

le projet PiKVM et ses enseignements

Entretien avec Maxim Devaev

Mathias Claußen (Elektor)

Installé sur un Raspberry Pi 4, le logiciel PiKVM permet d'accéder à distance à un PC, à un serveur ou à une autre machine par le biais d'une connexion réseau. Maxim Devaev, développeur de PiKVM, nous fait part, entre autres, de ses réflexions sur le projet...

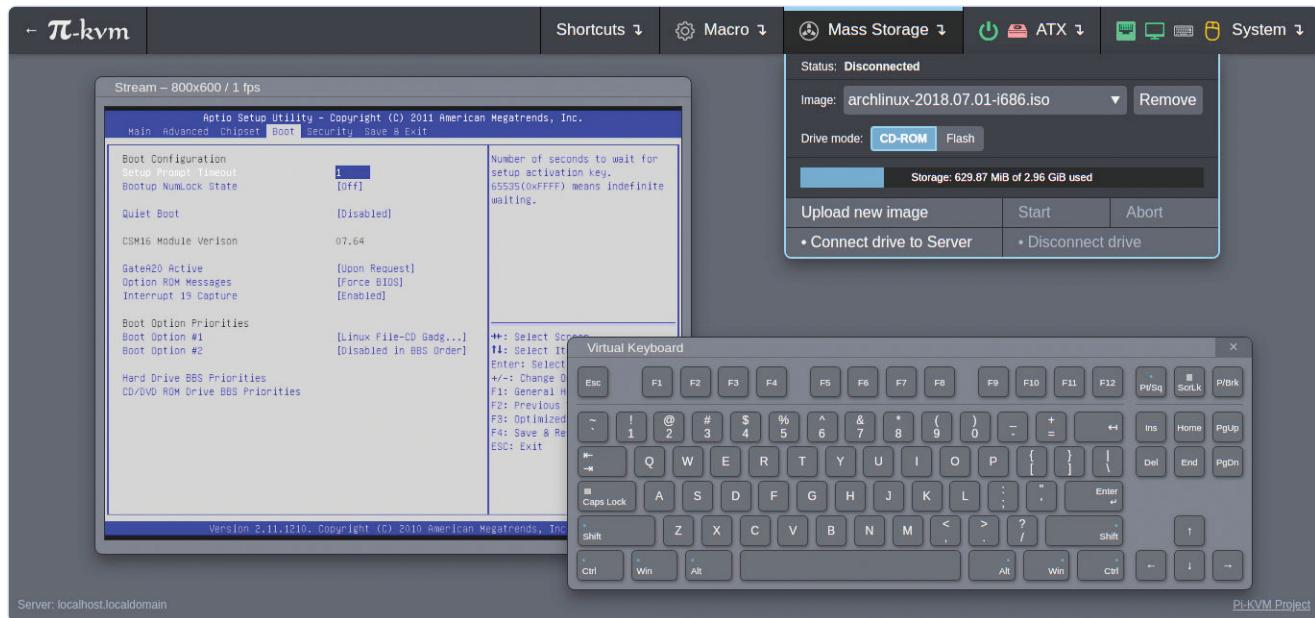


Figure 1. Interface utilisateur de PiKVM.

Le logiciel PiKVM permet de transformer un Raspberry Pi 4 en commande à distance peu coûteuse pour d'autres ordinateurs. Localement, le Raspberry Pi est connecté à l'ordinateur à contrôler. Il extrait l'image du moniteur et émule une souris, un clavier et une clé USB. Les signaux sont ensuite rendus accessibles via l'internet par le logiciel PiKVM exécuté sur le Raspberry Pi – tout

ce dont a besoin l'ordinateur de contrôle est un navigateur web. Le projet PiKVM (interface utilisateur illustrée sur la **figure 1**) est *open source* et le coût du matériel requis est inférieur à 100 € pour la version « maison », soit une fraction du prix des produits du commerce. Cela soulève quelques questions que nous avons posées au développeur du logiciel PiKVM, Maxim Devaev.

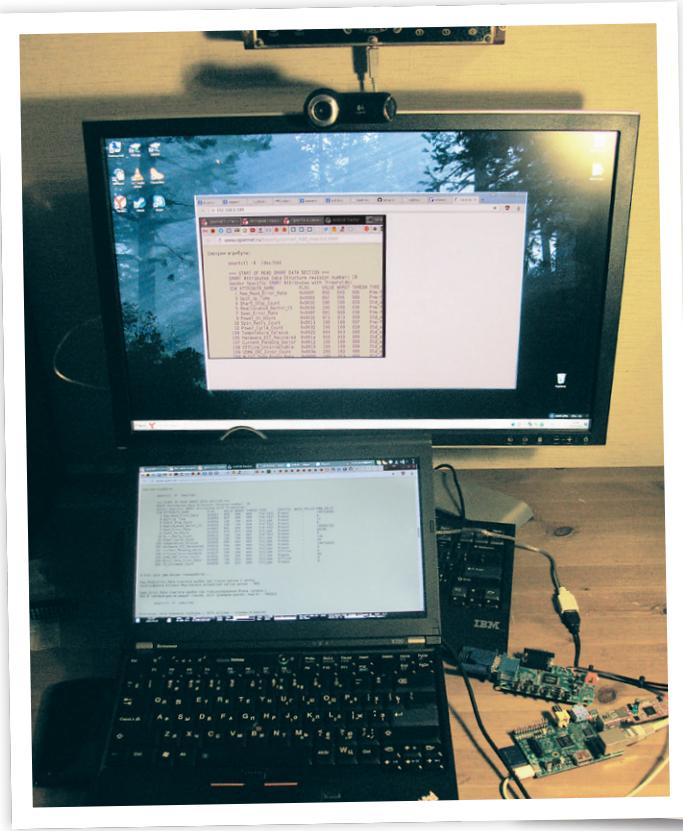


Figure 2. Prototype de PiKVM utilisant un Raspberry Pi 1.
(Source : Maxim Deveav)

Mathias : commençons par votre parcours. Comment avez-vous commencé à vous intéresser à l'électronique ? Avez-vous étudié à l'université ?

Maxim : bien sûr, mais il serait inexact de me qualifier de spécialiste en électronique. Je sais comment tenir un fer à souder, et je peux concevoir un schéma de câblage simple et l'assembler sans qu'il prenne feu. Mais je suis avant tout un ingénieur en logiciel. L'électronique est une compétence secondaire pour moi, et je l'ai beaucoup moins développée.

J'ai commencé à m'intéresser à la technique quand j'étais enfant et j'ai lu beaucoup de littérature scientifique et technique depuis. L'acquisition de mon premier PC a bien sûr beaucoup influé sur le contenu de ma bibliothèque, qui s'est étoffée avec tous mes livres d'informatique. Quelques années plus tard, lorsque le moment est venu pour moi de choisir une université, j'ai opté pour la sécurité informatique. J'ai cependant abandonné peu après, car le programme d'études m'a déçu. L'enseignement supérieur russe est très formel : il est souvent plus important de ne pas échouer à la douzaine de matières humanitaires qui l'accompagne.

J'ai donc suivi une voie d'autodidacte et trouvé un emploi de développeur de logiciels. J'ai travaillé, travaillé, et encore travaillé, je me suis amélioré, et au bout de quelque temps, j'ai atterri chez

Yandex (une sorte de Google russe). Je m'y suis installé pendant neuf années entières. Je développais des systèmes distribués, je résolvais de gros problèmes de surveillance et de gestion d'infrastructures, et tout ce qui va avec. Ce n'est peut-être pas l'aspect le plus intéressant, mais c'est ainsi.

Mathias : comment avez-vous eu l'idée de lancer le projet PiKVM ?

Maxim : c'est une histoire amusante, en fait. Une fois, je suis allé rendre visite à mes parents à 1400 km de chez moi. Mon serveur domestique, qui est actif en permanence, est tombé en panne alors que je dégustais le gâteau de ma mère. J'étais ennuyé et j'ai dû demander à mon ami d'aller chez moi pour le redémarrer. C'est ensuite que j'ai décidé d'acheter un KVM (commutateur keyboard-video-mouse) sur IP. Je suis allé en ligne, et j'ai vu... 500 \$? Sérieusement ? Qu'est-ce qu'il y a dans ce truc pour que cela coûte 500 \$? Et puis j'ai lu quelques critiques et j'ai réalisé que, pour 500 \$, vous aviez du matériel hors de prix, associé à un horrible logiciel propriétaire bogué, assorti de restrictions de licence. En outre, il n'y avait pas de bons appareils sur le marché : ils étaient tous aussi mauvais les uns que les autres. Je ne sais pas ce qui m'a pris, mais j'ai décidé que, par principe, je ne débourserai pas cette somme. Si vous voulez quelque chose de bien, faites-le vous-même. Et le reste appartient à l'histoire.

Mathias : pourquoi avoir choisi le Raspberry Pi pour ce projet ?

Maxim : le Raspberry Pi est arrivé à point nommé. Il était nouveau sur le marché et convenait parfaitement à un petit système embarqué. En fait, n'importe quel autre ordinateur ARM aurait convenu, mais le prix du nano-ordinateur, sa prédominance et l'excellent support du fabricant ont joué un rôle crucial. Je n'avais pas envie de passer du temps à déboguer des appareils d'origine inconnue, à créer des noyaux personnalisés, etc. Le Raspberry Pi m'a permis de me recentrer sur ma tâche.

Outre le Raspberry Pi, j'avais besoin d'un dispositif de capture vidéo et d'un moyen d'émuler un clavier (je n'avais pas de souris en tête à ce stade). À l'époque, il n'y avait pas de dispositif de capture VGA abordable sur le marché, j'ai donc opté pour un petit bricolage : j'ai utilisé un dongle RCA-USB servant habituellement à numériser de vieilles cassettes VHS, et j'ai introduit la vidéo analogique à partir d'un convertisseur VGA-RCA bon marché (NDLR : voir **fig. 2**). La qualité de la vidéo était horrible, mais de qualité suffisante pour configurer le BIOS et utiliser une console. Quant au clavier, j'ai acheté un convertisseur PS/2 vers USB et j'ai émulé les signaux PS/2 en utilisant le port GPIO. C'était bon marché, primitif, mais ça a marché. Finalement, j'ai abandonné tout ce matériel. J'ai opté pour une passerelle HDMI très diffusée pour la vidéo et j'ai remplacé mon convertisseur PS/2 par un Arduino avec un logiciel personnalisé.

Mathias : comment l'apparition du Raspberry Pi 4 a-t-elle influencé le projet ?

Maxim : outre de meilleures performances et un moyen d'améliorer la qualité vidéo, le RPi 4 disposait de deux contrôleurs USB. Je pouvais utiliser l'un d'eux pour émuler un clavier et une souris dans le noyau, ce qui me permettait de me passer de l'Arduino. J'ai également réussi à construire un lecteur de CD virtuel afin



Figure 3. Passerelle HDMI-CSI en deux variantes.



Figure 4. Dongle HDMI-USB.

de pouvoir réinstaller à distance un système d'exploitation sur le serveur. À ce moment, en matière de prix et de fonctions, PiKVM était en avance sur la plupart des dispositifs du commerce.

Mathias : le développement de PiKVM, c'est un hobby ?

Maxim : c'était une sorte de projet personnel. J'ai pris du plaisir à le faire, en me moquant de mon perfectionnisme. Je n'ai jamais pensé que quelqu'un d'autre que moi en aurait besoin jusqu'à ce que je tombe par hasard, sur GitHub (DIY-IPMI), sur un projet similaire à moitié abandonné. Son auteur avait suivi la même voie, mais l'avait laissé en l'état au stade de l'étude de faisabilité. Je me suis enthousiasmé. J'ai commencé à faire de la publicité pour mon projet et à m'engager à fond. J'ai ainsi résolu les problèmes les uns après les autres, ajouté de plus en plus de fonctions et consacré des milliers d'heures au projet. C'était formidable de voir combien de personnes commençaient à utiliser mon logiciel. Auparavant, je n'avais développé que de petits scripts utilitaires ou de très gros logiciels d'entreprise.

Mathias : qu'avez-vous appris de ce projet ? Des expériences à partager avec d'autres développeurs ?

Maxim : oui, en effet. L'interface web est le principal moyen d'utiliser PiKVM, mais il existe aussi un serveur VNC développé en interne. C'est plutôt cool d'utiliser un client VNC pour configurer le BIOS ! J'ai utilisé le format MJPEG pour transférer la vidéo à la fois pour le web et pour VNC. C'est un format simple, mais très peu gourmand en bande passante. Pour le navigateur, j'ai pu utiliser H.264 et WebRTC, mais pour VNC, il n'y avait aucun moyen d'utiliser un encodage différentiel complet.

J'ai pris les choses en main et j'ai écrit au responsable du dépôt TigerVNC et du protocole RFBProto (description de la norme VNC de facto) pour faire une proposition : créons un nouvel encodage, ce serait génial pour les utilisateurs à faible bande passante. Nous

avons longuement discuté des détails et nous sommes finalement parvenus à un format qui convenait à tous. Je me suis ensuite adressé à l'IANA pour demander l'enregistrement de cette nouvelle extension VNC. J'avais peur qu'ils refusent, car ils avaient déjà un format enregistré sans description, mais ils ont accepté en expliquant qu'il s'agissait d'un ancien format d'un client propriétaire. Nous sommes en train de développer un patch officiel TigerVNC pour la toute première extension open source H.264 sur VNC. La version bêta fonctionne sans problème ! Elle est peut-être toute nouvelle, mais certains clients et serveurs VNC sont déjà intéressés pour la mettre en œuvre.

La morale de cette histoire est la suivante : n'ayez pas peur de communiquer avec d'autres projets, même s'ils vous semblent monumentaux. Autre chose aussi : une personne, à elle seule, peut faire avancer le développement de l'ensemble du protocole ; il suffit de rêver grand.

Mathias : l'apparition du dongle USB-HDMI à 10 \$ a-t-elle fait avancer le projet PiKVM ?

Maxim : cela dépend. Avant le dongle USB, j'utilisais une passerelle HDMI-CSI (fig. 3) qui présentait de nombreux avantages. Le dongle USB (fig. 4) a certainement joué un grand rôle dans la popularisation de PiKVM car il était vraiment bon marché. Cependant, j'ai toujours déconseillé de l'utiliser en raison de problèmes matériels : il n'est pas très stable.

De nombreuses personnes ont déclaré l'avoir acheté et l'avoir regretté par la suite, car elles voyaient souvent un écran vide et ne pouvaient rien y faire. Un dongle USB est une boîte noire : vous n'avez aucune idée de son fonctionnement, et si quelque chose ne va pas, vous ne pouvez pas le réparer. La passerelle, en revanche, est beaucoup plus ouverte et presque tout le traitement vidéo est effectué par le noyau et mes logiciels.

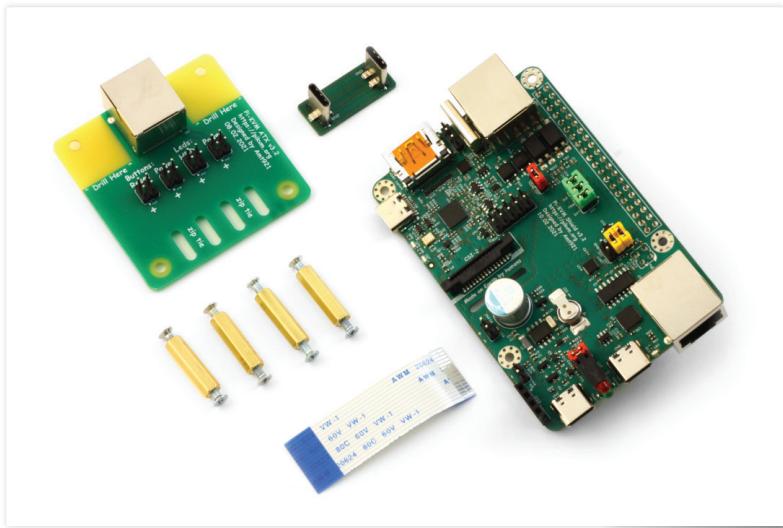


Figure 5. HAT PiKVM v3. (Source : Maxim Deveav)

Mathias : PiKVM est un projet *open source* que l'on peut utiliser gratuitement (pour l'instant). Comment son développement est-il financé ? Vous semblez maintenant travailler à plein temps sur le projet.

Maxim : je mentirais si je disais que j'ai une vision à long terme, un plan et une bonne compréhension du marché. Tout a commencé l'année dernière, lorsque j'ai quitté mon emploi pour faire une pause et réfléchir à ma carrière. J'ai continué à travailler sur PiKVM comme un hobby. La communauté s'est agrandie et je me suis dit que ce serait formidable si PiKVM devenait mon travail principal. J'ai aussi vu certains projets *open source* dont les auteurs vivaient. Ce serait une situation gagnant-gagnant : je ferais ce qui m'intéresse vraiment, et les utilisateurs obtiendraient un logiciel de qualité en contrepartie d'un petit don. En guise de test, j'ai créé Patreon et j'ai annoncé que j'acceptais les dons, car j'étais désormais sans emploi et prêt à améliorer PiKVM sur demande (au cas où vous auriez besoin rapidement d'une fonction particulière). À ma grande surprise, plusieurs mois plus tard, j'ai commencé à faire quelques bénéfices. C'était bien moins que mon salaire précédent, mais si l'on considère que j'ai arrêté de dépenser de l'argent pour me rendre au bureau, que je dors maintenant suffisamment et que j'ai même perdu un peu de poids, c'était un échange équitable. Les utilisateurs étaient également satisfaits, car ils pouvaient désormais s'écouler une demi-heure entre la découverte d'un bogue critique et sa correction, et je pouvais ajouter de nouvelles fonctions en quelques jours.

Mathias : comment la communauté soutient-elle votre projet, en plus de vous aider à développer à plein temps ?

Maxim : je ne suis pas très doué pour écrire de la documentation, mais beaucoup de personnes m'aident pour cela. Certains écrivent des recettes et des scripts, ou imaginent et décrivent de nouvelles façons

d'utiliser certaines des fonctions de PiKVM. D'autres créent des rapports de bogues et me donnent un accès à distance à leur matériel pour que je puisse résoudre les problèmes. Mais l'évolution la plus essentielle est ailleurs : lorsque le nombre d'utilisateurs est devenu vraiment élevé, il n'a plus été possible de répondre à chaque personne sur Discord. Les utilisateurs actifs m'ont donc aidé à intégrer de nouveaux et ils ne me contactent maintenant que lorsqu'ils ne peuvent pas résoudre un problème eux-mêmes. Je suis très reconnaissant envers ces personnes, car je peux maintenant consacrer tout mon temps à l'écriture de logiciels.

Parfois, il y a des personnes qui apportent des correctifs avec de nouvelles fonctions ou des corrections de bogues. Cependant, elles ne sont pas très nombreuses et je suis pratiquement le seul développeur PiKVM pour le moment. J'ai l'intention d'engager des personnes pour m'aider à plein temps avec les logiciels lorsque nous commencerons à vendre des appareils. Ainsi, si vous achetez notre matériel, vous sponsorisez le développement de logiciels *open source*.

Mathias : si quelqu'un souhaite soutenir votre travail, quelle serait la meilleure façon de le faire ?

Maxim : en dehors des dons, nous apprécions toute aide concernant ce que nous ne pouvons pas encore gérer par nous-mêmes. Certains de ces domaines sont décrits dans nos problèmes GitHub.

Mathias : avez-vous l'intention de commercialiser vos travaux ?

Maxim : actuellement, ma principale source de revenus est constituée de dons. On me suggère souvent que je pourrais vendre une licence commerciale pour le logiciel PiKVM avec des fonctions avancées, mais je ne veux pas le faire pour deux raisons. Premièrement, PiKVM a un accès direct au matériel et je ne ferais pas confiance à un tel dispositif s'il n'était pas *open source*. Deuxièmement, un système à péage conduirait inévitablement quelqu'un à forcer votre logiciel ouvert pour développer des fonctions pro gratuitement. Une partie de votre communauté serait ainsi rattachée à d'autres personnes.

L'autre moyen de monétisation, qui deviendra bientôt le principal, est la vente de mon propre matériel (HAT PiKVM v3) (fig. 5), que vous pouvez utiliser à la place de cette version « maison » pour obtenir plus de stabilité et des fonctions matérielles supplémentaires. Mais je n'ai pas l'intention d'abandonner la version « maison » de PiKVM et de forcer les utilisateurs à acheter la v3. Je ne recherche pas la richesse à tout prix, juste assez d'argent pour vivre et nourrir mon chat.

Mathias : pouvez-vous nous en dire un peu plus sur votre matériel, le HAT PiKVM v3 ?

Maxim : pour « bricoler » un PiKVM, vous devez procéder manuellement et prendre éventuellement en main un fer à souder. J'ai décidé de rendre la tâche un peu plus facile pour l'utilisateur, mais

aussi d'améliorer la stabilité matérielle de cette version et d'ajouter certaines fonctions difficiles à mettre au point. Bien que je sois à l'origine du concept original de l'appareil, je ne pouvais pas développer une carte entièrement personnalisée (je ne possède pas ce dont spécifique), j'ai donc invité mon ami, compétent dans le domaine du matériel, à travailler avec moi sur ce projet. Notre HAT PiKVM v3 dispose d'un contrôleur d'alimentation de serveur intégré, d'un dispositif de capture vidéo, d'un commutateur USB (pour émuler la sortie d'un câble, c'est parfois utile), et d'un port série (pour se connecter à un serveur ou une console dans un rack), d'une horloge pour une journalisation précise – le tout sur une seule carte. Pour l'essentiel, c'est une solution plus performante et moins chère que toutes celles à 500 \$ proposées sur le marché. Vous n'êtes pas obligé de me croire sur parole : les critiques de la v3 sont consultables sur YouTube. Et bientôt, tout le monde pourra l'acheter.

Mathias : vous avez également un boîtier en métal pour le Raspberry Pi 4 et votre module d'extension. Y avait-il une raison de ne pas utiliser de matière plastique ? Nous avons lu sur Discord qu'en Russie, il est plus facile de construire quelque chose en métal qu'en caoutchouc ou en matière plastique.

Maxim : c'est vrai. Le boîtier métallique s'est avéré plus lourd, plus solide et beaucoup moins cher qu'un boîtier en plastique. Ceux qui ont commandé le HAT v3 sur Kickstarter ont pu obtenir ce boîtier gratuitement. Il sera ensuite vendu dans les boutiques en ligne. Et nous avons également un plan gratuit du boîtier pour l'impression 3D, si vous aimez travailler avec vos mains.

Mathias : votre kit sera-t-il disponible pour des clients établis hors de Russie ?

Maxim : le premier lot est maintenant disponible aux États-Unis. Le deuxième lot devrait bientôt être disponible dans le monde entier.

Mathias : quels sont vos projets futurs pour PiKVM ?

Maxim : nous allons continuer à développer à la fois le logiciel et le matériel. Nous prévoyons d'implémenter la capture 60 Hz 1080p (ce qui n'est pas possible actuellement en raison des restrictions du RPi 4) et de créer une solution sécurisée pour l'accès au cloud pour les utilisateurs sans adresse IP externe ouverte.

Mathias : avez-vous d'autres projets en cours de développement ?
Sur quoi travaillez-vous ?

Maxim : uStreamer, par exemple. Il a été créé pour PiKVM, mais c'est un composant totalement distinct que beaucoup utilisent comme serveur pour les vidéos de webcam. (Il sert de périphérique de capture dans PiKVM.) Je l'ai écrit à partir de zéro comme une alternative à MJPG-Streamer pour résoudre le problème de performance. Aujourd'hui, uStreamer est le service vidéo MJPEG/H.264 spécialisé le plus rapide, et j'en suis vraiment fier.

Mathias : Maxim Deveav, merci pour cet entretien. Pour ceux qui se dirigent maintenant vers la campagne Kickstarter, nous avons de bonnes nouvelles. La campagne a été plus que réussie. Cela signifie que vous pourrez bientôt commander le HAT PiKVM v3 dans une boutique en ligne. Jetez également un coup d'œil à la boutique Elektor, car notre équipe d'acheteurs travaille actuellement pour pouvoir vous proposer également des HAT PiKVM v3 dans un avenir proche. ►

210405-04 - VF : Pascal Godart

Des questions, des commentaires ?

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



PRODUITS

► **Carte d'adaptation HDMI -> CSI-2 pour Raspberry Pi (supporte jusqu'à 1080p, 25fps)**
www.elektor.fr/19707

LIENS

- [1] M. Claußen, « RaspberryPi comme télécommande KVM », Elektor 01-02/2022 : <http://www.elektormagazine.fr/200523-04>
- [2] Page d'accueil Pi-KVM : <https://pikvm.org/>
- [3] Pi-KVM sur Discord : <https://discord.com/invite/bpmXfz5>
- [4] Problèmes Pi-KVM sur GitHub : <https://github.com/pikvm/pikvm/issues>
- [5] HAT Pi-KVM v3 HAT sur Kickstarter : <https://www.kickstarter.com/projects/mMaxim/pikvm-v3-hat>
- [6] Boutique Elektor : <http://www.elektor.fr>