

Altera の QuartusII Web Edition と NiosII EDS を使って NiosII で EPCSxx に NiosII のブートコードを格納するようなシステムを開発するときのデバッグサイクルメモです。

旧 SoPC Builder 時代のプロジェクトを引っ張り出してデバッグしたときのメモなので、現時点ではもっとスマートな方法が主流かもしれません。

デバッグサイクルを始めるための準備

Quartus , SoPC Builder の 設定注意

- ・ NiosII CPU のリセットベクタを epcs コンポーネントにすること
- ・ EPCS への接続線をトップレベルに持ってきて、自分でピンアサインすること
- ・ デバイスの専用ピン (EPCS の接続されているピン) を User I/O にすること。これをやらないと内部シンボルと衝突が起きたり、nios2-flash-programmer で正しいアドレスを指定しているのに失敗する。

NiosII ファームウェアデバッグ作業の準備

SoPC Builder で SoC 構成を変えたり、ロジックを変更したらここから

- ・ 書き込みの前に .sof でコンフィグして NiosII を動かしておくこと。(Quartus Programmer から書いておいてもよい)

```
bash-3.1$ nios2-configure-sof niostiny.sof
```

- ・ .sof から nios2-flash-programmer で書けるファイルの生成

```
bash-3.1$ sof2flash --epcs --input=niostiny.sof --output=niostiny.flash
```

- ・ .sof (ロジックのみ) の書き込み

```
bash-3.1$ nios2-flash-programmer --epcs --base=0x4001800 niostiny.flash
```

0x4001800 は EPCS のベースアドレス

これでハードウェアが EPCS からロードされた状態で NiosII EDS (or 新ツール) にてデバッグ開始できる

ファームウェア (とロジック) の書きこみ

ファームウェアを変更した場合はここから

- ・ ファームウェア (niostiny.elf) をロジック (niostiny.flash、前項で生成) の後ろに結合した書き込みファイル (niostiny.firmware) を作成

(以下を一行で入力)
bash-3.1\$ elf2flash

```
--epcs
--after=niostiny.flash
--input=software/niostiny/niostiny.elf
--output=niostiny.firmware
--boot=/cygdrive/c/altera/10.1 sp1 /nios2 eds/components/altera_nios2
/boot_loader_epcs_sii_siii_ciii.srec
```

--boot= 以降は NiosIIEDS(もしくはその後継ソフト) に添付されているブートローダファイルを指定してやる。パス名やファイル名は FPGA の種類や QuartusII のバージョンに依存するので注意。

- ・ 結合されたファームウェアを改めて EPCS に書き込む

```
bash-3.1$ nios2-flash-programmer --epcs --base=0x4001800 niostiny.firmware
```

書き込みファイルが異なる以外、前項の最後と同じコマンド

0x4001800 は EPCS のベースアドレス

これでリセットで NiosII が EPCS からコードをロードしてシステム全体が走る