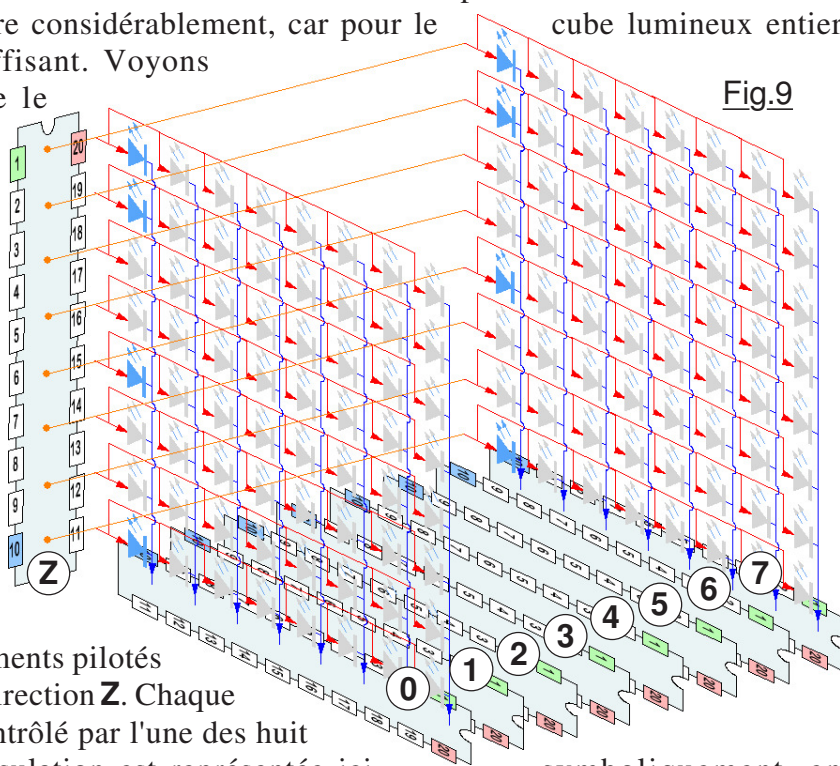
symboliquement, en position verticale. Il nous reste maintenant à résoudre l'un des problèmes intrinsèque au multiplexage, la gestion du courant électrique délivré par les multiplexeurs et traversant les LEDs.

➤ **Multiplexer c'est avant tout "pulser".**

Considérons la Fig.9 et évaluons le courant que doit fournir chaque opérateur de l'un des multiplexeurs. On va supposer dans ces explications que l'on désire faire fonctionner chaque LED sous 10mA pour obtenir une luminosité importante. Naturellement, nous avons au préalable consulté le document technique des modèles adoptés dans la réalisation. Cette intensité correspond à la valeur "statique" acceptable sans problème par le composant. *Quand une telle étude de faisabilité est engagée, il faut toujours la conduire dans le pire des cas* qui ici correspond à un cube dont toutes les LEDs sont allumées. Chaque sortie des multiplexeurs 0 à 8 doit fournir le courant simultanément pour huit diodes électroluminescentes soit 80mA. OK, c'est parfaitement compatible avec la *sortance* des 74HC573. Pour les opérateurs binaires du verrou contrôlant la direction **Z**, chaque amplificateur doit alimenter l'intégralité d'un plan horizontal soit soixante quatre éléments. Le courant total si chaque individu "pompe" 10mA sera alors de 640mA. **OUPS, ça commence à chauffer !**

Ce n'est que la partie émergée de l'iceberg. En effet, si vraiment l'on désire que la lumière fournie par chaque élément corresponde à l'intensité de 10mA, il faut tenir compte du mécanisme de multiplexage. Hors le pilotage du cube dans son ensemble n'est pas statique. Dans la direction verticale on doit balayer chaque plan l'un après l'autre. Quand un niveau est alimenté, nous avons vu que tous les autres sont éteints. Du coup, chaque tranche de direction XY n'est active qu'un huitième du temps. Si l'on veut retrouver la clarté correspondant à 10mA moyen, il faut durant le court instant d'activation multiplier l'intensité fournie par huit. Chaque LED va se voir traversée par un courant de pointe de 80mA et les opérateurs en sortie des 74HC573 devront commuter des courants de 5120mA soit plus de cinq ampères. Autant dire que l'iceberg va fondre et que la fête ne va pas durer longtemps. Qui va casser en premier, une LED ou le multiplexeur ?

Harf harf harf, faut prévenir immédiatement le ministre des cubes 3D ...



3) Le schéma électronique initial.

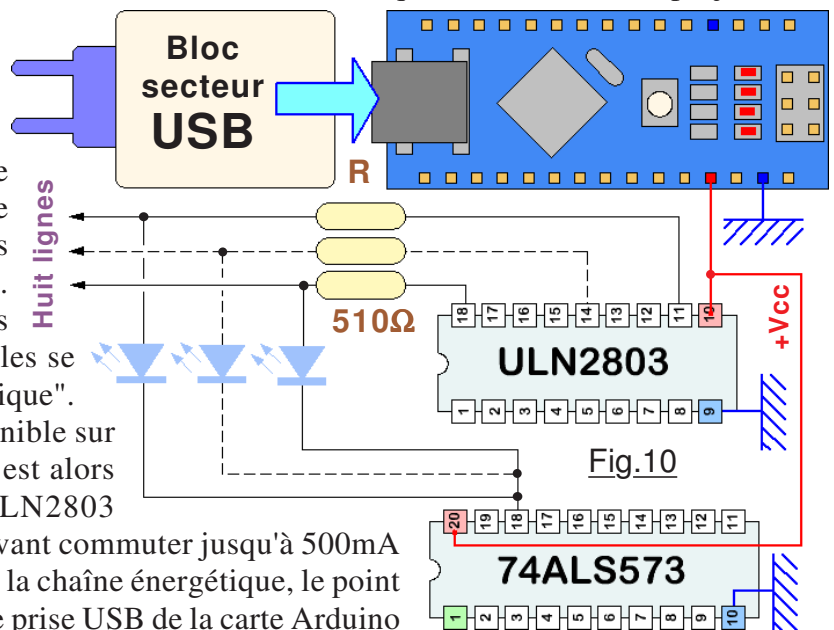
Possédant l'intégralité des préceptes qui conduisent à l'organisation d'un ensemble multiplexé, nous pouvons facilement décortiquer l'agencement du circuit électronique élaboré pour le KIT commercial qui nous sert de base à ce petit projet ludique, et analyser au passage les compromis inévitables pour garantir la fiabilité du dispositif. En particulier comment est résolu le problème des courants impulsifs qui ne doivent jamais dépasser ceux indiqués par les fabricants des composants en silicium.

➤ **Le maillon faible de la chaîne alimentaire.**

Fiable suppose "résister sans problème" dans le pire des cas, qui dans ce projet correspond à un cube entièrement éclairé. Et encore, le pire du pire correspond aux **512 LEDs allumées simultanément et en statique**. Cette configuration est parfaitement possible et se produit régulièrement quand sur un RESET du microcontrôleur l'intégralité des sorties des neuf multiplexeurs est constituée d'états logiques "1". Du reste, cette configuration fait partie intégrante des tests effectués initialement pour valider le bon fonctionnement de l'ensemble. Comment respecter toutes les contraintes électroniques ?

Il faut et il suffit de ne pas dépasser l'intensité acceptable par le composant "le plus faible".

Cette assertion impose d'examiner chaque maillon de la chaîne électronique de l'ensemble du projet. Bien que ne représentant qu'un seul multiplexeur de type 74ALS573 pilotant les huit colonnes verticales le long de l'axe X, le schéma élémentaire de la Fig.10 est assez représentatif de l'ensemble. Le multiplexeur vertical sur le schéma d'origine est constitué par l'un des ports binaires disponibles sur le microcontrôleur 80C52. On se doute que ces sorties ne sont pas capables de débiter un courant notable, elles se contentent de fournir un état électrique "logique". Quand cet état passe à "1" la tension disponible sur la sortie binaire ressemble à du +5Vcc. Il est alors présenté sur l'octuple amplificateur ULN2803 constitué de huit "Darlington" chacun pouvant commuter jusqu'à 500mA et à des fréquences très élevées. Dans toute la chaîne énergétique, le point le plus faible est constitué par la toute petite prise USB de la carte Arduino NANO. Car, n'oublions pas que durant le développement du programme d'exploitation, pour réduire l'encombrement sur le bureau de l'ordinateur c'est cette dernière qui alimente l'intégralité du cube. C'est donc le cube dans son ensemble qui ne devra pas consommer plus de 500mA.



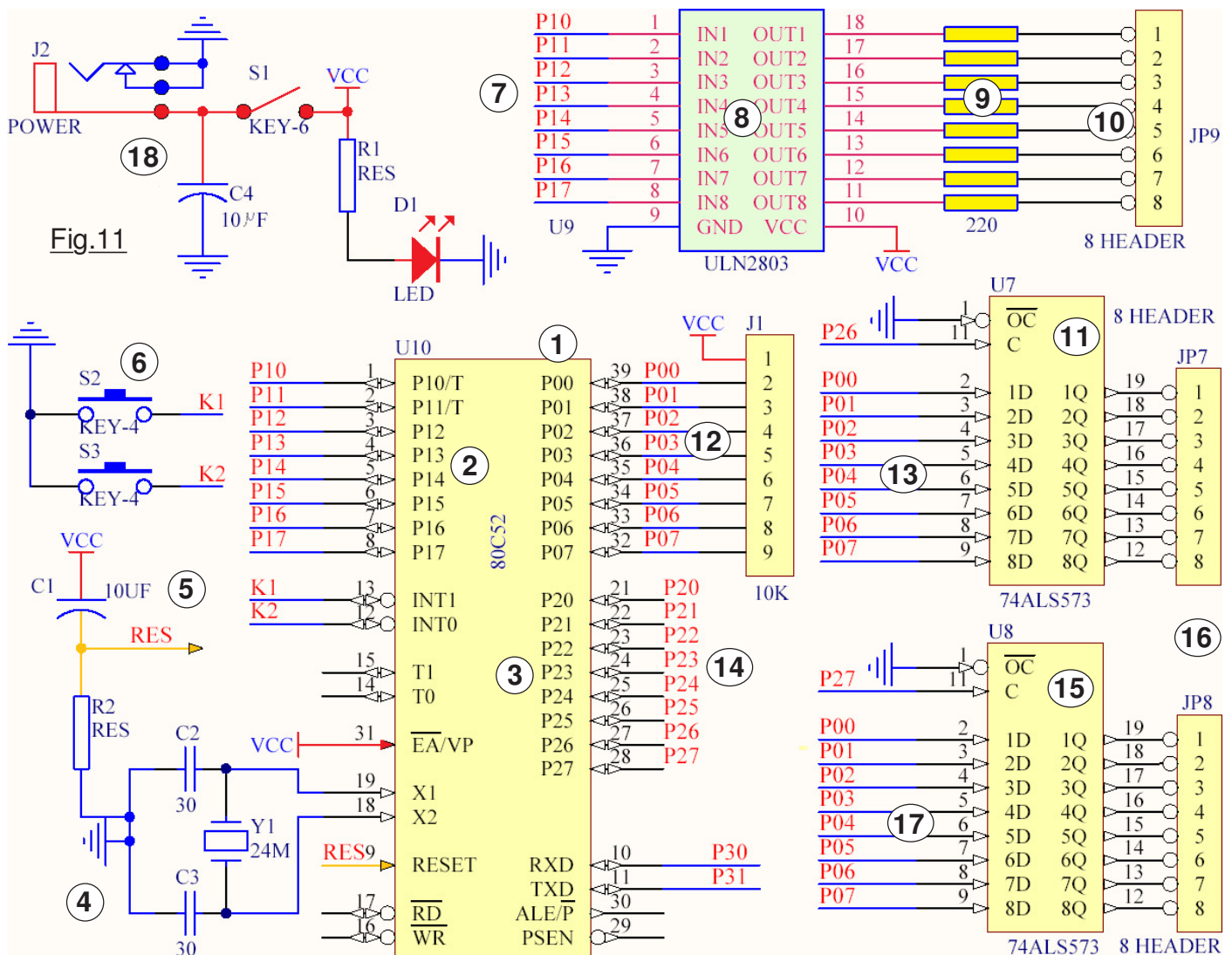
➤ **Compromission nocturne.**

Consultant le schéma d'origine trouvé sur Internet, (Voir la Fig.11) les résistances de limitation de courant en sortie de l'ULN2803 sont réputées faire 220Ω. Dans la pratique, celles fournies dans le KIT font 510Ω. Plusieurs raisons doivent conduire à ce choix, en particulier les caractéristiques propres aux diodes électroluminescentes de 3mm de diamètre. La plus grande intensité qui serait consommée sur l'une des sorties darlington serait de $5 / 510 = 10\text{mA}$ si elles étaient réunies directement à GND via R. Les huit lignes étant sollicitées, toutes les LEDs allumées, l'ensemble devrait théoriquement exiger 80mA en mode statique. Dans la pratique, la consommation est inférieure car la sortie des résistances de 510Ω n'est pas directement branchée à GND et les LEDs engendrent une chute de potentiel à leurs bornes. Du coup, l'intégralité du cube ne consomme au maximum qu'environ 60mA la tension fournie et mesurée étant de 5,13Vcc. Le test a été effectué lorsque c'est la carte du KIT qui alimentait Arduino, ceci étant précisé, dans l'autre sens les courants consommés sont pratiquement identiques. (Très légèrement plus faibles car la carte NANO délivre une tension régulée d'exactly +5Vcc.) On se doute que le courant total consommé n'alimente pas que le volume lumineux, mais également les circuits intégrés dont le microcontrôleur sans compter quelques diodes électroluminescentes dispersées sur le circuit imprimé du KIT et sur la carte Arduino NANO. Quand le volume du cube est exploité pour construire des tableaux artistiques, brusquement on doit balayer en hauteur, et l'intensité dégringole puisque l'on passe d'une technique statique à du multiplexage. Du coup, en exploitation "standard" l'intensité moyenne

devient considérablement plus faible. Comme dirait Lavoisier, il n'y a pas de miracle. Qui dit consommation réduite implique aussi luminosité réduite. Ce cube lumineux n'a pas été conçu pour rivaliser avec un soleil caniculaire. C'est dans une pénombre relative qu'il sera correctement admiré, tout particulièrement loin des sources lumineuses les plus importantes de la pièce à vivre dans laquelle il est exposé. N'oublions pas qu'à Noël, on laisse le sapin allumé quand on regarde le téléviseur, la pièce étant généralement sombre pour agrémenter le confort visuel. Vous constaterez dans ces conditions que le cube 3D ne se fait pas oublier, loin s'en faut ! Un éclairage suffisant des LEDs qui le compose résulte de leur rendement étonnant, issu des progrès foudroyants qui accompagnent ces dernières années l'industrie de l'optoélectronique.

➤ Étude du schéma électronique adopté sur le KIT de base.

Constitués de quelques extraits du schéma d'origine, celui proposé en Fig.11 contient les diverses zones dont nous pouvons avoir besoin pour comprendre les branchements que nous allons effectuer pour remplacer le 80C52 par notre petite carte Arduino NANO. Le schéma initial présente un circuit assurant une liaison RS232 sur TXD et RXD qui n'est plus d'actualité et non présent sur le KIT. Sont également représentés les huit multiplexeurs 74LS573 ce qui n'apporte rien à la compréhension, raison pour laquelle sur la Fig.11 seuls sont présents U7 en **11** et U8 en **15**. Enfin j'ai changé quelques couleurs,



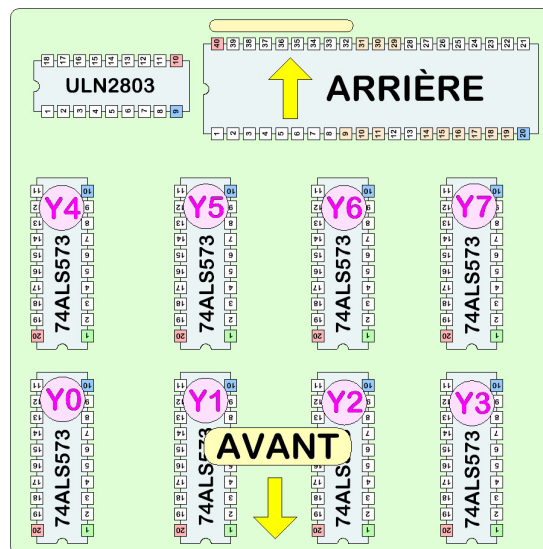
car la masse en rouge et le +5Vcc en bleu ne me séduisait pas beaucoup. Suite à ces remarques, passons à l'examen du montage initial. Le microcontrôleur 80C52 possède trois ports d'E/S binaires. Le premier P0 en **1** pilote en parallèle les huit multiplexeurs, par exemple les entrées **13** pour **11** et **17** pour **15**. Les cathodes des colonnes de LED sont branchées sur les connecteurs tels que **16**. Notez au passage que les sorties de P0 sont forcées à l'état logique "1" par l'octuple résistance de **10kΩ** montrée en **12**. Chaque multiplexeur 74ALS573 est validé individuellement sur sa broche **11** en C par l'une des sorties **14** du port P2 repéré **3** sur le schéma. C'est le port P1 du microcontrôleur 80C52 situé en **2** qui sert d'octuple mémoire au multiplexage. Ses broches initialisées en sorties fournissent les signaux aux "muscleurs de courant" **8** par les entrées repérées **7**. Par l'entremise du connecteur **10** les huit circuits

Darlington intégrés à U9 alimentent les anodes des LEDs à travers les résistances de limitation de courant de 510Ω visibles en 9. En 4 se trouvent les composants de l'horloge à quartz qui cadence le 80C52, avec en 5 l'agencement du RESET automatique à la mise sous tension. Ce circuit est totalement ignoré sur notre prototype. Sur le KIT, les boutons poussoir repérés 6 n'étaient pas fournis, car le programme inclus à U10 ne les utilise pas. Pour ma part, je les ai installés, car ils ouvrent des perspectives avantageuses pour l'exploitation. Enfin en 18 se trouve la prise Jack d'alimentation avec un condensateur de 10μF en découplage, et un interrupteur Marche/Arrêt qui dans cette application restera constamment activé, sauf épisodiquement en opérations de maintenance. Une LED rouge atteste la présence de l'énergie.

➤ Le circuit imprimé d'origine.

Fig.12

Provisoire si le programme animant le 80C52 vous suffit, le KIT tel qu'il se présente constitue la base sur laquelle viendra éventuellement se greffer notre complément intégrant une carte Arduino NANO. Pour mémoire, la Fig.12 propose un dessin qui situe la dispersion sur la carte des circuits intégrés qui constituent ce produit commercial. En particulier en rose est précisé l'ordre des divers multiplexeurs par rapport aux axes théoriques du cube représentés sur la Fig.4 de la fiche technique *Répartition géométrique des composants* qui accompagne ce didacticiel. Contrairement à la grande majorité de mes autres projets ludiques publiés sur la toile, c'est l'un des rares qui n'est pas accompagné d'un manuel d'utilisation et d'une notice technique pour la maintenance. Comme il y a vraiment peu de possibilités en exploitation, une seule fiche au format A5 est largement suffisante pour résumer les options possibles. Cette dernière est titrée *Comportement du programme* coté RECTO et *Les trois modes de visualisation* en face opposée. Les diverses fiches (*Exploitation, assemblage et maintenance.*) qui accompagnent ce tutoriel sont regroupées dans *Fiches techniques du cube 3D.pdf* dont la première page n'est pas à imprimer. Cette dernière donne quelques précisions relatives au contenu, et des conseils pas forcément inutiles. Notez que la Fig.12 est extraite de ce document. Avant de poursuivre notre cheminement vers la version "Arduino", en première étape nous allons réaliser le produit commercial. Ensuite, quand ce dernier fonctionnera correctement, il vous sera toujours possible d'y ajouter comme sur le prototype "la couche personnelle". Comme je ne savais pas au début qu'il y aurait cette facette secondaire dans le projet, initialement c'est le KIT seul qui était envisagé. Pour ajouter la gestion personnelle, il a suffi d'ajouter sur le coffret "un étage" supplémentaire. (*Qui en réalité est situé dessous !*) Nous allons donc passer à la réalisation initiale, sachant qu'il peut y avoir une suite qui imposera de surélever le coffret à sa partie inférieure. Naturellement, si le complément Arduino est envisagé dès ce stade, autant réaliser le coffret directement à la bonne hauteur, facilitant ainsi sa réalisation. (*Encore que la technique utilisant du polystyrène choc facilite grandement la "pièce rapportée".*)



4) Assemblage du KIT initial.

Souder le circuit imprimé confine à une formalité. La sérigraphie est très bien faite, elle équivaut pratiquement à une notice d'assemblage. Il y a peu de composants discrets, et le sens des supports de circuits intégrés est bien repéré. Attention à ne pas vous tromper dans l'orientation de l'octuple résistance de 10kΩ. L'une des broches d'extrémité est reliée en interne au commun. Elle doit impérativement être soudée au +Vcc. Toujours pour les composants dont la bonne orientation est incontournable, on trouve les deux condensateurs de 10μF, la LED rouge, l'interrupteur d'alimentation. Si vous complétez le produit par la version Arduino, il faudra également prendre garde à orienter correctement les deux boutons poussoir du clavier. Dans le répertoire <Documentation> se trouvent les caractéristiques des circuits intégrés assurant le multiplexage. Schematic.pdf est le fichier que j'ai glané en ligne qui correspond à une version très légèrement différente que celle du KIT. (Document sur lequel a été extraite la Fig.11 donnée en page 10.) Enfin, avec le fichier Steps of soldering.pdf vous disposerez du seul document qui accompagne le produit commercial mentionné dans ces lignes. Globalement il est très bien fait et permettrait de réaliser correctement le cube ... sauf qu'il ne permet absolument pas, du moins c'est ce qui ressort de mon expérience, de savoir comment orienter les diodes électroluminescentes. Il m'a fallu étudier

intégralement le schéma pour déduire cette information pourtant primordiale. Peu importe, le problème est résolu, (*Donc ce n'est plus un problème !*) et l'on va avec méthode concrétiser notre objet décoratif. Revenons au soudage du circuit imprimé qui rassemble l'électronique du KIT. Ce dernier étant de type "double face", il y aura à souder des deux cotés. Il importe donc de choisir un ordre d'assemblage qui facilite au maximum l'opération. D'une façon générale, je commence toujours par les éléments les moins hauts pour finir par ceux qui dominent. Bien faire attention au fait qu'il y a des soudures à faire coté "composants" à l'intérieur de la surface des supports de circuits intégrés. Il faut commencer par assembler les picots qui recevront les LEDs avant les supports DIL. Quand vous aurez achevé les étapes de **Step 1** à **Step 4** le circuit imprimé principal est achevé. (*Il reste à créer le cube de LEDs, surtout attendez d'avoir consulté les chapitres qui suivent.*) À ce stade il est déjà possible de procéder aux premières vérifications. Aucun circuit intégré n'est placé sur son support. On alimente en +5Vcc. On peut le faire par la prise Jack du circuit imprimé. J'ai également placé un petit connecteur HE14 coudé comme le circuit imprimé le permet à l'endroit repéré **CON4**. Outre la possibilité d'y trouver **GND** et **+Vcc**, j'y ai relié K1 et K2 ce qui permet en cours de développement du programme de déporter les deux boutons poussoir bien à portée de la main sans avoir à retourner le cube pour accéder à ceux situés en arrière sur le circuit imprimé. On clique sur l'inverseur d'alimentation. La LED rouge doit s'allumer. Puis on vérifie que la tension est présente sur les bonnes "broches" des supports des circuits intégrés. On place alors le 80C52 sur son support ... **Et dans le sens convenable !** Comme il fonctionne en aveugle sans prendre en compte son environnement, normalement il doit s'engager dans sa boucle de programme, et sur les sorties on doit observer des signaux binaires découpés à des cadences variables. Puis on coupe l'alimentation, on dispose les neuf autres circuits intégrés. Mise sous tension. Le circuit imprimé étant placé composants sur le dessous, en plaçant une LED quelconque, convenablement orientée, entre deux picots servant pour les branchements du cube, elle doit s'allumer et s'éteindre à une cadence qui semble aléatoire. Ouf, le circuit imprimé principal est vivant, tout va bien. Il ne reste plus qu'à assembler le cube ... et c'est ici que **patience**, méthode **et longueur de temps font plus que force ni que rage !**

➤ Les tranches de LED.

Concrétiser matériellement le cube de LEDs, (*Concrétiser et matériellement constitue un pléonasme, exactement comme l'utilisation de l'expression vieille baderne !*) impose trois qualités indispensables, incontournables, impératives pour ne pas dire totalement obligatoires :

- De la méthode,
 - De la méthode,
 - De la méthode.
- } Ces trois vérités bibliques taillées dans le marbre font précisément l'objet des chapitres qui suivent.

Maintenant que le circuit imprimé principal est achevé et que le moral est au beau fixe, nous connaissons exactement la distance qui devra séparer les diodes électroluminescentes. C'est l'écartement des picots de

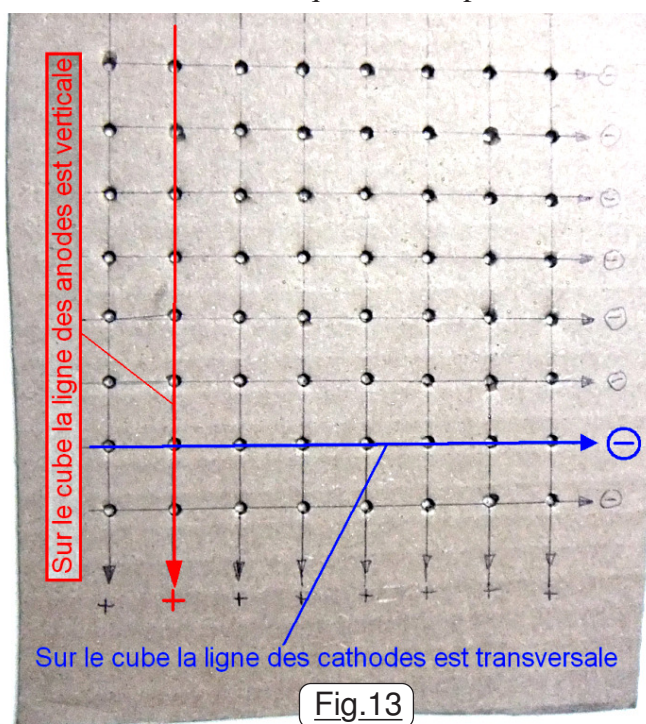


Fig.13

branchement qui détermine en largeur et en profondeur les dimensions de cube. Les écartements choisis par le concepteur du circuit imprimé autorisent l'adoption de LEDs quelconques, et tout particulièrement des diodes dont l'encapsulation est conditionnée en diamètre 5mm. Je vous invite fortement à lire en fin de ce tutoriel le chapitre **Les "loupés"**. Si c'était à refaire, j'utiliserais des composants en boîtier diffusants et non cristal. La couleur bleue est très belle, reste à choisir des éléments qui présentent un bon rendement. Enfin, avant de plier à angle droit les broches de ces DELs je commencerais par peindre en noir mat la face "arrière" plane située coté broches. **À vous de voir, il est encore temps pour faire un choix judicieux ...**

Première étape de la réalisation des "tranches" lumineuses, récupérer un morceau de carton rigide cellulaire. Puis, comme montré sur la Fig.13 tracer un quadrillage dont la trame correspond à l'écartement transversal des picots de branchement qui

sont soudés sur le circuit imprimé. Verticalement on adoptera un écartement identique pour des raisons esthétiques évidentes. Puis, à chaque croisement de ligne percer un trou relativement fin avec un poinçon quelconque. Notez au passage que le dossier <Galerie d'images> contient beaucoup de photographies commentées pour accompagner ce tutoriel. La Fig.13 est extraite de cette mine de renseignements. Les deux photographies commentées [Image01.JPG](#) et [Image02.JPG](#) montrent l'outil de positionnement en carton sous deux orientations différentes. (Sur la première la LED dans l'encadré donne l'échelle, cette zone est extraite de la même photographie et déplacée pour des raisons de taille des fichiers.)

Fonction du choix que vous aurez effectué, à savoir conserver les LED de diamètre 3mm du KIT ou adoption de LED 5mm, agrandir les trous le juste ce qu'il faut pour que les composants tiennent en position verticale "à frottement doux" les broches de liaison vers le haut. Quand on assemble une face carrée, les LEDs sont orientées coté "avant du cube" vers le bas. Les lignes sur le morceau de carton sont complétées par des signes indiquant la polarité des broches associées. Ainsi lors des contrôles on sait immédiatement où établir le contact et avec quelles pointes de touche du système de vérification.

NOTE très très importante : En particulier, dans ce projet, plus on avance vers la finalisation plus les composants s'imbriquent les uns dans les autres. Quand le cube est entièrement matérialisé, imaginez que vous constatiez que la LED qui est au centre est défectueuse. Peu importe qu'elle soit grillée ou soudée à l'envers, elle est totalement inaccessible. VLOUFFffff, le truc passe par la fenêtre, vous sautez à pieds joints sur le circuit imprimé, et vous êtes définitivement écœuré par l'électronique de loisir. Ainssi, remplacer une LED au milieu d'une ligne que vous être en train d'assembler est peu commode. Une vers le centre de la face devient un peu galère. Au centre du cube IMPOSSIBLE. Pas question de souder une LED sur l'ensemble si la dernière (Et toutes les autres.) n'a pas été vérifiée. **Chaque pas de l'assemblage sera accompagné d'une vérification et pour CHAQUE LED !**

Sachez que lorsque les huit faces de la grille de LEDs seront assemblées individuellement, il faudra les enfiler sur les broches de la matrice cubique. Huit tiges de LED à placer exactement en face de l'orifice de la broche pour que l'insertion soit faisable. C'est un tantinet galère. **Aussi il importe d'avoir exactement le bon écartement sur le tracé de la grille sur la plaque en carton.** C'est la première vérification à effectuer. Quand vous êtes certain que le dessin est parfait ... vérifiez une dernière fois !

➤ Le pliage des broches des LEDs.

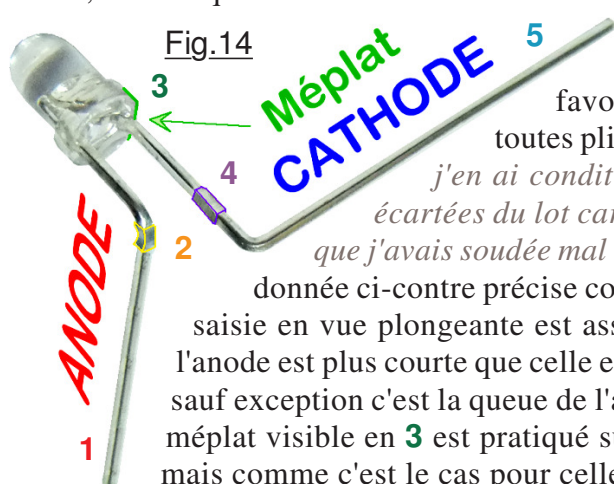
Étape préalable indispensable, avant d'amener une LED sur le carton de soudage, ses deux broches doivent être pliées à angle droit de façon à former un "triplet trirectangle". (Voir la Fig.14) De la rigueur avec laquelle vous allez vous acquitter de ce préambule dépend directement la facilité et la qualité des opérations qui suivront. Aussi, adopter une procédure rigoureuse est indispensable. Au début, et manquant d'expérience, je conditionnais huit éléments. Puis avec le "frère à souder" ces derniers étaient assemblés sur le carton. Ce n'est pas du tout la bonne approche et ce pour deux raisons :

- Comme il faut plusieurs heures pour les opérations de pliage, c'est autant de fonctionnement inutile du fer à souder qui durant cette phase chauffe pour rien. (Faut minimiser le réchauffement climatique !)
- Huit manipulations entrecoupées par les opérations de soudure délicates engendrent de l'oubli. À chaque fois il faut se remémorer les critères de pliage avec perte de temps et risque d'erreur.

Aussi, et bien que ce soit totalement rébarbatif et indigeste, je vous recommande de **plier TOUTES les**

LEDs avant d'envisager les opérations de soudage. Ce travail routinier indigeste conduit à un automatisme très favorable à la réussite. En procédant ainsi les 255 LEDs ont été toutes pliées parfaitement et sans erreur de polarité. (Personnellement j'en ai conditionné dix de plus pour parer les incidents. Quatre ont été écartées du lot car de luminosité estimée insuffisante à faible courant. Et une que j'avais soudée mal orientée. J'ai préféré la remplacer par une neuve.) La Fig.14

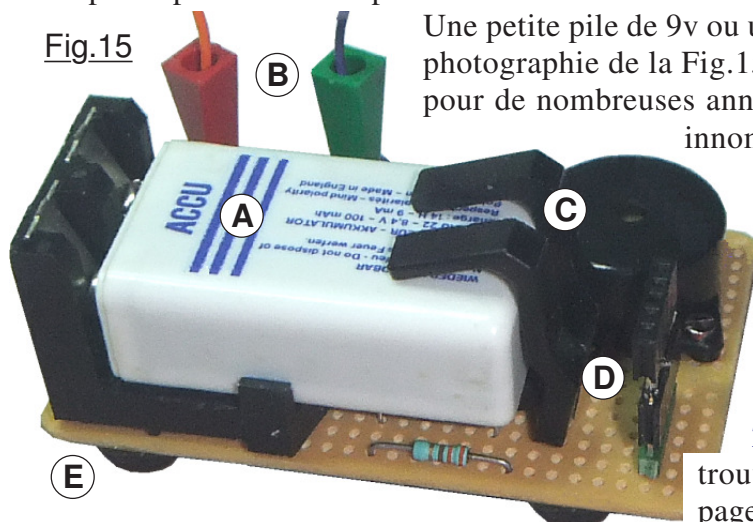
donnée ci-contre précise comment nous devons réaliser les pliages. Cette photographie saisie en vue plongeante est assez trompeuse, car elle donne à penser que la broche **1** de l'anode est plus courte que celle en **5** de la cathode. C'est l'inverse, pour repérer les brochages, sauf exception c'est la queue de l'anode qui est la plus longue. Pour compléter le repérage, un méplat visible en **3** est pratiqué sur la collerette de l'encapsulation. Ce n'est pas obligatoire, mais comme c'est le cas pour celles fournies dans le KIT, les deux



broches présentent en **2** et **4** une variation de section, une sorte d'évasement rectangulaire. Quand les deux broches sont tenues horizontalement, il faut plier **celle de l'anode 1 vers le bas AVANT l'excroissance 2**. En revanche, La broche **5 de la cathode doit se voir pliée à angle droit latéralement APRÈS 4 à environ 1,5mm à 2mm de l'excroissance**. (La LED est ici vue depuis sa face arrière.) Les broches **1** et **5** formeront un trièdre trirectangle dont on doit respecter le mieux possible les 90°. Si cette géométrie est respectée, en "profondeur" la cathode est décalée par rapport à l'anode. C'est indispensable car les grilles de diodes forment un treillis croisé qui ne doit pas engendrer de courts circuits entre anodes et cathodes. Dans **<Galerie d'images>** les photographies [Image03.JPG](#), [Image04.JPG](#) et [Image05.JPG](#) saisies sous des angles différents commentent les diverses contraintes géométriques à respecter. Avant de se voir considérée comme valide, la diode une fois correctement pliée est testée, car une inversion de polarité est potentiellement risquée. Pour ce faire, vous ouvrez le tiroir et placez sans tarder votre sonnette sur le plan de travail, car c'est une évidence, il y a belle heurette que vous possédez cet outil indispensable.

➤ **Drinnnnngggg ... la sonnette !**

Ainsi nommée car autrefois, il y a très longtemps quand c'est Grand-père qui bricolait, on utilisait réellement une sonnette secteur pour vérifier la continuité d'une ligne électrique. Autant dire que l'électronique a balayé ces antiquités au profit d'appareils portatifs infiniment plus opérationnels. Si vous ne disposez pas d'un tel dispositif ... **STOPPEZ TOUT POUR EN FAIRE UN IMMÉDIATEMENT !**



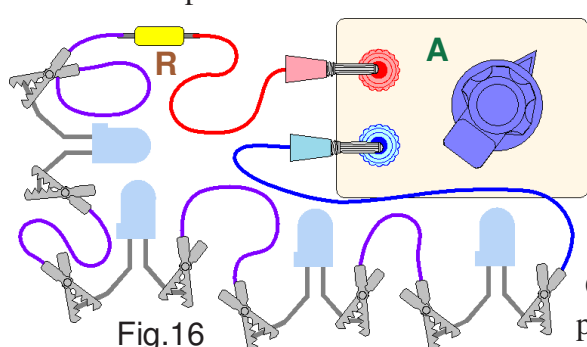
Une petite pile de 9v ou un petit accumulateur rechargeable comme sur la photographie de la Fig.15 associée à quelques bricoles et vous voilà paré pour de nombreuses années. L'exemplaire présenté ici n'est que l'un des innombrables "trucs" que j'ai agencé au cours de ma

vie d'électronicien ludique. (Hé oui, le Grand-père cité ci-avant ... c'est MÉZIGUE.) Un Buzzer actif comme celui prévu pour la version Arduino du cube 3D confirme de façon sonore la continuité d'une liaison testée. Comme il est hors de question d'encombrer ce chapitre avec une description détaillée, une fiche nommée **TESTEUR DE CONTINUITÉ ET DE LED** se

trouve dans [Fiches techniques du cube 3D.pdf](#) en page 11. En **A**, pour des raisons déontologiques la marque de l'accumulateur a été censurée. En **B** les fiches bananes de 2mm sont insérées dans des douilles de couleur pour ne pas se tromper de polarité quand l'opération en cours l'exige. En **C** le Buzzer actif. En **D** on trouve le connecteur HE14 mâle de couleur verte. Sur ce dernier un adaptateur change son genre en femelle bien plus apte à tester des LEDs par exemple. Enfin, pas très visible en **E** le dessous de la petite plaque de circuit imprimé qui sert d'ossature est munie de petits pieds en caoutchouc auto-adhésifs.

➤ **Une idée lumineuse !**

Vantardise éhontée, ce titre accrocheur a pour but d'attirer votre attention. En effet, nous avons vu que les diodes électroluminescentes fournies dans le produit commercial sont des éléments qui présentent un bon rendement. Cette caractéristique autorise les concepteurs à les utiliser "sous alimentés" tout en conservant un éclairage encore raisonnable. **ATTENTION : Autant les composants soumis à des courants nominaux présentent des luminosités comparables, autant employés à faible courant leurs caractéristiques peuvent diverger**. Aussi, il IMPORTE de soumettre tous les composants qui seront sélectionnés pour le cube à des tests d'éclairage de façon à éliminer ceux qui visiblement sont moins



lumineux en sous-alimentation. La Fig.16 représente la version "Grand-père" avec pinces crocodiles et fiches bananes. Vous utiliserez probablement des plaques d'expérimentation pour Arduino. Peu importe. La technique consiste à chaîner plusieurs LEDs de façon à voir simultanément leurs luminosités. Plus vous en placerez en série, plus rapide sera la sélection des éléments élus pour le cube. Comme chaque LED engendre une chute de tension de plusieurs volts, c'est la tension maximale disponible

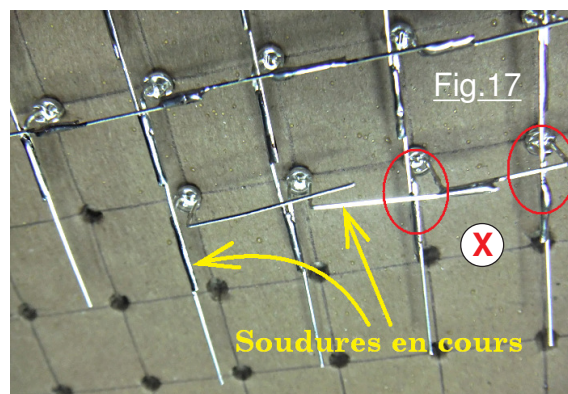
sur l'alimentation secteur **A** qui en limitera le nombre. Si vous ne disposez pas d'une alimentation de laboratoire, une ou deux piles de 9Vcc mises en série feront parfaitement l'affaire. Pour la résistance de limitation **R** il faut choisir une valeur qui conduira à un éclairage visible des LEDs, mais relativement faible. Dans ces conditions, sur le lot de composants qui étaient fournis avec le KIT, j'en ai écarté cinq qui manifestement éclairaient nettement plus faiblement que les autres.

Toutefois, il ne faut pas espérer de miracle. Le cube 3D réalisé consomme très peu, et chaque LED est fortement sous-alimentée. Dans ces conditions elle fonctionnera pendant des siècles certes, avec des différences de luminosité discernables quand tous les éléments seront sollicités. Ceci étant précisé, ne dramatisons pas. Les différences restent tout à fait raisonnables et ne nuisent pas à l'esthétique globale.

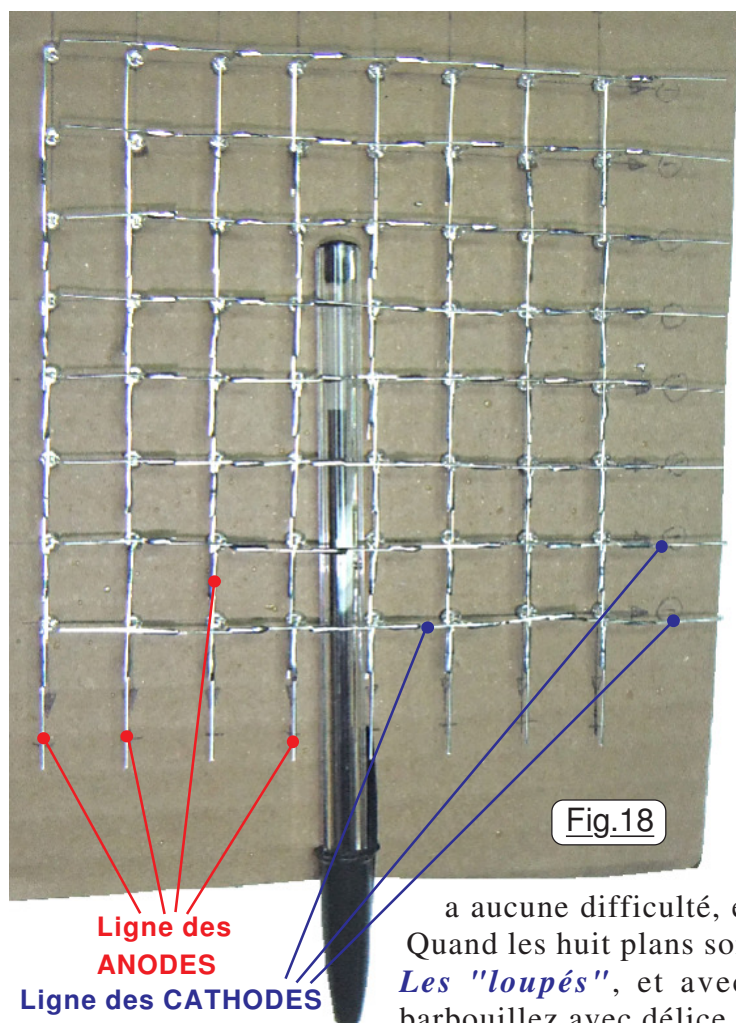
➤ Soudage d'une tranche de cube.

Lorsque toutes les LEDs ont été correctement pliées et vérifiées, nous pouvons commencer le soudage. Insérant les éléments les uns après les autres, pour la première rangée il n'y a qu'une seule soudure à effectuer sur chaque composant ... suivie il va sans dire par son contrôle immédiat avec la sonnette.

[Image06.JPG](#) montre la façon de placer les composants sur le carton d'assemblage, et surtout [Image07.JPG](#) dévoile cette étape initiale entièrement achevée. On continue ligne transversale par ligne transversale, sauf que maintenant, comme le montre [Image08.JPG](#), chaque diode mise en place impose deux soudures. Observez également [Image09.JPG](#) et [Image10.JPG](#) qui présentent en saisie rapprochée les grilles lumineuses. On comprend mieux pourquoi les deux broches étaient pliées avant et après l'excroissance. Ainsi les hauteurs des lignes transversales et "en hauteur" sont décalées assurant la séparation électrique. Vous pouvez vérifier sur



la Fig.17 en **X** qu'un écart d'environ 2mm sépare les "lignes verticales" des "lignes horizontales". Comme les broches pliées ne s'alignent pas forcément avec une orientation absolue, on peut facilement comme



montré sur la photographie d'[Image11.JPG](#) les faire légèrement fléchir avec une cale quelconque pour faciliter la soudure. Cent vingt huit soudures plus tard, on aboutit au résultat de la Fig.18 et par un balayage rapide des pointes de touche sur les broches du bas et de droite on vérifie une dernière fois l'**éclairage correct de chaque LED**. On se doute que le maillage des soixante quatre diodes est fragile car il manque de rigidité. Comme toutes sont légèrement serrées dans l'outil en carton, il faut les extraire avec prudence en forçant sur toute la hauteur avec un ustensile quelconque. On déboîte à gauche, au centre puis à droite et c'est la délivrance. Sur [Image12.JPG](#) on récolte avec fierté le fruit de notre patience. Si l'outil en carton a été tracé et percé avec précision, la géométrie de ce grillage optoélectronique est pratiquement parfaite. La grille est dirigée vers nous, comme ce sera le cas quand elle sera insérée dans le cube. Comme à gauche les cathodes sont orientées vers nous, on peut en déduire que ce sera le plan lumineux situé au fond du volume 3D. Il reste encore sept autres grilles à confectionner, il n'y

a aucune difficulté, et avec un peu de patience on finit par aboutir. Quand les huit plans sont achevés, vous lisez attentivement le chapitre *Les "loupés"*, et avec une peinture à maquettes **noir mat** vous barbouillez avec délicie l'arrière des LEDs ...

5) Le coffret servant de socle stable.

Asssembler toutes les "tranches" pour confectionner le cube lumineux conduit au début à un ensemble d'une grande fragilité. Aussi, chronologiquement j'ai réalisé le socle avant de passer "à la troisième dimension". Je vais vous détailler la solution adoptée sur le prototype et surtout la critiquer de façon à ce que vous puissiez envisager une version qui vous simplifierait l'assemblage, car pour ma réalisation, dire que ce fut une galère est presque sous-estimé. Avant de décrire en détail le socle tel qu'il se présente, abordons en préambule mes techniques actuelles pour réaliser de beaux petits coffrets à mes projets.

Bien qu'il soit commode d'utiliser des coffrets du commerce, personnellement je préfère de loin réaliser entièrement mes propres boîtiers. L'avantage incontestable réside dans le fait que l'on n'est jamais aussi bien servi que par soi-même. Traduisez : On peut faire exactement ce qui correspond à nos désirs, et en particulier optimiser les formes et les dimensions. Ma technique étant aisément abordable par tout un chacun, autant en faire profiter la collectivité. Le matériau miracle pour confectionner mes réalisations est connu sous le vocable de **polystyrène choc**.

Nombreux sont les didacticiels déjà publiés sous mon pseudonyme "Nulentout" qui, décrivant des appareils électroniques, informatiques, astronomiques, détaillent mes méthodes d'usinage. Il serait commode d'indiquer ici les liens et faire référence aux chapitres concernés. Pour vous éviter des allées et venues sur la toile, j'ai pensé plus convivial de copier honteusement les chapitres relatifs au travail de ce merveilleux matériau qu'est le polystyrène choc.

Commencez par lire les informations données dans l'encadré de la page 15 et vous aurez en main la marche à suivre pour créer toutes les merveilles issues de votre imagination.



ATTENTION : DANGER !



Libre à la vente n'implique pas "sans danger". Dans les drogueries, on peut aussi acheter de l'acide pour nettoyer du ciment par exemple. Bien que libre en vente il faut prendre des précautions pour les mains, le visage ou les yeux. **Pour le diluant cellulosique c'est analogue. Les vapeurs qui se dégagent de la réaction chimique sont nocives. Donc :**



- **Lisez bien les informations portées sur la bouteille.**



- **Travaillez dans un local bien aéré quand vous procédez au soudage.**

➤ Technique générale d'assemblage.

Façonner avec soin et précision les divers éléments à réunir pour construire l'ensemble sera un garant de qualité. Prendre son temps, vérifier les dimensions et la géométrie, ne pas se précipiter. Percer et réaliser si possible l'intégralité des trous et des orifices sur les éléments individuels, c'est tellement plus commode que lorsque le coffret est entièrement assemblé. D'une manière générale, la médiocrité d'une réalisation quelconque est directement impactée par son élément le plus faible. Ou si vous préférez, quand vous adoptez un niveau de qualité, tous les aspects du projet doivent présenter une valeur analogue. Bref, il faut tracer sur toutes les pièces à assembler la position précise des éléments qui viennent en contact. Pour ma part tous les repérages sont réalisés avec des feutres classiques du commerce. Quand les ensembles sont achevés, j'efface tous les traits avec de l'alcool ménager qui me sert également à nettoyer les pièces avant soudage ou avant peinture. Une petite équerre métallique me semble indispensable pour vérifier les perpendicularités un peu partout. Vous préparez tout le matériel dans un local bien aéré, sans oublier un support quelconque pour poser le pinceau qui sert à appliquer le diluant sur les zones à souder.

Consultant avec attention le dessin de l'ensemble, vous déterminez les tranches ou les surfaces à souder. Vous mouillez avec le diluant. Vous appliquez fortement la pièce sur les zones mouillées. La prise est immédiate, et si le local n'est pas trop chaud, vous disposez d'un temps largement suffisant pour assembler et positionner sept ou huit pièces. Puis, avant que l'ensemble ne soit solidifié, vous repassez tout en revue pour parfaire la géométrie : Les diverses cotes de positionnement, les orientations relatives, les équerrages ...

Toujours partir de dessins rigoureux réalisés à l'échelle unitaire pour façonner les divers éléments, après avoir tracé un **dessin d'ensemble précis pour s'assurer que l'intégration des éléments et leur câblage ne sera pas une corvée** sans nom. Méfiez-vous, certaines pièces ne sont pas symétriques, placer à gauche celle de droite engendrera à coup sûr une difficulté. Ceci dit, pas de panique. Si vous constatez une erreur inacceptable alors que l'ensemble est solidifié depuis deux jours, et que les éléments sont soudés entre eux par la tranche, en forçant un peu avec un cutter rigide on arrive à désolidariser sans détruire. Beaucoup d'erreurs seront pardonnées ... heureusement, car souvent mes prototypes souffrent leurs lots de maladresses ...



Suite en page 16.

Mettre en œuvre du polystyrène choc.

Facile à se procurer par le biais du commerce en ligne, particulièrement agréable à usiner, percer, limer, ce matériau vendu en plaques de diverses épaisseurs et diverses tailles, est absolument parfait pour le bricoleur du dimanche. Utilisé depuis des années pour réaliser des coffrets électroniques, des petites mécaniques de robotique etc, je ne saurais plus m'en passer. Pour trouver un magasin de vente par correspondance, donnez les deux mots clef **Polystyrène** et **Choc** à un quelconque moteur de recherche et vous n'aurez plus qu'à mettre dans le panier une ou deux plaques de votre choix. Pour ma part j'ai utilisé plusieurs épaisseurs, le meilleur compromis semble 3mm que j'approvisionne en grandes plaques blanches, car j'en fais une consommation abusive.

Outils indispensables : On ne peut vraiment pas se contenter d'un simple cutter, il faut impérativement investir dans du luxe. La débâche de matériel à acquérir est présentée sur la Fig.19

Fig.19



qui correspond au minimum vital. En **1** la petite chignole à manivelle qui m'a été donnée par mon amis Jules il y a plus de

quarante ans car je n'avais rien pour percer mes circuits imprimés. Elle n'avait déjà pas

le bouchon du manche prévu pour y loger des forêts tels que **2**. L'engrenage a pris du jeu, elle en a percé des milliers de trous cette fidèle compagne !

Il existe bien des petites perceuses électriques sur colonne, mais je n'ai jamais rien trouvé de plus précis que ma ringarde "tournicotine". En **3** des petites

limes de qualité, ou comme en **4** une lime pour réaffûter les

chaînes de tronçonneuses. Ajoutez deux ou trois feuilles de **papier verre**

pour poncer avec finesse et l'on a pratiquement tout ce qu'il faut. Avec ces ustensiles,

je découpe et peaufine mes pièces sans problème au dixième de millimètre. Je dispose d'un alésoir qui permet d'agrandir les trous de grand diamètre, mais avec une lime ronde ou semi-ronde on peut s'en passer. Ajoutez également une équerre et l'indispensable est réuni. Pour les forêts **2** il faut une série complète entre 1mm de diamètre et six millimètres, ou plus pour chanfreiner les trous.

Façonnage des pièces : Pour les découper, après avoir tracé avec précision le contour, on marque un peu en profondeur des deux cotés avec le cutter. En forçant entre les deux voisinages d'une rayure, les éléments se séparent. Petite amélioration de l'aspect visuel à la lime, puis *parfaire la rectitude et la perpendicularité éventuelle avec un autre coté* sur la feuille de papier verre et l'élément est paré pour son inclusion dans l'ensemble dont il va faire partie. La méthode présentée en Fig.20 pour ajourer des formes complexes peut sembler indigeste. Elle consiste à percer des trous de diamètre 3mm proches les uns des autres ou sécants pour fragiliser la zone centrale. Au cutter on coupe les tenons. Et quand "l'étoile" centrale se décroche, les limes ou le papier verre font le reste. C'est facile et très précis. Si des cotés sont long, quatre trous réunis permettent d'introduire une lame de scie. On coupe alors de façon rectiligne rapidement et sans galérer sur de fortes longueurs.

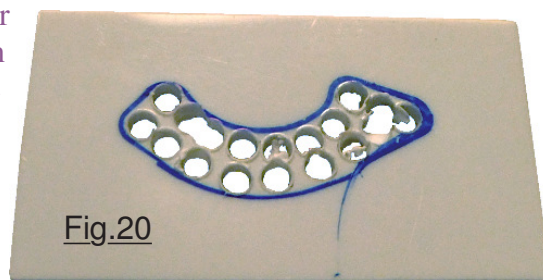


Fig.20

indigeste. Elle consiste à percer des trous de diamètre 3mm proches les uns des autres ou sécants pour fragiliser la zone centrale. Au cutter on coupe les tenons. Et quand "l'étoile" centrale se décroche, les limes ou le papier verre font le reste. C'est facile et très précis. Si des cotés sont long, quatre trous réunis permettent d'introduire une lame de scie. On coupe alors de façon rectiligne rapidement et sans galérer sur de fortes longueurs.

Souder les éléments entre eux : On trouve librement en droguerie ou dans les magasins de bricolage du **Diluant Cellulosique**. Avec un petit pinceau, vous en passez sur la tranche à souder, vous appliquez un peu fortement sur l'autre pièce et les deux se soudent. Ce n'est pas du collage, mais bien de la soudure autogène. On peut facilement en souder trois ou quatre, et avant que l'assemblage ne devienne résistant, les faire glisser pour les placer bien bord à bord ou corriger des équerrages. Pour aboutir à des éléments de forte épaisseur, on applique du diluant sur toute la surface de l'une des plaques et l'on presse sur l'autre. Un petit serrage à l'étau d'établi et l'on a définitivement une pièce unique de forte épaisseur. On peut ainsi, par exemple créer une semelle très rigide, empiler plusieurs couches. Quand le total est durci, on lime alors les cotés pour obtenir une pièce parfaite, avec arrondis, chanfreins etc.

➤ Trois outils indispensables.

Outre les habituelles ressources de bricolage qui permettent de percer, limer, visser, trois individus doivent absolument faire partie intégrante de vos outils. Ces "élus" sont à mon avis absolument incontournables pour un amateur qui consacrera une grande partie de ses loisirs à créer de ses mains une foule de petits dispositifs. Montré dans toute sa splendeur sur la Fig.21 le premier de ces nominés prend la forme d'un "tourne écrou" pour des boulons $\phi M3$. Il est absolument incontournable pour aller serrer un écrou sur un module placé tout au fond d'un coffret bien dans le coin. Aucune clef plate ni clef à pipe ne pourra manœuvrer l'élément rebelle. Hors, l'étude matérielle du prototype tient compte de la disponibilité d'un tel ustensile. STOP ! Avant de foncer au magasin attendez d'avoir la liste complète !



Fig.21

Les deux soldats qui suivent sont dédiés à la réalisation précise des trous circulaires. En premier on peut citer la queue de rat, c'est à dire une lime conique de petit diamètre. Présentée sur la Fig.22 la mienne est légèrement tordue, ce qui prouve que cette petite merveille de précision n'est pas du tout



Fig.22

adaptée pour forcer inconsidérément lors d'un usage pour lequel elle n'a jamais été prévue. Conçue pour usiner du métal, elle s'encrasse relativement vite quand on ajuste un orifice sur une pièce en polystyrène choc. Il faut régulièrement la carder. Le faible angle du cône en fait un outil de grande précision particulièrement précieux.

L'alésoir de la Fig.23 est probablement le plus couteux des trois compères. Quel qu'en soit le tarif annoncé sur l'étagère de la boutique, faites-en l'acquisition. C'est assurément l'allié totalement indispensable qui accompagnera vos heures de loisir durant toute votre vie de bricoleur invétéré. Franchement ça vaut le coup de cesser de fumer pendant deux mois pour économiser le financement de cette merveille. L'investissement consenti lors de l'achat sera très très très compensé par la précision obtenue avec cet alésoir quand vous chercherez à ouvrir des orifices circulaires avec des diamètres vraiment précis. Hors c'est indispensable pour réaliser les façades de nos coffrets électroniques. Les inverseurs, potentiomètres et autres LEDs doivent traverser les plaques "en sifflant". Quand aux boutons poussoir des claviers, un jeu minimal de passage sera le garant d'une esthétique soignée. L'angle d'ouverture très faible α de cet outil autorise sur la réalisation des trous, facilement des précisions de l'ordre du dixième de millimètre. Son faible diamètre d'amorçage autorise le travail à partir d'orifices aussi petits que des trous de 3mm de diamètre.

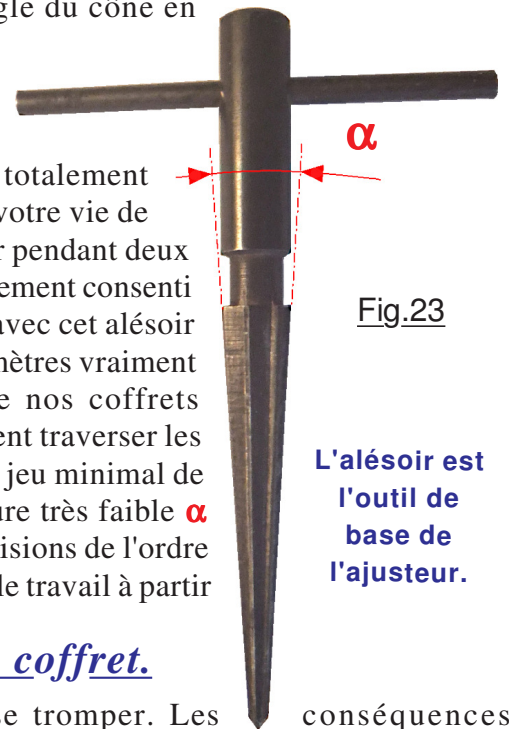


Fig.23

L'alésoir est l'outil de base de l'ajusteur.

6) Préparation des éléments, et assemblage du coffret.

Concevoir, c'est obligatoirement prendre le risque de se tromper. Les conséquences d'une erreur commise et détectée trop tard peuvent enliser définitivement un projet si la déconvenue s'avère trop brutale. Je n'insisterai jamais assez sur la **nécessité impérieuse de bien penser le projet avant de commencer à concrétiser**. Ne pas se précipiter sous l'influence insidieuse de l'enthousiasme, prendre le temps pour analyser tous les aspects importants et en particulier :

- Une belle boîte, esthétique et proportionnée dans ses formes.
- Encombrement optimisé pour des raisons évidentes de commodité d'usage.
- Disposer les prises, les douilles, les fusibles pour l'aspect visuel, sans toutefois oublier la qualité opérationnelle qui sera directement fonction de leur répartition sur la périphérie du coffret.
- **Pouvoir facilement intégrer, et plus tard en maintenance déposer, n'importe quel élément.**
- **Prévoir une distribution interne qui facilitera le câblage et les soudures.**
- **Passer impérativement en revue l'intégralité de ces aspects par des vérifications sérieuses effectuées sur des dessins réalisés proprement et à l'échelle unitaire : PRIMORDIAL !**

➤ La réalisation "primitive".

Riche d'enseignements, la photographie de la Fig.24 présente une vue plongeante lorsque la toute première version était globalement achevée et le microcontrôleur 80C52 donnant vie à l'ensemble.

À l'instar de la matière qui constitue notre corps, la Terre, les étoiles, le cube grillagé est constitué principalement de vide. (*Ou d'air si vous préférez ...*) Ce n'est pas évident en **3**, car l'image a été saisie dans un environnement sombre et l'arrière de la zone bien que constituée d'une feuille blanche, reste grise sur la photographie. Pour pouvoir faire de l'habillage avec les textes sans que l'arrière plan occupé par l'image ne soit visible et nuise à la lisibilité, tout le pourtour tel que **2** fait l'objet d'un traitement graphique de contournage. (*Effacement de zones par du blanc.*) Sur pratiquement tous ces dessins, j'effectue généralement du contournage en interne comme en **1**. Ces traitements graphiques rendent les figures bien plus compréhensibles. Pour bien faire, il aurait été préférable de contourner l'intégralité du cube, montrant ainsi, comme en zone **1**, que le total est pratiquement vide ... Désolé, mais une paresse coupable m'a découragé face à l'ampleur des effacements à effectuer ! En **4** on discerne les liaisons des lignes horizontales des cathodes, dont les picots sur le circuit imprimé se trouvent à l'arrière. Le socle est un parallélépipède rectangle constitué de cinq faces telles que **6**. Comme je ne voulais pas que le circuit imprimé soit visible, une super idée a constitué à fermer le dessus du socle, et d'y pratiquer 64 trous tel que **5** bien en face des plots sur lesquels se branchent les "tranches" de LEDs. C'est une idée géniale, c'est beau, c'est magnifique, esthétiquement sans critique ... sauf que placer les broches des "tranches" verticales dans les plots de branchement s'est avéré agassivement BERK BERK de chez BERK. Le chapitre suivant décrit ma version du coffret, et explique la difficulté rencontrée.

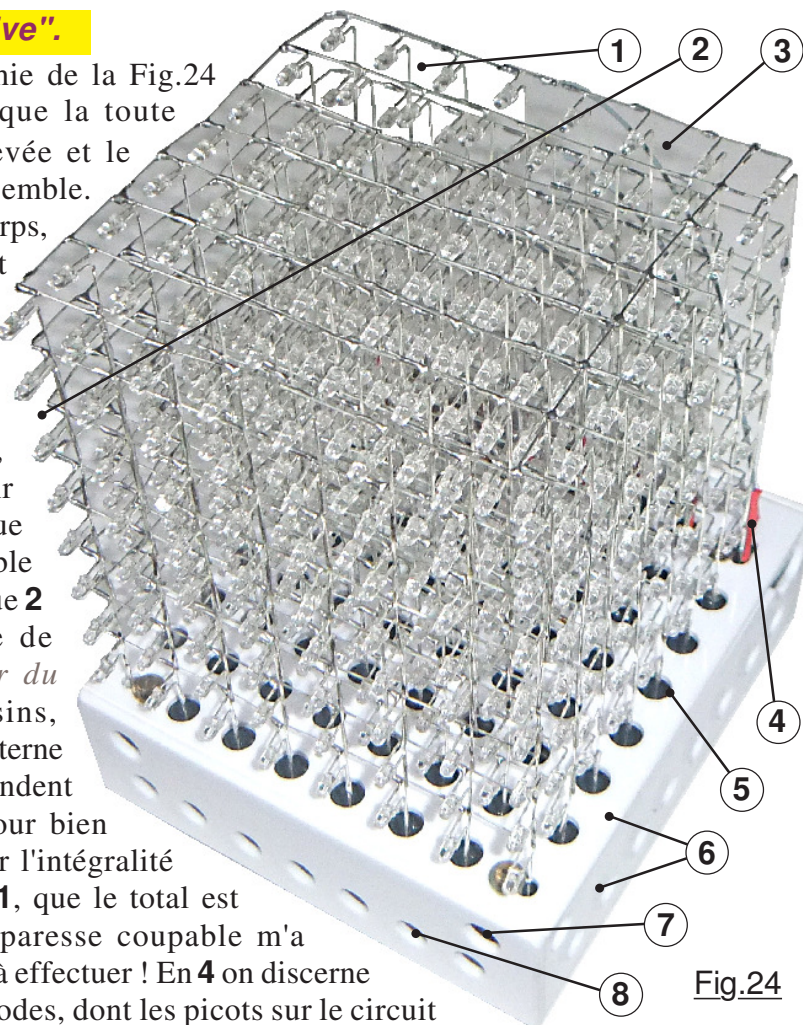
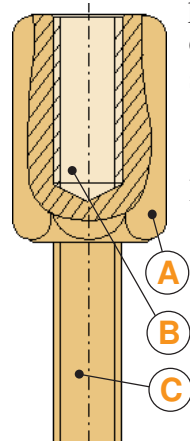


Fig.24

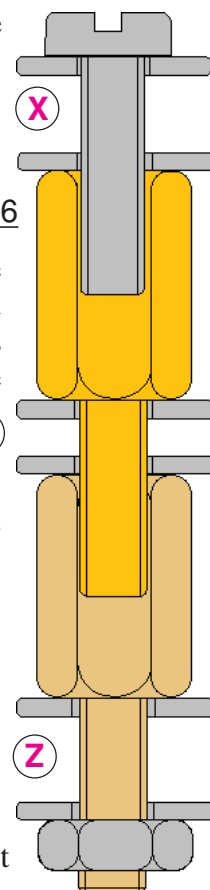
➤ Les dessins à l'échelle unitaire du socle.

Indispensable pour pouvoir engager le façonnage des éléments à souder, dans ce chapitre sera décrit en détails la solution adoptée sur le prototype. Il vous sera ainsi très commode d'envisager une solution personnelle contournant les difficultés rencontrées lors du développement. Les dessins du socle sont réunis dans le sous-dossier dédié <Les plans du coffret>. Ils sont réalisés à l'échelle unitaire et prévus pour être éventuellement imprimés. Comme les périphériques d'ordinateurs personnels peuvent potentiellement engendrer de légères pertes de précision en dimensions, pour lever le doute et pouvoir mesurer tout ce que vous désirez sur les dessins, une règle graduée est ajoutées aux diverses vues.



Structuruellement le coffret est agencé mécaniquement par des plaques horizontales réunies par des entretoises de type Mâle / Femelle. Sur la Fig.25 on constate qu'elles comportent trois zones fonctionnelles avec en **A** le corps hexagonal propice à un serrage par clef plate à fourche ou à oreille. En **B** repéré en couleur plus claire un taraudage ϕ M3 peut accueillir soit une vis, soit le tenon fileté **C** d'une autre entretoise analogue. Mes empilages mécaniques ressemblent souvent à ce que l'on voit sur la Fig.26 extraite du dessin d'ensemble. Deux entretoises de même type sont Fig.25 "empilées" pour former une colonne qui en **X** s'enracine sur la

Fig.26

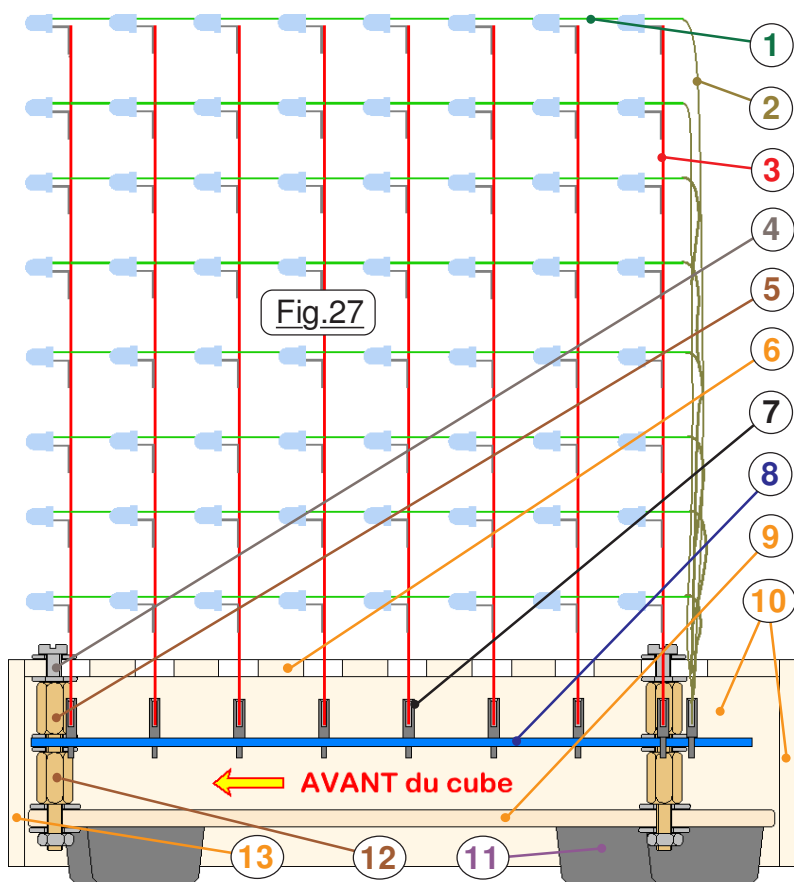


partie supérieure du socle. Celle du haut est colorée en orange pour mieux la situer. Sur mes systèmes mécaniques, toute la visserie est complétée par des rondelles plates assurant de meilleures surfaces d'appui sur lesquelles se répartissent les efforts de serrage. En **Y** l'épaisseur est plus faible puisqu'elle correspond à celle du circuit imprimé du KIT. Enfin, en **Z** est pincée la plaque du fond de 3mm d'épaisseur qui fait office de semelle et se voit munie de pieds en caoutchouc adhésifs.

➤ Coupe transversale de la solution de base.

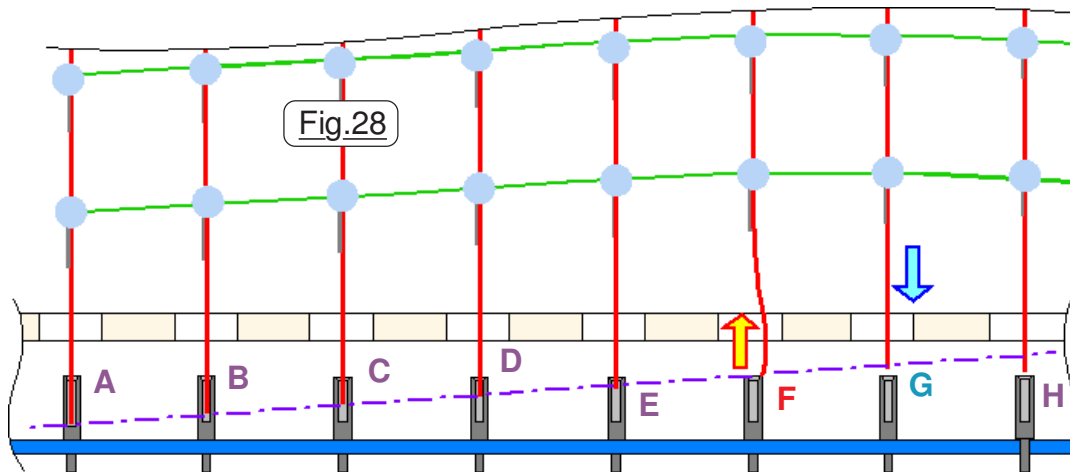
Disponible dans le dossier <Les plans du coffret>, le fichier [Plan A4 Page 1.pdf](#) le dessin repris en Fig.27 représente la coupe transversale du cube 3D. Les lignes des cathodes en **1** sont pontées sur les plots **7** du circuit imprimé **8** par des liens souples **2** à l'arrière du cube. Les anodes constituant les lignes verticales **3** sont repérées en rouge. Le circuit imprimé **8** fourni avec le KIT est solidaire du socle par les entretoises **5** assemblées par les vis **4** assorties des rondelles métalliques plates. Servant "d'écrou", les entretoises inférieures **12** assurent la liaison avec le fond du coffret **9** qui fait office de semelle. C'est sur ce dernier que sont collés les pieds en caoutchouc adhésifs **11**. La jupe du socle est donc constituée "du plafond" **6** percé de ses 64 trous de passage des liaisons avec le cube lumineux, et des faces latérales telles que **10** et **13**. Les décalages latéraux pour **11** constatés sur le dessin résultent de la position des orifices qui sur le circuit imprimé **8** sont forcément décalés par rapport aux composants. De ce fait ils ne

sont pas situés géométriquement aux angles d'un rectangle. Avant d'analyser les divers dessins du socle, voyons ensemble pourquoi cette solution confine un tantinet à une véritable galère au moment de l'assemblage. Comme la distance qui sépare les LEDs du haut des plots de branchement **7** avoisine 22mm, on se doute que la longueur des broches une fois avoir été pliées n'est pas suffisante. Aussi, les lignes verticales **3** constituées des anodes sont prolongées à la base par du fil de câblage rigide. En théorie, c'est du moins ce que j'avais imaginé initialement, l'assemblage est facile. Consultez [Image13.JPG](#) sur laquelle on vient d'achever l'assemblage de la troisième "tranche verticale" en commençant par le fond. **Ainsi on a connecté les anodes par simple branchement** sur les plots **7**, et prolongé les fils souples **2** des cathodes par de petites fiches rigides qui s'insèrent à frottement doux sur la ligne de plots située tout à l'arrière. Comme toutes les lignes sont connectées, il est ainsi possible de vérifier le bon fonctionnement de la couche



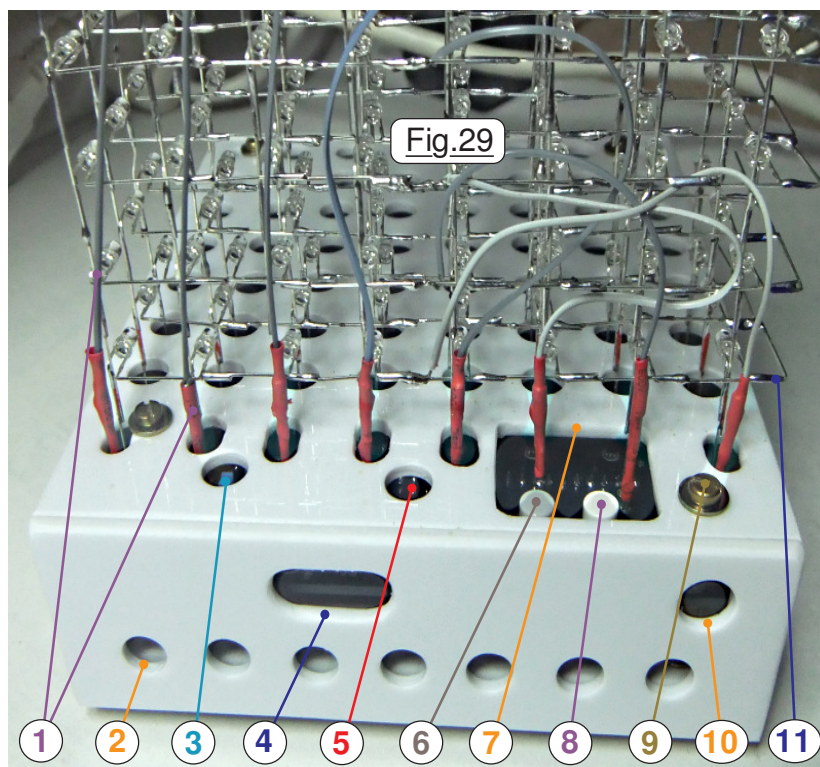
lumineuse. Par utilisation du programme [Test_matériel.ino](#) il devient évident de repérer une LED qui serait bien moins lumineuse que les autres, et la remplacer à ce stade, car dès qu'une "tranche" est ajoutée, l'opération devient impossible. Si c'est le 80C52 qui anime l'ensemble, observer le fonctionnement un bon moment permet aussi de détecter un individu à remplacer. L'information écrite en gras, italique est colorisée en rouge, car dans le "**on a connecté**", le "**on**" c'est moi ! En théorie, on place les huit fils qui prolongent les anodes vers le bas bien en ligne sur la rangée transversale des huit plots. Puis, avec délicatesse et une petite pince, on force tout doux vers le bas, les liaisons s'enfilent à frottement doux et assurent un contact électrique sans problème. Quand on cherche à insérer le fil rigide qui prolonge la ligne des anodes dans l'un des plots tenu à la main, c'est d'une facilité déconcertante. En revanche, arriver à faire pénétrer les huit liaisons simultanément, alors que les hauts des plots sont situés 12mm plus bas que le dessus de la plaque **6** devient une scongregneugneu de corne de gidouille sans nom. Quand les trous de passage ont été percés à un diamètre de ϕ 7mm, ils semblaient presque exagérément grands. Lorsque le fil traverse et que 12mm plus bas on cherche à voir si le fil rigide est bien coaxial avec le plot **7** on ne

distingue strictement rien. Par ailleurs, le moindre que ce fil ne soit pas strictement dans l'axe, comme en **F**, le fait buter sur la partie supérieure du plot. Bien que l'alésage présente une pénétration conique, rien à



Sur cette Fig.28 la grille de diodes électroluminescentes en cours d'assemblage est vue de l'avant vers l'arrière. On cherche à insérer les lignes de fil rigides verticales dans les plots en partant de la gauche vers la droite.

faire. Dès que l'on force un fifrelin sur la grille de LEDs juste à côté en **G** pour la faire descendre, elle se déforme par élasticité et refuse toute insertion. S'il n'y avait que l'effort de pénétration dans les plots, les extrémités de tous les fils rigides seraient alignées sur la droite légèrement inclinée tracée en trait mixte violet. (C'est le cas pour **A**, **B**, **C**, **D** et **E**.) Le blocage se produit systématiquement, et il faut y aller progressivement en partant de la gauche vers la droite, fil par fil. On fait pénétrer en biais en déformant subtilement la grille. Quand on constate la butée comme en **F**, il faut soulever un fifrelin en **G** pour dégager le fil rigide et le remettre dans l'axe du plot **F**. C'est à ce moment qu'en **E** le fil sort de l'alésage et il faut recommencer, avec une suite de @@@@. Bref, après plusieurs "crises existentielles" et un quart d'heure à vingt minutes par tranche, on arrive enfin à les positionner, alors que mise en place il ne faut que deux à trois minutes pour souder les cathodes de la "tranche" située en arrière. (@@@@@@ correspond à une censure car le vocabulaire employé n'était pas vraiment académique !) Autant préciser que lorsque le treillis cubique est entièrement assemblé, **il est hors de question de risquer un déboîtement toujours potentiel, même partiel**. Toutes les précautions seront prises pour l'interdire mécaniquement



comme ce sera explicité plus avant. Puisque nous sommes en train d'assembler les plans lumineux verticaux, un petit regard sur la face arrière, en Fig.29 n'est pas inutile. Examinons ensemble le revers de la médaille avec en **1** les fils souples gris qui relient les "plans horizontaux" des cathodes. Toutes les lignes horizontales d'un plan de LEDs sont reliées en **11** sur le côté gauche du cube quand il est vu de face. Pour pouvoir brancher correctement ces fils souples sur les plots situés à l'arrière du circuit imprimé, leurs extrémités sont munies de minuscules fiches de récupération dont les dimensions correspondent à celles des plots. Ces fiches sont recouvertes de gaine thermo-rétractable de couleur rouge pour renforcer mécaniquement le tout. Sur le dessus, on reconnaît en **3** l'interrupteur Marche/Arrêt qui restera en configuration

alimentation dans la version définitive, car il ne sera plus accessible quand le cube sera enfermé dans un protecteur transparent. L'orifice **5** est prévu initialement pour voir la LED rouge qui atteste de la présence de l'énergie 5Vcc sur le circuit imprimé. Trop présente, cette information n'étant pas d'une importance capitale, l'orifice est actuellement fermé sur le prototype. Donc il me semble inutile de le réaliser, raison pour laquelle il n'est pas représenté sur les dessins. Le grand orifice **7** est assez grand pour

laisser passer les lignes verticales des anodes telles que **1**, et surtout donne accès aux deux boutons poussoir **6** et **8** soudés sur le circuit imprimé. C'est très commode au tout début lors des premiers essais avec le complément Arduino. Par contre, en usage courant, *il serait infiniment plus judicieux de les placer en façade ou sur les cotés*. Aussi, je crois pertinent de ne pas perdre de temps à équiper le circuit imprimé de ces deux boutons poussoir qui ne sont pas exploités par le 80C52. Enfin, sur la plaque de dessus on trouve les vis **9** qui assurent la liaison entre le coffret en Polystyrène choc et le circuit imprimé du KIT. Sur le flanc arrière, en **2** se trouve une ligne d'orifices horizontaux que l'on retrouve sur les quatre faces. Ces derniers n'ont rien à voir avec des trous de ventilation car l'ensemble ne consomme qu'une puissance dérisoire et ne chauffe que très très faiblement. Il se trouve que je déteste les "grandes surfaces". Aussi, sur pratiquement toutes mes réalisations je perce à tout va, pour des raisons purement esthétiques ... encore que ! Tous ces trous permettent de voir l'intérieur, et parfois, lorsque plusieurs LEDs informent de l'état du système, au cours du développement observer ces éléments lumineux s'avère bien utile. Vous trouverez également des lumières fonctionnelles qui facilitent les opérations de maintenance. Par exemple le trou en **10** facilite la mise en place de l'entretoise immobilisée par **9**. Enfin, en **4** la lumière oblongue qui laisse entrevoir le circuit imprimé est pratiquée en regard du petit connecteur

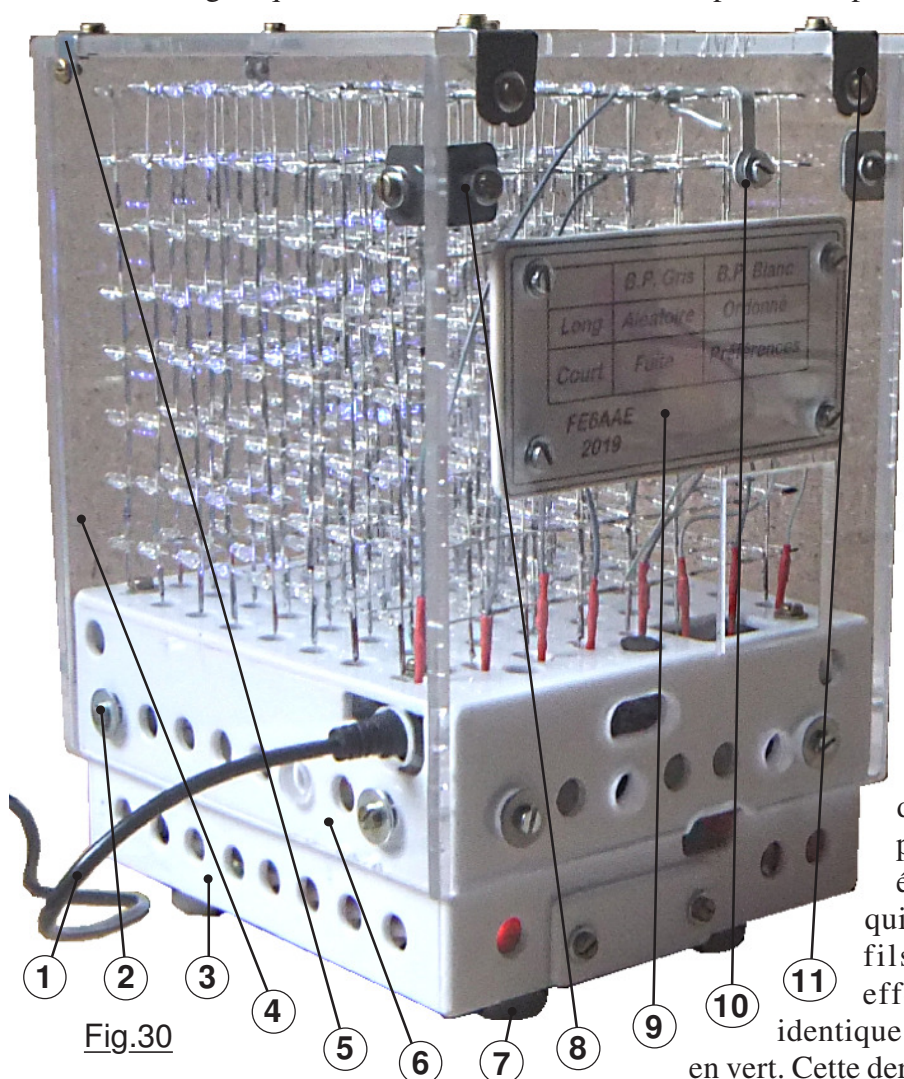


Fig.30

HE14 à quatre broches sur lequel on trouve **GND**, **+Vcc** et surtout les deux boutons poussoir K1 et K2. En cours de développement de programme, les boutons poussoir sont ainsi déportés sur le plan de travail. Il me semble plus rationnel d'oublier cette solution, et comme déjà explicité plus avant, de placer directement le petit clavier sur la face avant de la jupe servant de socle. Avant de passer aux essais, un petit regard sur [Image14.JPG](#) qui détaille une tranche juste avant @@@@ et [Image15.JPG](#) sur laquelle en violet on distingue une ligne soudée sur le dessus qui rigidifie mécaniquement le cube et maintien les "tranches verticales" à des écartements réguliers. Sur cette photographie on se rend bien compte du fait que toutes les cathodes d'un plan de diodes horizontal sont réunies électriquement. Ce sont ces "couches" qui sont reliées aux plots arrière par les fils souples gris. Par la suite, non effective à ce stade, une autre ligne identique a été ajoutée dans la zone surchargée en vert. Cette dernière est immobilisée sur la protection transparente qui englobe le total. (Voir **10** Fig.30) Ainsi, le haut de la grille cubique, qui intrinsèquement reste relativement souple, montrée en gros plan sur [Image16.JPG](#), est immobilisé mécaniquement dans l'espace. (Et ne risque strictement plus de se déboîter.) La Fig.30 anticipe largement la description, car elle représente le cube version Arduino entièrement achevé. On y distingue en priorité le cube extérieur transparent constitué de plaques d'**Altuglas 4** de 4mm d'épaisseur assemblées entre elles par de petites équerres métalliques **5** très fines quand elles masquent les LEDs, et plus grandes en **8** et **11** si elles sont situées vers l'arrière. Il faut surtout observer en **10** une petite équerre qui sert à immobiliser le haut du cube qui sur [Image15.JPG](#) est complétée par le tracé vert. Sur cette version définitive du prototype la "rehausse" **3** est ajoutée à la jupe **6** pour englober le complément Arduino. Le cube de protection ne recouvre que le socle de la version initiale car ses dimensions sont déterminées pour optimiser le choix de produits

commerciaux : Les plaques **Altuglas®** sont constituées de matériaux acryliques totalement translucides et disponibles à des dimensions "quelconques". **Franchement, une épaisseur de 3mm, voir 2mm serait largement suffisante.** Comme la jupe a été initialement percée de gros trous réputés "esthétiques", l'assemblage impose d'utiliser en **2** des rondelles très larges pour assurer à l'intérieur une portée correcte de l'écrou, et à l'extérieur pour cacher le grand orifice qui du coup n'est plus très artistique ! Sur les dessins, ces trous de liaison sont représentés avec un diamètre plus fonctionnel de $\phi 3\text{mm}$. Une belle étiquette **9** sert de "manuel d'utilisation". Notez au passage que sur le flanc droit un gros trou circulaire bien en face de la prise "Jack" du circuit imprimé facilite le branchement de la ligne **1** d'alimentation par un petit bloc secteur USB quelconque. Enfin, en **7** on distingue les pieds en caoutchouc collé sur la semelle.

➤ **Essais ultimes du KIT et décision de poursuivre en version "Arduino".**

Effectuer les essais de l'ensemble peut se faire sans protéger mécaniquement la grille cubique de LEDs en faisant bien attention. C'est d'autant plus vrai que chaque tranche verticale ayant été validée au fur et à mesure de l'assemblage, en principe la dernière face verticale située à l'avant achève l'agencement du KIT. En toute logique, si l'évolution a été conduite avec discipline, le 80C52 mouline son programme, et les multiples points lumineux scintillent. La satisfaction d'avoir abouti est pleinement justifiée. Si le résultat obtenu vous suffit, on peut se contenter de ce stade en sautant directement à la réalisation du cube translucide de protection en **Altuglas**. (Voir chapitre dédié en page 35) Si la phase programmation vous séduit, alors on va dans les chapitres suivant analyser en détails le complément **ATmega328** qui propose une animation à mon sens infiniment plus riche. Pour vous donner une idée, le cube affiche des plans 2D ou 3D mouvants pendant plus de 40 minutes sans jamais présenter deux fois une animation identique. Pour clore ce chapitre, notez que les dessins qui sont regroupées dans **<Les plans du coffret>** proposent les deux versions pour la jupe qui habille le socle. Il est naturellement possible de façonner directement la version haute. Toutefois, le prototype a été construit en deux phases. Coller les plaques ajoutées alors que le cube était en version initiale n'a posé strictement aucun problème. Il vous sera donc possible de créer en deux étapes comme pour le prototype, approche qui laisse libre du choix en différé.

➤ **Quelques remarques relatives aux dessins fournis.**

Confectionner le coffret qui sert de socle doit privilégier les techniques qui sont les vôtres. J'explicite les miennes, notamment dans le chapitre encadré de la page 15. Quelle que soit votre approche, globalement la conception et les dimensions du prototype ont fait leurs preuves de faisabilité et sont optimisées. Si vous observez la solution "de luxe" avec la carte Arduino NANO, les empilages d'entretoises forment des colonnes qui semblent manquer visuellement de rigidité. Il n'en est strictement rien, car chaque plaque horizontale constitue un "étage" qui structure très efficacement l'ensemble. Quand à la jupe qui habille le total, et qui dans ma réalisation est constituée de Polystyrène Choc, elle n'est qu'esthétique et ne supporte aucun effort. Il n'est donc à son sujet pas nécessaire de "construire ultra costaud". Tous les dessins du dossier **<Les plans du coffret>** ne sont pas à imprimer :

- **Plan A4 Page 1.pdf** : Coupe transversale pour montrer l'agencement. (Ne pas imprimer.)
- **Plan A4 Page 2.pdf** : Informations multiples sur l'agencement du socle. (Ne pas imprimer.)
- **Plan A4 Page 3.pdf** : Dessins des éléments du socle. (À imprimer si version simple adoptée.)
- **Plan A4 Page 4.pdf** : Éléments du socle version haute. (À imprimer si version de "luxe".)
- **Plan A4 Page 5.pdf** : Flanc Gauche et flanc Droit de la protection en Altuglas. (À imprimer.)
- **Plan A4 Page 6.pdf** : Face Avant / Arrière de la protection en Altuglas. (À imprimer.)
- **Plan A4 Page 7.pdf** : Dessus de la protection en Altuglas. (À imprimer.)
- **Plan A4 Page 8.pdf** : Dessins de la semelle et du support de la carte NANO. (À imprimer.)

Les couleurs adoptées sur les pages à imprimer sont choisies le plus clair possible quand c'est pertinent pour réduire l'usure des cartouches d'imprimantes ... écologie exige !

Hé Totoche, elle est franchement nulle ta page de bla bla bla, ya pas de photographie et que du baratin. C'est pas avec ça que tu vas faire de la "com" ! Vont même pas tourner la page tes lecteurs ...



7) La carte "informatisée" qui équipe le cube 3D lumineux.

Bien moins connue que le populaire module d'expérimentation et d'apprentissage Arduino UNO, la minuscule carte **Arduino NANO** constitue le cœur de mes projets actuels car de par ses dimensions particulièrement discrète, elle autorise des réalisations vraiment compacte. Bien que pour notre application la miniaturisation ne soit pas un critère, vu le prix d'achat de ces produits j'en équipe maintenant toutes mes nouveaux projets. D'une façon générale, les amateurs d'informatique qui débutent avec la famille des cartes électroniques Arduino, font leurs apprentissages avec l'incontournable module UNO. C'est par excellence un "KIT" extrêmement polyvalent parfaitement adapté à l'expérimentation. Gravitent dans cet environnement à base d'ATmega328 une kyrielle d'autres cartes plus ou moins puissantes et présentant des particularités alléchantes. *Lorsque l'on désire réaliser un ensemble très petit, la carte NANO fait merveille.*

Conçue pour présenter des dimensions vraiment petites, cette minuscule chose est *réputée pouvoir remplacer une UNO sans avoir à modifier une seule ligne de code, réputée 100% compatible.* Ben ... non seulement *c'est vrai*, mais elle est au final plus puissante puisqu'elle met à notre disposition deux entrées analogiques de plus ! On la trouve ce jour pour moins de 3,50• port compris et livrée à domicile avec son cordon de raccordement USB. À ce tarif, elle est moins onéreuse qu'un ATmega328 seul. 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 huit exemplaires commandés, tous à la même adresse, deux cordons USB étaient défectueux : Poubelle. *(Les cordons, pas les NANO !)* Par ailleurs, la ligne USB n'était effectivement pas reconnue par WINDOWS. Toutefois, sur Internet, les fournisseurs 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 l'un de mes projets 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 mon sens il ne faut pas hésiter, et ce d'autant plus qu'elle est encore plus petite que BOARDUINO, alors pour "miniaturiser" elle confine au parfait.

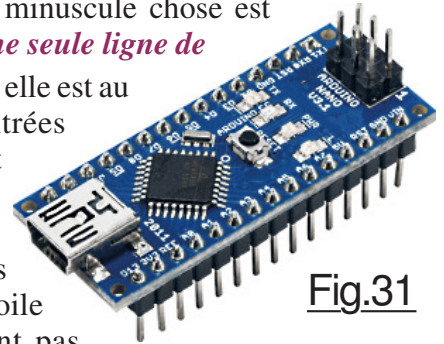


Fig.31

D'une façon générale, je ne mentionne jamais les adresses de mes fournisseurs, dans un souci d'équité pour toutes les entreprises qui font du télécommerce. Je fais toutefois une exception pour cette carte Arduino NANO, dans le but de vous éviter de rencontrer un problème de dialogue sur la ligne série USB de l'**IDE**. Ayant commandé en deux années huit de ces cartes chez :

https://www.amazon.fr/dp/B00QPUEFNW/ref=pe_3044141_185740131_TE_item

je n'ai eu que de bonnes surprises. Oui, deux cordons USB étaient défectueux, mais vu le prix de vente de la carte, franchement ce n'est pas un drame. Si vous approvisionnez chez ce fournisseur, *(Je ne touche aucune commission, vous avez ma parole !)* je joins dans un dossier nommé **<DRIVER USB2>** le module dont vous aurez besoin, accompagné d'un petit fichier texte explicatif.

Attention : Vu que cette carte dispose de deux entrées analogique de plus que sa grande sœur la belle carte UNO, un programme qui les utilise ne pourra pas "tourner" sur la UNO on s'en doute. Contrairement aux six autres entrées analogiques d'Arduino qui peuvent librement être conditionnées en sorties binaire, les broche **A5** et **A6** ne peuvent fonctionner qu'en entrée et n'ont pas de résistance interne de forçage de l'état "1". Pour ceux qui se sentent concerné, et déjà précisé en page 21, est jointe à ce didacticiel la fiche technique nommée **Carte Arduino NANO**.

Pour concrétiser une telle petite fiche, il faut imprimer le document au format classique A4. Puis, couper la feuille bien au milieu sur le trait central. Avec un bâtonnet de "colle U" enduire la surface de l'un des morceaux, *(Et si possible coté non imprimé !)* puis coller les deux éléments en veillant à orienter correctement le haut et le bas. *(Que celle ou celui qui n'a jamais cloué, collé, vissé en inversant le haut et le bas me jette la première pierre ... mais pas une trop grosse toutefois.)* Il ne reste plus qu'à découper tout le tour à environ deux à trois mm du cadre extérieur et vous posséderez une fiche qui se range facilement dans la chemise qui contient les autres fiches.

8) Préliminaires aux développements de la version "Arduino."

Avant toute autre action : **PROTÉGER MÉCANIQUEMENT le treillis de LEDs.** Passer à la version "de luxe" va imposer d'ajouter de la mécanique, de l'électronique, le cube étant retourné sens dessus passé en dessous. De très nombreuses actions vont imposer des manipulations dans tous les sens, dans toutes les attitudes. Par ailleurs, façonner le cube de protection en *Altuglas* qui protégerait efficacement exige encore quelques heures qui sont incompatibles avec la frénésie qui nous titille pour avancer le projet, cette phase peut attendre. Aussi, avant de poursuivre les étapes électroniques, on va réaliser une structure de protection provisoire. *(Nous avons largement le temps, car il faut approvisionner la carte Arduino NANO décrite dans la [fiche dédiée](#) nommée avec originalité **Carte Arduino NANO.**)* Non seulement cette dernière va permettre de poser l'ensemble dans n'importe quelle configuration tout en apportant une protection suffisante. De plus, étant ouverte on peut facilement effectuer des mesures sur la grille de diodes, possibilité indispensable lors de l'élaboration du programme. Les développements du projet ont été chronologiquement les suivants :

- *Créer et assembler la structure de protection provisoire de la grille de diodes électroluminescentes,*
- *Agencer une limande provisoire pour se brancher sur le support 40 broches du 80C52,*
- *Réaliser le circuit expérimental sur une plaque à essais supportant la carte Arduino NANO,*
- *Quand cette dernière sera disponible, (**Voir impérativement le chapitre 7 encadré en Page 22.**) créer le circuit imprimé de complément. Passer en version définitive de la limande de liaison,*
- *Téléverser les données en EEPROM et le logiciel d'exploitation,*
- *Quand l'ultime version fonctionnera parfaitement, remplacer la structure de protection provisoire par le cube translucide.* On peut par la suite à tout moment brancher Arduino par sa ligne USB et modifier à convenance le logiciel d'exploitation pour le personnaliser, le coffret servant de socle est muni à ce titre d'un orifice en face de la prise mini USB. *(Voir les chapitres sur la programmation.)*

➤ **Créer une structure provisoire de protection.**

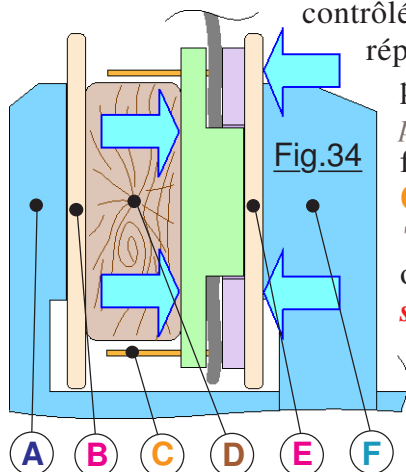
Étape prioritaire pour pouvoir manipuler librement l'ensemble sans prendre le risque d'abîmer la grille de LEDs, il faut réaliser une carapace qui s'inspire de celle montrée sur la Fig.32 dont la photographie présente celle utilisée sur le prototype. Le matériau Polystyrène Choc a été mis à contribution car cet ensemble est formé d'éléments de récupération issus d'anciens projets devenus caduques. Typique d'une organisation de développement "classique", avec en **9** le prototype retourné sens dessus dessous pour accéder au circuit imprimé **3** du KIT immobilisé sur la jupe **5** par les entretoises hexagonales **1**. La structure provisoire de protection et de manipulation est constituée de quatre colonnes telles que **7** et **8** associées au "plafond" **10**. Cet ensemble improvisé est immobilisé sur **5** par des boulons ϕ M3 dont les vis traversent les trous prévus pour l'assemblage du cube de protection en Altuglas. De récupération, le dessus **10** qui sert ici de semelle n'est pas ajouré et ne permet pas de voir les LEDs en vue plongeante. Ce n'est pas idéal, sans toutefois constituer une gêne notable. La petite carte Arduino NANO située en **13** est disposée sur une plaque à essais **16** bien connue des utilisateurs de l'ATmega328. Un petit module expérimental **14** assure au programmeur de pouvoir vérifier simultanément l'état de huit sorties du microcontrôleur, dispositif presque indispensable en développement logiciel. En **15** on devine le Buzzer qui équipera le cube 3D. Sur la plaque à essais on observe plusieurs fils de longueurs diverses en **12**. Pour procéder aux essais et à la programmation, il faut raccorder électriquement le circuit imprimé **3** à la platine **16**. Cette ligne composée du connecteur **2**, de la tignasse de fils **4** rassemblée en **6** et terminée par le HE14 provisoire **11** fait l'objet du chapitre qui suit.

Deuxième volet du préambule au développement de la "version de luxe", il faut impérativement pouvoir relier les broches idoines du support 40 broches du 80C52 qui est écarté définitivement du projet, à la platine d'expérimentation repérée **16** sur la Fig.32 de la page 23. Cette ligne électrique est réputée provisoire, car en version définitive elle va bénéficier des modifications suivantes :

- *Prévue initialement trop longue pour que la platine à essais puisse se voir écartée suffisamment de la structure, il faudra la raccourcir au plus juste pour minimiser son encombrement,*
- *Les fils électriques de cette ligne qui sont fonctionnels sont soudés directement sur le circuit imprimé de complément. On élimine de ce fait le connecteur provisoire HE14.*

Un chapitre dédié sera consacré au passage à la ligne définitive de raccordement. En particulier sur [Image16.JPG](#) on peut voir assez bien le connecteur HE14 double de récupération. Ce dernier assure les très nombreux raccordements effectués sur la platine **16**. Dans la pratique, comme il faut ajouter un circuit 74ALS573, ce sont deux plaques pour essais qui sont mises à contribution durant les validations matérielles. Sur [Image17.JPG](#) on distingue parfaitement les éléments de base de cette ligne de raccordement. Le cœur de cette ligne électrique, bien visible sur [Image18.JPG](#) est constitué d'un connecteur DIL 40 broches.

Attention : Un connecteur DIL à 40 broches prévu pour être associé à une *nappe d'ordinateur au pas normalisé* est relativement couteux, et l'opération de sertissage montrée en Fig.34 est assez délicate à conduire. Il faut exercer un effort symbolisé par les flèches bleues considérable (Voir la Fig.33) pour sortir les 40 lyres sur les fils de cuivre en "coupant" l'isolant gris. Il faut presser régulièrement sur toute la surface jusqu'à "entendre" le bruit discret des clips qui du dessous colorié en vert se verrouillent sur le dessus colorié en violet. Contrairement aux entreprises spécialisées, nous ne disposons pas des outillages dédiés. Je vous livre en Fig.34 ma technique pour mener à bien ce type d'opération qui impose l'usage d'un petit étau de modéliste dont les mâchoires **A** et **F** vont exercer un effort progressif et parfaitement contrôlé. Sur ce dessin la sortie de la nappe de fils est dirigée vers le haut. Pour répartir la pression régulièrement en latéral et en hauteur, on a intercalé deux plaques **B** et **E** d'un matériau relativement doux servant de mordaches. (Deux pièces en Polystyrène Choc pour ce qui me concerne.) Il importe de presser fortement sur le dessous vert du connecteur sans risquer d'abimer les broches **C** qui dépassent. Dans ce but, la pièce **D** (En bois et aux dimensions "calibrées".) est intercalée entre la mordache **B** et le connecteur DIL. Ainsi organisée, la manipulation est relativement facile mais présente un **piège sérieux** : Quand on positionne la nappe de conducteurs entre les deux parties du corps du connecteur, *il faut que les fils soient parfaitement centrés sur les "couteaux" des lyres. Vérifier avec beaucoup d'attention ce critère avant de procéder au sertissage.* Surtout, agissez avec un maximum de doigté, car les efforts exercés sur la poignée de l'étau sont considérablement amplifiés par le dispositif vis/écrou qui généralement constitue la cinématique de ce genre d'outil.



Etant donné que vous n'avez pas à concevoir l'électronique et à la valider, vous pouvez choisir de passer directement à la version définitive. Toutefois, cette étape intermédiaire n'est pas forcément inutile. Je vous la détaille, car elle pourra éventuellement vous donner des idées pour le jour où vous serez les initiateurs d'un projet, qui forcément imposera des étapes de validation intermédiaires. Comme montré sur la Fig.32 la ligne sera branchée sur les platines d'essai au fur et à mesure que l'on ajoute des composants au projet. (LEDs, Buzzer, Circuit intégré 74ALS573 ...) Dans ce but, j'ai utilisé un connecteur HE14 double de récupération coupé à la longueur de **deux fois** 20 broches. Les quarante fils de la nappe y sont soudés en respectant leur ordre sur le connecteur DIL. Pour faciliter l'opération de soudage, un morceau de papier est glissé entre les deux rangées de picots. Consultez [Image19.JPG](#) sur laquelle le n° des broches est ajouté à cette séparation papier. C'est pratiquement indispensable quand on vient brancher des fils qui vont aux platines d'essais, car haut, bas, dessus, dessous, gauche, droite, tous se mélange et l'on "perd facilement le nord". Pour clore ce chapitre, on observe attentivement [Image20.JPG](#) qui présente la limande de raccordement provisoire. On peut sereinement passer à la suite ...

9) Étude du circuit imprimé de complément qui supporte Arduino NANO.

Que vous choisissiez de passer directement à une version définitive, (*Ce qui vous épargnera le travail de soudure des 40 broches du HE14 à deux rangées.*) ou que vous optiez pour la version transitoire expérimentale, dans tous les cas il nous faut élaborer un schéma judicieux, répartir de façon pertinente les ressources de l'ATmega328, affiner quelques valeurs, comme les résistances de limitation de courant dans les LEDs etc. Les deux chapitres qui suivent justifient les choix effectués.

➤ **Le schéma électronique.**

Résumé sur la fiche nommée **Principe de fonctionnement**, on ajoute très peu de composants à la carte Arduino NANO pour pouvoir piloter la matrice cubique de LEDs, car l'essentiel du multiplexage est déjà disponible sur le circuit imprimé du KIT. Notez que c'est volontairement que la Fig.1 a été placée sur le verso de **Répartition géométrique des composants** pour pouvoir simultanément la consulter au regard du schéma 3D. Globalement la petite carte NANO va se substituer purement et simplement au 80C52 avec toutefois un petit hic : Sur l'ATmega328 nous ne disposons que de 22 broches d'interfaçage binaire alors que le grand frère 80C52 en possède 26 en comptant RXD et TXD qui dans notre cas deviennent **RX** et **TX**. Il nous faut impérativement trouver des ressources ... en multiplexant le multiplexage ! Nous allons simplement ajouter un neuvième 74ALS573 qui va servir d'octuple verrous binaires et piloter l'amplificateur de courant ULN2803. La Fig.1 de la fiche **Répartition géométrique des composants** précise que les deux boutons K1 et K2 sont reliés aux entrées analogiques **A6** et **A7** disponibles sur la carte NANO. En standard, sur Arduino on aurait placé les deux boutons poussoir sur une seule entrée analogique, mais ici nous n'avons pas le choix. Les ayant soudés sur le circuit imprimé du KIT leur câblage ne le permet pas. Par ailleurs, ces deux entrées **A6** et **A7** spécifiques au module NANO n'ont pas de résistances internes pour forcer un état logique électrique "1", raison pour laquelle on complète le schéma par deux résistances de **1kΩ**.

➤ **La répartition des Entrées/Sorties de l'ATmega328.**

Malgré le "gaspillage" qui consiste à mobiliser deux broches pour deux boutons poussoir, on dispose de plus de ressources qu'il n'en faut pour gérer les 256 LEDs. Aussi, sur **A5** on a ajouté un petit bruiteur dont initialement la mission consistait à confirmer l'enfoncement d'une touche à l'utilisateur. Du

AFFECTATION DES ENTRÉES / SORTIES	
Broche	Utilisation
D0	Sortie Bit 0 sur les multiplexeurs. (RX)
D1	Sortie Bit 1 sur les multiplexeurs. (TX)
D2	Sortie Bit 2 sur les multiplexeurs.
D3	Sortie Bit 3 sur les multiplexeurs.
D4	Sortie Bit 4 sur les multiplexeurs.
D5	Sortie Bit 5 sur les multiplexeurs.
D6	Sortie Bit 6 sur les multiplexeurs.
D7	Sortie Bit 7 sur les multiplexeurs.
D8	Pilotage du multiplexeur 0.
D9	Pilotage du multiplexeur 1.
D10	Pilotage du multiplexeur 2.
D11	Pilotage du multiplexeur 3.
D12	Pilotage du multiplexeur 4.
D13	Pilotage du multiplexeur 5.
A0	Pilotage du multiplexeur 6.
A1	Pilotage du multiplexeur 7.
A2	Pilotage du multiplexeur vers le ULN2803.
A3	Pilotage LED verte du clavier.
A4	Pilotage LED rouge du clavier.
A5	Pilotage du BUZZER.
A6	Entrée analogique de lecture du B.P. S1.
A7	Entrée analogique de lecture du B.P. S2.

coup, ajouter quelques petits BIPs ici et là s'est avéré bien tentant. Le 74ALS573 ajouté pour le multiplexage est placé en parallèle avec ses huit compères. Notez au passage que les sorties de pilotage sur **D0** et **D1** servent également à l'interfaçage de dialogue sur la ligne série USB du **Moniteur** de l'**IDE**. De ce fait, si durant les essais logiciels en phase de programmation on envoie des valeurs de test sur la ligne série, l'éclairement des LEDs devient "aléatoire" car le multiplexage est perturbé. Comme il nous restait deux lignes d'interfaçage disponibles **A3** et **A4**, autant en profiter pour compléter le tout par deux LEDs qui serviront à informer l'utilisateur sur les options qu'il privilégie lors de ses manipulations de mise en service du cube lumineux décoratif.

10) Le circuit imprimé de complément pour la carte Arduino NANO.

Ensemble matériel et logiciel bien au point, on peut passer à la version définitive. Pour ma part, mis à part des projets "élémentaires", je procède en deux phases. La première consiste à définir avec précision le matériel et écrire les routines fondamentales du "Sketch". Quand le total semble abouti et que l'électronique ne sera probablement plus modifiée, j'engage la version définitive, coffret compris. Ainsi, le développement du programme se fait dans un contexte matériel figé et permet de prendre en compte tous les risques potentiels d'interférences de proximité entre l'électronique d'exploitation et les diverses lignes de raccordement. En général, peu de "difficultés surnoises" sont à craindre, ceci dit, ce n'est jamais exclus. Enfin, c'est toujours la version "utilisateur" qui fait ressortir les manques de convivialité, les fausses "bonnes idées" etc. Voir à ce sujet le chapitre sur *Les "loupés"* proposé en fin du didacticiel.

► La limande de raccordement définitive.

Sachant qu'elle sera soudée directement sur le circuit imprimé de complément, on peut à ce stade la préparer, opération qui consiste à supprimer le connecteur HE14. Pour ma part j'ai coupé les fils de façon à octroyer à la limande la longueur qui sera la sienne, plus une marge de quelques centimètres, on ne sait jamais ! Pour mieux cerner l'agencement définitif de la ligne de raccordement électrique au circuit imprimé complémentaire, anticipons un peu en consultant la Fig.36 ainsi que [Image21.JPG](#) qui présentent l'ensemble pratiquement terminé, mis part l'absence de deux connecteurs HE14. Notez en tout premier qu'en **X** les petits picots qui dépassent du connecteur DIL sont très fragiles. Laissez à peine tomber la ligne sur le plan de travail du mauvais coté, et deux ou trois contacts seront pliés. **Commencez donc en tout premier par protéger ce composant** en le "piquant" sur un petit bloc de mousse. La Fig.36 sur laquelle la rehausse

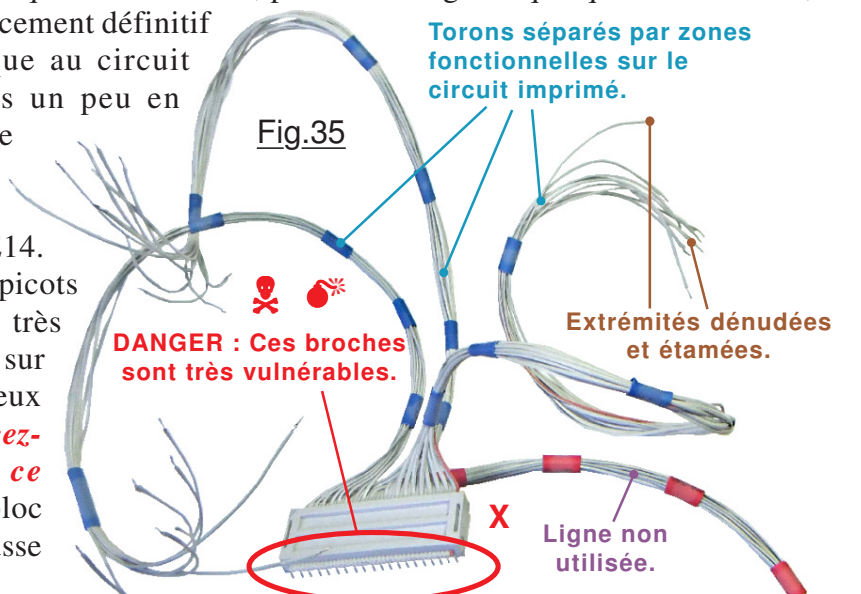
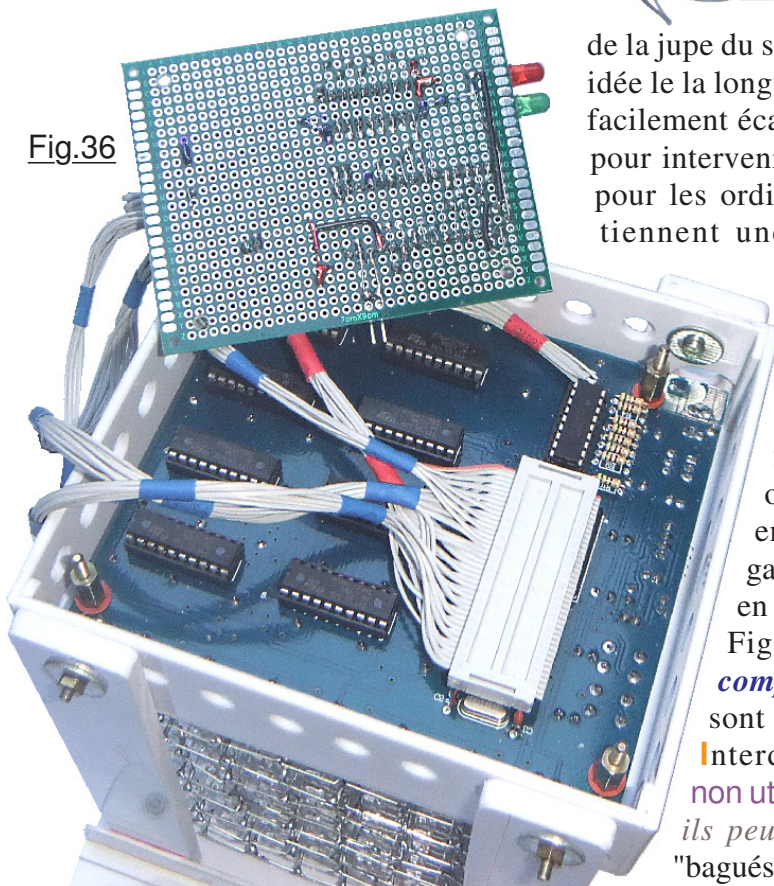


Fig.36



de la jupe du socle n'est pas encore réalisée, donne une idée de la longueur des lignes électriques. On peut ainsi facilement écarter le circuit imprimé dans tous les sens pour intervenir sur ce dernier. Du coup, comme c'est le cas pour les ordinateurs, les liaisons filaires dans les boîtiers tiennent une place aussi importante que les plaques électroniques que l'on miniaturise à loisir. Pour pouvoir loger facilement tous ces fils, et leur donner une souplesse suffisante, ils sont regroupés en torons. Pour atteindre ce but, on réunit divers conducteurs qui vont dans une zone fonctionnelle particulière du circuit imprimé, et on les assemble, comme bien visible sur la Fig.35 en torons à l'aide de bagues coupées dans de la gaine thermo-rétractable. Ce type de produit existe en diverses couleurs. Il saute aux yeux que sur la Fig.1 de la fiche *Répartition géométrique des composants* tous les contacts du DIL 40 broches ne sont pas utilisés. Nommés **C.I.** pour "**C**onnexion **I**nterdite", les fils sont conservés dans la **ligne non utilisée** baguée en rouge. (En développements futurs ils peuvent rendre service.) Les extrémités des fils "bagués" en bleu sont dénudées et étamées **Page 26**

Réalisation d'un pont filaire sur un circuit imprimé ou d'une ligne structurée.

L'opération de branchement proprement dite n'est pas particulièrement délicate, et n'exige qu'un peu de doigté et de la patience. Fil à fil on réalise les liaisons en veillant à ce que la longueur des lignes soit assez courtes tout en assurant une soudure sans contrainte. *Généralement, la difficulté réside dans le dénudage des extrémités.* Pour ne pas galérer, pour ma part j'utilise des limandes de raccordement pour ordinateur. Séparant fil à fil nous obtenons des conducteurs idéaux pour réaliser nos ponts de câblage. *Pour dénuder, rien de plus simple. Le fil est mis en contact avec la pointe fine du fer à souder qui fait fondre transversalement la moitié de l'isolant. Puis avec l'ongle on pince l'extrémité de la gaine grise et en tirant "tout doux" le fil se dénude.* On peut ainsi préparer le fil alors qu'il est déjà soudé à une extrémité. Au départ j'en prends plus long qu'il ne faut. Puis un coté étant soudé il devient facile de déterminer la longueur "juste ce qu'il faut". On coupe, on dénude, on soude et vla le truc ... c'est fait !

avec de la soudure. Ils seront raccourcis au dernier moment à la bonne longueur, avant d'être soudés sur le circuit imprimé de complément. La soudure les rend rigides et l'on peut alors tester facilement leur appartenance avec la sonnette électronique. Avant de souder ces extrémités sur le circuit imprimé, il faut bien entendu concrétiser ce dernier. Pour ce projet, j'ai été un peu obligé d'abandonner mes anciennes méthodes qui consistaient à choisir des plaques de circuit prépercées munies de pistes cuivrées linéaires que je coupais à loisir pour isoler les zones fonctionnelles. Ne trouvant plus en ligne ce type de produit,

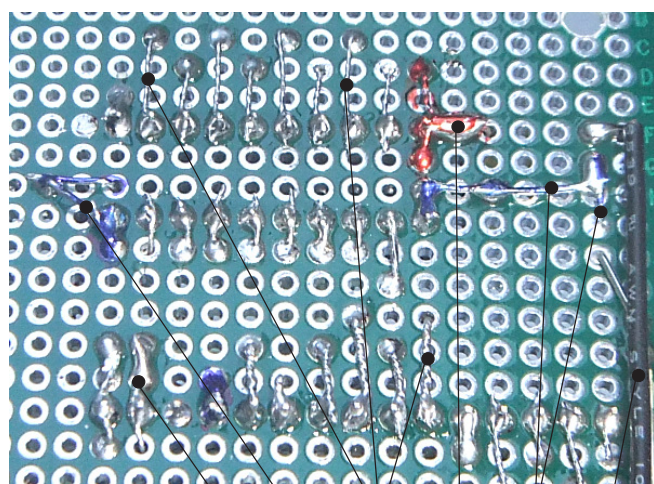
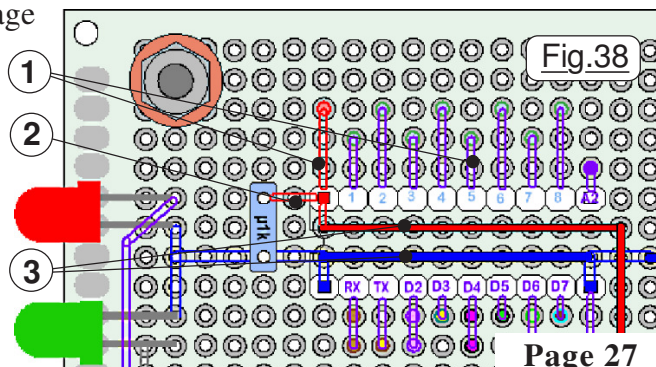


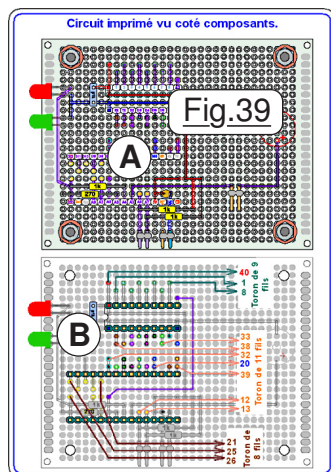
Fig.37

n'est qu'une illusion d'optique car cette vue est photographiée en "macro". En 1 et en 2 ce sont des extrémités de fils rigides ou de composants qui sont soudés. Notez que pour faciliter les repérages de polarité et d'orientation lors des opérations de maintenance, certains ponts sont coloriés avec du feutre d'encre dite permanente. En 4 c'est la queue du condensateur de découplage qui établit le lien alors qu'en 5 le pont est obtenu avec un petit bout de fil rigide. Enfin, en 6 un fil rigide assure un pont en étant isolé sur toute sa longueur.

➤ Réalisation du circuit imprimé de complément.

Soit vous avez approvisionné exactement le même produit disponible en ligne, soit vous avez taillé un équivalent à des dimensions similaires. (Notez au passage que je n'utilise pas les trous situés dans les angles, car à mon sens ils sont trop proches des bords et imposent de la visserie de 2mm. Disposant de place, on peut facilement opter pour des boulons classiques à ϕ M3.) Avant d'effectuer une quelconque soudure, on pratique les quatre trous de passage des vis d'immobilisation. Inutile de préciser qu'il vaut mieux au préalable avoir bien compris la structure du schéma électronique et que la fiche indispensable **Circuit imprimé vu coté composants** est bien en vue sur le plan de travail. La vue de dessous n'est pas fournie car inutile. En effet, comme par exemple en 1 sur la Fig.38 les liaisons qui sont situées sur le dessous sont représentées en mode filaire, alors que sur le dessus en 3 elles sont "pleines" et coloriées. Notez



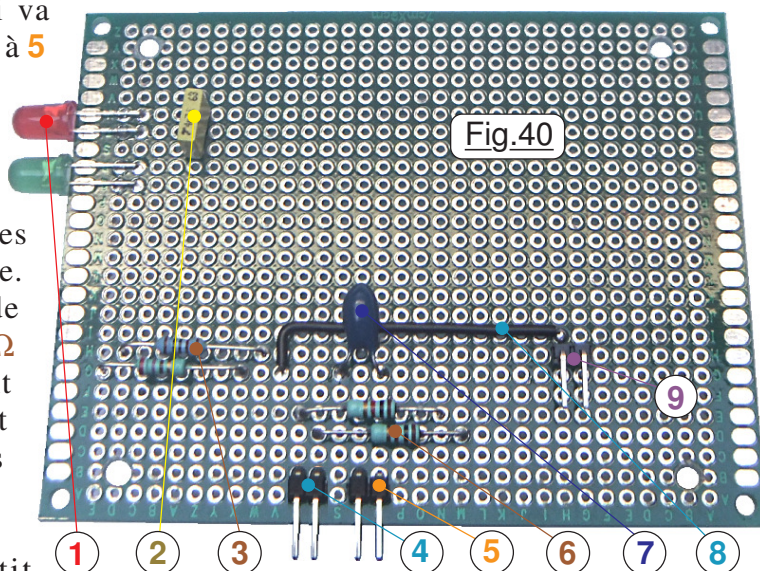


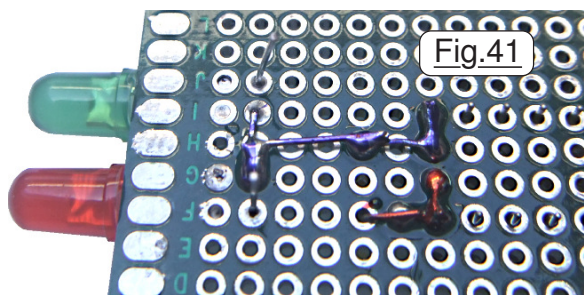
qu'en **2** ce sont les broches du condensateur de découplage qui sont pliées sur le dessous pour aller au support du circuit intégré 74ALS573. Outre ces conventions de représentation, vous pouvez observer que le haut et le bas représentent tous les deux le circuit vu côté composants. Toutefois, le haut en **A** est prévu pour situer les divers éléments ainsi que les ponts de câblage. (Voir la Fig.39) Sur le bas en **B** les pastilles sont plus discrètes car représentées en gris clair. C'est le repérage pour les lignes de la limande de raccordement qui est privilégié. Chaque toron est représenté par une couleur différente. Les fils élémentaires sont symbolisés par des flèches dont la pointe indique le n° de la broche sur le connecteur DIL. Globalement chaque toron comporte entre huit et dix fils. Tous ne sont pas représentés pour rendre plus clair le plan de câblage. Les diverses extrémités d'un même toron sont repérées par des disques de couleurs identiques au centre des

pastilles concernées, mis à part **GND** et **+Vcc** qui sont coloriés respectivement en bleu et en rouge. Entre les dessins de la fiche [Circuit imprimé vu côté composants](#) et le schéma Fig.1 de [Répartition géométrique des composants](#) la réalisation de ce circuit ne devrait pas poser de problème, car vous disposez également d'[Image22.JPG](#) à [Image27.JPG](#).

L'opération s'avère facile si l'on procède par étapes en respectant une chronologie "logique". C'est parti, le fer à souder est chaud. La photographie [Image22.JPG](#) constitue la première phase durant laquelle on soude les composants les moins hauts dont les queues sont pliées à la bonne longueur et passés à travers les pastilles de cuivre étamées. En retournant le circuit pour les souder, elles glissent dans les trous et s'écartent de la plaque imprimée. Pour minimiser cet inconvénient, on commence par les composants les moins hauts, et l'on termine par les plus proéminents. Du reste je ne soude qu'un côté. Puis retournant le circuit je les déforme avec douceur pour qu'elles soient relativement parallèles à la plaque, soignant ainsi l'esthétique du module électronique. Ensuite je soude l'autre côté et passe à un autre groupe de composants. On débute par les "composants horizontaux". En premier sont assemblés les résistances et le pont filaire **8**, qui relie l'ATmega328 au bruiteur via le connecteur HE14 situé en **9**. Celles en **6** de **1kΩ** forcent l'état

"1" sur le connecteur HE14 placé en **4** qui va aux deux boutons poussoir K1 et K2. Quand à **5** il relie la broche RESET du microcontrôleur au bouton poussoir situé sur le flanc gauche du cube. Situées en **3** les deux résistances qui limitent le courant dans les diodes électroluminescentes **1** doivent être calibrées pour que vous obteniez la luminosité souhaitée. Sur le prototype les valeurs adoptées sont de **1kΩ** pour la LED rouge et seulement **270Ω** pour la verte, car ces deux composants n'ont pas du tout des rendements équivalents et surtout présentent à leurs bornes des différences de potentiel notablement différentes. On soude ensuite le condensateur de découplage en **2**. Initialement le petit condensateur au tantale **7** de **10μF** était en orientation verticale. Comme il se trouve entre les deux lignes du connecteur HE14 qui supporte la carte Arduino NANO, il est légèrement incliné au dessus de **7** pour diminuer son dépassement. Séparés de sept trous, ces deux lignes HE14 sont repérées par les encadrés blancs sur [Image23.JPG](#) dont la photographie a été effectuée durant l'opération de soudage. En préambule, pour ma part j'insère les deux lignes de contacts sur la carte Arduino. Puis, le tout étant en place (*Dans les bons trous du circuit imprimé !*) je soude les quatre picots des angles comme on peut le remarquer en **A**, **B**, **C** et **D** sur la photographie. Je retire alors le module NANO et termine les soudures sur les autres picots. Cette technique nous assure la parfaite orientation des lignes HE14 et par la suite la facilité d'insérer ou de déposer le circuit du microcontrôleur. La photographie de la Fig.41, saisie en gros plan lors de la soudure du support DIL du



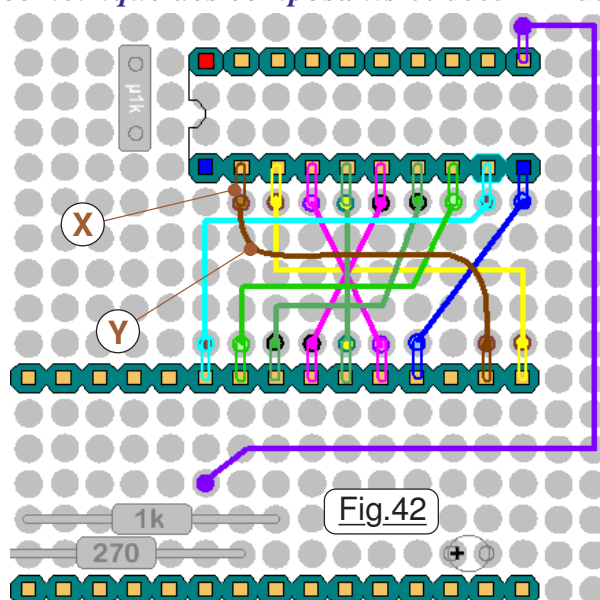


circuit intégré 74ALS573, démontre des soudures "bien mouillées" dont il faut contrôler la qualité au fur et à mesure de l'avancement des travaux. En particulier, avec une loupe à fort grossissement s'assurer que l'étain n'a pas réuni deux picots ou deux pistes voisines. Oui, je sais bien que ça a déjà été souligné, ceci dit je suis persuadé que si je n'insiste pas lourdement, vous serez trop tenté de "foncer la tête dans le guidon". **Un problème**


détecté le plus tôt possible sera infiniment plus facile à contourner. Par exemple, le risque est grand de souder les liaisons en se trompant de piste. La suite consiste à souder les composants les plus "hauts" tel que le BUZZER par exemple. L'[Image24.JPG](#) avec plus de recul situe les divers composants de base. Quand nous avons la certitude de l'assurance garantie que tout est validé, certifié et vérifié avec attention ... on peut passer à la soudure de la ligne de raccordement. Toutefois, il me semble judicieux à ce stade d'effectuer les premières vérifications dynamiques. Avant d'engager ces manipulations, il faut que le microcontrôleur puisse commander l'octuple verrou, c'est à dire qu'il faut établir les dix liaisons avec D0 et D7 ainsi que celle avec A2 et celle reliant GND.

➤ Les liaisons souples entre microcontrôleur et l'octuple verrou.

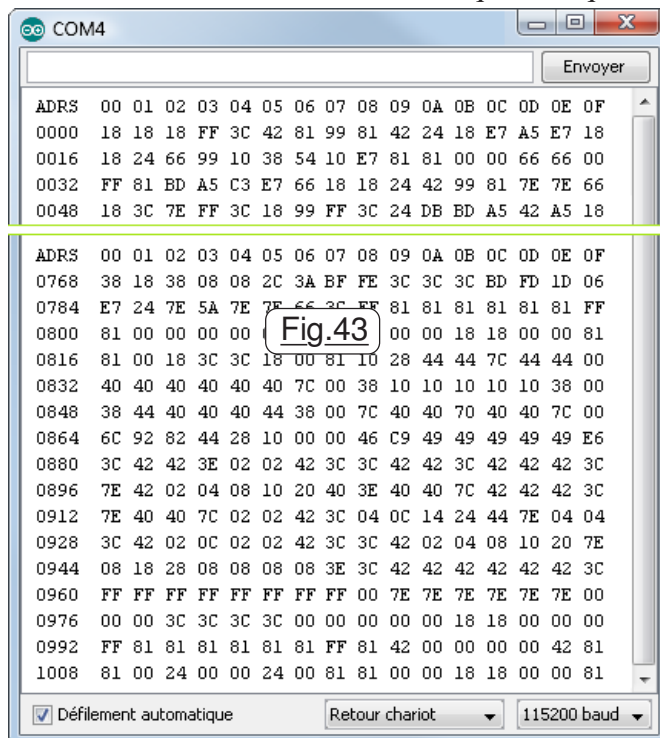
Schéma électronique Fig.1 de la fiche [Répartition géométrique des composants](#) et dessin **B** de la Fig.39 pourraient constituer le "suffisant" pour réaliser ces dix liaisons avec du fil souple. Toutefois, dans cette zone sont également soudés les fils de la limande de raccordement coloriés en orange. Pour vous faciliter au maximum l'opération, la Fig.42 focalise sur les actions à réaliser au cours de cette étape. Seuls sont représentés les ponts filaires à exécuter, et coloriés pour les mettre en évidence. Prendre garde au fait que les fils **Y** coté composants sont isolés par leur gaine grise, mais la partie cuivre de l'autre coté en **X** sert de liaison entre les pastilles. Donc, quand les extrémités des fils sont dénudées, prévoir une longueur suffisante pour assurer la liaison du dessous. Un dernier petit détail pratique : Lorsque l'on dénude à chaud l'extrémité du fil, comme explicité dans l'encadré rose en haut de la page 27, les brins en cuivre ont tendance à s'écarter les uns des autres. Aussi, une bonne pratique consiste à les torsader à la main et à étamer le bout facilitant grandement le passage à travers les pastilles. Ce conseil n'est pas négliger, car nous n'avons effectué que la moitié du travail. Quand on va ajouter les fils (*Oranges sur le dessin B*) de la limande de raccordement, il deviendra un soupçon délicat d'insérer les fils à travers les pastilles, car, comme on peut le voir sur [Image26.JPG](#), il y a pas mal de monde qui encombre la zone.



➤ Chargement du logiciel dans l'ATmega328.

Il est possible d'effectuer la validation de l'ensemble électronique sans la présence du microcontrôleur lui-même sous les directives d'un programme d'exploitation. Pour téléverser le programme d'exploitation et inscrire les données en mémoire non volatile EEPROM, la minuscule carte Arduino NANO est suffisante. Inutile pour le moment de l'insérer sur les lignes HE14 du circuit imprimé de complément. On se contente de relier sa mini-prise à une prise USB standard du P.C. avec le cordon fourni à l'achat. Conformément aux instructions qui accompagnent le "Driver" fourni dans le dossier **<DRIVER USB2>** vous activez l'exécutable [ch341ser.exe](#). Vous invoquez l'**IDE** soit en ouvrant un programme, soit en activant [arduino.exe](#). Comme "Sketch" on commence par téléverser **Ecrire les Empreintes en EEPROM.ino** qui va inscrire dans l'EEPROM les polices de caractères pour les textes et surtout les images 2D et les divers thèmes visuels 3D. Notez qu'en invoquant le **Moniteur série** en cliquant sur l'idéogramme  vous obtenez un listage du contenu actuel de l'EEPROM. **Le listage ne sera correct que si vous imposez à la ligne de dialogue un** **Page 29**

débit de 115200 bauds. Vous devez obtenir le contenu montré sur la Fig.43 extrait d'une copie d'écran. Pour le moment ces données ne sont pas utilisées, car nous allons téléverser provisoirement l'utilitaire **Test_materiel.ino** prévu pour acquérir les validations de base. Pour le moment, c'est suffisant pour procéder aux vérifications initiales. Le programme d'exploitation "définitif" **Cube_3D.ino** sera téléversé directement sur site quand le cube sera totalement opérationnel. Débrancher la ligne USB de l'ordinateur. Placez la carte Arduino sur son support HE14, brancher l'alimentation **+5Vcc** par sa ligne de programmation sur un module secteur USB quelconque et vérifiez immédiatement que la LED rouge **POW** s'illumine correctement. Les deux LEDs **TX** et **RX** sont éteintes. Vérifiez également que sur le support du 74ALS573 le **+5Vcc** est bien présent sur la lyre 20 par rapport à **GND** sur la broche 10. Sur les sorties de multiplexage **D0** à **D7** de l'ATmega328, on doit constater la présence de l'état électrique "1" puisque le programme de vérification commence par allumer toutes les LEDs en mode statique. (*Pratiquement +5V.*) Par contre, sur **A0** et **A1** ainsi que sur **D8** à **D13** c'est un état "0" qui doit être présent. Couper l'alimentation, puis placez le 74ALS573 sur son support en vérifiant bien son orientation. Remettre sous tension. Sur ses huit sorties, une tension statique de l'ordre de +3Vcc doit être présente. Avec une pointe de touche quelconque "cliquez" un court instant entre **A6** et **GND**. Immédiatement les deux LEDs **TX** et **RX** s'illuminent. La sortie **D6** par exemple se met à changer d'état entre environ 0,5V et 1,6V. (*Pour obtenir ces changements d'état "cadencés" sur D6 il faut certainement cliquer plusieurs fois pour obtenir le mode concerné.*) Enfin, mesurer sur **A3** qui est à "0". Tant que l'on maintien **A7** à "0", cette sortie "monte" à l'état "1". Comportement analogue pour **A4** quand **A7** est forcée à "0", sauf qu'il faut un "clic long". Si, comme on peut le voir sur [Image25.JPG](#) un "strap" à languette pont le connecteur du bruiteur, chaque fin de clic s'accompagne d'un petit son attestant de la prise en compte de la touche concernée. C'est tout bon, le processeur prend vie, il s'occupe normalement du clavier à deux touches K1 et K2, nous allons pouvoir passer au raccordement de la limande de liaison 40 fils qui se branche sur l'électronique du cube lumineux.



ADRS	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0000	18	18	18	FF	3C	42	81	99	81	42	24	18	E7	A5	E7	18
0016	18	24	66	99	10	38	54	10	E7	81	81	00	00	66	66	00
0032	FF	81	BD	A5	C3	E7	66	18	18	24	42	99	81	7E	7E	66
0048	18	3C	7E	FF	3C	18	99	FF	3C	24	DB	BD	A5	42	A5	18

ADRS	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0768	38	18	38	08	08	2C	3A	BF	FE	3C	3C	3C	BD	FD	1D	06
0784	E7	24	7E	5A	7E	7E	66	3C	FF	81	81	81	81	81	81	FF
0800	81	00	00	00	00	00	00	00	00	00	00	18	18	00	00	81
0816	81	00	18	3C	3C	18	00	81	10	28	44	44	7C	44	44	00
0832	40	40	40	40	40	40	7C	00	38	10	10	10	10	10	38	00
0848	38	44	40	40	40	44	38	00	7C	40	40	70	40	40	7C	00
0864	6C	92	82	44	28	10	00	00	46	C9	49	49	49	49	49	E6
0880	3C	42	42	3E	02	02	42	3C	3C	42	42	3C	42	42	42	3C
0896	7E	42	02	04	08	10	20	40	3E	40	40	7C	42	42	42	3C
0912	7E	40	40	7C	02	02	42	3C	04	0C	14	24	44	7E	04	04
0928	3C	42	02	0C	02	02	42	3C	3C	42	02	04	08	10	20	7E
0944	08	18	28	08	08	08	08	3E	3C	42	42	42	42	42	42	3C
0960	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	7E	7E	7E	7E	7E	00
0976	00	00	3C	3C	3C	3C	00	00	00	00	18	18	00	00	00	00
0992	FF	81	81	81	81	81	81	FF	81	42	00	00	00	00	42	81
1008	81	00	24	00	00	24	00	81	81	00	00	18	18	00	00	81

➤ Souder les autres lignes fonctionnelles de la limande de raccordement.

Étape ultime de la concrétisation des circuit électroniques, elle sera complétée par l'ajout de la rehausse de la jupe qui constitue le socle du cube décoratif. L'aboutissement présenté sur la Fig.44 laisse entrevoir trois lignes distinctes assurant à la limande complète une souplesse apte à la loger aisément dans le faible volume qui sera disponible. Notez au passage que la visserie ϕ M3 visible en 5 utilise des rondelles isolantes en nylon. Rien à voir avec des propriétés électriques de ce matériau. C'est leur disponibilité en nombre qui a engendré ce choix. En 4 est repérable le "strap" à languette qui durant les essais remplace le futur inverseur qui sera situé sur le coté gauche de la jupe du socle. Le toron 1 constitué de neuf fils amène le **+5Vcc** sur la broche 20 du 74ALS573 et relie les huit sorties aux

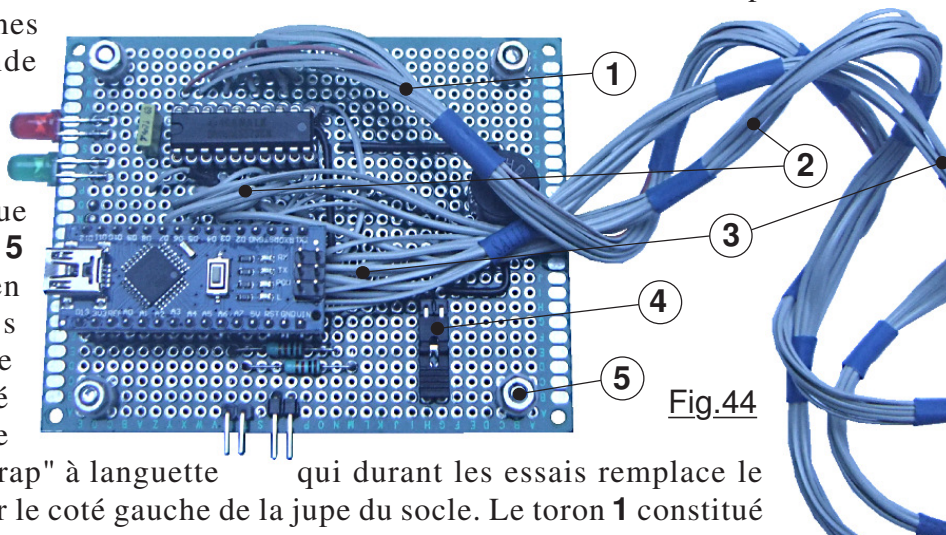
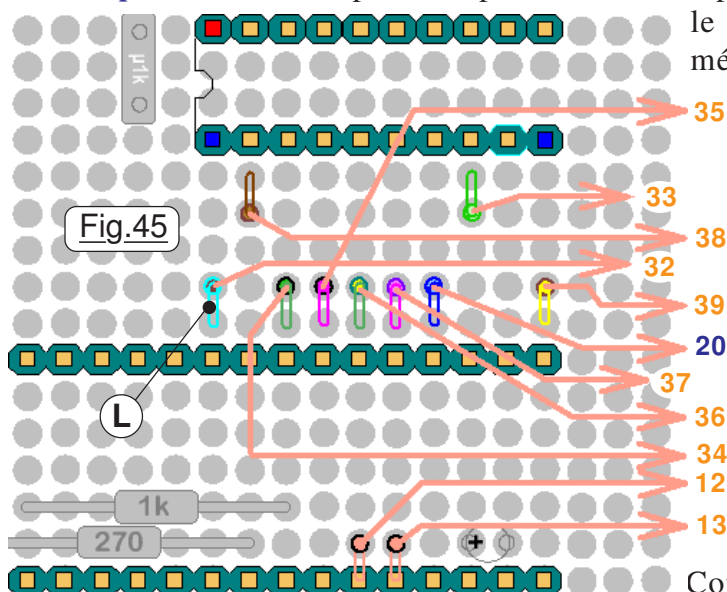


Fig.44

Une précaution systématique quand c'est possible.

Technologiquement, les microcontrôleurs et les circuits intégrés logiques sont peu sensibles à l'électricité statique, et rien n'interdit en général d'effectuer des soudures alors qu'ils sont en place. C'est ce qui se fait habituellement, car prévoir des supports DIL ou HE14 augmente les coûts. En revanche, en activité de loisir nous ne sommes pas en situation de concurrence commerciale acharnée. Il est toujours conseillé de placer les circuits intégrés sur des supports. Ainsi, lorsque l'on procède à des tests de validation, ou en maintenance, on peut librement les enlever du montage pour en isoler fonctionnellement des sections. Dans ce contexte, l'usage de supports nous offre une possibilité facile d'éliminer radicalement le danger potentiel d'un fer à souder qui génère une fuite électrique, ou de la moquette de votre local qui engendre des tensions statiques parfois très élevées. ***Par précaution, j'enlève systématiquement les circuits intégrés de leurs supports lorsque je pratique des soudures et tout particulièrement la platine Arduino NANO.*** Il est toujours préférable de prendre une précaution parfaitement exagérée, que de ne pas l'avoir fait ... précisément quand il le fallait dans un cas particulier pourtant si improbable !

huit entrées du circuit ULN2803 situé sur le circuit imprimé du KIT. Il sera facile par lecture de la Fig.1 disponible sur la fiche **Répartition géométrique des composants**, d'établir le plan de câblage à respecter. La ligne **3** pour sa part, (*Ligne comportant huit fils élémentaires.*) contrairement aux symboles des flèches marrons du dessin **B** de la fiche **Circuit imprimé vu coté composants**, ne sont pas à l'extérieur, mais comme les cinq autres, passent entre les deux lignes de HE14 qui supportent la carte Arduino NANO. Également aisés à repérer sur la Fig.1 de **Répartition géométrique des composants**, elle ne présente pas de difficulté particulière d'interprétation ou au soudage. C'est



le toron **2** repéré en orange sur le dessin **B** qui mérite quelques compléments d'information. Sans que ce soit très compliqué, les onze fils qu'il rassemble se séparent. Deux passent sous la carte Arduino les neuf autres viennent compléter la zone de la Fig.42 et imposent du soin. En effet, il faut faire traverser leurs extrémités dans les pastilles plus ou moins recouvertes par les autres ponts déjà en place. De ce fait, quand on soude la pastille, prendre garde à écarter les conducteurs présents pour ne pas risquer de faire fondre leur isolant. Reprenant la Fig.42 de la zone concernée, **la Fig.45 ne montre que ce qu'il faut ajouter à l'existant pour assembler le toron 2.**

Contrairement à l'artifice de simplification employé sur le dessin **B**, les onze fils du toron y sont tous représentés. L'extrémité de chaque symbole est complétée par le numéro de la broche du connecteur DIL terminant la limande de raccordement. Notez que les liaisons sur le dessous telles que **L** ne vont pas jusqu'au circuit intégré mais sont limitées à la pastille voisine, qui elle est en liaison avec le HE14 par les lignes filaires déjà en place. Mises à part les lignes qui iront aux inverseurs et boutons poussoir, l'électronique est achevée.

➤ Les essais de validation et passage au programme d'exploitation.

Valider intégralement l'électronique fait toujours appel au petit utilitaire **Test_materiel.ino** actuellement chargé dans la mémoire de programme de l'ATmega328. La limande de raccordement est branchée sur le support DIL à quarante broches. Éventuellement on a déporté en façade sur le plan de travail les deux boutons poussoir par utilisation du connecteur HE14 prévu à cet effet sur le circuit imprimé du KIT. L'alimentation de l'ensemble peut se faire indifféremment par la mini prise USB de la carte NANO ou par la prise Jack du circuit imprimé du KIT. Dès la mise sous tension, toutes les LED s'illuminent au maximum en régime statique. Ensuite, chaque appui sur l'un des deux B.P. active une séquence qui allume les LEDs d'un plan particulier dans l'espace, et permet ainsi de vérifier l'indépendance de pilotage de chaque LED individuelle.

11) Programme définitifs et ses options.

Lorsque le programme de validation a confirmé l'intégralité du bon fonctionnement de l'électronique complète du cube 3D, nous pouvons terminer la réalisation du matériel. Toutefois, avant de commencer à façonner, couper, coller, limer, ne boudons pas notre plaisir et installons le programme d'exploitation. Dans ce but, on relie la petite carte Arduino NANO à une prise USB du P.C. et l'on téléverse **Cube_3D.ino** en activant l'**IDE** sur l'ordinateur. Comme avec **Ecrire les Empreintes en EEPROM.ino** les données ont déjà été inscrites dans la mémoire EEPROM de l'ATmega328, il n'y a rien d'autre à faire. À la mise sous tension ou sur un RESET, la LED verte se met à clignoter invitant l'utilisateur à cliquer sur l'un des deux boutons poussoir :

- Si vous cliquez sur le B.P. gris la LED verte s'allume durant trois secondes, pour vous informer que les sons assez "tonitruants" émis lors des séquences MORSE ne seront pas émis.
 - Si vous cliquez sur le B.P. blanc la LED rouge s'allume durant trois secondes, et signale que les BIPs des messages MORSE seront émis ce qui sur le long terme pourrait s'avérer un peu trop bruyant.
- Puis, dans les deux cas le programme d'exploitation passe dans le mode **affichage dans l'ordre croissant d'exploration des thèmes**. Dans ce mode, il y a "balayage" systématique de toutes les variantes puis retour au début pour recommencer un cycle. Rassurez-vous, l'ensemble n'est pas spécialement routinier puisqu'il faut quarante minutes pour consommer une boucle sans jamais afficher la même chose. Chaque thème modifie le plan 3D, ou son mode de déplacement, ou affiche un volume 3D avec une animation spécifique. (Par exemple **Image28.JPG** est un thème 3D.)

NOTE : Si vous optez pour ne pas avoir les BIPs des séquences de textes sous forme de lettres également accompagnées en code MORSE, **tous les autres petits bruitages étant discrets restent effectifs**. Toutefois, vous pouvez désirer le silence total, raison pour laquelle un inverseur est prévu et se branche à la place du "strap" à languette indiqué sur **Image25.JPG**.

➤ **Notice d'utilisation.**

Contrairement à mes autres projets "d'envergure" imposant un manuel d'utilisation généralement très étoffé et comportant un nombre de page non négligeable, pour notre cube lumineux de décoration nous pourrions nous contenter d'une petite étiquette placée à l'arrière du dispositif, à proximité du petit clavier. On se doute qu'avec deux touches, on ne peut disposer que de deux options. Consultez **Image29.JPG** réalisée lorsque le cube était entièrement achevé sur le plan matériel. On observe sur cette photographie que le tableau des options contient quatre variantes et non deux !

En effet, **pour doubler la combinatoire** nous allons faire la différence entre un **clic court** de durée "ordinaire" et un **clic long** qui sera considéré comme tel dès que l'enfoncement d'une touche dépasse la durée de 0,7S. Le comportement du clavier est le suivant :

- Durant l'enfoncement d'une touche la LED verte s'allume.
- Si la durée d'enfoncement dépasse 0,7S la LED rouge s'illumine également.
- La touche est prise en compte au moment de son relâcher. Si c'est un **clic court** le son émis par le bruiteur est un simple "clic" discret. Si c'est un **clic long** générant une action plus importante, l'opérateur est averti par un BIP sonore "plus agressif".

➤ **Les modes de fonctionnement.**

Trois modes d'exploitation sont prévus et l'on peut à tout moment en changer. Par contre, l'option sonore qui autorise ou interdit le bruitage Morse n'est initialisable que sur un RESET.

- **Bouton blanc & clic long** : Balayage ordonné de l'intégralité des thèmes avec rebouclage cyclique.
- **Bouton gris & clic long** : Affichage aléatoire des thèmes puisés au hasard dans la totalité disponible.
- **Bouton blanc & clic court** : **Mode PRÉFÉRENCES** qui regroupe les thèmes les plus significatifs.
- **Bouton gris & clic court** : Fuite d'un thème pour les modes à balayage automatique. Passage à la PRÉFÉRENCES suivante dans la liste car le changement de thème est manuel dans ce mode.

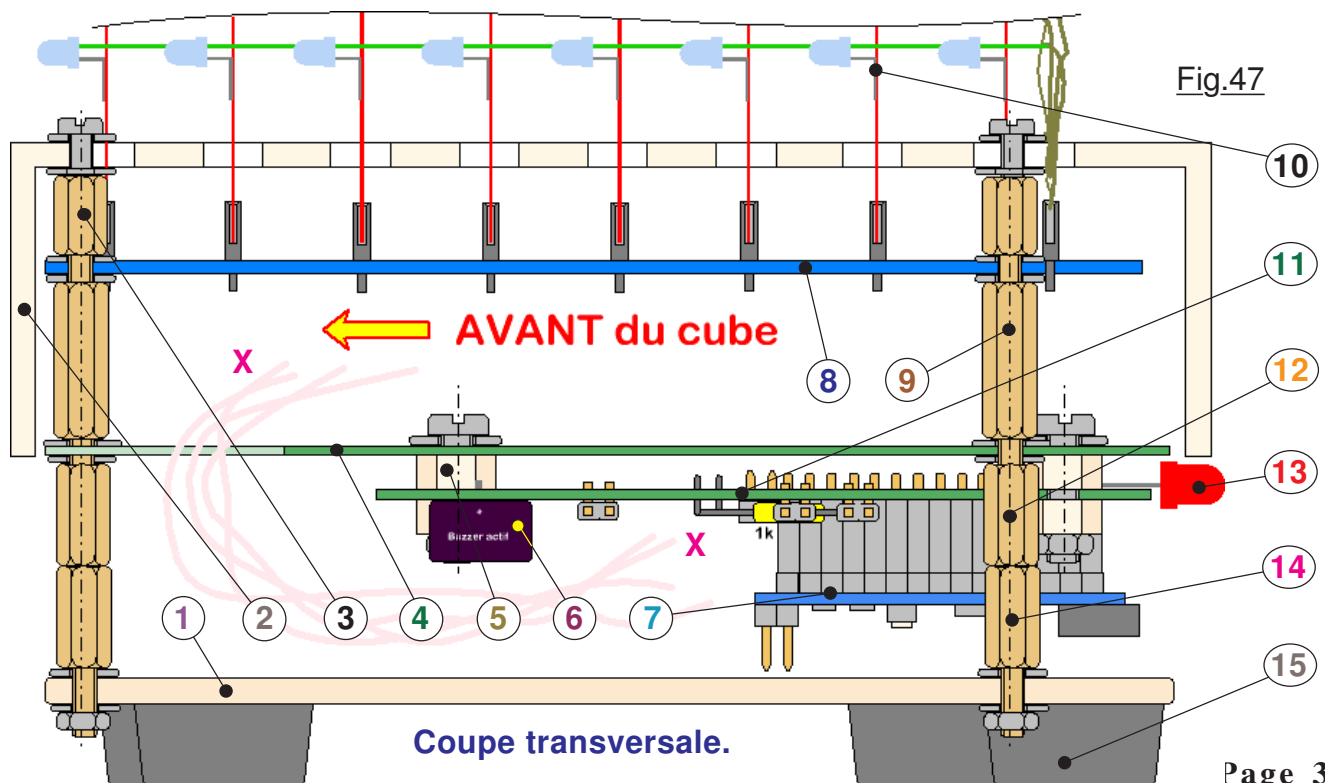
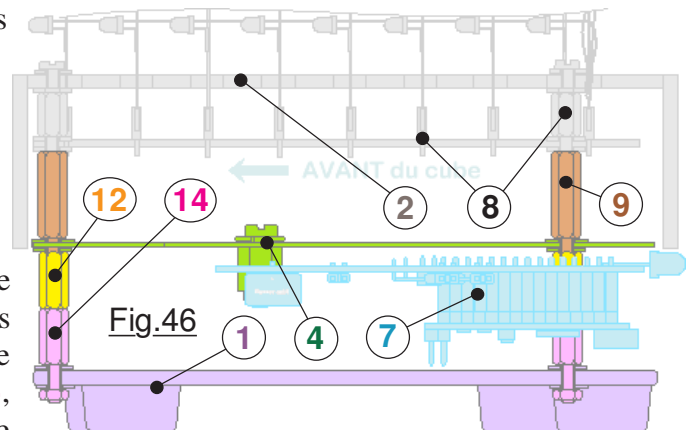
Le manuel d'utilisation est toutefois disponible avec la fiche **Les trois modes de visualisation** qui au verso présente le **Comportement du programme**. Simultanément à cette fiche, il est également possible dans la pratique, de consulter **Ordre des PRÉFÉRENCES** dont le mode est spécifiquement commode quand on désire présenter quelques particularités les plus "séduisantes" à des invités qui ne vont pas forcément rester figés devant le cube durant quarante minutes ...

12) Intégration du circuit complémentaire et finalisation du socle.

Encore quelques "bricolages" à l'atelier et l'élément qui décorera la salle à vivre au prochain Noël sera achevé. Il nous reste deux étapes à franchir pour que l'objet lumineux soit complètement terminé. En premier il faut immobiliser le circuit imprimé complémentaire sur le cube actuel ne comportant que le KIT. Puis compléter la jupe du socle par la rehausse inférieure. Enfin, la structure de protection doit se voir remplacée par un cube qui enferme entièrement la matrice cubique de LEDs tout en masquant le moins possible cette dernière.

➤ **Assemblage du circuit imprimé complémentaire sur le cube 3D.**

Mécaniquement, cette phase consiste à enlever la semelle **1** du cube "version initiale". On la remplace par une plaque **4** dont la fonction consiste à supporter le circuit imprimé secondaire **7**, tout en laissant passer en **X** de la Fig.47 les torons de la limande qui est branchée en dessous par son connecteur DIL à quarante broches. La Fig.46 qui reprend les divers repères de la coupe transversale proposée en Fig.47 constitue le résumé de l'architecture définitive. En gris clair l'ancien socle. On remplace les entretoises initiales par des plus longues en **9** sur lesquelles se trouve immobilisée la plaque intermédiaire **4** observable sur la photographie [Image30.JPG](#) et entièrement équipée sur [Image31.JPG](#). Ce sont les entretoises hexagonales **12** qui servent d'écrou pour serrer cette liaison. Associées à des entretoises identiques **14**, on réalise une colonne assez haute pour que le circuit imprimé secondaire **7** ne touche pas la semelle **1** dont on retrouve en **15** les pieds en caoutchouc. Sur la Fig.47 la jupe **2** ne couvre plus l'électronique jusqu'à la semelle, il faut l'agrandir vers le bas. En **3** les anciennes entretoises hexagonales qui continuent à supporter le circuit imprimé principal **8** du KIT restent inchangées. On ne fait que déposer celle remplacées par **9** quand on dépose la semelle, puis on ajoute **4** et l'ensemble est complet. Plaque support **4** et circuit imprimé de complément **11** sont assemblés au moyen des entretoises en nylon **5** par des petits boulons ϕ M3. En **6** on reconnaît le bruiteur actif et en **13** les LEDs utilisateur avec en **10** la matrice cubique de LEDs. Notez que l'assemblage ainsi complété se fait avec la jupe et avec la structure de protection étant en place. C'est d'autant plus justifié que cette étape est réalisée sens dessus placé dessous.



Coupe transversale.

➤ **Allongement de la jupe vers le bas.**

Transformation qui ne concerne que selles et ceux qui ont opté par une approche en deux volets. Le premier consistait à agencer la version de base issue du KIT. Le deuxième volet consistant à ajouter le complément électronique Arduino, "version de luxe" qui nous impose maintenant d'allonger la jupe vers le bas pour masquer entièrement l'électronique. Comme on va travailler en "sous-œuvre", il faut déposer la semelle et le circuit imprimé de complément. On se contente de dévisser l'entretoise **12**. Il est alors facile d'enlever l'ensemble supportant la carte Arduino NANO ... et dès que c'est fait, immédiatement vous protégez le connecteur DIL 40 contact sur son bloc de mousse synthétique. Pour ma part, j'ai remplacé provisoirement l'entretoise **9** par un simple écrou, ce qui laisse un maximum de place pour travailler. On va coller les pièces complémentaires qui sont représentées en noir sur [Plan A4 Page4.pdf](#) alors que tout le reste de l'électronique est en place. *(La partie actuelle à compléter de la jupe est coloriée en rose sur les dessins.)* Les colonnes de la structure de protection n'empêchent absolument pas ces manipulations de collage.

Diluant cellulosique et électronique ne font certainement pas bon ménage, et j'imagine que cette chimie doit rester un tantinet agressive pour les composants du cube. Aussi, avant que la version actuelle n'émigre vers l'atelier, on commence par entièrement masquer le circuit imprimé du KIT par un couvercle en carton rigide. La matrice est également protégée par un manchon en papier faisant tout le tour à l'intérieur des colonnes de la structure de protection. Enfin, quand on passe du diluant sur les tranches des pièces à coller, l'opération se fait bien à l'écart du cube pour éviter qu'une goutte de ce liquide oxydant ne tombe sur notre bébé si fragile. Surtout travailler dans un local bien aéré pour évacuer les vapeurs éventuelles dégagées par la réaction chimique, autant pour vous que pour l'électronique préservée dans la structure du cube.

NOTE : En observant les empilages d'entretoises hexagonales sur la Fig.47 on pourrait penser que les colonnes doivent manquer singulièrement de rigidité. Et bien pas du tout. Au contraire, comme il y a "triangulation" par le cloisonnement des diverses plaques horizontales, on aboutit à un ensemble très rigide. Notez au passage que la jupe du socle ne supporte strictement aucun effort. C'est la raison pour laquelle sur la partie ajoutée on ne voit pas les petits renforts d'angles présents sur la version initiale prévue avec uniquement le KIT commercial.

Les deux photographies [Image32.JPG](#) et [Image33.JPG](#) ont été réalisées lorsque l'ajout de la rallonge de jupe était achevée et la soudure sur tranche correctement durcie. Le socle est terminé, les protections carton et papier ont été retirées. Les rondelles sont en place pour passer à l'intégration du circuit imprimé de complément. Observez au passage sur [Image33.JPG](#), que de part et d'autre de l'ouverture qui à l'arrière du socle permet de brancher la ligne USB sur la carte Arduino pour la programmer, se trouvent deux trous servant à la traversées de vis \varnothing M3. Quand le cube est pleinement opérationnel et mis en service, on obture cette lumière pour "surprotéger" l'électronique. Sur [Image34.JPG](#) les derniers tests de vérification sont en cours. On voit que l'ensemble de complément peut être facilement écarté du socle.

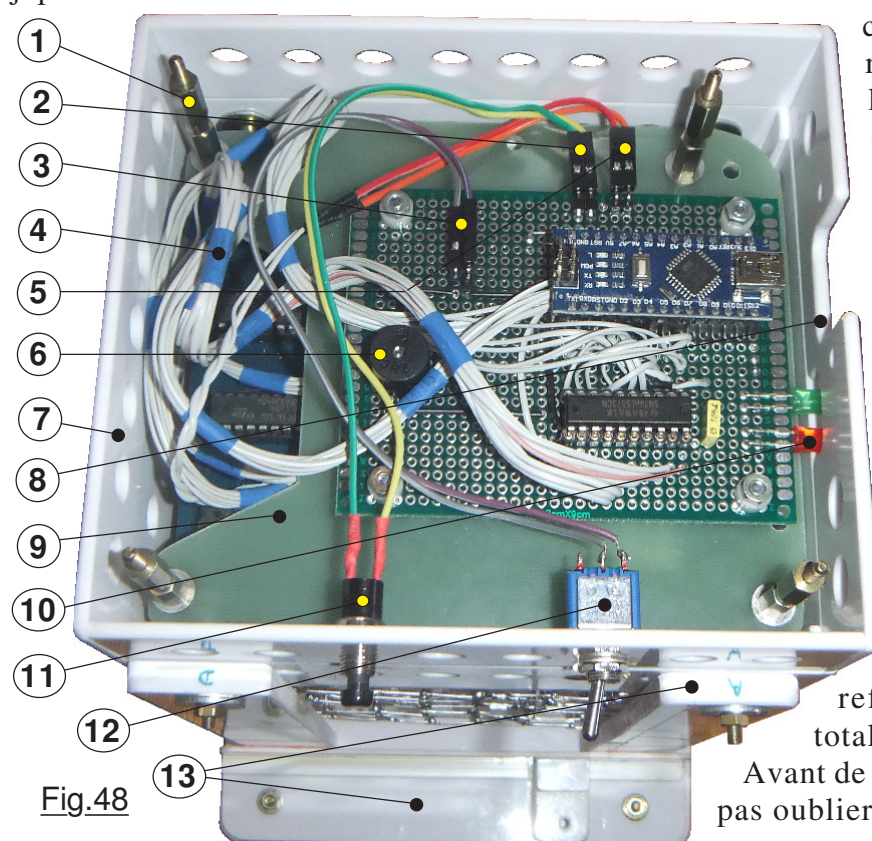
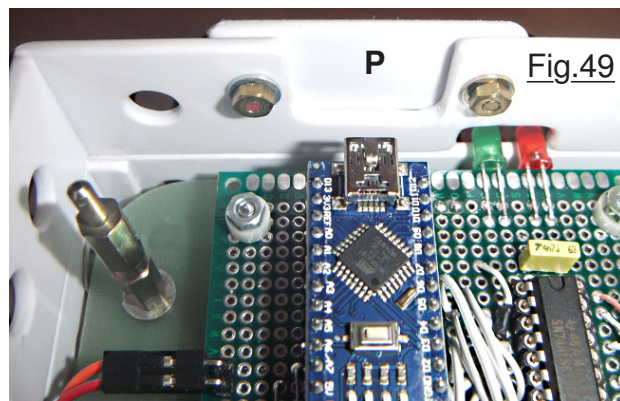


Fig.48

Enfin, en Fig.48 on se prépare à refermer "le couvercle". Le cube est totalement achevé sur le plan fonctionnel. Avant de mettre en place la semelle, il ne faudra pas oublier d'obturer l'ouverture **8**.

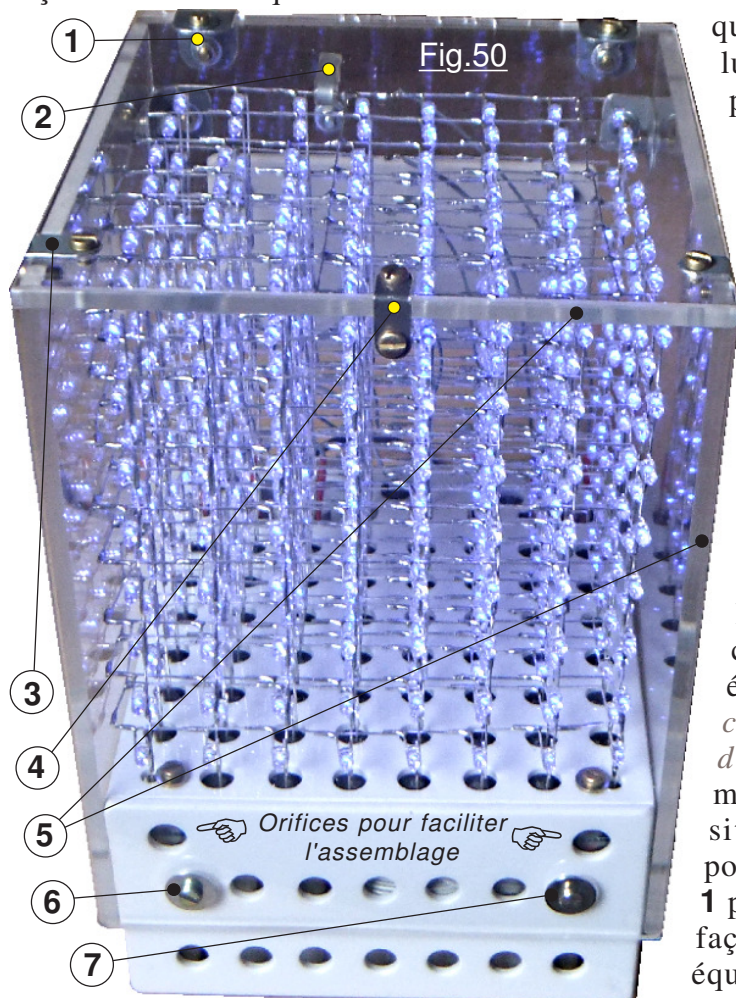
Page 34

Sur la photographie, on devine que la réhausse **7** est de hauteur suffisante pour masquer les extrémités des entretoises hexagonales **1** ajoutées à celles qui immobilisent la plaque support **9**. (*Deux entretoises identiques assemblées pour obtenir la longueur désirée.*) Pour allonger la jupe en **7** vers le bas, le cube est retourné et stable sur la structure provisoire de protection à colonnes **13**. Pour arriver à insérer les deux LEDs **10** dans l'ouverture idoine de la jupe du socle et introduire la plaque **9** sur les entretoises "du bas", il faut incliner l'ensemble comme le précise la Fig.3 donnée en fiche technique nommée *Opérations de maintenance* 1/2. La ligne branchée en **2** va au petit bouton poussoir **11** de RESET, alors que branché en **3** l'inverseur **12** peut couper totalement les bruitages générés par **6**. Comme les divers fils individuels de la limande ont été regroupés en torons **4**, il est manifeste que les loger dans le relativement peu d'espace disponible n'est vraiment pas délicat. La facilité pour passer sur le dessous résulte de la grande échancrure pratiquée dans **9**. Notez que deux fils de la limande sont séparés à leur extrémité des torons et torsadés pour se voir prolonger en **5** pour se brancher sur le HE14 du clavier. (*Les connecteurs HE14 de **2**, **3** et **5** sont de récupération ce qui explique la taille des fils plus importante.*) Les diverses photographies [Image35.JPG](#) à [Image37.JPG](#) complètent les explications de ce paragraphe par des vues en gros plan montrant de nombreux détails commentés. Un dernier petit conseil : N'oubliez-pas de refermer l'échancrure de programmation avec la plaque **P** de la Fig.49 avant d'immobiliser la semelle ...



13) Le cube de protection translucide en Altuglas.

Toutes les lectrices et les lecteurs ne seront pas obligatoirement concernées par la programmation et ne chercheront pas forcément, tout au moins dans l'immédiat, à modifier le programme d'exploitation. Aussi, nous allons enchaîner directement avec l'achèvement des aspects matériels en façonnant le cube qui enferme entièrement la matrice de diodes électroluminescentes, une mutation

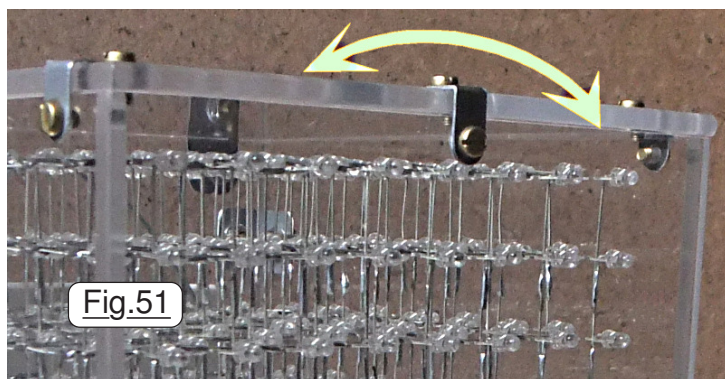


qui fait muter notre projet en une sorte d'aquarium lumineux. Mécaniquement l'étape est élémentaire puisqu'elle se résume à tailler cinq plaques aux bonnes dimensions, et à percer quelques trous pour assurer leur assemblage. *C'est pourtant l'opération de loin la plus délicate*, car ces éléments sont en *Altuglas*, un matériau particulièrement vulnérable aux rayures. *Il va falloir travailler avec d'extrêmes précautions lors des manipulations et des usinages.*

La vue plongeante de la Fig.50 affectée d'une bonne dose de distorsion trapézoïdale présente "le produit fini". La première remarque qui s'impose, est que la fixation du cube sur le socle se fait dans la partie haute de la jupe. Pour pouvoir mettre en place les rondelles d'appui et les écrous ϕ M3 en **6** et **7** il faut dégager l'espace, c'est à dire enlever la semelle et déposer le circuit électronique complémentaire. (*Pas de ralente, en cinq minutes c'est fait ... et surtout n'oubliez pas d'enlever toutes les rondelles.*) Pour masquer le moins possible la grille lumineuse, les équerres situées à l'avant en **3** et **4** sont les plus fines possibles, alors que celles situées à l'arrière comme **1** par exemple sont plus larges pour en faciliter le façonnage. Observez, pas très visible en **2**, la petite équerre (10 sur la Fig.30) immobilisée

sur le fond, qui soutient le cube sur sa partie haute. Le cube de protection, manipulé comme le Saint Graal n'est victime d'aucune rayure, mais quand j'affirme qu'il a été traité avec des pincettes et des gants blancs, c'est à peine exagéré. Autant dire que chaque fois qu'une plaque est serrée à l'étau pour limer ou poncer, les mordaches sont complétées par une bonne épaisseur de papier ou de carton ... propre ! Est-il bien nécessaire de préciser que le serrage des mâchoires est "fifrelin tous doux et avec tendresse". Pas très évident sur la Fig.50, toutes les tranches telles que **5** sont polies et translucides pour parfaire l'esthétique. Pour arriver à la perfection, j'ai l'avantage de disposer de disques abrasifs d'une finesse totale, qui étaient dévolus à des machines à polir des échantillons dans une entreprise mécanique. Il allaient ... à la poubelle. Pour vous donner une idée, les disques d'abrasif de finition sont pourvus d'un "velours rugueux" qui ne ressemble pas vraiment à un matériau prévu pour éroder. En résumé, il vous faudra trouver le papier verre le plus fin possible disponible dans le commerce.

Mis à part le fait qu'*il faut du soin, du soin et encore plus de soin*, peu de commentaires s'imposent, car les dessins réalisés à l'échelle unitaire et proposés dans [Plan A4 Page 5.pdf](#) à [Plan A4 Page 7.pdf](#) sont déjà très documentés par des textes appropriés. L'intégralité des lumières, des orifices et des échancrures à pratiquer s'y trouve représentée. Il me semble urgent de vous inviter à relire les explications sur cette protection en Page 20 et surtout le haut de la Page 21 relatif à l'épaisseur des plaques approvisionnées dans le commerce sans compter un examen attentif de la Fig.30 qui encombre la Page 20. Autre détail qui a son importance : Sur le prototype, le fil de câblage qui sur le dessus de la grille de LEDs rigidifie les écartements a été soudé tout à la fin. C'est toujours dans ces conditions, que ... Plop, une grosse goutte de soudure se détache de la pointe du fer à souder et va se coller tout en bas pour faire un court-circuit entre deux lignes. C'est tout au centre, impossible d'aller détacher ce parasite. Aussi, pour de telles opérations, l'artifice [Image38.JPG](#) consiste au préalable à glisser dans le grillage cubique des "rigoles" en papier qui seront chargées de parer ce type d'incident particulièrement agassif. Pour information, les équerres sont taillées dans de toutes petites plaques en aluminium de récupération, matériau qui s'usine très facilement, surtout lorsque l'on façonne de toutes petites pièces comme celle en **4** de la Page 35. C'est du travail d'horlogerie, car elle présente une largeur de 6mm et percée à ϕ 3,3mm pour le passage des vis. Autant dire qu'à la périphérie il ne reste pas beaucoup de matière.



➤ Une fausse "bonne idée".

Certains effets ne peuvent être constatés qu'une fois avoir concrétisé matériellement leur source. Mécanicien de formation initiale, je pense tout naturellement vis, écrous, rondelles et équerres, le Mécano ayant été mon joué de loin le plus sollicité durant toute ma prime jeunesse. Sur la Fig.51 la flèche verte pastel symbolise les vibrations possibles de la face avant, qui n'étant immobilisée que par les boulons **6** et **7** ne serait pas tenue dans sa partie haute. (Voir la Fig.50) Aussi, la petite équerre **4** rend solidaire le haut de la façade avec le dessus. Et bien on peut faire plus simple avec pour bénéfice collatéral une meilleure visibilité en vue plongeante. Bien que de faible largeur, cette petite équerre se trouve exactement où il ne faut pas. Elle est délicate à façonner. Aussi, il suffit de trouver une petite goupille d'un diamètre inférieur ou égal à 1mm. Par exemple l'axe d'un petit rouage de récupération, un petit clou scié à la bonne longueur etc. On perce la plaque du dessus exactement au milieu de la tranche de la face avant, avec un foret dont le diamètre est légèrement plus faible que celui de notre goupille. On assemble entièrement la protection. Ensuite on contre-perce la face avant, le trou ébauché sur le dessus servant de guide. Puis on change le foret par un outil présentant le diamètre de la goupille. On alèse à ce diamètre sur la longueur idoine. La goupille s'ajuste à frottements doux et assure pleinement son office. ATTENTION : Elle doit légèrement dépasser pour pouvoir la retirer avec une petite pince, car dans le cas contraire le dessus ne pourra pas se translater pour une opération de dépose. Terminons ce chapitre par [Image39.JPG](#) et [Image40.JPG](#) montrant des vues en gros plan. En particulier, sur la semelle bien

visible on observe la directive "**ATTENTION : Deux rondelles**" tracée avec un feutre dont l'encre est permanente. Lorsque l'on réalise les colonnes rigides avec tous ces empilages d'entretoises hexagonales et de rondelles diverses, la hauteur n'est pas strictement identique pour les quatre structures. Si on n'en tient pas compte, quand on serre l'écrou de la plus courte, la semelle fléchit légèrement. Dans ces conditions, lorsque le cube est posé sur une surface bien plate, il boite inexorablement. C'est particulièrement AGASSIF ! Aussi, il suffit d'ajuster la hauteur intérieure de la portée la plus courte en ajoutant une rondelle dont l'épaisseur est calibrée. Du reste, on peut parfaitement la réaliser en taillant dans du carton rigide présentant l'épaisseur de compensation.

O UFFFFFFFFF, on a enfin terminé, et la satisfaction est à la mesure des efforts investis. Le programme utilisateur est dans la mémoire de l'ATmega328. Les empreintes sont en EEPROM. On met sous tension, puis on clique sur le bouton gris ou le bouton blanc et la magie de Noël agit. Pour se faire rapidement une idée des scènes les plus typiques, on clique un court instant sur le bouton blanc pour passer en mode PRÉFÉRENCES. Puis, par des clics courts sur le bouton gris on saute de thème en thème. Le dernier de la liste comme on peut le voir sur la fiche *Ordre des PRÉFÉRENCES* est nommé **Vive NOËL**. Bien que la définition ne soit que de 8 x 8 x 8 on reconnaît un bougeoir avec son anse. La flamme scintille. Et, avec le temps la bougie fond, donc sa hauteur diminue. *Image41.JPG* présente ce thème. Comme les LEDs ne sont pas toutes correctement allumées, car le balayage de multiplexage est plus rapide que la pause de prise de vue, cette photographie est complétée par une surcharge qui situe le bougeoir. Une animation de circonstance.

14) Quelques informations relatives au logiciel.

P robablement que nombreuses et nombreux seront celles et ceux qui ne se sentiront pas concernés. On téléverse les données en EEPROM, on téléverse *Cube_3D.ino* et basta, le problème est réglé. Et bien lisez quand même le chapitre qui suit, il vous est dédié et vous ne le regretterez pas, y compris si aligner du code informatique n'est vraiment pas votre tasse de thé.

➤ Chapitre spécial pour ceux qui ne veulent pas programmer.

A vec ses mirette toute rondes, elle observe et découvre, étonnée, le monde qui l'entoure. Alice est son prénom, elle ne marche pas encore, et à Noël prochain elle aura moins de deux ans. Vous comprenez qu'étant son Grand Père, je n'ai pas résisté à intégrer les ébat d'Alice dans ce cube ludique. Aussi, à plusieurs reprises son prénom va défiler en textes défilants. J'imagine que vous préféreriez remplacer ce prénom par un plus personnel qui vous tient plus à cœur. C'est enfantin, y compris pour les allergiques à la programmation. Voici comment procéder :

- 1) Vous activez l'**IDE** et vous chargez *Cube_3D.ino* dans son éditeur de texte.
- 2) Vous recherchez l'instruction : `PRINT(3,"COUCOU ALICE * *");`
- 3) Vous remplacez le texte en bleu clair par celui désiré.
- 4) Vous téléversez à nouveau *Cube_3D.ino* et ... c'est fini !

Pour le texte qui défile verticalement, c'est plus compliqué, car les caractères sont figés en mémoire non volatile par le programme *Ecrire les Empreintes en EEPROM.ino* à partir du chapitre repéré par la remarque // 824 ----- Afficher le texte ALICE . ----- et il faut coder les matrices de caractères en binaire. Par ailleurs, pour le texte bleu clair que vous changez, sa longueur peut être quelconque, il suffit de ne pas dépasser 17 caractères. En revanche, pour le texte en EEPROM il faut impérativement se limiter à cinq caractères. À vous de voir ...

➤ Quelques fondamentaux qui ont conditionné le programme.

N aturellement, il est hors de question de décortiquer les 1661 lignes de programme. Comme en moyenne il y a deux instructions par ligne, c'est plus de 3000 instructions qu'il faudrait commenter. C'est d'autant moins justifié, que ce chapitre s'adresse aux "fana" d'Arduino qui sont parfaitement capables de comprendre une bonne partie du programme. Par ailleurs, pour aider votre sagacité à comprendre le code source, *Cube_3D.ino* inclus un nombre considérable de commentaires, c'est presque un journal. En tête de listage figure un historique qui présente l'évolution de la taille du code objet. Aussi, si vous désirez faire de la place pour inclure des séquences personnelles, vous pourrez choisir quels thèmes supprimer en fonction de l'intérêt relatif qu'ils présentent et de la place qu'ils occupent. De façon analogue, les diverses empreintes figées en mémoire non volatile sont parfaitement repérées dans le source de *Ecrire les Empreintes en EEPROM.ino* par des

commentaires qui précisent la nature des informations et l'adresse relative qu'elles occupent en EEPROM. Il vous sera ainsi très facile de remplacer certains plans 2D ou certains volumes 3D par des créations artistiques personnelles. L'intégralité des dessins plans sont représentés sur les fiches *Matrices symétriques* et *Images complètes* avec en violet l'adresse relative en EEPROM du premier octet de la grille binaire. Sur la fiche *Police de caractères* sont précisées les empreintes des éléments disponibles pour les textes défilants. Quand à la fiche nommée *Les matrices graphiques : Images / Textes*, quelques dessins sont représentés en bas du document. Le moins compréhensible est celui de l'*Oiseau qui vole*. Il est constitué d'une animation de sept plans, codés en mémoire par demi-images verticales. C'est au chargement qu'est reconstitué le carré entier. On divise ainsi par deux la place occupée en mémoire EEPROM. (*Cette technique est utilisée à outrance, et en particulier pour toutes les images de la fiche Matrices symétriques ainsi que pour les volumes symétriques comme les pyramides par exemple.*) Rien ne vous interdit d'en modifier à votre guise.

Résumé des points importants du logiciel d'exploitation.

- On ne peut afficher simultanément de façon statique qu'un plan horizontal de direction XY.
(*Le cas d'allumage statique simultané des 512 LEDs est un cas particulier mis à part.*)
- Tout affichage autre qu'un plan horizontal fixe impose un balayage du volume.
- Chaque transmission de donnée sur la ligne USB du moniteur force des "1" sur X6 et X7 et perturbe les affichages sur le cube. Il faudra en tenir compte lors de certains essais. Notez que sur le programme d'exploitation *Cube_3D.ino* il n'y a aucun échange sur la ligne série de l'IDE. En phase de développement, cette ressource est toutefois incontournable pour mettre au point des séquences délicates. On se fait visualiser des valeurs critiques par exemple, des indications de passage à des instructions spécifiques etc. Peu importe alors ce qui s'illumine dans le cube.
- Par choix logiciel, tout appel à un *PGMn* commence par *Eteindre_le_cube()*.

➤ Organisation des données et coordonnées cartésiennes.

Indissociables pour agencer un programme quel qu'il soit, deux notions sont intimement liées. La première est relative à la façon dont les données sont mémorisées pour minimiser la place occupée en mémoire, que ce soit en EEPROM ou en mémoire de programme. (*Ainsi qu'en RAM dynamique.*) La deuxième, c'est la technique d'utilisation de ces informations. Dans notre application, les données sont globalement constituées de points lumineux (*PIXEL*) que l'on dispose sur un plan ou dans un volume cubique. Les arrangements de type grille rectangulaire sont désignés par le vocable de *MATRICES pouvant présenter trois dimensions*, (*Les volumes.*) *ou plus si l'on associe ce volume à des paramètres physiques* par exemple. Dans notre cas, le nombre de "molécules" est très limité. Pour repérer chaque *PIXEL* dans un plan ou dans l'espace du cube, on utilisera des coordonnées cartésiennes exprimées dans un repère trirectangle dont la fiche *Implantation logicielle du cube 3D en RAM* explicite la géométrie retenue et l'implantation en RAM des couches horizontales qui composent un thème visualisant un volume complet. Pour analyser la genèse des "textes défilants", la fiche *Implantation logicielle des données à afficher* précise la gestion en mémoire vive. Enfin, certaines animations favorisent des géométries construites par des colonnes verticales de points lumineux. La fiche *Quelques aspects logiciels* décrit les techniques informatiques utilisées.

15) Note sur l'élaboration du programme.

Comme j'ai été en mesure de le constater à chaque développement d'un projet, les premières séquences écrites sont boulimiques en ressource du microcontrôleur. Aussi, c'est toujours un peu dans la peur de manquer d'espace mémoire que l'on avance, ajoutant de nouvelles routines. Au début l'encombrement du code augmente fiévreusement, car chaque nouveau thème d'affichage impose de nouvelles routines. Aussi, avec un acharnement qui confine à de l'obsession on optimise, on corrige et on modifie pour gagner un petit Octet ici, un autre là. Puis, et c'est ce que j'ai été en mesure de constater à chaque fois, quand le "noyau" est en place, on ajoute des options à profusion sans que le code ne diverge, car on réutilise à outrance les séquences paramétrées pour offrir une combinatoire "explosive".

Observant le programme source, vous pourrez lire en tête de listage l'historique du développement. Vous allez constater que les thèmes n'ont pas été ajoutés "dans l'ordre". C'est assez normal, car lorsque les séquences paramétrées étaient au point, pour éviter de

lasser le "spectateur" par des thèmes semblables, les programmes d'affichages ont été éparpillés. Vous pouvez vérifier que de `PGM_0()` à `PGM_17()` (*Alors que `PGM_11()` à `PGM_16()` ne sont pas encore installés*) on a consommé déjà 30% de la mémoire disponible. Puis, les programmes qui suivent deviennent de moins en moins voraces. Aussi, arrivé à `PGM_41()` on ne dépasse pas les 51% d'occupation. La richesse des affichages est considérable puisque déjà il faut une durée significative pour enchaîner les séquences écrites. Il reste pratiquement la moitié des ressources encore disponibles alors que toutes les idées "de base" sont effectives et en place.

Comme j'ai à cœur de toujours "saturer" l'ATmega328 quand je développe un projet "sérieux", (*En tant que programmeur, j'estime que ne pas employer le matériel à 100% relève un peu de la paresse.*) le problème s'inverse : Comment arriver "gloutonner" tous les octets disponibles ? Pour la mémoire EEPROM c'est gagné, la totalité des 1024 octets disponibles en mémoire non volatile est utilisée par l'utilitaire `Ecrire_les_Empreintes_en_EEPROM.ino` et maintenant toute nouvelle image éventuelle devra résider en mémoire de programme sous forme d'un tableau. En revanche, comme les 49% d'espace disponible constituent une abondance inespérée, on peut se permettre d'abuser, c'est à dire de concevoir de nouvelles séquences sans se préoccuper exagérément de la consommation en code objet. C'est la raison pour laquelle à partir de `PGM_42()` on peut observer des thèmes qui à eux seuls se gavent facilement de 5% d'un coup, sans vergogne. Par exemple le thème de `PGM_42()` juste pour "tricoter un cube" la bagatelle de 1810 octets. Autant dire que l'on dilapide l'héritage à tout va, comme si nous ne l'avions pas gagné avec peine !

Le thème le pire, `La_pluie()` qui a été intégré ultérieurement dans `PGM_5()` est de loin celui qui se goinfre le plus d'espace mémoire réservé au programme. À lui seul il réduit notre capital de 1934 octets et représente 6% du programme objet !

Pour ce qui relève de la mémoire dynamique, même abus, plus de recherche fébrile pour gagner un ou deux octets. C'est l'avantage de toujours aborder un développement avec de l'optimisation systématique ... du coup à la fin on peut se faire plaisir dans "des délires informatiques" sans retenue.

➤ **Durée actuelle d'un cycle de visualisation.**

Cycle de visualisation fait référence à un affichage en mode automatique, l'intégralité des thèmes étant parcourue dans "l'ordre croissant". Pour évaluer la durée d'une boucle complète, la ligne série USB du Moniteur a été mise à contribution en affichant en début de boucle la valeur du compteur `Millis`. La liste donnée ci-dessous présente en vert la chronologie sur quatre cycles, les thèmes qui suivent n'étant pas encore programmés. Puis, dilapidant sans vergogne les octets qui restaient de disponibles dans la mémoire de programme, les thèmes colorés en bleu ciel dans le texte ont été ajoutés, avec pour chacun la nouvelle valeur de durée du cycle complet qui en résulte.

Mise en service : `Millis` = 3000

Après une boucle complète : `Millis` = 2110458 soit 2104 S pour une boucle complète. (35 minutes.)

Après une deuxième boucle : `Millis` = 4217090 soit 2106 S pour une boucle complète. (35 minutes.)

Après la boucle n°3 : `Millis` = 6327930 soit 2110 S pour une boucle complète. (35 minutes.)

Après la boucle n°4 : `Millis` = 8443128 soit 2115 S pour une boucle complète. (35 minutes.)

`Rayons_filaires()` : 42 secondes; Total = 2157 S pour une boucle complète. (36 minutes.)

`Le_serpent()` : 40 secondes; Total = 2197 S pour une boucle complète. (37 minutes.)

`BUBULE()` : 55 secondes; Total = 2252 S pour une boucle complète. (37 minutes.)

`La pluie` : 64 secondes; Total = 2316 S pour une boucle complète. (38 minutes.)

`Sépare un cube en deux` : 20 secondes soit 2336 S pour une boucle complète. (39 minutes.)

`Sépare un cube en quatre` : 36 secondes soit 2372 S pour une boucle complète. (39 minutes.)

`Le bougeoir` : 36 secondes soit 2408 S pour une boucle complète. (40 minutes.)

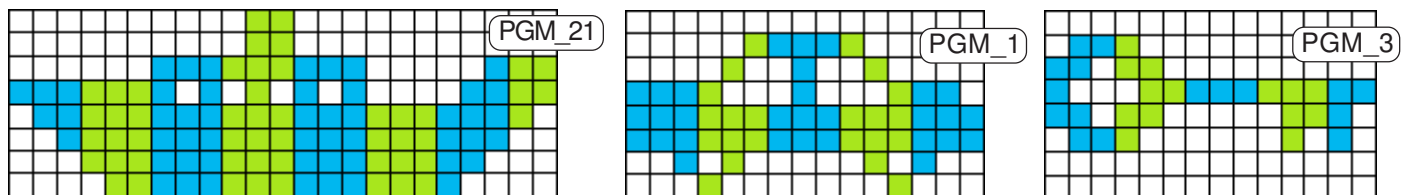
Bien que les informations des chapitres précédents soient déjà une aide précieuse pour celles et ceux qui désirent se plonger dans les arcanes du logiciel d'exploitation `Cube_3D.ino`, sans effectuer une analyse de toutes les particularités de codage éparpillées dans le code source, il me semble pertinent d'aborder quelques spécificités, dont l'agencement ou le déroulement n'est pas particulièrement élémentaire. Le traitement de matrices de points carrées ou cubiques en coordonnées cartésiennes semble assez élémentaire à première vue. Cette apparence "intuitive" est assez trompeuse, et ce d'autant plus que l'on cherche à minimiser le code objet. Tout particulièrement les rotations à 90° des plans font appel à des "chassés-croisés" assez indigestes : Voir la suite ...

16) Les matrices graphiques : Images / Textes.

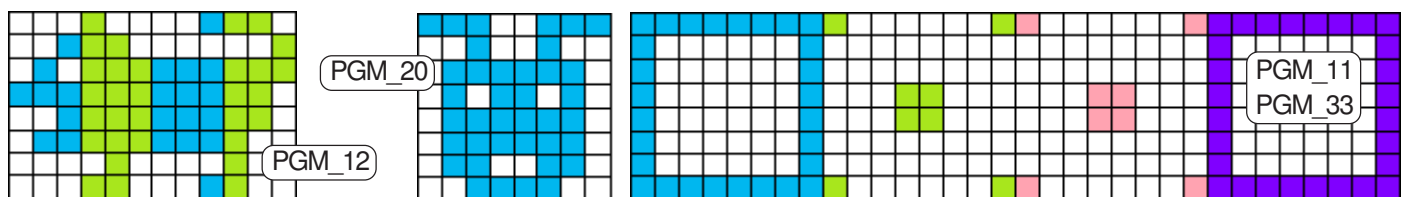
Figés en EEPROM, les éléments affichés en plan ou en volume sur le cube de LEDs sont élaborés en vue de minimiser leur encombrement dans la mémoire non volatile de l'ATmega328. Pour diviser par deux la place mémoire occupée, les modèles compris entre 4 et 40 compris sont construits par symétrie horizontale, seule la moitié du dessin étant logée en EEPROM.

Toutefois, certaines redondances résultent de l'historique du développement du programme. Par exemple le texte ALICE ainsi que les chiffres du compte à rebours semblent "en doublons". Ils ont été conservés pour ne pas avoir à reprendre entièrement les séquences de programme qui les gèrent. Par ailleurs, inclus dans des matrices de 7 x 6 et 8 x 8 ils sont géométriquement mieux adaptés à ces deux items. Que ce soit les dessins "symétriques" ou les "images complètes", leur répartition dans l'EEPROM est organisée pour éviter d'avoir en voisinage des dessins de nature proche. On peut ainsi faire afficher successivement toute une catégorie de matrices tout en évitant un effet de "lassitude". Par exemple un cercle tel que 2 est suivi d'un 'X' tel que 3, puis d'un carré en 4 etc.

Toujours pour des raisons d'optimisation, les sept images constituant l'oiseau qui vole utilisent également la particularité "miroir" des dessins, mais cette fois la symétrie est verticale. Noter également que pour simplifier la routine d'affichage de texte, ZERO et 'O' sont codés deux fois dans la police de caractères bien que strictement identiques géométriquement.



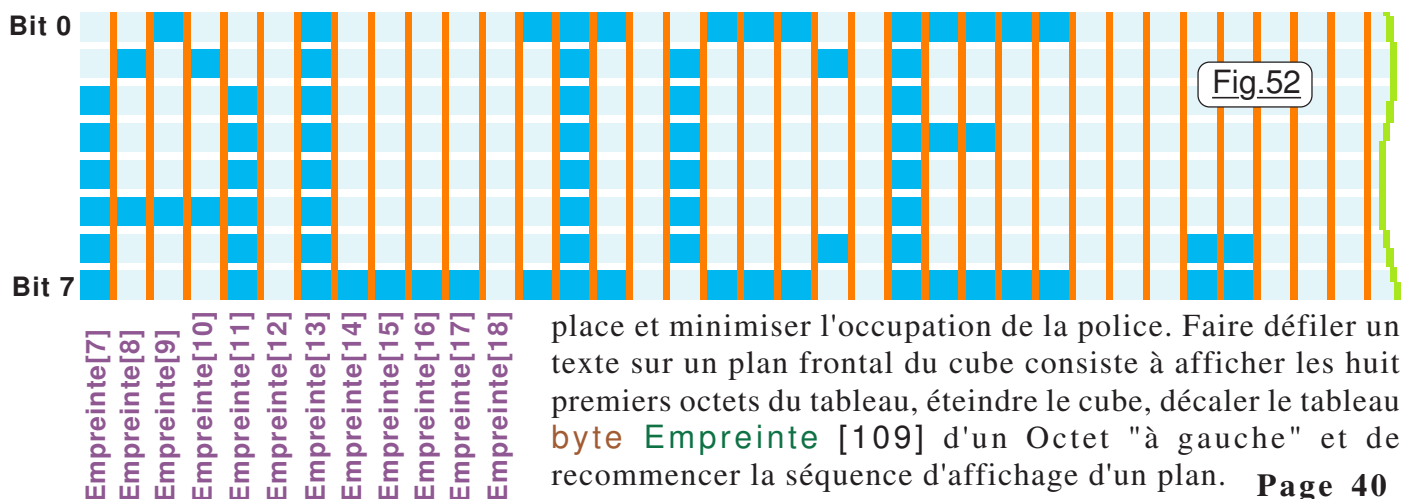
Consultez la Page 41 qui liste les thèmes ordonnés de la séquence complète d'animations.



NOTE : Que ce soit pour les matrices de textes ou celles des petits dessins spécifiques, des "zones voisines" sont repérées par tranches colorées en vert ou en bleu. Les zones peuvent concerner un caractère comme présenté sur la fiche [Les matrices graphiques : Images / Textes](#), ou les dessins proposés ci-dessus. Cet artifice de coloration facilite grandement le repérage visuel des octets représentatifs et leur codage en binaire dans `Ecrire_les_Empreintes_en_EEPROM.ino`.

➤ Étude du décalage à gauche du tableau Empreinte[64].

Mentionnées dans la fiche [Implantation logicielle des données à afficher](#), l'organisation en mémoire RAM des textes est reproduite en plus grand sur la Fig.52, les sept premiers octets du tableau `byte Empreinte` [109] ne sont pas représentés. Notez que chaque matrice de caractère est composée de cinq Octets. L'espace entre caractère n'est pas codé en EEPROM pour gagner de la



place et minimiser l'occupation de la police. Faire défiler un texte sur un plan frontal du cube consiste à afficher les huit premiers octets du tableau, éteindre le cube, décaler le tableau `byte Empreinte` [109] d'un Octet "à gauche" et de recommencer la séquence d'affichage d'un plan. **Page 40**

➤ Liste ordonnées des thèmes.

PGM	Thème
PGM_0	Compte à rebours
PGM_1	Affiche les 40 "symétriques verticaux en "direct". Affichage statique à des Y aléatoires. OISEAU VOLE ! Le_serpent() TAXI BONJOUR LES AMIS. CQ_F6AAE
PGM_2	Défilement du bas vers le haut du texte ALICE avec petit bruitage.
PGM_3	Affiche les 40 "symétriques verticaux en Inverse Haut / Bas". Déplacements en mode Fuite. Flèche périphérique tournante Houle Pyramide Nord/Sud TIC TAC TIC TAC Clef_du_bonheur
PGM_4	Affiche un cube filaire avec diagonales.
PGM_5	Affiche les 24 complets NON inversés. Déplacements en mode Approche. COUCOU ALICE * * AQUARIUM La pluie
PGM_6	Cube filaire avec demi-diagonales orientées vers le bas.
PGM_7	Affiche les 8 complet "valides en Inverse" avec Fuite si NON inversé et Approche si Inversé.
PGM_8	Affiche un cube entier avec pixels modifiés aléatoirement.
PGM_9	Affiche les 40 "symétriques verticaux Direct" NON inversé en Descente et Remontée en Inversé. Rayons_filaires() Pyramide Nord/Nord SOS
PGM_10	Construire les six faces d'un cube vide qui ensuite s'efface par tranches de la gauche vers la droite.
PGM_11	Affiche un cube filaire avec la croix centrale qui Clignote.
PGM_12	Faire Gigoter les (24 + 8) Images Complètes sur les trois axes : Image NON inversée. Pyramide Sud/Sud Le_CHIEN_CHIEN 1538 SOUDURES!!! Le bocal de BUBULE
PGM_13	Faire scintiller un cube plein pendant deux secondes.
PGM_14	Fait tourner dans le Sens Horaire les 24 + 8 images complètes NON Inversées et Fuite sur Y'Y. Pyramide Sud/Nord Le_petit_train AT AT ATCHOUM! Vive NOEL. (La bougie.)
PGM_15	Deux dièdres inclinés en vis à vis.
PGM_16	Faire Gigoter les 8 Images Complètes sur les trois axes en visualisation Inversée.
PGM_17	Affiche les 24 Images Complètes NON inversées et Aléatoirement en Descente ou en Montée. NON ALICE Sépare un cube en deux Sépare un cube en quatre
PGM_18	Deux dièdres en vis à vis de directions "réciproques".
PGM_19	Pyramide Nord/Sud enfermée puis Effacement de Droite vers Gauche.
PGM_20	Décale latéralement des images Complètes typiques. (Intègre Le_PITBULL) Diagonalise toutes les images complètes, en deux directions et alterne NON Inverse puis Inverse. Enferme la pyramide Nord/Nord Le_Paquebot BONNE ANNEE 2020.
PGM_22	Plans diagonaux avec augmentation de la rapidité d'alternance.
PGM_23	Affiche les 8 Images Complètes : Descente en NON inversé et remontée en Inversé.
PGM_24	Pyramide Sud/Sud enfermée puis Effacement de Droite vers Gauche.
PGM_25	Croissillon avec effacement rapide du bas vers le haut.
PGM_26	Ressort Vertical sur les quatre options possibles de pyramides.
PGM_27	Génération de volumes verticaux avec les images complètes "pertinentes".
PGM_28	Enferme le cube filaire avec diagonales.
PGM_29	Croissillon enfermée puis Effacement de Droite vers Gauche.
PGM_30	Génération de volumes verticaux avec les images symétriques "pertinentes".
PGM_31	Effacement Vertical, Transversal ou par Tranches d'un CUBE entier.
PGM_32	Construire Aléatoirement un cube par "barres transversales".
PGM_33	Enferme le cube filaire avec la Croix centrale.
PGM_34	Effacer Aléatoirement des Cubes pertinents.
PGM_35	Faire Gigoter des images Symétriques "pertinente" dans les trois directions.
PGM_36	Diagonalise des images symétriques.
PGM_37	Quelques volumes construits verticalement puis effacés.
PGM_38	Construire Aléatoirement un cube par PIXELs.
PGM_39	Construire une spirale_périphérique montante.
PGM_40	Enferme et efface latéralement des "volumes".
PGM_41	Faire Gigoter sur Y'Y les dernières images Symétriques "pertinente".
PGM_42	Construire un trièdre en montant puis l'effacer par tranches.
PGM_43	Construire une pyramide creuse à degrés.
PGM_44	Personne qui fait "au revoir" avec la main.
PGM_45	Cœur qui devient une souris.
PGM_46	Construit puis efface progressivement un cube plein par spirale plane.
PGM_47	Tir à l'arc sur une cible anglaise.

Chaque thème de la suite ordonnée est intégré dans un "programme artistique" nommé PGM_nn.

Un programme peut intégrer jusqu'à quatre animations spécifiques pour "rompre" la monotonie des figures géométriques successives. Les animations spécifiques sont repérées en couleur dans cette liste. En rouge sont précisées les séquences MORSE, En bleu ciel les petits dessins qui défilent de la droite vers la gauche, En bleu foncé les textes défilants et en violet les animations particulières avec texte puis dessin "mobile".

A vant de passer à l'étude des rotations planes, une petite parenthèse est souhaitable. Il n'est pas impensable, que certaines ou certains d'entre vous, un peu nostalgiques de la "télégraphie sans fil" puissent désirer intégrer des séquences personnalisées. Le codage des caractères Morse n'est pas très délicat, il suffit d'analyser le code source spécifique, et éventuellement d'aller sur Internet pour avoir la combinatoire des points et des traits. Ayant été radioamateur, je connais parfaitement le code Morse, l'ayant pratiqué durant des années. Aussi, pour vous éviter des recherches sur la toile, le petit utilitaire **CODE_MORSE_COMPLET.ino** contient la presque totalité de l'alphabet. Par de simples Copier / Coller il sera aisé d'aller chercher les lettres et les chiffres dont vous avez besoin. Enfin, dans la liste des "Sketchs" disponible figure **Provisoire.ino** qui n'intègre que les définitions et les séquences les plus importantes. Dépouillé de toutes les procédures "artistique", cet outil facilite la mise au point d'un thème particulier en évitant de ce perdre sur un listage de six cent lignes, en réduisant le temps nécessaire pour téléverser ce code objet. Ce n'est qu'un petit module de servitude mis à votre disposition. En particulier, figure en tête de la séquence des initialisations la remarque `//Serial.begin(115200); //Serial.println();Serial.print("----");` prévue pour puiser rapidement des instructions servant à visualiser sur la ligne USB du Moniteur les valeurs de divers paramètres pertinents facilitant la mise au point.

Abordons maintenant le dernier volet de cette saga informatique qui donne le tournis !

17) Les rotations : Pas si simple que ça de tourner en rond !

M athématiquement, faire tourner un objet géométrique n'est pas fondamentalement compliqué. Tout bachelier sait qu'avec la trigonométrie dans le plan ou le produit vectoriel dans l'espace, on peut facilement s'en sortir. Avec les techniques de traitement des matrices, un autre outil tout aussi maîtrisé est spécialement conçu par les mathématiciens pour qui l'outil matriciel est d'une évidence triviale. ***OK, j'avoue que dans ce préambule je frime outrageusement pour vous impressionner. N'en déduisez surtout pas que j'aligne aisément les équations de l'espace, bien au contraire.*** Toutefois, en informatique l'approche est différente. En effet, nos données ne sont pas des points rangés dans un espace cartésien, mais des Octets qui "se suivent en ligne" dans la mémoire du microcontrôleur. Trouver des traitements les moins boulimiques en code Objet n'est pas d'une évidence biblique, tout au moins pour votre narrateur quand il a désiré faire tourner des plans frontaux de 90° dans le sens horaire ou antihoraire. Pourtant j'avais imaginé que ce serait facile ! Arriver à scribouiller un code raisonnable qui fonctionne m'a pris au final pas mal de temps.

➤ La Page 42 résume le traitement relatif à une rotation horaire.

D étailler le processus impose de le faire sur un exemple concret. Celui utilisé dans ce chapitre n'est pas du tout un thème visuel intégré dans le programme d'exploitation. Il s'agit d'une image particulière utilisée uniquement pour valider les lignes de code. Elle est conçue pour faciliter le repérage des divers PIXELs dans la matrice et leurs "déplacements" durant le traitement. La relation entre l'image et les éléments de la "matrice numérique" est ainsi plus immédiat.

- En Fig.53 est représentée à droite l'image en bleu clair telle que nous la désirons, c'est à dire ce qu'affichera le cube sur l'un des plans frontaux accueillant ce dessin lumineux.
- En Fig.53 à gauche précise "l'étalement latéral" des divers BITS constituant les huit octets du thème. Un "0" logique correspondra à un point non allumé, et un "1" devra illuminer la LED concernée.
- Nous savons que pour visualiser un plan vertical, il faut effectuer un balayage de multiplexage. Compte tenu de l'agencement des procédures qui effectuent ce travail, l'image doit être transportée en RAM dans le tableau **Empreinte** [109]. Le balayage dynamique impose de disposer les Octets en inversant Haut et Bas ainsi que Gauche et Droite. C'est ce que montre la Fig.54 avec à gauche la présentation géométrique de la MATRICE des OCTETS. Dans cette représentation sont repérés les ordres des BITS avec ***à gauche le poids fort*** dans l'Octet et ***à droite le poids faible***.
- Sur la Fig.55 à droite le dessin bleu traduit l'image telle que nous la désirons visualisée sur le cube quand elle aura tourné de 90° dans le sens horaire. Le dessin en couleurs pour sa part présente l'inversion que doit subir latéralement et en hauteur cette image pour se voir multiplexée correctement. La grille numérique à gauche indique comment les BITS doivent "se déplacer" dans la MATRICE pour occuper la bonne position. Chaque individu des points allumé conserve la couleur qu'il avait en Fig.54 pour repérer facilement le saut virtuel qu'il doit effectuer.

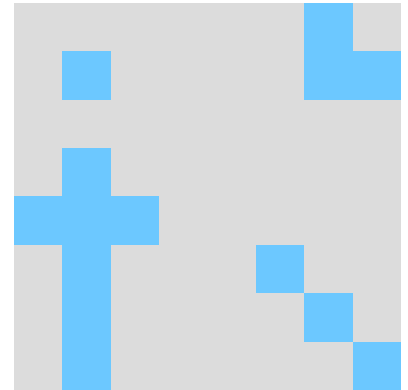
Rotation à 90° dans le sens Horaire.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
0	0	0	0	0	0	1	0	Premier octet (haut)
0	1	0	0	0	0	1	1	
0	0	0	0	0	0	0	0	
0	1	0	0	0	0	0	0	
1	1	1	0	0	0	0	0	
0	1	0	0	0	1	0	0	
0	1	0	0	0	0	1	0	
0	1	0	0	0	0	0	1	Dernier octet (Bas)
Gauche				Droite				

Codage direct en EEPROM

Fig.53

Affichage Désiré



Droite	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Gauche	
63	62	61	60	59	58	57	56			Premier octet (haut)
45	54	53	52	51	50	49	48			
47	46	45	44	43	42	41	40			
39	38	37	36	35	34	33	32			
31	30	29	28	27	26	25	24			
23	22	21	20	19	18	17	16			
15	14	13	12	11	10	09	08			
07	06	05	04	03	02	01	00			Dernier octet (Bas)

Fig.54

Empreinte image copiée en RAM par
Transfere_un_plan_en_RAM(PTR).

Engendre : L'inversion Gauche/Droite.
Engendre : L'inversion Haut/Bas.

Tableau Empreinte

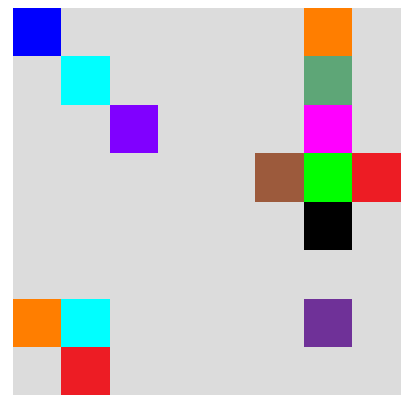
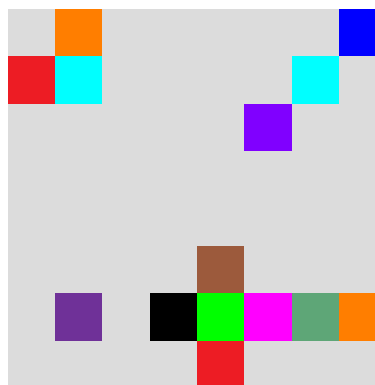


Tableau Original

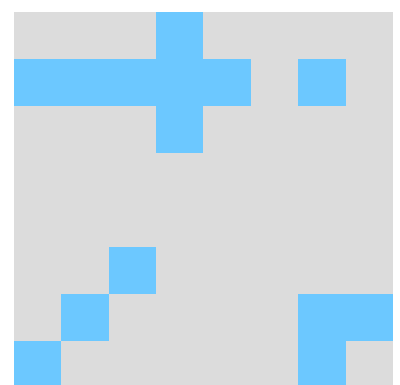
Droite	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Gauche	
63	62	61	60	59	58	57	56			Haut
45	54	53	52	51	50	49	48			
47	46	45	44	43	42	41	40			
39	38	37	36	35	34	33	32			
31	30	29	28	27	26	25	24			
23	22	21	20	19	18	17	16			
15	14	13	12	11	10	09	08			
07	06	05	04	03	02	01	00			Bas

Fig.55

Empreinte traitée



Affichage Désiré



Déplacement des BIT initiaux

07	15	23	31	39	47	55	63	Premier octet (haut)
06	14	22	30	38	46	54	62	
05	13	21	29	37	45	53	61	
04	12	20	28	36	44	52	60	
03	11	19	27	35	43	51	59	
02	10	18	26	34	42	50	58	
01	09	17	25	33	41	49	57	
00	08	16	24	32	40	48	56	Dernier octet (Bas)

Fig.56

63 va en 56

57 va en 08

54 va en 49

49 va en 09

45 va en 42

41 va en 10

34 va en 19

...

Comme on désire enchaîner un nombre quelconque de rotations, l'enchevêtrement déterminé sur la Fig.55 ne se fait pas dans **Empreinte** [109] qui sert à afficher, mais dans le tableau **Original** [109] chargé de conserver une image ainsi "réutilisable".

La procédure de rotation ne modifie que les huit premiers octets dans **Empreinte** [1].

NOTE : Comme c'est précisé dans le programme source, créer une procédure unique faisant tourner de 90° dans le sens Antihoraire conduit à une taille de programme bien plus grande vu le nombre de test à effectuer. Cette procédure a finalement été abandonnée. Nous arrivons au terme de notre cheminement, le précieux cube attend avec impatience l'arrivée de Noël ...

18) Les "loupés".

Aucun projet, qu'il soit industriel ou de loisir ne saurait aboutir à la perfection absolue. Entre les désirs initiaux, ce qui était envisagé et ce qui résulte de compromis inévitables, s'insinuera forcément des divergences. Le CUBE 3D n'échappe pas à ce principe non démontré mais qui frise l'absolu. Il a fallu composer avec les réalités matérielles, et l'on peut noter un certain nombre de "regrets" :

- Les deux LEDs situés à l'arrière conviennent parfaitement en usage normal. Toutefois, en programmation quand les deux B.P. sont remplacés par des boutons externes, il serait bien commode d'avoir ces deux LEDs en façade. Il en est de même pour les deux boutons poussoir.
Donc tenez compte de cette information pour concevoir un modèle personnel plus convivial.
- **Comme précisé dans le didacticiel, la base du multiplexage utilise un KIT du commerce. Ce dernier est livré avec des LEDs cristal de 3mm de diamètre. C'EST INSUFFISANT car les fils rigides de liaison masquent de façon importante les sources lumineuses. Donc approvisionnez des LEDs de 5mm de diamètre et si possible avec des corps diffusants au lieu de cristal.**
- Les corps des LEDs utilisées sont de type cristal, c'est à dire parfaitement translucides. Dans la pratique c'est un gros inconvénient car la lumière émise par la LED qui est derrière traverse en partie et illumine de façon parasite celle qui est devant. Si c'était à refaire, chaque fois qu'un plan de LEDs est terminé, avant de l'assembler sur le cube, toutes les encapsulations seraient peintes en noir mat sur la face arrière coté broches.
- Lorsque le moment d'assembler le circuit imprimé est venu, je ne disposais que d'un transducteur sonore actif, c'est à dire générant sa tonalité. Il serait plus amusant de le remplacer par un transducteur passif et ainsi pouvoir générer des petites "musiquettes". Ce regret va certainement faire partie d'une prochaine modification matérielle et logicielle sur le prototype.

Peu importe ces petites tracasseries. Ne boudons pas pour autant notre plaisir. Cube 3D est bien vivant et le voir s'illuminer reste vraiment étonnant. Quand je le présente à des amis, ils sont chaque fois un tantinet dubitatifs, et force est de constater, que durant le repas, certains regards se tournent régulièrement vers le petit objet espiègle. Tout compte fait, les humains ne sont que de grands enfants qui ne demandent qu'à s'émerveiller, et parfois il suffit de si peu de chose ...

Chères lectrices, chers lecteurs, cette (trop) longue saga arrive à son terme. Tout à une fin, mis à part l'Univers, et arrive forcément un moment où il faut raisonnablement considérer que "le travail" est terminé.

Je souhaite intensément que certaines et certains oseront s'engager dans la réalisation d'un clone, je ne doute pas de leur réussite. Surtout, je vous souhaite à toutes et à tous de trouver dans ces lignes le plaisir de la découverte. Si d'aventure vous engagez vos heures de liberté dans une telle réalisation et que vous rencontrez une difficulté, les amis du forum pourront probablement vous aider. Dans le pire des cas, vous pouvez me contacter sur : michel.droui@laposte.net et dans les limites de mon temps de libre, c'est avec grand plaisir que je tenterai de vous dépanner. Je vous souhaite à toutes et à tous agréable lecture et ...

Joyeux Noël !

Chaleureusement : Nulentout.