

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Die letzten ihrer Art

5V-CPLDs von Microchip
und wie man sie mit freien Tools
programmiert

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

CPLDs sind die ideale Plattform für programmierbare Logik in Retro-Systemen

- ♦ Ersatz von alten Chips (z.B. C64 PLA)
- ♦ Bau von Erweiterungen (EasyFlash³)
- ♦ Glue-Logik in Selbstbau-Projekten

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Problem mittlerweile: Verfügbarkeit!

- CPLDs werden immer seltener und teurer
- Xilinx hat CPLDs im Jahr 2024 abgekündigt

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Nur noch wenige Hersteller bieten CPLDs an

Einer davon: Microchip, ehem. Atmel

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Die ATF150x-Reihe von Microchip:

ATF1502 – 32 Makrozellen

ATF1504 – 64 Makrozellen

ATF1508 – 128 Makrozellen

ATF150xAS – nativ 5V

ATF150xASV – nativ 3,3V

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

“We do not have any plans to discontinue the ATF15xx CPLDs in the foreseeable future, and it is advisable to use them in new designs.”

- Typ bei Microchip, Juni 2025

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

ProChip Designer

- native IDE von Microchip
- Windows only :-)

ATMISP7 Programmierertool

- Windows only :-)
- Microchip Download Cable für \$90

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Wir wollen:

- Keine 90\$ für ein Download-Kabel ausgeben
- Kein Windows benutzen
- Möglichst Open Source einsetzen

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

ATF150x sind aber kompatibel zur MAX7000-Reihe von Altera!

EPM7xxx	ATF150x
EPM7032	ATF1502
EPM7064	ATF1504
EPM7128	ATF1508

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

.. die ist aber schon lange abgekündigt

Aktuelles Quartus II unterstützt sie nicht mehr

Aber!

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Quartus II Version 13.0sp1 unterstützt MAX7000

Und läuft unter Linux!

Aber!

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Software von 2013 auf aktuellem Linux ist nicht ohne Tücken

- Libpng 1.2 wird benötigt
- libtbb fehlerhaft (Quartus II hängt sich auf)
- Parameter '--64bit'

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Jetzt können wir mit Quartus II VHDL programmieren und synthetisieren!

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Wie kommt das jetzt auf den ATF150x?

2 Wege:

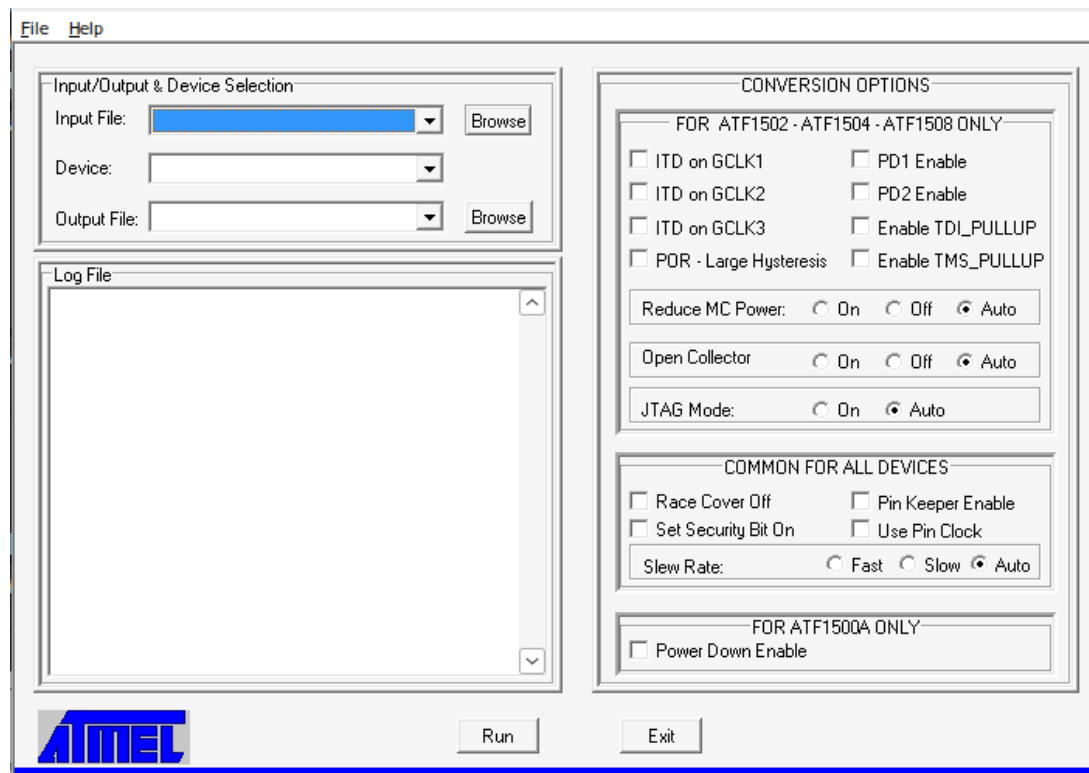
- Offiziell: 90\$ Download Kabel und ATMISP7
- Geizig: Adafruit FT232 Breakout + OpenOCD

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Für beide Wege wird zunächst "POF2JED" von
Microchip benötigt...

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

konvertiert POF-Datei aus Quartus in eine JED-Datei passend für ATF150x



PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

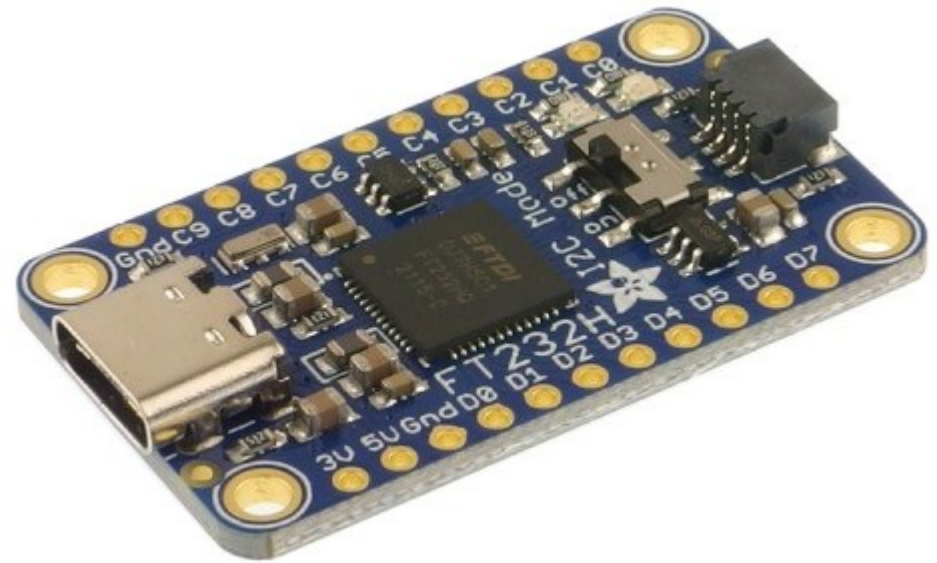
GUI-Only-Tools im Workflow sind lästig!

- Unix-Version wurde eingestellt
- Windows-Tool kommt mit CLI und läuft in WINE

PROGRAMMIERBARE LOGIK FUER GEIZIGE MENSCHEN

Adafruit FT232H Breakout

- Kostet ca. €15
- OpenOCD (alles)
- XC3SPROG (Xilinx)

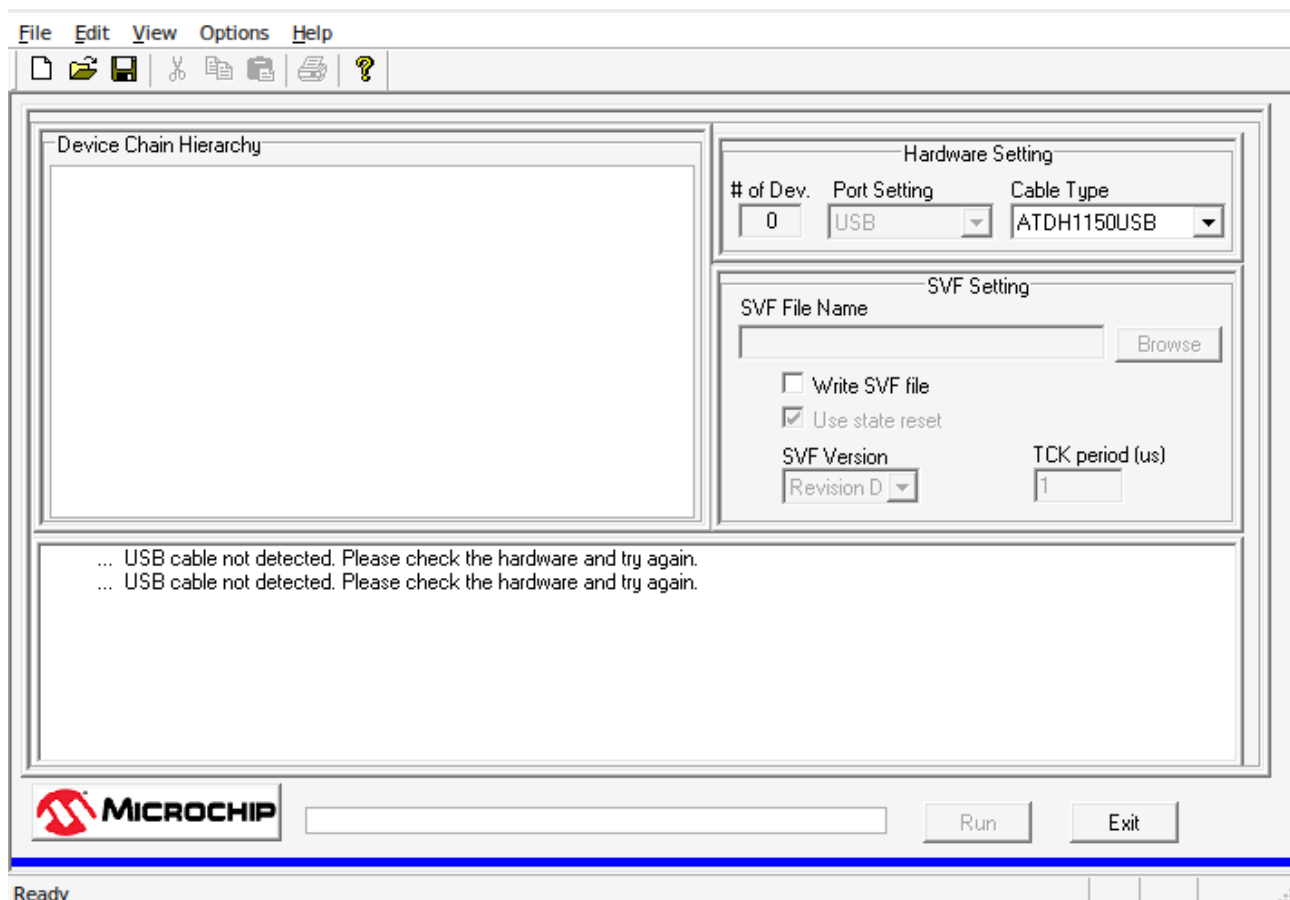


PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

OpenOCD braucht aber eine SVF-Datei statt JED

- ATMISP7 kann SVF erzeugen
 - Windows-Tool :-(
 - GUI only :-(

PROGRAMMIERBARE LOGIK FUER GEIZIGE MENSCHEN



PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

GUI-Only-Tools im Workflow sind lästig!

- ATMISP7 ist nicht parametrisierbar
- Alternative wäre schön

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

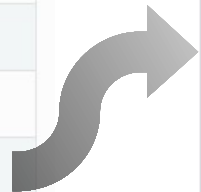
Project Bureau

“Project Bureau enables a fully open-source flow for [Microchip](#) (Atmel) ATF15xx family [CPLDs](#) by documenting their fuse maps and programming algorithms.”

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Project Bureau

Device(s)	Fuse database	Programming
ATF1502AS/ASV/ASL	Complete	Complete
ATF1504AS/ASV/ASL	Near-complete	Untested
ATF1508AS/ASV/ASL	Partial	Untested
ATF1502BE	Near-complete	None
ATF1504BE	Near-complete	None
ATF1508BE	Partial	None



◆◆Catherine◆◆

@whitequark@mastodon.social

@steckschwein_6502 fwiw fuseconv for ATF1508 should work fine, the incomplete part is bitstream functionality extraction

19. Juli 2025, 16:25 · 🌙 · Tusky

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Project Bureau

- Python-basiert
- CLI-Tool “fuseconv”

```
python3 -m util.fuseconv -d ATF1508AS \  
    chuck.jed \  
    chuck.svf
```

PROGRAMMIERBARE LOGIK FUER GEIZIGE MENSCHEN

```
#!/bin/bash
```

```
export PYTHONPATH=/home/dommas/src/prjbureau/
```

```
wine c:\\POF2JED\\bin\\POF2JED.exe \  
    -i output_files/chuck.pof \  
    -o output_files/chuck.jed || exit 1
```

```
python3 -m util.fuseconv -d ATF1508AS \  
    output_files/chuck.jed \  
    output_files/chuck.svf || exit 1
```

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

```
#!/bin/bash
```

```
./make_svf.sh
```

```
openocd -f adafruit-232h-breakout.cfg \  
-c "adapter speed 6400" \  
-c "transport select jtag" \  
-c "jtag newtap ATF1508AS tap -irlen 3 -expected-id 0x0150803f" \  
-c init \  
-c "svf output_files/chuck.svf quiet progress" \  
-c "sleep 200" \  
-c shutdown
```

PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN



PROGRAMMIERBARE LOGIK FÜR GEIZIGE MENSCHEN

Links

<https://github.com/peterzieba/5Vpld/>

<https://www.adafruit.com/product/2264>

<https://www.microchip.com/en-us/development-tool/pof2jed>

<https://whitequark.github.io/prjbureau/intro.html>

<https://jacobncalvert.com/blog-archive/2020/03/05/better-jtag-on-the-cheap-with-the-ft232h/>