

# Nios II メモリ構成

ver.14

## Nios II メモリ構成

### 目次

|  |    |
|--|----|
| 1. はじめに .....                                      | 3  |
| 2. Nios II のブート方法 .....                            | 3  |
| 2-1. ブート・メモリの選択 .....                              | 3  |
| 2-2. 実行メモリの選択 .....                                | 4  |
| 2-3. Nios II ブート(ブートコピアを使用する場合) .....              | 5  |
| 2-3-1. デフォルト・ブートコピア .....                          | 5  |
| 2-3-2. デフォルト・ブートコピアの使用 .....                       | 5  |
| 2-4. Nios II ブート(ブートコピアを使用しない場合) .....             | 7  |
| 2-4-1. オンチップ RAM の追加 .....                         | 7  |
| 2-4-2. 初期値ファイルの生成 .....                            | 8  |
| 3. メモリの構成例 .....                                   | 9  |
| 3-1. ブート・メモリと実行メモリが共にオンチップ RAM .....               | 9  |
| 3-2. ブート・メモリが CFI フラッシュ、実行メモリがオンチップ RAM .....      | 10 |
| 3-3. ブート・メモリが CFI フラッシュ、実行メモリが外部 RAM .....         | 11 |
| 3-4. ブート・メモリが EPCS/EPCQ デバイス、実行メモリがオンチップ RAM ..... | 12 |
| 3-5. ブート・メモリが EPCS/EPCQ デバイス、実行メモリが外部 RAM .....    | 13 |
| 4. 参考資料 .....                                      | 14 |
| 改版履歴 .....   | 15 |

## 1. はじめに

この資料は、Nios® II システムのメモリ構成のパターンを紹介しています。Nios II システムを組み込む際のブート・メモリや実行メモリの構成例を紹介しています。

### ■ 対象ツールおよびバージョン

- Quartus® II v14.1

## 2. Nios II のブート方法

### 2-1. ブート・メモリの選択

Nios II のブート・メモリはユーザが任意で選択することができます。FPGA のオンチップ RAM や汎用フラッシュ、コンフィギュレーション・デバイス (EPCS/EPCQ) からブートが可能です。

ブート・メモリは、Nios II 構成時に Reset Vector にて設定されたメモリ領域に割り当てられます。

### ブート・メモリの設定方法

Nios II のブート・メモリは、Qsys (図 2-1-1) 内の Nios II 設定ウィザードにて設定します。Nios II 設定用ウィザード内、[Vectors] タブの “Reset Vector” にて設定したメモリ・インタフェースが Nios II のブート・メモリになります。

### ■ Qsys におけるブート・メモリの設定 ■

The screenshot shows the Qsys configuration interface for the Nios II Gen2 Processor. The 'Vectors' tab is selected, and the 'Reset Vector' section is highlighted with a red box. The settings are as follows:

| Field                | Value        |
|----------------------|--------------|
| Reset vector memory: | ext_flashuas |
| Reset vector offset: | 0x00400000   |
| Reset vector:        | 0x00400000   |

Below the Reset Vector section, the Exception Vector and Fast TLB Miss Exception Vector sections are also visible.

**Reset vector offset:** Reset vector memory のベース・アドレスからのオフセットを設定

**Reset vector:** 物理アドレス表示

図 2-1-1 Qsys 内 Nios II 設定ウィザード

## 2-2. 実行メモリの選択

Nios II の実行メモリは、ユーザが任意に指定することができます。オフチップ・メモリやオンチップ RAM、CFI フラッシュ等を使用することができます。

※EPCS/EPCQ デバイスは実行メモリとしては使用できません。

### 実行メモリの設定方法

Nios II Software Build Tools for Eclipse (以下、Nios II SBT) の BSP Editor にて実行メモリの設定を行います。

なお、Nios II SBT における下記設定は、Qsys で論理の Generate 後に行います。

Generate を行うことによって、生成したシステムの各コンポーネントのパラメータや接続情報が格納されたファイル(例 qsys\_system.sopcinfo)が生成されます。このファイルをソフトウェア開発環境である Nios II SBT で読み込みソフトウェアの開発を進めます。

#### 【 実行メモリの設定手順 】

- (手順1) Qsys で Generate を実行
- (手順2) Nios II SBT でソフトウェア・プロジェクトを作成
- (手順3) BSP プロジェクトを選択し、右クリック ⇒ Nios II ⇒ BSP Editor を選択
- (手順4) [Linker Script] タブにて各セクションのメモリ配置を選択(図 2-2-1)

- ・.text : プログラム領域
- ・.rodata : データ領域 (read only)
- ・.rwdata : データ領域 (read write)
- ・.heap : ヒープ領域
- ・.stack : スタック領域
- ・.bss : 初期化していないデータ領域

※.text、.rodata 領域は、読み出しのみのため、ROM への配置が可能です。その他の領域は書き込みが発生するため、RAM のみに配置が可能です。

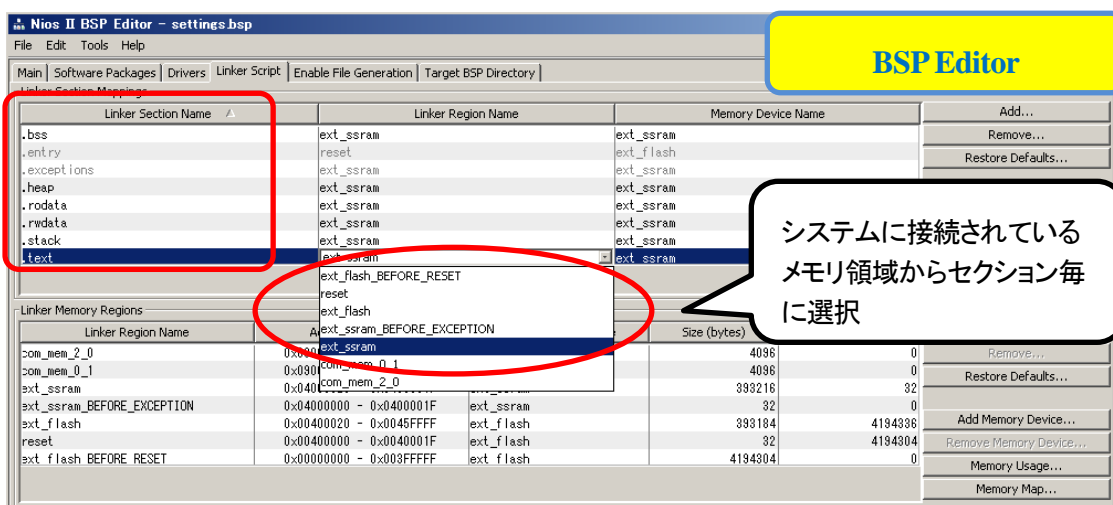


図 2-2-1 Linker Script

### 2-3. Nios II ブート(ブートコピアを使用する場合)

Nios II のプログラム格納に汎用フラッシュや EPCS/EPCQ デバイスを使用します。Nios II は、起動時に Reset Vector に配置されたブートコピアのプログラムによって、プログラム・データを ROM(フラッシュや EPCS/EPCQ デバイス)から RAM へコピーして実行します。

#### 2-3-1. デフォルト・ブートコピア

ブートコピアは、あらかじめアルテラ社によって提供されているものを使用することができます。アルテラ社によって提供されているものはデフォルト・ブートコピアと呼ばれ、下記の機能を持っています。

- ① CFI 準拠のフラッシュ・メモリや EPCS/EPCQ デバイスからのブート
- ② ブートイメージを RAM へコピー
- ③ 展開した RAM 内のアプリケーション・コードへジャンプ

#### 2-3-2. デフォルト・ブートコピアの使用

デフォルト・ブートコピアをプログラムに追加するには、Nios II SBT 内に搭載されている Flash Programmer を使用します。

Flash Programmer を使用することにより、.elf(実行ファイル)を .flash(S-record 形式のフラッシュ書き込み用ファイル)に変換します。ファイルの変換に使用される elf2flash ユーティリティがリセット・アドレスとアプリケーションの実行領域よりブートコピアの必要性を判断します。ブートコピアが必要と判断した場合は、実行コードにブートコピアを追加してフラッシュ書き込み用のファイル(.flash ファイル)を生成します。(図 2-3-2-1)

#### ■ elf2flash におけるファイル変換 ■

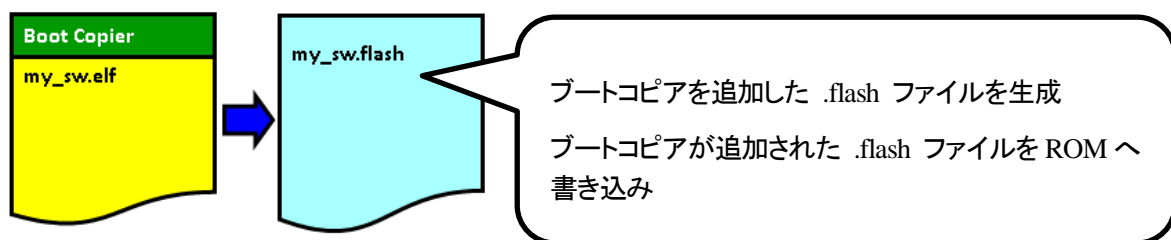


図 2-3-2-1 elf2flash におけるファイル変換

生成された .flash ファイルを CFI フラッシュ、もしくは EPCS/EPCQ デバイスへ書き込むことにより、Nios II 起動時にブートコピーが実行され、実行コードを ROM から RAM へ展開します。(図 2-3-2-2)

■ Nios II 起動時 ■

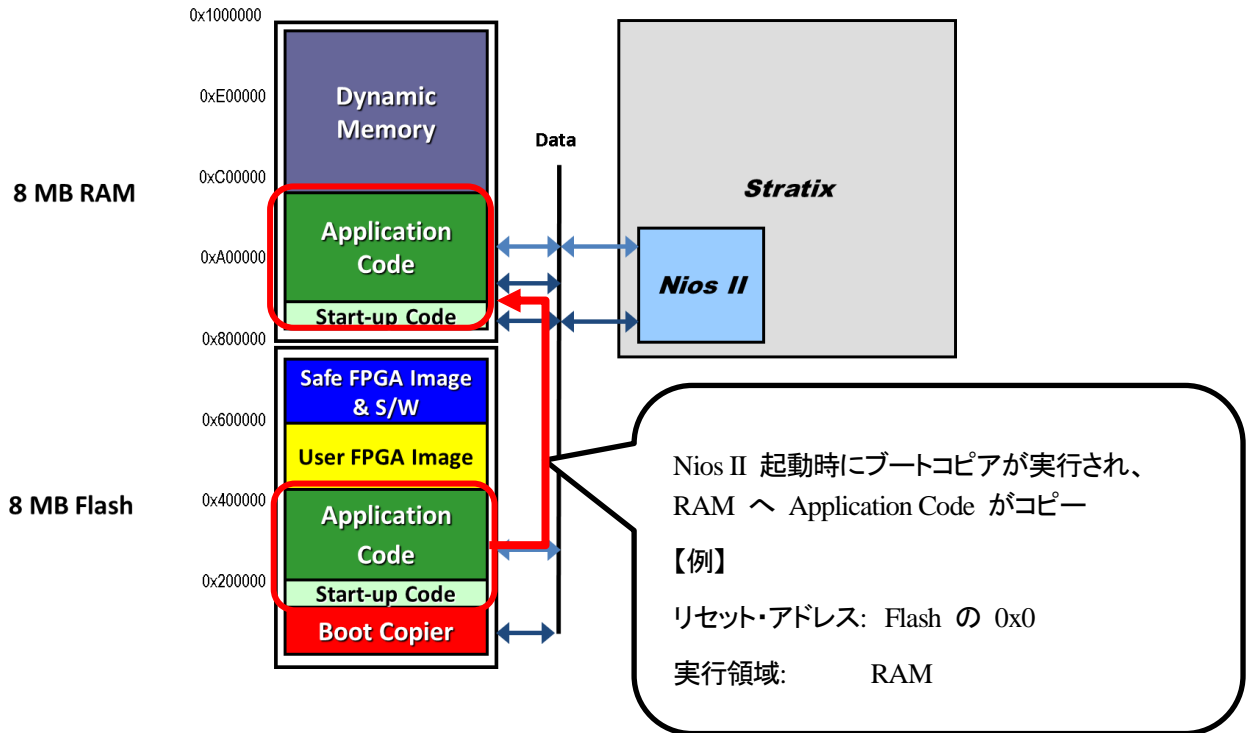


図 2-3-2-2 Nios II 起動イメージ

## 2-4. Nios II ブート(ブートコピアを使用しない場合)

ブート・メモリと実行メモリが同じ場合、Reset Vector が実行メモリに設定されている為、ブートコピアは必要ありません。従ってブート・メモリ内にブートコピアが存在しません。

ブートコピアを必要としない例として、オンチップ RAM をリセット・メモリと実行メモリに設定する場合を紹介します。

### 2-4-1. オンチップ RAM の追加

Nios II の実行メモリとしてオンチップ RAM をシステムに追加します。

Qsys 内 Basic Functions ⇒ On Chip Memory ⇒ On-Chip Memory (RAM or ROM) をダブルクリックし、パラメータ設定ウィザードを起動します。パラメータ設定ウィザード内 Memory initialization 欄で初期データの設定を行います。

図 2-4-1-1 Memory initialization

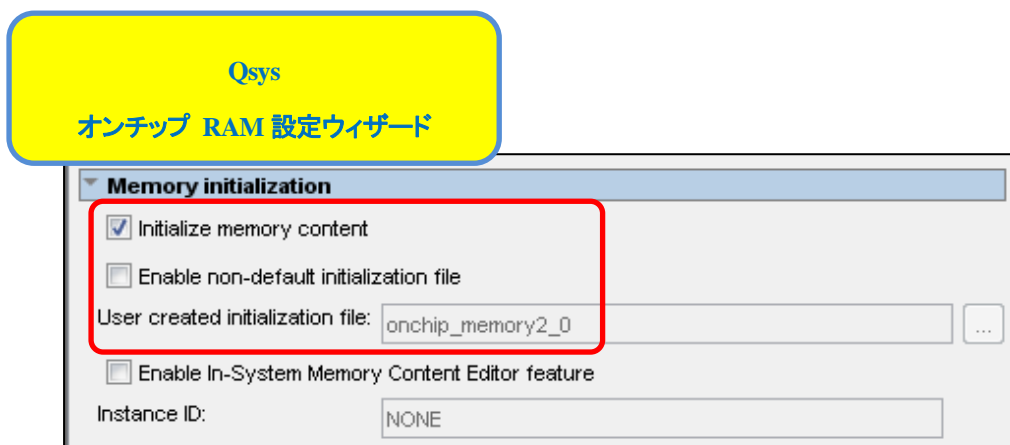


表 2-4-1-1 Memory initialization

| 項目                                     | 概要   |
|--|--|
| Initialize memory content              | On: 初期値ファイルを使用 (デフォルト)<br>On-Chip Memory のコンポーネント名で初期化用の HEX ファイル (.hex) が自動生成<br><br>Off: 初期値ファイルを未使用 |
| Enable non-default initialization file | On: ユーザが作成したファイルを使用<br><br>Off: 自動生成される初期値ファイルを使用 (デフォルト)  |

オンチップ RAM の Memory initialization の項目は表 2-4-1-1 のようになります。オンチップ RAM を Nios II のブート・メモリとして使用する場合には Initialize memory content を ON(デフォルト)にして、メモリの初期値を Nios II の起動時にデータがセットされた状態にすることが必要です。

Enable non-default initialization file は Initialize memory content を OFF(デフォルト)にすると、自動でオンチップ RAM のコンポーネントにつけた名前でも .hex ファイルがソフトウェア・ビルド時に生成されます。(第 2-4-2 章『初期値ファイルの生成』を参照。) ON にすると、ユーザが任意の .hex ファイルを指定することができます。

## 2-4-2. 初期値ファイルの生成

Nios II SBT でプログラム・データを含むオンチップ RAM の初期値データを生成します。

(手順 1) Nios II SBT にてプログラムをビルド

(手順 2) アプリケーションプロジェクト(\_bsp のついていないフォルダ)を選択し、右クリック ⇒ Make Targets  
⇒ Build を選択

mem\_init\_generate を実行するとメモリ初期化用 HEX ファイルと .qip ファイルが自動生成されます。

(手順 3) 自動生成された .qip ファイル(Quartus II IP ファイル)を Quartus II のプロジェクトに追加  
自動生成された 初期化用 HEX ファイルが Quartus II プロジェクトに反映されます。

(手順 4) Quartus II にて再コンパイルを実行

オンチップ RAM 用の初期化データを含むコンフィギュレーション・ファイル(.sof) が生成されます。



### 3. メモリの構成例

ブート・メモリと実行メモリの構成例について、基本的な 5 つのパターン(表 3-1)を紹介します。

表 3-1 メモリ構成例

| ブートメモリ         | 実行メモリ     | 章   |
|----------------|-----------|-----|
| オンチップ RAM      | オンチップ RAM | 3-1 |
| CFI フラッシュ      | オンチップ RAM | 3-2 |
| CFI フラッシュ      | 外部 RAM    | 3-3 |
| EPCS/EPCQ デバイス | オンチップ RAM | 3-4 |
| EPCS/EPCQ デバイス | 外部 RAM    | 3-5 |

#### 3-1. ブート・メモリと実行メモリが共にオンチップ RAM

ブート・メモリと実行メモリが共にオンチップ RAM の場合、ソフトウェア・プログラム用にフラッシュ・メモリや EPCS/EPCQ デバイスは必要ありません。

- Reset Vector : オンチップ RAM
- 実行メモリ : オンチップ RAM

オンチップ RAM を使用する場合、実行コードをメモリの初期値ファイルとしてコンフィギュレーション・ファイル (.sof) に組み込む為、コンフィギュレーション終了後にオンチップ RAM に実行コードがセットされている状態になります。

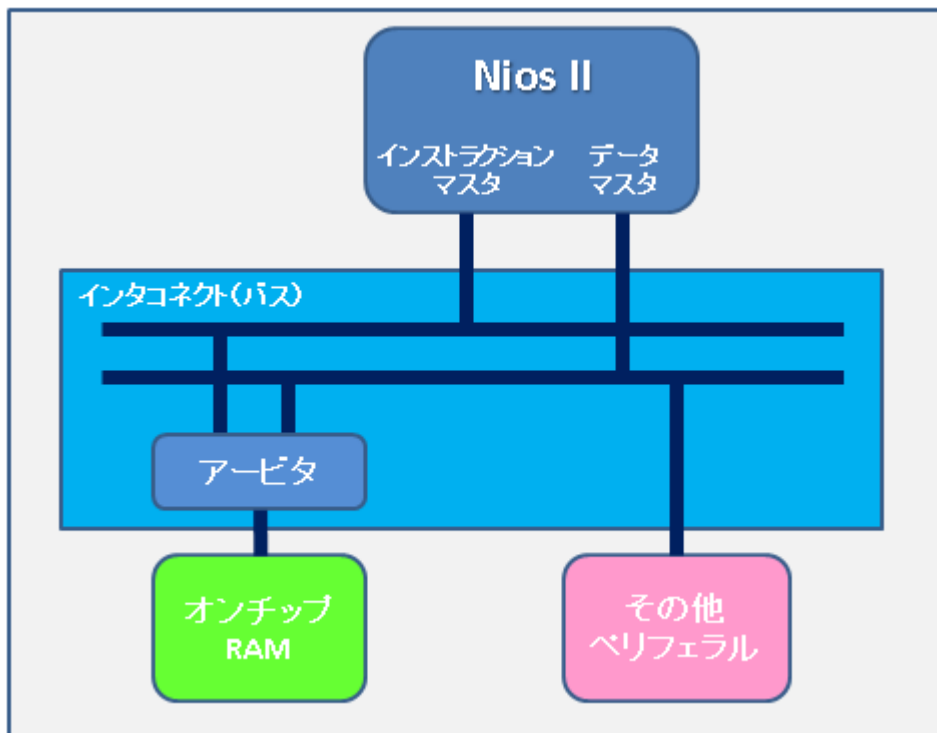


図 3-1-1 メモリ構成例 1

### 3-2. ブート・メモリが CFI フラッシュ、実行メモリがオンチップ RAM

フラッシュ・メモリに実行コードを格納し、起動時にオンチップ RAM に展開して実行します。

Reset Vector : CFI フラッシュ・メモリ

実行メモリ : オンチップ RAM

Flash Programmer によってフラッシュ・メモリに実行コードを書き込む際にブートコピーを追加して書き込みます。  
Nios II 起動時にブートコピーが動作し、オンチップ RAM に実行コードがコピーされます。

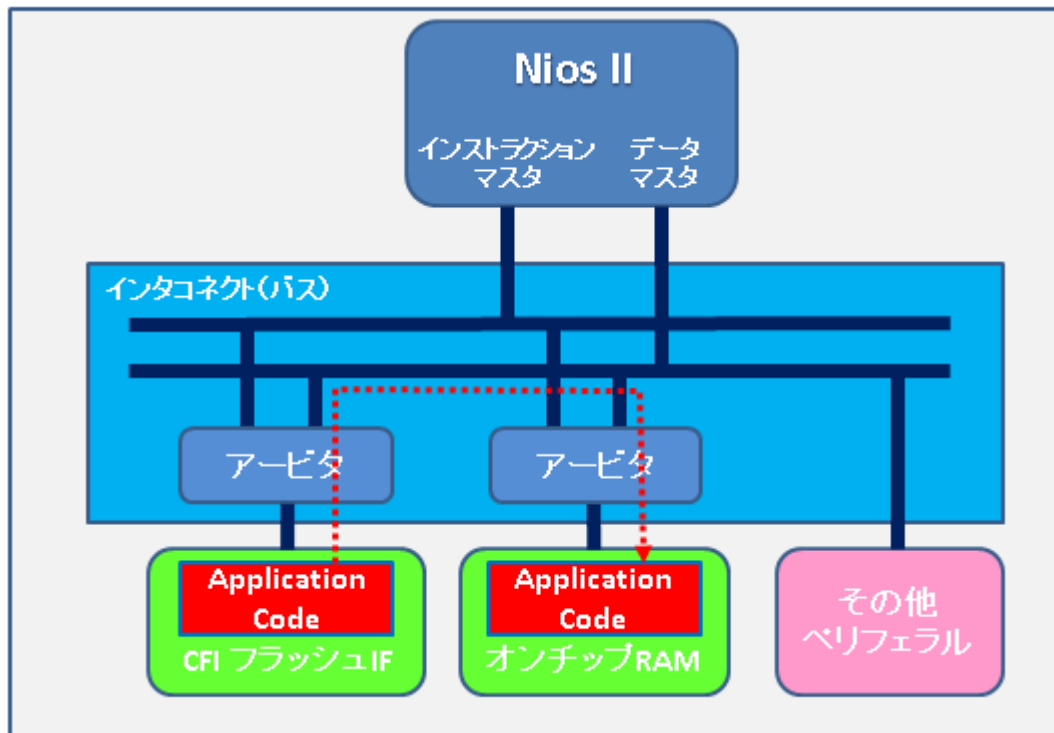


図 3-2-1 メモリ構成例 2

### 3-3. ブート・メモリが CFI フラッシュ、実行メモリが外部 RAM

Flash Programmer によってフラッシュ・メモリに実行コードを格納し、起動時に外部 RAM(SRAM 等)に展開して実行します。

- Reset Vector** : CFI フラッシュ
- 実行メモリ** : 外部 RAM(SRAM 等)

フラッシュ・メモリに実行コードを書き込む際にブートコピーを追加して書き込みます。Nios II 起動時にブートコピーが動作し、外部 RAM へ実行コードがコピーされます。

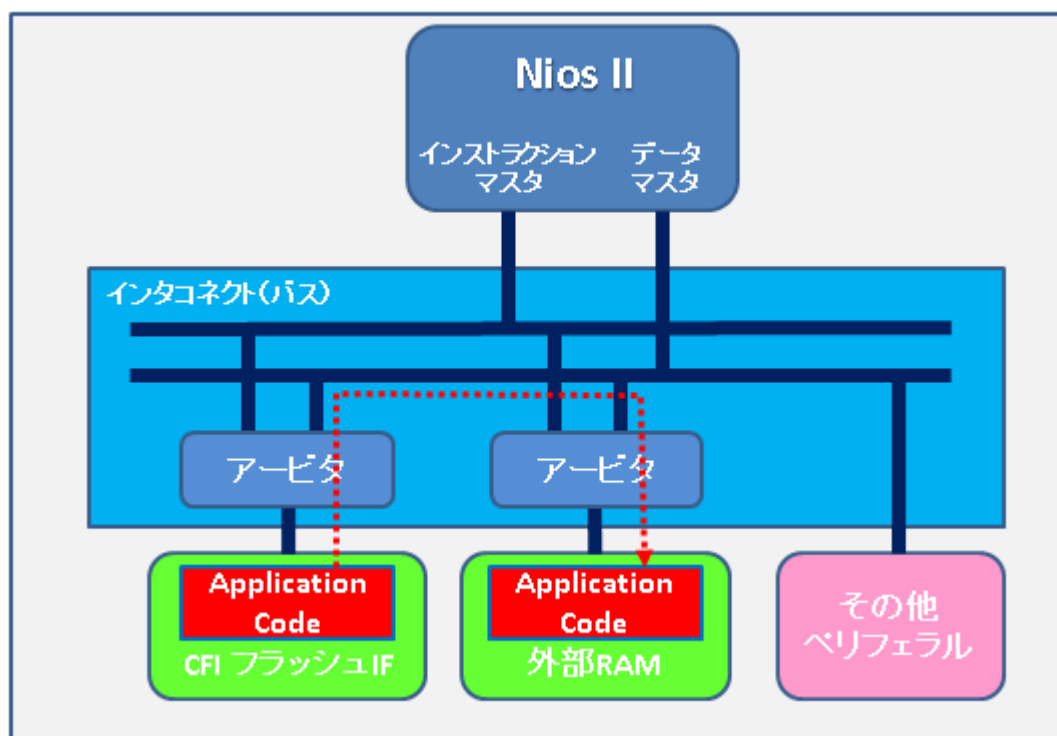


図 3-3-1 メモリ構成例 3

### 3-4. ブート・メモリが EPCS/EPCQ デバイス、実行メモリがオンチップ RAM

EPCS/EPCQ デバイスを使用したコンフィギュレーション・システムを構築し、EPCS/EPCQ デバイスに十分な空き領域がある場合、EPCS/EPCQ デバイスの空き領域にユーザ・データ(実行コード)を格納して使用することができます。

**Reset Vector** : EPCS/EPCQ コントローラ (※)

**実行メモリ** : オンチップ RAM

EPCS/EPCQ デバイスに実行コードを格納する場合は、図 3-4-1 のようにシステム内に EPCS/EPCQ コントローラが必要になります。EPCS/EPCQ コントローラを介して、EPCS/EPCQ デバイス内にある実行コードを読み出しオンチップ RAM にコピーします。

※EPCS/EPCQ コントローラ内のレジスタを介して EPCS/EPCQ デバイス内のメモリ領域にアクセスするため、EPCS/EPCQ デバイスを実行メモリに指定することはできません。

EPCS/EPCQ コントローラは内部に数 K ~ 十数 K バイトの小さなオンチップ ROM を持っており、この ROM にブートコピアが格納されます。オンチップ ROM 用の初期値データは、HEX ファイルが生成されコンフィギュレーション・ファイル(.sof)に含まれるため、コンフィギュレーション後にセットされた状態になります。

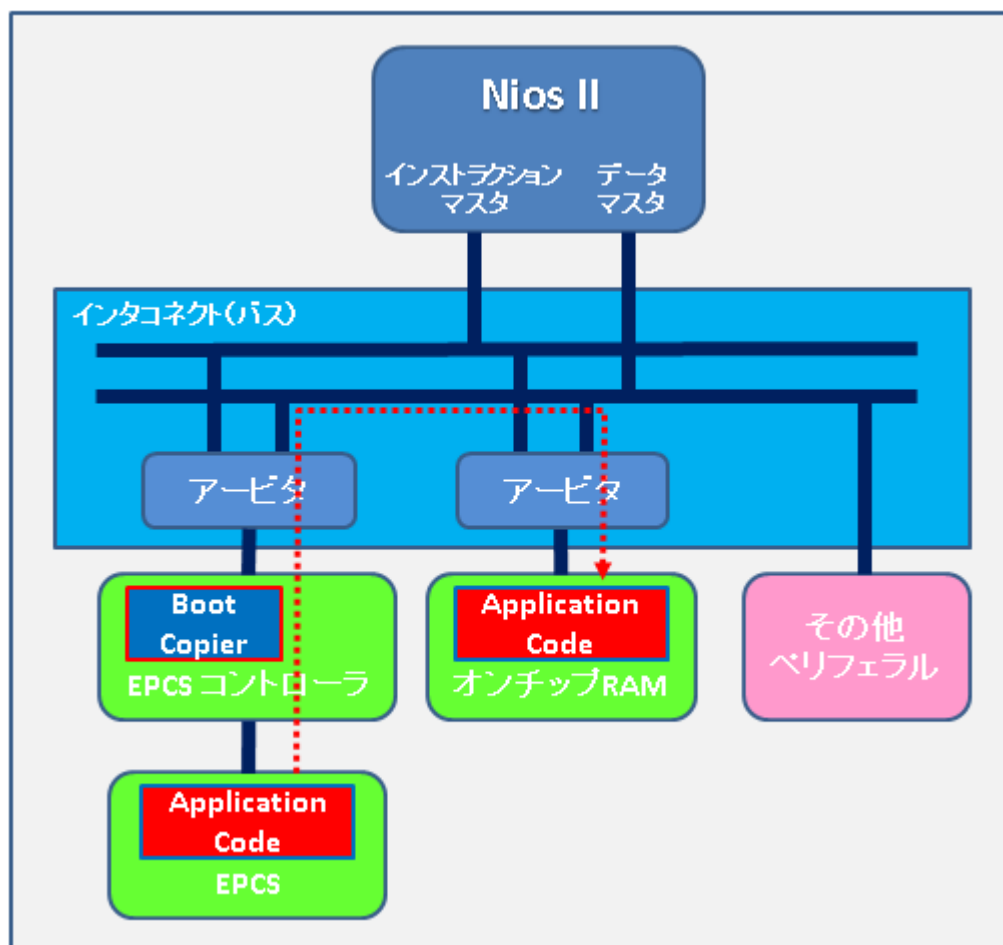


図 3-4-1 メモリ構成例 4

### 3-5. ブート・メモリが EPCS/EPCQ デバイス、実行メモリが外部 RAM

EPCS/EPCQ デバイスを使用したコンフィギュレーション・システムを構築し、EPCS/EPCQ デバイスに十分な空き領域がある場合、EPCS/EPCQ デバイスの空き領域にユーザ・データ(実行コード)を格納して使用することができます。

- Reset Vector : EPCS/EPCQ コントローラ (※)
- 実行メモリ : 外部 RAM(SRAM 等)

EPCS/EPCQ デバイスに実行コードを格納する場合は、図 3-5-1 のようにシステム内に EPCS/EPCQ コントローラが必要になります。EPCS/EPCQ コントローラを介して、EPCS/EPCQ デバイス内部にある実行コードを読み出し外部 RAM にコピーします。

※EPCS/EPCQ コントローラ内のレジスタを介して EPCS/EPCQ デバイス内のメモリ領域にアクセスするため、EPCS/EPCQ デバイスを実行メモリに指定することはできません。

EPCS/EPCQ コントローラは内部に数 K ~ 十数 K バイトの小さなオンチップ ROM を持っており、この ROM にブートコピアが格納されます。オンチップ ROM 用の初期値データは、コンフィギュレーション・ファイル(.sof)に含まれるため、コンフィギュレーション後にセットされた状態になります。

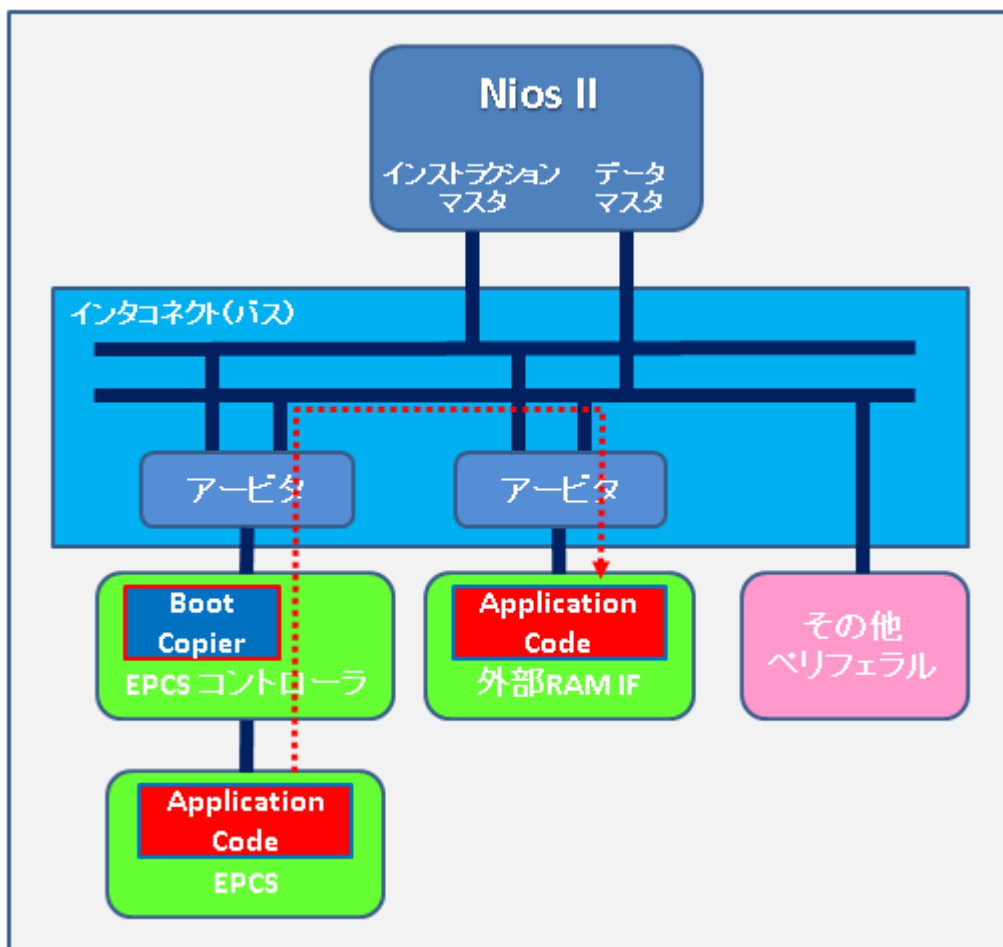


図 3-5-1 メモリ構成例 5

#### 4. 参考資料

- アプリケーション・ノート 458 『[Alternative Nios II Boot Methods](#)』
- ユーザ・ガイド 『[Nios II Flash Programmer](#)』
- ハンドブック 『[Nios II Software Build Tools](#)』

## 改版履歴

| Revision | 年月      | 概要 |
|----------|---------|----|
| 1        | 2015年6月 | 初版 |

### 免責およびご利用上の注意

弊社より資料を入手されましたお客様におかれましては、下記の使用上の注意を一読いただいた上でご使用ください。

1. 本資料は非売品です。許可無く転売することや無断複製することを禁じます。
2. 本資料は予告なく変更することがあります。
3. 本資料の作成には万全を期していますが、万一ご不明な点や誤り、記載漏れなどお気づきの点がありましたら、本資料を入手されました下記代理店までご一報いただければ幸いです。  
 株式会社アルティマ ホームページ: <http://www.altima.co.jp> 技術情報サイト EDISON: <https://www.altima.jp/members/index.cfm>  
 株式会社エルセナ ホームページ: <http://www.elsena.co.jp> 技術情報サイト ETS : <https://www.elsena.co.jp/elspear/members/index.cfm>
4. 本資料で取り扱っている回路、技術、プログラムに関して運用した結果の影響については、責任を負いかねますのであらかじめご了承ください。
5. 本資料は製品を利用する際の補助的な資料です。製品をご使用になる際は、各メーカー発行の英語版の資料もあわせてご利用ください。