

Toetsenbord in een USB-stick

Toetsenbord-emulator met AVR-controller en USB

Markus Hirsch
(Duitsland)

Voor de gemiddelde hobbyist lijkt RS232 vaak aantrekkelijker dan USB. Maar tegenwoordig is het ook mogelijk om met een kleine 8-bits microcontroller een USB-bus in device-modus te implementeren. De voordelen hiervan zijn duidelijk. In de eerste plaats kunnen we besparen op de kosten van een USB/RS232-converter, en daarnaast kunnen USB-apparaten zoals toetsenborden eenvoudig worden geëmuleerd zonder dat daar een aparte driver voor nodig is. We gaan in dit project met minimale hardwarekosten en software-inspanningen een programmeerbaar USB-toetsenbord maken. En zoals u zult zien is zo'n veelzijdige 'USB keyboard stick' niet alleen geschikt voor nuttige toepassingen, je kunt er ook plezier mee hebben!

Wat is het (de hardware)

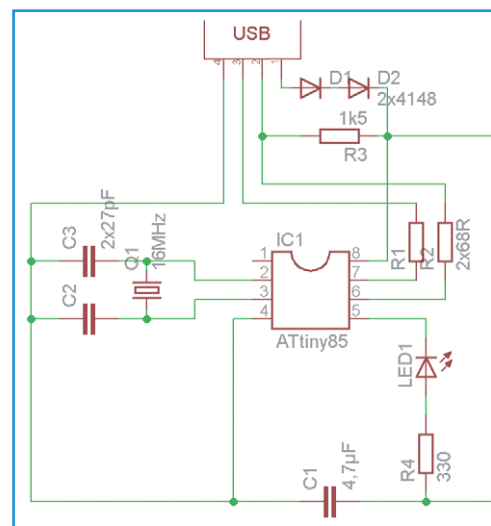
Zoals gezegd is de hardware simpel en goedkoop (**figuur 1**). Het hart van de USB keyboard stick is een Atmel AVR ATtiny85. De 5 V voedingsspanning wordt met standaard 1N4148 diodes (D1 en D2) gereduceerd tot ongeveer 3,6 V, en condensator C1 zorgt voor spanningsstabilisatie. Met R3 wordt de USB-snelheidsindicatie op low-speed ingesteld. Een 16 MHz kristal genereert een nauwkeurig kloksignaal. Het is ook mogelijk om dit kristal uit te sparen en de interne oscillator te gebruiken (op 16,5 MHz). Een LED met R4 als serieweerstand dient als statusindicator.

De schakeling kan snel en compact worden opgebouwd op gaatjesprint met een USB-A connector (zie **figuur 2**). Een SMD-versie met geëtste print is nog kleiner, en past makkelijk in de behuizing van een oude USB-geheugenstick.

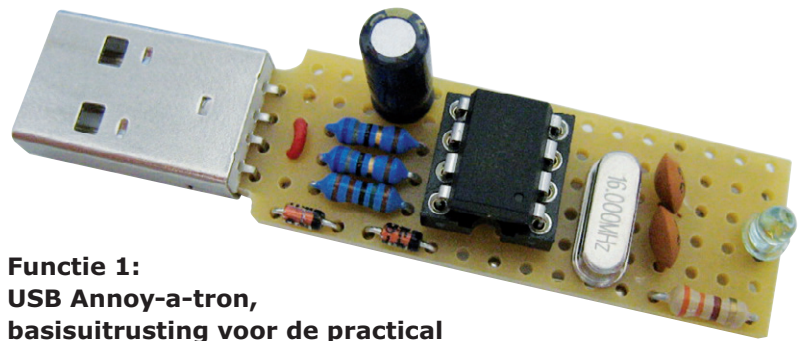
Door ruimtegebrek is afgezien van een SPI-header. De ATtiny85 kan met drie draden via de debug wire (dW) en de voedingsaansluiting worden geprogrammeerd. Hiervoor worden de aansluitingen met de controller tijdelijk met testclips gemaakt.

Wat doet het (de software)

De software werd in C ontworpen met het welbekende en gratis Atmel Studio 6 [1]. De kern van de software is de USB Device Stack. Deze is ook als aparte gratis bibliotheek beschikbaar. V-USB kan direct van de website van de ontwikkelaar worden gedownload [2] waar het een aparte microsite heeft [3]. Hier zijn ook handige voorbeelden te vinden. Met deze voorbeelden als uitgangspunt kunnen we USB-projecten



Figuur 1.
Schema van de USB
Keyboard Stick.



maken zonder ons teveel in het USB-protocol te hoeven verdiepen.

Als de USB-stack correct is geconfigureerd, wordt de AVR microprocessor door de meeste besturingssystemen gedetecteerd zonder dat daar een aparte driver voor nodig is, en vervolgens wordt de status van de LED's bij de caps, num en scroll lock toetsen naar de stick gestuurd.

Om toetsenbordaanslagen te emuleren worden toetscodes volgens de **tabellen 1, 2 en 3** naar de pc gestuurd. Hierbij kan ook onderscheid worden gemaakt tussen bijvoorbeeld een hoofdletter 'A' of een kleine letter 'a'.

Let op! Deze codes zijn niet gelijk aan de ASCII-codes voor de betreffende karakters.

We geven hieronder twee toepassingen van de USB keyboard stick. De code voor de voorbeelden is zowel voor AVR-studio als in de vorm van hex-bestanden beschikbaar op de website van Elektor.LABS [6].

**Functie 1:
USB Annoy-a-tron,
basisuitrusting voor de practical
joker**

De Annoy-a-tron is een populaire vorm van verstoppertje spelen, ontworpen door ThinkGeek [4]. In Elektor publiceerden we drie jaar geleden al een goedkope kleinere versie [5]. Nu introduceren we de USB-versie van de Annoy-a-tron, gebaseerd op een geheel ander concept en uitgerust met nog betere hinderlijke eigenschappen waar uw collega's en familieleden volkomen gek van zullen worden. Een bekende grap is om de letters 'v' en 'b' die in een standaard toetsenbord dicht bij elkaar zitten fysiek om te wisselen. De USB Annoy-a-tron gaat veel verder, let maar op...

In feite is de USB Annoy-a-tron een permanente caps lock. Zodra de stick in de pc is geplaatst zal deze als een toetsenbord worden herkend. Daarna zal de pc er de status van de drie LED's (caps, num en

Figuur 2.
De USB Keyboard
Stick kan eenvoudig op
gaatjesprint worden
gebouwd.

Tabel 1. Codes van de USB-toetsenbord karakters (standaard toetsen)

| | | | | | | | | | | | | | |
|----------|----------|----------|----------|----------|----------|----------|----------|--------------|--------------|------------|------------------|-----------------|--------------------|
| A | B | C | D | E | F | G | H | I | J | K | L | M | N |
| 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
| O | P | Q | R | S | T | U | V | W | X | Y | Z | 1 | 2 |
| 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | Enter | Space | Esc | Caps Lock | Num Lock | Scroll Lock |
| 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 44 | 41 | 57 | 83 | 71 |

**Tabel 2.
Codes van de USB-toetsenbord karakters (speciale toetsen)**

| | | | | | | | |
|-------------|--------------|------------|------------|----------------|-----------------|---------------|---------------|
| Ctrl | Shift | Alt | GUI | R. Ctrl | R. Shift | R. Alt | R. GUI |
| 0x01 | 0x02 | 0x04 | 0x08 | 0x10 | 0x20 | 0x40 | 0x80 |

**Table 3.
Bitcodes van de USB-toetsenbord LED's**

| | | |
|-----------------|------------------|--------------------|
| Num Lock | Caps Lock | Scroll Lock |
| 0x01 | 0x02 | 0x04 |

scroll lock) naartoe sturen. De stick houdt nu het bit in de gaten dat bij de caps lock toets hoort, en zodra deze toets wordt ingedrukt om caps lock uit te schakelen, zal de USB Annoy-a-tron een virtuele toetsaanslag simuleren met als resultaat dat caps lock weer aan gaat (**figuur 3**) en de gebruiker niets anders kan doen dan de grap lijdzaam ondergaan.

Nu moeten we de stick alleen nog op een strategische plaats aanbrengen, bijvoorbeeld aan de achterkant van een pc. En als je gemeen genoeg bent is het nog leuker om hem via een kleine kabeladapter binnenin een pc te plaatsen.

Functie 2: Masterkey, biedt professionele beveiliging

De USB Annoy-a-tron is een leuke grap, maar we moeten toegeven dat het gezien de mogelijkheden van dit apparaat niet echt een nuttige toepassing is. Daarom hebben we als tweede functie voor een wachtwoord-opslagsysteem gekozen. Tegenwoordig zijn wachtwoorden noodzakelijk, maar ze kunnen ook een probleem vormen. Waarschijnlijk heeft u meerdere wachtwoorden, maar sterke wachtwoorden zijn vaak niet makkelijk te onthouden. Voor dit doel zijn er wachtwoordmanagers zoals KeePass, maar die hebben zelf ook een sterk masterwachtwoord nodig. Hier komt de USB Masterkey in beeld. Daarmee introduceren we een belangrijke functie die niet aanwezig is in andere USB wachtwoord-opslagapparaten. Om het wachtwoord uit te lezen is het niet voldoende om alleen maar de USB-stick in de pc te plaatsen. In ons voorbeeld wordt gebruik gemaakt van een bepaalde volgorde van oplichten van de drie toetsenbord-LED's die in het bestand 'code.h' beschreven wordt. De volgorde is: Num Lock, Num Lock + Caps Lock, Num Lock. Dat betekent dat de invoer van de code al begonnen moet zijn voordat de USB-stick ingestoken wordt (Num Lock moet actief zijn vóór de stick wordt ingestoken). Dan kan de stick ingestoken worden en gaat de code verder met Caps Lock (is dus Num Lock + Caps Lock) en nogmaals Caps Lock (is dus alleen Num Lock). Deze korte reeks kan eenvoudig worden onthouden. Als er een verkeerde combinatie wordt ingevoerd, dan moet de stick eerst uit de pc wor-



Figuur 3.
Caps lock flitst weer aan op het toetsenbord. Leuk voor ons, niet voor het slachtoffer.

den gehaald en vervolgens weer worden teruggeplaatst.

Het is allemaal niet moeilijk

Zoals gezegd zijn alle bestanden beschikbaar op de Elektor.LABS website van dit project [6], inclusief de EAGLE-bestanden. We hebben hier laten zien dat de implementatie van een geëmuleerd toetsenbord niet moeilijk is, en de mogelijkheden van een 'vermomde' microcontroller zijn bijna ongelimiteerd.

Een suggestie van onze kant voor een derde toepassing zou kunnen zijn om verschillende wachtwoorden in één stick op te slaan en deze met verschillende combinaties van toetsaanslagen uit te lezen. Tenminste, als u een nuttige toepassing wilt. Zo niet, dan is elk ander grappig apparaat goed. Grappen zijn altijd welkom... tenzij wij het slachtoffer worden!

(120583)

Internet Links

- [1] www.atmel.com/microsite/atmel_studio6
- [2] www.obdev.at
- [3] www.obdev.at/products/vusb/index.html
- [4] www.thinkgeek.com/product/8c52/
- [5] www.elektor.com/090084
- [6] www.elektor-labs.com/120583

Onderdelenlijst

Weerstanden

R1, R2 = 68R, 0.25W, 1%
R3 = 1.5K, 0.25W, 1%
R4 = 330R, 0.25W, 1%

Condensatoren

C1 = elektrolytisch, 4.7uF, 100V
C2, C3 = keramisch, 27pF, 100V

Halfgeleiders

D1, D2 = 1N4148
LED1 = LED, 5mm, geel

Diversen

IC1 = ATtiny85
X1 = kristal 16MHz
K1 = USB A connector