

FABRICATION D'UN ADAPTATEUR SOURIS PS2 & JOYSTICK PLAYSTATION POUR MSX

Cette réalisation n'est que la traduction en français et en images de la version originale qui est en russe.

Cette interface est complètement autonome sur un MSX. Pas besoin de programmeur sous PC. Le logiciel utilisé pour la programmation est fonctionnel à 100% sur un MSX grâce à son interrupteur (SW1) ou à un cavalier à deux positions. J'ai choisi la deuxième solution pour la programmation, pour gagner de la place sur le circuit et surtout pour ne rien voir dépasser du boîtier.

Pas d'alimentation externe. L'alimentation est récupérée sur le port joystick du MSX.

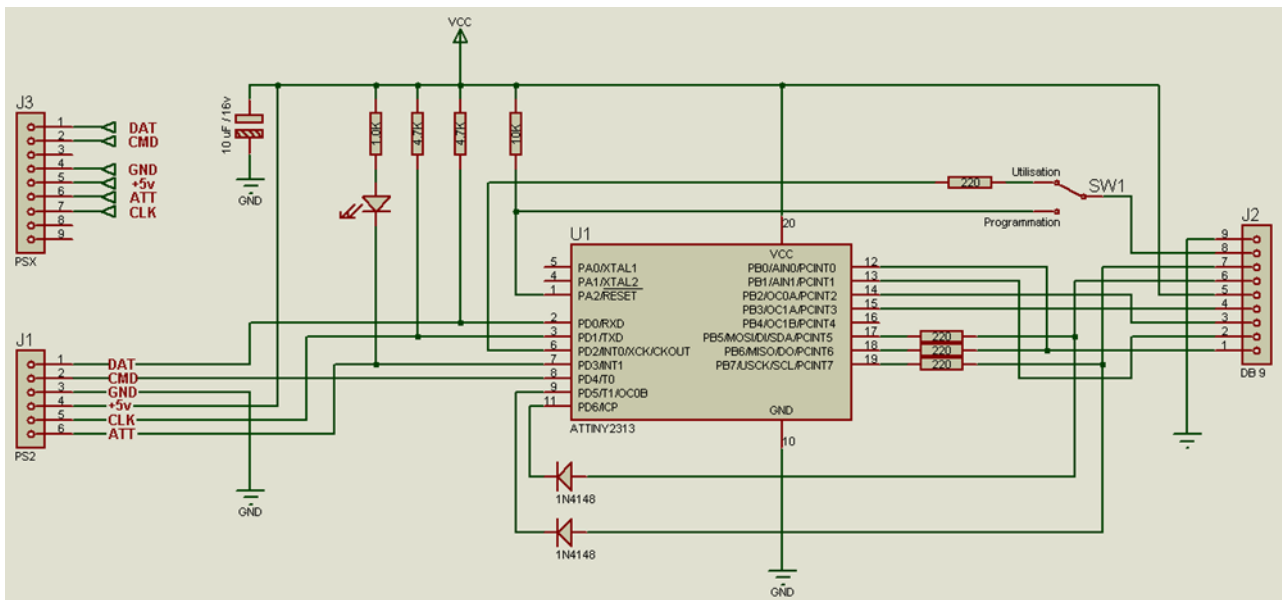
Le circuit utilisé est un ATTINY2313 de chez ATMEL. Là encore, pas besoin de quartz, il a sa propre fréquence programmable en interne. Là, en l'occurrence, il sera programmé sur 8 MHz.

Une LED permet de visualiser l'état de l'adaptateur.

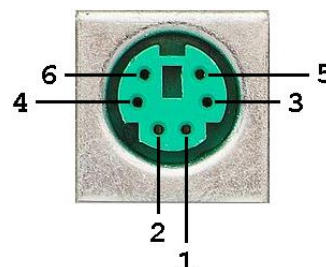
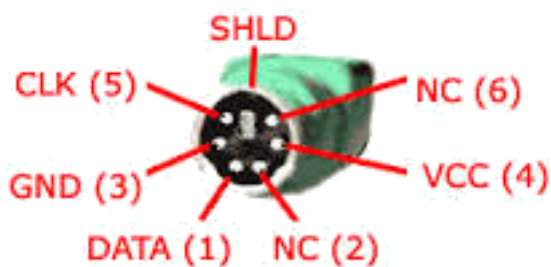
Schéma de principe de l'adaptateur :

Les deux diodes 1N4148 peuvent être remplacé par de la 1N914.

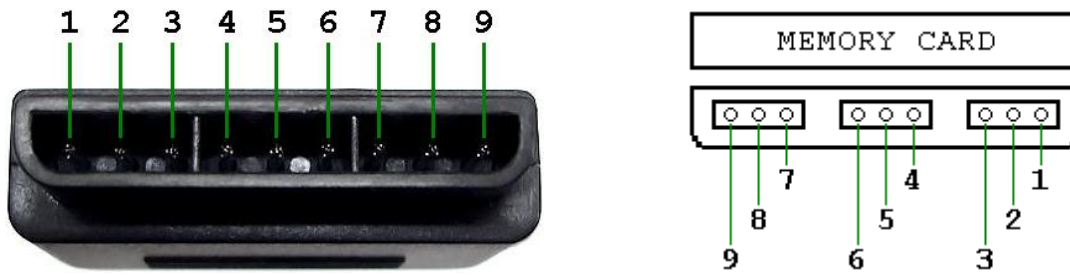
Vous pouvez aussi ne pas mettre le LED et la résistance de 1 kilo pour gagner de la place sur le circuit.



Brochage du connecteur PS2 mâle / femelle :



Brochage du connecteur PLAYSTATION mâle / femelle :



Exemple de conception d'un circuit sur une plaque veroboard :

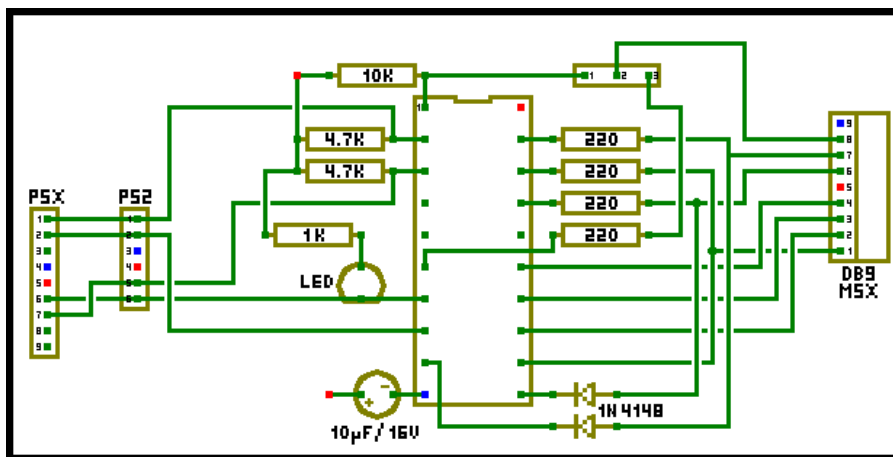
Relier les points rouge (VCC) ensemble et les points bleu (GND) ensemble.

Le connecteur PSX correspond à la manette PlayStation, le PS2 à la souris PS2 et le DB9 MSX au port joystick du MSX.

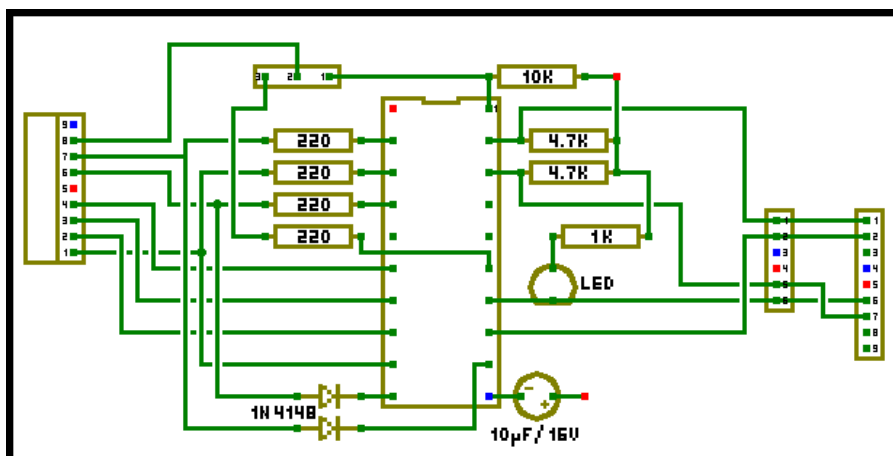
On peut voir que sur le brochage du connecteur PS2 (Image du dessus), les broches 2 et 6 ne sont pas utilisées (NC). Sur le circuit ci-dessus, elles le sont. Cela permet de faire un cordon avec une fiche PS2 mâle d'un côté et une prise joystick PlayStation femelle de l'autre.

Résultat : Plus besoin du connecteur PSX sur le circuit. La manette PlayStation ce branchera directement sur la prise femelle du connecteur PS2 à l'aide de ce cordon.

Côté composants



Côté soudures



Programmation de l'Attiny2313 :

Pour programmer l'ATTINY2313, il faut dans un premiers temps, basculer le switch en mode programmation. Cavalier en 1-2

Lancer la commande **PRG2313T.COM [fichier.hex]**
MSXMS20.HEX pour utiliser une souris PS2.
MSXJOY.HEX pour utiliser une manette PlayStation.

A la programmation, un écran identique aux deux images ci-dessous vous donnera des infos concernant la programmation.

On peut voir par exemple :

Que l'horloge interne est réglée sur 8 MHz.

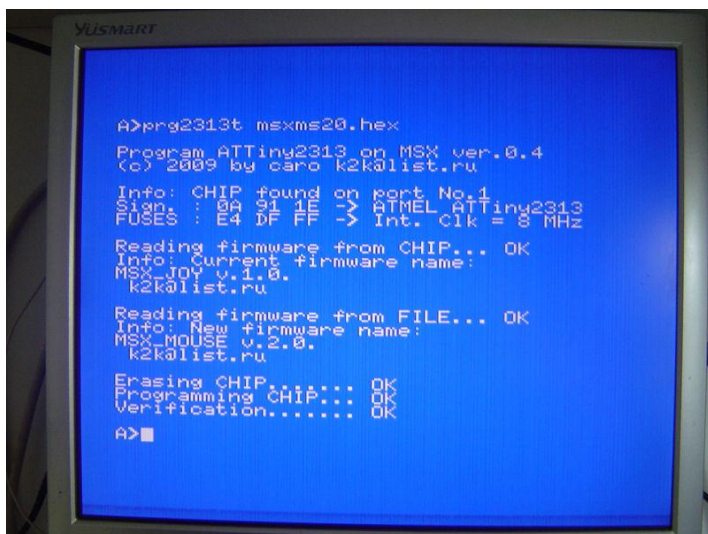
Sur la première image, que le firmware présent dans le chip est celui du joystick Playstation et que le nouveau sera celui de la souris PS2.

Sur la deuxième image, que le firmware présent dans le chip est celui de la souris PS2 et que le nouveau sera celui du joystick PlayStation.

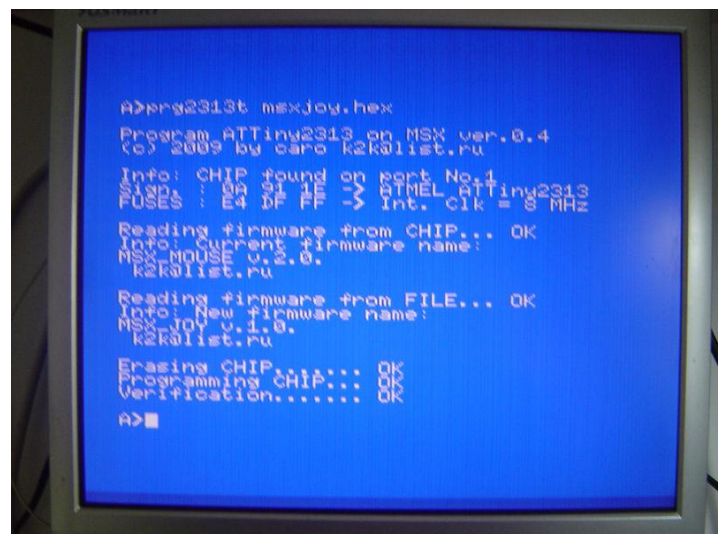
Une fois la programmation terminée, remettre le switch en mode lecture. Cavalier en 2-3

[Cliquer ici pour récupérer PRG2313T.COM et les fichiers *.HEX](#)

Programmation pour utiliser la souris PS2 :



Programmation pour utiliser la manette PlayStation :



Si vous programmez avec le même firmware que celui qui est présent dans L'ATTINY2313, le logiciel affichera qu'il n'est pas nécessaire de faire une programmation, voir l'image ci-dessous.

Programmation du même firmware présent dans l'ATTINY2313 :

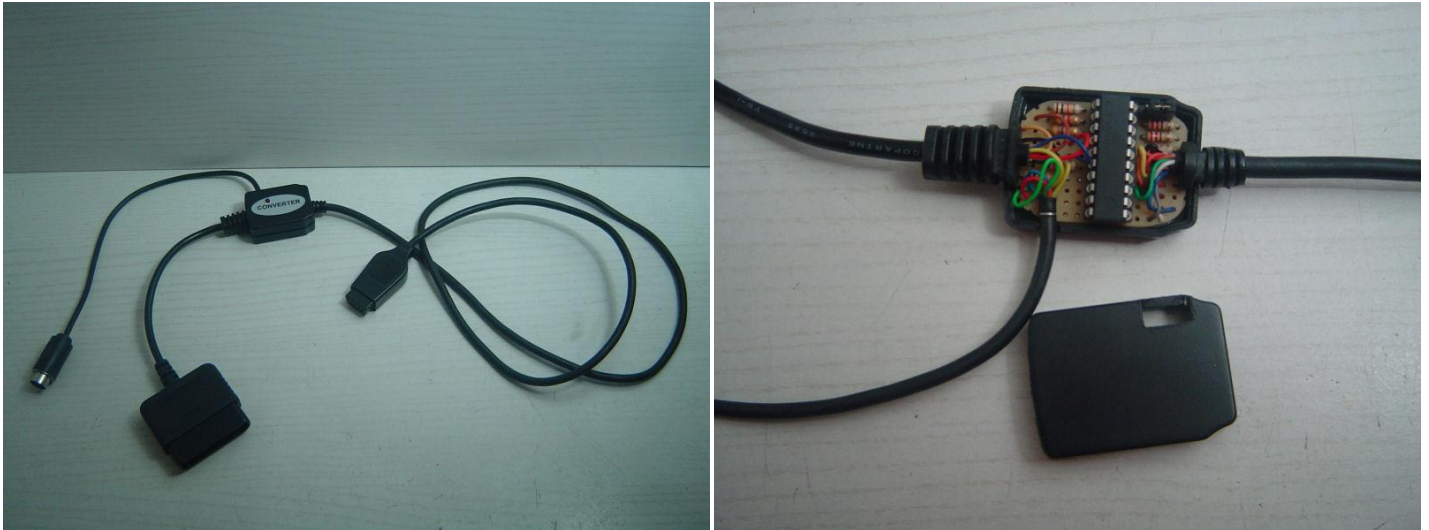


Etat de la LED :

Une fois l'adaptateur programmé avec le firmware voulu et connecter au MSX, si il n'y a pas de source (Souris PS2 ou manette PlayStation) branchée en sortie, la LED se mettra à clignoter. Dès que vous branchez une source en sortie, la LED s'éteindra.

Photos de l'adaptateur terminé :

On peut apercevoir l'encoche réalisée dans le boitier pour la sélection du cavalier qui se trouve en haut, a droite.
La LED est soudée côté soudure.



Et des vidéos du proto ici:

Utilisation de la souris PS2 : <https://www.youtube.com/watch?v=bUGaRmq-FN4>

Utilisation de la manette Playstation : <https://www.youtube.com/watch?v=q-g0yOKvrfo>

L'état de la LED : <https://www.youtube.com/watch?v=AKG0cRA6ZAw>

Projet réalisé par MetalGear2

