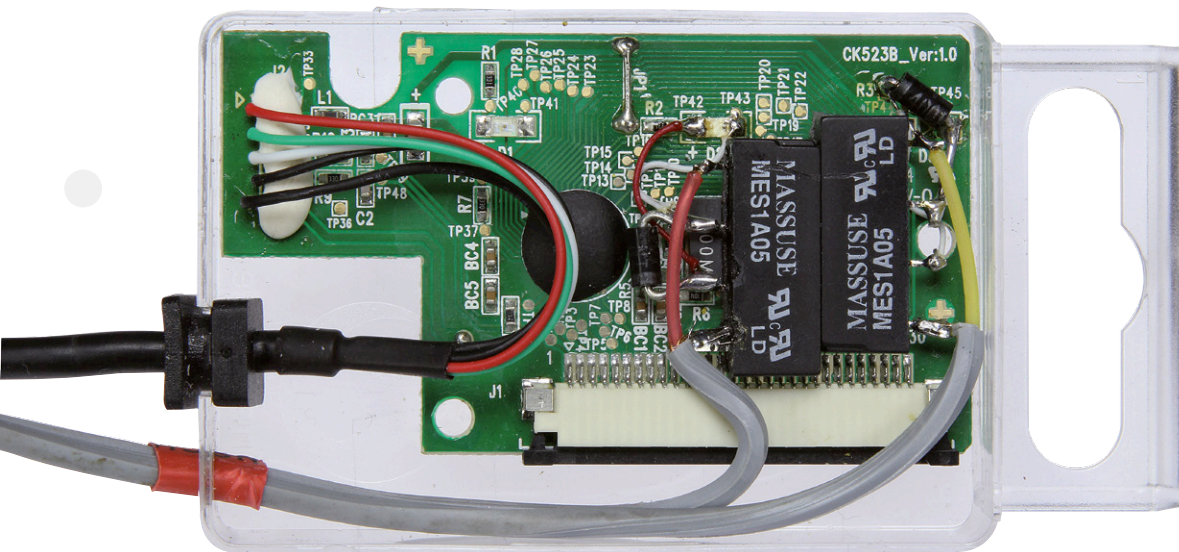


Schalten mit Keyboard-LEDs

Alte Tastaturen neu eingesetzt

Von Joachim Berg



Viele User bewahren alte Computertastaturen mit mechanischen Fehlern auf. Man könnte sie ja mal reparieren oder für irgendetwas anderes gebrauchen. Genau dieser Fall tritt jetzt ein!

Bei vielen Computernutzern liegen ausgemusterte Tastaturen herum, die lediglich mechanische Fehler haben. Die Controller in den Tastaturen steuern meistens drei LEDs für Numlock, Capslock und Scrolllock (Rollen) an. Da man Tastaturen parallel betreiben kann, lassen sich die Tastaturcontroller als einfache Schalt-Interfaces nutzen. Besonders das Scrolllock-Signal wird im normalen Rechnerbetrieb ja nicht gebraucht.

Alles, was man vom Keyboard benötigt, ist die kleine Controllerplatine, die üblicherweise rechts unter dem Nummernblock steckt und die drei LEDs trägt. Wie auf dem Titelfoto zu sehen, befindet sich auch der USB-Anschluss samt vierpoligem Kabel für die +5 V-Spannungsversorgung (rot/schwarz) und die Daten (grün/weiß) plus Abschirmung auf der Platine. Die Realisierung ist einfach, es ist nur ein

wenig Löterei auf der Platine erforderlich. Mini-Reed-Relais (5 V) nehmen so wenig Strom auf, dass sie die LEDs auf der Controllerplatine einfach ersetzen können. Dabei kann sogar der LED-Vorwiderstand auf der Platine verbleiben, um die Schalttransistoren im Chip zu schonen. Die SMD-LEDs lassen sich leicht abschleifen oder auslöten, um Drähte an die Kontakte zu löten.

Natürlich kann man die LEDs auch durch Optokoppler ersetzen oder durch größere 5-V-Relais, die Netzspannung schalten können. Dann müssen aber pnp- oder Feldeffekttransistoren vorgeschaltet werden, deren Basen beziehungsweise Gates direkt an den LEDs angeschlossen werden können. Die LEDs dürfen in diesem Falle intakt bleiben. Bei der Auswahl der Relais muss man aber auf die maximale Belastbarkeit der USB-Schnittstelle achten,

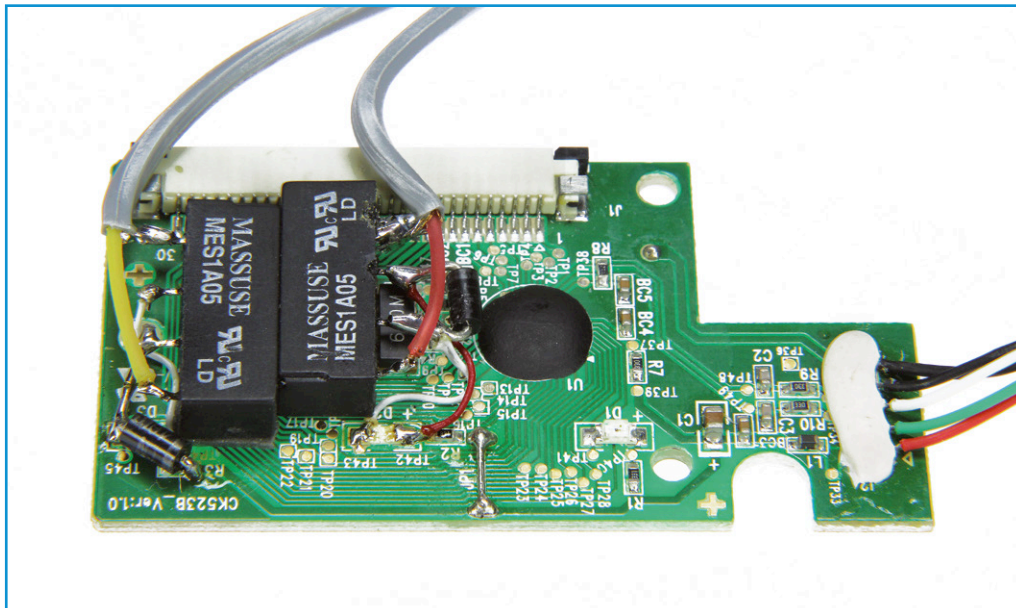


Bild 1.
Die Controllerplatine mit angelöteten Relais.

sie dürfen keine zu hohen Ströme ziehen. Den Regeln entsprechend darf einer USB-Schnittstelle 500 mA entnommen werden, Tastaturen melden jedoch meist einen geringeren Strombedarf an.

Bei den beiden zusätzlichen Bauteilen, die in **Bild 1** zu erkennen sind, handelt es sich um Freilaufdioden (1N4001 oder ähnliches) für die Relaispulen, die die Ausgänge des Controllers oder die eventuellen Treibertransistoren vor Überspannungen schützen. An die Relaisausgänge ist hier ein gewöhnliches Stereo-Line-Kabel angelötet, man kann hier natürlich ein beliebiges Schaltkabel verwenden. Ich habe die Controller kostengünstig in einer alten Schraubendose aus dem Baumarkt untergebracht.

Software

Und jetzt wird's erst interessant: Ich habe eine kleine Programmbibliothek (.dll) geschrieben, mit der man die LEDs aus beliebigen Programmiersprachen heraus programmgesteu-

ert schalten kann. Der Quellcode ist in C# für .Net 4.0 oder neuer geschrieben und kann mit Microsoft Visual Studio™ [1] übersetzt werden. Die .dll ist unter 32 bit und 64 bit nutzbar, kommentierter Quellcode (LdbLEDs.cs) und Test-Applikationen sind unter [2] erhältlich. **Tabelle 1** listet den Inhalt des Softwarepakets auf.

Die Bibliothek **KbdLEDs.dll** wurde sehr einfach gehalten. Durch getrennte Timer für die drei Schalter kommt sie ohne Multithreading aus. Die Bibliothek ist für *any CPU* kompiliert, passt sich also zur Laufzeit an 64-bit- oder 32-bit-Plattformen an. Für Ihr eigenes Projekt muss nur die .dll referenziert werden, dann stehen Ihnen die drei Schalter als „KbdLEDs.LED.ScrollLock“, „-.CapsLock“ und „-.NumLock“ zur Verfügung.

Zum Umschalten eines Schalters dient die Methode „Toggle(LED led)“, mit „SetToggleState(LED led, bool toggled)“ wird der Schalter gesetzt respektive zurückgesetzt.

Tabelle 1. Inhalt des Softwarepakets.	
KbdLEDs.dll	Die Bibliothek
Ordner "Sources"	Quellcode
CallKbdLedsC.bat	Beispiel-Batch für KbdLedsC
KbdLedsC.exe	Konsolenprogramm für Batches
TestKbdLEDs.exe	Testprogramm (WPF)
TestKeyLEDsForms.exe	Testprogramm (Windows Forms)

Der Status kann mit „*GetToggleState(LED led)*“ abgefragt werden. Impulse können mit „*Impulse(LED led, int msDuration)*“ mit einer Dauer in Millisekunden ausgelöst werden. Die LED wird dann für die Dauer des Impulses eingeschaltet. Impulse verschiedener LEDs dürfen sich überlappen.

Die Konsolen-Anwendung, **KbdLedsC.exe** erlaubt das Schalten aus Batch-Dateien heraus, wenn keine eigene Programmierung zum Einsatz kommen soll. Der Batchaufruf kann folgende Parameter enthalten:

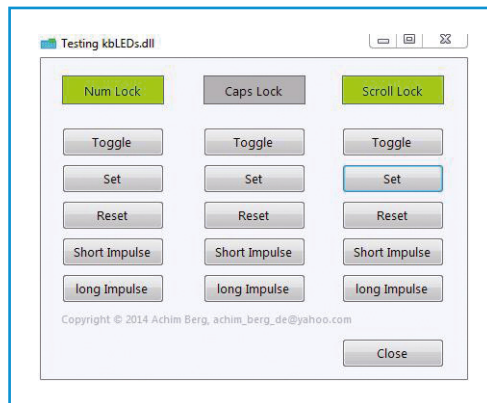


Bild 2. Die Testprogramme vermitteln einen schnellen Eindruck von der Funktion der .dll.

Ein- und Ausschalten

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

Impulse auslösen

```
KbdLedsC /s=1000 /n=500 /c=2000
(Zeiten in Millisekunden)
```

Mehrere Parameter in einem Aufruf sind möglich. Wenn der Konsolenanwendung etwas nicht gefällt, gibt sie eine Kurzanleitung aus. Die .dll-Bibliothek muss erreichbar sein. Man kann sie einfach neben die Konsolenanwendung kopieren.

In der .NET-Welt gibt es zwei gebräuchliche Frameworks, mit denen man grafische User-Interfaces erstellen kann (WPF und Windows Forms), daher finden sich im Download gleich zwei Demo/Test-Programme (**Bild 2**). Der Status der LEDs wird hier ständig abgefragt (Polling), das ist zum Testen

ganz ok, aber sonst ein eher schlechter Stil. Bei modernen Tastaturen und Laptops ist manchmal keine ScrollLock-LED vorhanden; die Demo-Programme zeigen aber immer den aktuellen Zustand aller drei Signale auf dem Bildschirm.

(140017)

Weblinks:

- [1] www.visualstudio.com/de-de/
- [2] www.elektor-magazine.de/post

