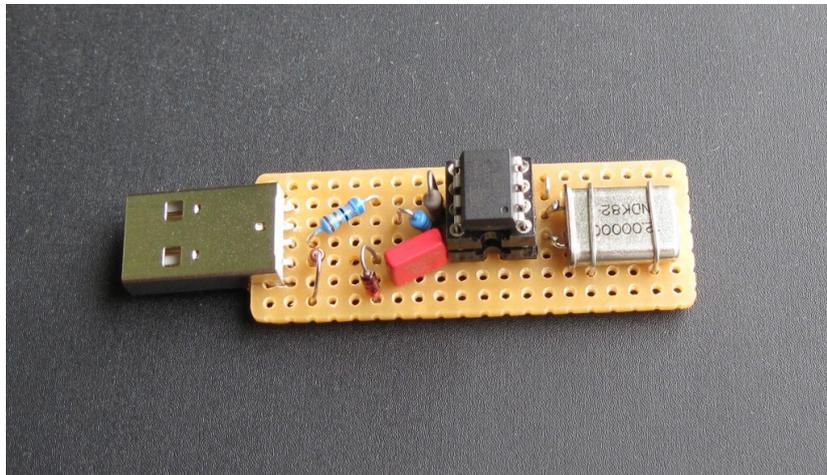


# wilde Maus

Ralf Beesner

1. August 2012



## 1 Übersicht

Mit dem VUSB- Treiberpaket kann man fast jeden AVR- Mikrocontroller in ein USB- Lowspeed- Gerät verwandeln.

Die unter <http://www.obdev.at/products/vusb/downloads-de.html> bereitgestellten Quellcode- Pakete des VUSB- Treibers enthalten als Demo- Anwendung eine virtuelle Maus, die den Mauszeiger auf dem Bildschirm langsam im Kreis bewegt. Diese Beispielanwendung lässt sich mit wenig Aufwand in eine Spaß- Schaltung umwandeln.

Ich habe den Code so modifiziert, dass der Mauszeiger stattdessen nach Ablauf einer Zufallszeit zu zittern beginnt (Versionsname: Tremor) bzw. nach Ablauf der Zufallszeit alle paar Sekunden willkürliche Sprünge in alle 4 Richtungen ausführt (Version: Weird\_Mouse).

## 2 Hardware

Verwendet wird die gleiche Hardware wie beim Windows- Rebooter. Schaltplan und Layout- Vorschlag sind daher ohne nähere Erläuterung beigefügt.

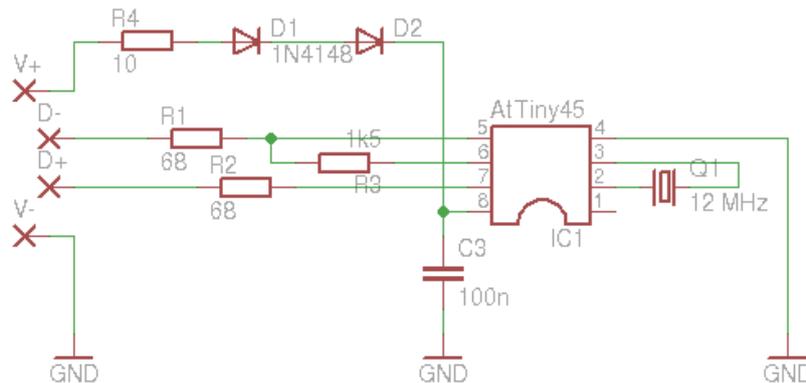


Abbildung 1: Schaltbild

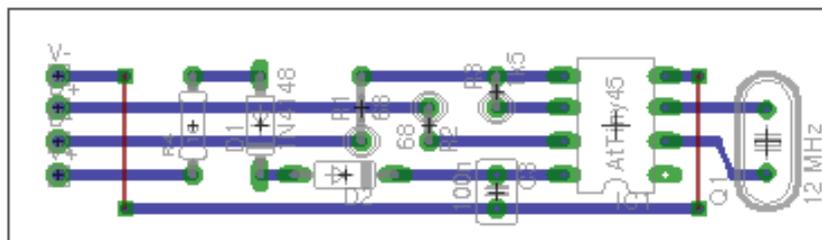


Abbildung 2: Streifenraster-Layout

## 3 Tremor

Die Software verhält sich für eine Zufallszeit von 2-70 Minuten passiv und legt dann den 1,5 kOhm- Widerstand auf high, um sich beim Betriebssystem als Low- Speed- Gerät anzumelden (die Testversion `main-test.hex` wartet nur 10 Sekunden).

Nach erfolgreicher Anmeldung springt sie aus der Main Loop nach jedem USB-Interrupt in die Funktion `drunken_Pointer`. Diese erzeugt jedesmal mit Hilfe der `rand()`- Funktion neue X- und Y- Koordinaten für den Mauszeiger. Die Konstante 2 und der Teilerwert 6200 sind experimentell ermittelt; sie bewirken, dass die Koordinaten um den Wert 0 herumzittern und der Mauszeiger nicht rasch wegdriftet.

## 4 Weird\_Mouse

Diese Version verhält sich zunächst gleich wie die Tremor- Version; in der Main Loop ruft sie jedoch die Funktion `weirdPointer` auf.

In dieser Funktion wird die `rand()`-Funktion jeweils drei Mal verwendet, um Zufallswerte für die Variablen `Delay`, `direction` und `distance` zu erhalten. Die Random- Werte werden per Shift- Befehl durch 512, 8192 und 256 geteilt, damit die Variablen im jeweils passenden Wertebereich liegen.

Der Zufallswert für `Delay` sorgt dafür, dass der Mauszeiger in unregelmäßigen Abständen springt, `distance` bestimmt die Sprungweite und `direction` die Richtung.

Ist `Delay` noch nicht abgelaufen, sendet die Software als x- und y- Koordinate jeweils eine "0". Ähnlich wie beim USB- Keyboard verhält sich auch der PC- Maustreiber nach der Devise: "der letzte Befehl gilt". Sendet man zwischendurch keine Nullen, "flitzt" der Mauszeiger sofort an den Bildschirmrand.

## 5 Flashen

Die erforderliche Fusebytes sind genau so wie beim Windows Rebooter:

`lfuse=0xef` aktiviert den Quarzoszillator

`hfuse=0xdd` aktiviert den Brownout- Detektor mit einer Schwelle von 2,7V

## 6 Anwendung

Die Schaltung verhält sich zufallsgesteuert für 2 - 70 min. passiv (damit man nach Platzierung der Schaltung erst mal "das Weite suchen kann"). War noch keine HID- Maus in dem USB- Port des Rechners eingesteckt, ploppt danach der Treiberdialog auf. Windows hat zwar den passenden Treiber (HID- Treiber sind Teil der Windows- Installation), er ist aber noch nicht an den benutzten USB- Port gebunden. Das Opfer klickt dann hoffentlich auf "automatisch installieren" - die meisten Windows- Opfer sind ja konditioniert, kryptische Windows- Dialoge mit "ja" oder "weiter" wegzuklicken.

Danach beginnen entweder das Mauszittern oder die unregelmäßigen Sprünge des Mauszeigers.

Nach einem Reboot oder Kaltstart läuft zunächst wieder die trügerische Wartezeit ab, bis die Schaltung erneut zuschlägt.