Logic Analyzer mit CY7C68013A von Cypress

Bei diesem Projekt geht es weniger um den Eigenbau als um ein bestehendes System umzubauen. Die Basis stellt bei diesem Analyzer der bereits erwähnte Chip von Cypress dar, der über einen 8051-Kern verfügt. Dieser besitzt eine USB-Verbindung, aber nur Ram, kein Flash. Die Firmware wird erst beim Start der Software geladen.

Die Erkennung der Hardware erfolgt durch die USB VID:PID, die in einem externen eeprom abgelegt ist. Aus diesem Grund können Geräte mit solchen ICs relativ einfach nachgebaut werden ;)

Ursprünglich wurde mit einem eigenen Nachbau von Stefan (&co.) experimentiert, der aber nur bis 1 MHz arbeiten möchte. Die Ursache dafür ist aber noch nicht gefunden. Als funktionstüchtiger Ersatz



wurden drei USBee AX PRO clone von DX geordert. (zu ca. 10 € pro Stück) Dieser soll mit der Software von USBee zusammen arbeiten, da die aber nicht unter Linux läuft und auch sigrok den so nicht akzeptieren möchte, wurde der Inhalt der eeproms angepasst.

EEPROM Inhalte und Änderungen

Orginaler eeprom Inhalt von allen 3 Analyzern:

```
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f    0123456789abcdef
00: c0 a9 08 14 00 00 1b 00 00 00 00 00 00 00 00 00    ????..?.........
10: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
20: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
30: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
40: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
50: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
60: 00 00 14 00 00 00 14 00 00 00 00 80 00 00 00 80    ..?...?....?...?
70: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
Rest: 0xff
```

Die USB VID:PID ist damit lsusb: 08a9:0014

Laut einer nicht mehr bekannten Internetquelle soll man den Inhalt des eeproms für die Saleae Software mit folgendem Inhalt füllen:

```
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f    0123456789abcdef
00: c0 25 09 81 38 10 00 08 ff ff ff ff ff ff ff ff    ?%??8?.?........
10: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
Rest: 0xff
```

Die USB VID:PID ändert sich damit zu lsusb: 0925:3881 Lakeview Research
Nach dem Start von Saleae Logic (64) 1.1.15 ändert sich der Inhalt von allen 3 Analyzern zu:

Analyzer (1)

```
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f    0123456789abcdef
00: c0 25 09 81 38 10 00 08 e1 e9 4f f3 6f 6d 2e c3    ?%??8?.???O?om.?
10: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
Rest: 0xff
```

Analyzer (2)

```
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f    0123456789abcdef
00: c0 25 09 81 38 10 00 08 39 17 e2 fe e8 e6 54 a6    ?%??8?.?9?????T?
10: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
Rest: 0xff
```

Analyzer (3)

```
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f    0123456789abcdef
00: c0 25 09 81 38 10 00 08 fa 0e 1b 25 9f f0 76 d2    ?%??8?.????%??v?
10: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
Rest: 0xff
```

Mit allen 3 Analyzern funktioniert Saleae Logic mit 24 MHz und 8 Kanälen problemlos.

Auf Basis dieser Infos wurde der Eigennachbau noch mal untersucht: Inhalt des eeprom:

```
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f    0123456789abcdef
00: c0 25 09 81 38 00 00 00 4f 83 c5 78 a4 22 6f 96    ?%??8...O??x?"o?
10: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
20: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
30: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
40: 02 02 00 02 00 07 05 88 02 00 02 00 09 02 20 00    ??.?.????.?.?? .
50: 01 01 00 c0 00 09 04 00 00 02 08 06 50 00 07 05    ??.?.??..???P.??
60: 02 02 40 00 00 07 05 88 02 40 00 00 04 03 09 04    ??@..????@..????
70: 2c 03 43 00 79 00 70 00 72 00 65 00 73 00 73 00    ,?C.y.p.r.e.s.s.
80: 20 00 53 00 65 00 6d 00 69 00 63 00 6f 00 6e 00     .S.e.m.i.c.o.n.
90: 64 00 75 00 63 00 74 00 6f 00 72 00 2c 03 55 00    d.u.c.t.o.r.,?U.
a0: 53 00 42 00 32 00 2e 00 30 00 20 00 53 00 74 00    S.B.2...0. .S.t.
b0: 6f 00 72 00 61 00 67 00 65 00 20 00 44 00 65 00    o.r.a.g.e. .D.e.
c0: 76 00 69 00 63 00 65 00 1a 03 44 00 45 00 46 00    v.i.c.e.??D.E.F.
d0: 31 00 30 00 39 00 44 00 41 00 35 00 37 00 37 00    1.0.9.D.A.5.7.7.
e0: 44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    D...............
f0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ff    ................
```

Zu erkennen ist dass hier leicht andere Infos in der ersten Zeile stehen und sonst auch deutlich mehr im eeprom enthalten ist. Nach dem flashen mit dem Inhalt aus dem Analyzer (1) bleibt dieser Inhalt konstant bei:

```
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f    0123456789abcdef
00: c0 25 09 81 38 10 00 08 e1 e9 4f f3 6f 6d 2e c3    ?%??8?.???O?om.?
10: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff    ................
Rest: 0xff
```

Die Funktion (oder nicht Funktion) und die USB VID:PID ändern sich aber dadurch nicht!

Hardware Vergleich (nicht komplett)

Eigenbausystem:

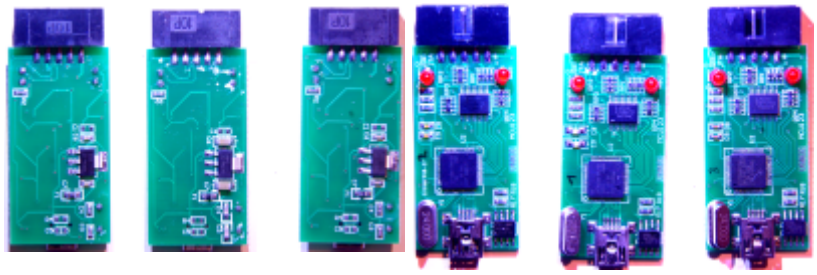
- CY7C68013A-56PVXC
- eeprom: (CAT) 24WC02J
- Eingangstreiber: (TI) SN74LVC8T245

Näheres: USB-Blaster

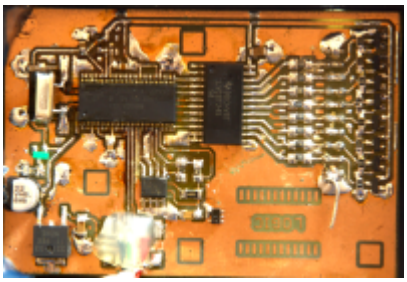
USBee AX PRO (China):

- CY7C68013A-56LTXC (bzw. -56LTXI)
- eeprom: (Atmel) 24C02BN
- Eingangstreiber: (NXP) 74HC245

Die drei AX PRO in Bildern:



Geringe Abweichungen im Routing und von den ICs sind hier erkennbar ;)

Die gebaute Platine: 

mögliche Mods

- umschaltbare eeproms → Saleae und USBee möglich
  - das ist mit der Platine von YC möglich, wurde auch schon implementiert
- umschaltbare Ein-/Ausgangstreiber → LA und Pattern Generator via USBee möglich
  - dafür ATTiny45 auf Platine von YC bestücken, der per I2C Kommando zum Umschalten bekommt

From:
https://www.loetlabor-jena.de/ - **Lötlabor Jena**

Permanent link:
https://www.loetlabor-jena.de/doku.php?id=projekte:la_cy7c68013a:start&rev=1428659280

Last update: **2015/04/10 09:48**