

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日

ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無いことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

M16C/60 シリーズ用 PC7501 エミュレータ

C0 カバレッジ機能

要旨

M16C/60 シリーズ用のフルスペックエミュレータ PC7501 は、C0 カバレッジ機能を搭載しています。本書では、PC7501 エミュレータを用いて C0 カバレッジ機能の使用方法について説明します。

本書の内容は、M16C/60 シリーズマイコンを搭載したユーザシステムと PC7501 エミュレータを組み合わせ使用する場合を対象としています。異なるターゲットマイコンであっても M16C/60 シリーズマイコンであれば共通で利用することが可能です。

目次

1. 概要	2
2. 使用機能説明	2
3. ソフトウェアの準備	2
3.1 はじめに	2
3.2 PC7501 エミュレータ付属品ソフトウェアのインストール	2
3.3 オートアップデートユーティリティ	2
4. 動作説明	3
4.1 High-performance Embedded Workshopの起動	3
4.2 ワークスペースを開く	4
4.3 ターゲットプログラムのダウンロード	8
4.4 C0 カバレッジの計測	9
5. よくある質問	18
5.1 ソース単位とカバレッジウィンドウで結果が異なるのは何故ですか？	18
6. 関連ドキュメント	20

1. 概要

カバレッジ機能は、テストの網羅度を計測するための機能です。テストプログラムを実行した際のカバレッジ率を測定することで、テストプログラムの妥当性を判断するために使用されます。カバレッジ機能にはC0カバレッジやC1カバレッジなどの種類があり、PC7501ではC0カバレッジを搭載しています。C0カバレッジは命令網羅度とも呼ばれ命令が実行されたかどうかを測定することが可能です。また、C1カバレッジは分岐網羅度とも呼ばれ、条件分岐命令の真／偽のどちらの分岐をしたかどうかの網羅度を測定することが可能です。

PC7501では、測定したC0カバレッジの結果をC0カバレッジウィンドウに表示するだけでなく、ソース行単位の実行/未実行をエディタウィンドウのカラムを利用して確認したり、アドレス単位の実行/未実行をメモリウィンドウ上で確認することが可能です。カバレッジ計測領域は、256Kバイト境界から始まる任意の1～32ブロック(最大8Mバイト)です。また、PC7501のC0カバレッジ機能では、プリフェッチも計測結果に含まれることに注意してください。

2. 使用機能説明

本書では、PC7501 エミュレータ付属の CD-ROM、またはルネサス Web サイトからのダウンロードパッケージに含まれるサンプルプログラムを使用して C0 カバレッジ計測方法を説明します。

使用するツールのバージョンは次のとおりです。

M16C シリーズ用 M16C R8C PC7501 エミュレータデバッグ V.1.03 Release 00

3. ソフトウェアの準備

3.1 はじめに

PC7501 エミュレータに付属する CD-ROM のソフトウェアをインストールします。

これにより本書で使用するサンプルプログラム(tutorial ワークスペース)がパソコン上に展開されます。

High-performance Embedded Workshop をインストール済みのパソコン上に PC7501 エミュレータに付属する CD-ROM のソフトウェアをインストールすることは可能です。この場合、インストール作業中に一部のダイアログ表示が省略されることがあります。

3.2 PC7501 エミュレータ付属品ソフトウェアのインストール

PC7501 エミュレータに付属する CD-ROM 内の M16cPc7501 Debugger.exe を実行してください。

インストールの詳細につきましては、ルネサス Web サイトに掲載の PC7501 エミュレータ用セットアップガイドを参照していただき、インストール作業中は画面の指示に従って操作をしてください。ここではインストール手順は省略します。

3.3 オートアップデートユーティリティ

ソフトウェアのインストールの際に、オートアップデートユーティリティを選択した場合、インターネット経由で各ツールの最新版の存在を確認することが可能になります。

4. 動作説明

本章では、High-performance Embedded Workshop (HEW) を起動して、C0 カバレッジ機能の使用方法について説明します。手順としては以下のようになります。

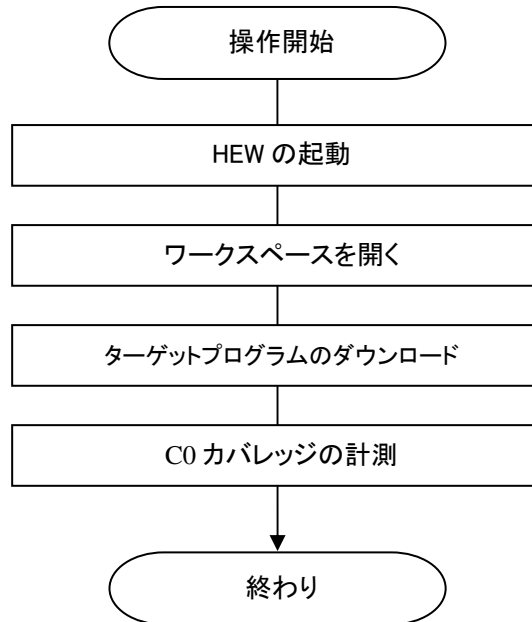


図 4.1 サンプルプログラムの実行手順

4.1 High-performance Embedded Workshop の起動

まず、始めにユーザシステムを接続したPC7501エミュレータとホストコンピュータをUSBケーブルで接続し、デバッグ操作が可能であることを確認してください。

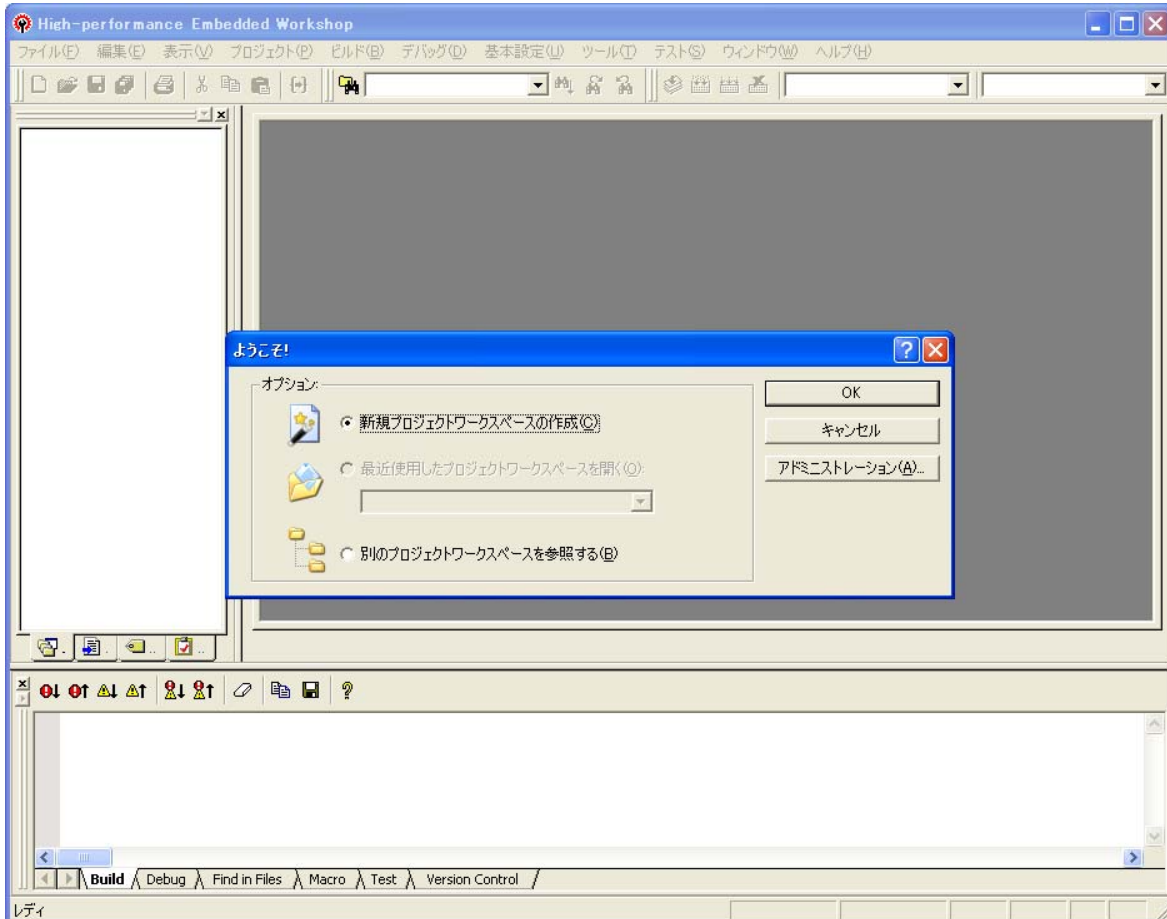
次にHigh-performance Embedded Workshopを起動します。

[スタート]メニューの[すべてのプログラム]から[Renesas]→[High-performance Embedded Workshop]→[High-performance Embedded Workshop]で起動できます。

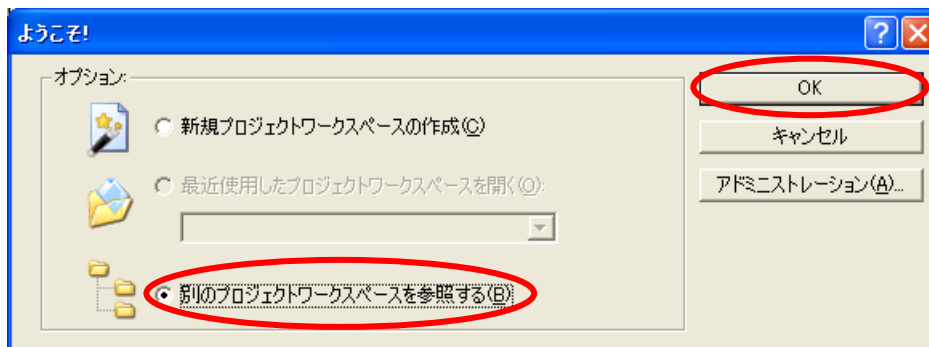


4.2 ワークスペースを開く

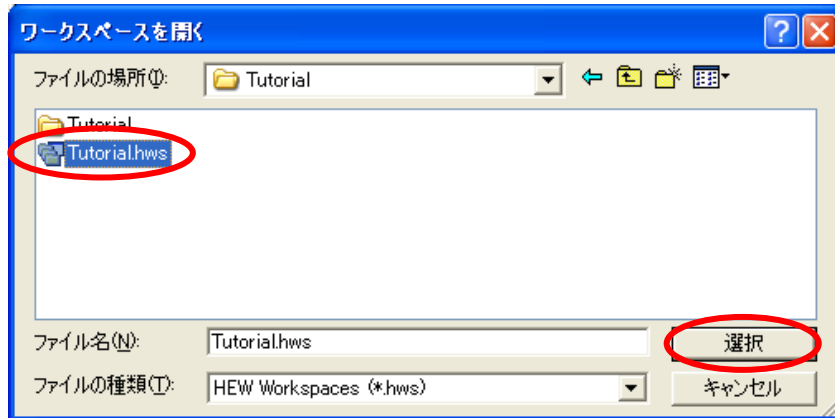
(1) High-performance Embedded Workshop 上に[ようこそ!]ダイアログボックスが表示されます。



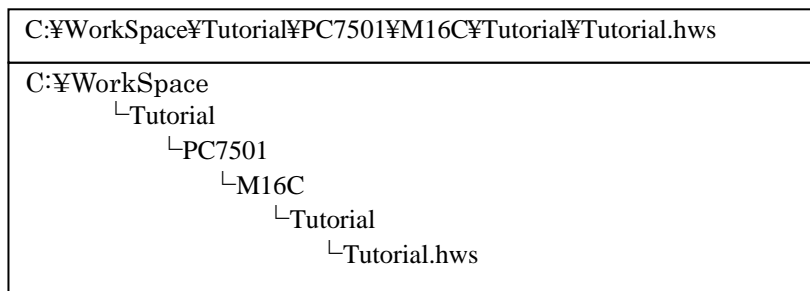
[ようこそ!]ダイアログボックス内の[別のプロジェクトワークスペースを参照する]ラジオボタンを選択して[OK]ボタンを押してください。



(2) [ワークスペースを開く]ダイアログボックスが表示されます。



本製品の CD-ROM のインストールが完了している場合、標準では次のフォルダ位置にワークスペース "Tutorial.hws" が格納されています。フォルダ位置を確認しながら順番に指定してください。ワークスペース "Tutorial.hws" が見つかりましたら指定し[選択]ボタンを押してください。



【注】 ソフトウェアのバージョンによっては、上記ディレクトリを指定できない場合があります。その場合は以下のディレクトリを指定してください。

<High-performance Embedded Workshop インストール先ディレクトリ>

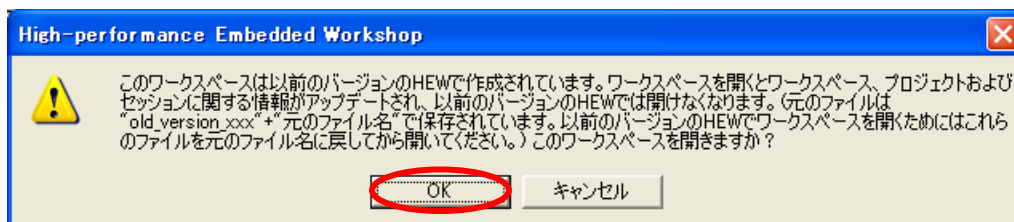
¥Tools¥Renesas¥DebugComp¥Platform¥E8¥M16C¥Tutorial

ディレクトリ例:

C:\¥hew3¥Tools¥Renesas¥DebugComp¥Platform¥E8¥M16C¥Tutorial

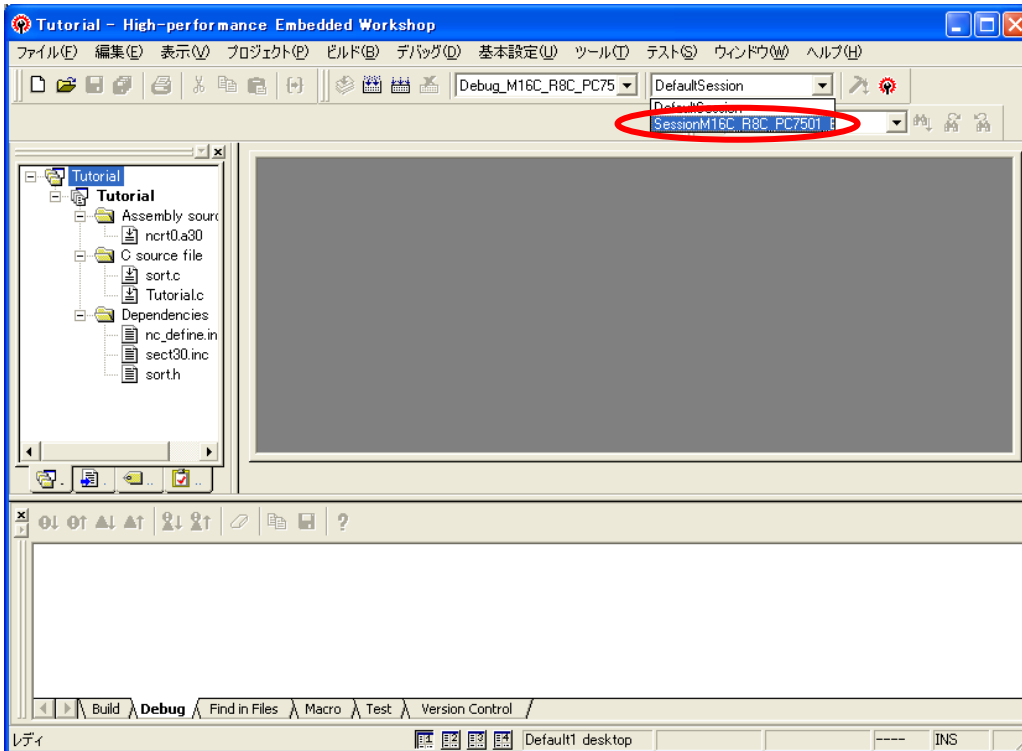
C:\¥hew2¥Tools¥Renesas¥DebugComp¥Platform¥E8¥M16C¥Tutorial

(3) ワークスペースのバージョンが古い場合は次のダイアログボックスが表示されます。新しいバージョンにアップデートするために[OK]ボタンを押してください。

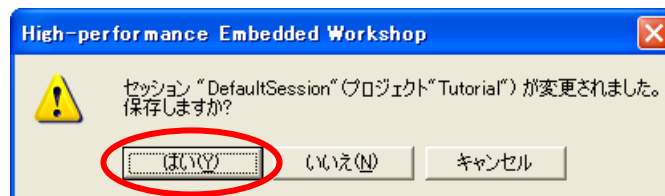


- (4) ワークスペースが開かれると、High-performance Embedded Workshop の操作が可能になります。セッションを切り替えて PC7501 エミュレータを接続します。ツールバー上の [DefaultSession] を [SessionM16C_R8C_PC7501_Emulator] に切り替えてください。

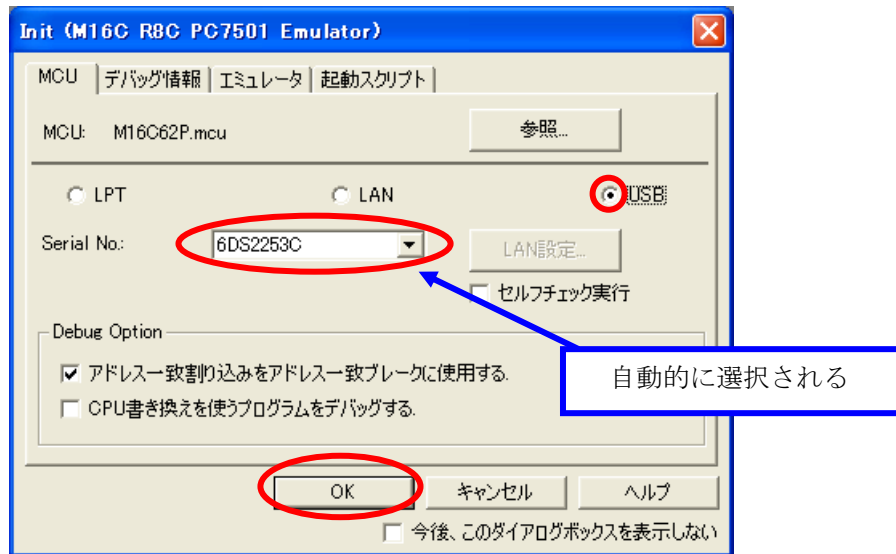
セッションを切り替える前に、あらかじめ PC7501 エミュレータ本体の後面パネルのインターフェイス選択スイッチを USB 側に切り替えて、電源を ON にしてください。



以前のセッションの保存を確認するダイアログボックスが表示された場合、[はい]を指定してください。

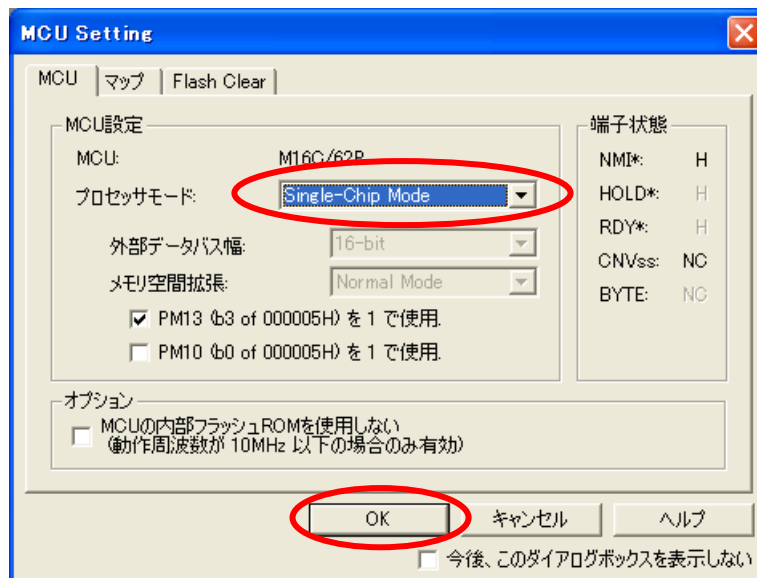


- (5) [Init(M16C R8C PC7501 Emulator)]ダイアログボックスが表示されるので、[MCU]タブ内の[USB]のラジオボタンをチェックして、[OK]を押してください。

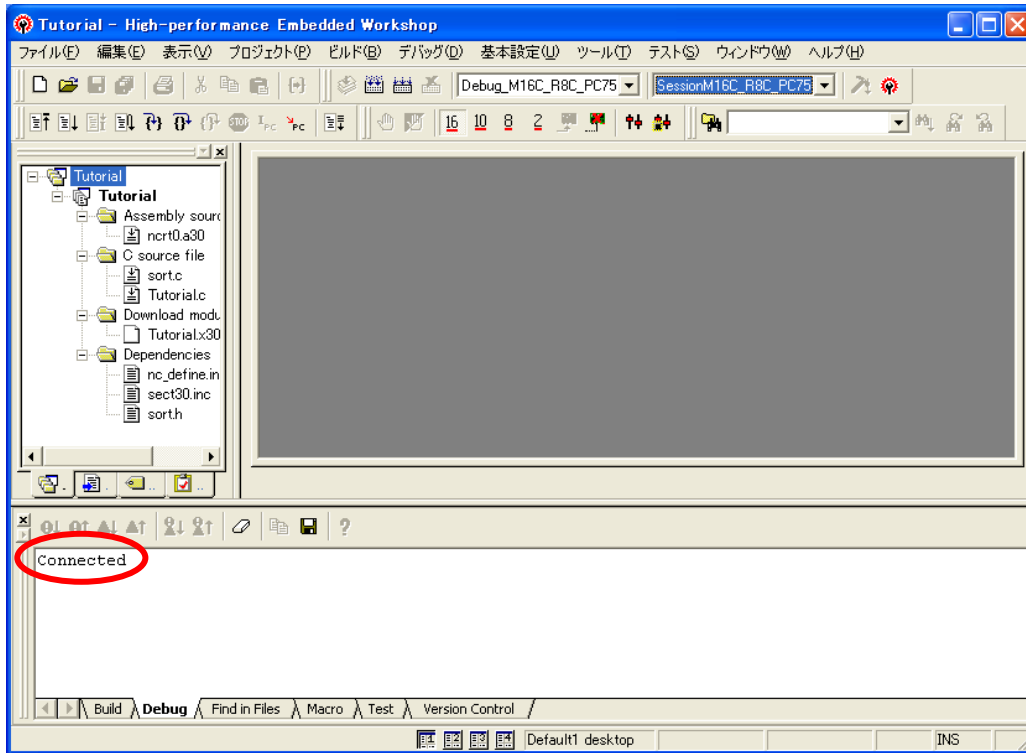


・USB ではなく LPT、LAN で接続する場合は、ルネサス Web サイトにある PC7501 のユーザーズマニュアルをご参照ください。

- (6) [MCU Setting]ダイアログボックスが表示されるので、[MCU]タブ内の[プロセッサモード]を“Single-Chip Mode”にして[OK]を押してください。



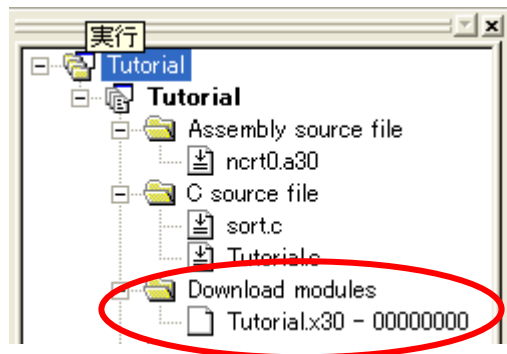
- (7) PC7501 エミュレータの接続が完了して High-performance Embedded Workshop の画面が操作可能になります。接続が完了するとアウトプットウィンドウの[Debug]タブ上に[Connected]と表示されます。



4.3 ターゲットプログラムのダウンロード

最初に、ターゲットプログラムのダウンロード・実行を行います。

- (1) 今回は[ワークスペース]上の[Download modules]内の[Tutorial.x30 - 00000000]をダブルクリックしてダウンロードを行ってください。



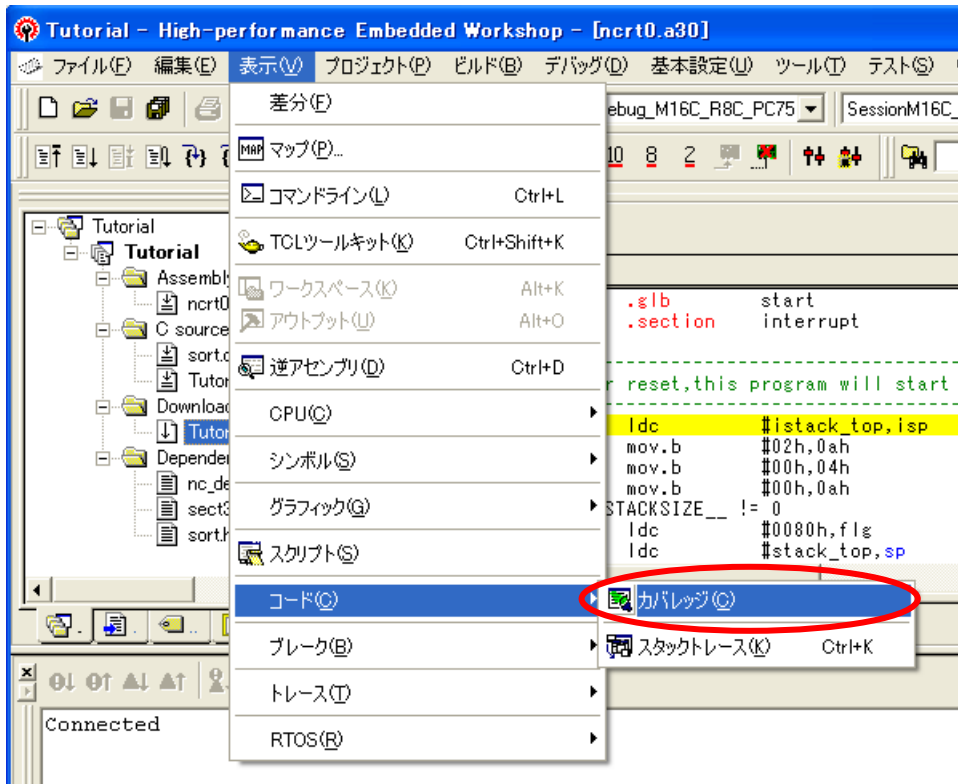
プログラムのダウンロードが終了すると、アイコンに下矢印が付きます。



4.4 C0 カバレッジの計測

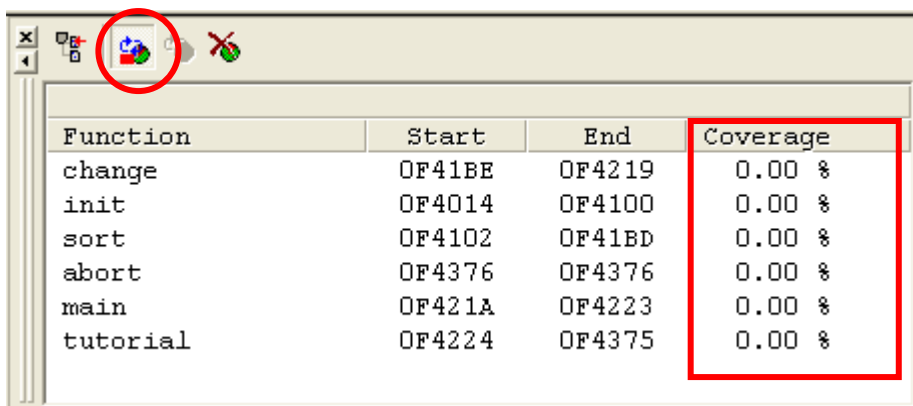
カバレッジの計測結果を見るために、カバレッジウィンドウを開きます。

- (1) [表示]メニューの[コード]の中の[カバレッジ]を選択してください。



- (2) カバレッジウィンドウが開きます。ウィンドウ上にはターゲットプログラムの各関数名([Function])、開始アドレス([Start])、終了アドレス([End])、C0 カバレッジ計測結果([Coverage])が表示されます。この時点では、プログラムを実行していないので[Coverage]の値は“0.00 %”になっています。

本書の説明では、[表示自動更新]アイコンは“有効”にしておきます(有効の場合はアイコンがくぼんでいます)。

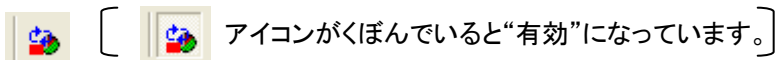
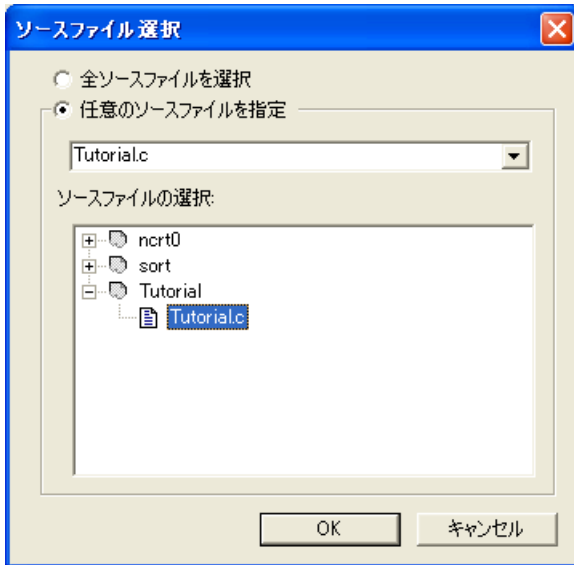


・アイコンの説明



ソース選択 : カバレッジ計測結果を表示するソースファイルを選択します。

[ソース選択]アイコンを選択すると[ソースファイル選択]ダイアログボックスが表示され、任意のソースファイルのみをカバレッジウィンドウに表示させることができます。



表示自動更新 : カバレッジ計測結果の表示をターゲットプログラム停止時に自動更新します。

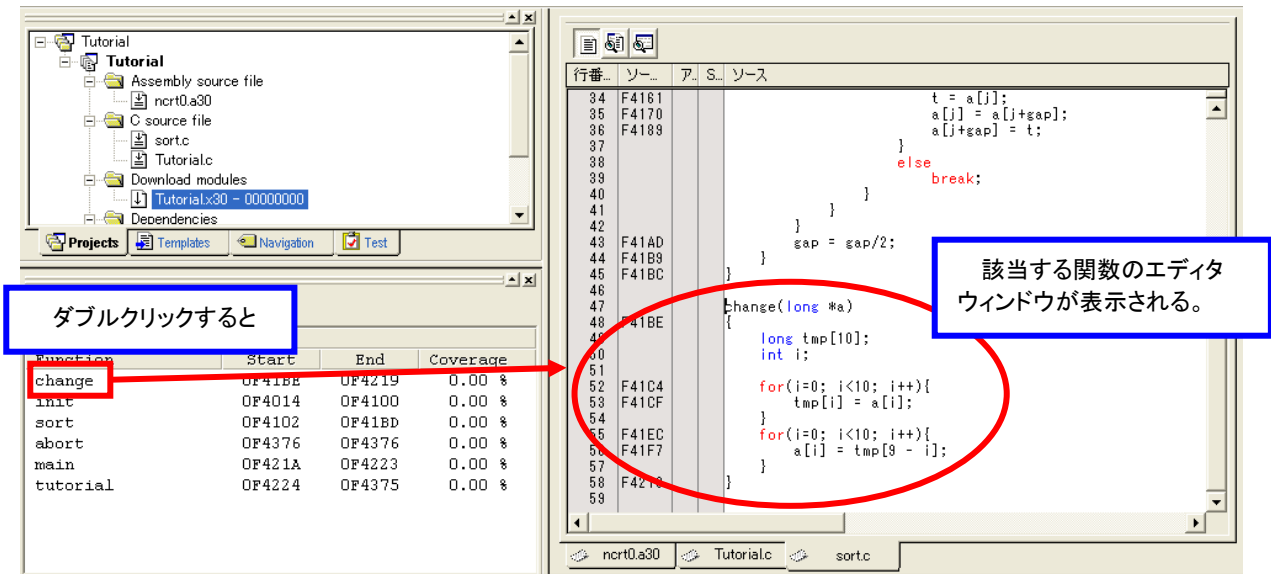


表示更新 : カバレッジ計測結果の表示を更新します。

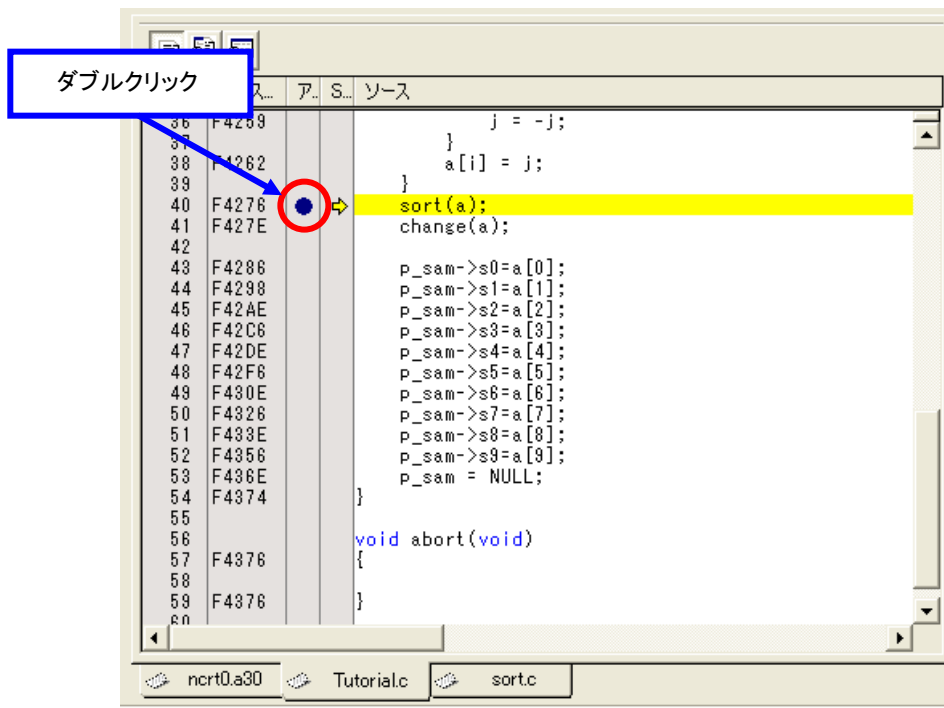


初期化 : カバレッジ計測結果を初期化します。

- (3) カバレッジウィンドウ内に表示されている関数の任意の行をダブルクリックすると、該当する関数のエディタウィンドウが表示されます。

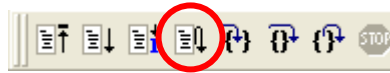


- (4) それでは、関数 sort の呼出し直前でプログラムを停止させ、どのように C0 カバレッジが計測されるかを確認してみましょう。
ソースファイル“Tutorial.c”の行番号 40 の [アドレス一致ブレイクポイント]カラムをダブルクリックしてください。

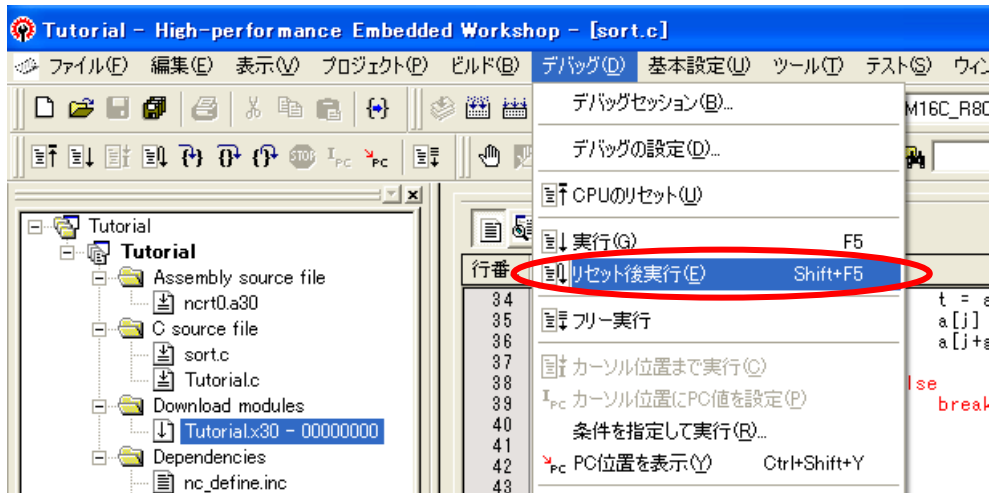


関数 sort のソースアドレス“F4276”の実行直前でブレークがかかるように設定したので、その先にある関数 change、abort は実行されません。

- (5) ターゲットプログラムを実行して C0 カバレッジを計測します。[リセット後実行]アイコンを選択してプログラムを実行してください。



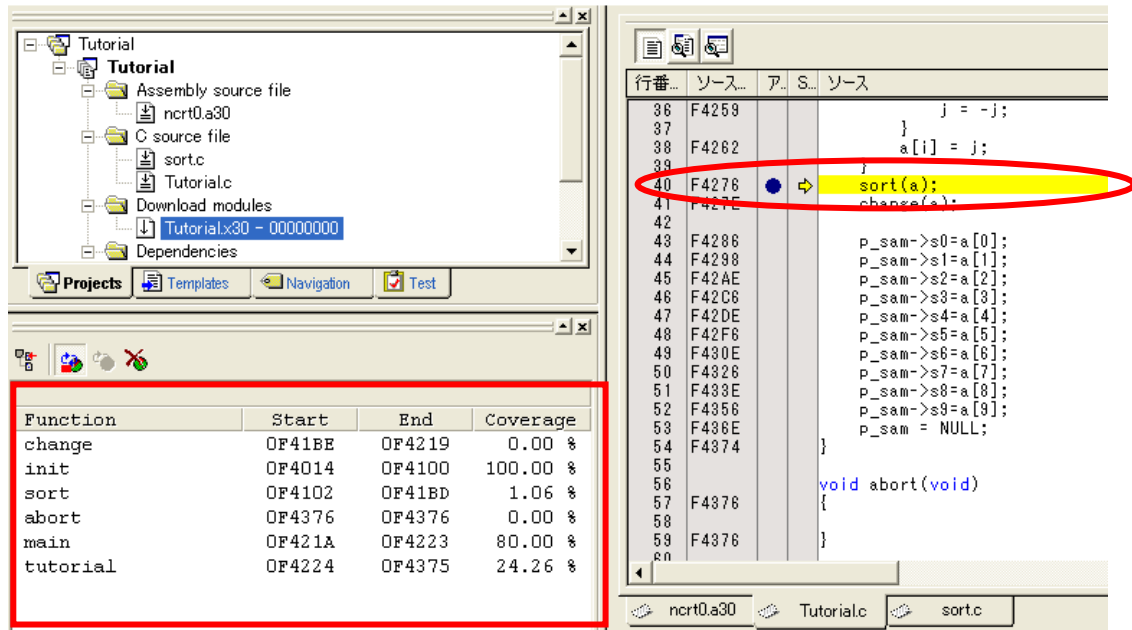
[デバッグ]メニューからも実行できます。



- (6) C0 カバレッジ計測中は、カバレッジウィンドウの[Coverage]の値は“- %”と表示されます。

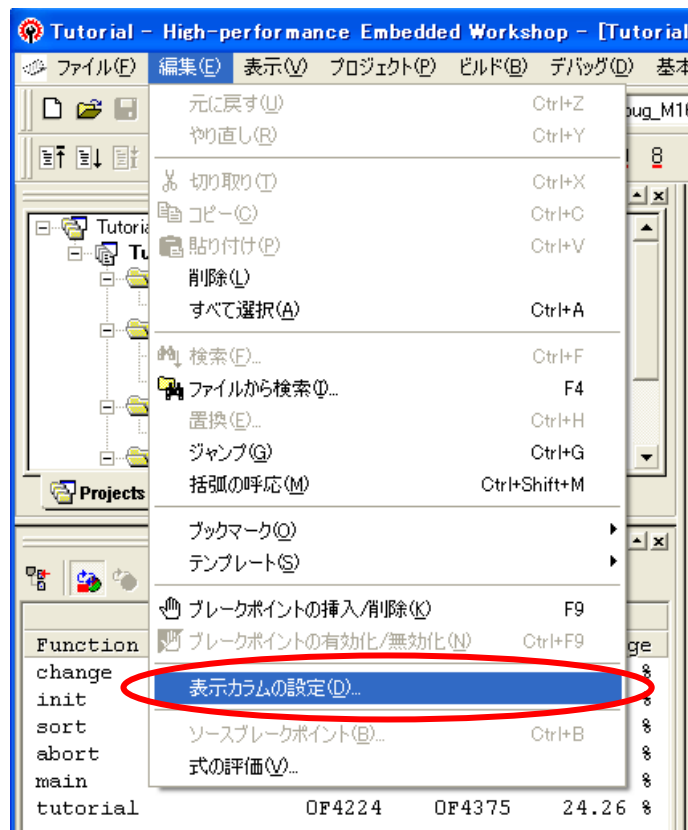
Function	Start	End	Coverage
change	0F41BE	0F4219	- %
init	0F4014	0F4100	- %
sort	0F4102	0F41BD	- %
abort	0F4376	0F4376	- %
main	0F421A	0F4223	- %
tutorial	0F4224	0F4375	- %

- (7) ブレークポイントでプログラムが停止し、カバレッジウィンドウに C0 カバレッジ計測結果が表示されます。

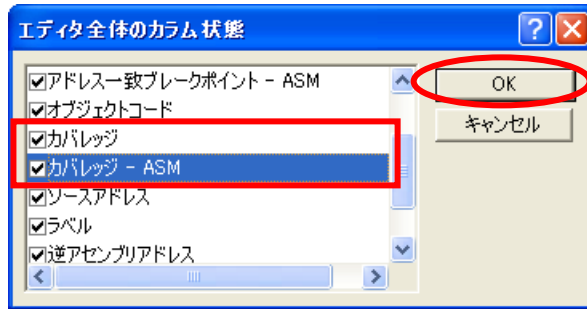


この場合、関数 sort 以降に呼び出される関数 change および割り込み用途の関数 abort は未実行のため“0.00 %”になっていることが確認できます。

- (8) C0 カバレッジの結果をエディタウィンドウ上で確認することもできます。[編集]メニューの[表示カラムの設定]を選択してください。



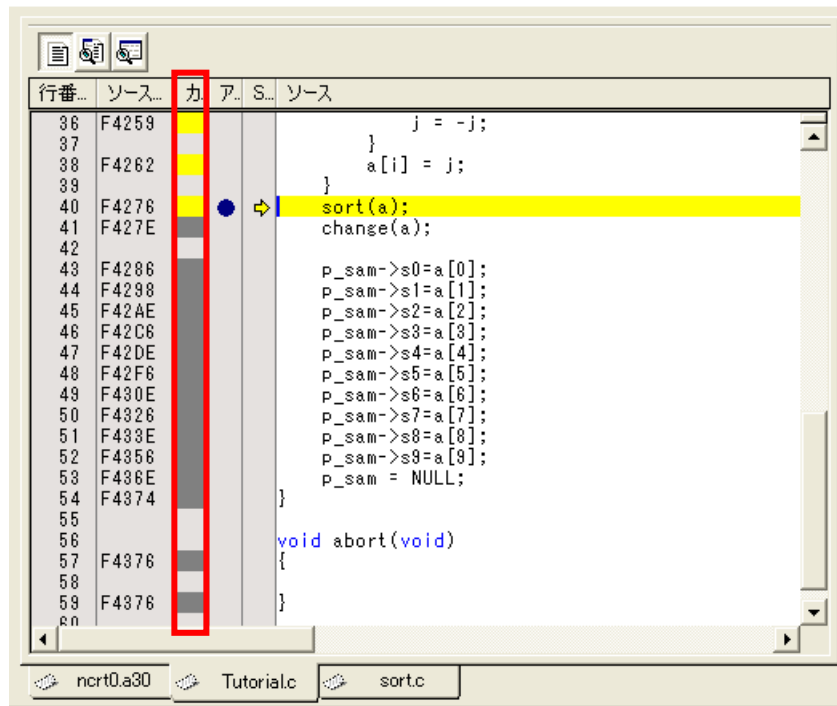
- (9) [エディタ全体のカラム状態]ダイアログボックスが表示されるので、[カバレッジ]と[カバレッジ-ASM]にチェックを入れて[OK]を押してください。



[カバレッジ] : C ソース単位で C0 カバレッジを確認できる。

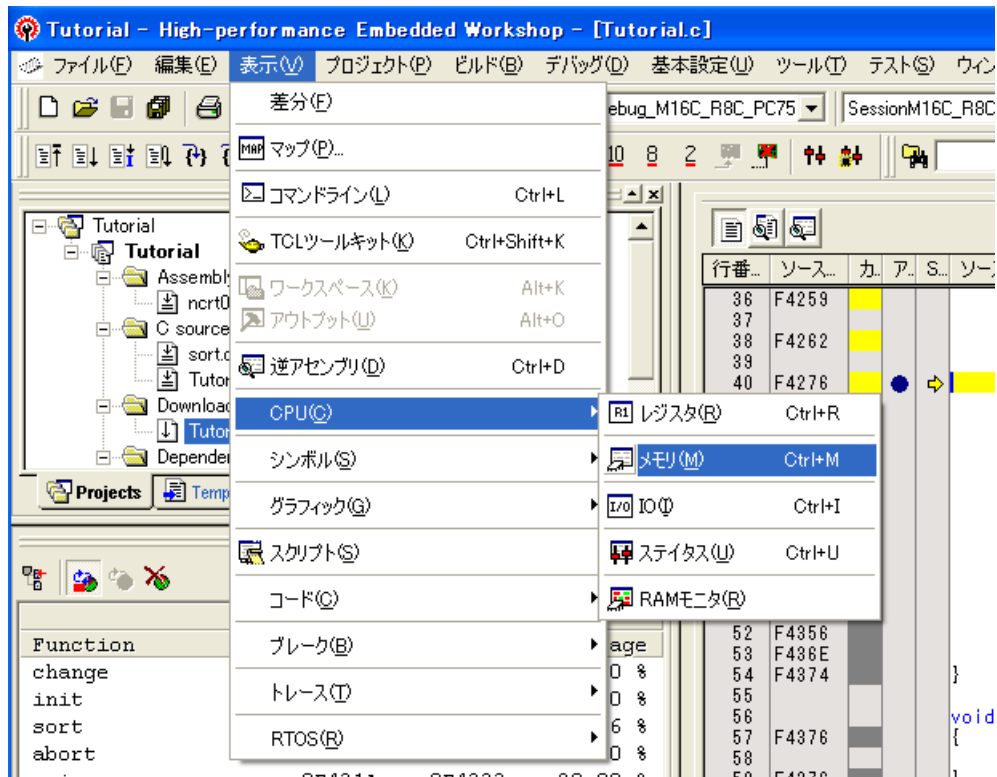
[カバレッジ-ASM] : アセンブリ単位で C0 カバレッジを確認できる。

- (10) エディタウィンドウに[カバレッジ]の行が追加され、C0 カバレッジを視覚的に確認できます。黄色の部分が実行された命令で、灰色の部分が未実行の命令を表しています。

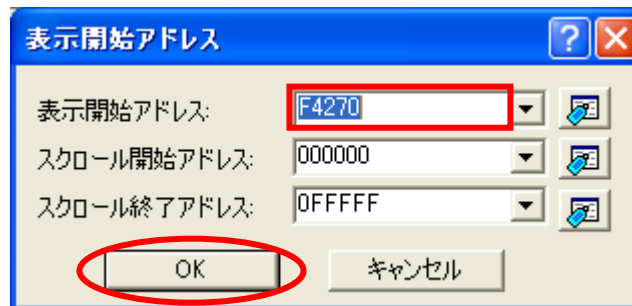


混合モード、逆アセンブリモード表示でも同じように C0 カバレッジを視覚的に確認することができます。

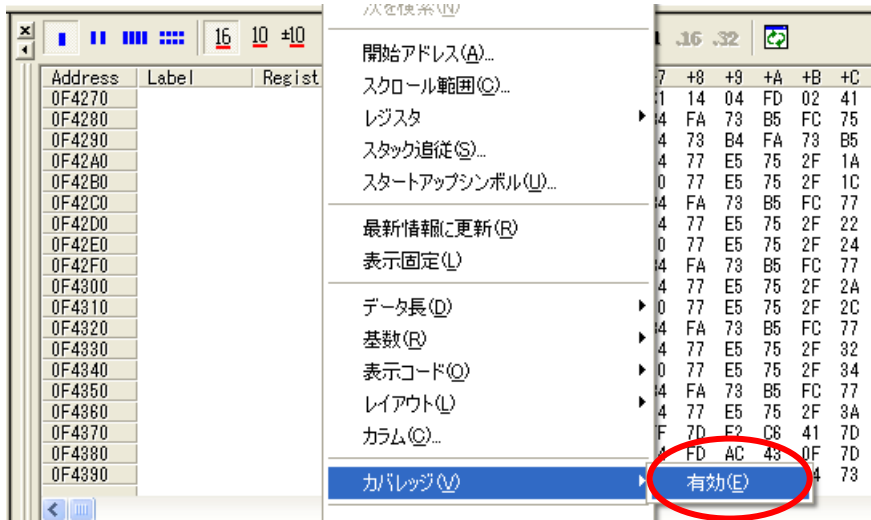
- (11) また、メモリウィンドウを使ってアドレス単位で C0 カバレッジを確認することもできます。[表示]メニューの[CPU]の中の[メモリ]を選択してください。



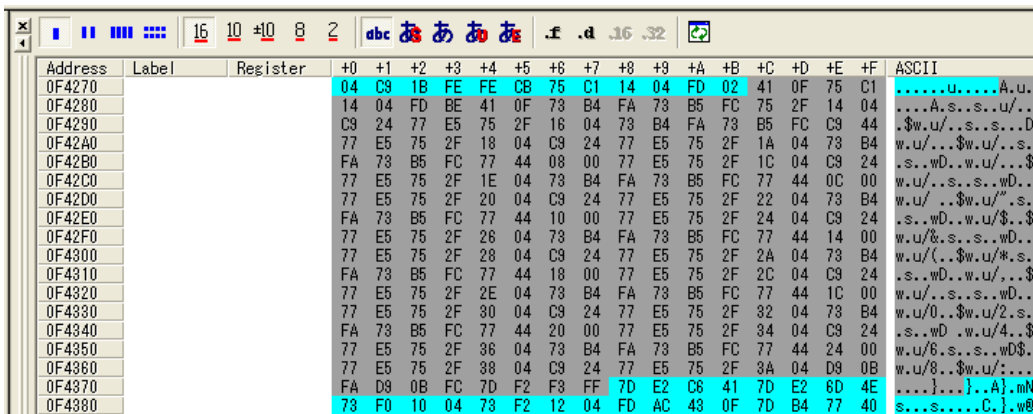
- (12) [表示開始アドレス]ダイアログボックスが表示されるので、[表示開始アドレス]を“H' F4270”にして[OK]を押してください。



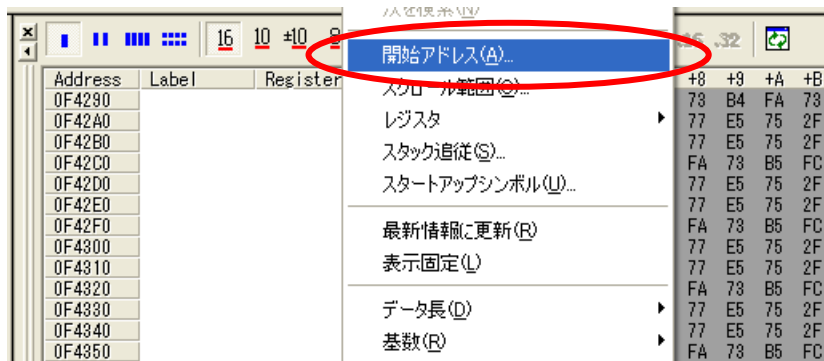
(13) メモリウィンドウが開きますので、ウィンドウ上で右クリックして[カバレッジ]の中の[有効]を選択してください。



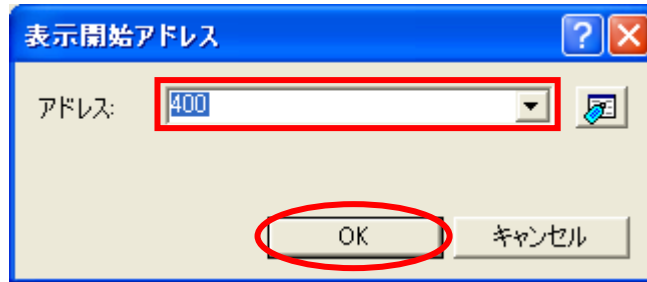
(14) メモリウィンドウが色分けされます。青色の部分が実行された命令で、灰色の部分が未実行の命令を表しています。



(15) 同じ手順で内蔵 RAM の内容を確認してみます。メモリウィンドウ上で右クリックして[開始アドレス]を選択してください。PC7501 のカバレッジ機能では、命令の実行網羅度だけでなく、データ領域の網羅度(アドレスをアクセスしたかどうか)も測定することが可能です。



(16) [表示開始アドレス]ダイアログボックスが表示されるので、[アドレス]を“H' 400”にして[OK]を押してください。



(17) 内蔵 RAM 内でアクセスがあった変数とアクセスされていない変数が確認できます。

Address	Label	Register	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ASCII
000400	_pool	[SB]	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000410			DF	D5	F8	0F	C8	41	00	00	7E	16	00	00	81	27	00	00A.....?
000420			6B	44	00	00	4B	79	00	00	FB	15	00	00	E2	59	00	00	kD..Ky.....Y..
000430			FB	1C	00	00	54	3F	00	00	F8	0F	00	00	00	00	00	00	...T?.....
000440			00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000450			00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000460			00	00	00	00	8C	0A	00	00	00	03	00	00	16	B5	44	2B	...l.....D+
000470			36	47	A3	00	91	D8	A3	C8	3F	79	27	FD	E7	8B	7F	FE	6G.....?y'....
000480			EE	D0	B0	7A	FA	B8	23	8C	0B	39	FB	67	F6	39	18	EB	...z..#.9.g.9..
000490			24	D8	FD	36	81	54	C5	13	FC	DB	73	7E	7B	51	A4	DE	\$.8.T....s~{Q..
0004A0			2E	A2	F6	44	64	3E	3A	47	EF	5D	FC	B9	3D	02	67	DC	...Dd>:G.]...=g.
0004B0			00	63	34	25	09	F3	80	92	BE	DC	37	78	FE	7E	EF	ED	.c4%.....7v...?
0004C0			BC	DB	6D	1A	FA	13	55	FD	D5	CE	E7	7F	B5	A1	26	EC	...m...U.....&.
0004D0			A0	01	64	CA	E2	4D	B9	DB	5A	09	B8	F7	20	EC	FF	FE	...d..M..Z.....
0004E0			A5	51	62	F6	59	58	80	47	D2	54	CF	2B	BD	76	98	78	..Ob.YX.G.T.+v.x
0004F0			D9	83	2D	98	7D	75	BD	AB	FF	5E	BC	6B	C8	43	00	0F	...}u...^k.C..
000500			41	8F	32	27	95	32	F2	68	8E	BA	71	AE	F3	55	7E	77	A.2'.2.h..q..U*w
000510			6F	AE	4C	12	CA	B1	74	F4	92	3E	06	BF	E3	78	39	BA	o.L...t...>...x9.

これらカバレッジ機能を使用することで、テストプログラムの妥当性の確認だけでなく、実際の動作における RAM 空間の使用率なども測定することも可能です。

5. よくある質問

5.1 ソース単位とカバレッジウィンドウで結果が異なるのは何故ですか？

サンプルプログラムを一通り実行して C0 カバレッジ計測結果を確認すると下図のようになります。

Function	Start	End	Coverage
change	0F41BE	0F4219	100.00 %
init	0F4014	0F4100	100.00 %
sort	0F4102	0F41BD	100.00 %
abort	0F4376	0F4376	100.00 %
main	0F421A	0F4223	100.00 %
tutorial	0F4224	0F4375	98.22 %

C ソース単位で関数 tutorial の C0 カバレッジを確認すると一見すべて実行されているかのように見えます。

```

16
17
18 F421A void main(void)
19 F421A {
20 F421E     while (1){
21 F4221         tutorial();
22 F4223     }
23
24 void tutorial(void)
25 F4224 {
26     long j;
27     int i;
28     struct Sample far *p_sam;
29
30 F4227     p_sam= &st;
31 F4231     init(p_sam);
32
33 F423D     for( i=0; i<10; i++ ){
34 F4248         j = rand();
35 F4254         if(j < 0){
36 F4259             j = -j;
37
38 F4262             a[i] = j;
39         }
40 F4278     sort(a);
41 F427E     change(a);
42
43 F4286     p_sam->s0=a[0];
44 F4298     p_sam->s1=a[1];
45 F42AE     p_sam->s2=a[2];
46 F42C8     p_sam->s3=a[3];
47 F42DE     p_sam->s4=a[4];
48 F42F6     p_sam->s5=a[5];
49 F430E     p_sam->s6=a[6];
50 F4328     p_sam->s7=a[7];
51 F433E     p_sam->s8=a[8];
52 F4356     p_sam->s9=a[9];
53 F436E     p_sam = NULL;
54 F4374 }
55
56 void abort(void)
57 F4376 {
58
59 F4376 }
60

```

しかし、アセンブリ単位で関数 tutorial の C0 カバレッジを確認すると一部未実行の部分が確認できます。

行番...	カ.	ア.	S...	逆アセ...	オブジェクトコー...	ラベル	混合
24							void tutorial(void)
25						_tutorial	{
26				F4224	7CF20A		ENTER #0AH
27							long j;
28							int i;
29							struct Sample far *p_sam;
30							p_sam= %st;
31				F4227	75CBFA3C04		MOV.W:G #043CH,-6H[FB]
				F422C	75CBFC0000		MOV.W:G #0000H,-4H[FB]
							init(p_sam);
				F4231	754BFC		PUSH.W:G -4H[FB]
				F4234	754BFA		PUSH.W:G -6H[FB]
				F4237	FD14400F		JSR.A init F4014H
				F423B	7DB4		ADD.B:Q #4H,SP
32							for(i=0; i<10; i++){
33				F423D	D90BFE		MOV.W:Q #0H,-2H[FB]
				F4240	778BFE0A00		CMP.W:G #000AH,-2H[FB]
				F4245	7DCA2F		JGE F4276H
34							j = rand();
				F4248	FD78430F		JSR.A _rand F4378H
				F424C	7CF3		EXTS.W R0
				F424E	732BF8		MOV.W:G R2,-8H[FB]
				F4251	730BF8		MOV.W:G R0,-AH[FB]
35							if(j < 0){
				F4254	7EBBCF		BTST:G 7,-7H[FB]
				F4257	6A0A		JEQ F4262H
36							j = -j;
				F4259	757BF8		NOT.W:G -8H[FB]
				F425C	755BF8		NEG.W -AH[FB]
				F425F	77EBF8		ADCF.W -8H[FB]
37							} a[i] = j;
38							MOV.W:G -2H[FB],A0
				F4262	73B4FE		SHL.W #2H,A0
				F4265	E914		MOV.W:G -AH[FB],0414H[A0]
				F4267	73BCF61404		MOV.W:G -8H[FB],0416H[A0]
				F426C	73BCF81604		ADD.W:Q #1H,-2H[FB]
				F4271	C91BFE		JMP.B F4240H
				F4274	FECB		}
39							sort(a);
40				F4276	75C11404		MOV.W:G #0414H,R1
				F427A	FD02410F		JSR.A \$sort F4102H
41							change(a);

このような現象が発生するのはプリフェッチが要因となっており、次のデータが必要かどうかに関わらず事前にキャッシュに読み込まれてしまいます。プリフェッチされるとアセンブリ単位では JEQ 命令を実行した後、次の命令まで実行してしまいます。これを C ソース単位で確認すると「j = -j」まで実行されたと判断され、見かけ上すべて実行されたように見えてしまいます。

ソース上で C0 カバレッジを確認する際は、混合モードか逆アセンブリモードで確認することをお勧めします。

6. 関連ドキュメント

PC7501 エミュレータおよび、HEW には本書で取り上げた機能以外にも便利な機能を豊富に備えています。各製品の仕様の詳細、技術情報、制限事項など有用な情報を記載していますので下記の関連ドキュメントも合わせて参照してください。

【PC7501 エミュレータ関連ドキュメント】

- ・ M16C R8C PC7501 エミュレータデバッグ ユーザーズマニュアル (M16C/60, M16C/30, M16C/Tiny, R8C/Tiny シリーズ用)
- ・ M16C R8C PC7501 エミュレータデバッグ リリースノート (M16C/60, M16C/30, M16C/Tiny, R8C/Tiny シリーズ用)
- ・ PC7501 セットアップガイド (M16C ファミリー用エミュレータ)
- ・ PC7501 ユーザーズマニュアル (M16C ファミリー用エミュレータ)

【High-performance Embedded Workshop 関連ドキュメント】

- ・ High-performance Embedded Workshop ユーザーズマニュアル
- ・ High-performance Embedded Workshop リリースノート

【CPU 関連ドキュメント】

- ・ M16C/60 シリーズハードウェアマニュアル
- ・ M16C/60、M16C/20、M16C/Tiny シリーズソフトウェアマニュアル

【M16C/60 シリーズ C コンパイラパッケージ関連ドキュメント】

- ・ M3T-NC30WA C コンパイラ ガイドブック (R8C/Tiny, M16C/60, M16C/30, M16C/20, M16C/10, M16C/Tiny シリーズ用 C コンパイラパッケージ)
- ・ M3T-NC30WA アセンブラ ユーザーズマニュアル (R8C/Tiny, M16C/60, M16C/30, M16C/20, M16C/10, M16C/Tiny シリーズ用 C コンパイラパッケージ)

本製品に関する情報は以下のルネサス・ウェブサイトをご覧ください:

日本サイト: <http://japan.renesas.com/pc7501>

グローバルサイト: <http://www.renesas.com/pc7501>

ホームページとサポート窓口

ルネサステクノロジホームページ

<http://japan.renesas.com/>

お問合せ先

<http://japan.renesas.com/inquiry>

csc@renesas.com

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2008.7.24	—	初版発行

本資料ご利用に際しての留意事項

1. 本資料は、お客様に用途に応じた適切な弊社製品をご購入いただくための参考資料であり、本資料中に記載の技術情報について弊社または第三者の知的財産権その他の権利の実施、使用を許諾または保証するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例など全ての情報の使用に起因する損害、第三者の知的財産権その他の権利に対する侵害に関し、弊社は責任を負いません。
3. 本資料に記載の製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的、あるいはその他軍事用途の目的で使用しないでください。また、輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
4. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例などの全ての情報は本資料発行時点のものであり、弊社は本資料に記載した製品または仕様等を予告なしに変更することがあります。弊社の半導体製品のご購入およびご使用に当たりましては、事前に弊社営業窓口で最新の情報をご確認頂きますとともに、弊社ホームページ(<http://www.renesas.com>)などを通じて公開される情報に常にご注意下さい。
5. 本資料に記載した情報は、正確を期すため慎重に制作したのですが、万一本資料の記述の誤りに起因する損害がお客様に生じた場合においても、弊社はその責任を負いません。
6. 本資料に記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を流用する場合は、流用する情報を単独で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断して下さい。弊社は、適用可否に対する責任を負いません。
7. 本資料に記載された製品は、各種安全装置や運輸・交通用、医療用、燃焼制御用、航空宇宙用、原子力、海底中継用の機器・システムなど、その故障や誤動作が直接人命を脅かしあるいは人体に危害を及ぼすおそれのあるような機器・システムや特に高度な品質・信頼性が要求される機器・システムでの使用を意図して設計、製造されたものではありません（弊社が自動車用と指定する製品を自動車に使用する場合を除きます）。これらの用途に利用されることをご検討の際には、必ず事前に弊社営業窓口へご照会下さい。なお、上記用途に使用されたことにより発生した損害等について弊社はその責任を負いかねますのでご了承願います。
8. 第7項にかかわらず、本資料に記載された製品は、下記の用途には使用しないで下さい。これらの用途に使用されたことにより発生した損害等につきましては、弊社は一切の責任を負いません。
 - 1) 生命維持装置。
 - 2) 人体に埋め込み使用するもの。
 - 3) 治療行為（患部切り出し、薬剤投与等）を行なうもの。
 - 4) その他、直接人命に影響を与えるもの。
9. 本資料に記載された製品のご使用につき、特に最大定格、動作電源電圧範囲、放熱特性、実装条件およびその他諸条件につきましては、弊社保証範囲内でご使用ください。弊社保証値を越えて製品をご使用された場合の故障および事故につきましては、弊社はその責任を負いません。
10. 弊社は製品の品質および信頼性の向上に努めておりますが、特に半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。弊社製品の故障または誤動作が生じた場合も人身事故、火災事故、社会的損害などを生じさせないよう、お客様の責任において冗長設計、延焼対策設計、誤動作防止設計などの安全設計（含むハードウェアおよびソフトウェア）およびエイジング処理等、機器またはシステムとしての出荷保証をお願いいたします。特にマイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願い致します。
11. 本資料に記載の製品は、これを搭載した製品から剥がれた場合、幼児が口に入れて誤飲する等の事故の危険性があります。お客様の製品への実装後に容易に本製品が剥がれることがなきよう、お客様の責任において十分な安全設計をお願い致します。お客様の製品から剥がれた場合の事故につきましては、弊社はその責任を負いません。
12. 本資料の全部または一部を弊社の文書による事前の承諾なしに転載または複製することを固くお断り致します。
13. 本資料に関する詳細についてのお問い合わせ、その他お気づきの点等がございましたら弊社営業窓口までご照会下さい。