

## シミュレーション機能 (オフライン操作) for AH8000

### 1. 機能と特徴

- AH8000 本体およびターゲット基板が無い状態でも、プログラム動作確認をメモリ上でシミュレーションする機能です。
- sleep 命令以外はシミュレートします。(sleep 命令はPC 加算のみ)
- ブレーク設定は8 点まで設定可能です。(データブレーク機能有り)
- 割り込みを擬似的に発生させることができます。
- 割り込みのプライオリティ判定をシミュレートします。(SH-2/H8S: 割り込みモード2 の動作時対応)
- 最新4K ステップ分の実行履歴 (トレース) と、実行毎の全レジスタ内容とメモリアクセス履歴 (最終8 ポイント) を記憶しています。

### 2. 対応品種

- RX シミュレーション
- SH-2/E シミュレーション
- SH-2A シミュレーション
- H8SX シミュレーション
- H8S シミュレーション
- H8/300 シミュレーション

### 3. 特記事項

3-1) 本書の説明は、[DEF8K Ver24.00B](#) からの対応になります。

3-2) シミュレーション能力は、PC 機的能力により影響を受けます。

下記計測データは、全てシミュレーション指定した場合です。

- 実測値 i5-7500 (3.40GHz 16.0GB-RAM) 1sec で 100,000 ステップ (RX シリーズ)
- 実測値 Pentium4 (2.80GHz 512MB-RAM) 1sec で 500,000 ステップ (SH-2/E/A)
- 実測値 Pentium4 (2.80GHz 512MB-RAM) 1sec で 100,000 ステップ (H8SX シリーズ)
- 実測値 Pentium4 (2.80GHz 512MB-RAM) 1sec で 100,000 ステップ (H8/300L, H8/300H, H8/300HTiny)
- 実測値 Pentium4 (2.80GHz 512MB-RAM) 1sec で 100,000 ステップ (H8S/2200, 2300, 2500, 2600 シリーズ)
- 実測値 Pentium4 (2.80GHz 512MB-RAM) 1sec で 100,000 ステップ (H8S/2100 シリーズ)

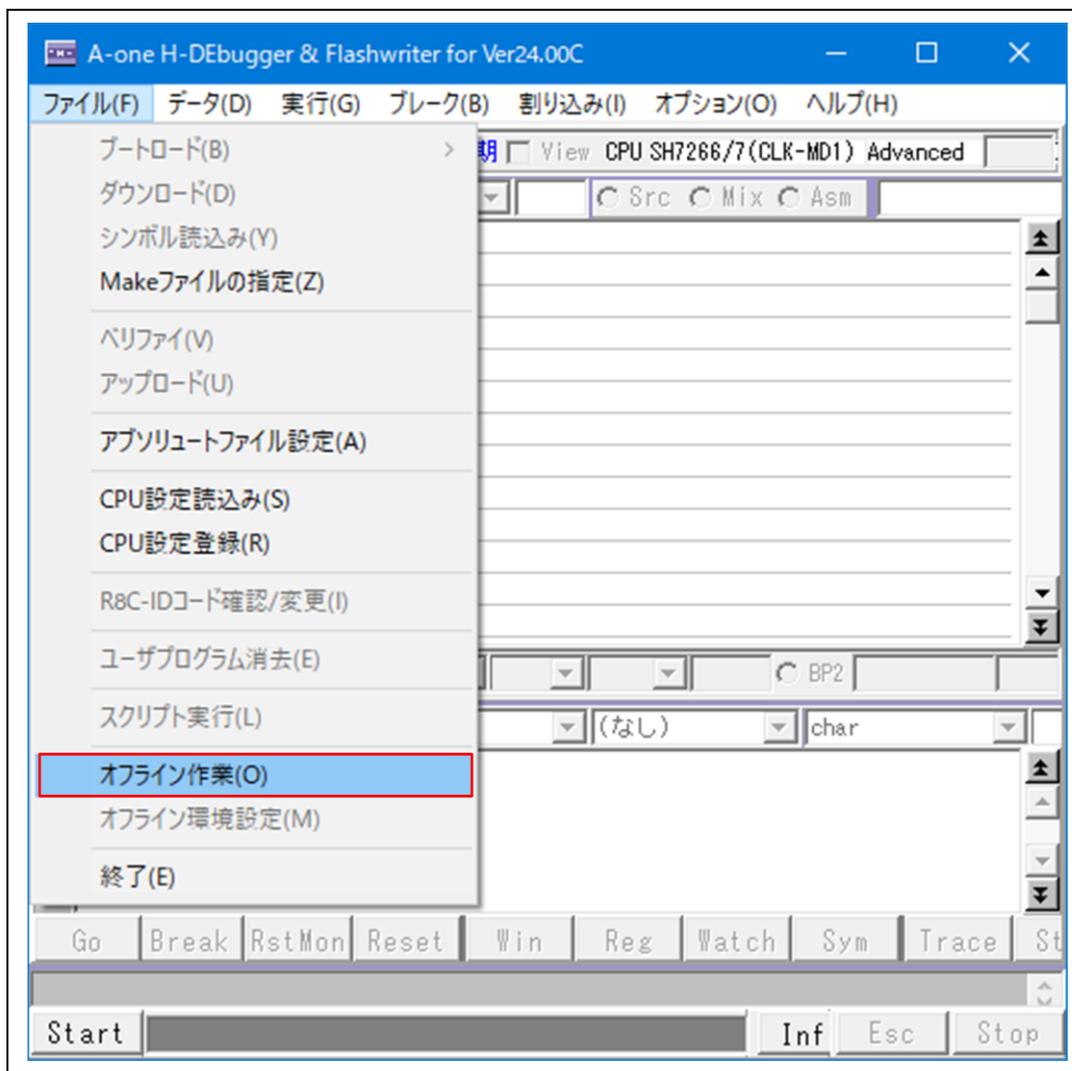
3-3) RX シリーズの特記事項

- 整数のゼロ除算時は、商をゼロにします。
- 浮動小数点演算の結果は、RX 側と許容誤差の範囲内で結果が違う場合があります。
- 浮動小数点の丸め処理で  $+\infty$  と  $-\infty$  は無視して「最近値への丸め」で処理しています。
- RX 内部演算で ACC を利用するコードの場合、シミュレーションでは ACC を使用していません。(FMUL 等)
- コード実行時レジスタ内容不定のコードの場合は、ゼロ払いしています。

#### 4. 操作説明

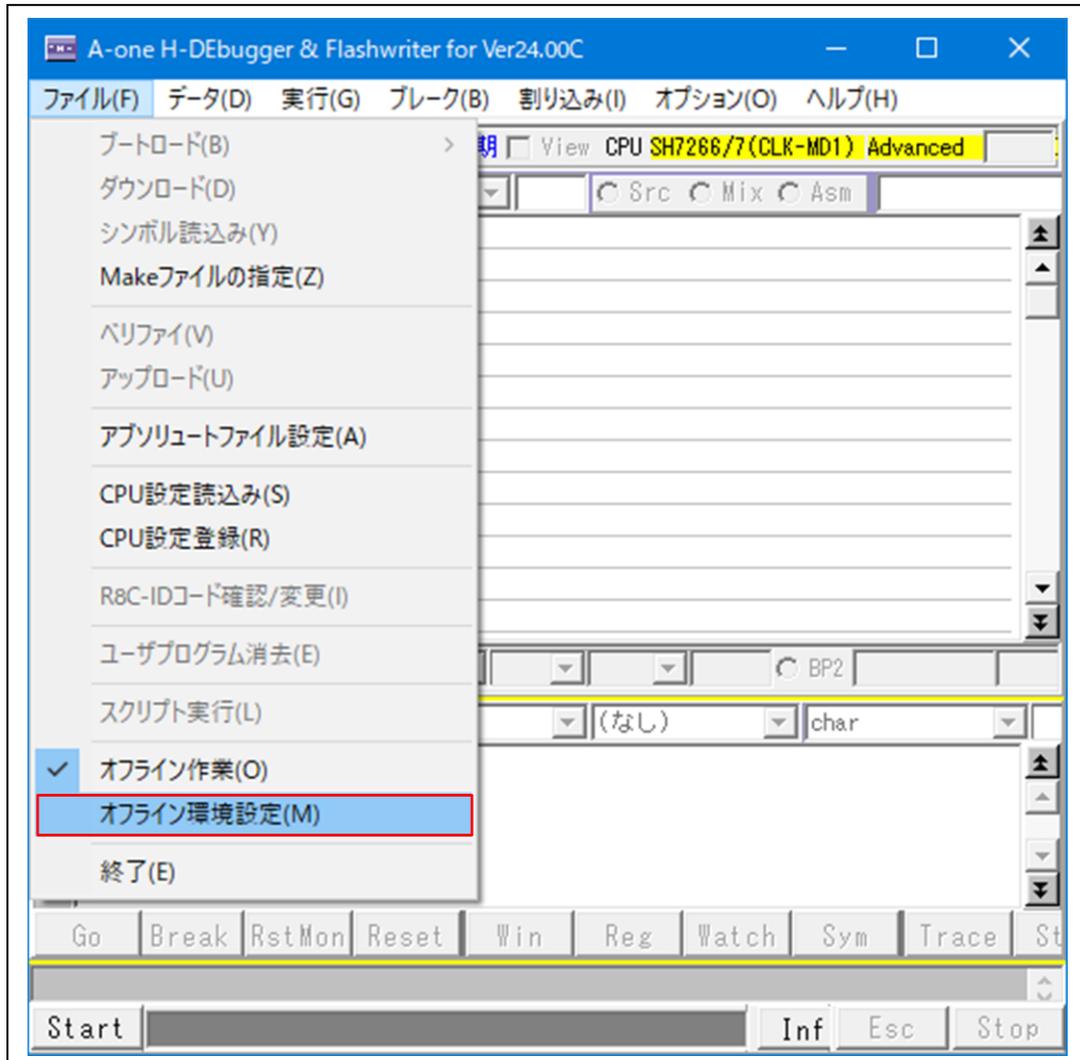
4-1) シミュレーションモードへの切り替え

<ファイルメニュー>—<オフライン作業>をクリック



#### 4-2) オフライン環境設定例

<ファイルメニュー>—<オフライン作業>をクリック

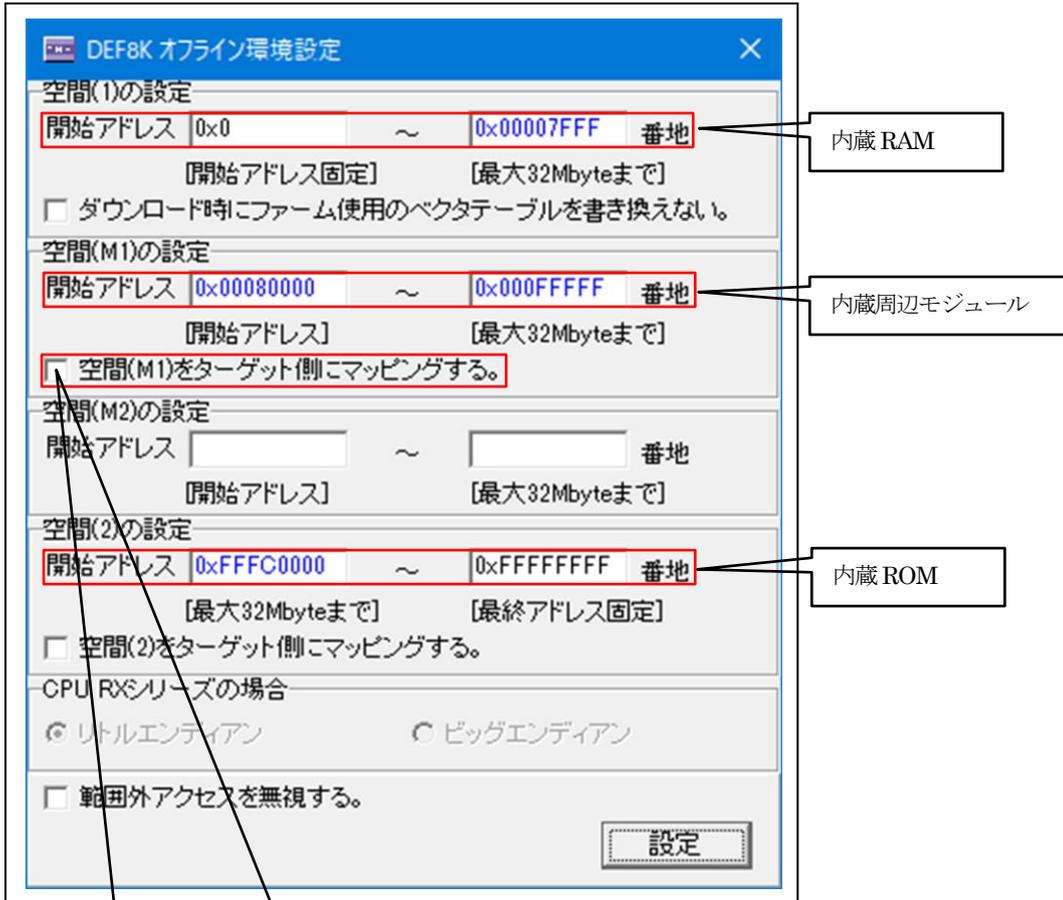


1) 「空間 (x) をターゲット側にマッピングする。」とは

この機能にチェックしますと、指定された空間への Read/Write アクセス時にターゲット側の空間をエミュレートします。

つまり、この機能を有効にする場合は、必ず、ターゲット側とエミュレート接続されていることが条件になります。

2) RX シミュレーション (例 RX230 の場合)



□ : マッピング不要の場合はノーチェック  
 ☑ : I/O のみをエミュレートする場合は、「空間 (M1) をターゲット側にマッピングする」にチェックする。

3) RX シミュレーション (例 RX63NE の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス 0x0 ~ 0x0001FFFF 番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス 0x00080000 ~ 0x000FFFFFFF 番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス 0xFFE00000 ~ 0xFFFFFFFF 番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU RXシリーズの場合  
 リトルエンディアン  ビッグエンディアン

範囲外アクセスを無視する。

設定

内蔵 RAM

内蔵周辺モジュール

内蔵 ROM

□ : マッピング不要の場合はノーチェック  
 ☑ : I/Oのみをエミュレートする場合は、「空間 (M1) をターゲット側にマッピングする」にチェックする。

4) RX シミュレーション (例 RX66NN の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス 0x0 ~ 0x0007FFFF 番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス 0x00080000 ~ 0x000FFFFF 番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス 0xFE7F5D00 ~ 0xFE7F5D7F 番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス 0xFFC00000 ~ 0xFFFFFFFF 番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU RXシリーズの場合  
 リトルエンディアン  ビッグエンディアン

範囲外アクセスを無視する。

設定

内蔵 RAM

内蔵周辺モジュール

オプション設定メモリ

内蔵 ROM

: マッピング不要の場合はノーチェック  
 : I/O のみをエミュレートする場合は、「空間 (M1) をターゲット側にマッピングする」にチェックする。

5) RX シミュレーション (例 RX72TK の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス 0x0 ~ 0x0001FFFF 番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス 0x00080000 ~ 0x000FFFFF 番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス 0x00120040 ~ 0x0012007F 番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス 0xFFFF0000 ~ 0xFFFFFFFF 番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU RXシリーズの場合  
 リトルエンディアン  ビッグエンディアン

範囲外アクセスを無視する。

設定

: マッピング不要の場合はノーチェック  
 : I/Oのみをエミュレートする場合は、「空間 (M1) をターゲット側にマッピングする」にチェックする。

6) SH-2 シミュレーション (例 SH7051 の場合)

: マッピング不要の場合はノーチェック  
 : 「空間 (M1) をターゲット側にマッピングする」の指定にする場合、  
 BOOT 仕様の品種は、モニタワークエリアを【固定番地方式】にする。

7) SH-2E シミュレーション (例 SH7055 の場合)

: マッピング不要の場合はノーチェック  
 : 「空間 (2) をターゲット側にマッピングする」の指定にする場合、  
 BOOT 仕様の品種は、モニタワークエリアを【固定番地方式】にする。

8) SH-2 シミュレーション (例 SH7125 の場合)

: マッピング不要の場合はノーチェック  
 : I/Oのみをエミュレートする場合は、「空間 (2) をターゲット側にマッピングする」にチェックする。

9) SH-2A シミュレーション (例 SH7211 の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス 0x0 ~ 0x0007FFFF 番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス 0xFFF80000 ~ 0xFFF87FFF 番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス 0xFFFC0000 ~ 0xFFFFFFFF 番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU H8/300Hシリーズの場合  
 16Mバイト空間  1Mバイト空間

範囲外アクセスを無視する。

設定

内蔵 ROM

高速内蔵 RAM

内蔵周辺モジュール

: マッピング不要の場合はノーチェック  
 : I/Oのみをエミュレートする場合は、「空間 (2) をターゲット側にマッピングする」にチェックする。

1 0) SH-2A シミュレーション (例 SH7267 の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス  ~  番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス  ~  番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス  ~  番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス  ~  番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU H8/300Hシリーズの場合  
 16Mバイト空間  1Mバイト空間  
 範囲外アクセスを無視する。

設定

内蔵 RAM

高速内蔵 RAM

内蔵周辺モジュール

: マッピング不要の場合はノーチェック  
 : I/Oのみをエミュレートする場合は、「空間(2)をターゲット側にマッピングする」にチェックする。

1 1) H8SX シミュレーション (例 H8SX/1544 の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス 0x0 ~ 0x00FFBFFF 番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス 0x00FFEA00 ~ 0x00FFFFFF 番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU H8/300Hシリーズの場合  
 16Mバイト空間  1Mバイト空間  
 範囲外アクセスを無視する。

設定

内蔵ROM/RAM

内蔵周辺モジュール

: マッピング不要の場合はノーチェック  
 : I/Oのみをエミュレートする場合は、「空間(2)をターゲット側にマッピングする」にチェックする。

1 2) H8SX シミュレーション (例 H8SX/1664 の場合)

□ : マッピング不要の場合はノーチェック  
 ☑ : I/Oのみをエミュレートする場合は、「空間 (2) をターゲット側にマッピングする」にチェックする。

1 3) H8S シミュレーション (例 H8S/20115R の場合)

: マッピング不要の場合はノーチェック  
 : I/Oのみをエミュレートする場合は、「空間 (2) をターゲット側にマッピングする」にチェックする。

: マッピング不要の場合はノーチェック  
 : I/Oのみをエミュレートする場合は、「空間 (M1) をターゲット側にマッピングする」にチェックする。

1 4) H8S シミュレーション (例 H8S/2144A の場合)

- : マッピング不要の場合はノーチェック
- : 「空間 (2) をターゲット側にマッピングする」の指定にする場合、BOOT 仕様の品種は、モニタワークエリアを【固定番地方式】にする。

1 5) H8S シミュレーション (例 H8S/2268 の場合)

DEF8K オフライン環境設定

空間(1)の設定

開始アドレス 0x0 ~ 0x00FFEFBF 番地

内蔵ROM/RAM

開始アドレス固定 [最大32Mbyteまで]

ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定

開始アドレス [ ] ~ [ ] 番地

開始アドレス [最大32Mbyteまで]

空間(M1)をターゲット側にマッピングする。

空間(M2)の設定

開始アドレス [ ] ~ [ ] 番地

開始アドレス [最大32Mbyteまで]

空間(2)の設定

開始アドレス 0x00FFFC00 ~ 0x00FFFFFF 番地

内蔵周辺モジュール

最大32Mbyteまで [最終アドレス固定]

空間(2)をターゲット側にマッピングする。

CPU H8/300Hシリーズの場合

16Mバイト空間  1Mバイト空間

範囲外アクセスを無視する。

設定

□ : マッピング不要の場合はノーチェック  
 : 「空間 (2) をターゲット側にマッピングする」の指定にする場合、  
 BOOT 仕様の品種は、モニタワークエリアを【固定番地方式】にする。

1 6) H8S シミュレーション (例 H8S/2378 の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス 0x0 ~ 0x00FFBFFF 番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換えなない。

空間(M1)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス 0x00FFFD00 ~ 0x00FFFFFF 番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU H8/300Hシリーズの場合  
 16Mバイト空間  1Mバイト空間

範囲外アクセスを無視する。

設定

内蔵 ROM/RAM

内蔵周辺モジュール

□: マッピング不要の場合はノーチェック  
 ☑: I/Oのみをエミュレートする場合は、「空間(2)をターゲット側にマッピングする」にチェックする。

1.7) H8S シミュレーション (例 H8S/2426 の場合)

DEF8K オフライン環境設定

空間(1)の設定

開始アドレス 0x0 ~ 0x00FFBFFF 番地

[開始アドレス固定] [最大32Mbyteまで]

ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定

開始アドレス [ ] ~ [ ] 番地

[開始アドレス] [最大32Mbyteまで]

空間(M1)をターゲット側にマッピングする。

空間(M2)の設定

開始アドレス [ ] ~ [ ] 番地

[開始アドレス] [最大32Mbyteまで]

空間(2)の設定

開始アドレス 0x00FFFC80 ~ 0x00FFFFFF 番地

[最大32Mbyteまで] [最終アドレス固定]

空間(2)をターゲット側にマッピングする。

CPU H8/300Hシリーズの場合

16Mバイト空間  1Mバイト空間

範囲外アクセスを無視する。

設定

内蔵 ROM/RAM

内蔵周辺モジュール

: マッピング不要の場合はノーチェック  
 : I/Oのみをエミュレートする場合は、「空間(2)をターゲット側にマッピングする」にチェックする。

1 8) H8S シミュレーション (例 H8S/2552 の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス 0x0 ~ 0x00FFEFBF 番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス 0x00FFF800 ~ 0x00FFFFFF 番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU H8/300Hシリーズの場合  
 16Mバイト空間  1Mバイト空間

範囲外アクセスを無視する。

設定

内蔵 ROM/RAM

内蔵周辺モジュール

- : マッピング不要の場合はノーチェック  
 ☑ : 「空間 (2) をターゲット側にマッピングする」の指定にする場合、  
 BOOT 仕様の品種は、モニタワークエリアを【固定番地方式】にする。

1 9) H8S シミュレーション (例 H8S/2612 の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス 0x0 ~ 0x00FFEFBF 番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス 0x00FF800 ~ 0x00FFFFFF 番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU H8/300Hシリーズの場合  
 16Mバイト空間  1Mバイト空間

範囲外アクセスを無視する。

設定

内蔵 ROM/RAM

内蔵周辺モジュール

□ : マッピング不要の場合はノーチェック  
 : 「空間(2)をターゲット側にマッピングする」の指定にする場合、BOOT仕様の品種は、モニタワークエリアを【固定番地方式】にする。

2 0) H8 シミュレーション (例 H8/38124 の場合)

: マッピング不要の場合はノーチェック  
 : 「空間 (2) をターゲット側にマッピングする」の指定にする場合、E8 仕様の品種は、モニタワークエリアを【固定番地方式】にする。

2 1) H8 シミュレーション (例 H8/3029F の場合)

: マッピング不要の場合はノーチェック  
 : I/O のみをエミュレートする場合は、「空間 (2) をターゲット側にマッピングする」にチェックする。

: マッピング不要の場合はノーチェック  
 : I/O のみをエミュレートする場合は、「空間 (M1) をターゲット側にマッピングする」にチェックする。

2 2) H8 シミュレーション (例 H8/3048Fone の場合)

: マッピング不要の場合はノーチェック  
 : I/O のみをエミュレートする場合は、「空間 (2) をターゲット側にマッピングする」にチェックする。

2 3) H8 シミュレーション (例 H8/3052BF の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス 0x0 ~ 0x00FFFF0F 番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス [ ] ~ [ ] 番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス 0x00FFFF20 ~ 0x00FFFFFF 番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU H8/300Hシリーズの場合  
 16Mバイト空間  1Mバイト空間

範囲外アクセスを無視する。

設定

内蔵 ROM/RAM

内蔵周辺モジュール

□ : マッピング不要の場合はノーチェック  
 ☑ : 「空間 (2) をターゲット側にマッピングする」の指定にする場合、  
 BOOT 仕様の品種は、モニタワークエリアを【固定番地方式】にする。

2 4) H8 シミュレーション (例 H8/36064 の場合)

DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス 0x0 ~ 0x00007FFF 番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス 0x0000F700 ~ 0x0000F77F 番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス 0x0000F780 ~ 0x0000FF7F 番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス 0x0000FF80 ~ 0x0000FFFF 番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU H8/300Hシリーズの場合  
 16Mバイト空間  1Mバイト空間

範囲外アクセスを無視する。

設定

内蔵 ROM

内蔵周辺モジュール

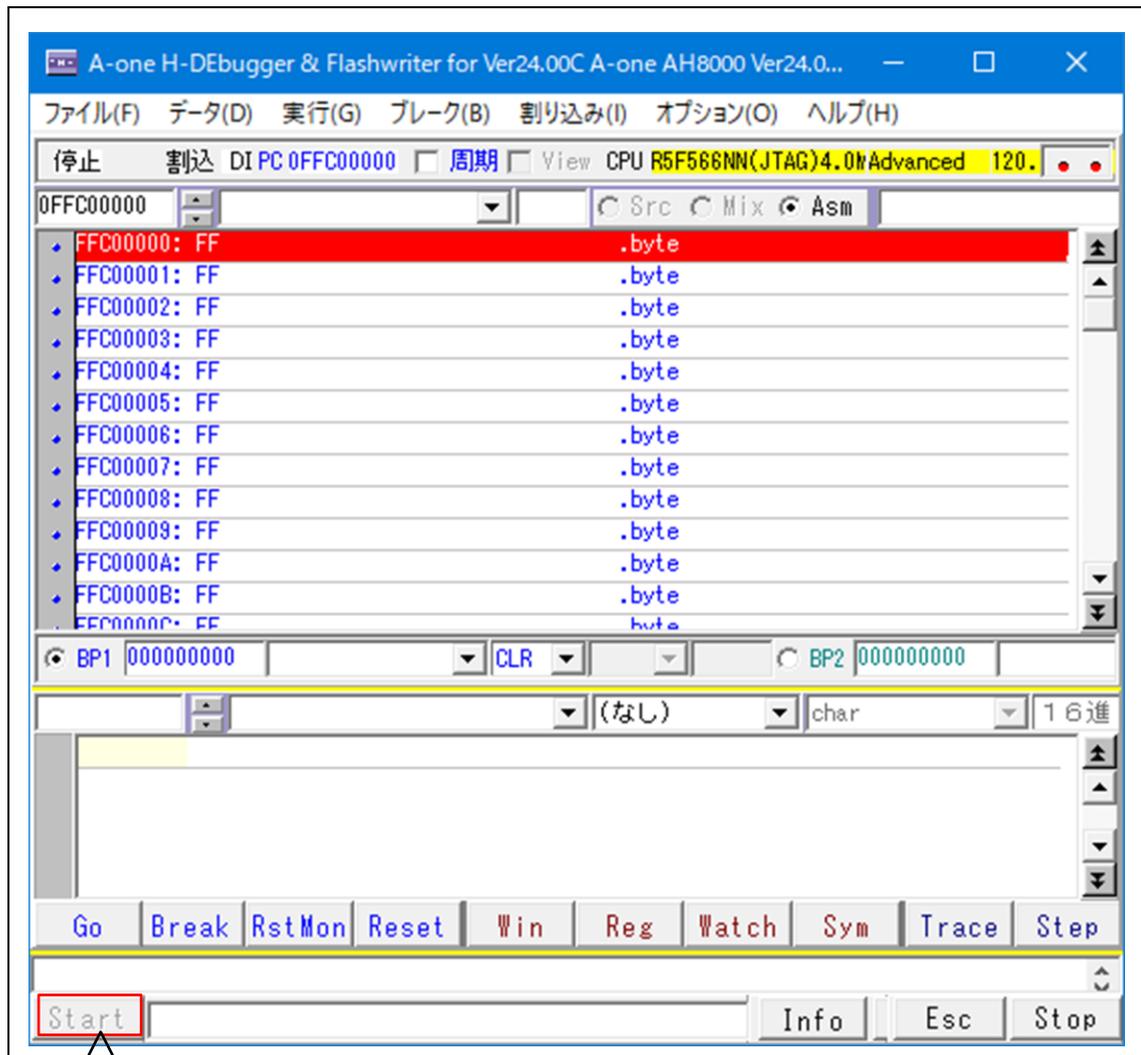
内蔵 RAM

内蔵周辺モジュール

: マッピング不要の場合はノーチェック  
 : 「空間 (2) をターゲット側にマッピングする」の指定にする場合、E8 仕様の品種は、モニタワークエリアを【固定番地方式】にする。

: マッピング不要の場合はノーチェック  
 : 「空間 (M1) をターゲット側にマッピングする」の指定にする場合、E8 仕様の品種は、モニタワークエリアを【固定番地方式】にする。

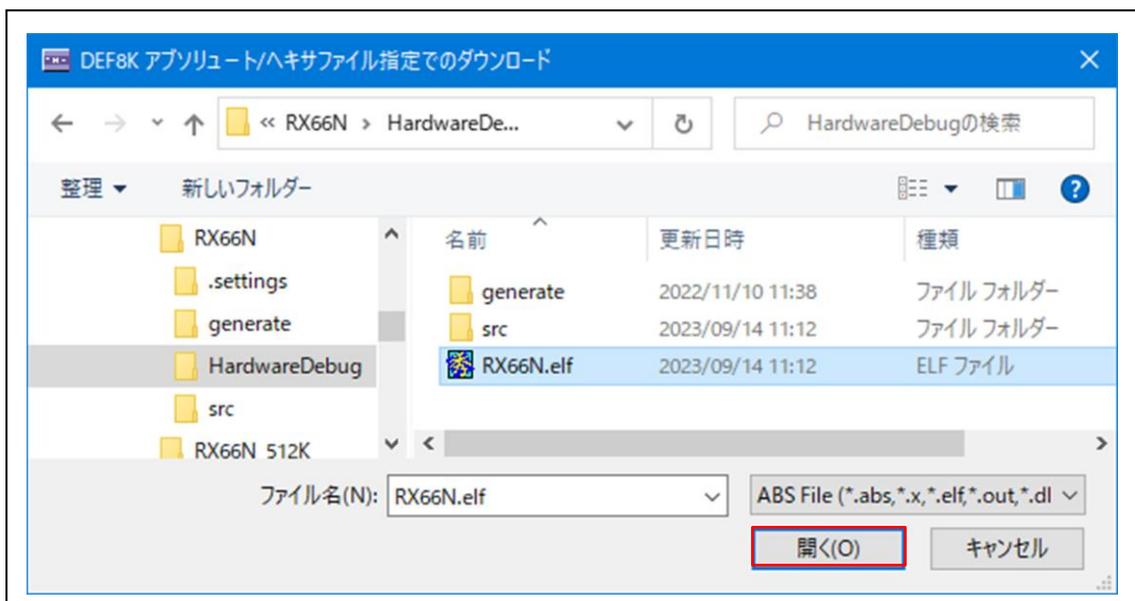
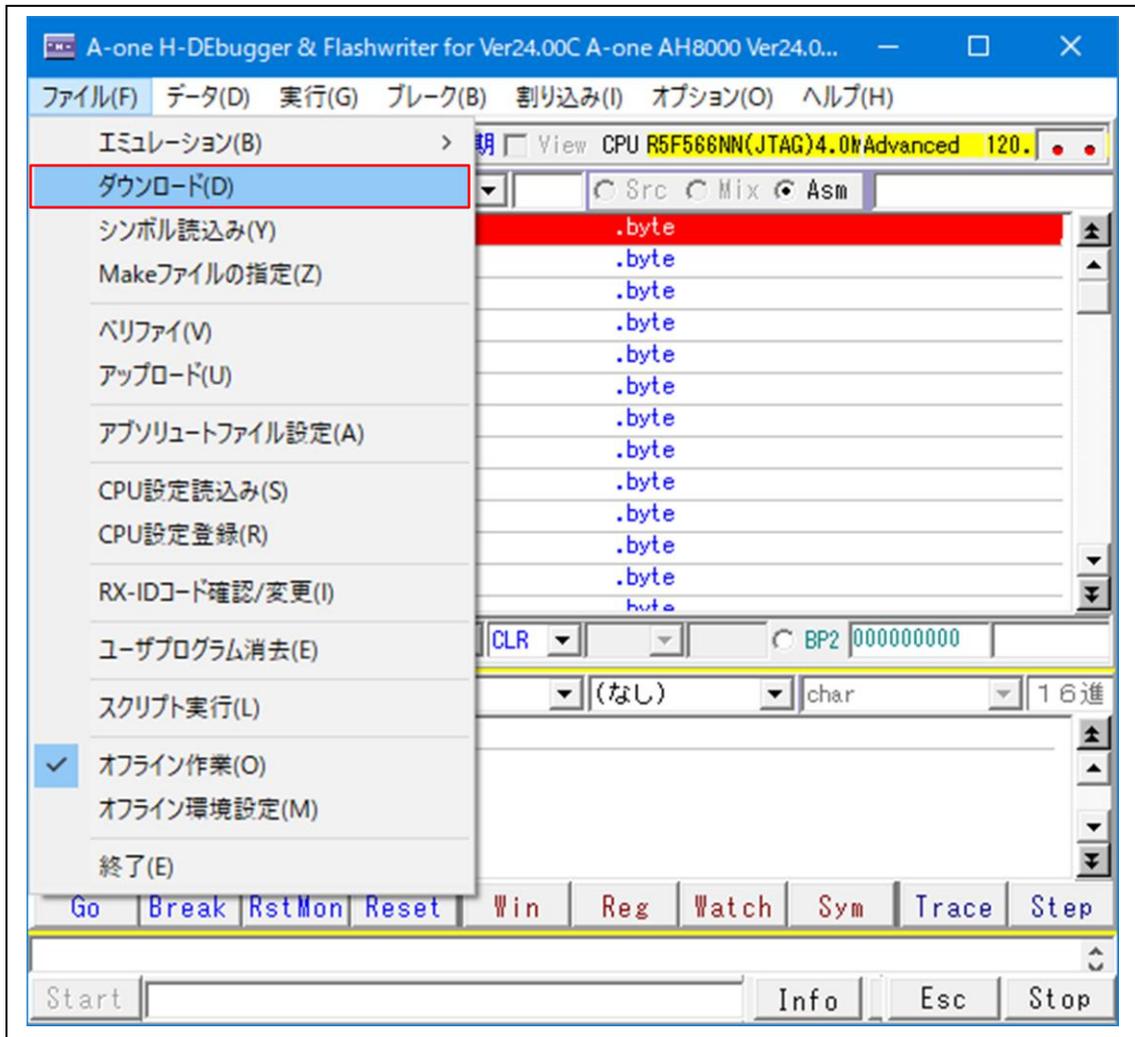
4-3) シミュレーションの開始



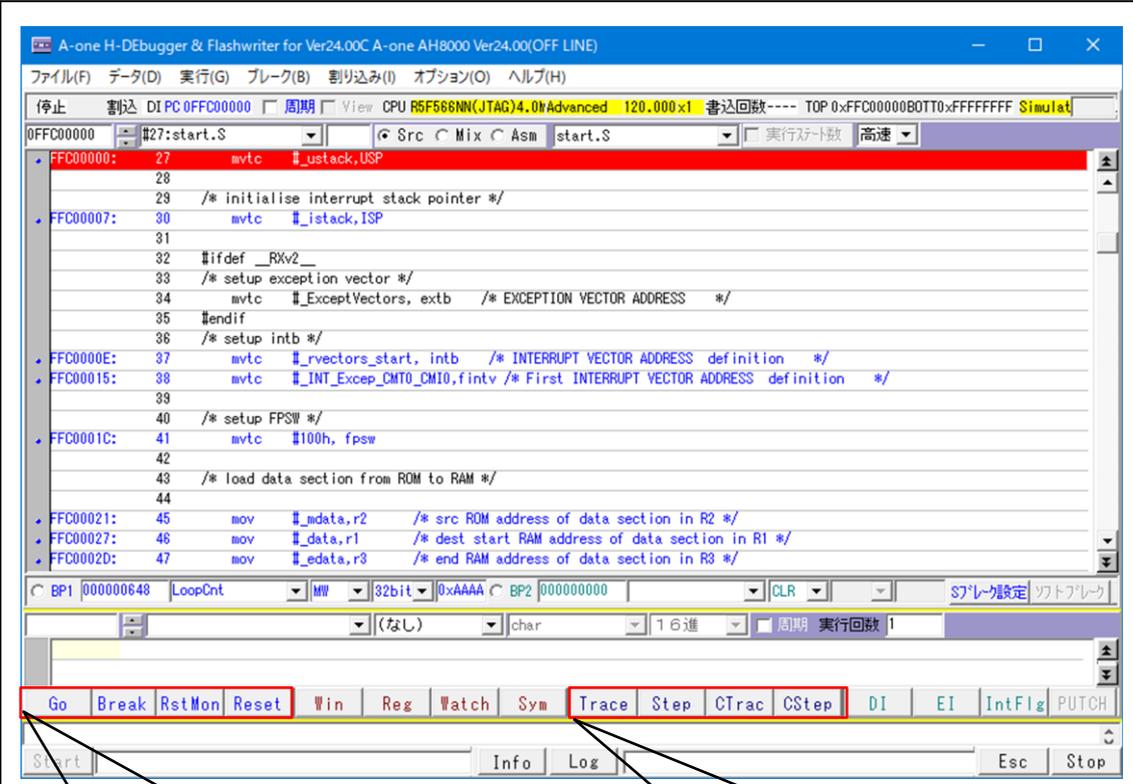
[Start]クリック

4-4) プログラムのダウンロード

<ファイル>-<ダウンロード>



## 4-5) 実行系操作

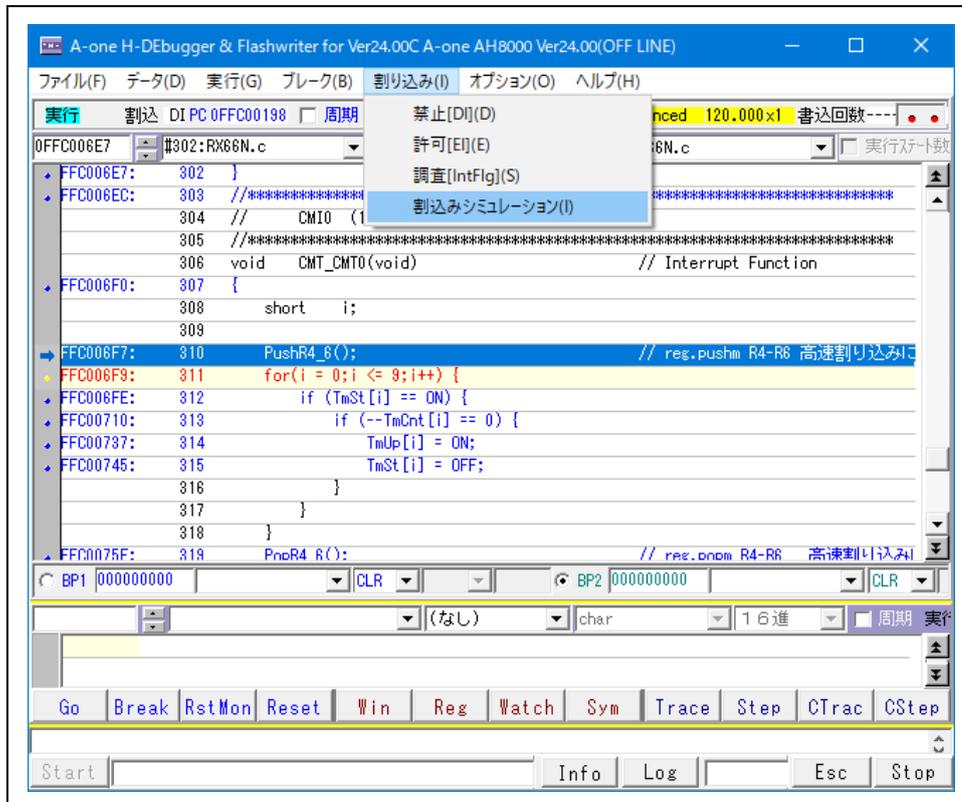


The screenshot shows the A-one H-Debugger & Flashwriter interface. The main window displays assembly code for a start.S file. The control bar at the bottom contains several buttons: Go, Break, RstMon, Reset, Win, Reg, Watch, Sym, Trace, Step, CTrac, CStep, DI, EI, IntFlg, and PUTC. The buttons Go, Break, RstMon, Reset, Trace, Step, CTrac, and CStep are highlighted with red boxes. Arrows from these boxes point to two text boxes below the screenshot.

[Go]	: 実行	[Trace]	: アセンブラレベルのステップイン
[Break]	: 強制ブレーク	[Step]	: アセンブラレベルのステップアウト
[RstMon]	: スタートアップ関数で停止	[CTrac]	: Cソースレベルのステップイン
[Reset]	: スタートアップ関数で停止	[CStep]	: Cソースレベルのステップアウト

#### 4-6) 割り込みシミュレーション

<割り込み>—<割り込みシミュレーション>



##### 1) 使用目的

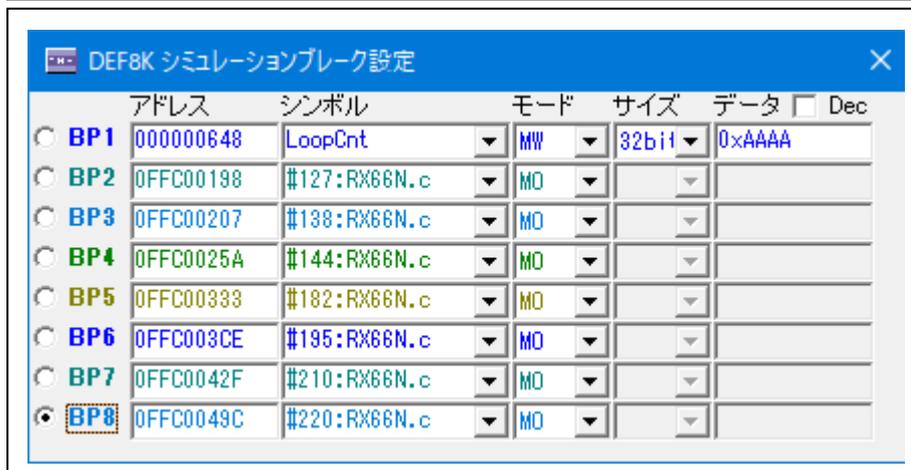
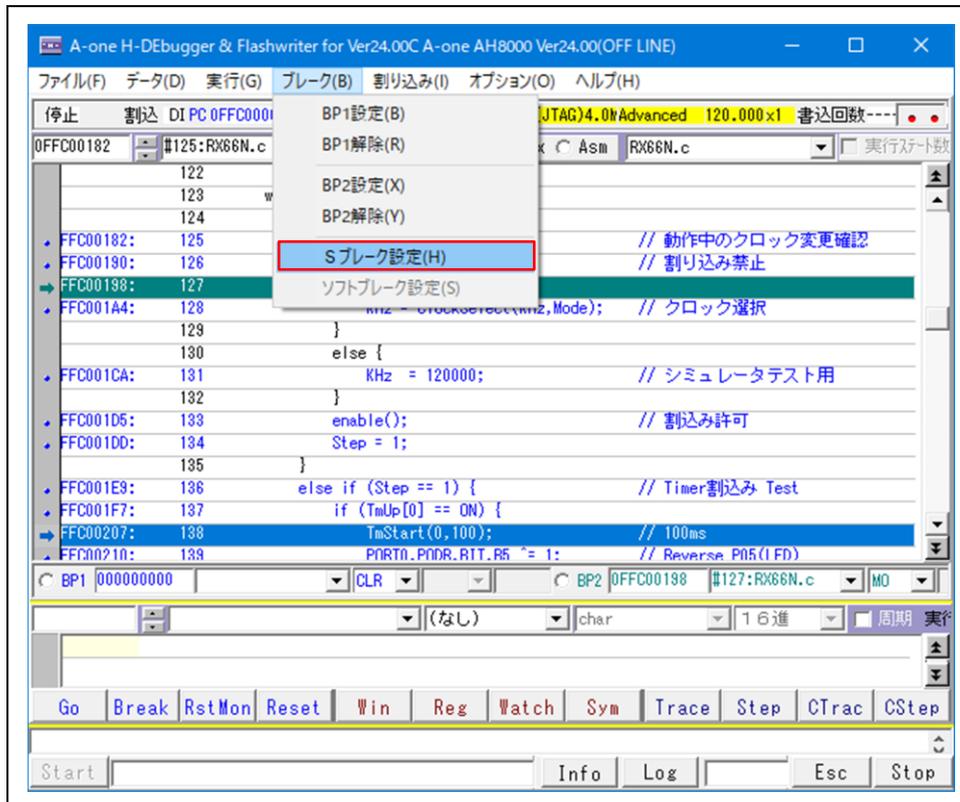
- 割り込み処理の定義が実装されているかを確認するために使用する。

##### 2) 操作概要

- <割り込みメニュー>—<割り込みシミュレーション>にて動作開始する。
- 割り込み要因を選択します。
- [割り込発生] P Bをオンすることにより擬似的に割り込みを発生させます。
- 回数指定しますと連続的に発生します。
- 割り込みのプライオリティ判定をシミュレーションしています。
- 割り込み受付処理が終了するまでは【割り込発生】 P Bをイネーブルにしません。

#### 4-7) ブレーク設定

<ブレーク>—<Sブレーク設定>



##### 1) 使用目的

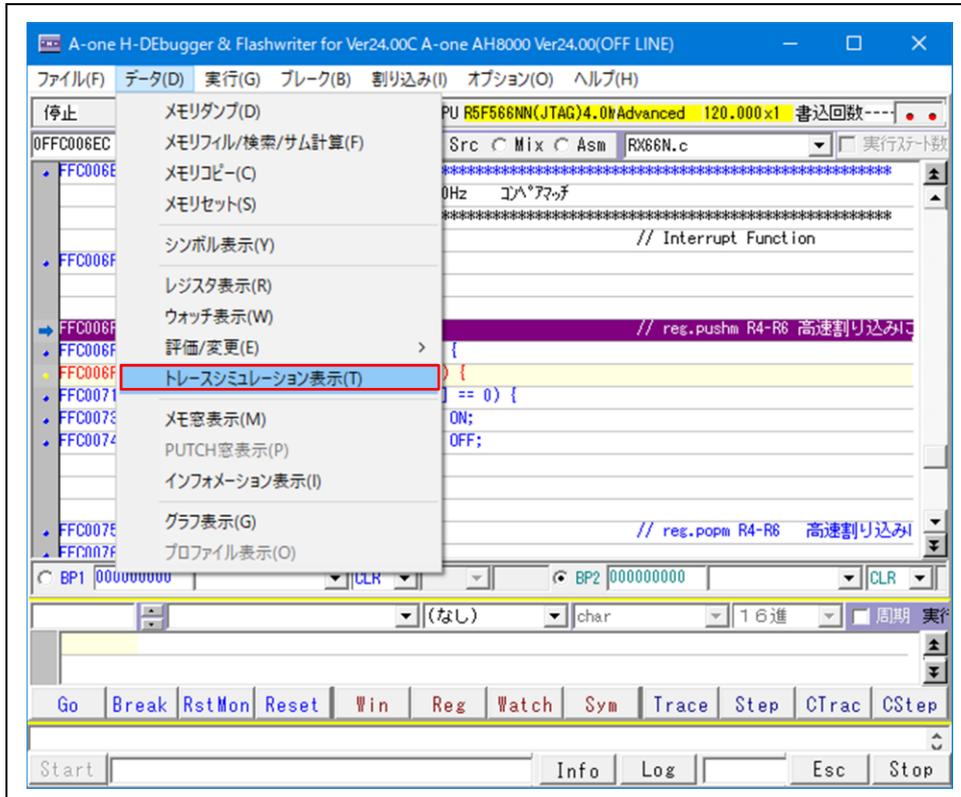
- ・割り込みシミュレーションとの確認のため使用する。
- ・プログラムが予定通りに実行しているかを確認するために使用する。

##### 2) 操作概要

- ・<ブレークメニュー>—<Sブレーク設定>又は[Sブレーク設定]PBにて画面表示。
- ・CソースView画面上で直接ダブルクリックで設定/解除ができる。
- ・ソース画面上で8点までブレーク設定可能です。
- ・データブレークも8点まで設定可能です。

#### 4-8) トレースシミュレーション

<データ>—<トレースシミュレーション表示>



##### 1) 使用目的

・停止位置より4Kステップ分の実行履歴とステップ毎の全レジスタとWatch情報とメモリアクセス履歴(最終8ポイントまで)の確認ができる。

##### 2) 操作概要

- ・<データ>—<トレースシミュレーション表示>にて表示する。
- ・Cを「チェック」するとトレースデータはCソース表示になる。
- ・Reg+Watを「チェック」するとトレース履歴位置の全レジスタ値とWatchデータの確認ができる。
- ・プルアップメニュー選択で検索文字のロック/解除を行いトレース結果の検索ができる。
- ・カーソル行でダブルクリックしますとCView窓にソース表示する。
- ・プルアップメニュー選択で実行履歴情報をファイル保存できる。
  - ①全トレースデータ
  - ②実行履歴
  - ③実行履歴+レジスタ履歴
  - ④実行履歴+Watch情報
  - ⑤実行履歴+メモリアクセス履歴

## 3) 初期画面


 4)  C 画面


5) Reg+Wat 画面

DEF8K トレースシミュレーション表示

```

*B[-22] RX86N.c(328) : FFC0078B: 328 TmSt[tno] = ON;
*B[-18] RX86N.c(329) : FFC00799: 329 }
*B[-16] RX86N.c(118) : FFC0015E: 118 Mode = 2;
*B[-14] RX86N.c(119) : FFC00167: 119 Step = 0;
*B[-12] RX86N.c(120) : FFC00170: 120 LoopCnt = 0;
*B[-10] RX86N.c(121) : FFC00179: 121 LoopCntW = 0;
*B[-8] RX86N.c(125) : FFC00182: 125 if (Step == 0) {
*B[-3] RX86N.c(126) : FFC00190: 126 disable();
*B[0] RX86N.c(127) : FFC00198: 127 if (Emulator == 1) {
    
```

ブレーク履歴  Reg+Wat  C  ↑検索 ↓検索

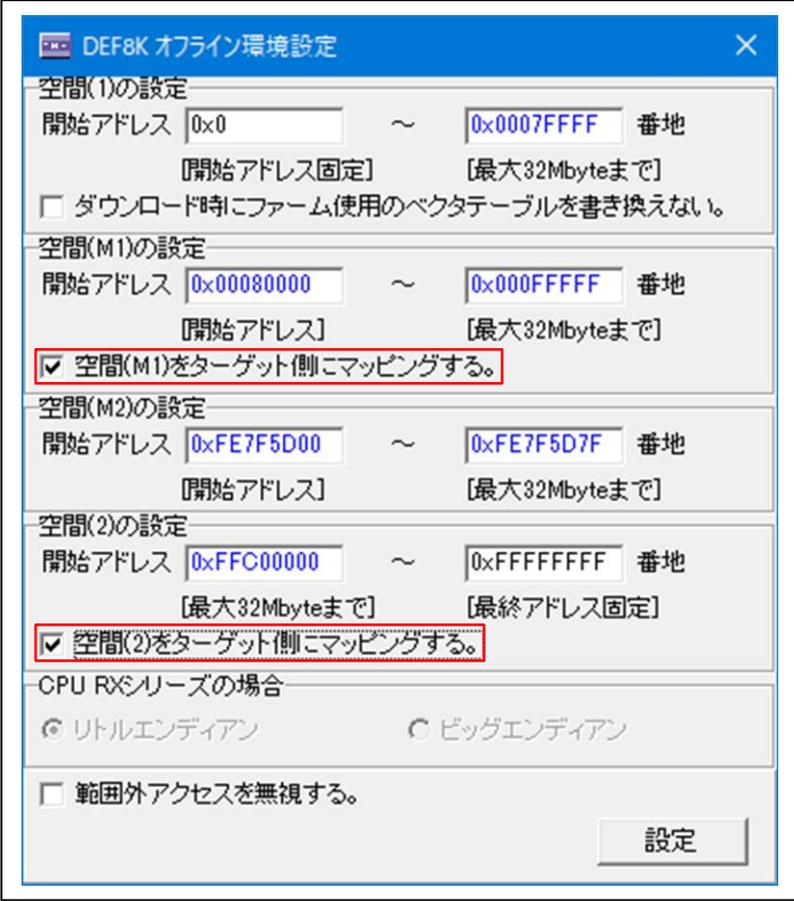
DEF8K Watch

実行	Type	更新	Data	<input type="checkbox"/> 周期サンプル
<input type="radio"/> LoopCnt	long	00000648:	Hex=00000000 Dec=000000000	
<input checked="" type="radio"/>	char			
<input type="radio"/>	char			
<input type="radio"/>	char			

DEF8K レジスタ RX

PSW	000120003	P_U_ZC	
PC	0FFC00198	SP	0000001F0
BPC	000000000	ISP	000000100
R1	000120003	USP	0000001F0
R2	000000064	R9	000000000
R3	000000064	R10	0000001F0
R4	000000674	R11	000000000
R5	000000000	R12	000000000
R6	000000000	R13	000000000
R7	0FFC000B0	R14	000000000
R8	000000000	R15	000000000
BPSW	000000000	INTB	0FFC010E0
FINT	0FFC008B3	FPSW	000000100
ACCH	000000000	ACCL	000000000
ACCG	000000000	AC1G	000000000
AC1H	000000000	AC1L	000000000
EXTB	0FFFFFFF80	FPU	

## 4-9) 空間 (M1) と空間 (2) をターゲット側にマッピングする場合の注意事項



DEF8K オフライン環境設定

空間(1)の設定  
 開始アドレス  ~  番地  
 [開始アドレス固定] [最大32Mbyteまで]  
 ダウンロード時にファーム使用のベクタテーブルを書き換ええない。

空間(M1)の設定  
 開始アドレス  ~  番地  
 [開始アドレス] [最大32Mbyteまで]  
 空間(M1)をターゲット側にマッピングする。

空間(M2)の設定  
 開始アドレス  ~  番地  
 [開始アドレス] [最大32Mbyteまで]

空間(2)の設定  
 開始アドレス  ~  番地  
 [最大32Mbyteまで] [最終アドレス固定]  
 空間(2)をターゲット側にマッピングする。

CPU RXシリーズの場合  
 リトルエンディアン  ビッグエンディアン

範囲外アクセスを無視する。

設定

1) JTAG/FINE/H-UDI 品種以外でターゲット側にマッピング指定をする場合は、ターゲット基板に「固定番地方式」のモニタ (ファーム) を実装する。

2) JTAG/FINE/H-UDI 品種の場合は、エミュレーション遷移が可能な状態で使用する。

---

---

## 5. 注意事項

- ・本文書の著作権は、エーワン（株）が保有します。
- ・本文書を無断での転載は一切禁止します。
- ・本文書に記載されている内容についての質問やサポートはお受けすることが出来ません。
- ・本文章に関して、ルネサス エレクトロニクス社への問い合わせは御遠慮願います。
- ・本文書の内容に従い、使用した結果、損害が発生しても、弊社では一切の責任を負わないものとします。
- ・本文書の内容に関して、万全を期して作成しましたが、ご不審な点、誤りなどの点がありましたら弊社までご連絡くだされば幸いです。
- ・本文書の内容は、予告なしに変更されることがあります。

〒486-0852  
愛知県春日井市下市場町 6-9-20  
エーワン株式会社  
<https://www.aone.co.jp>

