

MS-DOS sur le Portenta H7



exécuter des logiciels *old-school* sur du matériel moderne

Clemens Valens (Elektor)

Les cartes Arduino Portenta fournissent la puissance de calcul nécessaire pour émuler d'autres plateformes informatiques et virtualiser le matériel. Installez et exécutez MS-DOS sur le Portenta H7 grâce à cet article.

Dans les usines du monde entier, de nombreuses machines et lignes de production qui fabriquent les pièces des produits que nous utilisons aujourd'hui ont été construites et installées il y a plusieurs décennies. Elles exécutent des logiciels modernes sur des plateformes informatiques de pointe pour l'époque. Aujourd'hui, cependant, nombre d'entre elles sont obsolètes. Tant qu'elles continuent à fonctionner, leur modernisation peut attendre, mais que faire lorsqu'une carte tombe en panne et que les pièces de rechange sont difficiles à trouver ou, pire encore, ne sont plus disponibles ?

Virtualisez-le !

C'est là qu'interviennent les cartes Arduino Portenta. Comme elles sont généreuses en puissance de calcul pour un si petit module, elles peuvent être utilisées pour

virtualiser un autre calculateur. Cela consiste à exécuter un programme qui émule une autre manière à ce que le système émulé puisse exécuter son logiciel comme s'il s'agissait du vrai système. De l'extérieur, il n'y a aucune différence.

Une machine virtuelle peut fonctionner sur n'importe quel ordinateur, ce qui résout le problème des pièces de rechange, et elle exécute des programmes existants, ce qui limite les coûts de développement de logiciels personnalisés. Un autre avantage est la possibilité d'ajouter des fonctions modernes comme l'USB et le HDMI. Cela permet, par exemple, de remplacer les moniteurs CRT encombrants et gourmands en énergie par des écrans plats modernes et les claviers série difficiles à trouver par des modèles USB bon marché.

MS-DOS

Un système d'exploitation populaire de l'époque est le MS-DOS de Microsoft, destiné aux ordinateurs compatibles x86. Bien que son développement se soit arrêté en 2000, il est encore largement utilisé. C'est pourquoi l'équipe Arduino a décidé d'implémenter une machine x86 virtuelle pour le Portenta H7 capable d'exécuter MS-DOS. Grâce à cette technique, le Portenta peut remplacer un vieil ordinateur sur une machine qui peut ainsi continuer à fonctionner pendant 40 ans.

L'émulateur x86 pour le Portenta H7 est encore en cours de réalisation, mais vous pouvez déjà l'essayer. Voici comment.

Prérequis

Notez que pour que l'émulateur MS-DOS fonctionne, un hub USB-C avec un port HDMI et des capacités de charge (d'ordinateur portable) est nécessaire (c'est-à-dire qu'il doit avoir une entrée d'alimentation). Le Portenta émet la vidéo sur son port USB-C, et il est alimenté par le même port. Par conséquent, un hub capable de lui



Figure 1. Utilisez un hub USB-C doté d'une fonction chargeur pour ordinateur portable.

```

COM12
Available partition schemes:

Partition scheme 1
Partition 1: WiFi firmware and certificates 1MB
Partition 2: OTA and user data 13MB

Partition scheme 2
Partition 1: WiFi firmware and certificates 1MB
Partition 2: OTA 5MB
Partition 3: User data 8MB

Do you want to use partition scheme 1? Y/[n]
If No, partition scheme 2 will be used.

WARNING! Running the sketch all the content of the QSPI flash will be erased.
Do you want to proceed? Y/[n]

QSPI Flash formatted!
It's now safe to reboot or disconnect your board.

 Autoscroll  Show timestamp
Newline 115200 baud Clear output

```

Figure 2. Formatez la mémoire flash QSPI avec le schéma de partition 1.

fournir de l'énergie est pratique. Pour un hub USB-C, cette option est souvent appelée « chargeur USB-C ». Tous les hubs USB-C n'en sont pas dotés, il faut donc choisir avec soin (**figure 1**). Bien sûr, il est également possible d'alimenter le Portenta H7 à partir d'une alimentation à part, mais elle doit alors alimenter le hub et tout ce qui y est connecté également.

Aussi, pour jouer avec MS-DOS sur le Portenta, vous avez besoin d'un clavier USB. Celui-ci doit être de type standard *Low Speed* (USB 1.0), pas *Full Speed* ou plus rapide. Cette exigence est probablement remplie plus facilement par le clavier USB le plus ancien et le moins cher que vous pouvez trouver. J'ai essayé d'être malin en convertissant un vieux clavier PS/2 en clavier USB avec un Arduino Leonardo, mais cela ne fonctionne pas, car le Leonardo le transforme en clavier USB *Full Speed*.

Étape par étape

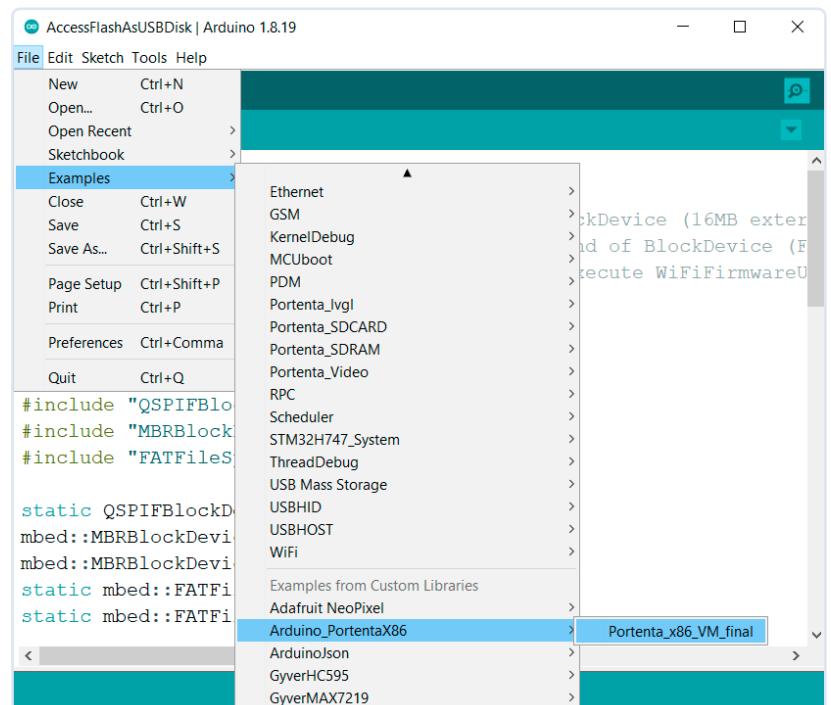
- Ouvrir l'IDE Arduino version 2.x ou 1.8.x. J'ai utilisé 1.8.19.
- Utilisez le Boards Manager pour installer Portenta Boards Package Arduino Mbed OS Portenta Board (v3.3.0 au moment de la rédaction).
- Pour la carte, sélectionnez l' *Arduino Portenta H7 (M7 core)*.
- Selectionnez le bon port.
- Utilisez le Library Manager pour installer la bibliothèque *Arduino_PortentaX86*. Si vous ne la trouvez pas, téléchargez-la depuis [1] et installez-la en tant que bibliothèque dans le dossier *libraries* de votre dossier *sketchbook*.

Maintenant, vous devez exécuter quelques sketches pour préparer la mémoire flash du Portenta :

- Exécutez *Examples* → *STM32H747_System* → *QSPIFormat* et assurez-vous d'utiliser *Partition scheme 1* (**figure 2**).
- Exécutez *Examples* → *USB Mass Storage* → *AccessFlashAsUSBDisk*.

L'étape 7 transforme le Portenta en clé USB sur laquelle vous devez copier l'image disque MS-DOS qui est incluse dans la bibliothèque *Arduino_PortentaX86* (basée sur

Figure 3. Téléchargez le sketch de la machine virtuelle (VM) *Portenta_x86_VM_final*.





```
A:\>dir
Volume in drive A is BOOT
Volume Serial Number is 1641-07BC
Directory of A:\

COMMAND   COM      54,645 12-19-96  5:09a
MOUSE      COM      13,950 07-25-12  2:22a
ATTRIB     EXE      11,208 05-31-94  6:22a
CHKDSK    EXE      12,241 05-31-94  6:22a
DEBUG      EXE      15,718 05-31-94  6:22a
EXPAND    EXE      16,129 05-31-94  6:22a
FDISK      EXE      29,336 05-31-94  6:22a
QBASIC    EXE      194,309 05-31-94  6:22a
EDIT       COM        413 05-31-94  6:22a
FORMAT    COM      22,974 05-31-94  6:22a
KEYB      COM      15,750 05-31-94  6:22a
SYS       COM      9,432 05-31-94  6:22a
AUTOEXEC  BAT        145 10-06-22 12:45p
MARIO     EXE      57,397 10-06-22 12:45p
POP1DEMO  <DIR>          10-06-22 12:52p
                           15 file(s)      453,647 bytes
                                         570,880 bytes free
```

Figure 4. Le contenu du disque A à l'invite MS-DOS.

l'émulateur MS-DOS Faux86), qui est située dans le dossier *libraries* de votre dossier *sketchbook*. Vous pouvez trouver l'emplacement de ce dernier dans les préférences de l'IDE (oui, c'est un peu compliqué).

Copiez l'image DOS dans la plus grande partition. Vérifiez-en la taille dans les propriétés du disque si vous n'êtes pas sûr.

8. Exécutez Examples → Arduino_PortentaX86 → Portenta_x86_VM_final (figure 3).

La machine virtuelle x86 est maintenant prête et chargée avec MS-DOS. Assurez-vous de connecter le clavier et

le moniteur USB au hub USB-C avant d'y connecter le Portenta, car le branchement à chaud n'est pas pris en charge (même si j'ai trouvé que cela fonctionnait la plupart du temps).

9. Allumez le hub USB-C.

Si tout est bon, vous devriez maintenant voir le BIOS émulé exécuter le test de mémoire et une fois terminé, l'invite MS-DOS devrait apparaître (figure 4) :

A:\>

Au moment de la rédaction de cet article, il n'y avait pas grand-chose à faire avec l'émulateur, mais *Edit* de MS-DOS fonctionnait (avec souris, en plus), et je pouvais aussi écrire et exécuter un programme simple en QBASIC. Une version de démonstration du jeu *Prince of Persia* de 1990 est incluse dans l'image disque (dans le dossier POP1DEMO), montrant des graphiques VGA colorés (figure 5).

Notes

Le Portenta H7 possède deux coeurs, un ARM Cortex-M4 et un ARM Cortex-M7. C'est le M7 qui émule le processeur x86 tandis que le M4 s'occupe de tout ce qui



Figure 5. L'image disque MS-DOS comprend une version jouable à durée limitée du jeu classique des années 90 *Prince of Persia*.

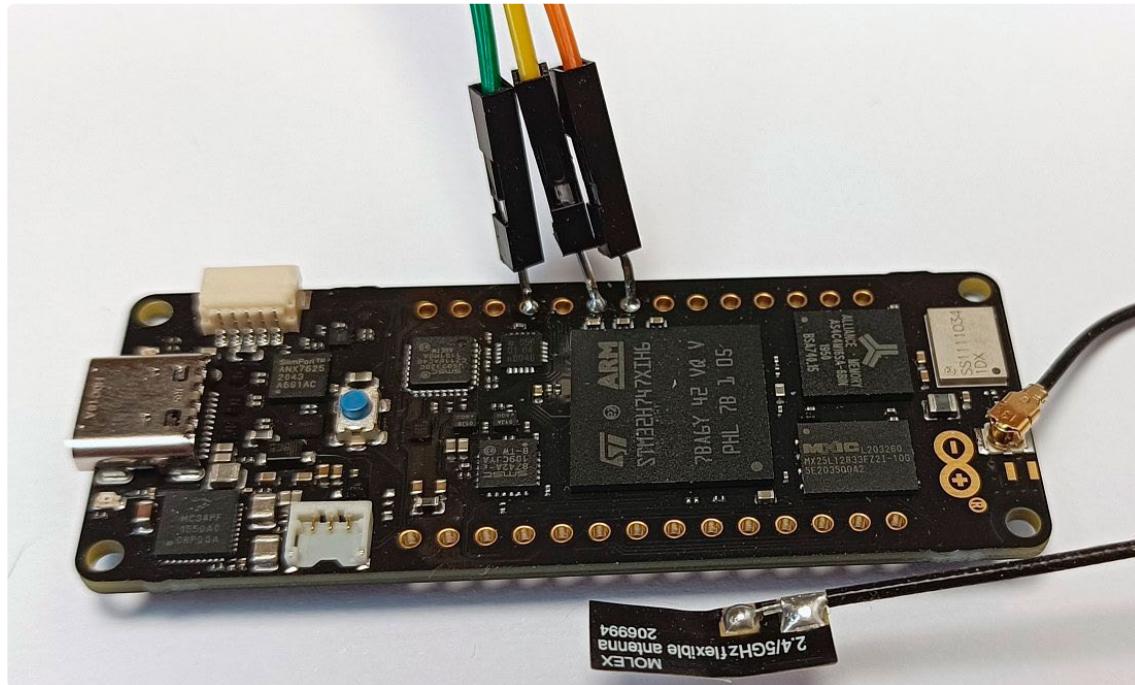


Figure 6. Rerouting des informations de débogage vers le port série sur les connecteurs d'extension.

concerne l'USB. Si le moniteur affiche un carré jaune au lieu d'une invite MS-DOS, alors ce dernier a rencontré une erreur Mbed OS. Éteindre, puis rallumer le Portenta résout généralement ce problème.

Le sketch de la machine virtuelle chargé dans l'étape 8 transmet des informations d'état et de débogage sur le port série *Serial3*, qui est disponible sur le connecteur haute densité J2. Ce port est accessible si vous disposez d'une carte d'extension Portenta, mais si vous n'en avez pas, comme moi, il est possible de dévier les données série vers un autre port. Il y en a un disponible sur les embases d'extension MKR, sur les broches D13 (RXD) et D14 (TXD), voir **figure 6**. Changez la ligne 3 du sketch [Portenta_x86_VM_final](#) pour y parvenir :

```
// UART mySerial(PG_14, PG_9);
UART mySerial(PA_9, PA_10);
```

Avec un convertisseur série-USB, vous pouvez maintenant capturer facilement les données avec un programme comme Tera Term ou RealTerm (**figure 7**).

En fonction de votre clavier USB, l'affectation des touches peut être (sera ?) erronée. Un clavier QWERTY est attendu, mais d'autres types de clavier peuvent également fonctionner. Une fois que vous avez compris où se trouvent les touches, il devient un peu plus facile d'écrire. ↵

220453-04 - VF : Maxime Valens

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (clemens.valens@elektor.com) ou contactez Elektor (redaction@elektor.fr).

Figure 7. La machine virtuelle envoie des informations de démarrage, de débogage et d'état sur un port série.

Produits

- **Carte de développement Arduino Portenta H7 (SKU 19351)**
www.elektor.fr/19351
- **Câble USB Vers RS232 TTL Série FTDI (SKU 20173)**
www.elektor.fr/20173

Lien

[1] Arduino_PortentaX86 library: http://github.com/arduino-libraries/Arduino_PortentaX86