

ローダプログラムに対応したアプリケーションの作成方法とシリアルFlashROMへの登録方法 (ブートモード1/3用)

Rev1.30 2012/10/4
 DEFバージョン11.00Aより
 DEFバージョン12.10A変更
 DEFバージョン12.20A変更

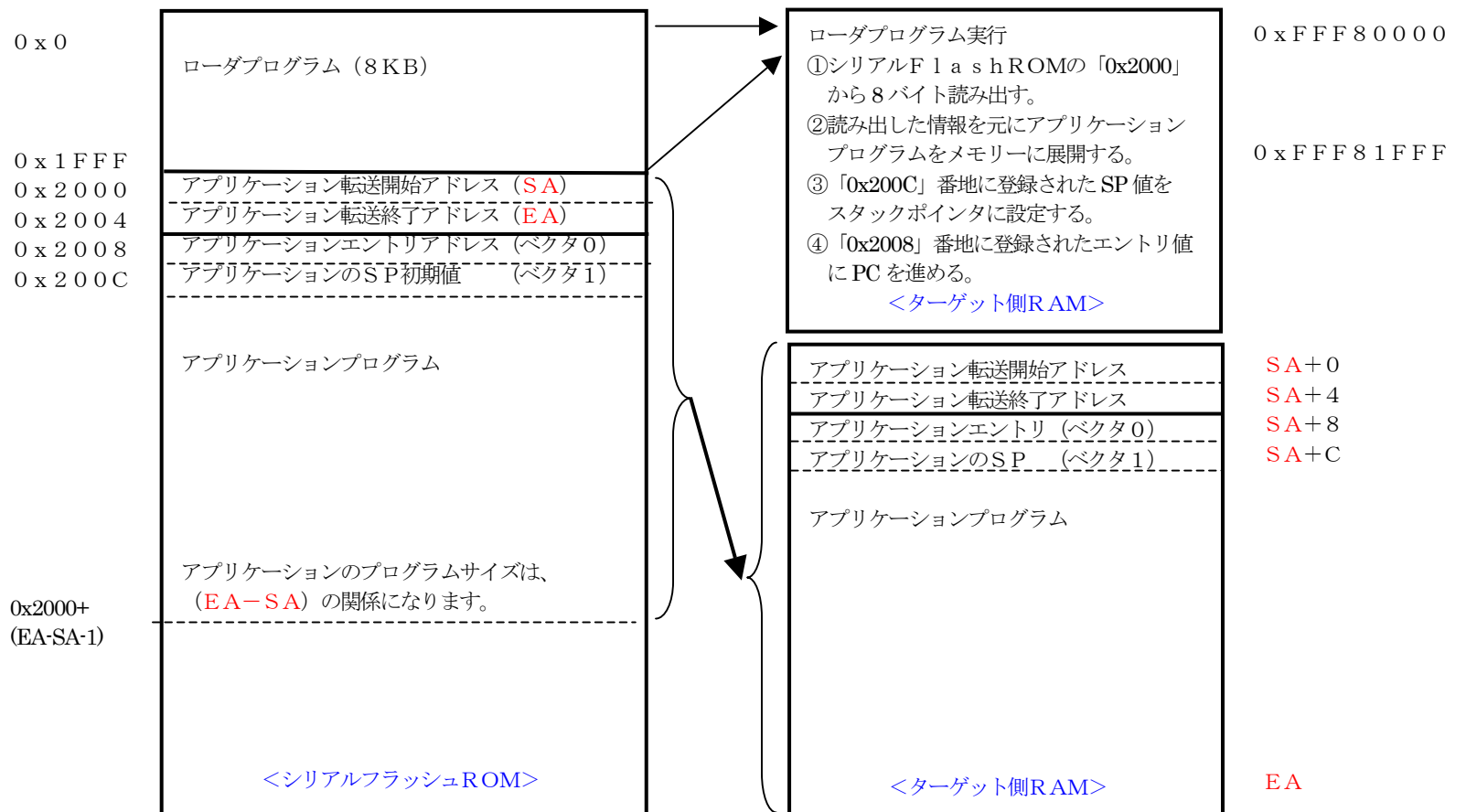
【対象CPU】

- ROMレス品種が対象になります。(SH7262/4/6/7)

【ローダとアプリケーションの関係】

- ブートモード1/3の場合、電源オン時にRSPiOに接続されているシリアルFlashROMの「0~0x1FFF」番地に格納されたローダプログラムを内蔵RAM「0xFFF80000~0xFFF81FFF」番地にロードし、「0xFFF80000」番地から実行する仕様になっています。
 ローダプログラムは、シリアルFlashROMの「0x2000」番地に格納されているアプリケーションのロード用管理テーブル情報に元づき指定アドレスに転送後、指定エントリーアドレスから実行させます。

<関係図>



【アプリケーションの作成】

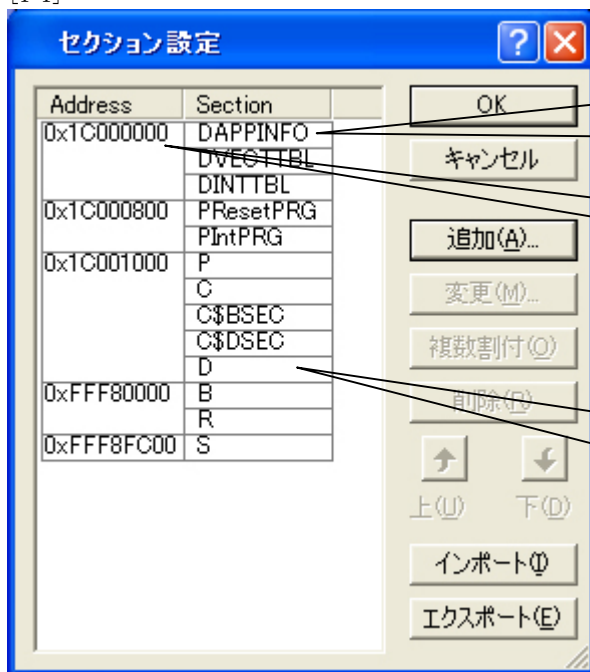
1. ロータ対応のアプリケーションを作成する場合、上図のSA/EAの情報をアプリケーションの先頭に追加する必要があります。追加方法として一例を記述します。
 - 1) アプリケーション情報用のソースファイル [appinfo.c]を準備します。

```
/******  
// appinfo.c  
/******  
#pragma section APPINFO  
/******  
typedef struct {  
    void *app_top; // アプリケーション開始アドレス  
    void *app_end; // アプリケーション終了アドレス  
} APPINFO;  
/******  
APPINFO appinfo = {  
    __sectop("DAPPINFO"),  
    __secend("D")  
};  
/******
```

- 2) Hewメニューの<プロジェクト>-<ファイルの追加>で「appinfo.c」を追加します。

- 3) <セクション設定>で、ロケーション指定をします。

[1-1]



⚠
アプリケーション情報セクション名を先頭で定義する。

__sectop("DAPPINFO")

ローダにより、転送させ実行する先頭アドレスを指定する。

このロケーション指定ですと、セクション"D"がシリアルFlashROMに書き込む最終セクションになります。

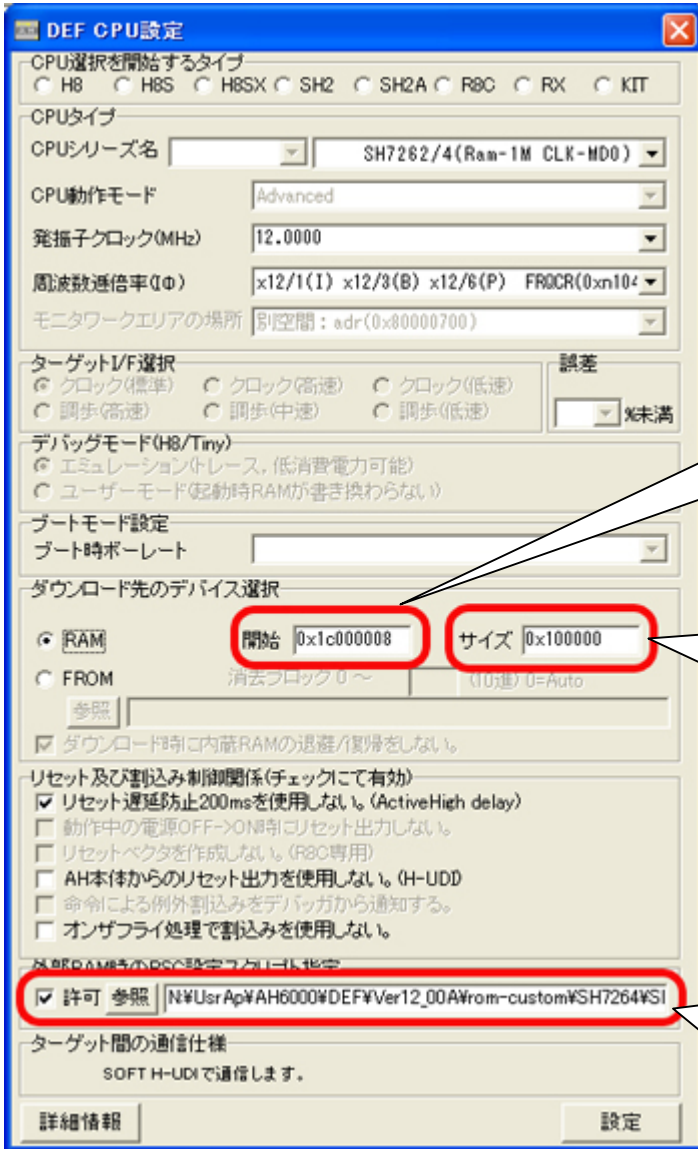
__secend("D")

- 4) サンプルソフト

「sample¥Hew4_SH2_Sflash」フォルダに、北斗製基板「HSB7264」に対応したサンプルを用意してありますので、詳細はサンプルをご覧ください。

【アプリケーションのデバッグ】

1. ロード対応のアプリケーションを単独でデバッグする場合、下記のCPU設定が必要になります。
[2-1]



⚠️ ポイント

ロード対応のアプリケーションプログラムは、通常のROM対応プログラムのリセットベクターアドレスが「TOP+8」された位置になります。よって、今回の例ですと開始は、「0x1C00_0008」になります。

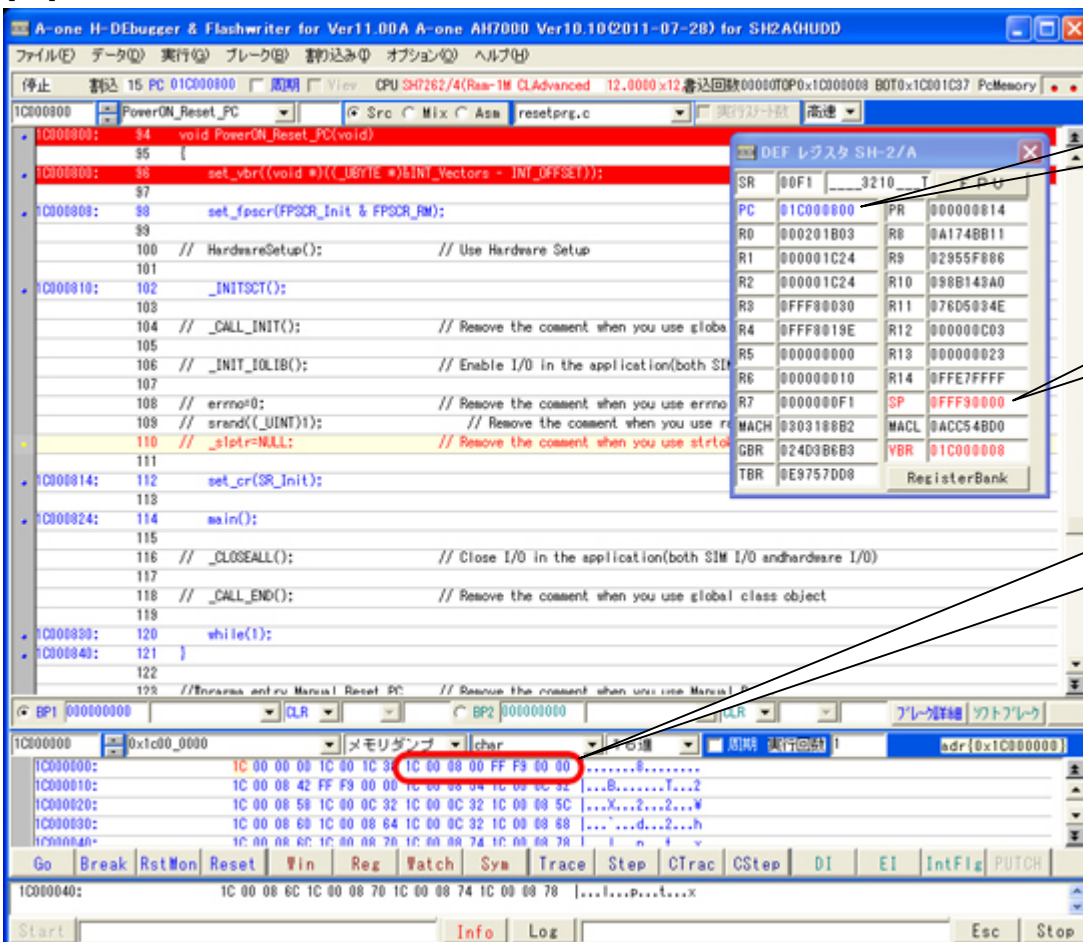
⚠️ ポイント

アプリケーションプログラムの実行可能なRAMサイズを指定します。「TOP+8」の「+8」は無視して「TOP」からのサイズになります。よって、今回の例ですとサイズ「0x10_0000」になります。

作成したアプリケーションプログラムのデバッグにBSC設定が必要な場合は、スクリプトファイルを指定します。

SH7264の場合
インストールディレクトリ
"c:\Program Files\A-one\DEF\rom-custom\SH7264"
に、例として「SH7264-BSC-HSB.log」が用意してありますので、目的ハードにカスタマイズして下さい。

2. サンプルソフトのダウンロード終了画面
[3-2]



PC値が「0x1C000800」になっているのが確認できます。

SP値が「0xFFF90000」になっているのが確認できます。

0x1C000008: 1C,00,08,00 エントリアドレス
0x1C00000C: FF,F9,00,00 初期SP値
の確認ができます。

デバッグ操作に関しては、標準仕様と同じ操作になります。

⚠️ デバッグ中は、「ブートモード0」にするか、もしくは「ブートモード0」に出来ない回路構成（アドレス・データバスを汎用I/Oに使用）の場合は、シリアルフラッシュROMの内容を全消去した状態でデバッグして下さい。

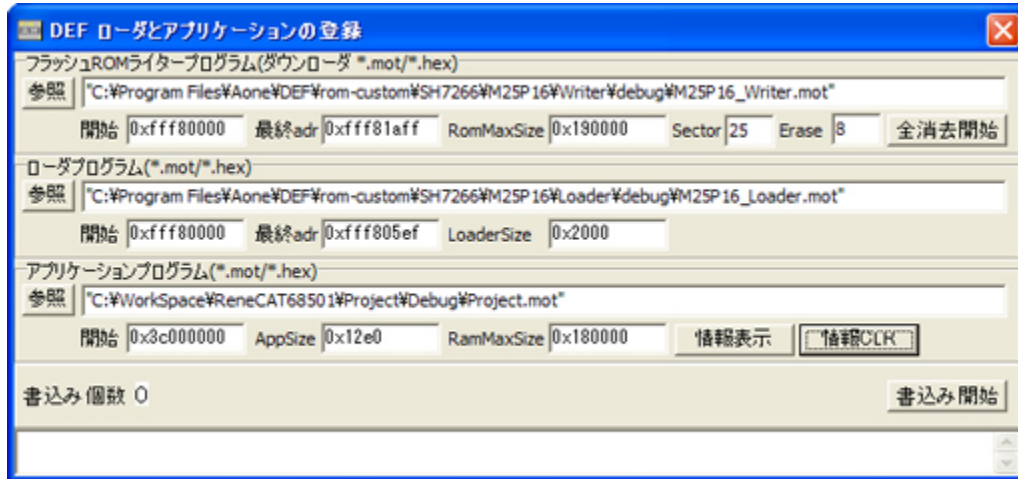
「ブートモード1・3」のままですとリセット解除後、内蔵高速RAMの先頭から「0x2000」エリアは常時「0xFF」になります。

【シリアルフラッシュROMに登録】

1. デバッグが完了し、シリアルFlashROMに書き込む手順を説明します。なお、CPU設定は、デバッグ時と同じ状態にしておいて下さい。

1) DEFメニュー <オプション>-<ローダとアプリケーションの登録>をクリックします。

[4-1]



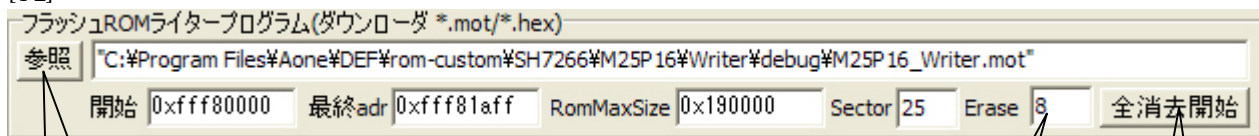
各プログラムを登録します。

- (1) フラッシュROMライタープログラム (*.mot)
- (2) ローダプログラム (*.mot)
- (3) アプリケーションプログラム (*.mot)

上記3プログラムファイルは、必ず指定して下さい。

2) フラッシュROMライタープログラム (*.mot) を指定します。

[4-2]

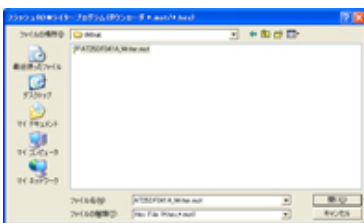


「開始」ライタープログラムの開始アドレス
 「最終 adr」ライタープログラムの終了アドレス
 「RomMaxSize」ROMの最大有効サイズ
 「sector」ライタープログラム内部に登録されたセクター数

<操作>

「参照」ボタンをクリックします。

[4-2-1]



インストールディレクトリ

"c:\Program Files\Aone\DEF\rom-custom\SH7264" 下の "\$(ProjectName)\Debug" に作成したヘキサファイルがありますので指定します。

例) \$(ProjectName):AT25DF041A_Writer

<操作1>

アプリケーションのサイズに応じて消去するセクター数を指定することができます。必要な場合は、ここにセクター数を指定して下さい。【1~n】

<操作2>

「全消去開始」ボタンをクリックするとシリアルフラッシュROMを消去します。

<状態>

メッセージBOXが表示され処理が推移します。

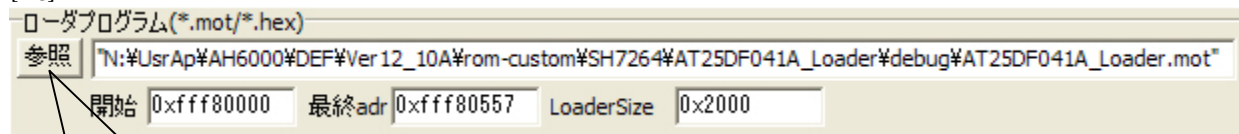
[4-6-1]



- (1) エミュレーションモードの遷移
- (2) 書き込みプログラムの転送
- (3) セクターイレーズ
- (4) 正常終了 (BOX消去)

3) ローダプログラム (*.mot) を指定します。

[4-3]



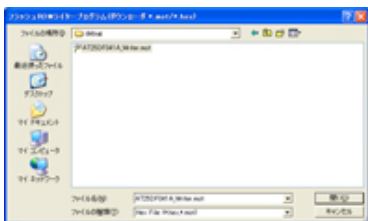
「開始」 ローダプログラムの開始アドレス
 「最終 adr」 ローダプログラムの終了アドレス
 「LoaderSize」 ローダプログラムの最大有効サイズ

ローダプログラムの開始アドレスは、MPU 品種によって固定になっていますので、仕様通りのアドレスにする必要があります。

<操作>

「参照」 ボタンをクリックします。

[4-2-1]



インストールディレクトリ

"c:\Program Files\Aone\DEF\rom-custom\SH7264"

下の"\${ProjectName}\Debug"

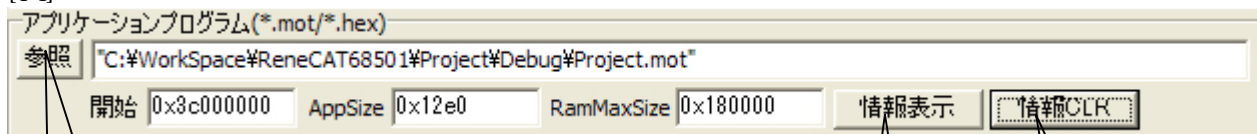
に作成したヘキサファイルがありますので指定します。

例) \${ProjectName}:AT25DF041A_Loader

「AT25DF041A_Loader.mot」

4) アプリケーションプログラム (*.mot) を指定します。

[4-4]



「開始」 アプリケーションプログラムの開始アドレス
 「AppSize」 アプリケーションプログラムのサイズ
 「RamMaxSize」 CPU 設定に登録した最大有効 RAM サイズ

<操作1>

「参照」 ボタンをクリックします。

[4-2-1]



\$(Workspace)\\$(Project)\Debug 下のヘキサファイルを指定します。

<操作2>

「情報」 ボタンをクリックします。

ライター/ローダ/アプリケーションプログラムの内部情報を表示します。

<操作3>

「情報 CLR」 ボタンをクリックしますと「情報」で表示した内部情報画面を消去します。

5) 「情報」画面の説明

[4-5]

```

<FlashRom Sector>
SA0000 (Start)0x00000000 (end)0x0000FFFF (size)0x00010000
SA0001 (Start)0x00010000 (end)0x0001FFFF (size)0x00010000
SA0002 (Start)0x00020000 (end)0x0002FFFF (size)0x00010000
SA0003 (Start)0x00030000 (end)0x0003FFFF (size)0x00010000
SA0004 (Start)0x00040000 (end)0x0004FFFF (size)0x00010000
SA0005 (Start)0x00050000 (end)0x0005FFFF (size)0x00010000
SA0006 (Start)0x00060000 (end)0x0006FFFF (size)0x00010000
SA0007 (Start)0x00070000 (end)0x0007FFFF (size)0x00010000
<Loader Program>
(Start)0x00000000 (end)0x00000557
<Application Program>
(Start)0x00000200 (end)0x00003C37
(ForwardStart)0x01C00000
(ForwardEnd)0x01C001C38
(ProgramEntry)0x01C000800
(StackPoint)0x0FFF90000
    
```

ライタープログラム内部に登録した「セクター別」開始/終了/サイズを表示します。

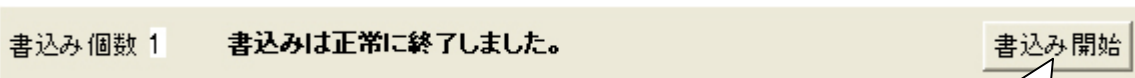
ローダプログラムのシリアルフラッシュROMに格納するアドレスを表示します。

アプリケーションプログラムのシリアルフラッシュROMに格納するアドレスを表示します。

(ForwardStart)-- 「ローダに転送させる開始アドレス」
 (ForwardEnd)--- 「ローダに転送させる終了アドレス」
 (ProgramEntry)-- 「アプリケーションの開始アドレス」
 (StackPoint)----- 「アプリケーションの初期S P値」

6) シリアルFlashROMに書き込みます。

[4-6]



正常終了しますと、メッセージが表示され、「書き込み回数」が+1されます。

<操作>
 「書き込み開始」ボタンをクリックします。
 <状態>
 メッセージBOXが表示され処理が推移します。

[4-6-1]



- (1)エミュレーションモードの遷移
- (2)書き込みプログラムの転送
- (3)セクターイレーズ
- (4)ユーザモードでダウンロード
- (5)正常終了 (BOX消去)

7) 動作確認をします。

- (1) ターゲット基板の電源をオフにします。
- (2) ターゲット基板のブートモードを「1 or 3」にします。
- (3) ターゲット基板のASEMDを「1:High」側に設定します。
- (4) ターゲット基板の電源をオンにします。
- (5) アプリケーションプログラムの動作確認が出来ましたら正常にシリアルFlashROMに書き込んだことになります。

8) プログラムコードのベリファイ方法

- (1) ターゲット基板の電源をオフにします。
- (2) ターゲット基板のブートモードを「1 or 3」にします。
- (3) ターゲット基板のASEMDを「1:High」側に設定します。
- (4) ターゲット基板の電源をオンにします。
- (5) ローダーによる転送終了を待ちます。
- (6) ターゲット基板のASEMDを「0:Low」側にします。(エミュレーション遷移可能状態)
- (7) DEFを起動して右下隅の「Start」をクリックします。
- (8) DEFメニューの<ファイル>-<ベリファイ>でRAMに転送されたプログラムと登録したプログラムとベリファイします。

以上で、アプリケーションの作成方法とシリアルFlashROMへの登録方法の説明が終了です。