

Die Hardware-Toolchain nutzen

Im Folgendem werden alle Schritte aufgeführt, um die Hardware-Toolchain zu benutzen.

Der für das ICEduino Entwicklungsboard angepasste Softcore sowie diverse Beispielpprogramme können hier heruntergeladen werden: <https://github.com/MrElliwood/iceduino-toolchain>

Diese Anleitung geht davon aus dass das Git Repository nach `~/iceduino` geklont wurde. Sollten die Dateipfade abweichen so ist dies zu berücksichtigen. Das Klonen ist mit dem folgenden Befehl möglich:

```
git clone https://github.com/MrElliwood/iceduino-toolchain ~/iceduino
```

Die Beispielpprogramme befinden sich nun in `~/iceduino/programs`

Das Kompilieren der Programme wird in dem Dokument „Nutzung der Software-Toolchain“ erklärt. Diese Anleitung geht davon aus, dass bereits ein Programm kompiliert wurde und in der `top.v` des Softcores als bootrom hinterlegt wurde.

Der Softcore befindet sich in `~/iceduino/SPU32-softcore`

Das überspielen eines Designs auf das ICEduino Entwicklungsboard besteht aus 4 Schritten.

1. Yosys Synthese

Die Synthese wird aus dem Verzeichnis `~/iceduino/SPU32-softcore` heraus gestartet.

Das Verzeichnis kann mit folgendem Befehl betreten werden:

```
cd ~/iceduino/SPU32-softcore
```

Die Synthese wird mit folgendem Befehl gestartet.

```
yosys -p 'synth_ice40 -top top -json ./boards/fpgarduino/top.json'
./boards/fpgarduino/top.v
```

Alternativ kann die Synthese ohne betreten des Verzeichnisses mit folgendem Befehl gestartet werden:

```
yosys -p 'synth_ice40 -top top -json ~/iceduino/SPU32-softcore/boards/fpgarduino/top.json' ~/iceduino/SPU32-softcore/boards/fpgarduino/top.v
```

2. NextPNR Place & Route

Place & Route ist innerhalb des Softcore-Verzeichnisses mit dem folgenden Befehl zu starten:

```
nextpnr-ice40 --hx8k --package tq144:4k --json ./boards/fpgarduino/top.json  
--pcf ./boards/fpgarduino/top.pcf -asc ./boards/fpgarduino/top.asc
```

Place & Route ist außerhalb des Softcore-Verzeichnisses mit dem folgenden Befehl zu starten:

```
nextpnr-ice40 --hx8k --package tq144:4k --json ~/iceduino/SPU32-  
softcore/boards/fpgarduino/top.json --pcf ~/iceduino/SPU32-  
softcore/boards/fpgarduino/top.pcf -asc ~/iceduino/SPU32-  
softcore/boards/fpgarduino/top.asc
```

Die NextPNR GUI kann mit dem Befehl `nextpnr-ice40 --gui` geöffnet werden. Die GUI bietet place & route funktionalität in einer leicht verständlichen grafischen Oberfläche.

3. IcePack flashbare bin-Datei packen

Eine flashbare Binärdatei wird innerhalb des Softcore-Verzeichnisses mit folgendem Befehl erzeugt:

```
icepack ./boards/fpgarduino/top.asc ./boards/fpgarduino/top.bin
```

Eine flashbare Binärdatei wird außerhalb des Softcore-Verzeichnisses mit folgendem Befehl erzeugt:

```
icepack ~/iceduino/SPU32-softcore/boards/fpgarduino/top.asc ~/iceduino/SPU32-  
softcore/boards/fpgarduino/top.bin
```

4. IceProg programmieren

ACHTUNG: Falls die Hardware-Toolchain in einer virtuellen Maschine ausgeführt wird, muss das ICEduino Entwicklungsboard zuerst an die VM angeschlossen werden. Das Vorgehen ist dem Handbuch der Virtualisierungslösung zu entnehmen.

Die Binärdatei kann innerhalb des Softcore-Verzeichnisses mit folgendem Befehl auf das ICEduino Entwicklungsboard geflasht werden:

```
sudo iceprog ./boards/fpgarduino/top.bin
```

Die Binärdatei kann außerhalb des Softcore-Verzeichnisses mit folgendem Befehl auf das ICEduino Entwicklungsboard geflasht werden:

```
sudo iceprog ~/iceduino/SPU32-softcore/boards/fpgarduino/top.bin
```

Weiterführende Informationen

- Yosys Dokumentation: <http://www.clifford.at/yosys/documentation.html>
- NextPNR FAQ: <https://github.com/YosysHQ/nextpnr/blob/master/docs/faq.md>
- Project IceStorm: <http://www.clifford.at/icestorm/>