

SIMULATEUR ALTAIR 8800

simulation matérielle d'un ordinateur ancien



Figure 1. Mon simulateur Altair 8800.

David Hansel (États-Unis)

Vous avez toujours voulu jouer avec l'ordinateur qui a ouvert l'ère du PC, mais vous n'avez pas les moyens de vous en offrir un ? Alors, construisez-le !

Mais c'est quoi, un Altair 8800 ? Si vous ne vous en souvenez pas (parce que vous n'êtes pas si vieux), jetez un coup d'œil à ce que Wikipedia nous en dit : « L'Altair 8800 est un micro-ordinateur conçu en 1974 par MITS et basé sur le microprocesseur Intel 8080... L'Altair est largement reconnu comme l'étincelle qui a déclenché la révolution de la micro-informatique en tant que premier ordinateur personnel à succès commercial. » [1] Donc, historiquement, c'est un système informatique vraiment important.

Si vous souhaitez construire un système qui ressemble et se comporte comme l'Altair 8800 vieux de 48 ans, mais que l'idée de partir à la chasse aux différentes pièces ne vous tente pas, vous pouvez acheter un kit Altair-Duino [2], qui comprend tous les composants, un circuit imprimé, un boîtier et un Arduino Due préprogrammé. Dans l'article *Arduino Project Hub* [3], vous trouverez des liens vers les œuvres d'autres fans.

Les dessous de l'histoire

Pendant longtemps, j'ai pensé qu'il serait sympathique de pouvoir jouer avec un ordinateur Altair 8800. Mais les anciens systèmes Altair en état de marche sont rares et donc chers, coûtant facilement 2 000 à 3 000 € quand on en trouve. Il existe d'autres options, comme celle d'altairclone.com, mais à 600 € quand même, trop cher pour moi pour un ordinateur qui, bien que très sympathique, reste d'une utilité limitée. Heureusement, Mike Douglas, le créateur de ce clone, a mis à la disposition de la communauté tous les documents et logiciels d'époque qu'il a dénichés et utilisés pour créer le clone. Grâce au travail de Mike, il existe une mine d'informations facilement accessibles sur le fonctionnement interne de l'Altair et de ses périphériques les plus populaires. Le moment venu, j'ai examiné les spécifications de l'Arduino Mega 2560 pour savoir s'il avait suffisamment de broches d'E/S pour connecter simplement les LED et les interrupteurs du panneau avant de l'Altair et si je pouvais programmer mon propre émulateur. Il s'est avéré que l'Arduino Mega possède exactement le nombre de broches d'E/S nécessaire et que je n'avais plus qu'à me mettre au travail. L'utilisation de l'Arduino Mega comme moteur du simulateur a bien

fonctionné et a été facile à mettre en œuvre, mais l'émulation ne fonctionne qu'à environ 25 % de la vitesse de l'Altair et ne peut fournir que 6 ko de mémoire émulée (une quantité considérable dans les années 1970). La capacité de la mémoire permanente (pour stocker les programmes et les données créées dans le simulateur) est également limitée, car l'EEPROM du Mega ne contient que 4 ko.

L'Arduino Due dispose de suffisamment de mémoire pour fournir 64 ko de RAM émulée et est beaucoup plus rapide que le Mega. En outre, l'Arduino Due peut stocker des données en mémoire flash au moment de l'exécution. Cela permet d'utiliser comme mémoire permanente toute la mémoire flash de 512 ko qui n'est pas utilisée par le simulateur lui-même. Avec le Due, j'ai pu créer un simulateur Altair 8800 qui fonctionne à peu près à la vitesse de l'original, fournit 64 ko de RAM émulée, contient de nombreux logiciels Altair, et peut encore fournir 32 ko de mémoire semi-permanente pour charger et stocker des programmes et des données dans l'émulateur.

Objectifs, exemples, et plus encore

Travailler avec le simulateur devait me donner la sensation la plus proche possible du « vrai » Altair 8800. Il fallait donc que les lumières du panneau avant imitent autant que possible le comportement original. L'un des critères était qu'il soit possible de jouer au jeu « Kill-the-bit » sur le panneau avant. La **figure 1** et la vidéo YouTube « *Arduino Altair 8800 Simulator - Entering and Playing Kill-the-Bit* » [4] montrent que j'y ai réussi.

Il s'est avéré que la simulation est si fidèle à l'original que même la démo musicale originale de l'Altair 8800 [5] fonctionne - à titre de comparaison, voici ma version [6]. Ce type de production musicale nécessitait une radio AM pour capter les interférences électromagnétiques générées par les circuits de l'Altair.

En 1977, Processor Technology a sorti une petite carte d'extension (juste quelques condensateurs et résistances) avec un logiciel d'accompagnement qui a transformé l'Altair en un système musical respectable (pour l'époque). Les mêmes ajouts peuvent être faits au simulateur (voir la documentation) pour qu'il joue les airs créés pour le système de musique de l'époque. Cette vidéo YouTube [7] montre mon clone jouant l'*Ouverture de Guillaume Tell*.

Une autre extension historiquement importante pour l'Altair était la carte graphique Cromemco Dazzler. À l'aide d'une extension logicielle ou matérielle, le simulateur peut également émuler cette carte [8], comme le montre la **figure 2**. Une autre extension logicielle/matérielle [9] apporte l'émulation de la carte graphique VDM-1 (**figure 3**) de Processor Technology.

Comme je ne possède pas d'Altair original, j'ai dû obtenir toutes les informations nécessaires à partir de documents et de vidéos (voir les remerciements dans [3]). Il peut y avoir quelques différences mineures, mais dans l'ensemble je pense qu'il reproduit assez bien le comportement original (voir l'encadré **Points forts**). Une différence bien connue (et intentionnelle) est le voyant d'état du HLDA qui indique dans l'original que l'unité centrale a été arrêtée par un dispositif externe, fonction absente dans le simulateur, où il signale qu'un fichier (série/enregistrement/lecture de bande) est actuellement ouvert.

Lorsque vous utilisez l'Arduino Due, veuillez noter que toutes les données capturées ou stockées dans le simulateur seront effacées si vous chargez une nouvelle version du croquis dans le Due, car les

données sont sauvegardées dans la mémoire flash, qui est effacée lorsqu'un nouveau croquis est chargé (le Due ne dispose pas d'une EEPROM pour un stockage permanent). Si une carte SD est connectée au Due, les données sont enregistrées sur cette carte et donc conservées lorsqu'un nouveau croquis est téléchargé.

La documentation originale de l'Altair (facile à trouver via Google) contient toutes les informations nécessaires à l'utilisation des interrupteurs du panneau avant. Toutefois, le simulateur contient des fonctions supplémentaires et des logiciels intégrés. Ceci est expliqué dans le fichier *Documentation.pdf* [10] dans le répertoire du code source.

Instructions de montage

Mon objectif pour ce projet était d'utiliser le moins de composants supplémentaires possible. L'Arduino Mega et le Due ont tous deux suffisamment de ports pour faire fonctionner le simulateur.

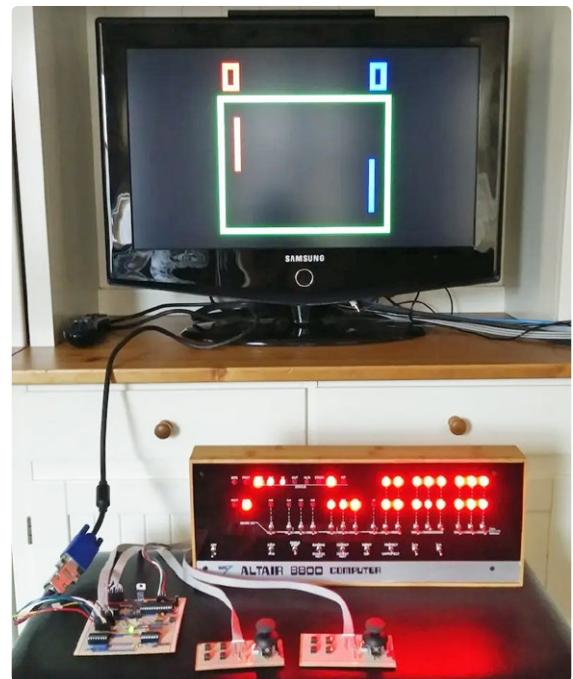


Figure 2. Emulation de la carte graphique Cromemco Dazzler en action.

```

L-R SENSOR 1-MINE -Processor Technology- & USS ENTERPRISE &
303 307 001 V-ENTPR 0I ... I STARDATE 300200
----- 0-BASE 1I ***** I CONDITION STBY
003 005 005 0-KBC 2I ^J''^I^ I QUADRANT 6-3
----- 0-KMT 3I | | I SECTOR 0-0
305 005 004 0-LNKN 4I [+0 0+] I PHOTON TORPS 10
PWR DIST % 5I : 0 : I POWER AVAIL 99%
WARP&IMP 20 6I ( 0 ) I KLINGONS LEFT 056
LR SENSOR 10 7I /1\ I ANTIMATTER PODS 03
SR SENSOR 20 8I ( E Z, Z ) I
DEFLECTORS 20 9I ( ( , , ) ) I COMMAND:
PHASERS 09 I-0-1-2-3-4-5-6-7-8-9-1 I
TORPEDOES 11 I
STARFLEET COMMAND:(CODE ?)THE KLINGONS HAVE BROKEN INTERGALACTIC
TREATY. YOUR ORDERS ARE TO ADVANCE TO KLINGON TERRITORY AND
DESTROY ALL KLINGON VESSELS.

```

Figure 3. Saisie d'écran de la carte graphique VDM-1.

samment de broches d'E/S pour gérer directement tous les éléments du panneau avant. Seuls des transistors et des résistances supplémentaires sont nécessaires pour piloter les 36 LED (si elles étaient connectées directement aux broches de sortie de l'Arduino et que trop d'entre elles étaient allumées en même temps, le courant total pourrait excéder les limites de l'Arduino).

Un schéma exhaustif serait redondant (36 circuits de pilotage de LED identiques, même câblage des 32 interrupteurs) et donc de peu d'intérêt. Le circuit est donc décrit par des sous-circuits (tels que les pilotes de LED, **figure 4**), et des tableaux [3] de connexions aux broches de l'Arduino. Un schéma Fritzing montre le câblage des LED et des résistances sur une carte à bandes conductrices (**figure 5**).

Pour créer le panneau de commande, j'ai commencé par un scan de haute qualité du panneau avant de l'Altair [11] que j'ai fait imprimer sur du carton dans une boutique de photocopie. Pour obtenir une bonne tenue mécanique, je l'ai doublé à l'arrière d'une plaque de métal de 0,8 mm d'épaisseur dans laquelle j'ai percé des trous pour les LED. Les LED et leurs résistances sont soudées sur les cartes. Le panneau avant constitue la façade d'un simple coffret en bois qui n'est pas aussi profond que l'Altair original, ne devant contenir que le panneau de commande et l'Arduino. La **figure 6** montre l'aspect intérieur.

Pour câbler l'interrupteur ON/OFF du panneau de commande, j'ai simplement fixé à l'arrière du coffret une prise de courant identique à celle de l'Arduino, que j'ai reliée à l'Arduino par un câble terminé par une fiche et coupé au milieu pour insérer l'interrupteur.

En utilisant l'Arduino Due, l'émulation de jusqu'à 16 lecteurs de disques 88-DCDD peut être activée en connectant une carte SD au port SPI de l'Arduino Due. Le détail du câblage nécessaire se trouve en dernière page du document [3].

Avant de télécharger le croquis sur l'Arduino Due, assurez-vous que le paramètre d'optimisation du compilateur Arduino est réglé sur « Performance ». Par défaut, il est réglé sur « Taille » (pourquoi ? mystère, puisque le Due a 512 Ko de mémoire flash !). Pour ce faire, chargez le fichier suivant sur dans un éditeur de texte et changez toute occurrence de -0 s en -3.

```
c:\Users\[user]\AppData\Local\Arduino15\packages\arduino\hardware\sam\1.6.9\platform.txt
```

Vous pouvez sauter cette étape, mais le simulateur fonctionnera alors nettement plus lentement.

Le logiciel du simulateur peut également fonctionner sur un Arduino seul (Mega ou Due) sans panneau de commande. Cela lui permet d'exécuter certains des programmes inclus (ceux qui utilisent essentiellement le terminal série et non les éléments du panneau). Pour ce faire, éditez le fichier source `config.h` et mettez `#define STANDALONE 1` (au lieu de `0`). Voir la section « *Debugging Capabilities* » de la documentation [3] pour apprendre à utiliser les éléments virtuels du panneau de commande dans cette configuration. Notez que ce n'est pas l'utilisation prévue du simulateur. Les émulateurs logiciels basés sur PC sont plus intuitifs si vous ne désirez pas construire vous-même le panneau de commande. Le fichier source `config.h` contient plusieurs paramètres qui activent ou désactivent des fonctionnalités du simulateur. Les valeurs par défaut fonctionnent bien, mais vous pouvez ici optimiser votre simulateur.

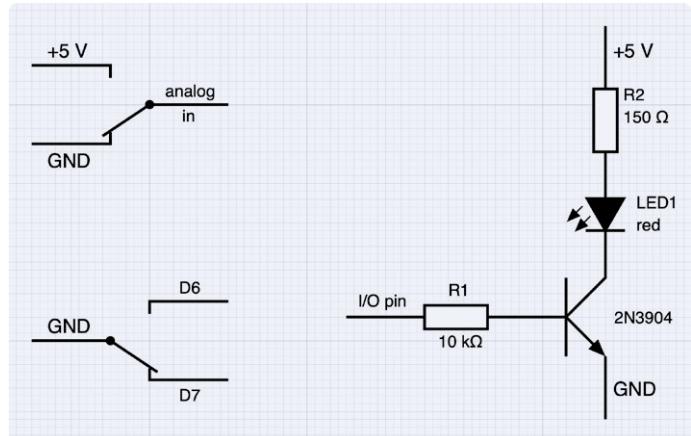


Figure 4. Circuits unitaires pour les interrupteurs et les LED.

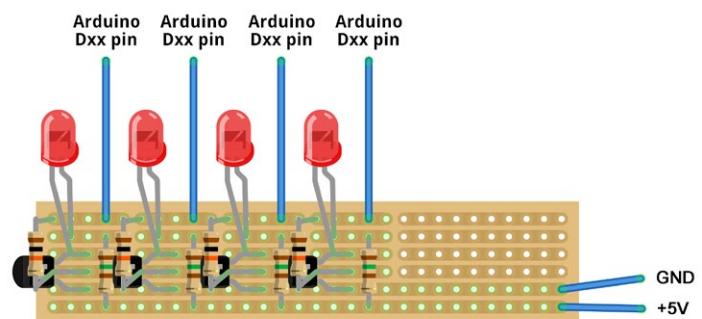


Figure 5. Composants soudés sur cartes à bandes (Fritzing).



Figure 6. Vue intérieure du clone de l'Arduino Altair.

Partageons !

Si vous trouvez ce projet intéressant et souhaitez obtenir plus d'informations, visitez le groupe Google, créé par Chris Davis, pour discuter des questions relatives à Altair-Duino [12]. N'hésitez pas à contribuer à la discussion !

Points forts

- Reproduit exactement le comportement des éléments du panneau de commande de l'Altair.
- Environ aussi rapide que l'Altair 8800 original en utilisant un Arduino Due (25 % en utilisant un Arduino Mega).
- Taille de la RAM émulée = 64 KB (Due) ou 6 KB (Mega).
- Quelques programmes Altair sont inclus et peuvent être facilement chargés sur l'émulateur : Pong, Altair 4K BASIC (le premier produit Microsoft), Altair Extended BASIC, MITS Programming System II (Due uniquement), Altair Timesharing BASIC (permet à plusieurs utilisateurs d'utiliser le BASIC simultanément).
- Des programmes d'exemple BASIC et assembleur sont inclus et peuvent être facilement chargés dans BASIC/Assembleur.
- Emule une carte 88-SIO, 88-2SIO et 88-ACR (interface pour enregistreur de cassettes audio). N'importe quel dispositif série simulé peut être mappé aux ports série de l'Arduino.

Par défaut, les deux plus communs (88-SIO et 88-2SIO port 1) sont mappés au port série principal de 115 200 bauds de l'Arduino (8N1), qui est accessible via le câble USB. Un dongle série -bluetooth connecté aux broches série RX/TX est recommandé pour que d'autres appareils puissent servir de terminal via Bluetooth.

- Sur l'Arduino Due, le port série principal (USB) et le port Serial1 (broches 18/19) peuvent être utilisés simultanément.
- Les données envoyées à chaque dispositif série (y compris la cassette ACR) peuvent être enregistrées et lues à partir de 256 fichiers au maximum, qui sont stockés dans la mémoire locale de l'Arduino (EEPROM ou FLASH).
- L'interface cassette prend en charge l'utilisation des commandes CSAVE/CLOAD dans Extended BASIC (la prise en charge est automatique et ne nécessite aucune interaction de l'utilisateur). Idéal pour développer vos propres programmes BASIC !
- Emule une carte graphique Cromemco Dazzler (nécessite du matériel/logiciel supplémentaire [9]).

- Emule une carte de terminal vidéo VDM1 de Processor Technology (nécessite du matériel/logiciel supplémentaire [10]).
- Emule jusqu'à 16 lecteurs de disques 88-DCDD (4 dans la configuration par défaut). Cette émulation est optionnelle, mais nécessite la connexion d'une carte SD au port SPI de l'Arduino. (Due uniquement).
- Emule un contrôleur de disque dur 88-HDSK avec jusqu'à 4 disques durs connectés (1 dans la configuration par défaut) et 4 plateaux par unité.
- Emule une carte 88-RTC VI avec horloge en temps réel et traitement des interruptions vectorielles. Permet l'utilisation du Altair Timesharing BASIC.
- Des pages de mémoire de 256 octets peuvent être sauvegardées en mémoire permanente et rechargées en RAM. Voilà un moyen facile de sauvegarder les programmes saisis à l'aide des interrupteurs du panneau de commande.
- De nombreux paramètres peuvent être facilement modifiés via l'éditeur de configuration intégré.

Si quelqu'un d'autre veut partager sa création, faites-le moi savoir, et je la posterai sur le site du Simulateur. J'adore passer en revue la variété des versions proposées !

220406-04 — Vf : Helmut Müller

À propos de l'auteur

David Hansel est un faiseur et un développeur. Il a découvert Arduino en 2012 et n'a cessé de l'utiliser (et d'autres microcontrôleurs) depuis pour créer des projets. Sur son profil GitHub : (<https://github.com/dhansel>), il présente des projets qu'il pense pouvoir être utiles à d'autres.

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (david@hansels.net) ou contactez Elektor (redaction@elektor.fr).

LIENS

- [1] Wikipedia sur l'Altair 8800 : https://en.wikipedia.org/wiki/Altair_8800
- [2] Kit Altair-Duino : <https://altairduino.com/>
- [3] Simulateur Arduino Altair 8800 sur le site du hub du projet Arduino : <https://tinyurl.com/yxkta8nz>
- [4] Jeu kill-the-bit sur simulateur Arduino Altair 8800 : <https://youtu.be/prdvkMP3FAA>
- [5] Démonstration musicale de l'Altair 8800 : <https://youtu.be/1FDigtF0dRQ>
- [6] Démonstration de musique avec mon simulateur : <https://youtu.be/q45ENdbz8EU>
- [7] Simulateur Altair jouant l'ouverture de Guillaume Tell : <https://youtu.be/nqy8v41q5as>
- [8] Écran Dazzler pour le simulateur Altair : <https://tinyurl.com/y8f2wzs>
- [9] Simulateur VDM-1 : <https://github.com/dhansel/VDM1>
- [10] Documentation, PDF : <https://tinyurl.com/yvxtxhb>
- [11] Panneau de commande de l'Altair : <https://vintage-computer.com/images/altairfrontpanelscan.jpg>
- [12] Groupe Google sur Altairduino.com : <https://groups.google.com/g/altair-duino>



Produits

Vous recherchez les principaux éléments mentionnés dans cet article ? Arduino et Elektor s'occupent de vous !

- **Arduino Due avec des connecteurs**
www.elektrormagazine.fr/arduino-due

