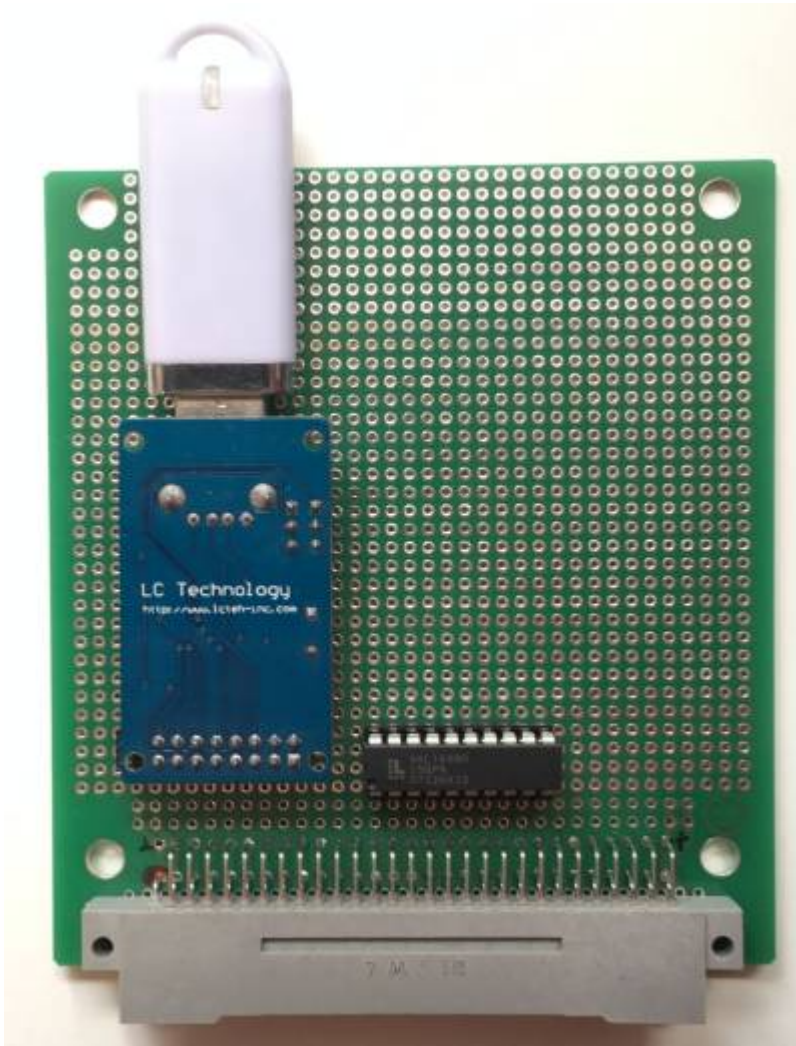


USB-Modul mit CH376

Als Alternative zum VDIP, und mit wesentlich weniger zusätzlicher Hardware (nur Port-Selektion nötig, die restl. Anschlussleitungen D7..D0, RD, WR, A0 gehen direkt zum Z80-BUS)

Das Modul ist aktuell in Entwicklung. Geplant sind noch 32K RAM (4000-BFFF) und 8K ROM (C000-E7FF) mit USB-OS auf dem Modul.

 **Fix Me!**

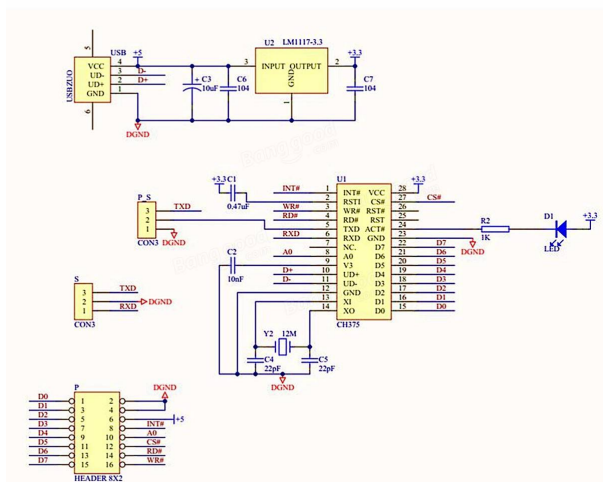
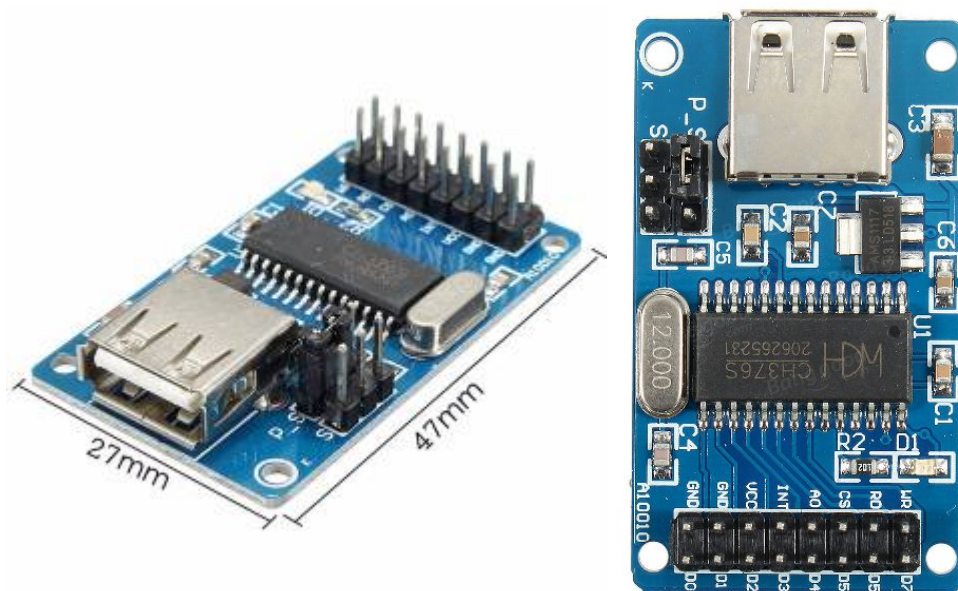


CH376

Der CH376 ist ein Controller für kleine Mikrocomputersysteme zum Lesen und Schreiben von Dateien auf USB-Disk oder SD-Karte. Der CH376 unterstützt den USB-Gerätmodus und den USB-Host-Modus, ist dabei kompatibel mit USB V2.0. Der CH376 unterstützt drei Kommunikationsschnittstellen: 8-Bit-parallel, SPI oder asynchron seriell. Mikrocomputersysteme können den CH376-Chip über eine der genannten Kommunikationsschnittstellen steuern und auf Dateien oder Dateien auf USB-Disk oder SD-Karte zugreifen. Der CH376 unterstützt FAT16 und FAT32, allg. das FAT12-Dateisystem mit Unterverzeichnissen und kurzen (8.3)-Dateinamen.

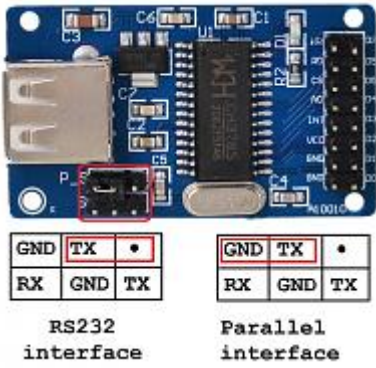
Der USB-Gerätmodus von CH376 ist vollständig mit dem CH372-Chip kompatibel, und der USB-Hostmodus von CH376 ist grundsätzlich mit dem CH375-Chip kompatibel.

Man könnte den Chip direkt verwenden, er kann auch mit +5V betrieben werden, doch für ca 3 Euro gibt es fertig aufgebaute Module, die per Pfostenstecker mit der eigenen Hardware verbunden werden. Ich habe mich für folgende Variante entschieden (CH375-kompatible Module):

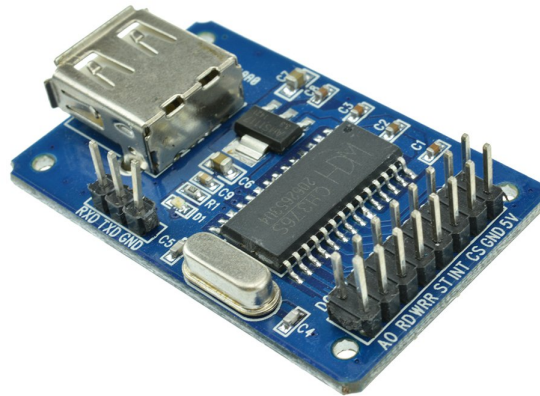
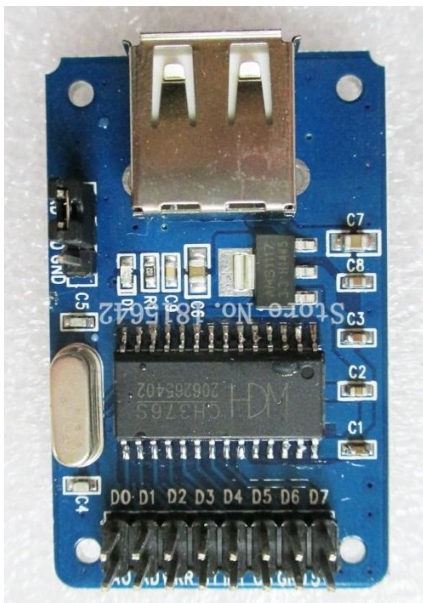


Modul, Draufsicht, Schaltplan (gilt auch für die Variante mit 376S)

Bedeutung der Steckverbinder:



Achtung: Es gibt ähnliche Module, die aber eine andere Steckverbinderbelegung und auch nur einen Jumper haben. Das muss bei der Verdrahtung beachtet werden! Die meisten der bei ebay angebotenen „CH376S U DISK READ WRITE MODULE“ sind von dieser folgenden Variante:



Größe: 50mm x 34mm

Vermutlich ist das die bessere Lösung, da leichter erhältlich, und es gibt auch einen Reset-Pin.

Pin-Reihen

die beiden oberen Reihen: Variante 1 (ohne Reset, mit 6pol. Jumperfeld)
 die beiden unteren Reihen: Variante 2 (mit Reset, mit 3pol. Jumperfeld)

Pins spiegelverkehrt zum Modul, das wird ja mit der Leiterseite nach oben aufgesteckt.

```

WR  RD  CS  A0  INT  VCC  GND  GND
D7  D6  D5  D4  D3  D2  D1  D0
5V  GND  CS  INT  RST  WR  RD  A0
    
```

Unterlagen

- Produktseite: <http://www.wch.cn/product/CH376.html>
- Anleitung CH376 (engl): ch376ds1.pdf http://www.wch.cn/downloads/CH376DS1_PDF.html

- zusätzliche Unterlagen + C-Code f. Microcontroller (8051), leider in chinesisich:
http://www.wch.cn/downloads/CH376EVT_ZIP.html

Es gibt auch den zweiten Teil der Anleitung CH376DS2.PDF auf obiger Produktseite, ebenfalls leider nur in chinesisich: Beschreibung der grundlegenden Übertragungsbefehle und Beschreibung der externen Firmware des Gerätemodus. Mit google translate kann man die PDF übersetzen.

Arduino

- Arduino-Library: <https://github.com/djuseeq/Ch376msc>

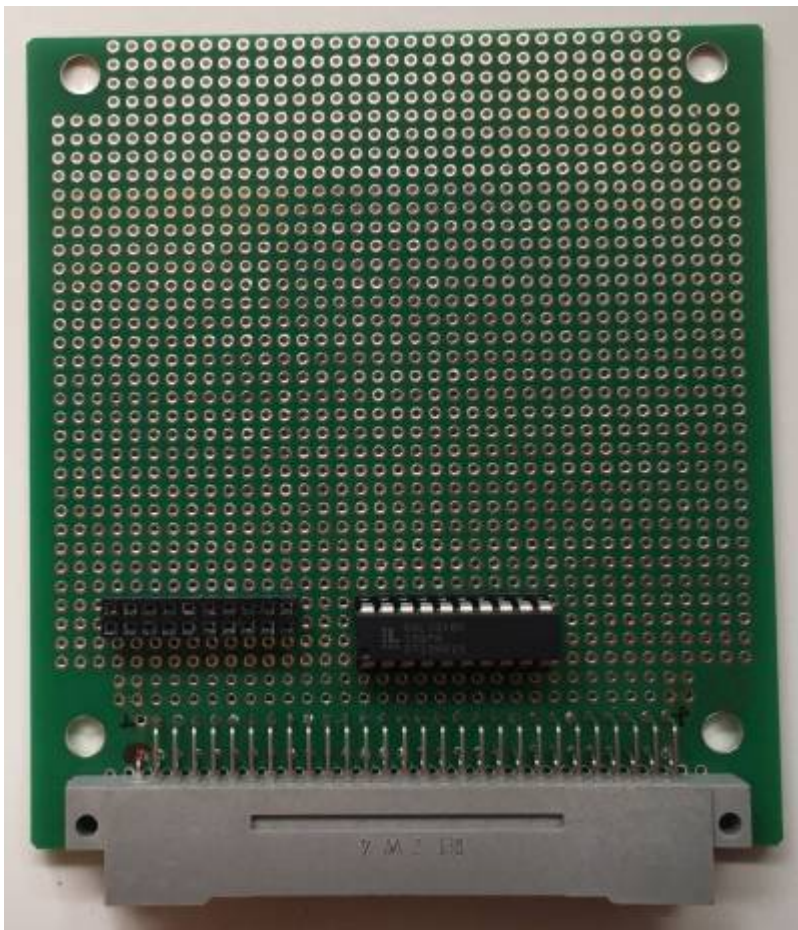
Z80-Anschluss

Bruce Abbott hat für seinen Mattel Aquarius ein Micro-Expander-Modul entwickelt. Sein Modul umfasst 32k RAM, 4x16k ROM und ein CH376-USB-Modul. Auf einer zweiten Leiterplatte ist ein Soundchip AY-3-8910 und ein zweites CH376-USB-Modul. Auf der Webseite gibt es komplette Z80-Assemblerquellen.

* http://www.bhabbott.net.nz/micro_expander.html

Inbetriebnahme

Das CH376-Modul wird im Parallel-Modus betrieben (JP P_S Stellung P)



Erster Test:



die minimalistische Hardware funktioniert

Der GAL selektiert lediglich Port 40h+41h, ansonsten gehen die Anschlüsse des Moduls direkt an den BUS. Anstelle eines GAL können auch diskrete TTL-Schaltkreise für die Portselektion und Ansteuerlogik genutzt werden.

Mit einfachen I/O-Abfragen kann die Funktion geprüft werden

```
IN 41 --> 80 device status
OUT 41 1
IN 40 --> 43 chip version (>40)
OUT 41 6
OUT 40 55
IN 40 --> AA (bit reverse)
```

GAL-Logik:

```
CH376 = /IORQ * /RD * /A7 * A6 * /A5 * /A4 * /A3 * /A2
        + /IORQ * /WR * /A7 * A6 * /A5 * /A4 * /A3 * /A2; /IORQ * (/RD + /WR) *
Port 40h
```

CP/M

Mittlerweile ist das Modul bei den Z80-Fans bekannt und wird auch genutzt. Ich habe zwei interessante Ansätze zum Thema CP/M gefunden:

- <https://github.com/gotaproblem/Z80Playground>
Ein orig. CP/M mit Diskettenimages A.DSK ... D.DSK auf dem USB-Stick
- <https://github.com/z80playground/cpm-fat>
ein neues BDOS, das direkt auf das FAT-Filesystem zugreift. Die CP/M-Dateien liegen so alle in Verzeichnissen A\...P\ auf dem USB-Stick. Allerdings werden keine direkten BIOS-Aufrufe unterstützt?

Last update:
2021/07/13 11:20 z9001:module_sonstige:usbmodul https://hc-ddr.hucki.net/wiki/doku.php/z9001/module_sonstige/usbmodul?rev=1626175236

From:
<https://hc-ddr.hucki.net/wiki/> - **Homecomputer DDR**

Permanent link:
https://hc-ddr.hucki.net/wiki/doku.php/z9001/module_sonstige/usbmodul?rev=1626175236

Last update: **2021/07/13 11:20**

