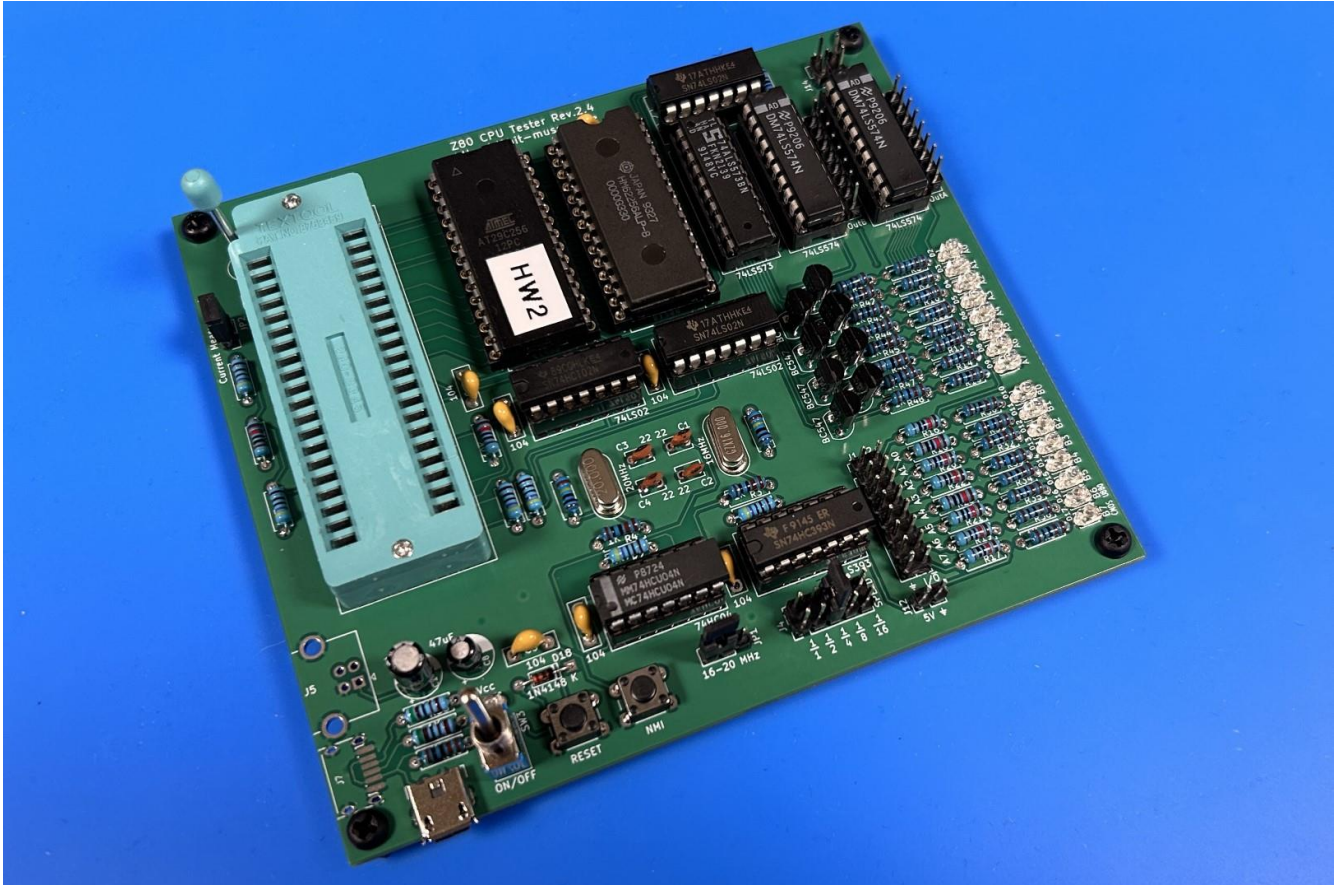


Version: 2025-07-03

# Z80 CPU TESTER (HW 2)

<http://8bit-museum.de>

## HANDBUCH



© Soft- und Hardware: Stephan Slabihoud, 8Bit-Museum.de

Alle Rechte vorbehalten. Kein Teil dieses Handbuchs, einschließlich der darin beschriebenen Produkte und Software, darf ohne ausdrückliche schriftliche Genehmigung des Entwicklers in irgendeiner Form oder auf irgendeine Weise reproduziert, übertragen oder kommerziell verwendet werden.

Die Verwendung der Hard- und Software geschieht auf eigene Gefahr. Es wird keine Haftung für etwaige Schäden übernommen, die beim Einsatz der vorliegenden Soft- und Hardware entstehen könnten. Es wird keine Garantie für die korrekte Funktionsfähigkeit der bereitgestellten Funktionen übernommen. Es besteht auch kein Anspruch auf eine Fehlerbehebung, auch wenn der Entwickler bemüht ist Fehler schnellstmöglich zu beseitigen. Sowohl Soft- als auch Hardware sind „closed source“ und urheberrechtlich geschützt. Ausgenommen davon ist die Hard- und Software, für die explizit eine andere Lizenz gilt.

Diese Seite wurde absichtlich leer gelassen

## Inhaltsverzeichnis

<b>1</b>	<b>Verwendung .....</b>	<b>5</b>
1.1	Einleitung .....	5
1.2	Repository und Lizenzen .....	5
1.3	Durchgeführte Tests .....	6
1.4	Einstellen der Taktfrequenz .....	6
1.5	Strommessung .....	6
1.6	Anzeige der Testergebnisse .....	7
1.6.1	Ausgabe "Port B" .....	7
1.6.2	Ausgabe "Port A" .....	8
1.6.3	Eingabe „Port A“ .....	8
1.6.4	Pin-Header von Port A und Port B (Out A, Out B).....	9
1.6.5	Pin-Header von Port A (I/O) .....	9
1.6.6	Ein- Ausgabe Port A und Port B .....	10
1.6.7	Ausgabe XF Ergebnis und XF/YF Zähler .....	11
1.7	Messung des Stromverbrauchs .....	11
<b>2</b>	<b>Bill of materials (BOM) Ergänzung .....</b>	<b>12</b>
2.1	USB-Buchse .....	12

Diese Seite wurde absichtlich leer gelassen

# 1 Verwendung

## 1.1 Einleitung

Mit dem Z80 CPU Tester können Z80 CPUs und kompatible CPUs nicht nur getestet werden, sondern es wird auch versucht den CPU- Hersteller und die Herstellungstechnologie (NMOS/CMOS) zu erkennen.

Der Tester versucht

- die CPU-Technologie (NMOS/CMOS) zu identifizieren,
- den CPU-Hersteller zu identifizieren und
- zu testen, ob die CPU noch funktionsfähig ist.

und hilft dabei die maximal mögliche CPU-Frequenz herauszufinden.

Folgendes wird nicht getestet:

- Interrupts,
- WAIT- und HALT-Eingangssignale,
- M1- und RFSH-Ausgangssignale und
- BUSRQ- und BUSACK-Signale.

Die Hardware hat folgende Features:

- 32 kb EPROM (27C256), alternativ kann ein AT29C256 verwendet werden
- 32 kb SRAM
- Ausgangsports (1x unidirektional, 1x bidirektional mit read-back) mit 16 LEDs
- Eingangsport (1x open-collector, kombiniert mit Ausgang A)
- RESET- und NMI-Taste
- 16 MHz / 20 MHz umschaltbar
- 1, 1/2, 1/4, 1/8, 1/16 Multiplikator (1-16 MHz / 1,25-20 MHz)
- Anschluss für CPU-Strommessung
- Stromversorgung über USB
- THT für einfaches Löten

## 1.2 Repository und Lizenzen

Die Gerber-Dateien zum Erstellen der Platine sowie die benötigte Software ist auf Github verfügbar. Alternativ kann die Platine auch von mir bezogen werden.

Das Repository ist unter

<https://github.com/slabb/Z80-CPU-Tester>

zu finden.

Zu den Lizenzen:

- Die Hardware darf zu privaten Zwecken frei verwendet werden. Eine kommerzielle Nutzung ist aber untersagt.
- Die Software, sofern sie von mir geschrieben wurde, unterliegt der freizügigen MIT Lizenz. Fremder Code kann aber andere Lizenzen verwenden, deshalb bitte die Hinweise im Quellcode beachten.

## 1.3 Durchgeführte Tests

Folgende Tests werden durchgeführt:

- Speicherzugriff, 16-Bit-Register load
- RL, RR, Register to Register load
- ADD, SUB
- PUSH, POP, SBC
- ADD, SUB, INC, DEC
- einige 16-Bit-Multiplikationen (ADD, RL)
- einige 32-Bit-Multiplikationen (ADD, ADC, SBC, PUSH, POP, RRA, EX)
- berechnet einige Quadratwurzeln (BIT, RL, ADD, SUB)
- spielt Towers of Hanoi (PUSH, POP, CALL)
- berechnet Pi (EX, EXX, IX, IY, INC, DEC, ADD, ADC, SBC, SRL, RR, PUSH, CALL)

Pi wird auf 100 Stellen berechnet. Dies dauert bei 4 MHz etwa 30 Sekunden. Wenn eine CPUs bei niedrigeren Geschwindigkeiten getestet werden soll, sollte die Anzahl der berechneten Stellen ggf. reduziert werden.

## 1.4 Einstellen der Taktfrequenz



Es müssen zwei Jumper gesetzt werden:

Links („16-20 MHz“) erfolgt die Auswahl des Taktgenerators. Befindet sich der Jumper in der linken Position wird der 16 MHz Quarz ausgewählt, in der rechten Position der 20 MHz Quarz.

Rechts („Speed“) erfolgt die Wahl des Taktfrequenzteilers. Es kann bis auf 1/16 Takt heruntergeteilt werden, d.h. man erhält Frequenzen von 1, 2, 4, 8, 16 MHz bzw. 1.25, 2.5, 5, 10, 20 MHz.

## 1.5 Strommessung



Der Jumper „Current measurement“ muss im Normalbetrieb geschlossen sein. Sie können die Verbindung öffnen und ein Strommessgerät anschließen, um den Stromverbrauch der CPU zu messen.

## 1.6 Anzeige der Testergebnisse

Es gibt zwei Ports (A und B), die zur Anzeige von Status und Ergebnis mithilfe von 16 LEDs dienen.



### 1.6.1 Ausgabe "Port B"

Port B zeigt die identifizierte Technologie und den Hersteller an:

```
CU00tttt (C = CMOS, U = UB880, tttt = type)
```

#### CMOS-LED (C)

Die LED zeigt das Ergebnis des Tests „OUT (C),0“ an.

- Wenn „an“, dann wurde eine CMOS-CPU erkannt (OUT (C),0 gibt \$ff aus).
- Wenn „blinkend“, dann wurde eine CMOS-CPU erkannt (OUT (C),0 gibt \$00 aus).
- Wenn „aus“, dann wurde eine NMOS-CPU erkannt (OUT (C),0 gibt \$00 aus).

Hinweis: Beim Sharp LH5080A (CMOS-Version vom LH0080A) schlägt der CMOS-Test fehl und der nicht dokumentierte Befehl OUT (C),0 verhält sich genauso wie bei einer NMOS-CPU. Die LED blinkt erst, nachdem alle Funktionstests abgeschlossen sind; sie blinkt nicht, wenn die Funktionstests noch laufen.

#### U880-LED (U)

Die LED zeigt das Ergebnis des U880-Tests an.

- Wenn „an“, dann wurde eine U880-CPU erkannt (U880, UA880, UB880 und ähnliche), andernfalls ist die LED „aus“.

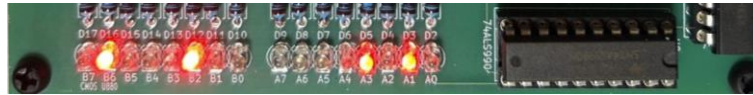
#### Typ (tttt)

Die unteren vier LEDs zeigen die identifizierte CPU an. Der Test kann keine 100%ige Genauigkeit garantieren, da die Erkennung auf der Auswertung nicht dokumentierten Verhaltens der CPU basiert.

0000	not used
0001	Z180
0010	Z280
0011	EZ80
0100	U880 (newer; MME U880, Thesys Z80, Microelectronica MMN 80CPU)
0101	U880 (older; MME U880, KR1858VM1)
0110	SHARP LH5080
0111	NMOS Z80 (Zilog Z80, Zilog Z08400 or similar NMOS CPU, Mosstek MK3880N, SGS/ST Z8400, Sharp LH0080A, KR1858VM1)
1000	NEC D780C (NEC D780C, GoldStar Z8400, possibly KR1858VM1)
1001	KR1858VM1 (overclocked)
1010	Unknown NMOS Z80 Clone
1011	CMOS Z80 (Zilog Z84C00)
1100	Toshiba Z80 (Toshiba TMPZ84C00AP, ST Z84C00AB)
1101	NEC D70008AC
1110	Unknown CMOS Z80 Clone
1111	NEC Z80 Clone (NMOS)

**Beispiel**

Zilog Z84C0020PEC von einem chinesischen Marktplatz. Es handelt sich um eine U880 (mit 10 MHz):



Die identifizierte U880 zeigt 01000100 (Port B) an. „U“ ist eingeschaltet, „tttt“ ist 0100.

**1.6.2 Ausgabe “Port A”**

Port A zeigt den aktuellen Status der Funktionstests an. Während Port B das Ergebnis sofort anzeigt, dauern die Funktionstests einige Zeit, daher zeigt Port A als Testfortschritt die Testnummer an.

Wenn die Funktionstests abgeschlossen sind:

- wird die Nummer des fehlgeschlagenen Tests mit einem blinkenden Bit 7 angezeigt, oder
- zeigt ein Lauflicht ein erfolgreiches Ergebnis (kein Fehler) an.

Im Bild oben zeigt Port A an, dass der Funktionstest Nr. 10 noch läuft.

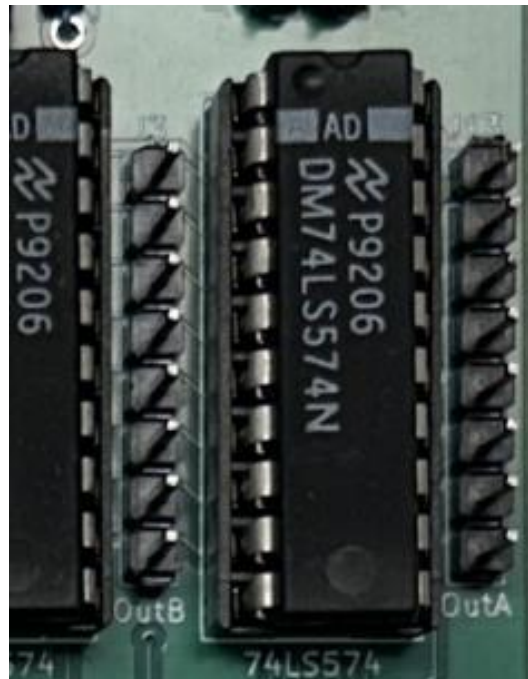
**1.6.3 Eingabe „Port A“**

Über Port A kann der im Latch-Register von Port A gespeicherte Wert zurückgelesen werden.



### 1.6.4 Pin-Header von Port A und Port B (Out A, Out B)

Die Ausgabe von Port A und Port B ist über einen Pin-Header verfügbar.



Es handelt sich um einen reinen Ausgabeport, der vom 74LS574 getrieben wird. Ein Kurzschluss nach Masse oder 5V kann den Port zerstören. Die Port A und Port B LEDs (A0-A7, B0-B7) sind jeweils am Pin-Header angeschlossen.

### 1.6.5 Pin-Header von Port A (I/O)

Die Ausgabe von Port A ist über einen Pin-Header **invertiert** verfügbar, d.h. leuchtet die entsprechende LED, liegt am Pin-Header ein Low-Signal an, ist die LED aus, liegt ein High-Signal an (open-collector Ausgang).



Der Port ist als bidirektionaler Port ausgelegt, d.h. es können am Pin-Header Signale sowohl ausgegeben als auch eingelesen werden.

Durch die open-collector Auslegung, kann der Eingang mit einem Taster direkt auf Masse gezogen werden (logisch „0“). Im offenen Zustand wird durch die 10 kOhm Pull-Ups ein logisch „1“ zurückgeliefert.

Damit ein Wert eingelesen werden kann, muss das Ausgabelatch an Port A zunächst auf „0“ gesetzt werden. Die Transistoren sperren und am Pin-Header liegt durch die Pullup-Widerstände ein High-Signal an. Danach kann der Wert gelesen werden.

**Beispiele:**

```
XOR  A
OUT  (PORTA), a
loop:
IN   A, (PORTA)
OUT  (PORTB), A
JR   loop
```

**1.6.6 Ein- Ausgabe Port A und Port B**

	Port A (Out A, LED)	Port A (Header)	Port B (Out B, LED)
Ausgabe	nicht invertiert	invertiert (*)	nicht invertiert
Eingabe	-----	nicht invertiert	-----

(\*) Wenn ein Wert auf “Port A” ausgegeben wird, ist dieser Wert verfügbar am Pin-Header “Out A” und invertiert am Pin-Header “I/O”.

### 1.6.7 Ausgabe XF Ergebnis und XF/YF Zähler

Die CPU-Identifizierung basiert auf der Auswertung von nicht dokumentiertem Verhalten der XF/YF-Flags. Das interne XY-Ergebnis und die XF/XF-Zähler können durch Drücken der NMI-Taste angezeigt werden.

Nach dem Drücken der NMI-Taste blinkt der CPU-Tester nach kurzer Zeit dreimal (abwechselnd ein „xxxxoooo oooooxxx“-Muster). Anschließend werden nacheinander angezeigt:

- das XFRESULT (Port A)
- der XFCOUNTER (Port B/A)
- der YFCOUNTER (Port B/A)
- das FLAGS-Ergebnis (Port A) für A = 0, F = FF / (Ergebnis kann \$00, \$08, \$20, \$28 sein).

Die Ausgabe erfolgt ziemlich schnell (abhängig von der CPU-Taktung), daher sollten die Ausgabe mit einem Mobiltelefon ggf. aufgezeichnet werden, um sie später analysieren zu können.

#### XFRESULT encoding

- A[7:6] - YF-Ergebnis von F = 0, A = C | 0x20 & 0xF7 (F.5 gesetzt, F.3 zurückgesetzt)
- A[5:4] - XF-Ergebnis von F = 0, A = C | 0x08 & 0xDF (F.3 gesetzt, F.5 zurückgesetzt)
- A[3:2] - YF-Ergebnis von F = C | 0x20 & 0xF7, A = 0 (F.5 gesetzt, F.3 zurückgesetzt)
- A[1:0] - XF-Ergebnis von F = C | 0x08 & 0xDF, A = 0 (F.3 gesetzt, F.5 zurückgesetzt)

Wobei die Ergebnisbits wie folgt gesetzt sind:

- 00 – Flag immer auf 0 gesetzt
- 11 – Flag immer auf 1 gesetzt
- 01 – Flag meistens auf 0 gesetzt
- 10 – Flag meistens auf 1 gesetzt

Hinweis: YF aka F.5, XF aka F.3

## 1.7 Messung des Stromverbrauchs

Wird der Jumper „Current measurement“ entfernt, kann dort ein Strommessgerät angeschlossen werden. Es kann so der Stromverbrauch der CPU gemessen werden.

## 2 Bill of materials (BOM) Ergänzung

### 2.1 USB-Buchse



z.B. bei AliExpress:

<https://de.aliexpress.com/item/1005003018898705.html>

<https://de.aliexpress.com/item/1005001629294526.html>

<https://de.aliexpress.com/item/1005004836287679.html>

<https://de.aliexpress.com/item/4000838345540.html>



z.B. bei AliExpress:

<https://www.aliexpress.com/item/32834923469.html>