Statusbericht OpenCL Einrichtung, Board Test

Inhalt

[Plattform 2](#_Toc486444687)

[System 2](#_Toc486444688)

[Quartus Umgebung 2](#_Toc486444689)

[Voraussetzungen 3](#_Toc486444690)

[Kompilierung (Board test) 4](#_Toc486444691)

[Vorbereitung des Boards 5](#_Toc486444692)

[Bereitstellung 6](#_Toc486444693)

[Ausführung 7](#_Toc486444694)

[Fazit 8](#_Toc486444695)

# Plattform

## System

* Core i7 6500u, 8GB RAM
* Windows 10 Pro x64
* Windows Subsystem for Linux
* PuTTY

## Quartus Umgebung

* Quartus 2 14.1.0.186
* AOCL 14.1.0.186
* SoC EDS 14.1.0.186

# Voraussetzungen

1. Altera Quartus Umgebung einrichten
2. BENUTZERSPEZIFISCHE[[1]](#footnote-1) Umgebungsvariablen einrichten
   1. Altera OpenCL License (LM\_LICENSE\_FILE anlegen)
   2. Altera OpenCL Binärdateien (zu PATH hinzufügen)
   3. Altera Board Package Pfad (AOCL\_BOARD\_PACKAGE\_ROOT anlegen)

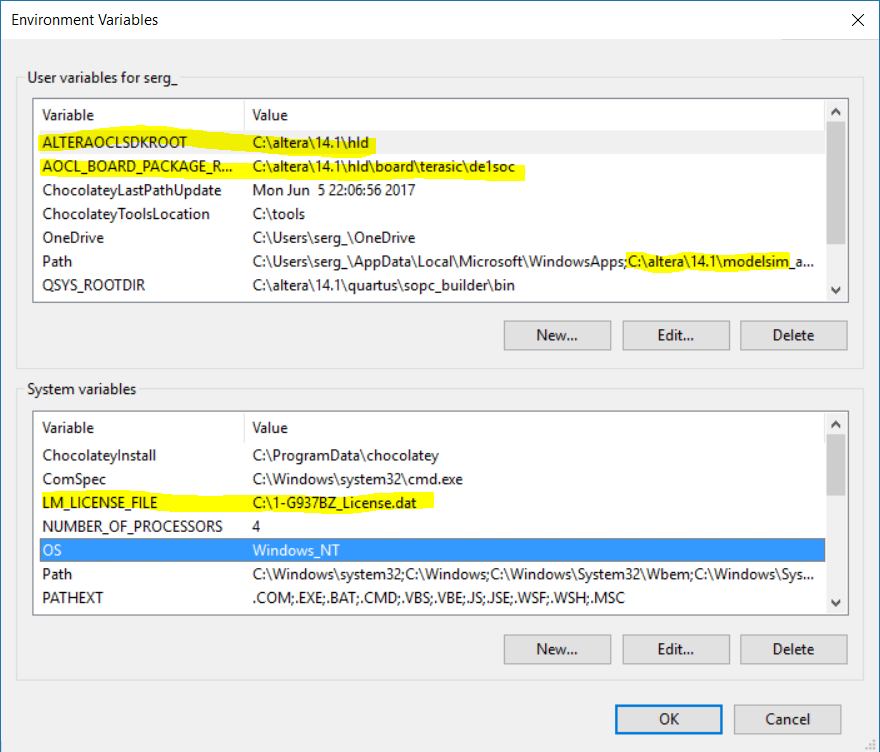


Abbildung 1 Für Altera OpenCL benötigte Umgebungsvariablen

# Kompilierung (Board test)

1. Gehen Sie zu “C:\altera\14.1\hld\board\terasic\de1soc\examples\boardtest”
2. Geben ie im Terminal ein: aoc device/boardtest.cl --sw-dimm-partition -o bin/boardtest.aocx
3. Führen Sie die Embedded Shell unter “C:\altera\14.1\embedded\Embedded\_Command\_Shell.bat” aus
   1. Es öffnet sich eine Cygwin[[2]](#footnote-2) Kommandozeile
   2. Gehen Sie zu /cygdrive/c/altera/14.1/hld/board/terasic/de1soc/examples/boardtest[[3]](#footnote-3)
   3. Erstellen Sie das Testprojekt

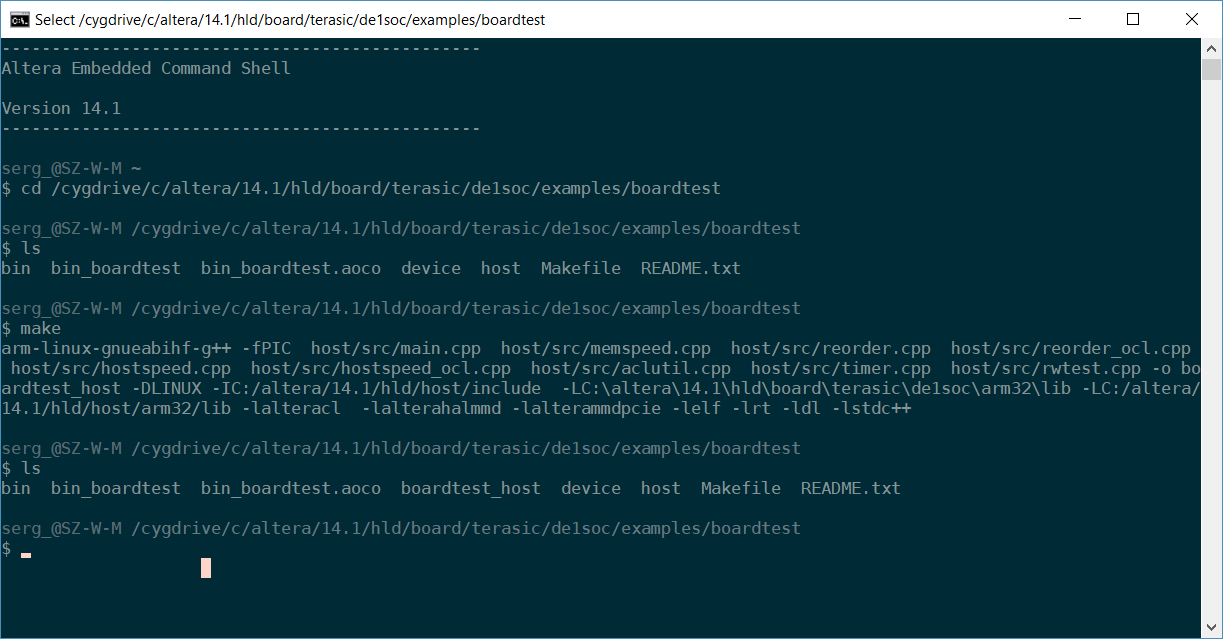


Abbildung 2 Altera Embedded Command Shell

# Vorbereitung des Boards

Vor der Software-Bereitstellung muss das DE1-SoC konfiguriert und lauffähig sein.

Ein vorbereiteter Flashkartenspeicher mit einem Linux-Basissystem ist erforderlich.

1. Schließen Sie die Strom-, USB/UART- und Netzwerkkabel an
2. Starten Sie PuTTY, wählen Sie den passenden COM-Port aus und stellen Sie die Baud-Rate auf *115200 bps*
3. Loggen Sie sich als “root”-Benutzer ein. Ein Passwort wird nicht benötigt.
4. Konfigurieren Sie eine (statische) IP-Adresse, wie z.B *192.168.0.123*

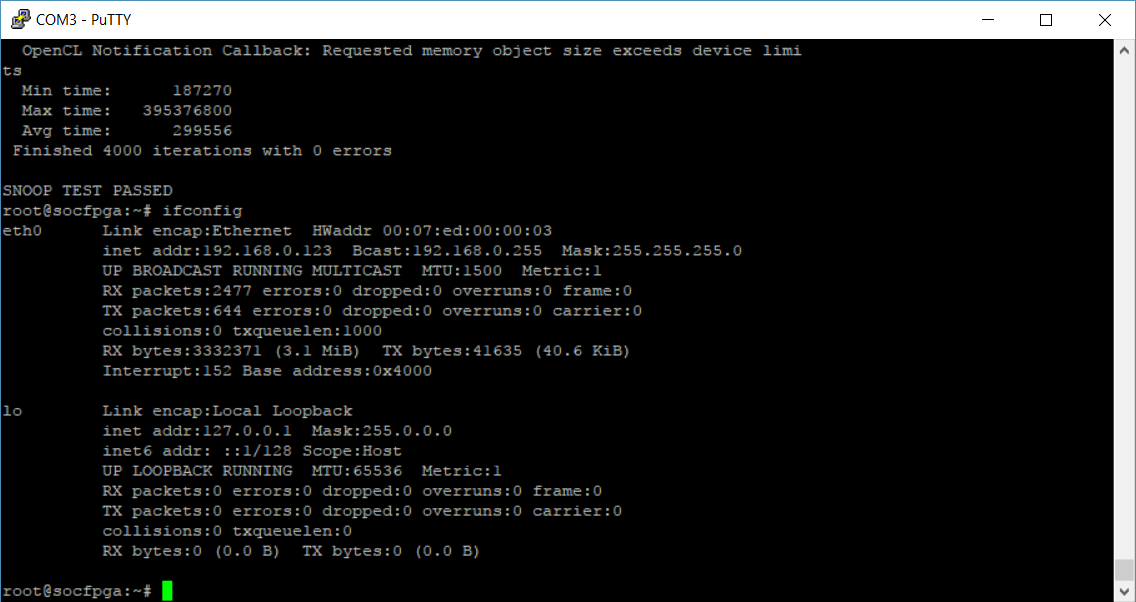


Abbildung 3 IP Konfiguration

# Bereitstellung

Im Terminal:

* Kopieren Sie die ACL Konfiguration: *scp bin/boardtest.aocx* [*root@192.168.0.123:/home/root*](mailto:root@192.168.0.123:/home/root)
* Kopieren Sie die OpenCL Hostanwendung: *scp boardtest\_host* [*root@192.168.0.123:/home/root*](mailto:root@192.168.0.123:/home/root)

Diese Befehle kopieren die Dateien auf das Board über SSH[[4]](#footnote-4).

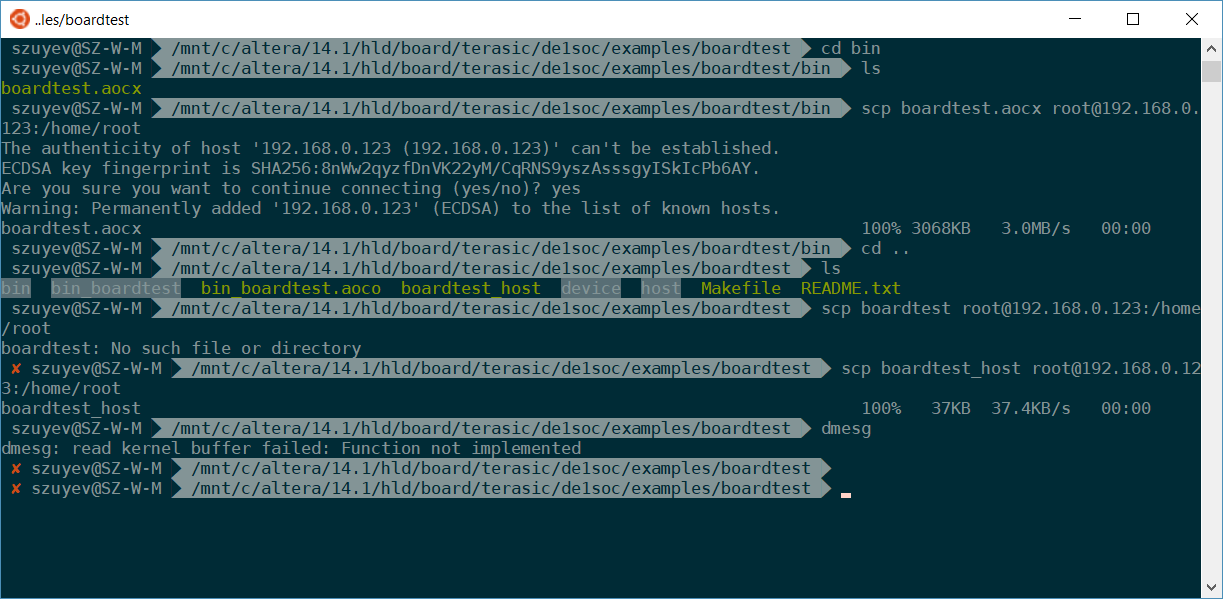


Abbildung 4 Kopieren der relevanten Dateien auf das Board

# Ausführung

* Machen Sie boardtest\_host ausführbar: *chmod +x ./boardtest\_host*
* Bereiten Sie die OpenCL Umgebung vor*: source ./init\_opencl.sh*
* Programmieren Sie das Gerät: *aocl program /dev/acl0 boardtest.aocx*
* Führen Sie die OpenCL Hostanwendung aus: *./boardtest\_host*

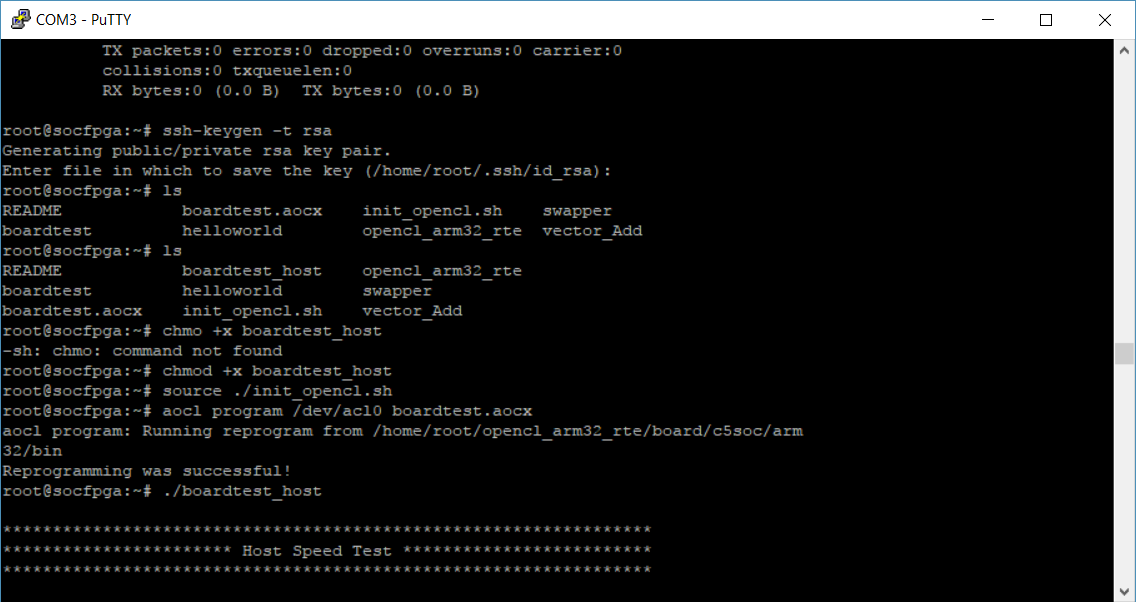


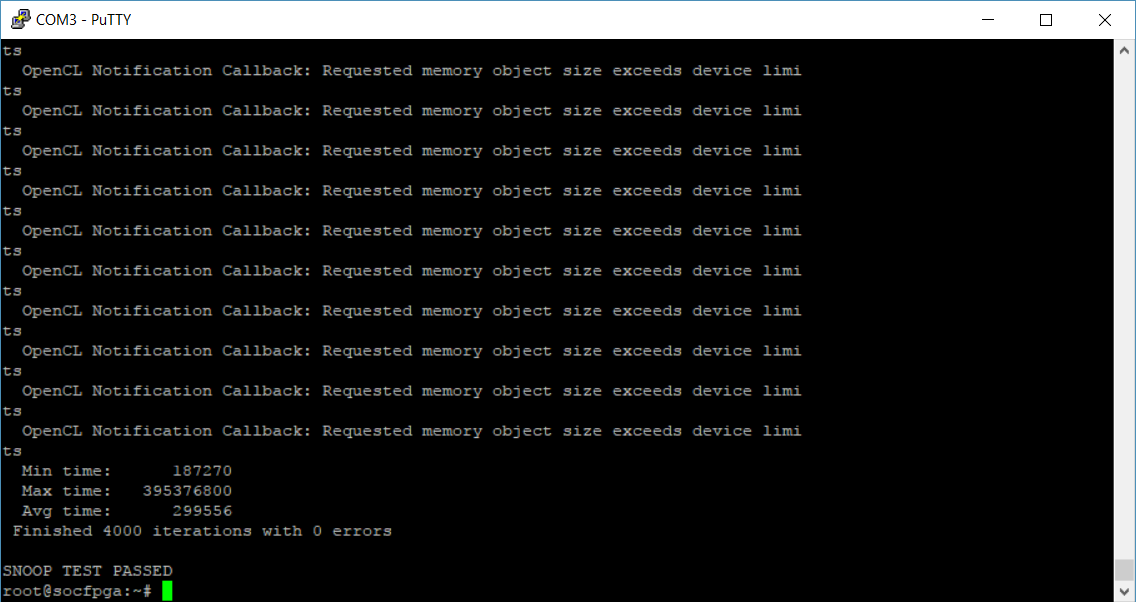
Abbildung 5 Schritte vor der Ausführung des OpenCL Programms

Abbildung 6 OpenCL Board-Tests erfolgreich

# Fazit

* Mindestens ein System (von Sergej Zuyev) ist richtig konfiguriert.
* Wir können OpenCL Programme auf dem zur Verfügung gestelltem DE1-SoC bereitstellen und ausführen
* Wir können dieses Wissen für unsere eigene LBP-Operator Umsetzung nutzen.

1. Systemweite Umgebungsvariablen funktionieren nicht ohne Altera Quartus’ eigene Variablen anpassen zu müssen [↑](#footnote-ref-1)
2. POSIX Emulationsschicht + Werkzeuge für Windows, https://www.cygwin.com/ [↑](#footnote-ref-2)
3. Um auf einen Host-Pfad über die Cygwin-Shell zuzugreifen, wird das /cygdrive “prefix” benötigt (Einhängepunkt) [↑](#footnote-ref-3)
4. Secure Shell [↑](#footnote-ref-4)