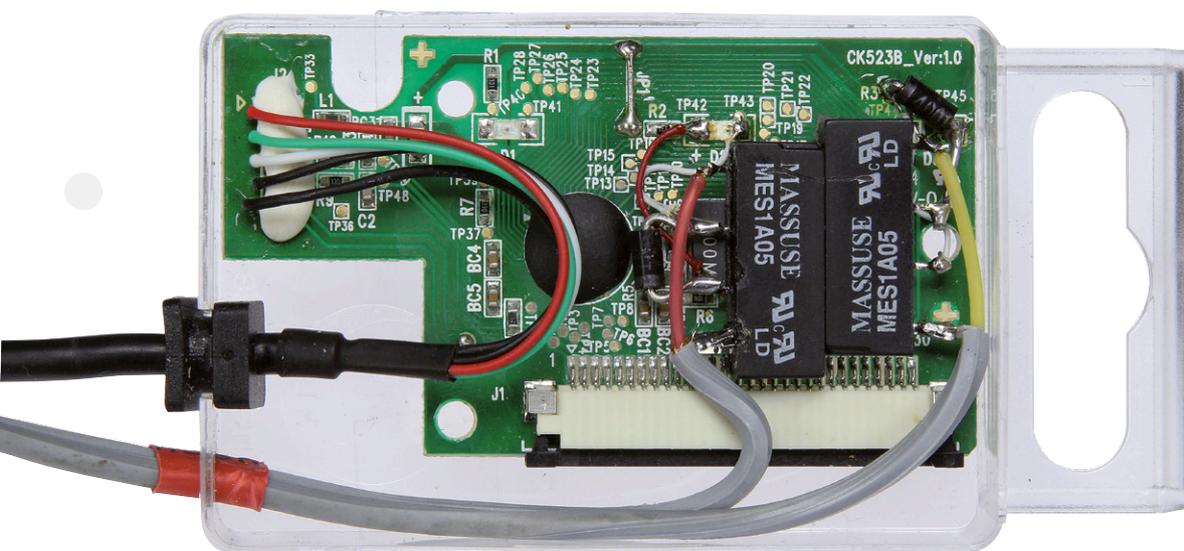


commuter des sorties avec les LED d'un clavier nouveau rôle pour vieux clavier de PC



Joachim Berg
(Allemagne)

Vous avez sans doute (au moins) un ancien clavier d'ordinateur relégué pour cause de déficience mécanique. Vous pensiez le réparer un jour... ou en faire un autre usage ? L'occasion se présente maintenant !

Les claviers de PC succombent plus souvent à un défaut mécanique qu'à une panne électronique : une touche ne répond plus et c'est le clavier entier qui est devenu inutilisable. Le contrôleur d'un clavier de PC commande d'habitude trois LED, qui indiquent l'activation du pavé numérique (*Num Lock*), le verrouillage des capitales (*Shift Lock*) et le blocage du défilement (*Scroll Lock*). Que direz-vous d'utiliser ces trois commandes comme simple interface de commutation, pour activer p. ex. des relais depuis votre PC ?

La seule chose qui nous intéresse dans le clavier à détourner, c'est sa petite carte avec le contrôleur, située généralement à droite sous le pavé numérique, et avec les trois LED en question. La photo montre qu'il y a aussi sur cette carte une connexion USB avec un câble à quatre conducteurs, deux pour l'alimenta-

tion +5 V (le rouge et le noir) et deux pour les données (vert et blanc) plus le blindage. La réalisation est simple, elle ne réclame qu'un peu de soudage sur la carte. Les relais miniatures à 5 V consomment si peu qu'ils peuvent remplacer directement les LED. Auquel cas, les résistances en série restent pour protéger les transistors de commutation de la puce. Pour atteindre les pastilles et y souder des fils, on dessouda les LED en CMS sur la carte (ou on les brisera).

S'il faut pouvoir commuter la tension secteur, on remplacera les LED soit par des photocoupleurs soit par de plus gros relais de 5 V, mais alors il faut intercaler des transistors PNP ou à effet de champ. Leur base ou leur grille pourra se relier directement à chacune des LED, lesquelles restent intactes dans ce cas. Dans le choix des relais, on tiendra compte

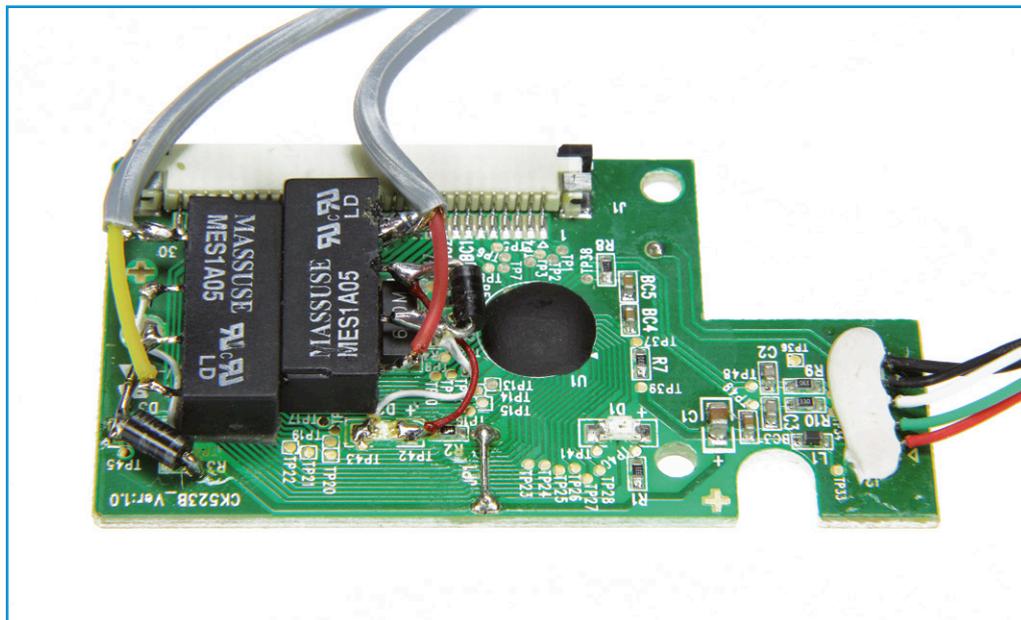


Figure 1.
La carte du contrôleur avec les relais soudés.

des limites de charge maximale de l’interface USB, normalement à 500 mA (les claviers ne consomment jamais autant).

Après des deux composants additionnels visibles sur la **figure 1**, on trouve les diodes de roue libre (1N4001 ou équivalentes) sur les bobines des relais, une protection contre les surtensions pour les sorties du contrôleur et éventuellement les transistors de commande. On voit aussi du câble stéréo ordinaire soudé sur les sorties des relais, mais n’importe quel câble fera l’affaire. J’ai installé la carte du contrôleur dans une vieille boîte de vis.

Logiciel

C’est ici que cela devient intéressant : j’ai préparé une petite bibliothèque de programmes (dll) pour commander automatiquement les LED à partir de n’importe quel langage de programmation. Le code source est en C# pour .Net 4.0 ou plus récent, on peut le traduire au moyen de *Visual Studio* [1] de *Microsoft*.

La dll est utilisable en 32 bits ou 64 bits, le code source commenté (*LdbLEDs.cs*) et des applications de test sont disponibles [2]. Le **tableau 1** détaille le contenu du paquet de logiciels.

La bibliothèque *KbdLEDs.dll* est restée très simple. L’utilisation de temporisateurs différents pour les trois interrupteurs nous dispense d’une aventure en *multithread*. La bibliothèque est compilée pour tout processeur, elle s’adapte aussi au temps d’exécution des plateformes à 64 ou 32 bits. Dans votre projet personnel, il vous suffira de faire référence à la dll et vous disposerez des trois interrupteurs comme *KbdLEDs.LED*, *ScrollLock*, *.CapsLock* et *.NumLock*.

Changer de position un interrupteur se fait par la méthode `Toggle(LED led)`, avec `SetToggleState(LED led, bool toggled)` l’interrupteur est mis dans l’autre position et retour. On peut interroger sur son état

Tableau 1. Contenu du paquet de logiciels.

KbdLEDs.dll	la bibliothèque
Dossier Sources	le code source
CallKbdLedsC.bat	groupe d’exemples pour KbdLedsC
KbdLedsC.exe	programme de console pour l’exécution en groupe
TestKbdLEDs.exe	programme de test (WPF)
TestKeyLEDsForms.exe	programme de test (Windows Forms)

actuel par `GetToggleState(LED led)` et produire des impulsions, d'une durée déterminée en millisecondes, par `Impulse(LED led, int msDuration)`. Chaque LED est alors allumée pour la durée indiquée, les impulsions des différentes LED peuvent se chevaucher. L'application de console `KbdLedsC.exe` permet la commutation à partir de données groupées s'il n'y a pas de programme spécial à utiliser. On se sert des paramètres suivants pour l'appel du groupe :

pas du meilleur style.

Sur les claviers et portables récents, il n'y a généralement plus de LED *Scroll Lock*, mais le programme de démonstration continue à indiquer à l'écran l'état courant des trois signaux.

(140017 - version française : Robert Grignard)

Liens :

[1] www.visualstudio.com

[2] www.elektor-magazine.fr/post

allumage et extinction

```
KbdLedsC /s=0 /s=1 /n=0 /n=1 /c=0 /c=1
(s = ScrollLock, n = NumLock, c = Capslock).
```

génération d'impulsion

```
KbdLedsC /s=1000 /n=500 /c=2000
(temps en millisecondes)
```

Il est possible de mettre plusieurs paramètres dans un même appel. Si quelque chose ne vous convient pas dans l'application de la console, elle peut au moins servir de guide pour démarrer.

La bibliothèque dll doit être accessible, vous pouvez la copier à côté de l'application de console.

Dans le monde de .NET, il existe deux *Framework* utilisables pour créer des interfaces graphiques (WPF et *Windows Forms*), on y trouve deux programmes de démonstration et de test à télécharger (**figure 2**). On demande en permanence l'état des LED, c'est pratique pour les tests, sinon, ce n'est

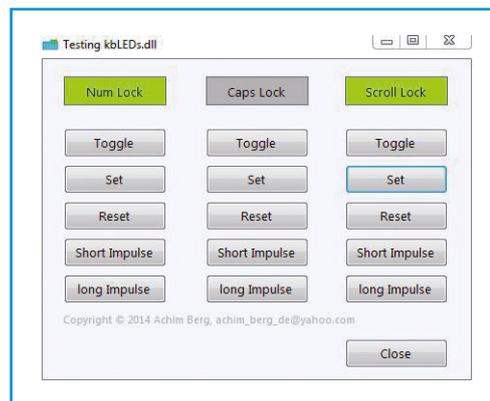


Figure 2. Les programmes de test donnent vite une idée de la fonction de la dll.

