

Électronique pour manipulateur double contacts

F6HQP le 6/05/2017

2eme partie : programmation

A- Configuration nécessaire :

- un PC avec le logiciel « PICAXE editor 6 » installé > sur le site *A4-Technologie*
- un câble de liaison série-série ou USB-série. Pour ce dernier, on en trouve un chez PICAXE (ref AXE027), il est assez cher mais le driver est dispo. On peut prendre d'autres modèles mais certains ne fonctionnent pas car la sortie série n'est pas dans les normes. Attention à ceux que l'on trouve sur le net à très bas prix.
- faire un raccord 3 fils qui va sur la platine . voir le schéma.

Lancement : connecter le cordon USB-série > lancer le logiciel « PICAXE editor 6 » > relier le cordon au montage > mettre le montage sous tension > cliquez sur *vérifier le type de PICAXE connecté* .

B- calcul de la vitesses de manipulation

la durée d'un trait vaut 3 points,
l'espace entre les lettres est de 3 points
l'espace entre les mots est de 5 points,

la cadence de manipulation lors de l'examen était de 20 mots par minute, c'est à dire 20 fois le mot PARIS en 60 seconde.

Donc le temps pour faire PARIS est de 48 points , soit pour 20 mots $48 \times 20 = 960$ points

960 points en 60s (1mn) fait une durée du **point** de **62 ms pour une vitesse de 20mots/mn**

La vitesse va être réglée par la lecture de la tension sur un potentiomètre branché entre la masse 0V et le +5V. Une commande va permettre de lire cette tension et de la convertir en un nombre compris entre 0 (pour 0V) et 255 (pour 5V) . La conversion analogique /digitale se faisant ici sur 1 octet.

Si le potentiomètre est au milieu de sa course, on a une tension de 2,5V soit 127. Si on utilise directement cette valeur en ms, la vitesse sera d'environ 10 mots/mn ce qui est très lent. Il suffit de diviser par deux ce nombre pour s'approcher des 20mots/mn et ainsi d'avoir une plage de réglage de vitesse plus étalée.

La vitesse fonctionne sur le principe suivant :

- on lit la valeur de la tension sur le potentiomètre de 0V à 5V
- on convertit en digital de 0 à 255 que l'on stocke en mémoire B
- on divise B par 2 ce qui nous donne la durée d'un point en ms que l'on stocke en mémoire A
- on multiplie A par 3 ce qui nous donne la durée d'un trait en ms que l'on stocke en mémoire C

Il suffit après d'utiliser les mémoires A, B et C quand on a besoin de la durée d'un point, de deux points ou d'un trait (3 points)

C- Programme sans mémoire

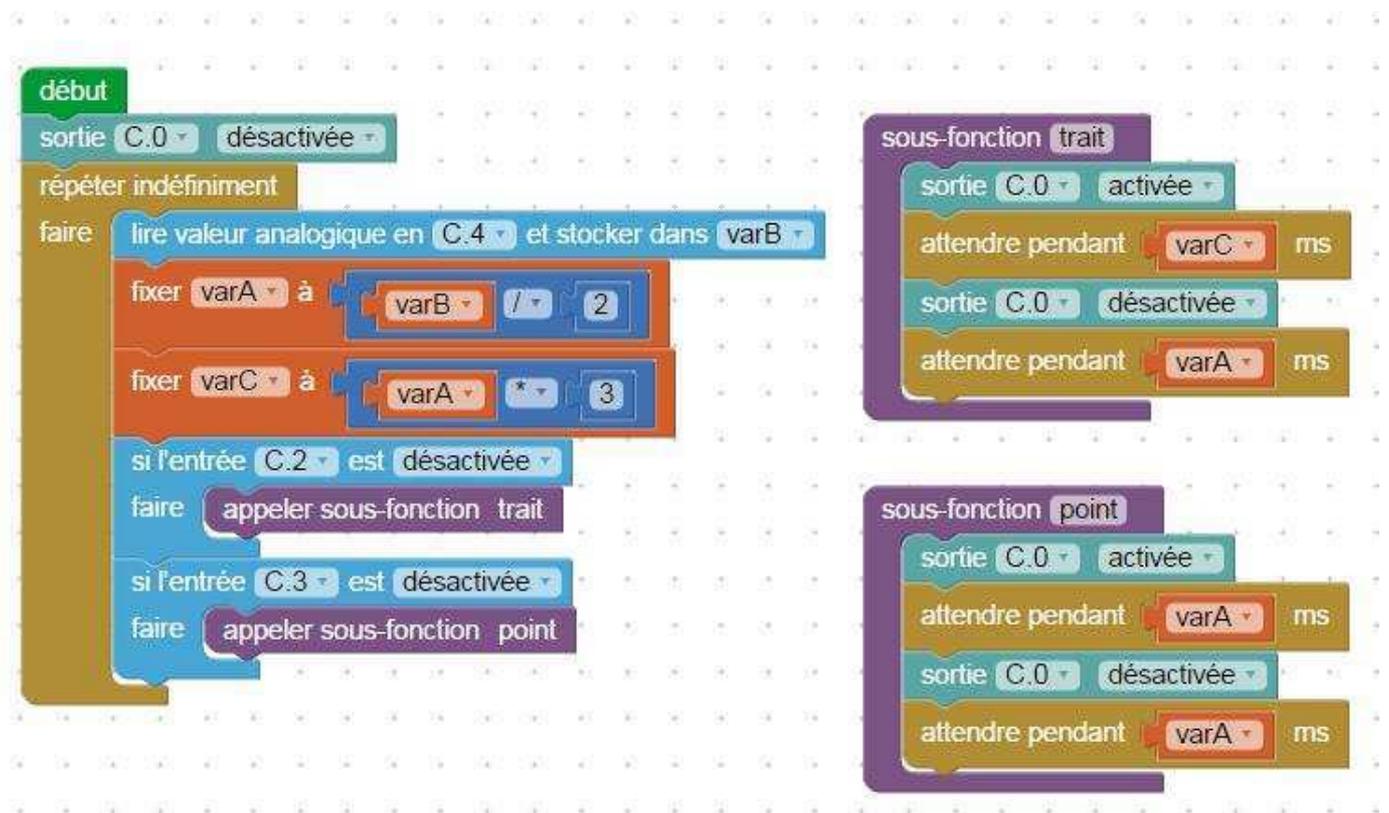
Rappel : le cœur du système est un Pic du type 08M2+ comportant 6 entrées/sorties utilisées de la façon suivante.

- 2 pour la communication avec le PC pour la programmation : **C0** et **C5**
- 1 en sortie : encore **C0**
- 2 en entrées analogiques : **C1** (choix des mémoires) et **C4** (réglage vitesse)
- 2 en entrées digitales : **C2** et **C3** pour le manipulateur

Le programme est très simple :

- > par sécurité , on force la sortie C0 à la masse
- > puis on lit la valeur du potentiomètre de vitesse
- > calcul des valeurs A, B et C
- > si C2 est à la masse , appel de la sous fonction trait
 - > sortie C0 active pendant le temps C (3 points)
 - > sortie C0 désactivée du temps A (1 point)
- > si C3 est à la masse , appel de la sous fonction point
 - > sortie C0 active pendant le temps A (1 point)
 - > sortie C0 désactivée du temps A (1 point)

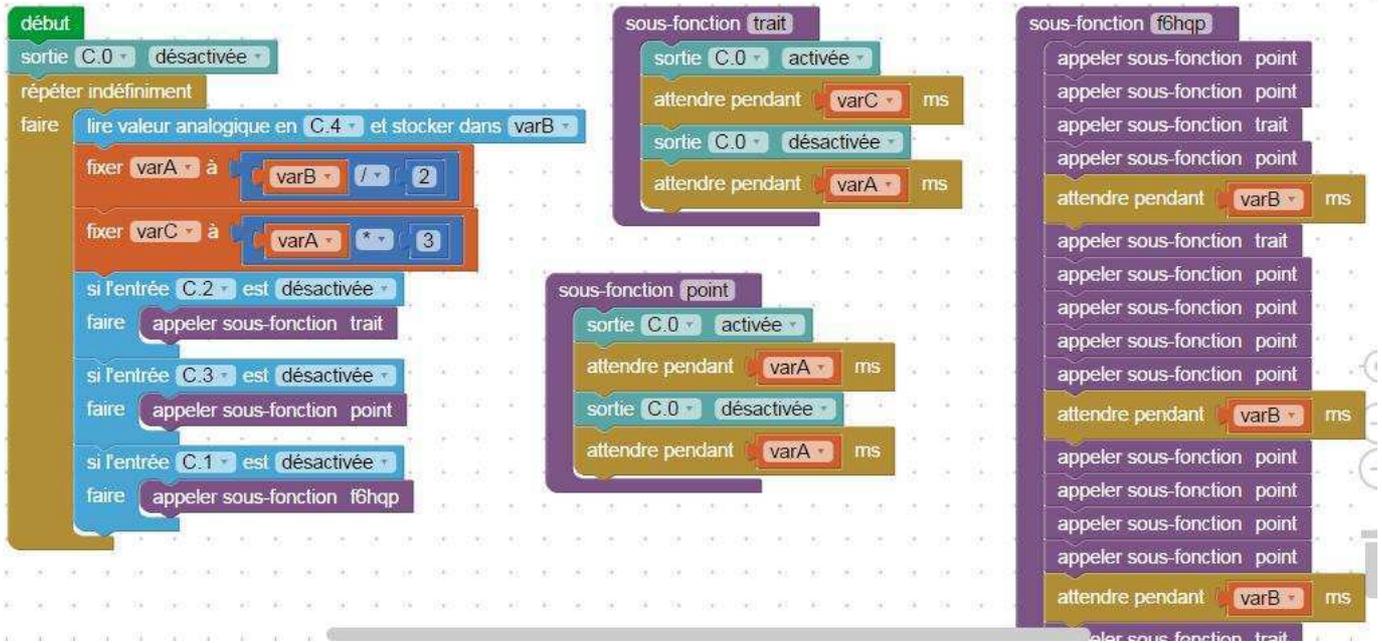
à noter que si l'on appuie sur les deux palettes du manip on obtient des ti-ta-ti-ta-ti-ta ...



D- Programme avec 1 mémoire

on ajoute une condition supplémentaire :

- > si C1 est à la masse , appel de la sous fonction f6hqp
- dans la sous fonction, il suffit de mettre des points et des traits à la suite.
 Ne pas oublier qu'il y a déjà une attente d'une valeur d'un point quand on fait un trait ou un point donc si vous mettez ensuite une attente de valeur B vous aurez l'équivalent de 3 points, si vous utilisez C vous en aurez 4 !



E- Programme avec 7 mémoires

Là, c'est un petit peu plus compliqué.

Il faut sur une seule entrée mettre 7 boutons poussoirs. On va donc utiliser l'entrée C1 en mode analogique, comme pour la vitesse, mais en mettant des résistances fixes, un appui sur un poussoir donnera une tension qui sera analysée. Si aucun poussoir n'est appuyé, R5 met C1 à +5V, si on appuie sur un poussoir, une tension intermédiaire sera issue de la chaîne de résistances R11 à R17, et avec bp1 C1 est à 0V. Ne pas oublier R5 pour le calcul de la tension.

Il faut convertir ces tensions en digital mais les résistances étant à 5 % de tolérance, l'analyse se fera dans des fourchettes mini/maxi.

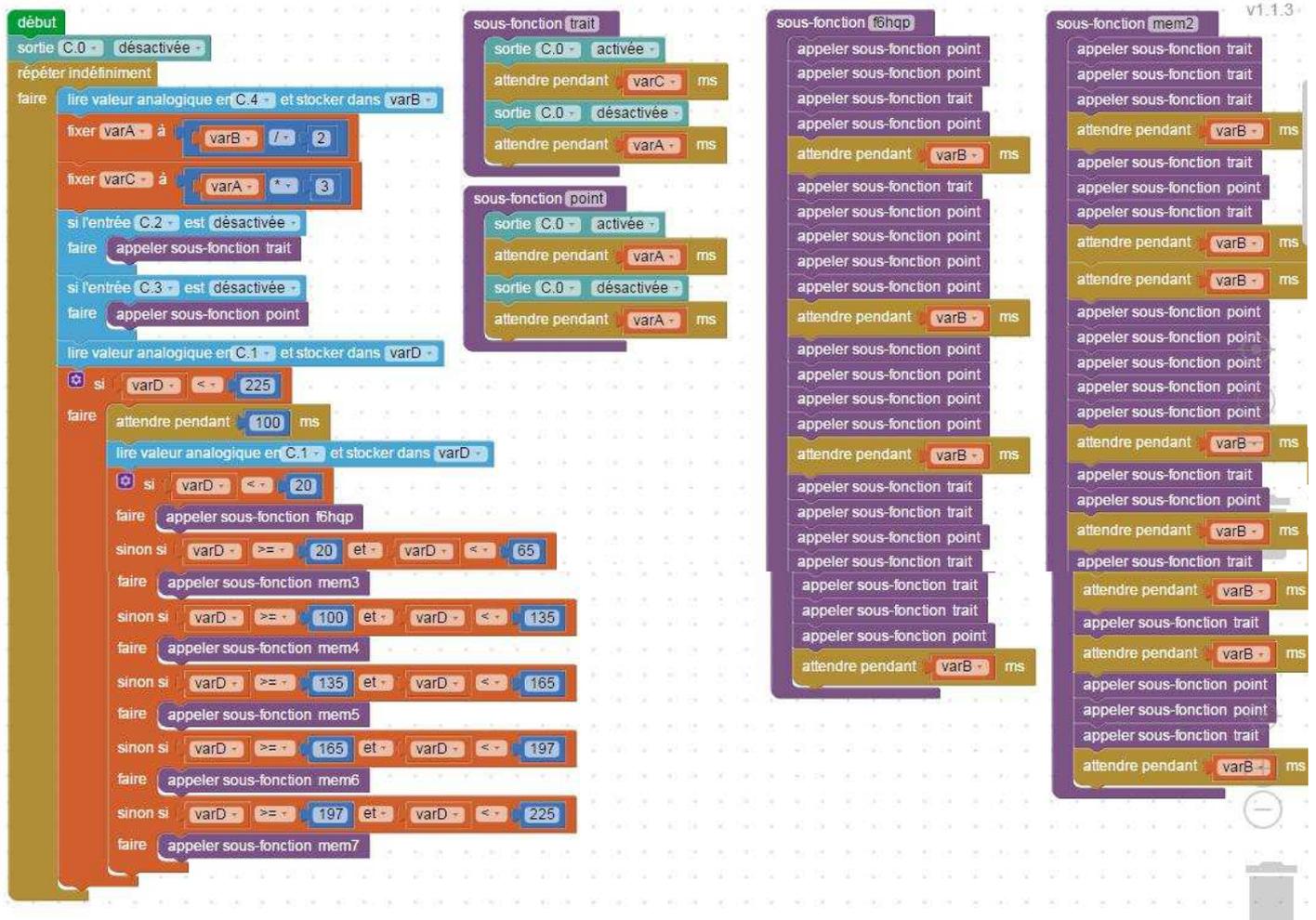
| | bp1 | bp2 | bp3 | bp4 | bp5 | bp6 | bp7 | |
|--------------|------|-------|--------|---------|---------|---------|---------|---------|
| Tension Volt | 0 | 0,88 | 1,64 | 2,33 | 2,96 | 3,57 | 4,18 | 5 |
| digital | 0 | 45 | 83 | 119 | 151 | 182 | 213 | 255 |
| Mini/maxi | 0/20 | 20/65 | 65/100 | 100/135 | 135/165 | 165/197 | 197/225 | 225/255 |

Un autre problème c'est posé lors des essais : parfois la mémoire demandée n'était pas la bonne !! cela était dû à la constante de temps du condensateur C5 et des résistances environnantes, et peut être aussi aux rebonds des poussoirs.

On va donc faire la lecture de la tension en deux temps :

1- si on détecte une tension différente de 5V, donc un nombre inférieur à 255, c'est que l'on demande un message en mémoire

2- dans ce cas, on attend 100ms et on relit la tension. Le temps de 100ms est suffisant pour que le condensateur soit correctement chargé et la tension stabilisée (on doit pouvoir mettre moins) On peut ensuite analyser cette tension pour appeler le bon message



à titre d'exemple, dans mes mémoires j'ai mis :

F6HQP

OK 5NN TU

CHAMBERY

MARC

CQ CQ DE F6HQP F6HQP +K (appeler deux fois la sous fonction f6hqp pour mettre F6HQP)

TEST TEST F6HQP F6HQP 73 K

RST 5NN73 73 K

cela utilise environ la moitié de la mémoire avec le programme, donc il y a encore de la place.