

## DE0-Nano-SoC+DE0/DE0-nano 用拡張ボードのフルカラーDVI 出力アダプタのデモ

2016/04/05 Kenji Ishimaru

### ○デモについて

このデモは、デモ用のデータを格納した SD カードを DE0-Nano-SoC に挿入して電源を ON にするだけで起動します。FPGA のコンフィギュレーションも電源 ON で自動的行われます。

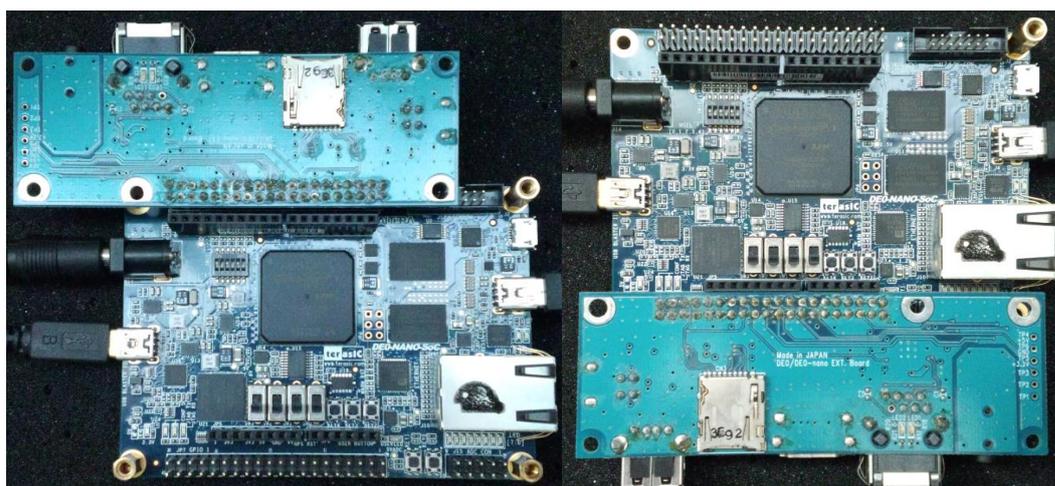
### ○ファイル構成

このデモ・データには、次のファイルが格納されています。

ファイル名	説明
preloader-mkpmimage.bin	プリローダーイメージ
bears_gpio0_fpga.bin	デモアプリ 1GPIO 拡張ヘッダ 0 用
cubes_gpio0_fpga.bin	デモアプリ 2GPIO 拡張ヘッダ 0 用
bears_gpio1_dsub.bin	デモアプリ 1GPIO 拡張ヘッダ 1 用
cubes_gpio1_fpga.bin	デモアプリ 2GPIO 拡張ヘッダ 1 用
readme.pdf	このファイル

### ○ボードの準備

DE0-Nano-SoC に DE0-nano 用拡張ボードを接続します。接続するとき、DE0-Nano-SoC の上部アクリル板とその支持用スペーサは取り外す必要があります。拡張ボードは、DE0-Nano-SoC の GPIO 拡張ヘッダ 0 または GPIO 拡張ヘッダ 1 に接続します。



(GPIO 拡張ヘッダ 0 に接続)

(GPIO 拡張ヘッダ 1 に接続)

### ○MSEL の設定

DE0-Nano-SoC の MSEL[4:0]を ON/OFF/ON/OFF/ON に設定します。(デフォルト状態

です。)

## OSD カードの準備

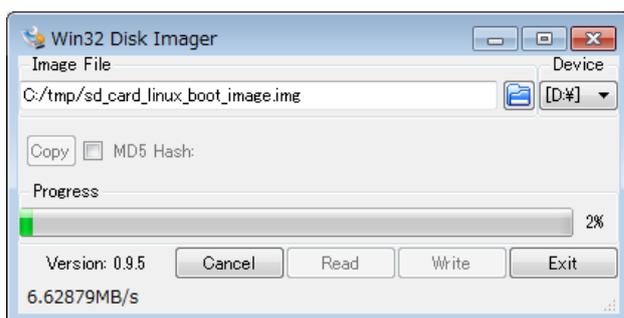
DE0-Nano-SoC に付属の SD カードと Quartus II 15.0 の組み合わせで動作確認を行います。SD カードの準備には、次の 1.2.3.のステップが必要です。

### 1. sd\_card\_linux\_boot\_image.img の書き込み

Altera 社の SoC EDS のインストールディレクトリに格納されている sd\_card\_linux\_boot\_image.img を SD カードに書き込みます。まず、インストールディレクトリから、sd\_card\_linux\_boot\_image.tar.gz を探します。SoC EDS のインストールディレクトリが C:/Altra の時、次のディレクトリに格納されています。

C:/Altera/15.0/embedded/embeddedsw/socfpga/prebuilt\_images

次に、sd\_card\_linux\_boot\_image.tar.gz を解凍します。解凍すると、sd\_card\_linux\_boot\_image.img が生成されます。このファイルを RAW データとして SD カードに書き込みます。書き込みには、例えばフリーツールの Win32DiskImager 等が利用できます。



書き込みが成功すると、その SD カードに FAT パーティションが作られて、PC などで認識できるようになります。

### 2. Preloader の書き込み

preloader-mkimage.bin を SD カードに書き込みます。書き込みには、専用のツール alt-boot-disk-util を使用します。

まず、Quartus II/SoC EDS の Embedded\_Command\_Shell を起動します。Windows の場合、/Altera/15.0/embedded/Embedded\_Command\_Shell.bat で起動します。

(注意: Embedded\_Command\_Shell.bat は管理者として実行してください)

次に、Embedded\_Command\_Shell で alt-boot-disk-util を実行します。

コマンドは、

```
alt-boot-disk-util -a write -p preloader-mkimage.bin -d <SD カードドライブ>
```

です。SD カードが D ドライブの場合、

```
alt-boot-disk-util -a write -p preloader-mkpimage.bin -d d
```

になります(Windows の場合)。Embedded\_Command\_Shell を管理者として実行していないと、書き込みでエラーが発生する場合があります。

```
$ alt-boot-disk-util -a write -p preloader-mkpimage.bin -d d
Altera Boot Disk Utility
Copyright (C) 1991-2014 Altera Corporation
Altera Boot Disk Utility was successful.
```

### 3. アプリケーションのコピー

アプリケーションのバイナリは、DE0/DE0-nano 用拡張ボードを接続した GPIO の番号で異なります。

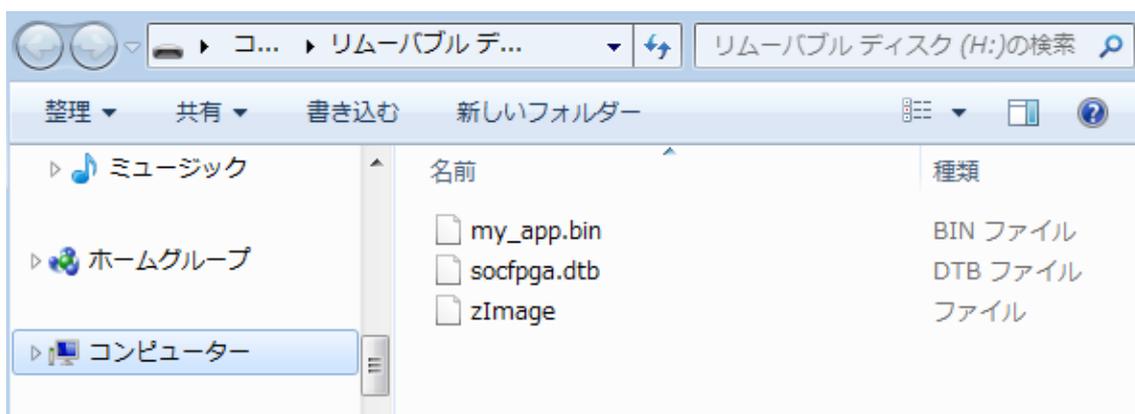
ファイル名	説明
bears_gpio0_fpga.bin	デモアプリ 1GPIO 拡張ヘッダ 0 用
cubes_gpio0_fpga.bin	デモアプリ 2GPIO 拡張ヘッダ 0 用

GPIO 拡張ヘッダ 0 用

ファイル名	説明
bears_gpio1_dsub.bin	デモアプリ 1GPIO 拡張ヘッダ 1 用
cubes_gpio1_fpga.bin	デモアプリ 2GPIO 拡張ヘッダ 1 用

GPIO 拡張ヘッダ 1 用

DE0/DE0-nano 用拡張ボードを接続した GPIO の番号を確認し、それに対応した.bin の 1 つを my\_app.bin にリネームして SD カードの FAT パーティションにコピーします。最終的に、SD カードには次の 3 ファイルが格納されることになります。



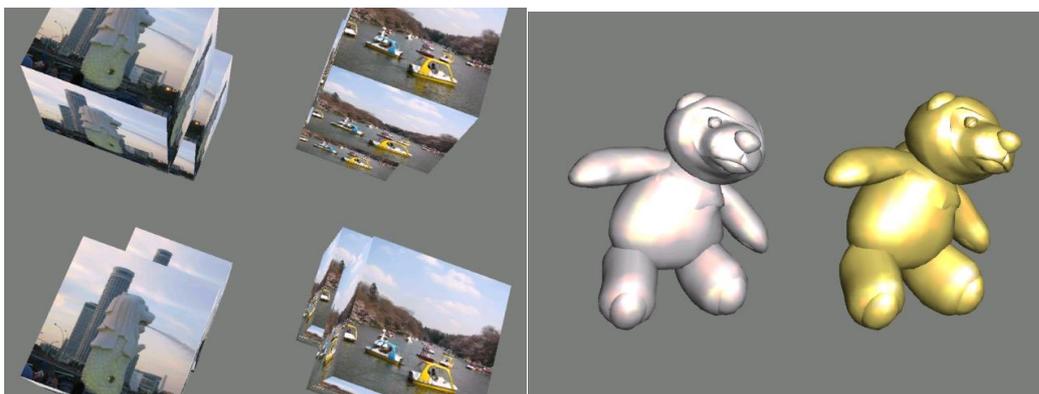
#### ○ディスプレイの接続

ZedBoard 用フルカラーDVI 出力アダプタの HDMI コネクタに、LCD などの表示装置を接

続します。

#### ○デモの実行

SD カードを DE0-Nano-SoC に挿入して電源を ON にします。数秒でコンフィギュレーションが終了します。ディスプレイには、次のような画面が表示されます。



(デモアプリ 1cubes\*.bin)

(デモアプリ 2bears\*.bin)

#### ○ターミナルの接続

DE0-Nano-SoC の UART を TeraTerm などのターミナルに接続すると、ブート時のログが確認できます。ボーレートは 115200 です。

(参考:デモ実行が成功した場合のログ )

```
U-Boot SPL 2013.01.01 (Jan 20 2016 - 07:19:29)
BOARD : Altera SOCFPGA Cyclone V Board
CLOCK: EOSC1 clock 25000 KHz
CLOCK: EOSC2 clock 25000 KHz
CLOCK: F2S_SDR_REF clock 0 KHz
CLOCK: F2S_PER_REF clock 0 KHz
CLOCK: MPU clock 925 MHz
CLOCK: DDR clock 400 MHz
CLOCK: UART clock 100000 KHz
CLOCK: MMC clock 50000 KHz
CLOCK: QSPI clock 3613 KHz
RESET: COLD
SDRAM: Initializing MMR registers
SDRAM: Calibrating PHY
SEQ.C: Preparing to start memory calibration
SEQ.C: CALIBRATION PASSED
SDRAM: 1024 MiB
ALTERA DWMMC: 0
reading hello-mkimage.bin
reading hello-mkimage.bin
INFO: Setup DMA System ...
INFO: Channel 0 allocated.
INFO: Setup of DMA successful.
```

INFO: FPGA Image binary at 0x1a62b8.  
INFO: FPGA Image size is 1960824 bytes.  
INFO: Setup FPGA System ...  
INFO: MSEL [10] configured correctly for compressed FPGA image.  
INFO: Setup of FPGA successful.

INFO: Setup Bridge [2] ...  
INFO: Setup of Bridge [2] successful.

INFO: Setup Bridge [1] ...  
INFO: Setup of Bridge [1] successful.

INFO: Setup Bridge [0] ...  
INFO: Setup of Bridge [0] successful.

RESULT: FPGA configuration completed successfully.