

Diese Beschreibung dokumentiert meine (bisherigen) Erfahrungen bei der Benutzung der neuen CADENCE Version 97A zusammen mit dem ES 2 DESIGNKIT.

1 Änderungen in CADENCE

Mit der neuen Version haben sich folgende Änderungen ergeben:

- Das interne Datenformat der Design-Bibliotheken wurde geändert. Alte Bibliotheken müssen in das neue Format konvertiert werden, wofür eine automatische Konvertierung zur Verfügung steht.

Der Vorteil der neuen Struktur ist eine einheitliche Baumstruktur der Bibliotheken, die jetzt von allen Tools (DF II, Synergy, Leapfrog...) genutzt wird. Außerdem ist die Festlegung der Suchpfade über eine Textdatei `cds.lib` wesentlich leichter zu handhaben.

- In den Bibliotheken wurde ein neues Konzept von *Technologiedateien* implementiert, so daß nicht mehr für jede Design-Bibliothek eine Technologiedatei eingelesen werden muß.
- Die Festlegung von Layern in den Design-Bibliotheken wurde geändert.
- Aufgrund der neuen internen Datenstrukturen, hat sich der SKILL-Programmierschnittstelle geändert.

2 Auswirkungen auf den ES 2 DESIGNKIT

Der aktuelle DesignKit v4.1E2 funktioniert mit der aktuellen CADENCE Version nicht mehr! Die Bibliotheken sind hier nicht das Problem, sondern der SKILL-Code, der die DF II Oberfläche an den ES 2 DESIGNKIT anpaßt.

Ich habe versucht die Bibliotheken zu konvertieren und einige „Workarounds“ zu erstellen, so daß man mit alten Designs weiterarbeiten kann, bis eine aktuelle DesignKit Version vorhanden ist.

In den folgenden Abschnitten werden in gewohnter Form (Kochrezept) die Schritte zur Konvertierung und Benutzung erläutert. Diese Beschreibung ist für „erfahrenere“ CADENCE-Anwender, Fehler sind nicht ausgeschlossen...

3 CADENCE allgemein

- Vorarbeiten

Achtung: Alle Initialisierungsdateien für den „alten“ DesignKit müssen aus dem \$HOME-Verzeichnis des Benutzers, sowie aus dem Arbeitsverzeichnis gelöscht werden.

Dies sind alle Dateien: `.cds*`

```
.sim*
.ES2*
.currentDesignKitProcess
.Libversion
```

Gerade wegen dieser Abhängigkeiten empfiehlt es sich, neue Designs in einem noch „jungfräulichen“ Verzeichnis anzufangen.

1. Setup

```
> source ~maeder/designSetup [xterm]
```

Anschließend ist **nur** `cds` einzutragen – das „nackte“ CADENCE.

```
> cd <working-dir> [xterm]
```

```
> cp /local/tech1.3/es2/97A/es2user/.cdsinit . [xterm]
```

```
> cp /local/tech1.3/es2/97A/es2user/.simrc . [xterm]
```

```
> cp /local/tech1.3/es2/97A/es2user/display.drf . [xterm]
```

```
> cp /local/tech1.3/es2/97A/es2user/cds.lib . [xterm]
```

```
> vi cds.lib [xterm]
```

Diese Datei ist, entsprechend den Bibliotheken, anzupassen. Für ein neues Design können alle Zeilen `DEFINE ...` gelöscht werden, eigene Bibliotheken, die man mit CADENCE erstellt, trägt das System selber ein.

Diese Schritte kopieren nach der allgemeinen Initialisierung passende CADENCE-Dateien in das aktuelle Arbeitsverzeichnis.

2. Start des Systems

```
> icfb & [xterm]
```

Das anschließend erscheinene Fenster (What's New in 4.4) sollte man durchlesen, es kann mit `□ File - Close [What's New...]` geschlossen werden.

3. Konvertierung bestehender Bibliotheken

```
□ Tools - Conversion Tool Box ... [icfb]
```

```
↑ Convert DFII-DM Libraries [Conversion Tool Box]
```

```
≡ - bestätigen [libcv: Information]
```

```
≡ Search Path = <old lib dir> [DFII-DM Library Conversion]
```

```
↑ Update [DFII-DM Library Conversion]
```

```
↑ <lib1> <lib2> ... [DFII-DM Library Conversion]
```

Nach der Auswahl der Bibliotheken kann die Konvertierung gestartet werden.

```
□ File - Close [Conversion Tool Box]
```

Anmerkung: Im Sinne des neuen Konzeptes von Technologiebibliotheken sind alle „alten“ 4.3.x-Bibliotheken mit einer eigenen Technologiedatei `techfile.cds` versehen. Bei der Konvertierung wurden diese Dateien automatisch angelegt. Aus Konsistenzgründen ist es jedoch besser eine „zentrale“ Technologiedatei zu haben – für den ES2-Prozess `ecpd07` ist dies die Bibliothek `techLib`!

Dementsprechend müssen die neu konvertierten Bibliotheken noch mit dieser Bibliothek verknüpft werden:

```

□ Technology File - Attach To ... [icfb]
≡ Design Library                = <lib> [Attach Design Library...]
  Attach To Technology          = techLib
  Library

```

Hier noch einige allgemeine Schritte die sich durch neue Konzepte oder durch die geänderte Bibliotheksstruktur ergeben haben. Erfahrung im Umgang mit der bisherigen Version wird als bekannt vorausgesetzt:

- Neue Bibliotheken erzeugen

```

□ File - New - Library... [icfb]
≡ Name                    = <lib name> [New Library]
  Technology File         = Attach to an existing techfile
≡ Attach To Technology    = techLib [Attach Design Library...]
  Library
  Verbindung zur Technologiedatei s.o.

```

- Neue Designs erzeugen

```

□ File - New - Cellview... [icfb]
≡ Library Name            = <lib name> [Create New File]
  Cell Name               = <cell name>
  Tool                    = Composer-Schematic / Virtuoso
  Damit können schematic / layout Cellviews erzeugt werden...

```

- Library Browser

```

□ Tools - Library Manager... [icfb]
  Der „neue“ Library-Manager arbeitet ähnlich dem vorher vorhandenen Browser.

```

4 Arbeit mit vorhandenen Entwürfen

Da ich auch erst vier Tage mit den Tools gearbeitet habe, beschränkt sich die jetzt folgende Aufzählung auf „erste Gehversuche“ – Erweiterungen folgen.

Schematic-Editor: läßt sich wie gewohnt bedienen.

Layout-Editor: läßt sich wie gewohnt bedienen. Wurde die Bibliothek „richtig“ (s.o.) mit der Technologiedatei verknüpft, so sind die passenden Regelsätze für den DRC und die Schaltungsextraktion vorhanden, z.B.:

≡ Switch Names	= slow	[DRC]
Rules File	= divaDRC.rul	
Rules Library	= techLib	

Layout Synthese: neues Layout der Fill-Form mit zusätzlichen Optionen (Einstellung:???), läuft aber.

elektrische Simulation: nach einigen Änderungen und zusätzlichen SKILL-Patches, verhält sich die HSpice Simulation wie gewohnt (Auswahl des Transistormodells u.s.w.), manuelle Eingriffe sind nicht mehr notwendig!

Verilog Simulation: ???

Leapfrog Simulation: ???

Synergy: ???

Placement & Routing: ???