

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い申し上げます。

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が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。



ユーザーズ・マニュアル

TW850 Ver.2.00

性能解析チューニング・ツール

対象デバイス V850シリーズ

資料番号 U17241JJ1V0UM00 (第1版)

発行年月 September 2004 CP(K)

© NEC Electronics Corporation 2004

〔メ モ〕

目次要約

第1章 概説	...	15
第2章 インストレーション	...	18
第3章 起動と終了	...	23
第4章 機能	...	26
第5章 操作方法	...	54
第6章 ウィンドウ・レファレンス	...	90
付録A メッセージ	...	191
付録B 制限値・限界値	...	215
付録C 中間ファイル	...	216
索引	...	221

Windows , Windows NT は米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

PC/AT は、米国 IBM Corp.の商標です。

Java およびすべての Java 関連の商標およびロゴは、米国およびその他の国における米国 Sun microsystems,Inc.の商標または登録商標です。

- 本資料に記載されている内容は2004年9月現在のもので、今後、予告なく変更することがあります。量産設計の際には最新の個別データ・シート等をご参照ください。
- 文書による当社の事前の承諾なしに本資料の転載複製を禁じます。当社は、本資料の誤りに関し、一切その責を負いません。
- 当社は、本資料に記載された当社製品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、一切その責を負いません。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。
- 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、一切その責を負いません。
- 当社は、当社製品の品質、信頼性の向上に努めておりますが、当社製品の不具合が完全に発生しないことを保証するものではありません。当社製品の不具合により生じた生命、身体および財産に対する損害の危険を最小限度にするために、冗長設計、延焼対策設計、誤動作防止設計等安全設計を行ってください。
- 当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定していただく「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。

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

特別水準：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

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、生命維持のための装置またはシステム等

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。意図されていない用途で当社製品の使用をお客様が希望する場合には、事前に当社販売窓口までお問い合わせください。

（注）

- （１）本事項において使用されている「当社」とは、NECエレクトロニクス株式会社およびNECエレクトロニクス株式会社が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う。
- （２）本事項において使用されている「当社製品」とは、（１）において定義された当社の開発、製造製品をいう。

はじめに

対 象 者 このマニュアルは、V850 シリーズの各製品の応用システムを設計、開発するユーザを対象としています。

目 的 このマニュアルは、次の構成に示す TW850 の機能をユーザに理解していただくことを目的としています。

構 成 このマニュアルは、大きく分けて次の内容で構成しています。

- ・概説
- ・インストレーション
- ・起動と終了
- ・機能
- ・操作方法
- ・ウィンドウ・レファレンス

読 み 方 このマニュアルの読者には、電気、論理回路、マイクロコンピュータ、C 言語、アセンブラに関する一般知識を必要とします。

V850 シリーズのハードウェア機能を知りたいとき

→ 各製品の**ユーザーズ・マニュアル ハードウェア編**を参照してください。

V850 シリーズの命令機能を知りたいとき

→ V850 シリーズ **ユーザーズ・マニュアル アーキテクチャ編**、または
V850E/MS1 **ユーザーズ・マニュアル アーキテクチャ編**を参照してください。

凡 例

データ表記の重み	: 左が上位桁、右が下位桁
注	: 本文中につけた注の説明
注意	: 気をつけて読んでいただきたい内容
備考	: 本文の補足説明
数の表記	: 2 進数 ...XXXX または XXXXB 10 進数...XXXX 16 進数...XXXXH
2 のべき数を示す接頭語（アドレス空間、メモリ容量）:	K（キロ）: $2^{10} = 1024$ M（メガ）: $2^{20} = 1024^2$ G（ギガ）: $2^{30} = 1024^3$
キーの表記	: このマニュアルでは、PC-9821 シリーズでのキー表記で説明しています。

関連資料 このマニュアルを使用する場合は、次の資料もあわせてご覧ください。

関連資料は暫定版の場合がありますが、この資料では「暫定」の表示をしておりません。

あらかじめご了承ください。

○ 開発ツールに関する資料（ユーザーズ・マニュアル）

（1/2）

資 料 名		資料番号	
		和文	英文
IE-703002-MC（V853, V850/SA1, V850/SB1, V850/SB2, V850/SC1, V850/SC2, V850/SC3, V850/SF1, V850/SV1 用インサークット・エミュレータ）		U11595J	U11595E
IE-V850E-MC, IE-V850E-MC-A（V850E/MA1, V850E/MA2, V850E/IA1, V850E/IA2, V850E/SV2 用インサークット・エミュレータ）		U14487J	U14487E
IE-703003-MC-EM1（V853 用インサークット・エミュレータ・オプション・ボード）		U11596J	U11596E
IE-703017-MC-EM1（V850/SA1 用インサークット・エミュレータ・オプション・ボード）		U12898J	U12898E
IE-703037-MC-EM1（V850/SB1, V850/SB2 用インサークット・エミュレータ・オプション・ボード）		U14151J	U14151E
IE-703040-MC-EM1（V850/SV1 用インサークット・エミュレータ・オプション・ボード）		U14337J	U14337E
IE-703079-MC-EM1（V850/SF1 用インサークット・エミュレータ・オプション・ボード）		U15447J	U15447E
IE-703102-MC（V850E/MS1 用インサークット・エミュレータ）		U13875J	U13875E
IE-703102-MC-EM1, IE-703102-MC-EM1-A（V850E/MS1 用インサークット・エミュレータ・オプション・ボード）		U13876J	U13876E
IE-703107-MC-EM1（V850E/MA1 用インサークット・エミュレータ・オプション・ボード）		U14481J	U14481E
IE-703116-MC-EM1（V850E/IA1 用インサークット・エミュレータ・オプション・ボード）		U14700J	U14700E
IE-703166-MC-EM1（V850E/SV2 用インサークット・エミュレータ・オプション・ボード）		U16688J	U16688E
CA850 Ver.3.00 C コンパイラ・パッケージ	操作編	U17293J	作成予定
	C 言語編	U17291J	作成予定
	アセンブリ言語編	U17292J	作成予定
	リンク・ディレクティブ編	U17294J	作成予定
PM plus Ver.6.00		U17178J	U17178E
ID850 Ver.2.50 統合ディバッガ	操作編	U16217J	U16217E
ID850NW Ver.2.51 統合ディバッガ	操作編	U16454J	U16454E
ID850QB Ver.2.80 統合ディバッガ	操作編	U16973J	U16973E
SM plus システム・シミュレータ	操作編	U17246J	U17246E
	ユーザ・オープン・インタフェース編	U17247J	U17247E
SM850 Ver.2.50 システム・シミュレータ	操作編	U16218J	U16218E
SM850 Ver.2.00 以上 システム・シミュレータ	外部部品ユーザ・オープン・インタフェース仕様編	U14873J	U14873E
RX850 Ver.3.13 以上 リアルタイム OS	基礎編	U13430J	U13430E
	インストレーション編	U13410J	U13410E
	テクニカル編	U13431J	U13431E
RX850 Pro Ver.3.15 リアルタイム OS	基礎編	U13773J	U13773E
	インストレーション編	U13774J	U13774E
	テクニカル編	U13772J	U13772E

○ 開発ツールに関する資料（ユーザーズ・マニュアル）

（2/2）

資 料 名	資料番号	
	和文	英文
RX-NET ネットワーク・ライブラリ（TCP/IP）Ver.1.30	U15083J	-
RX-NET ネットワーク・ライブラリ（PPP）Ver.1.30	U15303J	-
RX-NET ネットワーク・ライブラリ（DNS）Ver.1.30	U15304J	-
RX-NET ネットワーク・ライブラリ（DHCP）Ver.1.30	U15382J	-
RX-NET ネットワーク・ライブラリ（SMTP）	U15505J	-
RX-NET ネットワーク・ライブラリ（POP）	U15539J	-
RX-NET Ver.1.10 ネットワーク・ライブラリ（telnet）	U16085J	-
RX-NET Ver.1.00 ネットワーク・ライブラリ（FTP）	U15946J	-
RX-NET Ver.1.00 ネットワーク・ライブラリ（WebServer）	U16294J	-
RD850 Ver.3.01 タスク・ディバッガ	U13737J	U13737E
RD850 Pro Ver.3.01 タスク・ディバッガ	U13916J	U13916E
AZ850 Ver.3.20 システム・パフォーマンス・アナライザ	U14410J	U14410E
PG-FP4 フラッシュ・メモリ・プログラマ	U15260J	U15260E
TW850 Ver.2.00 性能解析チューニング・ツール	このマニュアル	U17241E

目次

第 1 章 概説	15
1.1 用語解説	15
1.2 機能概要	16
1.3 特徴	16
1.4 動作環境	17
1.4.1 ハードウェア環境	17
1.4.2 ソフトウェア環境	17
第 2 章 インストール	18
2.1 インストール手順	19
2.2 アンインストール手順	20
2.3 フォルダ構成	21
第 3 章 起動と終了	23
3.1 起動方法	24
3.1.1 PM plus からの起動	24
3.1.2 起動オプション	24
3.2 終了方法	25
3.2.1 メニュー・バーからの終了	25
3.2.2 タイトル・バーからの終了	25
第 4 章 機能	26
4.1 性能解析機能	26
4.1.1 性能解析対象	26
4.1.2 性能解析情報の獲得方式	28
4.1.3 性能解析機能詳細	30
4.1.4 性能解析処理における注意事項	32
4.2 チューニング機能	33
4.2.1 チューニング対象	33
4.2.2 チューニング機能詳細	34
4.2.3 チューニング処理における注意事項	36
4.3 GUI 機能	37
4.3.1 GUI 機能詳細	37
4.4 タイマ制御コード部	38
4.4.1 タイマ制御コード部の作成	38
4.4.2 タイマ制御コード部の検査	40
4.4.3 時間計測処理における注意事項	41
4.5 TW850 用システム・ライブラリ	42
4.6 アセンブリ言語ソース・ファイルにおける注意事項	43
4.6.1 性能解析対象関数，チューニング対象関数の検出	43
4.6.2 関数呼び出し箇所の検出	45
4.6.3 アセンブリ言語ソース・ファイルの記述形式	47
第 5 章 操作方法	54
5.1 TW850 の起動	57
5.1.1 TW850 の開始	57
5.1.2 プロジェクトの読み込み	57
5.2 ウィザードの起動	58
5.2.1 ウィザードの開始	58

5.2.2	プロジェクト情報の獲得	59
5.2.3	解析対象ソース・ファイルの選択	60
5.2.4	実行モジュール情報の獲得	61
5.3	実行項目の選択	62
5.4	[トレース・データ解析を行う (T)] 選択時	63
5.4.1	トレース解析オプションの選択	63
5.5	[プローブ解析を行う (オリジナル実行モジュールのみ) (P)] 選択時	64
5.5.1	プローブ解析オプションの選択	64
5.6	[オリジナル実行モジュールの性能解析を行う (A)] のみ選択時	65
5.6.1	実行解析範囲の選択	65
5.6.2	トレース・イベントの選択	66
5.6.3	命令キャッシュの選択	67
5.6.4	配置先の選択	68
5.7	[チューニングとチューニング後実行モジュールの解析を行う (O)] 選択時	69
5.7.1	実行解析範囲の選択	69
5.7.2	トレース・イベントの選択	70
5.7.3	命令キャッシュの選択	71
5.7.4	高速アクセス・メモリ領域の選択	72
5.7.5	チューニング方法の選択	73
5.8	[標準 (D)] 選択時	74
5.8.1	配置先の選択	74
5.9	[ユーザ指定 (U)] 選択時	75
5.9.1	高速アクセス・メモリ領域チューニングの選択	75
5.9.2	命令キャッシュ・チューニングの選択	76
5.9.3	コールペア補正オプションの選択	77
5.9.4	優先配置関数の選択	78
5.9.5	配置先の選択	79
5.10	性能解析処理 / チューニング処理の実行	80
5.10.1	実行前の確認	80
5.10.2	性能解析処理 / チューニング処理の実行	81
5.10.3	実行完了の確認	82
5.11	結果確認	83
5.11.1	履歴情報	83
5.11.2	実行モジュール情報	84
5.11.3	コールグラフ情報	85
5.11.4	関数情報テーブル	86
5.11.5	関数詳細情報	87
5.11.6	関数遷移情報	88
5.12	TW850 の終了	89

第 6 章 ウィンドウ・レファレンス 90

6.1	入力規則	93
6.2	ウィンドウ解説	94
	メイン・ウィンドウ	95
	コールグラフ・ウィンドウ	103
	関数情報テーブル・ウィンドウ	106
	関数遷移情報ウィンドウ	109
	実行モジュール情報ウィンドウ	114
	関数詳細情報ウィンドウ	116
	初期画面パネル	120
	プロジェクト情報獲得パネル	122
	解析対象ソース・ファイル選択パネル	123
	実行モジュール情報獲得パネル	125
	実行項目選択パネル	126

トレース解析オプション選択パネル	128
プローブ解析オプション選択パネル	130
実行解析範囲選択パネル	133
トレース・イベント選択パネル	137
命令キャッシュ選択パネル	141
高速アクセス・メモリ領域選択パネル	144
チューニング方法選択パネル	146
高速アクセス・メモリ領域チューニング選択パネル	149
命令キャッシュ・チューニング選択パネル	151
コールペア補正オプション選択パネル	153
優先配置関数選択パネル	155
配置先選択パネル	158
実行前確認パネル	162
性能解析 / チューニング実行パネル	163
実行完了確認パネル	165
表示関数選択ダイアログ	166
関数遷移情報検索ダイアログ	170
オプション・ダイアログ	174
バージョン情報ダイアログ	177
ファイル選択ダイアログ	179
メッセージ・ダイアログ	181
エラー・ダイアログ	182
確認ダイアログ	184
入力ダイアログ	186
TW850 起動オプション設定ダイアログ	188
 付録 A メッセージ	 191
A.1 表示形式	191
A.2 情報メッセージ	192
A.3 警告メッセージ	194
A.4 エラー・メッセージ	196
 付録 B 制限値・限界値	 215
B.1 制限値	215
B.2 限界値	215
 付録 C 中間ファイル	 216
 索引	 221

図目次

図 2-1	フォルダ構成	21
図 3-1	メイン・ウィンドウ	24
図 3-2	確認ダイアログ	25
図 4-1	RTOS 情報の更新イメージ	31
図 5-1	ウィザード処理 (性能解析処理, チューニング処理) の流れ	54
図 5-2	TW850 の操作手順 / 性能解析結果確認手順	55
図 5-3	メイン・ウィンドウ	57
図 5-4	初期画面パネル	58
図 5-5	プロジェクト情報獲得パネル	59
図 5-6	解析ソース・ファイル選択パネル	60
図 5-7	実行モジュール情報獲得パネル	61
図 5-8	実行項目選択パネル	62
図 5-9	トレース解析オプション選択パネル	63
図 5-10	プローブ解析オプション選択パネル	64
図 5-11	実行解析範囲選択パネル	65
図 5-12	トレース・イベント選択パネル	66
図 5-13	命令キャッシュ選択パネル	67
図 5-14	配置先選択パネル	68
図 5-15	実行解析範囲選択パネル	69
図 5-16	トレース・イベント選択パネル	70
図 5-17	命令キャッシュ選択パネル	71
図 5-18	高速アクセス・メモリ領域選択パネル	72
図 5-19	チューニング方法選択パネル	73
図 5-20	配置先選択パネル	74
図 5-21	高速アクセス・メモリ領域チューニング選択パネル	75
図 5-22	命令キャッシュ・チューニング選択パネル	76
図 5-23	コールベア補正オプション選択パネル	77
図 5-24	優先配置関数選択パネル	78
図 5-25	配置先選択パネル	79
図 5-26	実行前確認パネル	80
図 5-27	性能解析 / チューニング実行パネル	81
図 5-28	実行完了確認パネル	82
図 5-29	履歴表示領域	83
図 5-30	実行モジュール情報ウィンドウ	84
図 5-31	コールグラフ・ウィンドウ	85
図 5-32	関数情報テーブル・ウィンドウ	86
図 5-33	関数詳細情報ウィンドウ	87
図 5-34	関数遷移情報ウィンドウ	88
図 5-35	確認ダイアログ	89
図 6-1	メイン・ウィンドウ	95
図 6-2	コールグラフ・ウィンドウ	103
図 6-3	関数情報テーブル・ウィンドウ	106
図 6-4	関数遷移情報ウィンドウ	109
図 6-5	実行モジュール情報ウィンドウ	114
図 6-6	関数詳細情報ウィンドウ	116
図 6-7	初期画面パネル	120
図 6-8	プロジェクト情報獲得パネル	122
図 6-9	解析対象ソース・ファイル選択パネル	123
図 6-10	実行モジュール情報獲得パネル	125

図 6-11	実行項目選択パネル	126
図 6-12	トレース解析オプション選択パネル	128
図 6-13	ブローブ解析オプション選択パネル	130
図 6-14	実行解析範囲選択パネル	133
図 6-15	トレース・イベント選択パネル	137
図 6-16	命令キャッシュ選択パネル	141
図 6-17	高速アクセス・メモリ領域選択パネル	144
図 6-18	チューニング方法選択パネル	146
図 6-19	高速アクセス・メモリ領域チューニング選択パネル	149
図 6-20	命令キャッシュ・チューニング選択パネル	151
図 6-21	コールペア補正オプション選択パネル	153
図 6-22	優先配置関数選択パネル	155
図 6-23	配置先選択パネル	158
図 6-24	実行前確認パネル	162
図 6-25	性能解析 / チューニング実行パネル	163
図 6-26	実行完了確認パネル	165
図 6-27	表示関数選択ダイアログ	166
図 6-28	関数遷移情報検索ダイアログ	170
図 6-29	関数遷移情報検索ダイアログ (拡張検索条件表示時)	170
図 6-30	オプション・ダイアログ	174
図 6-31	[システム関連] タブ	175
図 6-32	バージョン情報ダイアログ	177
図 6-33	ファイル選択ダイアログ	179
図 6-34	メッセージ・ダイアログ	181
図 6-35	エラー・ダイアログ (TW850 起動時)	182
図 6-36	エラー・ダイアログ (TW850 実行中)	182
図 6-37	確認ダイアログ	184
図 6-38	入力ダイアログ	186
図 6-39	TW850 起動オプション設定ダイアログ	188
図 A-1	メイン・ウィンドウのメッセージ表示領域	191
図 A-2	メッセージ・ダイアログ	191
図 A-3	エラー・ダイアログ	191

表目次

表 2-1	TW850 の提供形式	18
表 4-1	アセンブリ言語ソース・ファイルの制限事項	47
表 6-1	TW850 のウィンドウ, パネル, ダイアログ	90
表 B-1	TW850 の制限値	215
表 B-2	TW850 の限界値	215

第 1 章 概説

TW850 は、NEC エレクトロニクス製の組み込み型制御用 32 ビット・マイクロコンピュータ V850 シリーズ上で動作するユーザ・プログラムの実行性能を解析 / チューニングするためのソフトウェア(性能解析チューニング・ツール)です。

1.1 用語解説

以下に、TW850 で出てくる用語について説明します。

- コールペア

ある関数からもうひとつの関数への呼び出し関係を“コールペア”と呼びます。
このとき、呼び出し側の関数を“親関数”，呼び出された側の関数を“子関数”と呼びます。

- ホット・パス

プログラム中で、最も実行時間がかかる関数遷移のパスを“ホット・パス”と呼びます。
ホット・パスにあたる関数を優先してチューニングを行うと、性能改善に効果を発揮する場合があります。

- 実行モジュール

TW850 では、必要に応じてオリジナル解析用実行モジュール、プロファイル用実行モジュール、チューニング後実行モジュールが生成されます。
ここで、オリジナル解析用実行モジュールとは、ソース・プログラム中の関数の静的情報を収集するために、関数の先頭にラベルを挿入したり、ファイル・シンボルを追加したものです。なお、性能的にはラベルやファイル・シンボルを挿入しない場合と変わりません。

- 命令キャッシュ・ミス

高速にアクセスできる命令キャッシュは、プログラム・サイズよりも小さい場合がほとんどです。そのため、命令キャッシュを搭載したプロセッサでは、プログラムを実行すると、その前に命令キャッシュに取り込まれていたプログラムが追い出され、そのとき実行しているプログラムを取り込む事象が発生します。このとき、命令キャッシュ上に実行するプログラムが見つからない事象を“命令キャッシュ・ミス”と呼びます。
なお、命令キャッシュ・ミスの発生率は、下記の計算式から算出された値となります。

$$\text{命令キャッシュ・ミスの回数} / (\text{命令キャッシュ・ヒットの回数} + \text{命令キャッシュ・ミスの回数})$$

- 高速アクセス・メモリ領域

TW850 では、ターゲット・システムの中でより高速にアクセスできるメモリ領域のことを“高速アクセス・メモリ領域”と呼んでいます。
使用頻度の高い関数や高速性を要求される関数などをこの高速アクセス・メモリ領域に配置することにより、プログラムを高速化することができます。

- 実行エンジン

ディバッガやシミュレータのように、実行モジュールを実行し、そのプロファイル情報を取得するために用いるものを“実行エンジン”と呼びます。

- far jump 呼び出し

全アドレス空間に分岐可能な命令による呼び出しを“far jump 呼び出し”と呼びます。
プローブ・コード挿入方式を用いた場合、挿入するプローブ・コードがプロファイル・ルーチンを呼び出しますが、プローブ・コードとプロファイル・ルーチンの配置距離が遠くなると、コンパイル・エラーが発生する場合があります。そのような場合、プロファイル・ルーチンの呼び出しに far jump 呼び出しを使用すると、配置距離が遠いプロファイル・ルーチンも呼び出すことができます。

1.2 機能概要

TW850 では、以下に示した性能解析機能 / チューニング機能 / GUI 機能を提供しています。
なお、各機能についての詳細は、「第 4 章 機能」を参照してください。

- 性能解析機能
 - メモリ配置情報の獲得
 - コールペア情報の獲得
 - コールカウント情報の獲得
 - 実行時間情報の獲得
 - 命令キャッシュ情報の獲得
 - RTOS 情報の獲得
- チューニング機能
 - 高速アクセス・メモリ領域のチューニング
 - 命令キャッシュのチューニング
 - コールペア情報の補正
 - 優先配置関数の指定
- GUI 機能
 - ウィザード形式での設定
 - 性能解析結果、チューニング結果の表示

1.3 特徴

以下に、TW850 の特徴を示します。

- 性能予測，性能見積り
ターゲット・システムの評価前にユーザ・プログラム，実行モジュールの性能予測作業，性能見積り作業を行いたい場合には，TW850 が提供している性能解析機能を利用することにより，これらの作業を実現することができます。
- 簡便な性能解析 / チューニング
TW850 が提供している性能解析機能 / チューニング機能を利用する際に必要となる各種設定（性能解析区間，実行環境の設定など）をウィザード形式による視覚的な案内で実現することにより，各種設定を簡便で効率の良いものとしています。
- 詳細な性能解析情報の提供
実行モジュールの実行性能を低下させる原因となっている箇所を関数単位で検出することができます。
- チューニング後実行モジュールの自動生成
ウィザード形式による視覚的な案内で実現された各種設定（該当情報の入力）を行うことにより，チューニングされた実行モジュール（チューニング後実行モジュール）を TW850 が自動生成します。

1.4 動作環境

1.4.1 ハードウェア環境

以下に、TW850 が動作するうえで必要となるハードウェアを示します。

- ホスト・マシン
PC-98NX シリーズ，または，IBM-PC/AT 互換機
- CPU
Pentium 450MHz 以上
- メイン・メモリ
128M バイト以上 (256M バイト以上を推奨)

1.4.2 ソフトウェア環境

以下に、TW850 が動作するうえで必要となるソフトウェアを示します。

- 基本ソフトウェア (必須)
Windows 2000 Professional , Windows 98 Second Edition , Windows Me , Windows XP Professional , Windows XP Home Edition , Windows NT 4.0 Service Pack 5 以上
注意 1 いずれの OS の場合も，最新の Service Pack がインストールされていることを推奨します。
注意 2 Windows XP のユーザ切り替えはサポートしていません。
- Java 実行環境 (必須)
Java2 Runtime Environment Standard Edition V1.4.2_04 : 米国 Sun Microsystems, Inc. 製
- C コンパイラ・パッケージ (必須)
CA850 Ver.3.00 以上 : NEC エレクトロニクス製
- デバイス・ファイル (必須)
使用するターゲット・デバイスのデバイス・ファイル : NEC エレクトロニクス製
- プロジェクト・マネージャ (必須)
PM plus Ver.6.00 以上 : NEC エレクトロニクス製
- RTOS(リアルタイム OS)
RX850 Ver.3.20 : NEC エレクトロニクス製
RX850 Pro Ver.3.20 : NEC エレクトロニクス製

以下から何れかを選択 (必須)

- ディバugg
ID850 Ver.3.00 以上 : NEC エレクトロニクス製
ID850NW Ver.3.00 以上 : NEC エレクトロニクス製
注意 (株)マイダス・ラボ製の ICE サーバを使用する場合は，ID850NW と RTE for WIN32 (Ver.5.10 以上) の組み合わせで使用するください。
- シミュレータ
SM850 Ver.3.00 以上 : NEC エレクトロニクス製
SM plus for V850 Ver.2.00 以上 : NEC エレクトロニクス製
注意 SM plus for V850 は，ToolLink 対応版のみとなります。

第2章 インストール

本章では、TW850 の提供媒体に格納されているファイル群をユーザの開発環境（ホスト・マシン）上にインストールする際の手順、および、ユーザの開発環境上からアンインストールする際の手順について解説しています。

TW850 の提供媒体は、ホスト・マシンの種類（Windows ベース）に合わせて用意されています。

表 2-1 に、TW850 の提供形式を示します。

表 2-1 TW850 の提供形式

ホスト・マシン	提供形式	提供媒体
Windows ベース - PC-98NX シリーズ - IBM-PC/AT 互換機	NEC エレクトロニクス・ツール・チェーン対応版	CD-ROM

2.1 インストール手順

以下に、TW850 の提供媒体に格納されているファイル群をホスト・マシン上にインストールする際の手順を示します。

1) Windows の起動

ホスト・マシン、および、周辺機器などの電源を投入し、Windows を起動します。

2) 提供媒体のセット

TW850 の提供媒体をホスト・マシンの該当デバイス装置 (CD-ROM ドライブ) にセットすることにより、セットアップ・プログラムが自動実行します。

以降、モニタ画面に表示されるメッセージに従ってインストール作業を実行します。

注意 セットアップ・プログラムが自動実行しない場合には、エクスプローラなどから CD-ROM ドライブの Install.exe を実行します。

3) ファイル群の確認

Windows の標準アプリケーション Explorer などを用いて、TW850 の提供媒体に格納されていたファイル群がホスト・マシン上にインストールされたことを確認します。

なお、各フォルダについての詳細は、「[2.3 フォルダ構成](#)」を参照してください。

注意 1 TW850 を再インストールする場合は、TW850 のアンインストール完了後に行ってください。

注意 2 同時に ToolLinkManager もインストールされます。

2.2 アンインストール手順

以下に、ホスト・マシンにインストールされているファイル群をアンインストールする際の手順を示します。

1) Windows の起動

ホスト・マシン、および、周辺機器などの電源を投入し、Windows を起動します。

2) [コントロール パネル] ウィンドウのオープン

<スタート> ボタンの [設定 (S)] メニュー → [コントロール パネル (C)] を選択し、[コントロール パネル] ウィンドウをオープンします。

3) [アプリケーションの追加と削除のプロパティ] ウィンドウのオープン

[コントロール パネル] ウィンドウの [アプリケーションの追加と削除] アイコンをダブル・クリックし、[アプリケーションの追加と削除のプロパティ] ウィンドウをオープンします。

注意 Windows XP の場合、[アプリケーションの追加と削除のプロパティ] ウィンドウは [プログラムの追加と削除] ウィンドウとなります。

4) TW850 の削除

[アプリケーションの追加と削除のプロパティ] ウィンドウの [セットアップと削除] タブに表示されているインストール済みソフトウェア一覧の中から “ NEC EL TW850 V2.00 ” を選択した後、< 追加と削除 (R)... > ボタンを押下します。

なお、[ファイル削除の確認] ウィンドウがオープンした際には、< はい (Y) > ボタンを押下します。

5) ファイル群の確認

Windows の標準アプリケーション Explorer などを用いて、ホスト・マシンにインストールされていたファイル群がアンインストールされたことを確認します。

なお、各フォルダについての詳細は、「[2.3 フォルダ構成](#)」を参照してください。

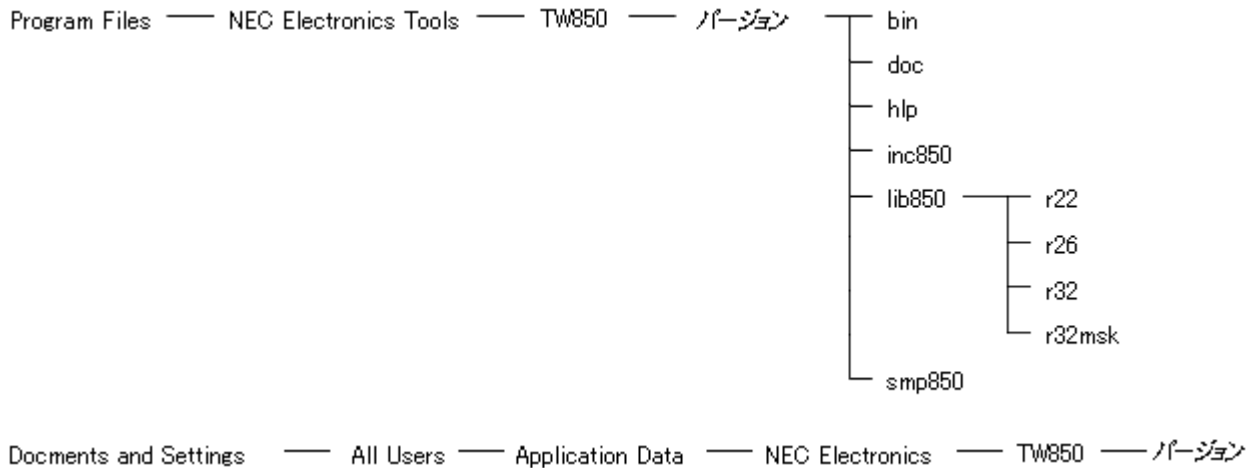
注意 ToolLinkManager はアンインストールされません。ToolLinkManager をアンインストールする場合は、上記と同様に、[コントロール パネル] ウィンドウから行ってください。

2.3 フォルダ構成

図 2-1 に、TW850 の提供媒体に格納されているファイル群をホスト・マシン上にインストールした際に生成されるフォルダ構成を示します。

なお、図 2-1 は、Program Files¥NEC Electronics Tools にインストールした場合の例となります。

図 2-1 フォルダ構成



以下に、各フォルダの概要を示します。

- 1) Program Files¥NEC Electronics Tools¥TW850¥バージョン¥bin
TW850 の実行形式ファイルが格納されているフォルダです。
TW850.EXE : TW850 の実行形式ファイル
- 2) Program Files¥NEC Electronics Tools¥TW850¥バージョン¥doc
TW850 のドキュメントが格納されているフォルダです。
- 3) Program Files¥NEC Electronics Tools¥TW850¥バージョン¥hlp
TW850 のヘルプが格納されているフォルダです。
TW850.CHM : ヘルプ
- 4) Program Files¥NEC Electronics Tools¥TW850¥バージョン¥inc850
TW850 のヘッダ・ファイルが格納されているフォルダです。
- 5) Program Files¥NEC Electronics Tools¥TW850¥バージョン¥lib850
TW850 用システム・ライブラリ、および、性能解析 / チューニング用データ・ファイルが格納されているフォルダです。
- 6) Program Files¥NEC Electronics Tools¥TW850¥バージョン¥lib850¥r22
TW850 用システム・ライブラリ (22 レジスタ・モード)、および、性能解析 / チューニング用データ・ファイル (22 レジスタ・モード) が格納されているフォルダです。
- 7) Program Files¥NEC Electronics Tools¥TW850¥バージョン¥lib850¥r26
TW850 用システム・ライブラリ (26 レジスタ・モード)、および、性能解析 / チューニング用データ・ファイル (26 レジスタ・モード) が格納されているフォルダです。
- 8) Program Files¥NEC Electronics Tools¥TW850¥バージョン¥lib850¥r32
TW850 用システム・ライブラリ (32 レジスタ・モード)、および、性能解析 / チューニング用データ・ファイル (32 レジスタ・モード) が格納されているフォルダです。

- 9) Program Files¥NEC Electronics Tools¥TW850¥バージョン¥lib850¥r32msk
TW850 用システム・ライブラリ (32 レジスタ・マスク・モード), および, 性能解析 / チューニング用データ・ファイル (32 レジスタ・マスク・モード) が格納されているフォルダです。
- 10) Program Files¥NEC Electronics Tools¥TW850¥バージョン¥smp850
タイマ制御コード部のサンプル・プログラムが格納されているフォルダです。
なお, タイマ制御コード部についての詳細は, 「[4.4 タイマ制御コード部](#)」を参照してください。
TWTIMER85.S : V853 TM11 用タイマ制御コード部
TWTIMER85E.S : V850E/MS1 TM10 用タイマ制御コード部
- 11) Documents and Settings¥All Users¥Application Data¥NEC Electronics¥TW850¥バージョン
TW850 で共通な情報を保存するファイルが格納されているフォルダです。
TW850.INI : TW850 で共通な情報を保存するファイル
- 注意 Documents and Settings は, ユーザ毎の設定情報が保存されているフォルダです。なお, 本フォルダは, 使用する OS によって異なります (上記は Windows XP の場合です)。

第3章 起動と終了

本章では、TW850 の起動方法、および、終了方法について解説しています。

TW850 を起動する前に、以下の事項を確認してください。

- Java 実行環境について

TW850 を起動するためには Java 実行環境が必要となります。

その際、TW850 は以下の順序で Java アプリケーション起動コマンド (JAVAW.EXE) を検索します。

ただし、NEC エレクトロニクス・ツール・チェーン対応版の場合、デフォルトで環境変数 PATH の先頭に、対応する Java 実行環境のインストール・ディレクトリが追加されます。

- 1) 環境変数 PATH で指定したフォルダ
- 2) Windows システム・フォルダ
- 3) Windows フォルダ

- NEC エレクトロニクス・ツール・チェーンについて

NEC エレクトロニクス・ツール・チェーンにおいて TW850 を起動する際には、PM plus、ディバッガ、または、シミュレータで TW850 の対象とするプロジェクトのビルド、ディバグが正常に行える状態にある必要があります。

また、TW850 の対象とするプロジェクトで使用するデバイス・ファイルは、デバイス・ファイル・インストーラで所定の場所にインストールする必要があります。

3.1 起動方法

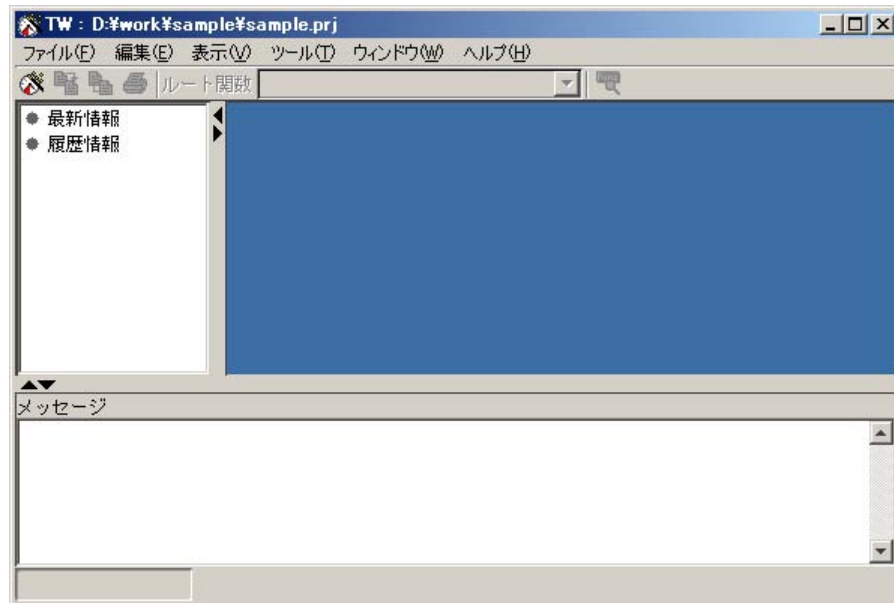
3.1.1 PM plus からの起動

以下に、TW850 を PM plus のメニュー・バーから起動する際の起動方法を示します。

なお、TW850 が正常に起動した際には、[図 3-1](#) に示した**メイン・ウィンドウ**が自動的にオープンします。

- 1) [ツール (T)] メニュー → [TW850 の起動 (U)] を選択します。

図 3-1 メイン・ウィンドウ



注意 1 起動オプションの指定は、[TW850 起動オプション設定ダイアログ](#)において行います。

注意 2 PM plus のツール・バーの < TW850 の起動 > ボタンを押下した場合、上記操作と同等の処理が実行されます。

3.1.2 起動オプション

TW850の起動オプションは、[TW850起動オプション設定ダイアログ](#)の[その他のオプション(I)]指定領域で指定します。

以下に、TW850 の起動オプションについて示します。

- -Dtw.NWireRealTimeTrace
N-Wire トレースで、トレースをオン時にリアルタイム・モードでディバグを実行することを指定します。
省略時 TW850 は、ノン・リアルタイム・モードを指定したものとして処理を実行します。
- -Dtw.DebuggerTimeout=DbgTmout
TW850 がディバグの起動処理を開始してから操作可能状態となるまでの応答待ち時間 (単位: 秒) を指定します。
なお、DbgTmout として指定可能な値は、1 ~ 3600 に限られます。
省略時 TW850 は、-Dtw.DebuggerTimeout=300 を指定したものとして処理を実行します。
- -Dtw.IgnoreJavaVersionWaning
JavaVM のバージョンが TW850 のサポートするバージョンよりも新しい場合に表示する [メッセージ・ダイアログ](#) を表示しないように指定します。
省略時 TW850 は、-Dtw.IgnoreJavaVersionWaning が指定されていないものとして処理を実行します。

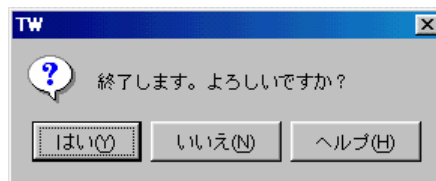
3.2 終了方法

3.2.1 メニュー・バーからの終了

以下に、TW850 を **メイン・ウィンドウ** のメニュー・バーから終了する際の終了方法を示します。
なお、TW850 が正常に終了する際には、[図 3-2](#) に示した**確認ダイアログ**が自動的にオープンします。

- 1) [ファイル (E)] メニュー → [終了 (X)...] を選択します。
- 2) **確認ダイアログ** の <はい (Y)> ボタンを押下します。

図 3-2 確認ダイアログ



3.2.2 タイトル・バーからの終了

以下に、TW850 を **メイン・ウィンドウ** のタイトル・バーから終了する際の終了方法を示します。
なお、TW850 が正常に終了する際には、**確認ダイアログ**が自動的にオープンします。

- 1) <閉じる> ボタンを押下します。
- 2) **確認ダイアログ** の <はい (Y)> ボタンを押下します。

第 4 章 機能

本章では、TW850 が提供している性能解析機能 / チューニング機能 / GUI 機能について解説しています。

4.1 性能解析機能

この節では、以下の項目について解説します。

- [性能解析対象](#)
- [性能解析情報の獲得方式](#)
- [性能解析機能詳細](#)
- [性能解析処理における注意事項](#)

4.1.1 性能解析対象

以下に、性能解析対象となる関数、プロジェクト、および、ターゲット環境について示します。

- “.text” セクションに割り付けられている関数

TW850 では、“.text” セクションに割り付けられている関数を性能解析対象としています。

なお、TW850 では、C 言語で記述されたユーザ・プログラムについては、一旦、アセンブリ言語に変換した後、各種性能解析情報の獲得、解析を行っています。このため、C 言語で記述されたユーザ・プログラム内のインライン展開関数は、TW850 の性能解析対象から除外されます。また、セクションに割り付けられていない関数は、“.text” セクションに割り付けられます。

- 内蔵 ROM の 1 番目のイメージ領域に存在する関数

性能解析対象となる関数が内蔵 ROM の 2 番目以降のイメージ領域に存在する場合、TW850 は性能解析を正しく行うことができません。

- 性能解析対象となる関数が存在するプロジェクト

性能解析対象となる関数がひとつも存在しないプロジェクトは、TW850 の対象外となります。

- ユーザ操作を必要としないプロジェクト

TW850 は、ディバグの起動から実行モジュールのダウンロード、および、実行を自動的に行うため、ディバグの起動から実行までの間にユーザ操作を必要とするプロジェクトは、TW850 の対象外となります。

また、解析対象範囲にユーザ操作が必要となる部分がある場合、その操作により解析結果が変化するため、正しくチューニングを行うことができません。

ただし、スクリプト・ファイルにフック・プロシジャを記述することによって、そのユーザ操作が実現可能な場合、TW850 の対象とすることができます。

以下に、V850E/ME2 の内蔵命令 RAM を使用する場合の例を示します。

V850E/ME2 の内蔵命令 RAM を使用する場合、実行モジュールのダウンロード前に、内蔵命令 RAM をライト・モードに設定し、CPU リセット後に、内蔵命令 RAM をリード・モードに設定する必要があります。

生成するプロジェクトの実行モジュールと同じフォルダに、以下のようなスクリプト・ファイル (実行モジュール名).tcl を作成してください。既に (実行モジュール名).tcl が存在する場合は、以下のスクリプトを追記してください。

```
proc BeforeDownload {} {  
    # ダウンロード前に、内蔵命令 RAM をライト・モードに設定  
    register IRAMM 0x3  
}  
  
proc AfterCpuReset {} {  
    # CPU リセット後に、内蔵命令 RAM をリード・モードに設定  
    register IRAMM 0x0  
}
```

TW850 では、適用するプロジェクトの実行モジュールのダウンロード後に CPU リセットを行うため、このスクリプトにより、ダウンロード前のフックで内蔵命令 RAM をライト・モードに設定し、CPU リセット後のフックで内蔵命令 RAM をリード・モードに設定することができます。

また、スクリプト・ファイル名を `tw.tcl` に変更することにより、TW850 実行時のみに実行するスクリプト・ファイルにすることもできます。

- 複数の実行モジュールのダウンロードを必要としないプロジェクト

実行モジュールを実行するために、複数の実行モジュールをダウンロードする必要があるプロジェクトは、TW850 の対象外となります。

ただし、スクリプト・ファイルにフック・プロシジャを記述することによって、その複数の実行モジュールのダウンロードが可能な場合、TW850 の対象とすることができます。

以下に、プロジェクトの実行モジュールを `module.out`、実行時に `module.out` と同時にダウンロードする必要がある実行モジュールを `submodule.out` とした場合の例を示します。

生成するプロジェクトの実行モジュールと同じフォルダに、以下のようなスクリプト・ファイル `module.tcl` を作成してください。既に `module.tcl` が存在する場合は、以下のスクリプトを追記してください。

```
proc AfterCpuReset {} {
    # CPU リセット後に、追加ダウンロード
    download -append submodule.out
}
```

TW850 では、適用するプロジェクトの実行モジュールのダウンロード後に CPU リセットを行うため、このスクリプトにより、CPU リセット後のフックで `submodule.out` を追加ダウンロードすることができます。

また、スクリプト・ファイル名を `tw.tcl` に変更することにより、TW850 実行時のみに実行するスクリプト・ファイルにすることもできます。

- CPU リセット時に正常に動作するターゲット環境

TW850 は、実行モジュールのダウンロード後に CPU リセットを行うため、CPU リセットを行うと正常に動作しなくなるようなターゲット環境の場合、TW850 は性能解析を正しく行うことができません。

ただし、スクリプト・ファイルにフック・プロシジャを記述することによって、CPU リセット後に正常にターゲット環境を動作させることが可能な場合、TW850 の対象とすることができます。

以下に、例を示します。

生成するプロジェクトの実行モジュールと同じフォルダに、以下のような CPU リセット後に実行するための必要な初期化処理を記述したスクリプト・ファイル (実行モジュール名).tcl を作成してください。

```
proc AfterCpuReset {} {
    # ここに、CPU リセット後に実行するための必要な初期化処理を記述
    .....
}
```

TW850 では、このスクリプトにより、CPU リセット後のフックで CPU リセット後に実行するための必要な初期化処理を行うことができます。

また、スクリプト・ファイル名を `tw.tcl` に変更することにより、TW850 実行時のみに実行するスクリプト・ファイルにすることもできます。

4.1.2 性能解析情報の獲得方式

TW850 では、各種性能解析情報の獲得方式として以下に示した 2 種類を提供しています。

- [トレース・データ解析方式](#)
- [プローブ・コード挿入方式](#)

1) トレース・データ解析方式

トレース・データ解析方式では、実行モジュールを実行した際に、実行エンジン (ディバッガ、シミュレータなど) が出力するハードウェア・トレース・データを解析することにより、各種性能解析情報を獲得します。また、この情報をもとに、チューニングを行うこともできます。

この方式では、実行エンジンで全トレース・データ、または、N-Wire トレース・データを取得できる必要があります。ただし、N-Wire トレース・データを取得する場合 (N-Wire トレース・データの取得をリアル・タイム・モードに設定できる CPU で、TW850 の起動オプションに `-Dtw.NWireRealTimeTrace` を指定した場合を除く)、CPU の機能特性として、トレース・データの出力中はプログラムの実行速度が低下します。そのため、TW850 の実行時間がトレース・データを出力しないときよりも長くなります。

以下に、トレース・データ解析方式を用いた性能解析を行うことができない場合について示します。

- 実行エンジンのトレース・バッファに空き領域がなくなった際、および、性能解析開始位置、性能解析終了位置において、実行モジュールの実行を一旦停止した際に正常な動作を保証することができない場合

TW850 では、ハードウェア・トレース・データを格納するための実行エンジンのトレース・バッファに空き領域がなくなった際 (トレース・イベントが指定されている場合を除く)、および、性能解析開始位置、性能解析終了位置において、実行モジュールの実行を一旦停止し、トレース・バッファの内容 (ハードウェア・トレース・データ)、および、各種性能解析情報を獲得、解析しています。このため、実行モジュールの実行を一旦停止した際に正常な動作を保証することができない実行モジュールでは、トレース・データ解析方式を用いた性能解析を行うことはできません。

2) プローブ・コード挿入方式

プローブ・コード挿入方式では、性能解析対象関数の呼び出し箇所に関数情報を獲得するための計測用プローブ・コードを挿入した状態で実行モジュールを実行することにより、各種性能解析情報 (関数の実行回数、実行時間) を獲得します。また、この情報を元にチューニングを行うこともできます。

この方式では、プログラムのコード・サイズが増加し、ターゲット上に関数の解析情報を格納するメモリ領域が必要になります。

トレース・データ解析方式に比べ、取得できる情報は少なくなりますが、トレース・データを取得することのできないターゲット環境での使用も可能で、解析区間でプログラムを停止することがありませんので、プログラムの実行を停止できない区間を解析区間に指定することができます。また、トレース・データ解析方式よりも高速に性能解析とチューニングをすることができます。

以下に、プローブ・コード挿入方式を用いた性能解析を行うことができない場合について示します。

- 性能解析対象とする関数が TW850 で認識できない場合

TW850 は、関数呼び出し部分にプローブ・コードを挿入するため、TW850 で認識できない関数についてはコールペア情報を出力することができません。

- チューニング後実行モジュールの性能解析を行う場合

チューニング後実行モジュールにプローブ・コードを挿入すると性能が変動してしまうため、関数単位での性能解析を行うことはできません。チューニング後実行モジュールについては、チューニング後実行モジュール全体についての項目、および、静的解析によって得られる項目のみ性能解析を行うことができます。

- プローブ・コード内で使用するレジスタが、割り込みにより破壊される場合

プローブ・コード内で使用するレジスタが、割り込みにより退避 / 復帰されずに破壊された場合、プロファイルのためのコードが正常に動作しないため、正しく性能解析を行うことができません。

- スタック・ポインタの初期化前に性能解析を行う場合

プローブ・コード内において、スタックにレジスタを退避するため、スタック・ポインタの初期化を行う前に性能解析を行うことはできません。

- 十分なテキスト領域やデータ領域が確保できない場合

プローブ・コードを挿入すると、プロファイル・データを格納するためのデータ領域が増加するため、十分なテキスト領域やデータ領域が確保できない場合、プローブ・コード挿入方式を用いた性能解析を行うことはできません。

- 性能解析開始位置、および、性能解析終了位置において、実行モジュールの実行を一旦停止した際に正常な動作を保証することができない場合

TW850 では、性能解析開始位置、および、性能解析終了位置において、実行モジュールの実行を一旦停止し、各種性能解析情報を獲得、解析しています。このため、実行モジュールの実行を一旦停止した際に正常な動作を保証することができない実行モジュールでは、プローブ・コード挿入方式を用いた性能解析を行うことはできません。

- 全アドレス空間に分岐できない分岐命令での分岐が、その命令の分岐限界を超える場合

全アドレス空間に分岐できない分岐命令での分岐が、TW850 が挿入するプローブ・コードによるテキスト・サイズの増加により、その命令の分岐限界を超える場合があります。そのため、そのような関数呼び出しを行っている実行モジュールでは、プローブ・コード挿入を用いた性能解析を行うことはできません。このような場合、関数呼び出しに `far jump` を用いたり、呼び合う関数同士を近くに宣言したり、分岐距離を短くすることで回避することができます。また、C 言語ソース・ファイルの場合は、関数呼び出しに `far jmp` を用いるコンパイル・オプションを指定することもできます。

プローブ・コード挿入方式では、コールペアの呼び出し回数、および、各関数の実行時間がメモリ上のプロファイル・テーブルに格納されます。以下に、プローブ・コード挿入方式で使用するメモリ容量 (単位: バイト) について示します。

注意 計算によるサイズにはホール・サイズが含まれていないため、実際に占有するメモリ容量はホール・サイズ分増加します。

- スタックのメモリ容量は、下記の計算式から算出された値となります。

最大 $24 + 20 \times$ プログラムの割り込み多重度

注意 割り込み多重度は、割り込みがない場合は 0、多重割り込みを許可しない割り込みを使用している場合は 1、多重割り込みを許可する割り込みを使用している場合は 2 以上として計算されます。

- 以下に、使用メモリの算出方法を示します。

【実行回数のみ計測する場合】

使用メモリ =
 コールペア数 $\times 4 +$
 動的呼び出しを持つ関数の数 $\times$ 関数の数 $+ 14 \sim 3200$ 注

【RTOS を使用せず、実行回数と時間を計測する場合】

使用メモリ =
 静的コールペア数 $\times 4 +$
 動的呼び出しを持つ関数の数 $\times$ 関数の数 $\times 4 +$
 関数の数 $\times 4 + 26 \sim 3200$ 注

関数 `main` から `func1`, `func2` を呼び出すプログラムにおいて、TW850 用システム・ライブラリ関数を使用していない場合、メモリ容量は以下のように算出されます。

使用メモリ = $2 \times 4 + 0 \times 3 \times 4 + 3 \times 4 + 26 = 46$

【RX850 を使用し、実行回数と時間を計測する場合】

使用メモリ =
 静的コールペア数 $\times 4 +$
 動的呼び出しを持つ関数の数 $\times$ 関数の数 $\times 4 +$
 関数の数 $\times 4 +$
 (ユーザ指定最大タスク ID + 1) $\times 2 + 102 \sim 3200$ 注

【RX850 Pro を使用し、実行回数と時間を計測する場合】

使用メモリ =
 静的コールペア数 * 4 +
 動的呼び出しを持つ関数の数 * 関数の数 * 4 +
 関数の数 * 4 +
 (ユーザ指定最大タスク ID + 1) * 2 +
 70 ~ 3200 注

注 TW850 用システム・ライブラリの使用状況により変化します。

注意 1 静的コールペア数は、TW850 用システム・ライブラリを含みますが、動的呼び出しを持つ関数を親とするコールペアは除きます。

注意 2 関数の数は、TW850 用システム・ライブラリを含みます。

4.1.3 性能解析機能詳細

TW850 では、以下に示した性能解析機能を提供しています。

- メモリ配置情報の獲得
- コールペア情報の獲得
- コールカウント情報の獲得
- 実行時間情報の獲得
- 命令キャッシュ情報の獲得
- RTOS 情報の獲得

1) メモリ配置情報の獲得

ユーザ・プログラム、実行モジュールを静的解析することにより、関数単位のメモリ配置情報を獲得します。
 なお、獲得したメモリ配置情報については、各種ウィンドウ ([関数詳細情報ウィンドウ](#) など) から参照することができます。

2) コールペア情報の獲得

ユーザ・プログラム、実行モジュールを静的解析することにより、関数単位のコールペア情報 (関数間の呼び出し関係) を獲得します。
 なお、獲得したコールペア情報については、各種ウィンドウ ([コールグラフ・ウィンドウ](#) , [関数詳細情報ウィンドウ](#) など) から参照することができます。

3) コールカウント情報の獲得

実行モジュールを実行した際に実行エンジン (デバッガ、シミュレータなど) が出力するハードウェア・トレース・データを解析する、あるいは、性能解析対象関数の呼び出し箇所に計測用プローブ・コードを挿入した状態で実行モジュールを実行することにより、関数単位のコールカウント情報 (どの関数からどの関数を何回呼び出したか) を獲得します。
 なお、獲得したコールカウント情報については、各種ウィンドウ ([コールグラフ・ウィンドウ](#) , [関数情報テーブル・ウィンドウ](#) , [関数詳細情報ウィンドウ](#) など) から参照することができます。

4) 実行時間情報の獲得

実行モジュールを実行した際に実行エンジン (デバッガ、シミュレータなど) が出力するハードウェア・トレース・データを解析する、あるいは、性能解析対象関数の呼び出し箇所、および、戻り箇所に計測用プローブ・コードを挿入した状態で実行モジュールを実行することにより、関数単位の実行時間情報を獲得します。
 なお、獲得した実行時間情報については、各種ウィンドウ ([関数情報テーブル・ウィンドウ](#) , [関数詳細情報ウィンドウ](#) など) から参照することができます。

5) 命令キャッシュ情報の獲得

実行モジュールを実行した際に実行エンジン (ディバッガ, シミュレータなど) が出力するハードウェア・トレース・データ (全トレース・データ) を解析することにより, 関数単位の命令キャッシュ情報 (命令キャッシュ・ヒットの回数, 命令キャッシュ・ミスの回数) を獲得します。

なお, 獲得した命令キャッシュ情報については, 各種ウィンドウ ([実行モジュール情報ウィンドウ](#), [関数情報テーブル・ウィンドウ](#), [関数詳細情報ウィンドウ](#)など) から参照することができます。

6) RTOS 情報の獲得

実行モジュールを実行した際に実行エンジン (ディバッガ, シミュレータなど) が出力するハードウェア・トレース・データを解析する, あるいは, RTOS(RX850, RX850 Pro) が提供している機能の呼び出し箇所, および, 戻り箇所に計測用プローブ・コードを挿入した状態で実行モジュールを実行することにより, RTOS 情報 (実行回数, 実行時間など) を獲得します。

• 実行回数

TW850 では, 以下に示した箇所を通過した際, 実行回数のカウント・アップを行います。

- 拡張 SVC ハンドラの入り口
- システム・コールの入り口

• 実行時間

TW850 では, 以下に示した箇所を通過した際, 実行時間の測定を開始します。

- 拡張 SVC ハンドラの入り口
- システム・コールの入り口
- スケジューリング処理の入り口

TW850 では, 以下に示した箇所を通過した際, 実行時間の測定を終了します。

- システム・コールの出口
- 割り込みハンドラの入り口
- 割り込みハンドラの出口
- アイドル・ハンドラの入り口

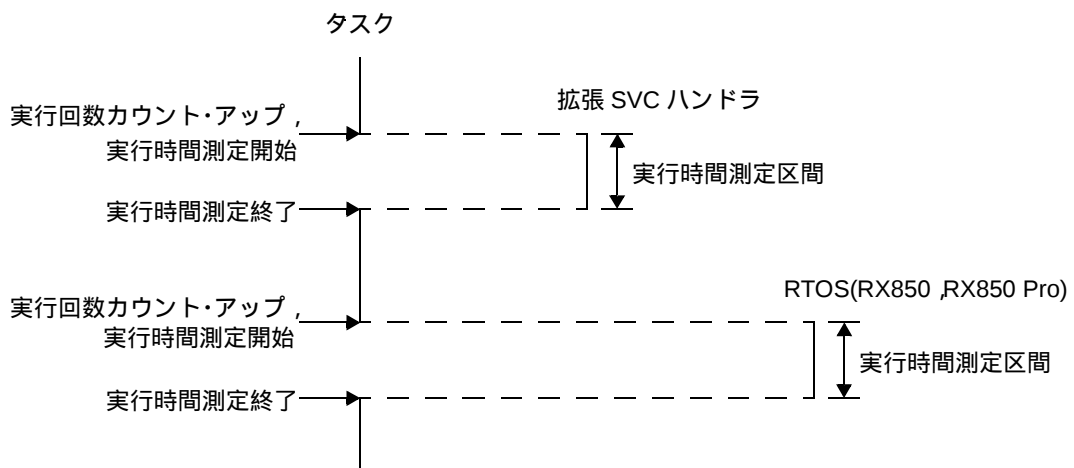
• GROSS 時間

トレース解析時, システム・コールの実行時間は, タスクの GROSS 時間には加算されません。

なお, 獲得した RTOS 情報については, [関数情報テーブル・ウィンドウ](#)から参照することができます。

[図 4-1](#) に, RTOS 情報の更新イメージを示します。

図 4-1 RTOS 情報の更新イメージ



注意 1 RTOS が提供している機能を利用したユーザ・プログラム、実行モジュールの性能解析時に各種性能解析情報の獲得方式として“プローブ・コード挿入方式”を指定した場合、RTOS の初期化処理が完了する以前に呼び出された関数を性能解析対象関数とすることはできません。

注意 2 トレース解析の場合、拡張 SVC ハンドラ内は解析対象外です。

4.1.4 性能解析処理における注意事項

性能解析処理について、以下に示したことに注意する必要があります。

- プローブ・コード挿入方式による性能解析を行う場合、TW850 は、以下に示すセグメントを生成し、プロファイル用ルーチンなどをそのセグメントに配置します。

TWROUTINE : プロファイル・ルーチン用セグメント

TWTABLE : プロファイル・テーブル用セグメント

- ユーザ・プログラム、実行モジュールの性能解析時にプローブ・コード挿入方式を利用した時間計測処理を行う場合、該当タイマ・カウンタを TW850 用に初期化、および、使用するための処理ルーチン（タイマ制御コード部）が必要となります。

なお、タイマ制御コード部についての詳細は、「[4.4 タイマ制御コード部](#)」を参照してください。

- 関数の動的な呼び出しを行っている場合、TW850 は、実際に呼び出し関係のあったものについてのみコールペアを生成します。

ただし、コールペアは、プロジェクトに変更が行われた場合にのみ消去されるため、コールペアが存在しても実行回数が 0 となる可能性があります。

例えば、1 回目の性能解析時に、func1 から動的呼び出しにより func2 を呼び出し、2 回目の性能解析時に、func1 から func2 を呼び出さなかった場合 func1 から func2 を呼び出すコールペアは存在しますが、実行回数は 0 となります。

- コールペアの生成は、性能解析情報の獲得方式により異なります。

【トレース・データ解析方式の場合】

TW850 が動的呼び出しを行っている認識した関数から他の関数の先頭に処理が移ったときに、コールペアが生成されます。

【プローブ・コード挿入方式の場合】

TW850 が動的呼び出しと認識できる方法（割り込みなど）で、ある関数から他の関数を呼び出したときに、コールペアが生成されます。

- ICE サーバを用いている場合、プログラムの実行を再開した直後のトレース・フレームについて、正しい実行時間を取得できません。また、トレース停止時に数フレーム取りこぼした場合、その部分について正しい実行時間を取得できません。

- アセンブリ言語ソース・ファイルを性能解析対象とする際、記述によっては TW850 が認識できない場合があります。なお、性能解析対象となるアセンブリ言語・ソース・ファイルについての詳細は、「[4.6.3 アセンブリ言語ソース・ファイルの記述形式](#)」を参照してください。

4.2 チューニング機能

この節では、以下の項目について解説します。

- [チューニング対象](#)
- [チューニング機能詳細](#)
- [チューニング処理における注意事項](#)

4.2.1 チューニング対象

以下に、チューニング対象となる関数、および、プロジェクトについて示します。

- “.text ” セクションに割り付けられている関数
TW850 では、“.text ” セクションに割り付けられている関数をチューニング対象としています。
なお、TW850 では、C 言語で記述されたユーザ・プログラムについては、一旦、アセンブリ言語に変換した後、各種チューニング情報の獲得、チューニングを行っています。このため、C 言語で記述されたユーザ・プログラム内のインライン展開関数は、TW850 のチューニング対象から除外されます。また、セクションに割り付けられていない関数は、“.text ” セクションに割り付けられます。
- 高速アクセス・メモリ領域の空きサイズや命令キャッシュ・メモリ・サイズよりサイズが小さい関数
関数のサイズが高速アクセス・メモリ領域の空きサイズや命令キャッシュ・メモリ・サイズより大きい場合、その関数はチューニング対象から除外されます。
そのような関数については、関数のサブルーチン化、細分化によってサイズを小さくすることにより、チューニング対象とすることができます。
- 内蔵 ROM の 1 番目のイメージ領域に存在する関数
チューニング対象となる関数が内蔵 ROM の 2 番目以降のイメージ領域に存在する場合、TW850 はチューニングを正しく行うことができません。
- チューニング対象となる関数が存在するプロジェクト
チューニング対象となる関数がひとつも存在しないプロジェクトは、TW850 の対象外となります。
- ユーザ操作を必要としないプロジェクト
TW850 は、ディバッガの起動から実行モジュールのダウンロード、および、実行を自動的に行うため、ディバッガの起動から実行までの間にユーザ操作を必要とするプロジェクトは、TW850 の対象外となります。
また、解析対象範囲にユーザ操作が必要となる部分がある場合、その操作により解析結果が変化するため、正しくチューニングを行うことができません。

ただし、スクリプト・ファイルにフック・プロシジャを記述することによって、そのユーザ操作が実現可能な場合、TW850 の対象とすることができます。

以下に、V850E/ME2 の内蔵命令 RAM を使用する場合の例を示します。

V850E/ME2 の内蔵命令 RAM を使用する場合、実行モジュールのダウンロード前に、内蔵命令 RAM をライト・モードに設定し、CPU リセット後に、内蔵命令 RAM をリード・モードに設定する必要があります。

生成するプロジェクトの実行モジュールと同じフォルダに、以下のようなスクリプト・ファイル (実行モジュール名).tcl を作成してください。既に (モジュール名).tcl が存在する場合は、以下のスクリプトを追記してください。

```
proc BeforeDownload {} {
    # ダウンロード前に、内蔵命令 RAM をライト・モードに設定
    register IRAMM 0x3
}

proc AfterCpuReset {} {
    # CPU リセット後に、内蔵命令 RAM をリード・モードに設定
    register IRAMM 0x0
}
```

TW850 では、適用するプロジェクトの実行モジュールのダウンロード後に CPU リセットを行うため、このスクリプトにより、ダウンロード前のフックで内蔵命令 RAM をライト・モードに設定し、CPU リセット後のフックで内蔵命令 RAM をリード・モードに設定することができます。
また、スクリプト・ファイル名を tw.tcl に変更することにより、TW850 実行時のみに実行するスクリプト・ファイルにすることもできます。

4.2.2 チューニング機能詳細

TW850 では、以下に示したチューニング機能を提供しています。

- 高速アクセス・メモリ領域のチューニング
- 命令キャッシュのチューニング
- コールペア情報の補正
- 優先配置関数の指定

注意 TW850 では、“命令キャッシュのチューニング”、“高速アクセス・メモリ領域のチューニング”が同時に指定された場合、以下に示した順序でユーザ・プログラム、実行モジュールのチューニングを行います。

高速アクセス・メモリ領域のチューニング > 命令キャッシュのチューニング

1) 高速アクセス・メモリ領域のチューニング

TW850 では、ターゲット・システムの中でより高速にアクセスできるメモリ領域のことを高速アクセス・メモリと呼んでいます。使用頻度の高い関数や高速性を要求される関数などをこの高速アクセス・メモリ領域に配置することにより、プログラムを高速化することができます。しかし、その関数を手動で選定することは、ユーザにとって負担となり、また、プログラム規模が大きくなるほど困難な作業となります。

TW850 は、実行モジュールの実行性能を落とす原因となっている関数を優先的に高速アクセス・メモリ領域（内蔵 ROM など）へと配置することにより、実行モジュールの実行性能を向上させるチューニング機能を搭載しています。

このチューニング機能により、ユーザは高速アクセス・メモリ領域の存在を意識することなく、プログラミングを行うことができます。また、高速アクセス・メモリ領域に対する配置方式も容易に変更することができるため、さまざまな配置方式で高速化を試すことができます。

なお、TW850 では、高速アクセス・メモリ領域に対する関数配置方式として、以下に示した 3 種類（コールペアの呼び出し回数が多い関数を優先、実行に時間がかかる関数を優先、命令キャッシュ・ミスが多い関数を優先）を提供しています。

- コールペアの呼び出し回数が多い関数を優先
各種関数のコールペア情報（関数間の呼び出し関係）をもとに、呼び出し回数の多い関数（呼び出し関数、呼び出され関数）のうち、コード・サイズの小さい関数を優先的に高速アクセス・メモリ領域に配置します。
- 実行に時間がかかる関数を優先
各種関数の実行時間情報をもとに、1 命令当たりの実行に時間を要する関数を優先的に高速アクセス・メモリ領域に配置します。
なお、ユーザ・プログラム、実行モジュールの性能解析時に“プローブ・コード解析方式”を指定し、“関数の実行時間を計測”を指定しなかった場合、本関数配置方式を指定することはできません。
- 命令キャッシュ・ミスが多い関数を優先
各種関数の命令キャッシュ情報（命令キャッシュ・ヒットの回数、命令キャッシュ・ミスの回数）をもとに、命令キャッシュ・ミスの回数が多い関数を優先的に高速アクセス・メモリ領域に配置します。
なお、ユーザ・プログラム、実行モジュールの性能解析時に各種性能解析情報の獲得方式として“プローブ・コード解析方式”を指定した場合、または、命令キャッシュ・ミス情報を獲得することができない場合、本関数配置方式を指定することはできません。

注意 TW850 の高速アクセス・メモリ領域のチューニングでは、既にプログラムやデータで使用されている高速アクセス・メモリ領域への関数の配置は行いません。リンク・ディレクティブ・ファイルなどで高速アクセス・メモリ領域への関数の配置を行っている場合、最終アドレスに最も近い関数の直後から高速アクセス・メモリ領域の最後まで範囲が関数の配置対象となります。ただし、その関数の配置対象となった範囲に NMI 割り込みベクタ領域、セキュリティ ID コード格納領域、または、フラッシュ・マスク・オプション領域が含まれている場合は、その領域の直後から高速アクセス・メモリ領域の最後まで範囲が関数の配置対象となります。

効率の良い高速アクセス・メモリ領域のチューニングを行うことができない場合は、TW850 を適用する前に、高速アクセス・メモリ領域への関数の配置を必要最小限にすることで、TW850 は空いている高速アクセス・メモリ領域を有効に利用するチューニングを行うことができます。また、高速アクセス・メモリ領域への関数の配置をまったく行わないことで、TW850 は高速アクセス・メモリ領域をより有効に利用するチューニングを行うことができます。

2) 命令キャッシュのチューニング

高速にアクセスできる命令キャッシュは、プログラム・サイズよりも小さい場合がほとんどです。そのため、命令キャッシュを搭載したプロセッサでは、プログラムを実行すると、その前に命令キャッシュに取り込まれていたプログラムが追い出され、その時実行しているプログラムを取り込む事象が発生します。このとき、命令キャッシュ上に実行するプログラムが見つからない事象を命令キャッシュ・ミスと呼びます。命令キャッシュ上のプログラムの入れ替えには時間を要し、プログラマが命令キャッシュ・ミスを減らすように関数配置することは、多大な負担のかかる作業となります。

TW850 は、各種関数を命令キャッシュ・ミスの回数が軽減されるような関数配置とすることにより、実行モジュールの実行性能を向上させるチューニング機能を搭載しています。

このチューニング機能により、プログラマは関数配置アルゴリズムとそのアルゴリズムに関するオプションを容易に変更することができるため、さまざまな配置方法で高速化を試すことができます。

なお、TW850 では、命令キャッシュに対する関数配置方式として以下に示した 2 種類 (カラリング・アルゴリズム、グルーピング・アルゴリズム) を提供しています。

- カラリング・アルゴリズム

各種関数のコールペア情報 (関数間の呼び出し関係) をもとに、可能な範囲で全てのコールペア同士の命令キャッシュ・ミスを減らすように関数配置を行います。

- グルーピング・アルゴリズム

各種関数のコールペア情報 (関数間の呼び出し関係) をもとに、命令キャッシュのサイズをグループ・サイズの上限值として関数間の呼び出し関係が強い関数をグループ化し、グループ内での命令キャッシュ・ミスを減らすように配置を行います。

また、TW850 では、命令キャッシュに対する配置優先度補正オプションとして以下に示した 2 種類 (呼び出し回数優先、深さ優先) を提供しています。

- 呼び出し回数優先

各種関数のコールペア情報 (関数間の呼び出し関係) をもとに、呼び出し回数の多い関数を優先的に命令キャッシュに配置します。

- 深さ優先

各種関数のコールペア情報 (関数間の呼び出し関係) をもとに、呼び出し関係の末端に近い関数を優先的に命令キャッシュに配置します。

注意 TW850 では、キャッシュ機構が 2-ウェイ以上の場合には、“ウェイ数考慮” オプションを指定することもできます。

“ウェイ数考慮” オプションを指定した場合、キャッシュ上における複数の関数の重なりを許容できるため、“呼び出し回数優先”、および、“深さ優先” の場合に比べて、関数を最適に配置することができる可能性があります。

3) コールペア情報の補正

各種関数のコールペア情報 (関数間の呼び出し関係) を補正することにより、チューニング処理 (高速アクセス・メモリ領域のチューニング、命令キャッシュのチューニング) の改善率を向上させています。

なお、TW850 では、コールペア情報の補正方式として以下に示した 3 種類 (float ランタイム・ライブラリ、ホット・パスに当たる関数、ループ処理内関数) を提供しています。

- float ランタイム・ライブラリ

float ランタイム・ライブラリ関数 (浮動小数点演算などを行った場合に、コンパイラによって使用されるライブラリ関数) に関わるコールペアの呼び出し回数を補正することにより、実行モジュールの実行性能を向上させます。

- ホット・パスに当たる関数

関数の子関数を含めた実行時間 (GROSS 時間) の情報をもとに、その関数に関わるコールペアの呼び出し回数を補正することにより、実行モジュールの実行性能を向上させます。

なお、ユーザ・プログラム、実行モジュールの性能解析時に “トレース・データ解析方式” を指定しなかった場合、本補正方式を指定することはできません。

- ループ処理内関数

同一ループ処理に関わるコールペアの呼び出し回数を補正することにより、実行モジュールの実行性能を向上させます。

注意 TW850 では、“float ランタイム・ライブラリ”、“ホット・パスに当たる関数”、“ループ処理内関数” が同時に指定された場合、以下に示した順序でコールペア情報の補正を行います。

float ランタイム・ライブラリ > ホット・パスにあたる関数 > ループ処理内関数

4) 優先配置関数の指定

ユーザが優先的にチューニング対象としたい関数を指定することにより、チューニング処理 (高速アクセス・メモリ領域のチューニング、命令キャッシュのチューニング) の改善率を向上させています。

なお、TW850 では、“優先配置関数の指定” が行われた場合、基本的に該当関数を “高速アクセス・メモリ領域のチューニング” の最優先配置対象関数としています。

注意 TW850 では、該当関数を高速アクセス・メモリ領域に配置することができなかった場合、“命令キャッシュのチューニング” の最優先配置対象関数としています。

4.2.3 チューニング処理における注意事項

チューニング処理について、以下に示したことに注意する必要があります。

- TW850 は、関数の再配置を行うときにその距離を考慮しないため、全アドレス空間に分岐できない分岐命令での分岐が、その命令の分岐限界を超える場合があります。そのため、そのような関数呼び出しを行っている実行モジュールのチューニングを行うことができません。

このような場合、関数呼び出しに far jump を用いることで回避することができます。また、C 言語ソース・ファイルの場合は、関数呼び出しに far jmp を用いるコンパイル・オプションを指定することもできます。

- TW850 は、ユーザがセグメントを指定しない限り、以下に示すセグメントを生成し、再配置の対象となった関数をそのセグメントに配置します。

TWTEXT : 関数配置用セグメント

TWFAMEM : 高速アクセス・メモリ関数配置用セグメント

なお、プロジェクトのリンク・ディレクティブ・ファイル中に TWTEXT セグメントが存在している場合は、TW850 ではそのセグメントに関数を配置することができます。

TWFAMEM セグメントが存在している場合は、チューニング後実行モジュールのビルドを行うことができません。

- アセンブリ言語ソースをチューニング対象とする際、記述によっては TW850 が認識できない場合があります。なお、アセンブリ言語ソースについての詳細は、「4.6.3 アセンブリ言語ソース・ファイルの記述形式」を参照してください。

4.3 GUI 機能

この節では、以下の項目について解説します。

- [GUI 機能詳細](#)

4.3.1 GUI 機能詳細

TW850 では、以下に示した GUI 機能を提供しています。

- [ウィザード形式での設定](#)
- [性能解析結果，チューニング結果の表示](#)

1) ウィザード形式での設定

TW850 が提供している性能解析機能 / チューニング機能を利用する際に必要となる各種設定（性能解析区間，実行環境の設定など）をウィザード形式による視覚的な案内で実現することにより，各種設定を簡便で効率の良いものとしています。

2) 性能解析結果，チューニング結果の表示

TW850 が行った性能解析結果，および，チューニング結果を視覚的に確認しやすいグラフィカルな形式で表示させることにより，性能解析結果，および，チューニング結果の比較，検討を容易なものとしています。
なお，性能解析結果，および，チューニング結果については，各種ウィンドウ（[実行モジュール情報ウィンドウ](#)，[コールグラフ・ウィンドウ](#)，[関数情報テーブル・ウィンドウ](#)，[関数遷移情報ウィンドウ](#)，[関数詳細情報ウィンドウ](#)）から参照することができます。

4.4 タイマ制御コード部

TW850 では、プローブ・コード挿入方式を利用した時間計測処理において、ターゲット・システム上のタイマ・カウンタを 1 つ占有します。

このため、ユーザ・プログラム、実行モジュールの性能解析時に“プローブ・コード挿入方式を利用した時間計測処理”を行う場合は、該当タイマ・カウンタを TW850 用に初期化、および、使用するための処理ルーチン（タイマ制御コード部）が必要となります。

注意 プローブ・コード挿入方式を利用する際に時間計測を行わない場合は、ターゲット・システム上のタイマ・カウンタを占有しないため、タイマ制御コード部は必要ありません。

この節では、以下の項目について解説します。

- タイマ制御コード部の作成
- タイマ制御コード部の検査
- 時間計測処理における注意事項

4.4.1 タイマ制御コード部の作成

タイマ制御コード部は、使用するターゲットに合わせて作成します。

注意 TW850 では、タイマ制御コード部のサンプル・プログラムを提供しています。

- Program Files¥NEC Electronics Tools¥TW850 ¥バージョン¥smp850 ¥tw ¥TWTIMER85.S
V853 TM11 用タイマ制御コード部です。
- Program Files¥NEC Electronics Tools¥TW850 ¥バージョン¥smp850 ¥tw ¥TWTIMER85E.S
V850E/MS1 TM10 用タイマ制御コード部です。

以下に、タイマ制御コード部の構成を示します。

```
.section      ".tw_profile_routine", text

.globl      tw_init_timer_timing_address
.globl      tw_clock_count
.globl      tw_init_timer
.globl      tw_start_timer
.globl      tw_stop_and_clear_timer
.globl      tw_get_timer_count

.option     new_fcall

.align      0x4
tw_init_timer_timing_address :
    .word    < タイマを初期化するアドレス >

    .align   0x4
tw_clock_count :
    .word    < タイマ 1 カウントあたりの CPU クロック数 >

tw_init_timer :
    < タイマの初期化処理コード >

tw_start_timer :
    < タイマのカウント開始処理コード >

tw_stop_and_clear_timer :
    < タイマのカウント停止とクリア処理コード >

tw_get_timer_count :
    < タイマのカウント値の取得処理コード >
```

なお、TW850 で指定できるタイマは、以下に示す条件を満たすタイマに限られます。

- カウント・アップが可能
- カウントをクリア (0 に) することが可能
- カウント・アップの開始 / 停止が可能

ただし、カウントをクリアすることができない、および、カウント・アップの停止ができないカウンタでも、カウント・アップの開始処理時にカウントをメモリに保存し、カウント値を取得時に現在のカウント値と保存したカウント値の差分を求めることにより、指定することが可能です。

以下に、タイマ制御コード部で使用しているラベルの一覧を示します。

- tw_init_timer_timing_address

タイマを初期化するアドレスを指定します。

データ・タイプ : 4 バイト整数形式

注意 1 このラベルを省略した場合、タイマの初期化位置は性能解析開始位置の直前となります。

注意 2 RX850, または, RX850 Pro を使用している場合、このラベルは無視されます。

注意 3 実行開始位置から解析開始位置の間までに必ず実行されるアドレスを指定してください。
また、実行開始位置、および、解析開始位置とは異なるアドレスを指定してください。

- tw_clock_count (必須)

タイマ 1 カウントあたりの CPU クロック数を指定します。

データ・タイプ : 4 バイト整数形式

- tw_init_timer (必須)

タイマの初期化処理コードを記述します。

入力レジスタ : lp(戻りアドレス)

出力レジスタ : なし

破壊可能レジスタ : r10, r11, r12

注意 1 アセンブラにおける命令展開で、r1 が破壊される場合があります。

注意 2 分周率を設定できるタイマの場合、tw_clock_count で指定する値と初期化時の分周率の設定に不整合がないようにしてください。

- tw_start_timer (必須)

タイマのカウント開始処理コードを記述します。

入力レジスタ : lp(戻りアドレス)

出力レジスタ : なし

破壊可能レジスタ : r10, r11

注意 アセンブラにおける命令展開で、r1 が破壊される場合があります。

- tw_stop_and_clear_timer (必須)

タイマのカウント停止とクリア処理コードを記述します。

入力レジスタ : lp(戻りアドレス)

出力レジスタ : なし

破壊可能レジスタ : r10, r11

注意 アセンブラにおける命令展開で、r1 が破壊される場合があります。

- tw_get_timer_count (必須)

タイマのカウント値の取得処理コードを記述します。

入力レジスタ : lp(戻りアドレス)

出力レジスタ : r10(カウント値)

破壊可能レジスタ : r11

- 注意 1 アセンブラにおける命令展開で、r1 が破壊される場合があります。
- 注意 2 出力レジスタに代入する値が負の数にならないようにしてください。
例えば、ターゲットが V850 で、タイマ・カウンタが 2 バイトの場合、ld.h 命令による符号拡張のため、ld.h 命令でレジスタにロードした後、andi 命令によりそのレジスタを 0xffff でマスクする必要があります。
- 注意 3 イベント（関数呼び出し、RTOS のシステム・コールなど）が発生せず、時間計測中にタイマがオーバーフロー、または、ラウンドするような場合、時間計測を正しく行うことができません。

4.4.2 タイマ制御コード部の検査

作成したタイマ制御コード部を検査用オブジェクトとリンクして実行モジュールを作成し、そのモジュールを実行することにより、タイマ制御コード部を検査することができます。

この検査用オブジェクトでは、以下に示した項目を検査することができます。

- タイマの不正動作（動作していない / 停止しない / クリアされない）
 - 各処理部（タイマの初期化 / カウント開始 / カウント停止とクリア / カウンタ値取得）における破壊可能レジスタ以外のレジスタ内容の破壊
- 注意 スタック・ポインタ、および、戻りアドレスを格納しているレジスタの破壊は検査しません。
- タイマ 1 カウントあたりの CPU クロック数の明らかな設定ミス

以下に示した手順に従い、タイマ制御コード部の検査を行います。

1) 検査用実行モジュールの作成

必要に応じて、リンク・ディレクティブ・ファイルの内容を変更し、タイマ制御コード部と検査用オブジェクト、および、スタート・アップ・ファイルをリンクし、実行モジュールを作成してください。

検査用実行モジュールの作成には、Program Files¥NEC Electronics Tools¥TW850¥バージョン¥smp850¥tw に格納されている以下に示したファイルが必要となります。

TWTMCHK85.O	: V850 用検査オブジェクト
TWTMCHK85E.O	: V850E 用検査オブジェクト
TWTMCHK85.DIR	: CA850 V850 用リンク・ディレクティブ・ファイルの内蔵 ROM / RAM 使用サンプル
TWTMCHK85E.DIR	: CA850 V850E 用リンク・ディレクティブ・ファイルの内蔵 ROM / RAM 使用サンプル
TWTMCHK85.MAK	: V850 用メイク・ファイルのサンプル
TWTMCHK85E.MAK	: V850E 用メイク・ファイルのサンプル
TWTMCHK85.PRJ	: V850 用 PM plus プロジェクト・ファイルのサンプル
TWTMCHK85E.PRJ	: V850E 用 PM plus プロジェクト・ファイルのサンプル

注意 1 スタート・アップ・ファイルをカスタマイズする場合、以下の処理をファイル中に記述する必要があります。

- スタック領域の 512 バイト以上の確保
- スタック・ポインタの初期化
- 環境設定、および、上記処理後の _main 関数の呼び出し

注意 2 PM plus プロジェクト・ファイルのサンプルを使用する場合、サンプルのフォルダをそのまま使用すると、ビルド時にメイク・ファイルのサンプルが上書きされてしまうため、必要なファイルは別のフォルダにコピーした上で使用してください。

注意 3 V850E 用 PM plus プロジェクト・ファイルのサンプルにおいて、ヘキサ・ファイルの形式として 32 ビット・モトローラ S タイプ・ヘキサ・フォーマットが指定されていますが、これは、デフォルトのインテル拡張ヘキサ・フォーマットの 16 ビットではアドレスが足りないというワーニング・メッセージの表示を避けるためです。

2) 検査用実行モジュールの実行

以下に示した手順で、作成した検査用実行モジュールを実行し、検査を行ってください。

- (1) デバッグ、シミュレータなどに検査用実行モジュールをダウンロードします。
- (2) ラベル “tw_timer_check” にブレーク・ポイントを設定します。
- (3) ラベル “tw_timer_check” まで検査用実行モジュールを実行します。
- (4) ラベル “tw_timer_check” で検査用実行モジュールの実行が停止していることを確認します。
- (5) レジスタ番号 1 のレジスタ (r1) の値を確認します。

レジスタ番号 1 のレジスタには、エラー・コードが格納されています。そのエラー・コードによって、タイマ制御コードの検査結果を確認することができます。

以下に、エラー・コードの一覧を示します。

- 0x00000000 : 正常終了。
- 0x000011XX : タイマの初期化処理内で、破壊可能レジスタ以外のレジスタ XX(XX はレジスタ番号) を破壊しています。
タイマの初期化処理内で該当レジスタを破壊しないように修正してください。
- 0x000012XX : タイマのカウント開始処理内で、破壊可能レジスタ以外のレジスタ XX(XX はレジスタ番号) を破壊しています。
タイマのカウント開始処理内で該当レジスタを破壊しないように修正してください。
- 0x000013XX : タイマのカウント停止とクリア処理内で、破壊可能レジスタ以外のレジスタ XX(XX はレジスタ番号) を破壊しています。
タイマのカウント停止とクリア処理内で該当レジスタを破壊しないように修正してください。
- 0x000014XX : タイマのカウント値の取得処理内で、破壊可能レジスタ以外のレジスタ XX(XX はレジスタ番号) を破壊しています。
タイマのカウント値の取得処理内で該当レジスタを破壊しないように修正してください。
- 0x00002201 : タイマが動作していません。
タイマの初期化処理、カウント開始処理、カウント値の取得処理の内容の確認、および、タイマ 1 カウントあたりの CPU クロック数が正常に設定されているかを確認してください。
- 0x00002301 : タイマが停止しません。
タイマのカウント停止とクリア処理内のカウント停止処理の内容を確認してください。
- 0x00002302 : タイマ・カウンタがクリアされません。
タイマのカウント停止とクリア処理内でクリア処理の内容を確認してください。
- 0x00003001 : タイマ 1 カウントあたりの CPU クロック数が 0 です。
タイマ 1 カウントあたりの CPU クロック数の設定値を確認してください。

(6) 正常終了しなかった場合は、該当箇所を修正し、上記手順を再度実行します。

4.4.3 時間計測処理における注意事項

時間計測処理について、以下に示したことに注意する必要があります。

- プローブ・コード挿入方式の場合、TW850 は、関数呼び出しの戻り部分に時間計測用プローブ・コードを挿入します。そのため、関数の呼び出し元に戻らない関数呼び出しを行っている場合、その呼び出し元と呼び出し先の関数の時間計測を正しく行うことができない場合があります。
例えば、C 言語ソース・ファイルにおいて、setjmp、および、longjmp が該当関数で、libtw を使用した場合でも、longjmp の実行時間は計測することができません。また、その時間は、原則として setjmp を呼び出している関数に加算されます。libtw を使用しない場合は、longjmp を呼び出している関数の実行時間の一部が setjmp を呼び出している関数の実行時間に加算されます。
- 時間計測は、関数の呼び出し、リターン、システム・コールの呼び出しフックなどの事象が発生するたびに、プロファイル・ルーチン呼び出すことにより行われます。そのため、関数の途中から解析を開始し、何の事象も発生せずに解析を終了するような場合、どの関数を実行していたのか判別できないため、その関数の実行時間は 0 とみなします。
- 実行停止時にタイマを停止しない実行エンジン (ICE サーバなど) を使用している場合、以下に示すことに注意する必要があります。
 - 解析対象に指定した関数、または、解析開始位置、および、解析終了位置が所属する関数については、実行時間を正しく計測することができません。このような場合は、その関数を呼び出している関数を解析対象に指定することにより、その関数の実行時間を正しく計測することができます。
 - 解析対象に指定した関数、または、解析開始位置から、アセンブリ言語ソース・ファイルにおいて、TW850 が認識できない関数呼び出し命令により他の関数を呼び出し、その呼び出された関数の中で何らかの事象が起きた場合、その呼び出された関数の実行時間を正しく計測することができません。
 - 解析対象に指定した関数、または、解析終了位置に、アセンブリ言語ソース・ファイルにおいて、TW850 が認識できない命令により復帰 (分岐) し、解析対象に指定した関数の終了、または、解析終了位置までに何の事象も起きない場合、その復帰元関数の実行時間を正しく計測することができません。

- タイマのカウント・アップのタイミングとプログラムの実行のタイミングがずれることにより、同一プログラムの同一解析区間でも、計測するたびに実行時間がずれる場合があります。

4.5 TW850 用システム・ライブラリ

TW850 ではユーザ・プログラムだけでなく標準ライブラリ関数 (libc.a , libm.a に含まれるもの) もチューニング時の再配置の対象にするため、libc.a , libm.a に代わるものとして libtw.a を提供しています。TW850 での実行モジュールの生成にはこの libtw.a が必要となります。

4.6 アセンブリ言語ソース・ファイルにおける注意事項

この節では、以下の項目について解説します。

- 性能解析対象関数，チューニング対象関数の検出
- 関数呼び出し箇所の検出
- アセンブリ言語ソース・ファイルの記述形式

4.6.1 性能解析対象関数，チューニング対象関数の検出

TW850 では、性能解析対象範囲，チューニング対象範囲の関数ラベル宣言において、以下に示した何れかの条件が成立したラベル宣言を性能解析対象関数，チューニング対象関数として認識します。

なお、TW850 が認識することのできなかった関数については、[関数情報テーブル・ウィンドウ](#)などで表示されません。

1) グローバル・ラベル宣言

TW850 では、性能解析対象範囲，チューニング対象範囲においてグローバル・ラベル宣言された関数を性能解析対象関数，チューニング対象関数として認識します。

```
.globl  _func          -- グローバル・ラベル宣言
_func:
```

2) 関数属性を有するラベル宣言

TW850 では、性能解析対象範囲，チューニング対象範囲において関数属性を有するラベル宣言された関数を性能解析対象関数，チューニング対象関数として認識します。

```
.frame  _func, 0x4      -- 関数属性を有するラベル宣言
_func:
```

ただし、上記条件（グローバル・ラベル宣言，関数属性を持つラベル宣言）が成立する場合であっても、該当ラベル宣言の直前命令を解析した結果によっては該当ラベル宣言された関数を性能解析対象関数，チューニング対象関数として認識しない場合があります。

1) ラベル宣言の直前命令（ラベル宣言）

TW850 では、ラベル宣言の直前命令がラベル宣言であった場合には、該当ラベル宣言された関数を性能解析対象関数，チューニング対象関数として認識しません。

```
.text
.globl  _func1
_func1:

.globl  _func2
_func2:          -- 対象関数として認識されない
```

このような場合、関数中のラベルをグローバル・ラベルからローカル・ラベルに変更し、関数属性を設定しないようにすることで、関数ラベルとして検出しないようにすることが可能です。

以下に示す例においては、戻りラベルのグローバル宣言の行 (*) を削除してください。

```

        .text
        .globl  _func1
_func1 :
        mov     #_func2, r15
        mov     #_ret_label, lp
        jmp     [r15]

        .globl  _ret_label          -- 戻りラベルのグローバル宣言 (*)
_ret_label:                          -- 戻りラベル
        jmp     [lp]

        .globl  _func2
_func2 :                              -- 対象関数として認識されない
        jmp     [lp]

```

2) ラベル宣言の直前命令 (疑似命令, 無条件分岐命令以外)

TW850 では, ラベル宣言の直前命令が疑似命令, または, 無条件分岐命令でなかった場合には, 該当ラベル宣言された関数を性能解析対象関数, チューニング対象関数として認識しません。

```

        .text
        .globl  _func1
_func1 :
        nop

        .globl  _func2
_func2 :                              -- 対象関数として認識されない

```

4.6.2 関数呼び出し箇所の検出

TW850 では、性能解析対象範囲、チューニング対象範囲の関数呼び出しにおいて、以下に示した何れかの条件が成立した関数呼び出しを関数呼び出し箇所として認識します。

なお、TW850 が認識することのできなかった関数呼び出しについては、[コールグラフ・ウィンドウ](#)などで表示されません。

1) 分岐命令 (jarl) の発行

TW850 では、性能解析対象範囲、チューニング対象範囲において分岐命令 jarl の発行箇所を関数呼び出し箇所として認識します。

```
.globl  _func1
_func1 :
    mov    lp, r14
    jarl    _func2, lp          -- 分岐命令 jarl の発行
```

2) 命令パターン (算術演算命令 movea , 分岐命令 jmp) の一致

TW850 では、性能解析対象範囲、チューニング対象範囲において算術演算命令 movea , 分岐命令 jmp の直後にラベル宣言された箇所を関数呼び出し箇所として認識します。

```
.globl  _func
_func :
    movea   xxx, xxx, r31      -- 算術演算命令 movea の発行
    jmp     [ REGISTER ]      -- 分岐命令 jmp の発行

_label1 :                     -- ラベル宣言
    movea   xxx, xxx, lp      -- 算術演算命令 movea の発行
    jmp     [ REGISTER ]      -- 分岐命令 jmp の発行

_label2 :                     -- ラベル宣言
```

注意 REGISTER は、r31, lp 以外のレジスタに限られます。

3) 命令パターン (算術演算命令 mov , 分岐命令 jmp) の一致

TW850 では、性能解析対象範囲、チューニング対象範囲において算術演算命令 mov , 分岐命令 jmp の直後にラベル宣言された箇所を関数呼び出し箇所として認識します。

```
.globl  _func
_func :
    mov     xxx, r31          -- 算術演算命令 mov の発行
    jmp     [ REGISTER ]      -- 分岐命令 jmp の発行

_label1 :                     -- ラベル宣言
    mov     xxx, lp          -- 算術演算命令 mov の発行
    jmp     [ REGISTER ]      -- 分岐命令 jmp の発行

_label2 :                     -- ラベル宣言
```

注意 REGISTER は、r31, lp 以外のレジスタに限られます。

TW850 では、性能解析対象範囲、チューニング対象範囲において算術演算命令 mov , 分岐命令 jmp の直後がラベル宣言でなかった場合には、該当箇所を関数呼び出し箇所として認識しません。

```
.text
    .globl    _func1
_func1 :
    mov     lp, r14
    jarl    _func2, lp
    mov     #_ret_label, lp
    mov     #_func2, r15
    jmp     [r15]                -- 分岐命令 jmp の発行
_ret_label:
    mov     #_func_end - 4, lp
    jmp     [r15]                -- 分岐命令 jmp が発行されない
    mov     lp, r14
    jmp     [lp]
_func_end:

    .globl    _func2
_func2:
    jmp     [lp]
```

4.6.3 アセンブリ言語ソース・ファイルの記述形式

TW850 では、特定の記述形式で記述されたアセンブリ言語ソース・ファイルを性能解析対象、チューニング対象とすることができません。

表 4-1 に、TW850 が特定の記述形式で記述されたアセンブリ言語ソース・ファイルを性能解析対象 (トレース・データ解析方式、プローブ・コード挿入方式)、チューニング対象とすることができるか否かを示します。

表 4-1 アセンブリ言語ソース・ファイルの制限事項

記述形式	トレース	プローブ	チューニング
オフセットで関数ラベルの上位アドレスをアクセス	○	×	×
条件が変化するアセンブル疑似命令	×	×	×
対象範囲外の仮パラメータ	×	×	×
定義済みのマクロ呼び出し	×	×	×
32 段階を超えるマクロ定義の多重展開	×	×	×
疑似命令内の数値表現、演算結果が有効範囲外	-	-	-
他ブロックにまたがるブロック宣言	×	×	×
関数の境界	-	-	-
関数内ラベルを参照する関数外への分岐	○	×	×
関数の入り口が 2 つ以上ある場合の後宣言関数ラベルの認識	×	×	×
関数呼び出し命令をまたぐオフセット分岐命令	○	×	○
セクションをまたぐことのできない命令における関数外ラベルの参照 / 分岐	○	○	×
異なる関数内のラベルを用いたアドレス計算	○	○	×
スタック・ポインタの未初期化	○	×	○
割り込み発生時に退避対象となるレジスタの使用	○	×	○
想定されている用途以外の I/O レジスタの使用	×	×	×
ラベル宣言、条件アセンブル疑似命令、シンボル定義疑似命令に対する連結演算子	×	×	×
システム予約マクロの使用	×	×	×
関数配置順序、関数配置アドレスの指定	○	×	×
関数間をまたがる PSW(プログラム・ステータス・ワード)の参照	○	×	○
\$ マクロ・オペレータの利用	×	×	×

注意 関数とは、「4.6.1 性能解析対象関数、チューニング対象関数の検出」で示した条件に合う関数を指しています。
また、関数呼び出しとは、「4.6.2 関数呼び出し箇所の検出」で示した条件に合う関数呼び出しを指しています。

以下に、表 4-1 の各記述形式について示します。

1) オフセットで関数ラベルの上位アドレスをアクセス

オフセットで関数ラベルの上位アドレス (0x0 に近いアドレス) をアクセスする記述が行われているアセンブリ言語ソース・ファイルに対して、TW850 は性能解析処理 (プローブ・コード挿入方式)、および、チューニング処理を実行することができません。

```
.text
.word    0x0
.globl   _func
_func :
mov      #_func, r12
ld.w     -4 [ r12 ], r13
```

-- オフセットで関数ラベルの上位アドレスをアクセス

2) 条件が変化するアセンブル疑似命令

繰り返し疑似命令の対象範囲 (.repeat ~ .endm, .rept ~ .endr) において、繰り返すことによって条件が変化するアセンブル疑似命令 (.if, .elseif など) が記述されたアセンブリ言語ソース・ファイルに対して、TW850 は性能解析処理 (トレース・データ解析方式、プローブ・コード挿入方式)、および、チューニング処理を実行することができません。

```
.set     VAR, 0x0

.repeat  0x10
.if      ( VAR == 0x5 )
nop
.endif

.set     VAR, VAR + 0x1
.endm
```

-- 条件が変化するアセンブル疑似命令

3) 対象範囲外の仮パラメータ

仮パラメータを有するマクロ定義疑似命令、および、繰り返し疑似命令の対象範囲外 (.macro ~ .endm の外, .repeat ~ .endm の外) において、仮パラメータが記述されたアセンブリ言語ソース・ファイルに対して、TW850 は性能解析処理 (トレース・データ解析方式、プローブ・コード挿入方式)、および、チューニング処理を実行することができません。

```
.macro   MCR1  ARG
mov      ARG, r12
.endm

MCR1     0x1

.if      ARG
nop
.endif
```

-- 対象範囲外の仮パラメータ

4) 定義済みのマクロ呼び出し

マクロ定義疑似命令の対象範囲 (.macro ~ .endm) において、定義済みマクロと同名のマクロ呼び出しが記述されたアセンブリ言語ソース・ファイルに対して、TW850 は性能解析処理（トレース・データ解析方式、プローブ・コード挿入方式）、および、チューニング処理を実行することができません。

```
.macro MCR1
    .macro MCR2
        nop
    .endm

    MCR2
    nop
.endm

.macro MCR2
    MCR1
    nop
.endm

MCR2
```

-- 定義済みのマクロ呼び出し

5) 32 段階を超えるマクロ定義の多重展開

32 段階を超えるマクロ定義の多重展開が行われるアセンブリ言語ソース・ファイルに対して、TW850 は性能解析処理（トレース・データ解析方式、プローブ・コード挿入方式）、および、チューニング処理を実行することができません。

```
.macro MCR33
    nop
.endm
.macro MCR32
    MCR33
.endm
.macro MCR31
    MCR32
.endm

.....
.....
.....

.macro MCR3
    MCR4
.endm
.macro MCR2
    MCR3
.endm
.macro MCR1
    MCR2
.endm

MCR1
```

-- 32 段階を超えるマクロの多重展開

6) 疑似命令内の数値表現, 演算結果が有効範囲外

疑似命令内の数値表現, 演算結果が有効範囲外となるアセンブリ言語ソース・ファイルの場合, TW850 は有効範囲外の部分を無視 (数値表現: 有効ビット以外を切り捨て, 演算結果: 有効ビット以外をラウンド) して性能解析処理 (トレース・データ解析方式, プローブ・コード挿入方式), および, チューニング処理を行います。

```
.if      ( 0xffffffffffff == 0xffffffff )      -- 疑似命令内の数値表現が有効範囲外
    nop
.endif
```

7) 他ブロックにまたがるブロック宣言

ブロック宣言 (条件アセンブル・ブロック宣言, 繰り返し定義ブロック宣言, マクロ定義ブロック宣言など) において, 指定されたブロックが他ブロックをまたぐアセンブリ言語ソース・ファイルに対して, TW850 は性能解析処理 (トレース・データ解析方式, プローブ・コード挿入方式), および, チューニング処理を実行することができません。

```
.set      EVAL, 0x1

.if      EVAL
    .repeat EVAL                      -- 他ブロックにまたがる繰り返し定義ブロック宣言
.endif

nop

.if      EVAL
    .endm
.endif
```

8) 関数の境界

アライン指定疑似命令, または, シンボル属性設定疑似命令がラベル宣言 (関数シンボル) の直前に記述されたアセンブリ言語ソース・ファイルの場合, TW850 は該当疑似命令をラベル宣言された関数に含まれるものとして性能解析処理 (トレース・データ解析方式, プローブ・コード挿入方式), および, チューニング処理を行います。

```
        .align    0x4                -- 関数の境界
        .globl    _func
_func :
        nop
```

9) 関数内ラベルを参照する関数外への分岐

関数内ラベルを参照して関数外への分岐を行う処理が記述されているアセンブリ言語ソース・ファイルに対して, TW850 は性能解析処理 (プローブ・コード挿入方式), および, チューニング処理を実行することができません。

```
_func1 : .globl    _func1
        jr      _func1 + 0x6          -- 関数内ラベルを参照する関数外への分岐
        jmp     [lp]

_func2 : .globl    _func2
        jmp     [lp]
```

10) 関数の入り口が 2 つ以上ある場合の後宣言関数ラベルの認識

関数の入り口（関数ラベル）が 2 つ以上あるアセンブリ言語ソース・ファイルの場合，TW850 は最初に宣言された関数レベルのみを認識して性能解析処理（トレース・データ解析方式，ブローブ・コード挿入方式），および，チューニング処理を行います。

```

        .globl  _func1
_func1:
        nop

        .globl  _func2
_func2:                                -- 認識されない後宣言関数ラベル
        jmp    [lp]
```

11) 関数呼び出し命令をまたぐオフセット分岐命令

関数呼び出し命令をまたぐオフセット分岐命令（参照ラベル・アドレスと分岐先アドレスとの間に関数呼び出し命令を含む）が記述されているアセンブリ言語ソース・ファイルに対して，TW850 は性能解析処理（ブローブ・コード挿入方式）を実行することができません。

```

        .globl  _func1
_func1:
        br     _func1 + 0x6           -- 関数呼び出し命令をまたぐオフセット分岐命令
        jarl   _func2, r11
        jmp    [lp]

        .globl  _func2
_func2:
        jmp    [r11]
```

12) セクションをまたぐことのできない命令における関数外ラベルの参照 / 分岐

セクションをまたぐことのできない命令において，関数外ラベルの参照 / 分岐を用いたアセンブリ言語ソース・ファイルに対して，TW850 はチューニング処理を実行することができません。

```

        .globl  _func1
_func1:
        jbr    #_func2                -- 関数外ラベルへの分岐
        jmp    [lp]

        .globl  _func2
_func2:
        jmp    [lp]
```

13) 異なる関数内のラベルを用いたアドレス計算

異なる関数内のラベルを用いたアドレス計算が記述されているアセンブリ言語ソース・ファイルに対して、TW850 はチューニング処理を実行することができません。

```
.globl  _func1
_func1:
    mov    #data - #_func2, r11    -- 異なる関数内のラベルを用いたアドレス計算
    jmp    [lp]

.globl  _func2
_func2:
    jmp    [lp]

data:
    .word  0x0
```

14) スタック・ポインタの未初期化

スタック・ポインタを初期化する前に、TW850 が関数呼び出しと認識する場所が実行されるようなアセンブリ言語ソース・ファイルに対して、TW850 は性能解析処理 (プローブ・コード挿入方式) を実行することができません。

15) 割り込み発生時に退避対象となるレジスタの使用

割り込み発生時にプロファイル・ルーチンが使用するレジスタ r10 ~ r14 を退避することなく起動された割り込み処理専用ルーチンに対して、TW850 は性能解析処理 (プローブ・コード挿入方式) を実行することができません。

16) 想定されている用途以外の I/O レジスタの使用

TW850 では、デバイス・ファイルに定義されている I/O レジスタ用マクロの使用条件を ld/st 命令、および、ビット操作命令のオペランド部に限定しています。

したがって、I/O レジスタ用マクロを上記以外の用途に使用したアセンブリ言語ソース・ファイルに対して、TW850 は性能解析処理 (トレース・データ解析方式、プローブ・コード挿入方式)、および、チューニング処理を実行することができません。

17) ラベル宣言、条件アセンブル疑似命令、シンボル定義疑似命令に対する連結演算子

ラベル宣言、条件アセンブル疑似命令、シンボル定義疑似命令に連結演算子を用いたアセンブリ言語ソース・ファイルに対して、TW850 は性能解析処理 (トレース・データ解析方式、プローブ・コード挿入方式)、および、チューニング処理を実行することができません。

```
.macro  labeldef ARG
    .globl  _func~ARG
    _func~ARG:
    -- ラベル宣言に対する連結演算子
.endm
```

18) システム予約マクロの使用

TW850 では、“tw_” で始まる単語、および、“__e_sysfnc” (RX850/RX850 Pro 使用時) をシステム予約マクロとして扱います。

したがって、該当単語を用いたアセンブリ言語ソース・ファイルに対して、TW850 は性能解析処理 (トレース・データ解析方式、プローブ・コード挿入方式)、および、チューニング処理を実行することができません。

また、“_AZMON_BaseTable” (RX850/RX850 Pro 使用時) をシステム予約マクロとして扱います。

したがって、該当単語が存在するプロジェクトに対して、TW850 を実行することができません。

```
.globl  tw_func    -- システム予約マクロの使用
tw_func:
    jmp    [lp]
```

19) 関数配置順序, 関数配置アドレスの指定

関数の配置順序, または, 関数の配置アドレスが変更された場合に動作を保証することができないアセンブリ言語ソース・ファイルに対して, TW850 は性能解析処理 (プローブ・コード挿入方式), および, チューニング処理を実行することができません。

```

        .globl  _func1
_func1:
        mov     #_func, r11          -- 関数配置アドレスの指定
        cmp     0x10000, r11
        je      _func2
        halt

_func2:
        jmp     [lp]

```

20) 関数間をまたがる PSW(プログラム・ステータス・ワード)の参照

関数呼び出し, または, 関数呼び出しからの復帰時に, 関数呼び出し先と関数呼び出し元をまたいでの PSW の参照をするようなアセンブリ言語ソース・ファイルに対して, TW850 は性能解析処理 (プローブ・コード挿入方式), および, チューニング処理を実行することができません。

```

        .globl  _func1
_func1:
        cmp     r11, r12             -- 比較 (PSW の設定)
        jarl    _func2, r13
        jmp     [lp]

        .globl  _func2
_func2:
        bne     _func2_ret           -- 関数呼び出し後に比較結果 (PSW) を参照
        mov     r0, r11

_func2_ret:
        jmp     [r13]

```

21) \$ マクロ・オペレータの利用

\$ マクロ・オペレータを利用したマクロを使用しているアセンブリ言語ソース・ファイルに対して, TW850 は性能解析処理 (トレース・データ解析方式, プローブ・コード挿入方式), および, チューニング処理を実行することができません。

```

        .macro   mac1 x
            mov    x, r10
        .endm

        .macro   mac2
            .set    value, 10
            mac1    $value      -- $value を指定
        .endm
        mac2

```

第 5 章 操作方法

本章では，TW850 の操作方法，および，性能解析結果確認方法について解説しています。

図 5-1 に，TW850 が実行するウィザード処理（性能解析処理，チューニング処理）の流れを示します。

図 5-1 ウィザード処理（性能解析処理，チューニング処理）の流れ

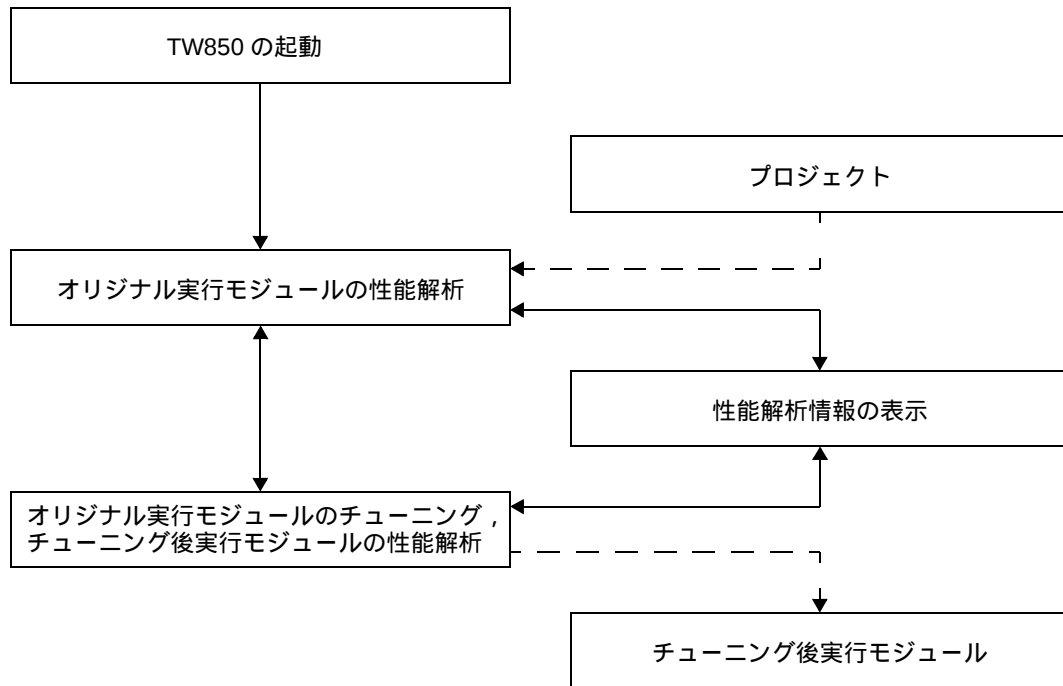
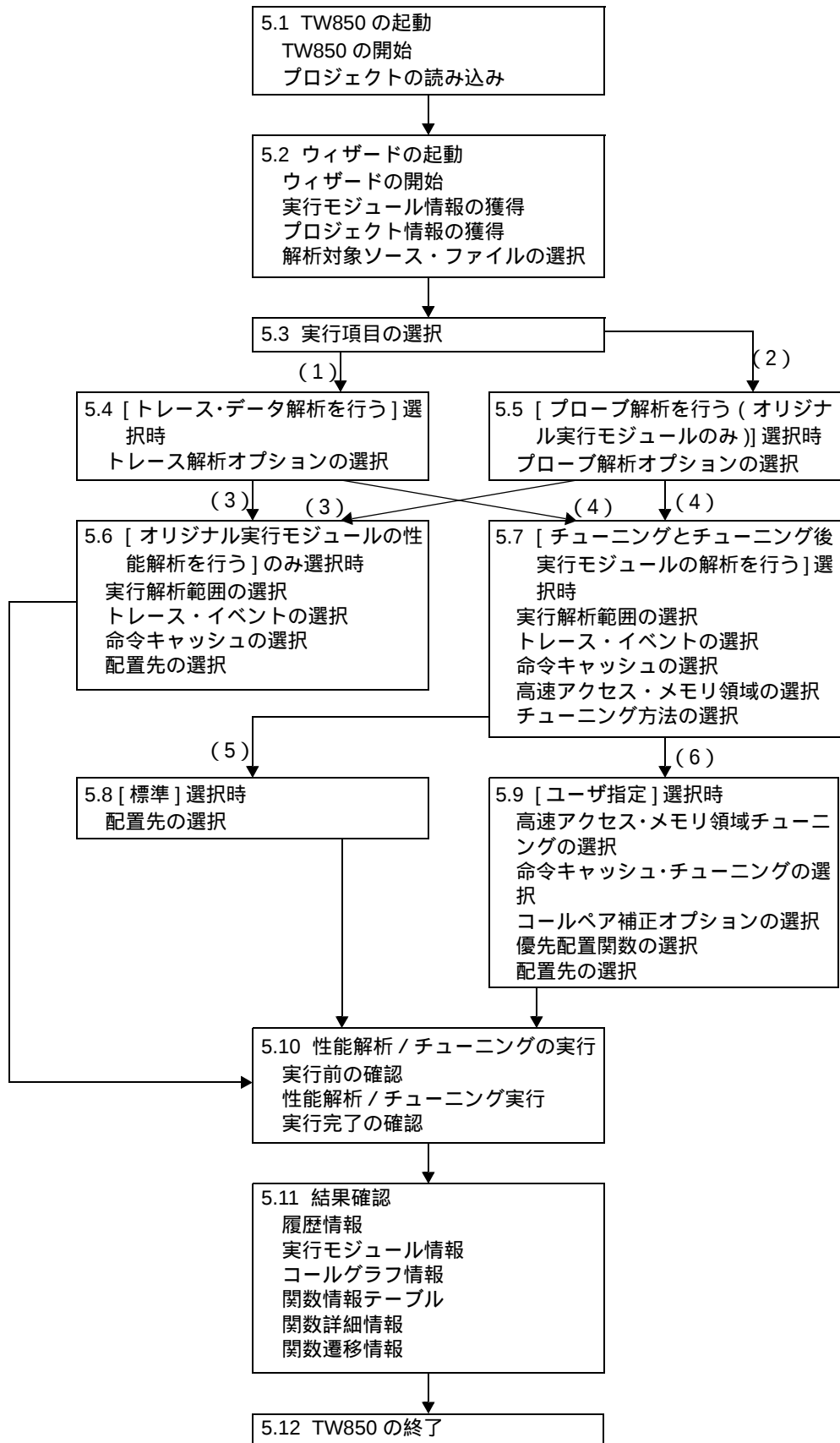


図 5-2 に、TW850 の操作手順、および、性能解析結果確認手順を示します。

図 5-2 TW850 の操作手順 / 性能解析結果確認手順



- (1) 実行項目の選択で [トレース・データ解析を行う (I)] ラジオ・ボタンをチェック状態にした場合
- (2) 実行項目の選択で [プローブ解析を行う (オリジナル実行モジュールのみ) (P)] ラジオ・ボタンをチェック状態にした場合
- (3) 実行項目の選択で [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスのみをチェック状態にした場合
- (4) 実行項目の選択で [チューニングとチューニング後実行モジュールの解析を行う (Q)] チェック・ボックスをチェック状態にした場合
- (5) チューニング方法の選択で [標準 (D)] ラジオ・ボタンをチェック状態にした場合
- (6) チューニング方法の選択で [ユーザ指定 (U)] ラジオ・ボタンをチェック状態にした場合

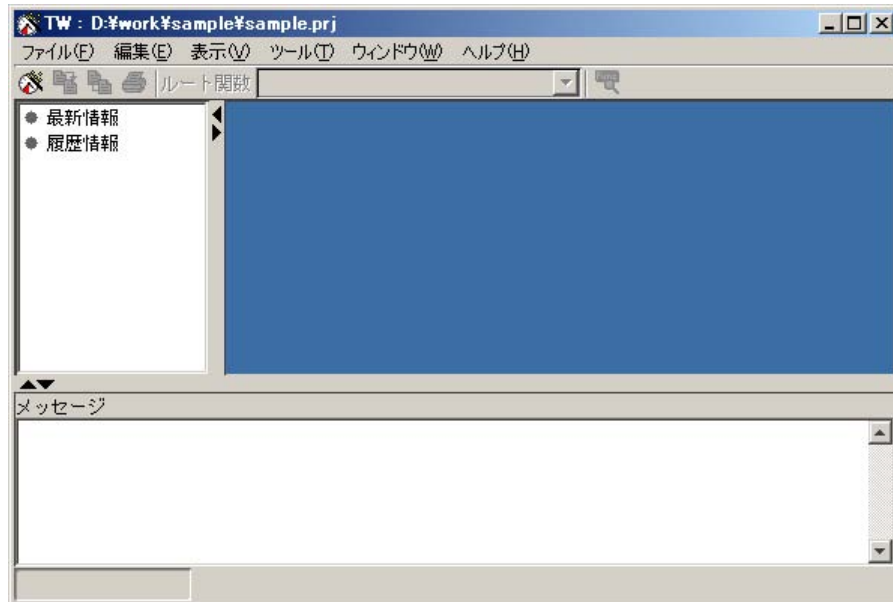
5.1 TW850 の起動

5.1.1 TW850 の開始

TW850 の起動方法についての詳細は、「[3.1 起動方法](#)」を参照してください。

なお、TW850 が正常に起動した際には、以下に示した[メイン・ウィンドウ](#)が自動的にオープンします。

図 5-3 メイン・ウィンドウ



5.1.2 プロジェクトの読み込み

TW850 は、PM plus においてアクティブなプロジェクト (プロジェクト・ファイル) を性能解析処理 / チューニング処理の対象とするプロジェクトとして認識します。

なお、プロジェクトの切り替え時、または、TW850 の起動時、対象となるプロジェクトの TW850 用プロジェクト・ファイルが存在する場合は、そのファイルを読み込み、TW850 の各種情報 (実行モジュール情報など) を復元します。

注意 使用中の TW850 と異なるバージョンの TW850 用プロジェクト・ファイルを読み込んだ場合は、[確認ダイアログ](#)がオープンし、そのファイルをバックアップ・ファイルとして保存するか否かを指定することができます。ただし、新しいバージョンの TW850 用プロジェクト・ファイルは、古いバージョンの TW850 で読み込むことはできません。

5.2 ウィザードの起動

5.2.1 ウィザードの開始

メイン・ウィンドウにおいて、チューニング・ウィザードボタンを押下、または、[ツール (T)] メニュー → [チューニング ウィザード (W)...] を選択します。

TW850 のウィザードが正常に起動した際には、以下に示した初期画面パネルが自動的にオープンします。

図 5-4 初期画面パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，メッセージ表示領域に表示されている“TW850のウィザード処理（性能解析処理，チューニング処理）が開始されたことを通知するためのメッセージ”を確認し，各種項目を指定した後，<次へ(N)> ボタンを押下します。

なお，各種項目の詳細については，ウィンドウ・レファレンスの初期画面パネルを参照してください。

5.2.2 プロジェクト情報の獲得

プロジェクト・ファイルを読み込んで、プロジェクト情報（ソース・ファイル構成など）を獲得します。
ただし、このフェーズは、「5.2.1 ウィザードの開始」において初期画面パネルの<次へ (N)> ボタンを押下した際、TW850 が自動的に行う処理であるため、ユーザが意識する必要はありません。
以下に、プロジェクト情報の獲得処理が行われている最中にオープンされるプロジェクト情報獲得パネルを示します。

図 5-5 プロジェクト情報獲得パネル

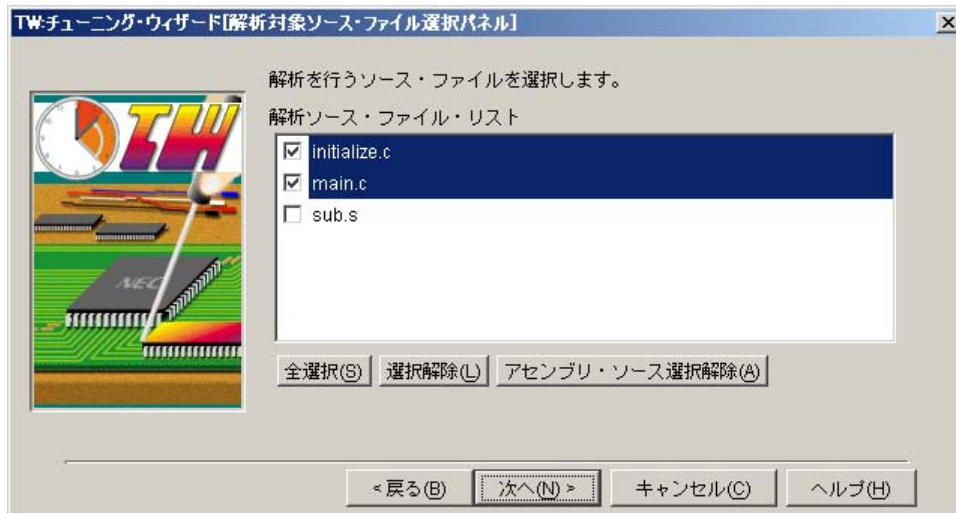


5.2.3 解析対象ソース・ファイルの選択

ウィザード処理（性能解析処理，チューニング処理）において，性能解析／チューニング対象とするソース・ファイルを指定します。

以下に示した[解析対象ソース・ファイル選択パネル](#)は，「[5.2.2 プロジェクト情報の獲得](#)」において行われるプロジェクト情報の獲得処理が正常終了した際，自動的にオープンします。

図 5-6 解析ソース・ファイル選択パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，各種項目を指定した後，<次へ (N)> ボタンを押下します。

なお，各種項目の詳細については，ウィンドウ・レファレンスの[解析対象ソース・ファイル選択パネル](#)を参照してください。

5.2.4 実行モジュール情報の獲得

TW850 が性能解析処理を実行するうえで必要となる実行モジュール情報を「5.2.1 ウィザードの開始」において読み込んだプロジェクト(プロジェクト・ファイル)に記載されているユーザ・プログラム、実行モジュール(オリジナル解析用実行モジュール)から獲得します。

ただし、このフェーズは、「5.2.3 解析対象ソース・ファイルの選択」において解析対象ソース・ファイル選択パネルの<次へ(N)>ボタンを押下した際、TW850 が自動的に行う処理であるため、ユーザが意識する必要はありません。

以下に、実行モジュール情報の獲得処理が行われている最中にオープンされる**実行モジュール情報獲得パネル**を示します。

図 5-7 実行モジュール情報獲得パネル

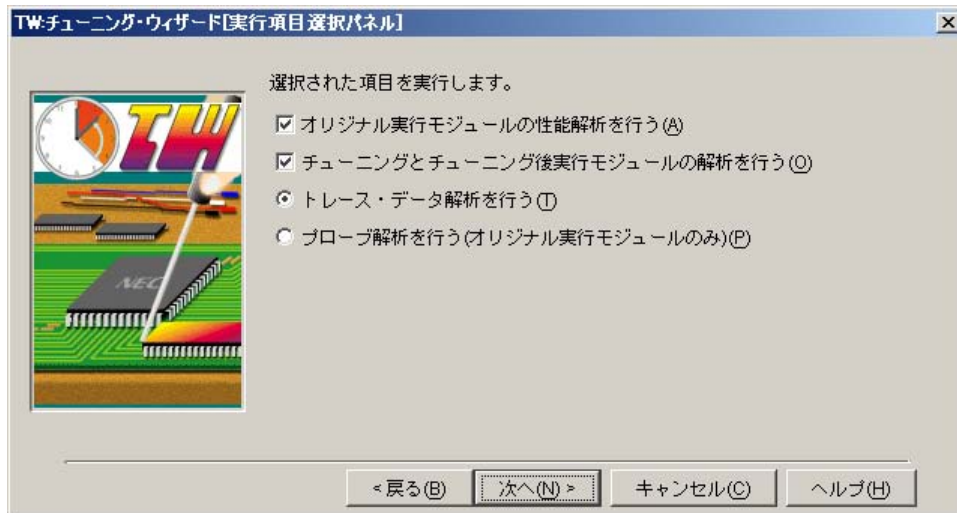


5.3 実行項目の選択

ウィザード処理（性能解析処理，チューニング処理）の実行項目を指定します。

以下に示した[実行項目選択パネル](#)は、「[5.2.4 実行モジュール情報の獲得](#)」において行われる実行モジュール情報の獲得処理が正常終了した際，自動的にオープンします。

図 5-8 実行項目選択パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，各種項目を指定した後，<次へ (N)> ボタンを押下します。

なお，各種項目の詳細については，ウィンドウ・レファレンスの[実行項目選択パネル](#)を参照してください。

5.4 [トレース・データ解析を行う (I)] 選択時

「5.3 実行項目の選択」において [トレース・データ解析を行う (I)] ラジオ・ボタンをチェック状態にした場合の操作手順を以下に示します。

なお、「5.3 実行項目の選択」において [プローブ解析を行う (オリジナル実行モジュールのみ) (P)] ラジオ・ボタンをチェック状態にした場合の操作手順については「5.5 [プローブ解析を行う (オリジナル実行モジュールのみ) (P)] 選択時」を参照してください。

5.4.1 トレース解析オプションの選択

ウィザード処理 (性能解析処理) において、トレース・データ解析方式を用いた性能解析情報の獲得オプションを指定します。

以下に示した **トレース解析オプション選択パネル** は、「5.3 実行項目の選択」において各種項目を指定した後、<次へ (N)> ボタンを押下した際にオープンします。

図 5-9 トレース解析オプション選択パネル



ウィザード処理 (性能解析処理) を継続する場合、各種項目を指定した後、<次へ (N)> ボタンを押下します。

なお、各種項目の詳細については、ウィンドウ・レファレンスの **トレース解析オプション選択パネル** を参照してください。

5.5 [プローブ解析を行う (オリジナル実行モジュールのみ) (P)] 選択時

「5.3 実行項目の選択」において [プローブ解析を行う (オリジナル実行モジュールのみ) (P)] ラジオ・ボタンをチェック状態にした場合の操作手順を以下に示します。

なお、「5.3 実行項目の選択」において [トレース・データ解析を行う (I)] ラジオ・ボタンをチェック状態にした場合の操作手順については「5.4 [トレース・データ解析を行う (T)] 選択時」を参照してください。

5.5.1 プローブ解析オプションの選択

ウィザード処理 (性能解析処理) において、プローブ・コード挿入方式を用いた性能解析情報の獲得オプションを指定します。

以下に示した **プローブ解析オプション選択パネル** は、「5.3 実行項目の選択」において各種項目を指定した後、<次へ (N)> ボタンを押下した際にオープンします。

図 5-10 プローブ解析オプション選択パネル



ウィザード処理 (性能解析処理) を継続する場合、各種項目を指定した後、<次へ (N)> ボタンを押下します。

なお、各種項目の詳細については、ウィンドウ・レファレンスの **プローブ解析オプション選択パネル** を参照してください。

5.6 [オリジナル実行モジュールの性能解析を行う (A)] のみ選択時

「5.3 実行項目の選択」において [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスのみをチェック状態にした場合の操作手順を以下に示します。

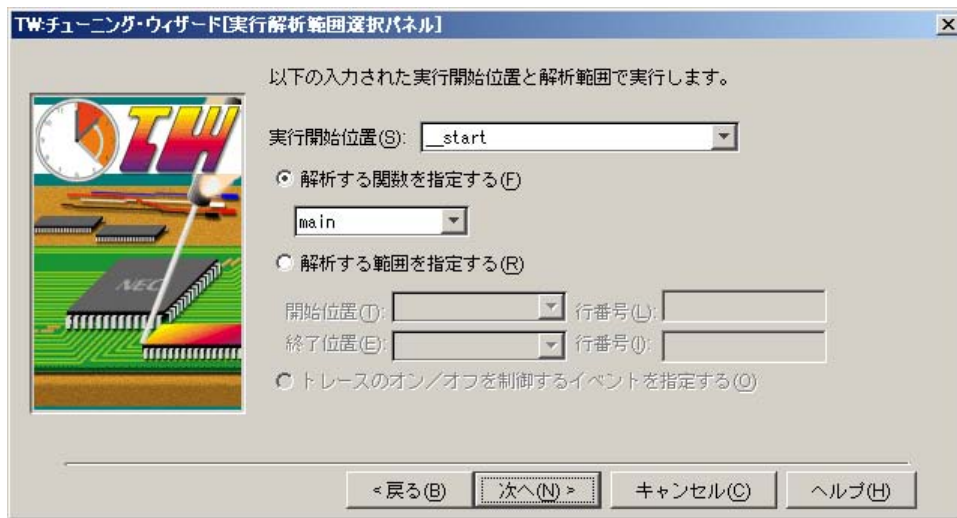
なお、「5.3 実行項目の選択」において [チューニングとチューニング後実行モジュールの解析を行う (Q)] チェック・ボックスをチェック状態にした場合の操作手順については「5.7 [チューニングとチューニング後実行モジュールの解析を行う (Q)] 選択時」を参照してください。

5.6.1 実行解析範囲の選択

ユーザ・プログラム、実行モジュールの実行開始位置、および、ウィザード処理 (性能解析処理) における性能解析範囲を指定します。

以下に示した**実行解析範囲選択パネル**は、「5.3 実行項目の選択」において各種項目を指定した後、<次へ (N)> ボタンを押下した際にオープンします。

図 5-11 実行解析範囲選択パネル



ウィザード処理 (性能解析処理) を継続する場合、各種項目を指定した後、<次へ (N)> ボタンを押下します。

なお、各種項目の詳細については、ウィンドウ・レファレンスの**実行解析範囲選択パネル**を参照してください。

5.6.2 トレース・イベントの選択

ウィザード処理 (性能解析処理) における性能解析範囲をトレース・イベントにより指定します。

以下に示した [トレース・イベント選択パネル](#) は、「[5.6.1 実行解析範囲の選択](#)」において、[トレースのオン/オフを制御するイベントを指定する (Q)] ラジオ・ボタンを選択した後、< 次へ (N) > ボタンを押下した際にオープンします。

注意 本パネルは、以下に示した条件が成立した場合はオープンされません。

- デバッグに ID850NW を使用していない

図 5-12 トレース・イベント選択パネル



ウィザード処理 (性能解析処理) を継続する場合、各種項目を指定した後、< 次へ (N) > ボタンを押下します。

なお、各種項目の詳細については、ウィンドウ・レファレンスの [トレース・イベント選択パネル](#) を参照してください。

5.6.3 命令キャッシュの選択

ウィザード処理 (性能解析処理) において、使用する命令キャッシュの種別、および、パラメータを指定します。

以下に示した**命令キャッシュ選択パネル**は、「5.6.1 実行解析範囲の選択」において、[トレースのオン/オフを制御するイベントを指定する (Q)] ラジオ・ボタンを非チェック状態とし、各種項目を指定した後、<次へ (N)> ボタンを押下した際、または、「5.6.2 トレース・イベントの選択」において各種項目を指定した後、<次へ (N)> ボタンを押下した際にオープンします。

図 5-13 命令キャッシュ選択パネル



ウィザード処理 (性能解析処理) を継続する場合、各種項目を指定した後、<次へ (N)> ボタンを押下します。

なお、各種項目の詳細については、ウィンドウ・レファレンスの**命令キャッシュ選択パネル**を参照してください。

5.6.4 配置先の選択

性能解析時に中間ファイルとして出力されるプロファイル・ルーチン、プロファイル・テーブルの配置用メモリ領域を指定します。

以下に示した配置先選択パネルは、「5.6.3 命令キャッシュの選択」において各種項目を指定した後、<次へ(N)> ボタンを押下した際にオープンします。

図 5-14 配置先選択パネル



ウィザード処理（性能解析処理）を継続する場合、各種項目を指定した後、<次へ(N)> ボタンを押下します。

なお、各種項目の詳細については、ウィンドウ・レファレンスの配置先選択パネルを参照してください。

5.7 [チューニングとチューニング後実行モジュールの解析を行う (Q)] 選択時

「5.3 実行項目の選択」において [チューニングとチューニング後実行モジュールの解析を行う (Q)] チェック・ボックスをチェック状態にした場合の操作手順を以下に示します。

なお、「5.3 実行項目の選択」において [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスのみをチェック状態にした場合の操作手順については「5.6 [オリジナル実行モジュールの性能解析を行う (A)] のみ選択時」を参照してください。

5.7.1 実行解析範囲の選択

ユーザ・プログラム、実行モジュールの実行開始位置、および、ウィザード処理 (性能解析処理) における性能解析範囲を指定します。

以下に示した**実行解析範囲選択パネル**は、「5.3 実行項目の選択」において各種項目を指定した後、< 次へ (N) > ボタンを押下した際にオープンします。

図 5-15 実行解析範囲選択パネル



ウィザード処理 (性能解析処理) を継続する場合、各種項目を指定した後、< 次へ (N) > ボタンを押下します。

なお、各種項目の詳細については、ウィンドウ・レファレンスの**実行解析範囲選択パネル**を参照してください。

5.7.2 トレース・イベントの選択

ウィザード処理 (性能解析処理) における性能解析範囲をトレース・イベントにより指定します。

以下に示した [トレース・イベント選択パネル](#) は、「[5.7.1 実行解析範囲の選択](#)」において、[トレースのオン/オフを制御するイベントを指定する (Q)] ラジオ・ボタンを選択した後、< 次へ (N) > ボタンを押下した際にオープンします。

注意 本パネルは、以下に示した条件が成立した場合はオープンされません。

- デバッグに ID850NW を使用していない

図 5-16 トレース・イベント選択パネル



ウィザード処理 (性能解析処理) を継続する場合、各種項目を指定した後、< 次へ (N) > ボタンを押下します。

なお、各種項目の詳細については、ウィンドウ・レファレンスの [トレース・イベント選択パネル](#) を参照してください。

5.7.3 命令キャッシュの選択

ウィザード処理（性能解析処理，チューニング処理）において，使用する命令キャッシュの種別，および，パラメータを指定します。

以下に示した**命令キャッシュ選択パネル**は，「5.3 実行項目の選択」において各種項目を指定した後，＜次へ (N)> ボタンを押下した際にオープンします。

図 5-17 命令キャッシュ選択パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，各種項目を指定した後，＜次へ (N)> ボタンを押下します。

なお，各種項目の詳細については，ウィンドウ・レファレンスの**命令キャッシュ選択パネル**を参照してください。

5.7.4 高速アクセス・メモリ領域の選択

ウィザード処理（チューニング処理）において、使用する高速アクセス・メモリ領域の種別、および、パラメータを指定します。

以下に示した[高速アクセス・メモリ領域選択パネル](#)は、「[5.7.3 命令キャッシュの選択](#)」において各種項目を指定した後、<次へ (N)> ボタンを押下した際にオープンします。

図 5-18 高速アクセス・メモリ領域選択パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，各種項目を指定した後，<次へ (N)> ボタンを押下します。

なお，各種項目の詳細については，ウィンドウ・レファレンスの[高速アクセス・メモリ領域選択パネル](#)を参照してください。

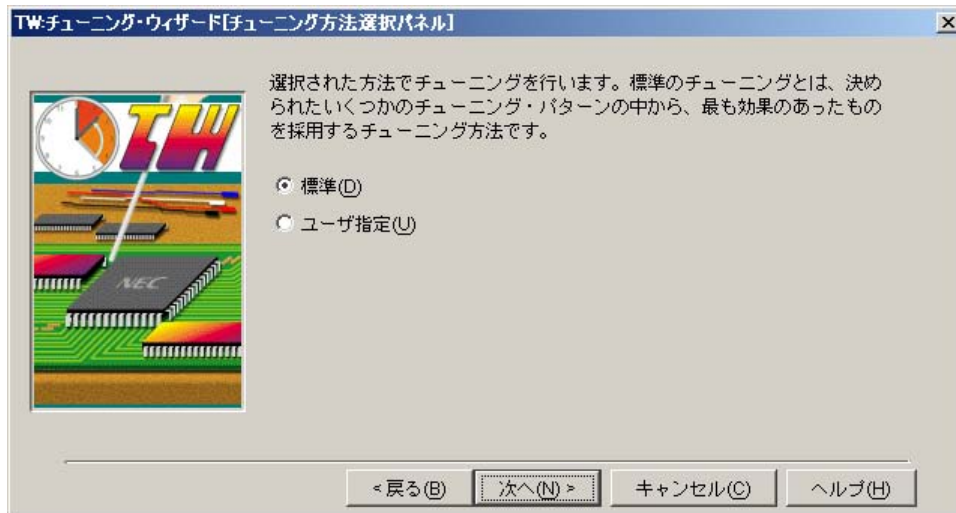
5.7.5 チューニング方法の選択

ユーザ・プログラム、実行モジュールのチューニング方法（標準パターン、ユーザ指定パターン）を指定します。

チューニング機能についての詳細は、「[4.2 チューニング機能](#)」を参照してください。

以下に示した[チューニング方法選択パネル](#)は、「[5.7.4 高速アクセス・メモリ領域の選択](#)」において各種項目を指定した後、<次へ (N)> ボタンを押下した際にオープンします。

図 5-19 チューニング方法選択パネル



ウィザード処理（性能解析処理、チューニング処理）を継続する場合、各種項目を指定した後、<次へ (N)> ボタンを押下します。

なお、各種項目の詳細については、ウィンドウ・レファレンスの[チューニング方法選択パネル](#)を参照してください。

5.8 [標準 (D)] 選択時

「5.7.5 チューニング方法の選択」において[標準 (D)]ラジオ・ボタンをチェック状態にした場合の操作手順を以下に示します。

なお、「5.7.5 チューニング方法の選択」において[ユーザ指定 (U)]ラジオ・ボタンをチェック状態にした場合の操作手順については「5.9 [ユーザ指定 (U)] 選択時」を参照してください。

5.8.1 配置先の選択

チューニング時に再配置対象となった関数の再配置用メモリ領域、および、性能解析時に中間ファイルとして出力されるプロファイル・ルーチン、プロファイル・テーブルの配置用メモリ領域を指定します。

以下に示した配置先選択パネルは、「5.7.5 チューニング方法の選択」において[標準 (D)]チェック・ボックスをチェック状態とした後、<次へ (N)> ボタンを押下した際にオープンします。

図 5-20 配置先選択パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，各種項目を指定した後，<次へ (N)> ボタンを押下します。

なお，各種項目の詳細については，ウィンドウ・レファレンスの配置先選択パネルを参照してください。

5.9 [ユーザ指定 (U)] 選択時

「5.7.5 チューニング方法の選択」において[ユーザ指定 (U)] ラジオ・ボタンをチェック状態にした場合の操作手順を以下に示します。

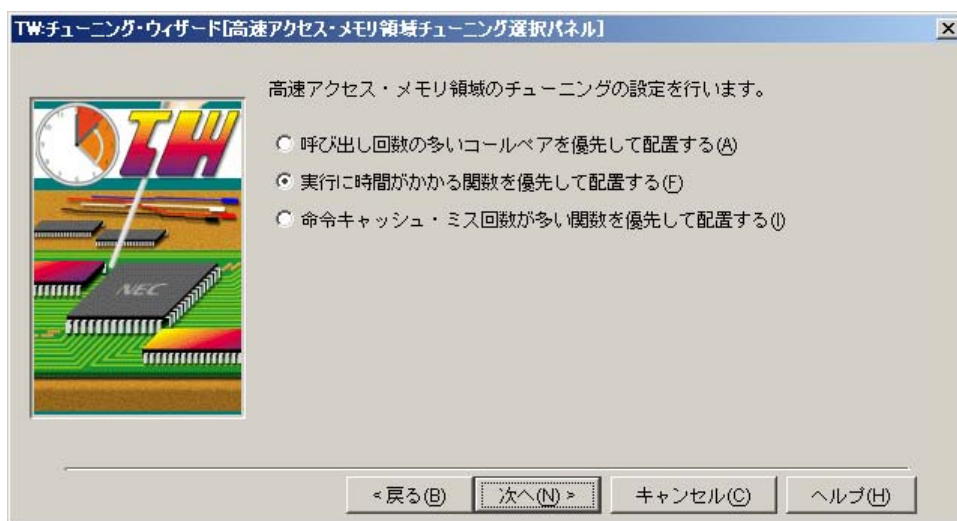
なお、「5.7.5 チューニング方法の選択」において[標準 (D)] ラジオ・ボタンをチェック状態にした場合の操作手順については「5.8 [標準 (D)] 選択時」を参照してください。

5.9.1 高速アクセス・メモリ領域チューニングの選択

ユーザ・プログラム、実行モジュールの性能解析時に獲得したコールペア情報をもとに、高速アクセス・メモリ領域（内蔵 ROM など）に対する関数配置方式を指定します。

以下に示した高速アクセス・メモリ領域チューニング選択パネルは、「5.7.5 チューニング方法の選択」において[ユーザ指定 (U)] チェック・ボックスをチェック状態とした後、< 次へ (N) > ボタンを押下した際にオープンします。

図 5-21 高速アクセス・メモリ領域チューニング選択パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，各種項目を指定した後，< 次へ (N) > ボタンを押下します。

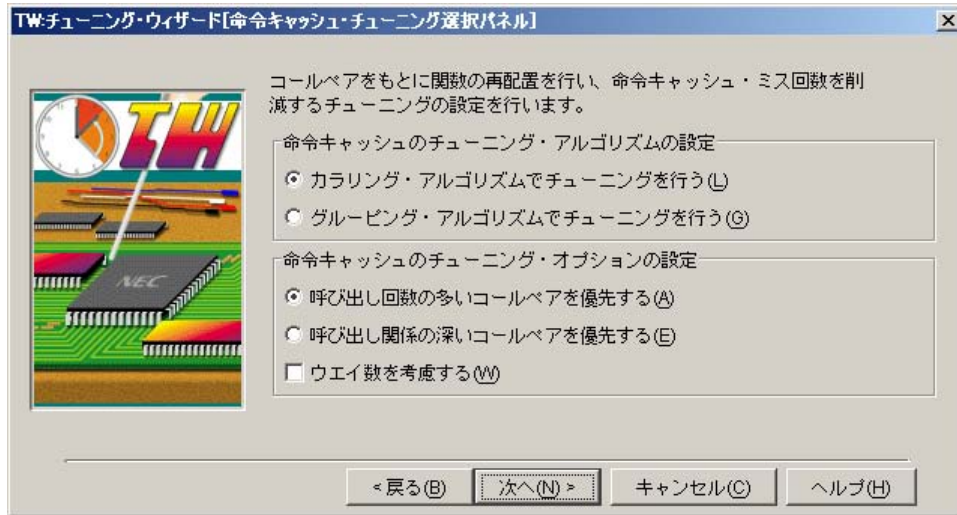
なお，各種項目の詳細については，ウィンドウ・レファレンスの高速アクセス・メモリ領域チューニング選択パネルを参照してください。

5.9.2 命令キャッシュ・チューニングの選択

ユーザ・プログラム 実行モジュールの性能解析時に獲得したコールペア情報をもとに、命令キャッシュに対するチューニング方法（チューニング・アルゴリズム、配置優先度補正オプション）を指定します。

以下に示した命令キャッシュ・チューニング選択パネルは、「5.9.1 高速アクセス・メモリ領域チューニングの選択」において各種項目を指定した後、＜次へ (N)> ボタンを押下した際にオープンします。

図 5-22 命令キャッシュ・チューニング選択パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，各種項目を指定した後，＜次へ (N)> ボタンを押下します。

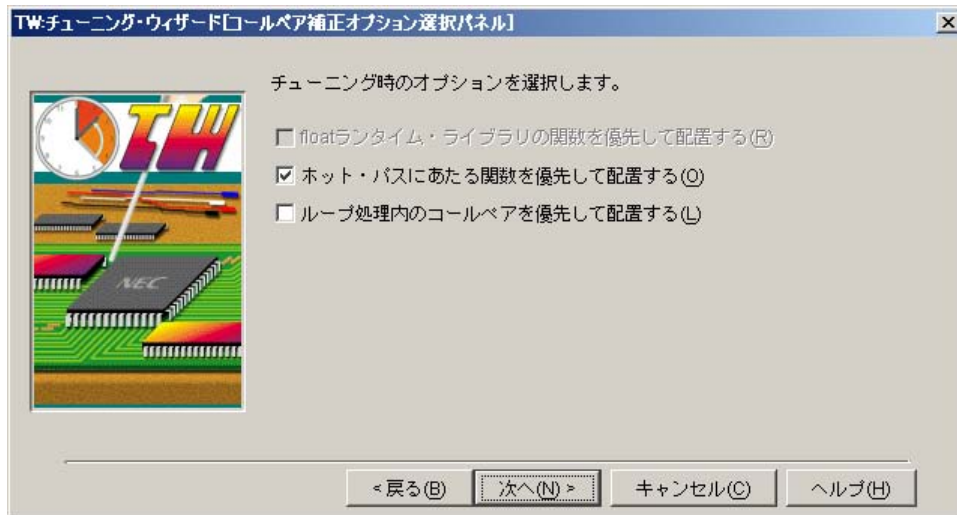
なお，各種項目の詳細については，ウィンドウ・レファレンスの命令キャッシュ・チューニング選択パネルを参照してください。

5.9.3 コールペア補正オプションの選択

ユーザ・プログラム 実行モジュールの性能解析時に獲得したコールペア情報をもとに、再配置対象関数に対するチューニング方法（高速アクセス・メモリ・チューニングと命令キャッシュ・チューニングにおけるコールペアの優先度の補正方式）を指定します。

以下に示した**コールペア補正オプション選択パネル**は、「5.9.2 命令キャッシュ・チューニングの選択」において各種項目を指定した後、<次へ (N)> ボタンを押下した際にオープンします。

図 5-23 コールペア補正オプション選択パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，各種項目を指定した後，<次へ (N)> ボタンを押下します。

なお，各種項目の詳細については，ウィンドウ・レファレンスの**コールペア補正オプション選択パネル**を参照してください。

5.9.4 優先配置関数の選択

ウィザード処理（チューニング処理）において、高速アクセス・メモリ・チューニングと命令キャッシュ・チューニングにおけるコールペアの優先度を補正するために、優先的に再配置対象とする関数を指定します。

以下に示した優先配置関数選択パネルは、「5.9.3 コールペア補正オプションの選択」において各種項目を指定した後、<次へ (N)> ボタンを押下した際にオープンします。

図 5-24 優先配置関数選択パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，各種項目を指定した後，<次へ (N)> ボタンを押下します。

なお，各種項目の詳細については，ウィンドウ・レファレンスの優先配置関数選択パネルを参照してください。

5.9.5 配置先の選択

チューニング時に再配置対象となった関数の再配置用メモリ領域、および、性能解析時に中間ファイルとして出力されるプロファイル・ルーチン、プロファイル・テーブルの配置用メモリ領域を指定します。

以下に示した配置先選択パネルは、「5.9.4 優先配置関数の選択」において各種項目を指定した後、<次へ (N)> ボタンを押下した際にオープンします。

図 5-25 配置先選択パネル



ウィザード処理（性能解析処理，チューニング処理）を継続する場合，各種項目を指定した後，<次へ (N)> ボタンを押下します。

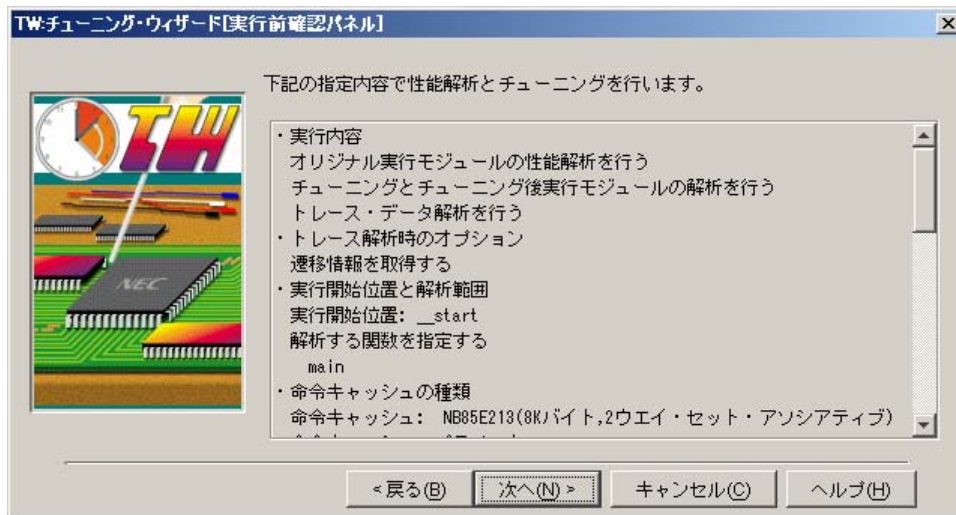
なお，各種項目の詳細については，ウィンドウ・レファレンスの配置先選択パネルを参照してください。

5.10 性能解析処理 / チューニング処理の実行

5.10.1 実行前の確認

TW850 がウィザード処理（性能解析処理，チューニング処理）を実行するうえで必要となる性能解析項目，チューニング項目の設定操作（「5.6.4 配置先の選択」，「5.8.1 配置先の選択」，または，「5.9.5 配置先の選択」までの設定操作）が正常に完了した際には，以下に示した**実行前確認パネル**がオープンします。

図 5-26 実行前確認パネル



ウィザード処理（性能解析処理，チューニング処理）を実行する場合，メッセージ表示領域に表示されている“TW850 がウィザード処理（性能解析処理，チューニング処理）を実行するうえで必要となる性能解析項目，チューニング項目の設定操作が完了したことを通知するためのメッセージ”を確認した後，<次へ(N)> ボタンを押下します。

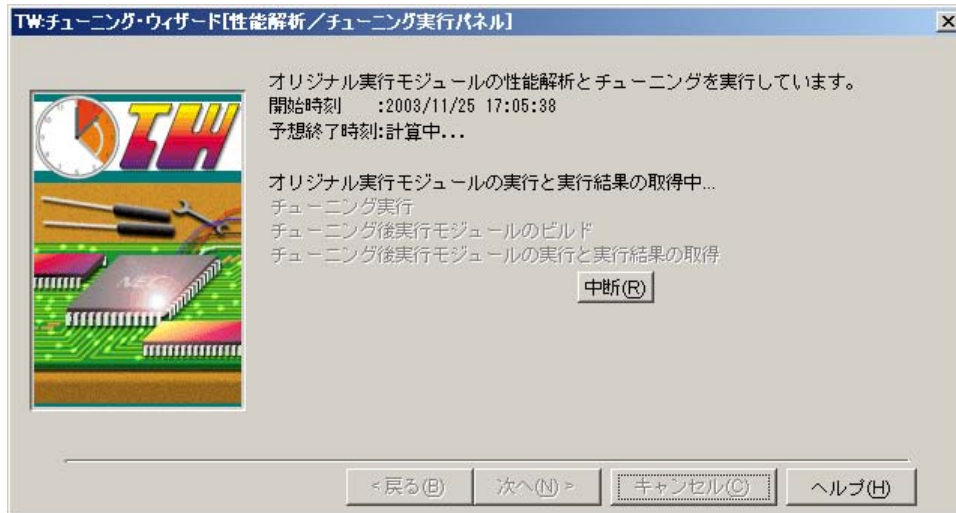
5.10.2 性能解析処理 / チューニング処理の実行

「5.10.1 実行前の確認」において表示された性能解析項目、チューニング項目の内容に従ったウィザード処理（性能解析処理、チューニング処理）を実行します。

ただし、このフェーズは、「5.10.1 実行前の確認」において表示メッセージを確認した後、＜次へ (N)> ボタンを押下した際、TW850 が自動的に行う処理であるため、ユーザが意識する必要はありません。

以下に、TW850 のウィザード処理（性能解析処理、チューニング処理）が行われている最中にオープンされる性能解析 / チューニング実行パネルを示します。

図 5-27 性能解析 / チューニング実行パネル



5.10.3 実行完了の確認

TW850 のウィザード処理 (性能解析処理, チューニング処理) が完了したことを確認します。

以下に示した**実行完了確認パネル**は,「5.10.2 性能解析処理 / チューニング処理の実行」において行われる TW850 のウィザード処理 (性能解析処理, チューニング処理) が正常終了した際,自動的にオープンします。

図 5-28 実行完了確認パネル



ウィザード処理 (性能解析処理, チューニング処理) を完了する場合,メッセージ表示領域に表示されている “ TW850 のウィザード処理 (性能解析処理, チューニング処理) が完了したことを通知するためのメッセージ ” を確認した後, <完了 (N)> ボタンを押下します。

なお, <完了 (N)> ボタンの押下により,「5.2.1 ウィザードの開始」において起動した TW850 のウィザードは終了します。

注意 「5.3 実行項目の選択」において [チューニングとチューニング後実行モジュールの解析を行う (Q)] チェック・ボックスをチェック状態にした場合, チューニング後実行モジュールが “ (オリジナル・プロジェクト・ファイル名).tw ” フォルダに出力されます。

5.11 結果確認

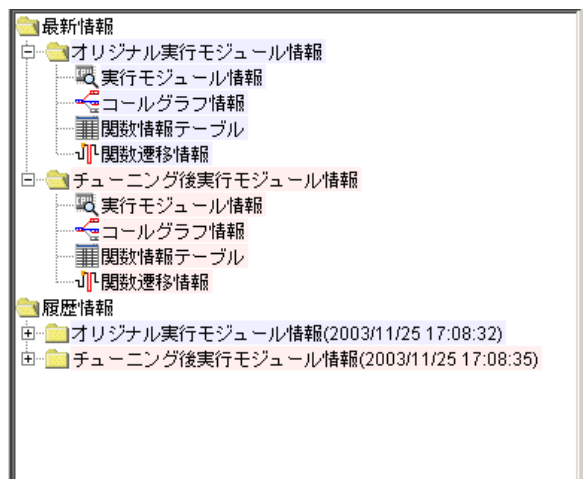
ユーザ・プログラム、実行モジュールの性能解析結果、または、チューニング結果については、[メイン・ウィンドウ](#)から確認することができます。

5.11.1 履歴情報

[メイン・ウィンドウ](#)の履歴表示領域からオリジナル実行モジュール、および、チューニング後実行モジュールの性能解析結果、または、チューニング結果を確認することができます。

以下に示した履歴表示領域は、[メイン・ウィンドウ](#)の[表示(V)]メニュー→[履歴(H)]を選択することにより表示/非表示切り替えすることができます。

図 5-29 履歴表示領域



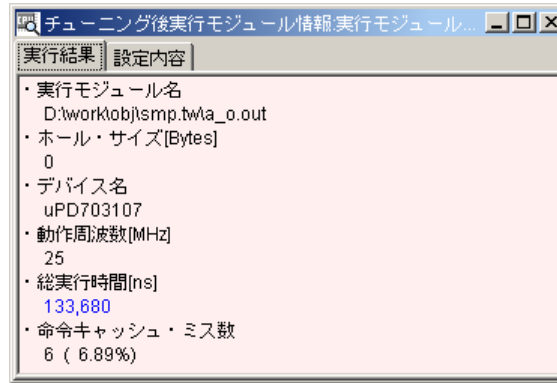
なお、本表示領域の詳細については、ウィンドウ・レファレンスの[メイン・ウィンドウ](#)を参照してください。

5.11.2 実行モジュール情報

実行モジュールの実行モジュール情報 (実行結果, 設定内容) を表示します。

以下に示した[実行モジュール情報ウィンドウ](#)は、[メイン・ウィンドウ](#)の履歴表示領域において、[実行モジュール情報] 項目を選択した後、右クリック → [開く (O)] を選択した際にオープンします。

図 5-30 実行モジュール情報ウィンドウ



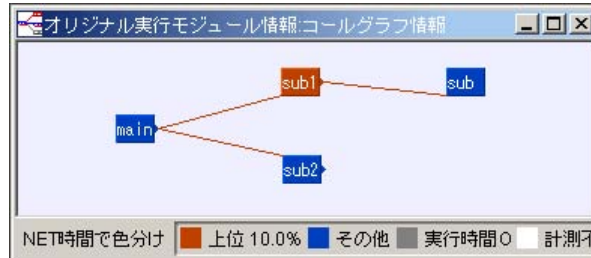
なお、本ウィンドウの詳細については、ウィンドウ・レファレンスの[実行モジュール情報ウィンドウ](#)を参照してください。

5.11.3 コールグラフ情報

実行モジュールの性能解析情報（コールペア情報，コールカウント情報など）をグラフ形式で一覧表示します。

以下に示した[コールグラフ・ウィンドウ](#)は，[メイン・ウィンドウ](#)の履歴表示領域において，[コールグラフ情報] 項目を選択した後，右クリック→[開く (O)] を選択した際にオープンします。

図 5-31 コールグラフ・ウィンドウ



なお，本ウィンドウの詳細については，ウィンドウ・レファレンスの[コールグラフ・ウィンドウ](#)を参照してください。

5.11.4 関数情報テーブル

実行モジュールの関数情報（関数名，実行回数，NET 時間など）をテーブル形式で一覧表示します。

以下に示した[関数情報テーブル・ウィンドウ](#)は，[メイン・ウィンドウ](#)の履歴表示領域において，[関数情報テーブル] 項目を選択した後，右クリック→[開く (O)]を選択した際にオープンします。

図 5-32 関数情報テーブル・ウィンドウ

オリジナル実行モジュール情報:関数情報テーブル							
関数名	実行回数	NET時間[ns]	平均NET時間[ns]	GROSS時間[ns]	平均GROSS時間[ns]	不明関数実行時間[ns]	命令キャッシュ・ミス数
sub1	1	42,080 (40.77%)	42,080 (40.77%)	54,560 (52.86%)	54,560 (52.86%)	0 (0.00%)	4 (1.90%)
func	5	20,320 (19.68%)	4,064 (3.93%)	20,320 (19.68%)	4,064 (3.93%)	0 (0.00%)	1 (1.17%)
sub2	1	19,280 (18.68%)	19,280 (18.68%)	39,600 (38.37%)	39,600 (38.37%)	0 (0.00%)	4 (18.18%)
sub	1	12,480 (12.09%)	12,480 (12.09%)	12,480 (12.09%)	12,480 (12.09%)	0 (0.00%)	2 (7.69%)
main	1	9,040 (8.75%)	9,040 (8.75%)	103,200 (100.00%)	103,200 (100.00%)	0 (0.00%)	2 (22.22%)
_sub	0	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (---.--)
_sub2	0	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (---.--)
不明関数		0 (0.00%)					0 (---.--)

なお、本ウィンドウの詳細については、ウィンドウ・レファレンスの[関数情報テーブル・ウィンドウ](#)を参照してください。

5.11.5 関数詳細情報

実行モジュールの関数詳細情報（性能解析情報，親関数情報，子関数情報）を表示します。

以下に示した[関数詳細情報ウィンドウ](#)は，[コールグラフ・ウィンドウ](#)のコールグラフ表示領域に表示されている関数名を選択した後，[メイン・ウィンドウ](#)において，関数詳細表示ボタンを押下した際，[関数情報テーブル・ウィンドウ](#)の関数名表示領域に表示されている関数名を選択した後，[メイン・ウィンドウ](#)において，関数詳細表示ボタンを押下した際，[関数詳細情報ウィンドウ](#)の[親関数]タブ，または，[子関数]タブに表示されている関数名を選択した後，[メイン・ウィンドウ](#)において，関数詳細表示ボタンを押下した際にオープンします。

図 5-33 関数詳細情報ウィンドウ

関数詳細情報	
関数名	sub1
ファイル名	main.c
関数サイズ[Bytes]	42
関数配置アドレス	0x1802f8
NET時間[ns]	42,080 (40.77%)
平均NET時間[ns]	42,080 (40.77%)
GROSS時間[ns]	54,560 (52.86%)
平均GROSS時間[ns]	54,560 (52.86%)
実行回数	1
命令キャッシュ・ミス数	4 (0.96%)

親関数	
関数名	呼び出され回数(%)
main	1 (100.00%)

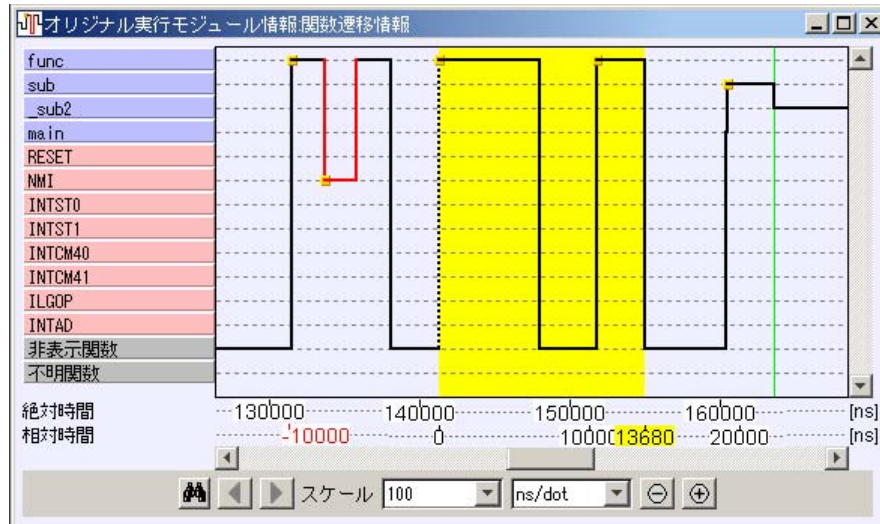
なお、本ウィンドウの詳細については、ウィンドウ・レファレンスの[関数情報テーブル・ウィンドウ](#)を参照してください。

5.11.6 関数遷移情報

実行モジュールの関数遷移情報をグラフ形式で一覧表示します。

以下に示した関数遷移情報ウィンドウは、メイン・ウィンドウの履歴表示領域において、[関数遷移情報] 項目を選択した後、右クリック→[開く (O)] を選択した際にオープンします。

図 5-34 関数遷移情報ウィンドウ



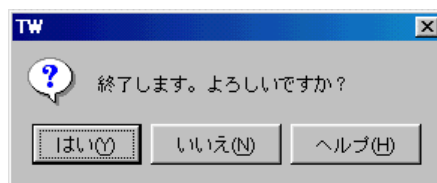
なお、本ウィンドウの詳細については、ウィンドウ・レファレンスの関数遷移情報ウィンドウを参照してください。

5.12 TW850 の終了

TW850 の終了方法についての詳細は、「[3.2 終了方法](#)」を参照してください。

なお、TW850 に対する終了操作を実行した際には、以下に示した[確認ダイアログ](#)が自動的にオープンします。

図 5-35 確認ダイアログ



TW850 に対する終了操作を継続する場合、メッセージ表示領域に表示されている“問い合わせ情報”を確認した後、[<はい\(Y\)>](#) ボタンを押下します。

第6章 ウィンドウ・レファレンス

本章では、TW850 のウィンドウ、パネル、ダイアログの機能詳細について解説しています。

表 6-1 に、TW850 のウィンドウ、パネル、ダイアログの一覧を示します。

表 6-1 TW850 のウィンドウ、パネル、ダイアログ

ウィンドウ、パネル、ダイアログ名	機能概要
メイン・ウィンドウ	ウィザード処理 (性能解析処理, チューニング処理) の起動, および, 性能解析情報 (コールペア情報, コールカウント情報など) の表示などを行うためのウィンドウ
コールグラフ・ウィンドウ	実行モジュールの性能解析情報 (コールペア情報, コールカウント情報など) をグラフ形式で一覧表示するためのウィンドウ
関数情報テーブル・ウィンドウ	実行モジュールの関数情報 (関数名, 実行回数, NET 時間など) をテーブル形式で一覧表示するためのウィンドウ
関数遷移情報ウィンドウ	実行モジュールの関数遷移情報をグラフ形式で一覧表示するためのウィンドウ
実行モジュール情報ウィンドウ	実行モジュールの実行モジュール情報 (実行結果, 設定内容) を表示するためのウィンドウ
関数詳細情報ウィンドウ	実行モジュールの関数詳細情報 (性能解析情報, 親関数情報, 子関数情報) を表示するためのウィンドウ
チューニング・ウィザード 初期画面パネル	TW850 のウィザード処理 (性能解析処理, チューニング処理) が開始されたことの通知と, 性能解析を行うプログラムに使用している RTOS の指定を行うためのパネル
チューニング・ウィザード プロジェクト情報獲得パネル	TW850 のウィザード処理 (プロジェクト情報の獲得処理) が実行中であることを通知するためのパネル
チューニング・ウィザード 解析対象ソース・ファイル選択パネル	TW850 のウィザード処理 (性能解析処理, チューニング処理) において, 性能解析 / チューニング対象とするソース・ファイルを指定するためのパネル
チューニング・ウィザード 実行モジュール情報獲得パネル	TW850 のウィザード処理 (オリジナル解析用実行モジュールの生成処理, 関数解析, オリジナル実行モジュール解析処理) が実行中であることを通知するためのパネル
チューニング・ウィザード 実行項目選択パネル	TW850 のウィザード処理 (性能解析処理, チューニング処理) の実行項目を指定するためのパネル
チューニング・ウィザード トレース解析オプション選択パネル	TW850 のウィザード処理 (性能解析処理) において, トレース・データ解析方式を用いた性能解析情報の獲得オプションを指定するためのパネル
チューニング・ウィザード プローブ解析オプション選択パネル	TW850 のウィザード処理 (性能解析処理) において, プローブ・コード挿入方式を用いた性能解析情報の獲得オプションを指定するためのパネル
チューニング・ウィザード 実行解析範囲選択パネル	ユーザ・プログラム, 実行モジュールの実行開始位置, および, ウィザード処理 (性能解析処理) における性能解析範囲を指定するためのパネル
チューニング・ウィザード トレース・イベント選択パネル	TW850 のウィザード処理 (性能解析処理) における性能解析範囲のトレース・イベントを選択するためのパネル
チューニング・ウィザード 命令キャッシュ選択パネル	TW850 のウィザード処理 (性能解析処理, チューニング処理) において, 使用する命令キャッシュの種別, および, 命令キャッシュ・パラメータを指定するためのパネル

ウィンドウ、パネル、ダイアログ名	機能概要
チューニング・ウィザード 高速アクセス・メモリ領域選択パネル	TW850 のウィザード処理 (チューニング処理) において、使用する高速アクセス・メモリ領域の種別、および、パラメータを指定するためのパネル
チューニング・ウィザード チューニング方法選択パネル	ユーザ・プログラム、実行モジュールのチューニング方法 (標準パターン、ユーザ指定パターン) を指定するためのパネル
チューニング・ウィザード 高速アクセス・メモリ領域チューニング選択パネル	ユーザ・プログラム、実行モジュールの性能解析時に獲得したコールペア情報をもとに、高速アクセス・メモリ領域 (内蔵 ROM など) に対する関数配置方式を指定するためのパネル
チューニング・ウィザード 命令キャッシュ・チューニング選択パネル	ユーザ・プログラム、実行モジュールの性能解析時に獲得したコールペア情報をもとに、命令キャッシュに対するチューニング方法 (チューニング・アルゴリズム、配置優先度補正オプション) を指定するためのパネル
チューニング・ウィザード コールペア補正オプション選択パネル	ユーザ・プログラム、実行モジュールの性能解析時に獲得したコールペア情報に対するチューニング方法 (高速アクセス・メモリ・チューニングと命令キャッシュ・チューニングにおけるコールペアの優先度の補正方式) を指定するためのパネル
チューニング・ウィザード 優先配置関数選択パネル	TW850 のウィザード処理 (チューニング処理) において、高速アクセス・メモリ・チューニングと命令キャッシュ・チューニングにおけるコールペアの優先度を補正するために、優先的に再配置対象とする関数を指定するためのパネル
チューニング・ウィザード 配置先選択パネル	チューニング時に再配置対象となった関数の再配置用メモリ領域、および、性能解析時に中間ファイルとして出力されるプロファイル・ルーチン、プロファイル・テーブルの配置用メモリ領域を指定するためのパネル
チューニング・ウィザード 実行前確認パネル	TW850 がウィザード処理 (性能解析処理、チューニング処理) を実行するうえで必要となる性能解析項目、チューニング項目の設定操作が完了したことを通知するためのパネル
チューニング・ウィザード 性能解析 / チューニング実行パネル	TW850 のウィザード処理 (性能解析処理、チューニング処理) が実行中であることを通知するためのパネル
チューニング・ウィザード 実行完了確認パネル	TW850 のウィザード処理 (性能解析処理、チューニング処理) が正常終了したことを通知するためのパネル
表示関数選択ダイアログ	関数遷移情報ウィンドウに表示する関数、または、割り込みを選択するためのダイアログ
関数遷移情報検索ダイアログ	関数遷移情報ウィンドウにおいて、関数実行線、または、関数遷移線を検索するためのダイアログ
オプション・ダイアログ	メイン・ウィンドウの内部ウィンドウ表示領域に表示されている コールグラフ・ウィンドウ の印刷方法、TW850 が性能解析処理 / チューニング処理を実行した際に出力した中間ファイル (一時ファイル) を削除するか否かなどを指定するためのダイアログ
バージョン情報ダイアログ	TW850 のバージョン情報 (バージョン、著作権など) を表示するためのダイアログ
ファイル選択ダイアログ	TW850 が性能解析処理 / チューニング処理を実行する際に必要となるファイル (タイマ制御コード部など) を指定するためのダイアログ
メッセージ・ダイアログ	TW850 が性能解析処理 / チューニング処理を実行中に検出したユーザに通知すべき情報 (処理の経過状況、警告事項の発生など) を表示するためのダイアログ
エラー・ダイアログ	TW850 の起動時や、TW850 が性能解析処理 / チューニング処理を実行中に検出した“ユーザに通知すべき情報 (致命的エラーの発生など)”を表示するためのダイアログ
確認ダイアログ	TW850 が性能解析処理 / チューニング処理を実行中に検出した問い合わせ情報 (処理を続行するうえでの問題点など) を表示するためのダイアログ

ウィンドウ , パネル , ダイアログ名	機能概要
入力ダイアログ	TW850 から要求されている情報 (色分け境界 , 表示倍率 , ルート関数など) を指定するためのダイアログ
TW850 起動オプション設定ダイアログ	TW850 の起動オプションを指定するためのダイアログ

6.1 入力規則

以下に、TW850 のウィンドウ、パネル、ダイアログで文字セットを入力する際の規則を示します。

- ファイル名

- ファイル名は、a-z, A-Z, 0-9, ., _ , + , - のいずれかの文字で構成されます。
- ファイル名の先頭に . , + , - は使えません。
- ファイル名の最後に . は使えません。
- 英大文字 (A-Z), 英小文字 (a-z) は区別されません。
- TW850 では、ユーザ・プログラム、実行モジュールの性能解析時、チューニング時にオリジナル・ファイル名に最大 5 文字を付加した中間ファイルを作成します。
したがって、ユーザが指定可能なオリジナル・ファイル名は、C コンパイラ・パッケージで規定されている最大文字数 (259 文字) よりも 5 文字少ない文字列となります。

- パス名

- パス名は、a-z, A-Z, 0-9, ., _ , + , - , 半角スペースのいずれかの文字で構成されます。
- パス名の先頭に . , + , - , 半角スペースは使えません。
- パス名の最後に . , 半角スペースは使えません。
- 英大文字 (A-Z), 英小文字 (a-z) は区別されません。
- TW850 では、ユーザ・プログラム、実行モジュールの性能解析時、チューニング時にオリジナル・ファイルが格納されているフォルダ、または、TW850 が規定しているフォルダに中間ファイルを作成します。
したがって、ユーザが指定可能なパス名は、C コンパイラ・パッケージで規定されている最大文字数 (259 文字) となります。

- 関数名、シンボル名

- TW850 では、ユーザ・プログラム、実行モジュールの性能解析時、チューニング時にオリジナル関数名、および、オリジナル・シンボル名に最大 12 文字を付加した性能解析情報を出力します。
したがって、ユーザが指定可能なオリジナル関数名、および、オリジナル・シンボル名は、C コンパイラ・パッケージで規定されている最大文字数 (259 文字) よりも 12 文字少ない文字列となります。
- TW850 では、“TW_”, および、“tw_” で始まる関数名、および、シンボル名を、TW850 が予約する関数名、および、シンボル名として扱うため、使用することはできません。

- セクション名

- TW850 では、“TW_”, および、“tw_” で始まるセクション名を、TW850 が予約するセクション名として扱うため、使用することはできません。

- マクロ名

- TW850 では、“tw_”, または、“..” で始まる単語、および、“__e_sysfnc” をシステム予約マクロとして扱うため、使用することはできません。

- 数値

- TW850 では、数字 (0 ~ 9) で始まる単語が数値として扱われます。
なお、TW850 では、通常、数値を以下のように区別しています。

```
XXXX      : 10 進数
0xFFFF    : 16 進数
```

6.2 ウィンドウ解説

TW850 のウィンドウ、パネル、ダイアログについて、以下の記述フォーマットに従って解説します。

ウィンドウ名、パネル名、ダイアログ名

ウィンドウ、パネル、ダイアログの機能概要を記述した後、表示イメージを示しています。

オープン方法

ウィンドウ、パネル、ダイアログのオープン方法を記述しています。

エリア説明

ウィンドウ、パネル、ダイアログの構成要素を箇条書きで示した後、各構成要素の詳細情報を記述しています。

なお、パネルの機能ボタンは、以下に示したボタン群から構成されています。

(1) << 戻る (B) > ボタン

直前のウィザード・ステップに戻ります。

なお、本ボタンは、初期画面パネル、実行完了確認パネルでは無効状態となります。

(2) < 次へ (N) > ボタン

次のウィザード・ステップに進みます（設定内容によって、次に表示されるパネルは異なります）。

なお、本ボタンは、プロジェクト情報獲得パネル、実行モジュール情報獲得パネル、性能解析 / チューニング実行パネルでは無効状態となります。

また、実行完了確認パネルでは < 完了 (N) > ボタンとなり、パネルを閉じます。

(3) < キャンセル (C) > ボタン

ウィザード処理を実行することなく、パネルを閉じます。

なお、本ボタンは、実行完了確認パネルでは無効状態となります。

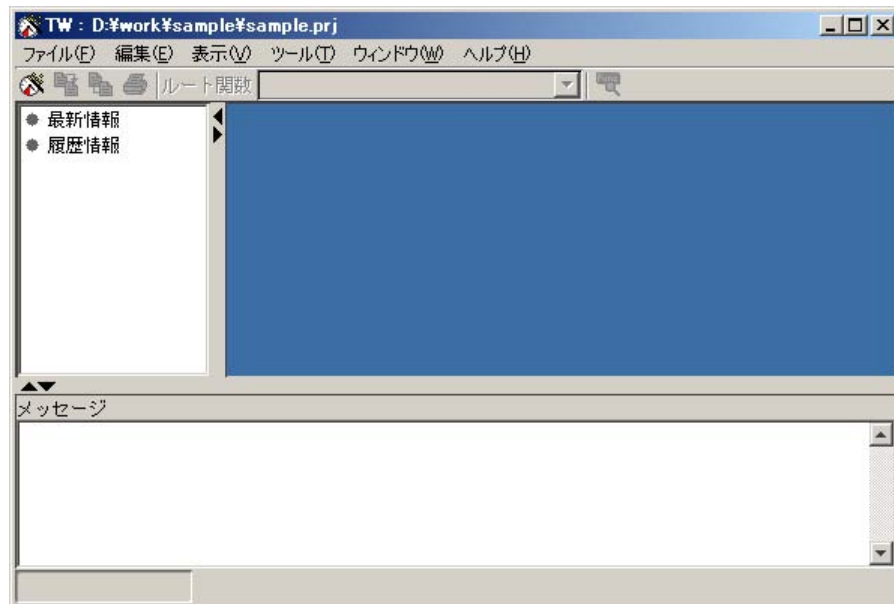
(4) < ヘルプ (H) > ボタン

ヘルプを起動します。

メイン・ウィンドウ

ウィザード処理（性能解析処理，チューニング処理）の起動，および，性能解析情報（コールペア情報，コールカウント情報など）の表示などを行うためのウィンドウです。

図 6-1 メイン・ウィンドウ



ここでは，次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)
- [コンテキスト・メニュー](#)

オープン方法

本ウィンドウは，TW850 を起動した際，まず最初に自動的にオープンします。

エリア説明

本ウィンドウは，次の項目から構成されています。

- [メニュー・バー](#)
- [ツール・バー](#)
- [履歴表示領域](#)
- [内部ウィンドウ表示領域](#)
- [メッセージ表示領域](#)
- [ステータス・バー](#)

(1) メニュー・バー

メニュー・バーは、以下に示したメニューから構成されています。

(a) [ファイル (E)] メニュー

以下に示した項目から構成されています。

- [プロジェクト情報の保存 (S)]
TW850 の実行状況を保存します。
- [ファイルから入力 (I)...]
本項目を選択することにより、[ファイルに出力 (E)...] で保存した関数遷移情報データを読み込むためのファイルを指定するためのダイアログ ([ファイル選択ダイアログ](#)) がオープンします。
- [ファイルに出力 (E)...]
内部ウィンドウ ([関数情報テーブル・ウィンドウ](#), [関数遷移情報ウィンドウ](#), [実行モジュール情報ウィンドウ](#), [関数詳細情報ウィンドウ](#) など) を選択した後、本項目を選択することにより、該当ウィンドウをテキスト形式で保存するためのファイルを指定するためのダイアログ ([ファイル選択ダイアログ](#)) がオープンします。
- [印刷 (P)...]
内部ウィンドウ ([コールグラフ・ウィンドウ](#), [関数情報テーブル・ウィンドウ](#), [関数遷移情報ウィンドウ](#) など) を選択した後、本項目を選択することにより、該当ウィンドウを印刷します。
- [終了 (X)...]
本ウィンドウを閉じる [確認ダイアログ](#) をオープンします。

(b) [編集 (E)] メニュー

以下に示した項目から構成されています。

- [切り取り (I)]
テキスト入力領域において選択したテキストをコピー・バッファに切り取ります。
注意 本項目は、以下に示した条件が成立した場合に無効状態となります。
- TW850 がウィザード処理を実行中
- [コピー (C)]
テキスト入力領域において選択したテキストをコピー・バッファにコピーします。
注意 本項目は、以下に示した条件が成立した場合に無効状態となります。
- TW850 がウィザード処理を実行中
- [貼り付け (P)]
[切り取り (I)], または, [コピー (C)] でコピー・バッファに格納 (切り取り, または, コピー) したテキストを指定されたテキスト入力領域に貼り付けます。
注意 本項目は、以下に示した条件が成立した場合に無効状態となります。
- TW850 がウィザード処理を実行中
- [削除 (D)]
テキスト入力領域において選択したテキストを削除します。
なお、履歴表示領域にフォーカスがある場合、選択した履歴情報項目を削除します。
注意 本項目は、以下に示した条件が成立した場合に無効状態となります。
- TW850 がウィザード処理を実行中
- [検索 (S)...]
関数の遷移を検索するためのダイアログ ([関数遷移情報検索ダイアログ](#)) がオープンします。
注意 本項目は、以下に示した条件が成立した場合に無効状態となります。
- [関数遷移情報ウィンドウ](#) が非選択状態

- [前を検索 (M)]

関数遷移情報ウィンドウの関数遷移情報表示領域において、選択状態である関数実行線、または、関数遷移線を基準に、関数遷移情報ウィンドウで指定された条件（関数、または、割り込み）の関数実行線、または、関数遷移線を左方向に検索します。

注意 本項目は、以下に示した何れかの条件が成立した場合に無効状態となります。

- 関数遷移情報ウィンドウが非選択状態
- 関数遷移情報ウィンドウにおいて、指定されている検索条件が無効

- [次を検索 (N)]

関数遷移情報ウィンドウの関数遷移情報表示領域において、選択状態である関数実行線、または、関数遷移線を基準に、関数遷移情報ウィンドウで指定された条件（関数、または、割り込み）の関数実行線、または、関数遷移線を右方向に検索します。

注意 本項目は、以下に示した何れかの条件が成立した場合に無効状態となります。

- 関数遷移情報ウィンドウが非選択状態
- 関数遷移情報ウィンドウにおいて、指定されている検索条件が無効

- [履歴に追加 (H)]

履歴表示領域の [最新情報] 項目に表示されている項目 ([オリジナル実行モジュール情報] 項目 [チューニング後実行モジュール情報] 項目など) を選択した後、本項目を選択することにより、該当項目を [履歴情報] 項目に追加登録します。

- [履歴から復帰 (R)...]

履歴表示領域の [履歴情報] 項目に表示されている項目 ([オリジナル実行モジュール情報] 項目 [チューニング後実行モジュール情報] 項目など) を選択した後、本項目を選択することにより、該当項目の設定内容がウィザードに反映されます。

(c) [表示 (V)] メニュー

以下に示した項目から構成されています。

- [ツールバー (T)]

ツール・バー（チューニング・ウィザードボタン、履歴に追加ボタン、履歴から復帰ボタンなど）を表示するか否かを指定します。

なお、本項目がチェック状態の場合、“ツール・バーを表示する”となります。

- [履歴 (H)]

履歴表示領域（ウィザード処理の結果履歴をツリー表示するための領域）を表示するか否かを指定します。

なお、本項目がチェック状態の場合、“履歴表示領域を表示する”となります。

- [メッセージ (M)]

メッセージ表示領域（TW850 が検出した“ユーザに通知すべき情報”を表示するための領域）を表示するか否かを指定します。

なお、本項目がチェック状態の場合、“メッセージ表示領域を表示する”となります。

- [開く (O)]

履歴表示領域に表示されている項目（コールグラフ情報、関数情報テーブル、関数遷移情報、実行モジュール情報）を選択した後、本項目を選択することにより、該当項目に対応したウィンドウ（[コールグラフ・ウィンドウ](#)、[関数情報テーブル・ウィンドウ](#)、[関数遷移情報ウィンドウ](#)、[実行モジュール情報ウィンドウ](#)）がオープンします。

- [関数詳細情報 (D)]

[コールグラフ・ウィンドウ](#)、[関数情報テーブル・ウィンドウ](#)、[関数遷移情報ウィンドウ](#)、[関数詳細情報ウィンドウ](#)に表示されている関数名を選択した後、本項目を選択することにより、実行モジュールの関数詳細情報（性能解析情報、親関数情報、子関数情報）を表示するためのウィンドウ（[関数詳細情報ウィンドウ](#)）がオープンします。

注意 本項目は、以下に示した何れかの条件が成立した場合に無効状態となります。

- [コールグラフ・ウィンドウ](#)、[関数情報テーブル・ウィンドウ](#)、[関数遷移情報ウィンドウ](#)、[関数詳細情報ウィンドウ](#)の何れも非選択状態
- [コールグラフ・ウィンドウ](#)、[関数情報テーブル・ウィンドウ](#)、[関数遷移情報ウィンドウ](#)、[関数詳細情報ウィンドウ](#)の何れかが選択状態、かつ、選択状態のウィンドウにおいて、関数が非選択状態

- [コールグラフ表示方法 (C)]

以下に示した項目から構成されています。

- [NET 時間で色分け]

コールグラフ・ウィンドウのコールグラフ表示領域に表示される関数ボックスの色分け基準を NET 時間で行うか否かを指定します。

なお、本項目がチェック状態の場合、関数ボックスの色分け基準は“ NET 時間で色分け ”となります。

- [GROSS 時間で色分け]

コールグラフ・ウィンドウのコールグラフ表示領域に表示される関数ボックスの色分け基準を GROSS 時間で行うか否かを指定します。

なお、本項目がチェック状態の場合、関数ボックスの色分け基準は“ GROSS 時間で色分け ”となります。

- [実行回数で色分け (C)]

コールグラフ・ウィンドウのコールグラフ表示領域に表示される関数ボックスの色分け基準を実行回数で行うか否かを指定します。

なお、本項目がチェック状態の場合、関数ボックスの色分け基準は“ 実行回数で色分け ”となります。

- [命令キャッシュ・ミス数で色分け (I)]

コールグラフ・ウィンドウのコールグラフ表示領域に表示される関数ボックスの色分け基準を命令キャッシュ・ミス数で行うか否かを指定します。

なお、本項目がチェック状態の場合、関数ボックスの色分け基準は“ 命令キャッシュ・ミス数で色分け ”となります。

- [色分け境界設定 (I)...]

コールグラフ・ウィンドウのコールグラフ表示領域に表示される関数ボックスの色分け境界値を指定するためのダイアログ (**入力ダイアログ**) がオープンします。

なお、指定可能な値の範囲は、0% ~ 100% となります。デフォルトでは、10% が指定されます。

- [展開 (N)], [折りたたみ (N)]

コールグラフ・ウィンドウのコールグラフ表示領域に表示されている関数ボックスを選択した後、本項目を選択することにより、該当関数ボックスが展開 (子関数を表示) 状態、または、折りたたみ (子関数を非表示) 状態となります。

- [ズーム (Z)...]

コールグラフ・ウィンドウを選択した後、本項目を選択することにより、コールグラフ表示領域の表示倍率を変更するためのウィンドウ (**入力ダイアログ**) がオープンします。

選択項目は、“ 200% ”、“ 150% ”、“ 100% ”、“ 75% ”、“ 50% ”、“ ウィンドウにあわせる ”となります。

また、テキスト入力も可能です。

なお、指定可能な値の範囲は、10% ~ 1000% となります。

- [ルート関数変更 (R)...]

コールグラフ・ウィンドウを選択した後、本項目を選択することにより、コールグラフ表示領域に表示されているルート関数を変更するためのダイアログ (**入力ダイアログ**) がオープンします。

選択項目は、TW850 が認識しているすべての関数名となります。

注意 同じ名前の関数が複数存在する場合、“ 関数名 (ファイル名) ”のように表示されます。

また、関数名、および、ファイル名が同じ名前である関数が複数存在する場合、“ 関数名 (絶対パス名) ”のように表示されます。

- [実行時間 0 の関数を表示しない (H)], [実行回数 0 の関数を表示しない (H)]

内部ウィンドウ (**コールグラフ・ウィンドウ**、**関数情報テーブル・ウィンドウ**、**実行モジュール情報ウィンドウ**など)を表示する際、実行時間 0、または、実行回数 0 の関数を表示するか否かを指定します。

なお、本項目を選択した場合、“ 実行時間 0、または、実行回数 0 の関数を表示しない ”となります。

- [関数遷移情報表示方法 (E)]

以下に示した項目から構成されています。

- [ソート (S)]

関数遷移情報ウィンドウの関数遷移情報表示領域において、関数ボックス、および、割り込みボックスのソート条件を選択します。

本項目は、以下に示した項目から構成されています。

- [名前順 (N)]

関数遷移情報ウィンドウの関数遷移情報表示領域において、関数ボックス、および、割り込みボックスを名前順にソートします。

ただし、関数ボックス、割り込みボックス、その他のボックスの表示順は変わりません。

- [アドレス順 (A)]

関数遷移情報ウィンドウの関数遷移情報表示領域において、関数ボックス、および、割り込みボックスをアドレス順にソートします。

ただし、関数ボックス、割り込みボックス、その他のボックスの表示順は変わりません。

- [ファイル順 (F)]

関数遷移情報ウィンドウの関数遷移情報表示領域において、関数ボックス、および、割り込みボックスをファイル順にソートします。

ただし、関数ボックス、割り込みボックス、その他のボックスの表示順は変わりません。

- [表示関数選択 (D) ...]

関数遷移情報ウィンドウの関数遷移情報表示領域において、表示する関数ボックス、および、割り込みボックスを選択するためのダイアログ (表示関数選択ダイアログ) がオープンします。

- [非表示関数に関する遷移線を表示しない (H)]

関数遷移情報ウィンドウの関数遷移情報表示領域において、非表示となっている関数に関する関数遷移線を表示するか否かを指定します。

なお、本項目がチェック状態の場合、“非表示となっている関数に関する関数遷移線を表示する”となります。

注意 本項目は、以下に示した条件が成立した場合に無効状態となります。

- 関数遷移情報ウィンドウが非選択状態

(d) [ツール (T)] メニュー

以下に示した項目から構成されています。

- [チューニングウィザード (W) ...]

TW850 のウィザード処理 (性能解析処理、チューニング処理) を開始します。

なお、本項目の選択により、TW850 のウィザード処理 (性能解析処理、チューニング処理) が開始されたことを通知するためのパネル (初期画面パネル) がオープンします。

- [オプション (O) ...]

オプション設定処理を開始します。

なお、本項目の選択により、コールグラフ・ウィンドウや関数遷移情報ウィンドウの印刷方法、TW850 が性能解析処理 / チューニング処理を実行した際に出力した中間ファイル (一時ファイル) を削除するか否かを指定するためのダイアログ (オプション・ダイアログ) がオープンします。

(e) [ウィンドウ (W)] メニュー

以下に示した項目から構成されています。

- [重ねて表示 (C)]

内部ウィンドウを“重ねて表示 (カスケード表示)”します。

- [並べて表示 (I)]

内部ウィンドウを“並べて表示 (タイル表示)”します。

- [分割 (S)]

関数遷移情報ウィンドウの関数遷移情報表示領域を左右ふたつに分割して表示するか否かを指定します。

なお、本項目がチェック状態の場合、“関数遷移情報ウィンドウの関数遷移情報表示領域を左右ふたつに分割して表示する”となります。

注意 本項目は、以下に示した条件が成立した場合に無効状態となります。

- 関数遷移情報ウィンドウが非選択状態

(f) [ヘルプ (H)] メニュー

以下に示した項目から構成されています。

- [TW のヘルプ (H)]

TW850 全般に関するヘルプを起動します。

- [メイン ウィンドウ (M)]
本ウィンドウに関するヘルプを起動します。
- [カレント ウィンドウ (C)]
アクティブ状態のウィンドウ、パネル、ダイアログに関するヘルプを起動します。
- [バージョン情報 (A)...]
TW850 のバージョン情報 (バージョン、著作権など) を表示するためのダイアログ ([バージョン情報ダイアログ](#)) がオープンします。

(2) ツール・バー

ツール・バーは、以下に示したボタン群、コンボ・ボックスから構成されています。

(a) チューニング・ウィザードボタン



TW850 ウィザード処理 (性能解析処理、チューニング処理) を開始します。

なお、本ボタンの押下により、TW850 のウィザード処理 (性能解析処理、チューニング処理) が開始されたことを通知するためのパネル ([初期画面パネル](#)) がオープンします。

注意 チューニング・ウィザードボタンにマウス・カーソルを当てた状態にすると、“チューニング・ウィザードを用いて実行モジュールの性能解析、チューニングを行います。” がポップ・アップ表示されます。

(b) 履歴に追加ボタン



履歴表示領域の [最新情報] 項目に表示されている項目 ([オリジナル実行モジュール情報] 項目、[チューニング後実行モジュール情報] 項目など) を選択した後、本ボタンを押下することにより、該当項目を [履歴情報] 項目に追加登録します。

なお、[履歴情報] 項目には、項目を 20 個まで追加することができます。

注意 本ボタンにマウス・カーソルを当てた状態にすると、“選択されている最新情報を履歴情報に追加します。” がポップ・アップ表示されます。

(c) 履歴から復帰ボタン



履歴表示領域の [履歴情報] 項目に表示されている項目 ([オリジナル実行モジュール情報] 項目、[チューニング後実行モジュール情報] 項目など) を選択した後、本ボタンを押下することにより、該当項目の設定内容がウィザードに反映されます。

注意 本ボタンにマウス・カーソルを当てた状態にすると、“選択されている履歴情報の設定内容を TW850 に反映します。” がポップ・アップ表示されます。

(d) 印刷ボタン



内部ウィンドウ ([コールグラフ・ウィンドウ](#)、[関数情報テーブル・ウィンドウ](#)、[関数遷移情報ウィンドウ](#)) を選択した後、本ボタンを選択することにより、該当ウィンドウを印刷します。

注意 本ボタンにマウス・カーソルを当てた状態にすると、“選択されているウィンドウの内容を印刷します。” がポップ・アップ表示されます。

(e) ルート関数変更コンボ・ボックス



[コールグラフ・ウィンドウ](#) を選択した後、本コンボ・ボックスを選択することにより、コールグラフ表示領域に表示されているルート関数が変更されます。

選択項目は、TW850 が認識しているすべての関数名となります。

注意 1 同じ名前の関数が複数存在する場合、“関数名 (ファイル名)” のように表示されます。

また、関数名、および、ファイル名が同じ名前である関数が複数存在する場合、“関数名 (絶対パス名)” のように表示されます。

注意 2 本コンボ・ボックスにマウス・カーソルを当てた状態にすると、“コールグラフのルート関数を変更します。” がポップ・アップ表示されます。

(f) 関数詳細表示ボタン



コールグラフ・ウィンドウ、関数情報テーブル・ウィンドウ、関数遷移情報ウィンドウに表示されている関数名を選択した後、本ボタンを押下することにより、実行モジュールの関数詳細情報（性能解析情報、親関数情報、子関数情報）を表示するためのウィンドウ（[関数詳細情報ウィンドウ](#)）がオープンします。

注意 本ボタンにマウス・カーソルを当てた状態にすると、“選択されている関数の詳細情報を表示します。”がポップ・アップ表示されます。

(3) 履歴表示領域

ウィザード処理（性能解析処理、チューニング処理）の結果履歴をツリー表示するための領域です。
 なお、本表示領域の表示／非表示は、[表示 (V)] メニュー→[履歴 (H)] で切り替えることができます。
 以下に、本表示領域の表示項目を示します。

(a) [最新情報] 項目

TW850 が実行したウィザード処理（性能解析処理、チューニング処理）の最新の結果を表示します。

(b) [履歴情報] 項目

TW850 が実行したウィザード処理（性能解析処理、チューニング処理）の結果（最大過去 20 件）を表示します。

(c) [オリジナル実行モジュール情報] 項目

[最新情報] 項目、[履歴情報] 項目の子項目です。

なお、本項目は、TW850 がオリジナル実行モジュールに対して行ったウィザード処理（性能解析処理）の結果を水色表示します。

(d) [チューニング後実行モジュール情報] 項目

[最新情報] 項目、[履歴情報] 項目の子項目です。

なお、本項目は、TW850 がチューニング後実行モジュールに対して行ったウィザード処理（性能解析処理）の結果をピンク色表示します。

(e) [実行モジュール情報] 項目

[オリジナル実行モジュール情報] 項目、[チューニング後実行モジュール情報] 項目の子項目です。

なお、本項目を選択した後、右クリック→[開く (Q)] を選択、または、[表示 (V)] メニュー→[開く (Q)] を選択することにより、該当実行モジュールの実行モジュール情報（実行結果、設定内容）を表示するためのウィンドウ（[実行モジュール情報ウィンドウ](#)）がオープンします。

(f) [コールグラフ情報] 項目

[オリジナル実行モジュール情報] 項目、[チューニング後実行モジュール情報] 項目の子項目です。

なお、本項目を選択した後、右クリック→[開く (Q)] を選択、または、[表示 (V)] メニュー→[開く (Q)] を選択することにより、該当実行モジュールの性能解析情報（コールペア情報、コールカウント情報など）をグラフ形式で一覧表示するためのウィンドウ（[コールグラフ・ウィンドウ](#)）がオープンします。

(g) [関数情報テーブル] 項目

[オリジナル実行モジュール情報] 項目、[チューニング後実行モジュール情報] 項目の子項目です。

なお、本項目を選択した後、右クリック→[開く (Q)] を選択、または、[表示 (V)] メニュー→[開く (Q)] を選択することにより、該当実行モジュールの関数詳細情報を表示するためのウィンドウ（[関数情報テーブル・ウィンドウ](#)）がオープンします。

(h) [関数遷移情報] 項目

[オリジナル実行モジュール情報] 項目、[チューニング後実行モジュール情報] 項目の子項目です。

なお、本項目を選択した後、右クリック→[開く (Q)] を選択、または、[表示 (V)] メニュー→[開く (Q)] を選択することにより、該当実行モジュールの関数遷移情報を表示するためのウィンドウ（[関数遷移情報ウィンドウ](#)）がオープンします。

(4) 内部ウィンドウ表示領域

内部ウィンドウ ([コールグラフ・ウィンドウ](#), [関数情報テーブル・ウィンドウ](#), [関数遷移情報ウィンドウ](#), [実行モジュール情報ウィンドウ](#), [関数詳細情報ウィンドウ](#) など) を表示するための領域です。

なお、本表示領域の表示形式は、[ウィンドウ (W)] メニューを選択することにより表示される項目 (重ねて表示、並べて表示など) で変更することができます。

(5) メッセージ表示領域

TW850 が検出した “ ユーザに通知すべき情報 ” を表示するための領域です。

なお、本表示領域の表示 / 非表示は、[表示 (V)] メニュー → [メッセージ (M)] で切り替えることができます。

(6) ステータス・バー

TW850 の処理の進捗状況を表示するための領域です。

実行に時間がかかる処理を行っている場合、プログレス・バーを表示します。

コンテキスト・メニュー

履歴表示領域の表示項目を選択後に右クリックを行った場合、コンテキスト・メニューが表示されます。

コンテキスト・メニューは、以下に示した項目から構成されています。

(1) [開く (O)]

選択項目に対応したウィンドウ ([コールグラフ・ウィンドウ](#), [関数情報テーブル・ウィンドウ](#), [実行モジュール情報ウィンドウ](#)) がオープンします。

(2) [履歴に追加 (H)]

選択項目を [履歴情報] 項目に追加登録します。

(3) [履歴から復帰 (R)...]

選択項目の設定内容 (性能解析情報、チューニング情報など) が TW850 に反映されます。

(4) [削除 (D)]

選択項目を削除します。

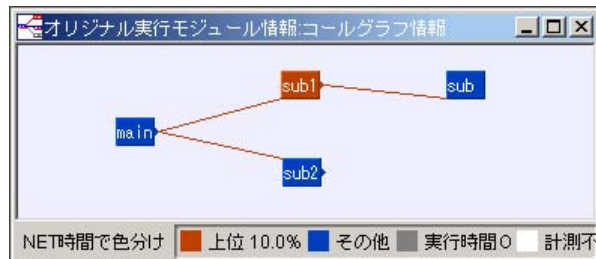
(5) [名前の変更 (N)]

選択項目の名前を変更するためのダイアログがオープンします。

コールグラフ・ウィンドウ

実行モジュールの性能解析情報（コールペア情報，コールカウント情報など）をグラフ形式で一覧表示するためのウィンドウです。

図 6-2 コールグラフ・ウィンドウ



ここでは，次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)
- [コンテキスト・メニュー](#)

オープン方法

本ウィンドウは，次の方法でオープンすることができます。

- [メイン・ウィンドウ](#)の履歴表示領域において，[コールグラフ情報] 項目を選択した後，
 - 右クリック→[開く (O)] を選択したとき
 - [表示 (V)] メニュー→[開く (O)] を選択したとき

エリア説明

本ウィンドウは，次の項目から構成されています。

- [コールグラフ表示領域](#)
- [色分け基準表示領域](#)

(1) コールグラフ表示領域

性能解析処理の実行結果に関する情報 (コールペア情報, コールカウント情報など) をグラフ形式で表示する領域です。

背景色は, オリジナル実行モジュール情報の場合は水色, チューニング後実行モジュール情報の場合はピンク色となります。

なお, 本表示領域は, 以下に示したボックス, 関係線から構成されています。

(a) 関数ボックス

対象関数の関数名を色分け表示します。

なお, 関数名が 20 文字を超える場合, 18 文字目以降は “ ... ” と表示されます。

また, 対象関数が関数呼び出し (子関数の呼び出し) を行っている場合は, 関数ボックスの右端に “ ● ” が付加表示されます。

白色 : 計測不能

灰色 : 不実行の性能解析対象関数

赤色 : 上位 xxx% 以上の性能解析対象関数

青色 : 上位 xxx% 以下の性能解析対象関数

注意 1 色分け境界値 xxx% は, [メイン・ウィンドウ](#)において, [表示 (V)] メニュー → [コールグラフ表示方法 (C)] → [色分け境界設定 (I)...] を選択することによりオープンする “ [入力ダイアログ](#) ” で変更することができます。

注意 2 関数ボックスをクリックした場合, 関数ボックスは選択状態となり, 対象関数の親関数ボックス, 対象関数の子関数ボックス, および, ルート関数ボックスから対象関数ボックスに至るまでの呼び出し関係線が強調表示されます。

また, 対象関数を表示しているコールグラフ・ウィンドウ, または, [関数情報テーブル・ウィンドウ](#) が他に存在していた場合, TW850 は各対象関数に対する選択操作を実行します。

注意 3 関数ボックスをダブル・クリックした場合, 対象関数の詳細情報を表示するための [関数詳細情報ウィンドウ](#) がオープンします。

なお, 関数ボックスをドラッグすることにより, 対象関数ボックスを移動することができます。

注意 4 関数ボックス, および, 呼び出し関係線以外の領域をクリックした場合, すべての関数ボックスが非選択状態となります。

注意 5 上位とは, [メイン・ウィンドウ](#)の [コールグラフ表示方法 (C)] で指定された色分け表示形式 (NET 時間で色分け, GROSS 時間で色分けなど) のソート結果 “ 上位 ” を意味しています。

(b) 呼び出し関係線

関数間 (親関数, および, 子関数) の呼び出し関係を色分け表示します。

灰色 : 対象子関数の呼び出し回数が 0

赤色 : 対象子関数の呼び出し回数が性能解析対象範囲の総呼び出し回数の上位 xxx% 以上

青色 : 対象子関数の呼び出し回数が性能解析対象範囲の総呼び出し回数の上位 xxx% 以下

注意 1 色分け境界値 xxx% は, [メイン・ウィンドウ](#)において, [表示 (V)] メニュー → [コールグラフ表示方法 (C)] → [色分け境界設定 (I)...] を選択することによりオープンする “ [入力ダイアログ](#) ” で変更することができます。

注意 2 呼び出し関係線が右側中央から出ている関数ボックスは親関数, 左側角から出ている関数ボックスは子関数であることを意味しています。

なお, 再帰関数の場合, TW850 では右側中央から左上角に曲線を描くことにより “ 再帰関数 ” を表現しています。

注意 3 呼び出し関係線をクリックした場合, すべての関数ボックスが非選択状態となります。

注意 4 上位とは, [メイン・ウィンドウ](#)の [コールグラフ表示方法 (C)] で指定された色分け表示形式 (NET 時間で色分け, GROSS 時間で色分けなど) のソート結果 “ 上位 ” を意味しています。

(2) 色分け基準表示領域

関数ボックスの色分け基準 (NET 時間で色分け, GROSS 時間で色分け, 実行回数で色分け, 命令キャッシュ・ミス数で色分け) を表示します。

なお, 関数ボックスの色分け基準は, [メイン・ウィンドウ](#)において, [表示 (V)] メニュー → [コールグラフ表示方法 (C)] を選択することにより表示される項目 (NET 時間で色分け, GROSS 時間で色分け, 実行回数で色分け, 命令キャッシュ・ミス数で色分け) で変更することができます。

コンテキスト・メニュー

以下の領域で右クリックを行った場合、コンテキスト・メニューが表示されます。

(1) 関数ボックス

関数ボックス上で右クリックを行った場合に表示されるコンテキスト・メニューは、以下に示した項目から構成されています。

(a) [ルート関数に設定 (R)]

選択位置の関数をルート関数として、コールグラフ表示領域に再描画します。

(b) [関数詳細情報表示 (D)]

選択位置の関数について、関数詳細情報 (性能解析情報、親関数情報、子関数情報) を表示するためのウィンドウ ([関数詳細情報ウィンドウ](#)) がオープンします。

(c) [展開 (N)], [折りたたみ (N)]

選択位置の関数ボックスが展開状態、または、折りたたみ状態となります。

(2) 関数ボックス以外の領域

関数ボックス以外の領域で右クリックを行った場合に表示されるコンテキスト・メニューは、以下に示した項目から構成されています。

(a) [NET 時間で色分け]

関数ボックスの色分け基準を NET 時間で行うか否かを指定します。

(b) [GROSS 時間で色分け]

関数ボックスの色分け基準を GROSS 時間で行うか否かを指定します。

(c) [実行回数で色分け (C)]

関数ボックスの色分け基準を実行回数で行うか否かを指定します。

(d) [命令キャッシュ・ミス数で色分け (I)]

関数ボックスの色分け基準を命令キャッシュ・ミス数で行うか否かを指定します。

(e) [色分け境界設定 (I)...]

関数ボックスの色分け境界値を指定するためのダイアログ ([入カダイアログ](#)) がオープンします。

(f) [ズーム (Z)...]

コールグラフ表示領域の表示倍率を変更するためのウィンドウ ([入カダイアログ](#)) がオープンします。

(g) [ルート関数変更 (R)...]

コールグラフ表示領域に表示されているルート関数を変更するためのダイアログ ([入カダイアログ](#)) がオープンします。

(h) [実行時間 0 の関数を表示しない (H)], [実行回数 0 の関数を表示しない (H)]

実行時間 0、または、実行回数 0 の関数を表示するか否かを指定します。

関数情報テーブル・ウィンドウ

実行モジュールの関数情報 (関数名, 実行回数, NET 時間など) をテーブル形式で一覧表示するためのウィンドウです。

図 6-3 関数情報テーブル・ウィンドウ

オリジナル実行モジュール情報: 関数情報テーブル							
関数名	実行回数	NET時間[ns]	平均NET時間[ns]	GROSS時間[ns]	平均GROSS時間[ns]	不明関数実行時間[ns]	命令キャッシュ・ミス数
sub1	1	42,080 (40.77%)	42,080 (40.77%)	54,560 (52.86%)	54,560 (52.86%)	0 (0.00%)	4 (1.90%)
func	5	20,320 (19.68%)	4,064 (3.93%)	20,320 (19.68%)	4,064 (3.93%)	0 (0.00%)	1 (1.17%)
sub2	1	19,280 (18.68%)	19,280 (18.68%)	39,600 (38.37%)	39,600 (38.37%)	0 (0.00%)	4 (18.18%)
sub	1	12,480 (12.09%)	12,480 (12.09%)	12,480 (12.09%)	12,480 (12.09%)	0 (0.00%)	2 (7.69%)
main	1	9,040 (8.75%)	9,040 (8.75%)	103,200 (100.00%)	103,200 (100.00%)	0 (0.00%)	2 (22.22%)
_sub	0	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (---.---%)
_sub2	0	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (---.---%)
不明関数	0	0 (0.00%)					0 (---.---%)

ここでは、次の項目について説明しています。

- ・ [オープン方法](#)
- ・ [エリア説明](#)
- ・ [コンテキスト・メニュー](#)

オープン方法

本ウィンドウは、次の方法でオープンすることができます。

- ・ [メイン・ウィンドウ](#)の履歴表示領域において、[関数情報テーブル] 項目を選択した後、
 - 右クリック → [開く (O)] を選択したとき
 - [表示 (V)] メニュー → [開く (O)] を選択したとき

エリア説明

本ウィンドウは、次の項目から構成されています。

- ・ [関数情報表示領域](#)

(1) 関数情報表示領域

性能解析処理の実行結果に関する情報 (関数情報: 関数名, 実行回数, NET 時間など) をテーブル形式で一覧表示する領域です。

背景色は, オリジナル実行モジュール情報の場合は水色, チューニング後実行モジュール情報の場合はピンク色となります。

なお, 本表示領域は, 以下に示した表示領域から構成されています。

(a) 関数名表示領域

対象関数の関数名を表示します。

注意 同じ名前の関数が複数存在する場合, “ 関数名 (ファイル名) ” のように表示されます。

また, 関数名, および, ファイル名が同じ名前である関数が複数存在する場合, “ 関数名 (絶対パス名) ” のように表示されます。

(b) 実行回数表示領域

対象関数の実行回数を表示します。

なお, オーバフローしている場合は, 実行回数のあとに “ (オーバフロー) ” と表示されます。

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- 性能解析処理において, プローブ・コード挿入方式を用いた場合, かつ, チューニング後実行モジュール情報の場合

(c) NET 時間 (子関数の実行時間を含まない) 表示領域

対象関数の NET 時間 (単位: ns, または, clock), および, 性能解析対象範囲の総実行時間に対する割合 (単位: %) を表示します。

対象関数がチューニング後実行モジュールであった場合, オリジナル解析用実行モジュールとの性能比較結果が色分け表示されます。

なお, オーバフローしている場合は, 実行時間のあとに “ (オーバフロー) ” と表示されます。

青色 : オリジナル解析用実行モジュールよりも NET 時間が短い

黒色 : オリジナル解析用実行モジュールの NET 時間と差がない

赤色 : オリジナル解析用実行モジュールよりも NET 時間が長い

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- 性能解析処理において, プローブ・コード挿入方式を用いた場合, かつ, 時間計測用ファイルを用いて対象関数ごとの実行時間計測処理を行わなかった場合

(d) 平均 NET 時間表示領域

対象関数の平均 NET 時間 (単位: ns, または, clock), および, 性能解析対象範囲の総実行時間に対する割合 (単位: %) を表示します。

対象関数がチューニング後実行モジュールであった場合, オリジナル解析用実行モジュールとの性能比較結果が色分け表示されます。

なお, オーバフローしている場合は, 実行時間のあとに “ (オーバフロー) ” と表示されます。

青色 : オリジナル解析用実行モジュールよりも平均 NET 時間が短い

黒色 : オリジナル解析用実行モジュールの平均 NET 時間と差がない

赤色 : オリジナル解析用実行モジュールよりも平均 NET 時間が長い

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- NET 時間表示領域が非表示状態

(e) GROSS 時間 (子関数の実行時間を含む) 表示領域

対象関数の GROSS 時間 (単位: ns, または, clock), および, 性能解析対象範囲の総実行時間に対する割合 (単位: %) を表示します。

対象関数がチューニング後実行モジュールであった場合, オリジナル解析用実行モジュールとの性能比較結果が色分け表示されます。

なお, オーバフローしている場合は, 実行時間のあとに “ (オーバフロー) ” と表示されます。

青色 : オリジナル解析用実行モジュールよりも GROSS 時間が短い

黒色 : オリジナル解析用実行モジュールの GROSS 時間と差がない

赤色 : オリジナル解析用実行モジュールよりも GROSS 時間が長い

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- 性能解析処理において, プローブ・コード挿入方式を用いた場合

(f) 平均 GROSS 時間表示領域

対象関数の平均 GROSS 時間 (単位: ns, または, clock), および, 性能解析対象範囲の総実行時間に対する割合 (単位: %) を表示します。

対象関数がチューニング後実行モジュールであった場合, オリジナル解析用実行モジュールとの性能比較結果が色分け表示されます。

なお, オーバフローしている場合は, 実行時間のあとに“(オーバーフロー)”と表示されます。

青色 : オリジナル解析用実行モジュールよりも平均 GROSS 時間が短い

黒色 : オリジナル解析用実行モジュールの平均 GROSS 時間と差がない

赤色 : オリジナル解析用実行モジュールよりも平均 GROSS 時間が長い

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- GROSS 時間表示領域が非表示状態

(g) 不明関数 (性能解析対象外関数) 実行時間表示領域

対象関数から呼び出された不明関数の GROSS 時間 (単位: ns, または, clock), および, 性能解析対象範囲の総実行時間に対する割合 (単位: %) を表示します。

対象関数がチューニング後実行モジュールであった場合, オリジナル解析用実行モジュールとの性能比較結果が色分け表示されます。

なお, オーバフローしている場合は, 実行時間のあとに“(オーバーフロー)”と表示されます。

青色 : オリジナル解析用実行モジュールよりも不明関数実行時間が短い

黒色 : オリジナル解析用実行モジュールの不明関数実行時間と差がない

赤色 : オリジナル解析用実行モジュールよりも不明関数実行時間が長い

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- 性能解析処理において, プローブ・コード挿入方式を用いた場合

(h) 命令キャッシュ・ミス表示領域

対象関数における命令キャッシュ・ミスの回数, および, 発生率を表示します。

なお, 発生率は, 対象関数における命令キャッシュ・ヒットの回数, および, 命令キャッシュ・ミスの回数をもとに, 下記計算式から算出された値となります。

命令キャッシュ・ミスの回数 / (命令キャッシュ・ヒットの回数 + 命令キャッシュ・ミスの回数)

注意 本表示領域は, 以下に示した何れかの条件が成立した場合に非表示状態となります。

- 性能解析処理において, プローブ・コード挿入方式を用いた場合
- 性能解析処理において, トレース・データ解析方式, かつ, トレース・データからキャッシュ・ミス情報が取得可能な実行エンジンを指定していない

注意 1 各項目の情報を取得する設定がされていない場合や実行エンジンにより取得できない場合, その項目は表示されません。

注意 2 各項目のヘッダ部分をクリックすることにより, 関数情報を該当項目を基準としたソート表示することができます。

ただし, 不明関数, および, RTOS のシステム部分については, ソート対象から除外されています。

また, 各項目のヘッダ部分をドラッグした場合, 各項目の表示順番を入れ替えることができます。

注意 3 各関数の情報部分をダブル・クリックした場合, 対象関数の詳細情報を表示するための[関数詳細情報ウィンドウ](#)がオープンします。

なお, 関数ボックスをドラッグすることにより, 対象関数ボックスを移動することができます。

コンテキスト・メニュー

関数情報テーブル表示領域内で該当関数を選択後に右クリックを行った場合, コンテキスト・メニューが表示されます。コンテキスト・メニューは, 以下に示した項目から構成されています。

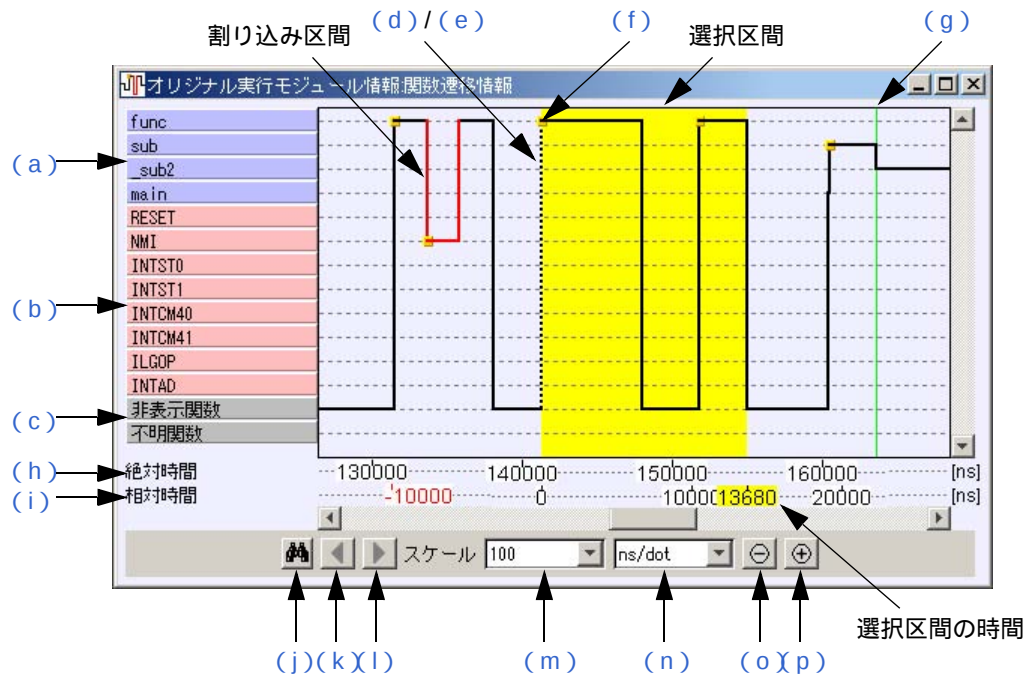
(1) [関数詳細情報表示 (D)]

該当関数の関数詳細情報 (性能解析情報, 親関数情報, 子関数情報) を表示するためのウィンドウ ([関数詳細情報ウィンドウ](#)) がオープンします。

関数遷移情報ウィンドウ

実行モジュールの関数遷移情報をグラフ形式で一覧表示するためのウィンドウです。

図 6-4 関数遷移情報ウィンドウ



ここでは、次の項目について説明しています。

- オープン方法
- エリア説明
- コンテキスト・メニュー

オープン方法

本ウィンドウは、次の方法でオープンすることができます。

- **メイン・ウィンドウ**の履歴表示領域において、[関数遷移情報] 項目を選択した後、
 - 右クリック→[開く (O)] を選択したとき
 - [表示 (V)] メニュー→[開く (O)] を選択したとき

エリア説明

本ウィンドウは、次の項目から構成されています。

- 関数遷移情報表示領域

(1) 関数遷移情報表示領域

性能解析処理の実行結果に関する情報（関数遷移情報）をグラフ形式で一覧表示する領域です。

背景色は、オリジナル実行モジュール情報の場合は水色、チューニング後実行モジュール情報の場合はピンク色となります。

なお、本表示領域は、以下に示した領域から構成されています。

(a) 関数ボックス

対象関数の関数名（関数シンボル）を表示します。

なお、関数名が 20 文字を超える場合、18 文字目以降は“...”と表示されます。

注意 1 関数ボックスをクリックした場合、関数ボックスは選択状態となり、表示している [コールグラフ・ウィンドウ](#)、[関数情報テーブル・ウィンドウ](#)、または、他の関数遷移情報ウィンドウがあれば、それぞれの同じ関数が選択状態となります。

注意 2 関数ボックスをダブル・クリックした場合、対象関数の詳細情報を表示するための [関数詳細情報ウィンドウ](#) がオープンします。
なお、関数ボックスをドラッグすることにより、関数ボックス間において、対象関数ボックスを移動することができます。

(b) 割り込みボックス

対象割り込みの割り込み名を表示します。

注意 1 割り込みボックスをクリックした場合、割り込みボックスは選択状態となり、表示している他の関数遷移情報ウィンドウがあれば、それぞれの同じ割り込みボックスが選択状態となります。

注意 2 割り込みボックスをドラッグすることにより、割り込みボックス間において、対象割り込みボックスを移動することができます。

(c) その他のボックス

“非表示関数”、RTOS システム名、“不明関数”を表示します。

なお、RTOS システム名は、認識する RTOS によって異なります。認識する RTOS が RX850 の場合は“RX850 システム”、RX850 Pro の場合は“RX850 Pro システム”となります。

注意 その他のボックスをクリックした場合、そのボックスは選択状態となり、表示している他の関数遷移情報ウィンドウがあれば、それぞれの同じボックスが選択状態となります。

(d) 関数実行線

関数の実行を表示します。

なお、割り込み区間は赤色表示、それ以外の区間は黒色表示されます。

注意 1 関数実行線をクリックした場合、その関数実行線、および、関数ボックスは選択状態となり、表示している [コールグラフ・ウィンドウ](#)、[関数情報テーブル・ウィンドウ](#)、または、他の関数遷移情報ウィンドウがあれば、それぞれの同じ関数が選択状態となります。

注意 2 関数実行線をダブル・クリックした場合、対象関数の詳細情報を表示するための [関数詳細情報ウィンドウ](#) がオープンします。

注意 3 関数実行線を選択した後、ドラッグした場合、選択した関数実行線の先頭から関数遷移単位で区間が選択状態となり、選択区間にマウス・カーソルを当てた状態にすると、選択区間の時間（単位：ナノ秒）がポップ・アップ表示されます。

注意 4 関数実行線にマウス・カーソルを当てた状態にすると、関数名（割り込み名）、開始時刻、実行時間がポップ・アップ表示されます。

(e) 関数遷移線

関数の遷移を表示します。

なお、割り込み区間は赤色表示、それ以外の区間は黒色表示されます。

注意 1 関数遷移線をクリックした場合、その関数遷移線は選択状態となります。

注意 2 関数遷移線を選択した後、ドラッグした場合、選択した関数遷移線から関数遷移単位で区間が選択状態となり、選択区間にマウス・カーソルを当てた状態にすると、選択区間の時間（単位：ナノ秒）がポップ・アップ表示されます。

注意 3 関数遷移線にマウス・カーソルを当てた状態にすると、遷移元関数名（割り込み名）、遷移先関数名（割り込み名）、遷移時刻がポップ・アップ表示されます。

(f) 関数先頭マーカ

遷移した関数、および、割り込みの先頭に表示します。

注意 関数先頭マーカは、非表示関数、RTOS システム、不明関数については表示しません。

(g) トレース・データの切れ目線

性能解析対象区間において、トレース・データの切れ目が発生した時間位置を表示します。

(h) 絶対時間スケール

性能解析対象区間の先頭を基準に、時刻のスケールを表示します。

なお、時刻表示間隔は 100 ドットごとに固定、表示単位は単位コンボ・ボックスで指定した時間単位となります。

(i) 相対時間スケール

選択状態である関数実行線の見頭、または、関数遷移線を基準に、時刻のスケールを表示します。

なお、時刻表示間隔は 100 ドットごとに固定、表示単位は単位コンボ・ボックスで指定した時間単位となります。また、相対時間が負の値となる場合、赤色表示されます。

注意 選択区間が存在する場合、スケール上に選択区間の時間が表示されます。

(j) 検索ボタン

関数の遷移を検索するためのダイアログ (関数遷移情報検索ダイアログ) がオープンします。

注意 本ボタンにマウス・カーソルを当てた状態にすると、“関数や関数遷移を検索します。”がポップ・アップ表示されます。

(k) 前を検索ボタン

選択状態である関数実行線、または、関数遷移線を基準に、関数遷移情報検索ダイアログで指定された条件 (関数、または、割り込み) の関数実行線、または、関数遷移線を左方向に検索します。

注意 本ボタンにマウス・カーソルを当てた状態にすると、“前の関数や関数遷移を検索します。”がポップ・アップ表示されます。

(l) 次を検索ボタン

選択状態である関数実行線、または、関数遷移線を基準に、関数遷移情報検索ダイアログで指定された条件 (関数、または、割り込み) の関数実行線、または、関数遷移線を右方向に検索します。

注意 本ボタンにマウス・カーソルを当てた状態にすると、“次の関数や関数遷移を検索します。”がポップ・アップ表示されます。

(m) スケール・コンボ・ボックス

関数遷移情報表示領域の 1 ドットが示すクロック、または、時間を指定します。

選択項目は、1, 2, 5, 10, 25, 50, 100, 250, 500, 1000 となります。

また、テキスト入力も可能です。入力可能な値の範囲は、1 ~ 1000 となります。

デフォルトでは、1 が指定されます。

注意 本コンボ・ボックスにマウス・カーソルを当てた状態にすると、“1 ドットあたりのクロック数または時間を指定します。”がポップ・アップ表示されます。

(n) 単位コンボ・ボックス

関数遷移情報表示領域の 1 ドットが示すクロック、または、時間の単位を指定します。

選択項目は、clock/dot, Kclock/dot, Mclock/dot, ns/dot, us/dot, ms/dot となります。

デフォルトでは、実行時間がナノ秒で取得可能ならば ns/dot、クロックのみ取得可能ならば clock/dot が指定されます。

注意 1 単位の大きさは、以下のようになります。

Mclock/dot, ms/dot > Kclock/dot, us/dot > clock/dot, ns/dot

なお、単位コンボ・ボックスの指定項目を変更することにより、スケール・コンボ・ボックスの指定項目が変更されます。単位を小さく変更した場合は 1000、大きく変更した場合は 1 に変更されます。

注意 2 本コンボ・ボックスにマウス・カーソルを当てた状態にすると、“1 ドットあたりのクロック数または時間の単位を指定します。”がポップ・アップ表示されます。

(o) 縮小ボタン

関数遷移情報表示領域のスケール・コンボ・ボックスの指定項目を以下の順番で変更し、表示範囲を縮小します。

1, 2, 5, 10, 25, 50, 100, 250, 500, 1000

注意 1 スケール・コンボ・ボックスの指定項目が上記以外の場合、上記の内で最も近い大きい値に変更されます。また、スケール・コンボ・ボックスの指定項目が 1000 である場合、単位コンボ・ボックスの指定項目を 1 つ大きくし、スケール・コンボ・ボックスの指定項目が 2 に変更されます。

注意 2 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- スケール・コンボ・ボックスにおいて、1000 が選択状態、かつ、単位コンボ・ボックスにおいて、Mclock/dot、または、ms/dot が選択状態

注意 3 本ボタンにマウス・カーソルを当てた状態にすると、“縮小”がポップ・アップ表示されます。

(p) 拡大ボタン

関数遷移情報表示領域のスケール・コンボ・ボックスの指定項目を以下の順番で変更し、表示範囲を拡大します。

1000, 500, 250, 100, 50, 25, 10, 5, 2, 1

注意 1 スケール・コンボ・ボックスの指定項目が上記以外の場合、上記の内で最も近い小さい値に変更されます。また、スケール・コンボ・ボックスの指定項目が 1 である場合、単位コンボ・ボックスの指定項目を 1 つ小さくし、スケール・コンボ・ボックスの指定項目が 500 に変更されます。

注意 2 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- スケール・コンボ・ボックスにおいて、1 が選択状態、かつ、単位コンボ・ボックスにおいて、clock/dot、または、ns/dot が選択状態

注意 3 本ボタンにマウス・カーソルを当てた状態にすると、“拡大”がポップ・アップ表示されます。

コンテキスト・メニュー

以下の領域で右クリックを行った場合、コンテキスト・メニューが表示されます。

(1) 関数ボックス

関数ボックス上で右クリックを行った場合に表示されるコンテキスト・メニューは、以下に示した項目から構成されています。

(a) [この関数を表示しない (H)]

該当関数を非表示にして、関数遷移情報表示領域を再描画します。

(b) [関数詳細情報表示 (D)]

選択位置の関数について、関数詳細情報 (性能解析情報、親関数情報、子関数情報) を表示するためのウィンドウ ([関数詳細情報ウィンドウ](#)) がオープンします。

(2) 割り込みボックス

割り込みボックス上で右クリックを行った場合に表示されるコンテキスト・メニューは、以下に示した項目から構成されています。

(a) [この割り込みを表示しない (H)]

該当割り込みを非表示にして、関数遷移情報表示領域を再描画します。

(3) 関数ボックス、割り込みボックス以外の領域

関数ボックス、割り込みボックス以外の領域で右クリックを行った場合に表示されるコンテキスト・メニューは、以下に示した項目から構成されています。

(a) [ソート (S)]

関数ボックス、および、割り込みボックスのソート条件を選択します。

本項目は、以下に示した項目から構成されています。

- [名前順 (N)]

関数ボックス、および、割り込みボックスを名前順にソートします。

ただし、関数ボックス、割り込みボックス、その他のボックスの表示順は変わりません。

- [アドレス順 (A)]

関数ボックス、および、割り込みボックスをアドレス順にソートします。

ただし、関数ボックス、割り込みボックス、その他のボックスの表示順は変わりません。

- [ファイル順 (E)]

関数ボックス，および，割り込みボックスをファイル順にソートします。
ただし，関数ボックス，割り込みボックス，その他のボックスの表示順は変わりません。

(b) [表示関数選択 (D)...]

関数ボックス，および，割り込みボックスを選択するためのダイアログ ([表示関数選択ダイアログ](#)) がオープンします。

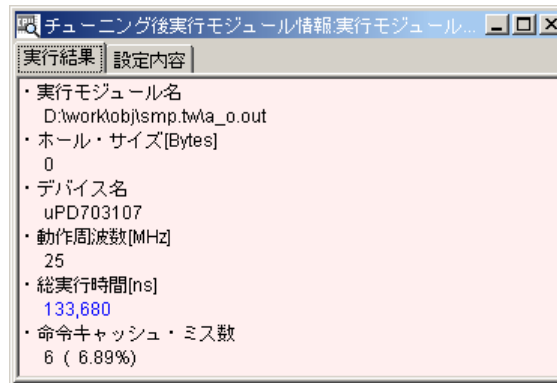
(c) [非表示関数に関する遷移線を表示しない (H)]

非表示となっている関数に関する関数遷移線を表示するか否かを指定します。
なお，本項目がチェック状態の場合，“非表示となっている関数に関する関数遷移線を表示する”となります。

実行モジュール情報ウィンドウ

実行モジュールの実行モジュール情報 (実行結果, 設定内容) を表示するためのウィンドウです。

図 6-5 実行モジュール情報ウィンドウ



ここでは, 次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ウィンドウは, 次の方法でオープンすることができます。

- [メイン・ウィンドウ](#)の履歴表示領域において, [実行モジュール情報] 項目を選択した後,
 - 右クリック→ [開く (O)] を選択したとき
 - [表示 (V)] メニュー→ [開く (O)] を選択したとき

エリア説明

本ウィンドウは, 次の項目から構成されています。

- [\[実行結果 \] タブ](#)
- [\[設定内容 \] タブ](#)

(1) [実行結果] タブ

性能解析処理の実行結果に関する情報 (実行結果情報: 実行モジュール名, デバイス名, 総実行時間など) を表示するタブです。

背景色は, オリジナル実行モジュール情報の場合は水色, チューニング後実行モジュール情報の場合はピンク色となります。

なお, 本タブは, 以下に示した表示領域から構成されています。

(a) 実行モジュール名表示領域

実行モジュールのファイル名 (実行モジュール名) を表示します。

(b) ホール・サイズ表示領域

チューニング処理により発生したホール・サイズを表示します。

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- オリジナル実行モジュール情報の場合

(c) デバイス名表示領域

実行モジュールの対象デバイス名を表示します。

(d) 動作周波数表示領域

実行モジュールの動作周波数 (単位: MHz) を表示します。

(e) 総実行時間表示領域

性能解析対象範囲の総実行時間 (単位: ns, または, clock) を表示します。

なお, 該当実行モジュールがチューニング後実行モジュールであった場合, オリジナル解析用実行モジュールとの性能比較結果が色分け表示されます。

青色 : オリジナル解析用実行モジュールよりも総実行時間が短い

黒色 : オリジナル解析用実行モジュールの総実行時間と差がない

赤色 : オリジナル解析用実行モジュールよりも総実行時間が長い

(f) 命令キャッシュ・ミス情報表示領域

性能解析対象範囲における命令キャッシュ・ミスの回数, および, 発生率を表示します。

なお, 発生率は, 性能解析対象範囲における命令キャッシュ・ヒットの回数, および, 命令キャッシュ・ミスの回数をもとに, 下記計算式から算出された値となります。

命令キャッシュ・ミスの回数 / (命令キャッシュ・ヒットの回数 + 命令キャッシュ・ミスの回数)

注意 本表示領域は, 以下に示した何れかの条件が成立した場合に非表示状態となります。

- 性能解析処理において, プローブ・コード挿入方式を用いた場合
- 性能解析処理において, トレース・データ解析方式, かつ, トレース・データからキャッシュ・ミス情報, および, キャッシュ・ヒット情報が取得可能な実行エンジンを指定していない

(2) [設定内容] タブ

性能解析処理の設定内容に関する情報 (設定内容情報: 実行内容, 実行開始位置, 実行解析範囲など) を表示するタブです。

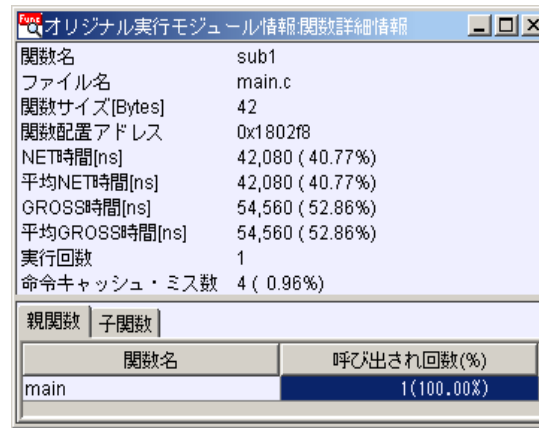
ウィザードの各パネルにおいて設定した内容を表示します。

背景色は, オリジナル実行モジュール情報の場合は水色, チューニング後実行モジュール情報の場合はピンク色となります。

関数詳細情報ウィンドウ

実行モジュールの関数詳細情報（性能解析情報，親関数情報，子関数情報）を表示するためのウィンドウです。

図 6-6 関数詳細情報ウィンドウ



ここでは，次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ウィンドウは，次の方法でオープンすることができます。

- [コールグラフ・ウィンドウ](#)のコールグラフ表示領域に表示されている関数名を選択した後，[メイン・ウィンドウ](#)において，
 - 関数詳細表示ボタンを押下したとき
 - [表示 (V)] メニュー → [関数詳細情報 (D)] を選択したとき
- [コールグラフ・ウィンドウ](#)において，
 - コールグラフ表示領域に表示されている関数名を選択 (ダブル・クリック) したとき
- [関数情報テーブル・ウィンドウ](#)の関数名表示領域に表示されている関数名を選択した後，[メイン・ウィンドウ](#)において，
 - 関数詳細表示ボタンを押下したとき
 - [表示 (V)] メニュー → [関数詳細情報 (D)] を選択したとき
- [関数情報テーブル・ウィンドウ](#)において，
 - 関数名表示領域に表示されている関数名を選択 (ダブル・クリック) したとき
- [関数遷移情報ウィンドウ](#)の関数遷移情報表示領域に表示されている関数実行線を選択した後，[メイン・ウィンドウ](#)において，
 - 関数詳細表示ボタンを押下したとき
 - [表示 (V)] メニュー → [関数詳細情報 (D)] を選択したとき

- 関数遷移情報ウィンドウにおいて ,
 - 関数遷移情報表示領域に表示されている関数ボックスを選択 (ダブル・クリック) したとき
 - 関数遷移情報表示領域に表示されている関数実行線を選択 (ダブル・クリック) したとき
- 本ウィンドウの [親関数] タブ , または , [子関数] タブに表示されている関数名を選択した後 , メイン・ウィンドウにおいて ,
 - 関数詳細表示ボタンを押下したとき
 - [表示 (V)] メニュー → [関数詳細情報 (D)] を選択したとき
- 本ウィンドウにおいて ,
 - [親関数] タブ , または , [子関数] タブに表示されている関数名を選択 (ダブル・クリック) したとき

エリア説明

本ウィンドウは , 次の項目から構成されています。

- 性能解析情報表示領域
- [親関数] タブ
- [子関数] タブ

(1) 性能解析情報表示領域

性能解析処理の実行結果に関する情報 (性能解析情報 : 関数名 , ファイル名 , 関数サイズなど) を表示する領域です。背景色は , オリジナル実行モジュール情報の場合は水色 , チューニング後実行モジュール情報の場合はピンク色となります。

なお , 本表示領域は , 以下に示した表示領域から構成されています。

(a) 関数名表示領域

対象関数の関数名を表示します。

(b) ファイル名表示領域

対象関数が記述されているファイル名を表示します。

注意 関数がライブラリ内で定義された関数である場合 “ファイル名 (ライブラリ名)” のように表示されます。

(c) 関数サイズ表示領域

対象関数がテキスト領域として占めるサイズ (単位 : バイト) を表示します。

(d) 関数配置アドレス表示領域

対象関数のテキスト領域が割り付けられているアドレス (先頭アドレス) を表示します。

(e) NET 時間 (子関数の実行時間を含まない) 表示領域

対象関数の NET 時間 (単位 : ns , または , clock) , および , 性能解析対象範囲の総実行時間に対する割合 (単位 : %) を表示します。

対象関数がチューニング後実行モジュールであった場合 , オリジナル解析用実行モジュールとの性能比較結果が色分け表示されます。

なお , オーバフローしている場合は , 実行時間のあとに “ (オーバフロー) ” と表示されます。

青色 : オリジナル解析用実行モジュールよりも NET 時間が短い

黒色 : オリジナル解析用実行モジュールの NET 時間と差がない

赤色 : オリジナル解析用実行モジュールよりも NET 時間が長い

注意 本表示領域は , 以下に示した条件が成立した場合に非表示状態となります。

- 性能解析処理において , プローブ・コード挿入方式を用いた場合 , かつ , 時間計測用ファイルを用いて対象関数ごとの実行時間計測処理を行わなかった場合

(f) 平均 NET 時間表示領域

対象関数の平均 NET 時間 (単位: ns, または, clock), および, 性能解析対象範囲の総実行時間に対する割合 (単位: %) を表示します。

対象関数がチューニング後実行モジュールであった場合, オリジナル解析用実行モジュールとの性能比較結果が色分け表示されます。

なお, オーバフローしている場合は, 実行時間のあとに“(オーバーフロー)”と表示されます。

青色 : オリジナル解析用実行モジュールよりも平均 NET 時間が短い

黒色 : オリジナル解析用実行モジュールの平均 NET 時間と差がない

赤色 : オリジナル解析用実行モジュールよりも平均 NET 時間が長い

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- NET 時間表示領域が非表示状態

(g) GROSS 時間 (子関数の実行時間を含む) 表示領域

対象関数の GROSS 時間 (単位: ns, または, clock), および, 性能解析対象範囲の総実行時間に対する割合 (単位: %) を表示します。

対象関数がチューニング後実行モジュールであった場合, オリジナル解析用実行モジュールとの性能比較結果が色分け表示されます。

なお, オーバフローしている場合は, 実行時間のあとに“(オーバーフロー)”と表示されます。

青色 : オリジナル解析用実行モジュールよりも GROSS 時間が短い

黒色 : オリジナル解析用実行モジュールの GROSS 時間と差がない

赤色 : オリジナル解析用実行モジュールよりも GROSS 時間が長い

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- 性能解析処理において, プローブ・コード挿入方式を用いた場合

(h) 平均 GROSS 時間表示領域

対象関数の平均 GROSS 時間 (単位: ns, または, clock), および, 性能解析対象範囲の総実行時間に対する割合 (単位: %) を表示します。

対象関数がチューニング後実行モジュールであった場合, オリジナル解析用実行モジュールとの性能比較結果が色分け表示されます。

なお, オーバフローしている場合は, 実行時間のあとに“(オーバーフロー)”と表示されます。

青色 : オリジナル解析用実行モジュールよりも平均 GROSS 時間が短い

黒色 : オリジナル解析用実行モジュールの平均 GROSS 時間と差がない

赤色 : オリジナル解析用実行モジュールよりも平均 GROSS 時間が長い

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- GROSS 時間表示領域が非表示状態

(i) 実行回数表示領域

対象関数の実行回数を表示します。

なお, オーバフローしている場合は, 実行回数のあとに“(オーバーフロー)”と表示されます。

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- 性能解析処理において, プローブ・コード挿入方式を用いた場合, かつ, チューニング後実行モジュール情報の場合

(j) 命令キャッシュ・ミス表示領域

対象関数における命令キャッシュ・ミスの回数, および, 発生率を表示します。

なお, 発生率は, 対象関数における命令キャッシュ・ヒットの回数, および, 命令キャッシュ・ミスの回数をもとに, 下記計算式から算出された値となります。

命令キャッシュ・ミスの回数 / (命令キャッシュ・ヒットの回数 + 命令キャッシュ・ミスの回数)

注意 本表示領域は, 以下に示した何れかの条件が成立した場合に非表示状態となります。

- 命令キャッシュのサイズが 0
- 性能解析処理において, プローブ・コード挿入方式を用いた場合
- 性能解析処理において, トレース・データ解析方式, かつ, トレース・データからキャッシュ・ミス情報が取得可能な実行エンジンを指定していない

(2) [親関数] タブ

性能解析処理の実行結果に関する情報 (親関数情報: 親関数名, 呼び出され回数など) をテーブル形式で一覧表示するタブです。

なお, 本タブは, 以下に示した表示領域から構成されています。

(a) 関数名表示領域

対象関数の親関数を表示します。

注意 同じ名前の関数が複数存在する場合, “関数名 (ファイル名)” のように表示されます。

また, 関数名, および, ファイル名が同じ名前である関数が複数存在する場合, “関数名 (絶対パス名)” のように表示されます。

(b) 呼び出され回数表示領域

親関数から呼び出された対象関数の呼び出され回数, および, 総呼び出され回数に対する割合 (単位: %) を表示します。

なお, オーバフローしている場合は, 実行回数のあとに “(オーバフロー)” と表示されます。

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- 性能解析処理において, プローブ・コード挿入方式を用いた場合, かつ, チューニング後実行モジュール情報の場合

注意 1 各項目のヘッダ部分をクリックすることにより, 関数情報を該当項目を基準としたソート表示することができます。

注意 2 各関数の情報部分をダブル・クリックした場合, 対象関数の詳細情報を表示するための[関数詳細情報ウィンドウ](#)がオープンします。

(3) [子関数] タブ

性能解析処理の実行結果に関する情報 (子関数情報: 子関数名, 呼び出し回数など) をテーブル形式で一覧表示するタブです。

なお, 本タブは, 以下に示した表示領域から構成されています。

(a) 子関数名表示領域

対象関数の子関数を表示します。

注意 同じ名前の関数が複数存在する場合, “関数名 (ファイル名)” のように表示されます。

また, 関数名, および, ファイル名が同じ名前である関数が複数存在する場合, “関数名 (絶対パス名)” のように表示されます。

(b) 呼び出し回数表示領域

対象関数から呼び出した子関数の呼び出し回数, および, 総呼び出し回数に対する割合 (単位: %) を表示します。

なお, オーバフローしている場合は, 実行回数のあとに “(オーバフロー)” と表示されます。

注意 本表示領域は, 以下に示した条件が成立した場合に非表示状態となります。

- 性能解析処理において, プローブ・コード挿入方式を用いた場合, かつ, チューニング後実行モジュール情報の場合

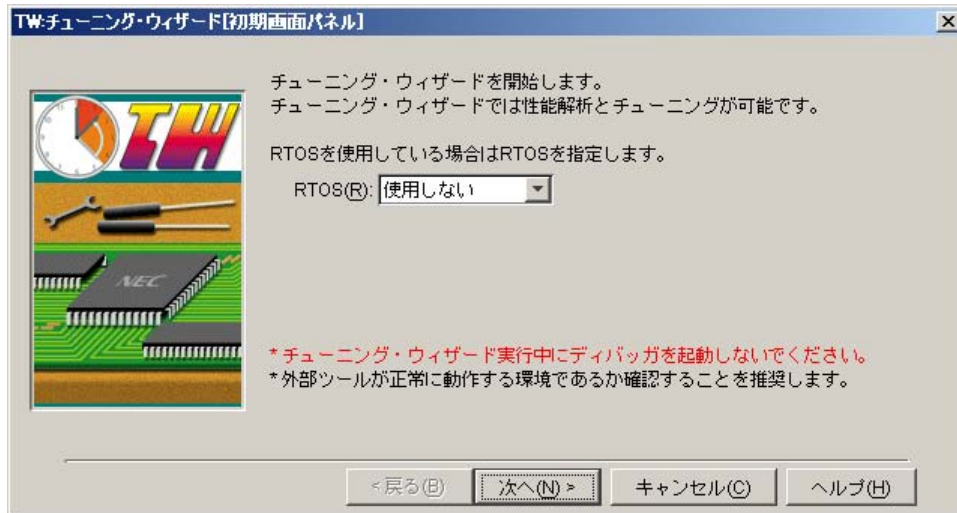
注意 1 各項目のヘッダ部分をクリックすることにより, 関数情報を該当項目を基準としたソート表示することができます。

注意 2 各関数の情報部分をダブル・クリックした場合, 対象関数の詳細情報を表示するための[関数詳細情報ウィンドウ](#)がオープンします。

初期画面パネル

TW850 のウィザード処理（性能解析処理，チューニング処理）が開始されたことの通知と，性能解析を行うプログラムに使用している RTOS の指定を行うためのパネルです。

図 6-7 初期画面パネル



ここでは，次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本パネルは，次の方法でオープンすることができます。

- [メイン・ウィンドウ](#)において，
 - チューニング・ウィザードボタンを押下したとき
 - [ツール (T)] メニュー → [チューニング ウィザード (W)...] を選択したとき

エリア説明

本パネルは，次の項目から構成されています。

- [メッセージ表示領域](#)
- [\[RTOS\(R\)\] コンボ・ボックス](#)

(1) メッセージ表示領域

TW850 のウィザード処理 (性能解析処理 , チューニング処理) が開始されたことを通知するためのメッセージを表示します。

(2) [RTOS(R)] コンボ・ボックス

使用する RTOS の種類 , または “ 使用しない ” を指定します。

デフォルトでは , “ 使用しない ” が指定されます。

なお , RTOS の種類を指定した場合 , TW850 は指定した RTOS を使用していることを前提として性能解析を行います。

注意 RTOS を使用している場合は , 必ず RTOS の種類を指定してください。

プロジェクト情報獲得パネル

TW850 のウィザード処理 (プロジェクト情報の獲得処理) が実行中であることを通知するためのパネルです。

図 6-8 プロジェクト情報獲得パネル



ここでは、次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは、次の項目から構成されています。

- [メッセージ表示領域](#)

(1) メッセージ表示領域

ウィザード処理 (プロジェクト情報の獲得処理) が実行中であることを通知するためのメッセージを表示します。

(a) <中断 (R) > ボタン

ウィザード処理 (プロジェクト情報の獲得処理) を中止します。

注意 本ボタンを押下してから実際にウィザード処理が中止されるまでの間には、時間を要する場合があります。

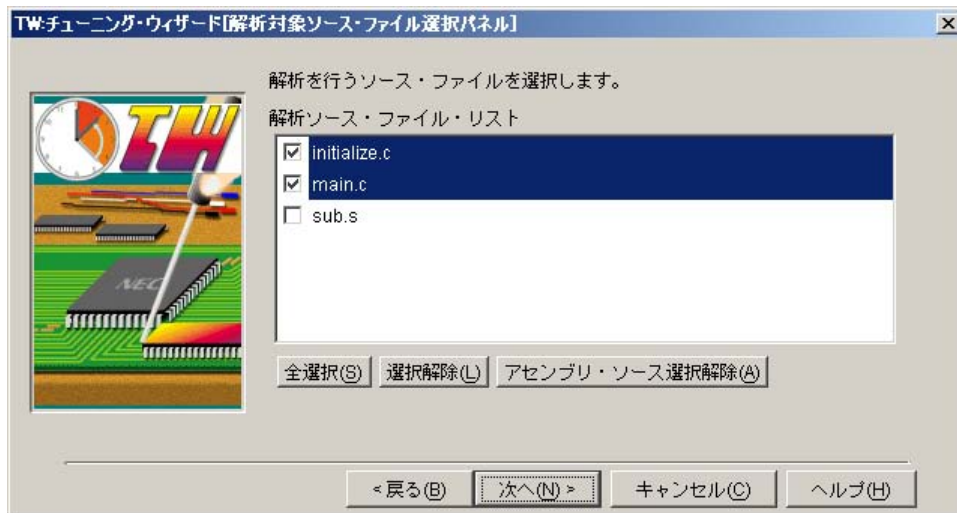
(b) <再試行 (R) > ボタン

ウィザード処理 (プロジェクト情報の獲得処理) を再実行します。

解析対象ソース・ファイル選択パネル

TW850 のウィザード処理 (性能解析処理, チューニング処理) において, 性能解析 / チューニング対象とするソース・ファイルを指定するためのパネルです。

図 6-9 解析対象ソース・ファイル選択パネル



ここでは, 次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは, 次の項目から構成されています。

- [メッセージ表示領域](#)
- [解析対象ソース・ファイル指定領域](#)

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) 解析対象ソース・ファイル指定領域

性能解析 / チューニング対象とするソース・ファイルを指定する領域です。

性能解析 / チューニング対象とするソース・ファイルを限定することにより、ウィザード処理 (性能解析処理, チューニング処理) にかかる時間や、プローブ・コード挿入方式で必要となるターゲット上のメモリ領域を削減することができます。ソース・ファイルと関数が多く存在する場合には、特に効果的となります。

注意 TW850 が認識できないアセンブリ言語・ソース・ファイルが存在する場合、TW850 は正常にウィザード処理 (性能解析処理, チューニング処理) を行うことができません。該当するアセンブリ言語・ソース・ファイルは非選択状態にして、性能解析 / チューニング対象外となるようにしてください。

アセンブリ言語・ソース・ファイルの記述についての詳細は、「[4.6.3 アセンブリ言語ソース・ファイルの記述形式](#)」を参照してください。

なお、本指定領域は、以下に示した表示領域、ボタン群から構成されています。

(a) 解析ソース・ファイル・リスト表示領域

性能解析 / チューニング対象とするソース・ファイルを指定します。

デフォルトでは、アセンブリ言語ソース・ファイル以外のファイルが選択状態となります。

(b) <全選択 (S)> ボタン

表示しているすべてのソース・ファイルを選択状態にします。

(c) <選択解除 (L)> ボタン

表示しているすべてのソース・ファイルを非選択状態にします。

(d) <アセンブリ・ソース選択解除 (A)> ボタン

表示しているソース・ファイル中で、アセンブリ言語・ソース・ファイルのみを非選択状態にします。

実行モジュール情報獲得パネル

TW850 のウィザード処理（オリジナル解析用実行モジュールの生成処理，関数解析，オリジナル実行モジュール解析処理）が実行中であることを通知するためのパネルです。

以下に，TW850 の実行内容を示します。

- 解析用アセンブリ言語ソース・ファイルの出力
- 関数情報の解析
- オリジナル解析用実行モジュールのビルド，解析

図 6-10 実行モジュール情報獲得パネル



ここでは，次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは，次の項目から構成されています。

- [メッセージ表示領域](#)

(1) メッセージ表示領域

ウィザード処理（性能解析処理：オリジナル解析用実行モジュールの生成処理，性能解析処理）が実行中であることを通知するためのメッセージを表示します。

(a) <中断 (R)> ボタン

ウィザード処理（性能解析処理：オリジナル解析用実行モジュールの生成処理，性能解析処理）を中止します。

注意 本ボタンを押下してから実際にウィザード処理が中止されるまでの間には，時間を要する場合があります。

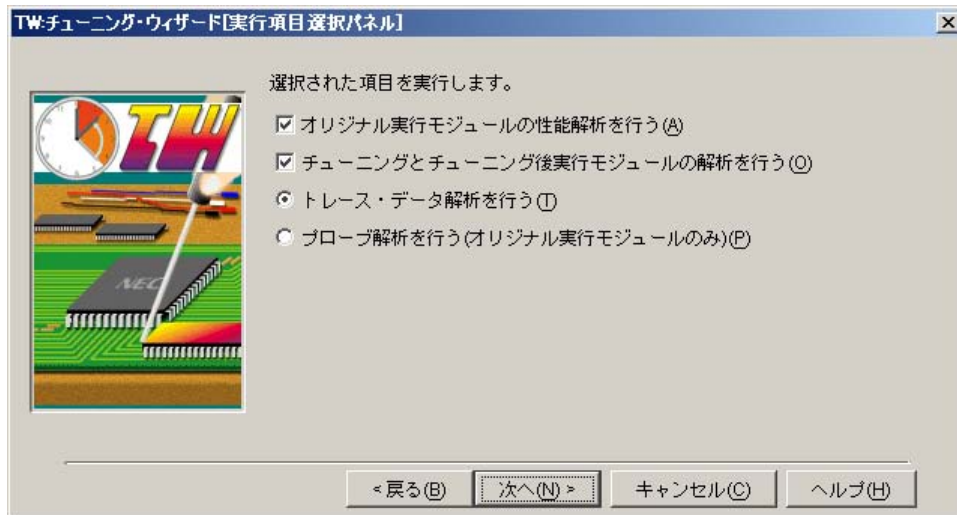
(b) <再試行 (R)> ボタン

ウィザード処理（性能解析処理：オリジナル解析用実行モジュールの生成処理，性能解析処理）を再実行します。

実行項目選択パネル

TW850 のウィザード処理 (性能解析処理 , チューニング処理) の実行項目を指定するためのパネルです。

図 6-11 実行項目選択パネル



ここでは , 次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは , 次の項目から構成されています。

- [メッセージ表示領域](#)
- [実行項目指定領域](#)
- [性能解析情報獲得方式指定領域](#)

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) 実行項目指定領域

ウィザード処理 (性能解析処理, チューニング処理) の実行項目に関する設定を行う領域です。

なお, 本指定領域は, 以下に示したチェック・ボックスから構成されています。

(a) [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックス

TW850 のウィザード処理として, オリジナル実行モジュールに対する性能解析処理を行うか否かを指定します。

デフォルトでは, チェック状態となります。

なお, 本チェック・ボックスがチェック状態の場合, “オリジナル実行モジュールに対する性能解析処理を行う” となります。

(b) [チューニングとチューニング後実行モジュールの解析を行う (Q)] チェック・ボックス

TW850 のウィザード処理として, オリジナル実行モジュールに対するチューニング処理 (チューニング後実行モジュールの生成処理), および, チューニング後実行モジュールに対する性能解析処理を行うか否かを指定します。

デフォルトでは, チェック状態となります。

なお, 本チェック・ボックスがチェック状態の場合, “オリジナル実行モジュールに対するチューニング処理 (チューニング後実行モジュールの生成処理), および, チューニング後実行モジュールに対する性能解析処理を行う” となります。

(3) 性能解析情報獲得方式指定領域

ウィザード処理 (性能解析処理, チューニング処理) における性能解析情報の獲得方式に関する設定を行う領域です。

なお, 本指定領域は, 以下に示したラジオ・ボタンから構成されています。

(a) [トレース・データ解析を行う (I)] ラジオ・ボタン

ウィザード処理における性能解析情報の獲得方式として, トレース・データ解析方式を用いるか否かを指定します。

トレース・データ解析方式についての詳細は, 「[1\) トレース・データ解析方式](#)」を参照してください。

デフォルトでは, チェック状態となります。

なお, 本ラジオ・ボタンがチェック状態の場合, “性能解析情報の獲得方式として, トレース・データ解析方式を用いる” となります。また, [チューニングとチューニング後実行モジュールの解析を行う (Q)] チェック・ボックスがチェック状態の場合, オリジナル実行モジュールだけでなく, チューニング後実行モジュールにもトレース・データ解析方式が用いられます。

注意 本ラジオ・ボタンは, 以下に示した条件が成立した場合に無効状態となります。

- [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

(b) [プローブ解析を行う (オリジナル実行モジュールのみ) (P)] ラジオ・ボタン

ウィザード処理における性能解析情報の獲得方式として, プローブ・コード挿入方式を用いるか否かを指定します。

プローブ・コード挿入方式についての詳細は, 「[2\) プローブ・コード挿入方式](#)」を参照してください。

[トレース・データ解析を行う (I)] ラジオ・ボタンが非チェック状態の場合, デフォルトでチェック状態となります。

なお, 本ラジオ・ボタンがチェック状態の場合, “性能解析情報の獲得方式として, プローブ・コード挿入方式を用いる” となります。

注意 1 本ラジオ・ボタンは, 以下に示した条件が成立した場合に無効状態となります。

- [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

注意 2 チューニング後実行モジュールに対してはプローブ・コードの挿入は行わないため, 本ラジオ・ボタンがチェック状態の場合, チューニング後の関数ごとの実行回数 / 実行時間などの情報の取得はできません。

トレース解析オプション選択パネル

TW850 のウィザード処理 (性能解析処理) において、トレース・データ解析方式を用いた性能解析情報の獲得オプションを指定するためのパネルです。

図 6-12 トレース解析オプション選択パネル



ここでは、次の項目について説明しています。

- エリア説明

エリア説明

本パネルは、次の項目から構成されています。

- メッセージ表示領域
- RTOS 領域指定領域
- RTOS タスク遷移監視用アドレス指定領域
- [関数遷移情報を取得する (T)] チェック・ボックス

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) RTOS 領域指定領域

(a) [RTOS の領域を指定する (G)] チェック・ボックス

RTOS システムが使用する領域を指定するか否かを指定します。

デフォルトでチェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“RTOS システムが使用する領域を指定する”となります。

注意 本チェック・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- 初期画面パネルの [RTOS(R)] コンボ・ボックスにおいて、V850 シリーズの RTOS、または、“使用しない”が選択状態

(b) [開始アドレスまたは開始シンボル (S)] コンボ・ボックス

RTOS システムが使用する領域の開始アドレス, または, 開始シンボルを指定します。

注意 本コンボ・ボックスは, 以下に示した条件が成立した場合に無効状態となります。

- 初期画面パネルの [RTOS(R)] コンボ・ボックスにおいて, V850 シリーズの RTOS, または, “ 使用しない ” が選択状態
- [RTOS の領域を指定する (G)] チェック・ボックスが非チェック状態

(c) [サイズ [bytes] または終了シンボル (Z)] コンボ・ボックス

RTOS 領域のサイズ, または, 終了シンボルを指定します。

注意 本コンボ・ボックスは, 以下に示した条件が成立した場合に無効状態となります。

- 初期画面パネルの [RTOS(R)] コンボ・ボックスにおいて, V850 シリーズの RTOS, または, “ 使用しない ” が選択状態
- [RTOS の領域を指定する (G)] チェック・ボックスが非チェック状態

(3) RTOS タスク遷移監視用アドレス指定領域

(a) [RTOS のタスク遷移監視用アドレス指定する (W)] チェック・ボックス

RTOS のタスク遷移を監視するためにアドレスを指定するか否かを指定します。

デフォルトでチェック状態となります。

なお, 本チェック・ボックスがチェック状態の場合, “ RTOS のタスク遷移を監視するためにアドレス指定する ” となります。

注意 本チェック・ボックスは, 以下に示した条件が成立した場合に無効状態となります。

- 初期画面パネルの [RTOS(R)] コンボ・ボックスにおいて, V850 シリーズの RTOS, または, “ 使用しない ” が選択状態
- BUS トレース・データが取得できず, N-Wire トレース・データのみ取得可能な場合

(b) [アドレス (D)] コンボ・ボックス

RTOS 領域の開始アドレス, または, 開始シンボルを指定します。

注意 1 本コンボ・ボックスは, 以下に示した条件が成立した場合に無効状態となります。

- 初期画面パネルの [RTOS(R)] コンボ・ボックスにおいて, V850 シリーズの RTOS, または, “ 使用しない ” が選択状態
- [RTOS のタスク遷移監視用アドレス指定する (W)] チェック・ボックスが非チェック状態
- キャッシュابل・メモリ領域, および, ユーザ・プログラムで使用する領域を指定した場合

注意 2 ユーザはタスク遷移を監視し, 指定アドレスに遷移元タスク ID 情報と遷移先タスク ID 情報を書き込む処理コードをプログラム中に記述する必要があります。指定アドレスから 4 バイト領域に遷移元タスク ID, 指定アドレス +4 バイトからの 4 バイト領域に遷移先タスク ID を書き込む処理コードを記述してください。

(4) [関数遷移情報を取得する (I)] チェック・ボックス

トレース・データの性能解析時に, 関数遷移情報を取得するか否かを指定します。

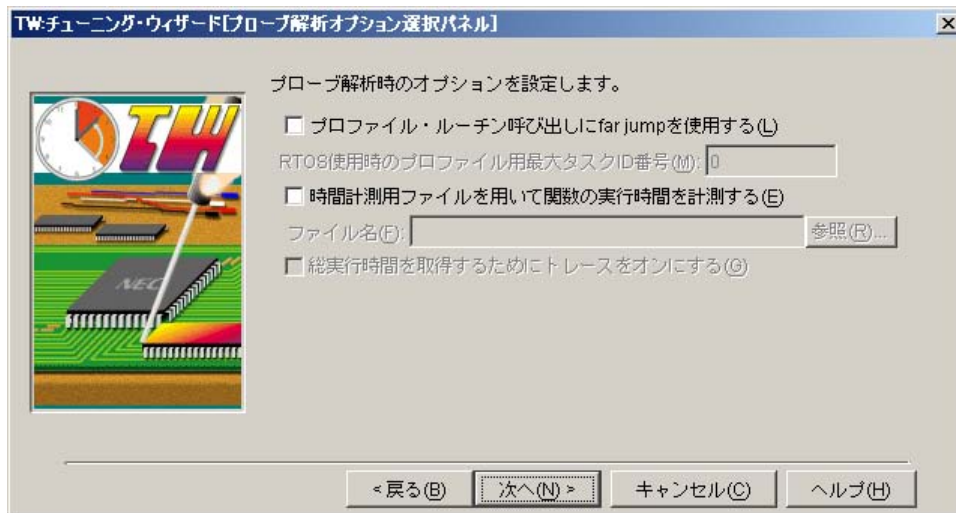
デフォルトでは, 非チェック状態となります。

なお, 本チェック・ボックスがチェック状態の場合, “ 関数遷移情報を取得する ” となります。

プローブ解析オプション選択パネル

TW850 のウィザード処理 (性能解析処理) において、プローブ・コード挿入方式を用いた性能解析情報の獲得オプションを指定するためのパネルです。

図 6-13 プローブ解析オプション選択パネル



ここでは、次の項目について説明しています。

- エリア説明

エリア説明

本パネルは、次の項目から構成されています。

- メッセージ表示領域
- [プロファイル・ルーチン呼び出しに far jump を使用する (L)] チェック・ボックス
- [RTOS 使用時のプロファイル用最大タスク ID 番号 (M)] 指定領域
- 性能解析対象関数の実行時間計測処理指定領域
- [総実行時間を取得するためにトレースをオンにする (G)] チェック・ボックス

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) [プロファイル・ルーチン呼び出しに far jump を使用する (L)] チェック・ボックス

TW850 のウィザード処理 (チューニング処理) で中間ファイルとして出力されるプロファイル・ルーチンの呼び出し方法として far jump 命令を用いるか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“プロファイル・ルーチンの呼び出し方法として far jump 命令を用いる”となります。

プローブ・コード挿入方式で挿入するプローブ・コードがプロファイル・ルーチンを呼び出しますが、プローブ・コードとプロファイル・ルーチンの配置距離が遠くなると、コンパイル・エラーが発生する場合があります。そのような場合、プロファイル・ルーチンの呼び出しに全アドレス空間を指定できる呼び出し (far jump) を使用すると、配置距離が遠いプロファイル・ルーチンも呼び出すことができます。

注意 1 本チェック・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルにおいて、“[プローブ解析を行う (オリジナル実行モジュールのみ) (E)] ラジオ・ボタン” が非チェック状態

注意 2 プロファイル・ルーチンの呼び出し方法として far jump 命令を用いると、挿入するプローブ・コードのサイズが増加します。

注意 3 プロファイル・ルーチンの配置先が適切ではない可能性がある場合は、配置先選択パネルのプロファイル・ルーチン指定領域において、プロファイル・ルーチンをなるべくプログラムの配置先の近くに配置してください。

(3) [RTOS 使用時のプロファイル用最大タスク ID 番号 (M)] 指定領域

TW850 がウィザード処理 (性能解析処理: RTOS 情報獲得処理) を実行する際に必要となるプロファイル用最大タスク ID 番号を指定します。

プロファイル用最大タスク ID 番号とは、RTOS 情報獲得処理を実行する必要があるタスク ID 番号の最大値を意味しています。ここで指定したタスク ID 番号までのタスクの遷移を監視します。

なお、指定可能な値の範囲は、使用している RTOS が RX850 の場合は 1 ~ 127, RX850 Pro の場合は 1 ~ 32767 となります。

デフォルトでは、RX850, または、RX850 Pro を使用し、かつ、実行モジュールから最大タスク ID が取得可能な場合はその最大タスク ID, 最大タスク ID が取得不可能な場合は 127 が指定されます。

注意 1 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルにおいて、“[プローブ解析を行う (オリジナル実行モジュールのみ) (E)] ラジオ・ボタン” が非チェック状態
- RTOS を使用していない、または、初期画面パネルの [RTOS(R)] コンボ・ボックスにおいて、“使用しない”を指定

注意 2 指定したタスク ID 番号より大きいタスクの遷移がある場合、そのタスクの遷移が監視されないため、タスクの性能解析を正常に行うことができません。ただし、必要以上に大きな数値を指定すると、ターゲット上に必要となるメモリ領域が必要以上に増加することになります。

(4) 性能解析対象関数の実行時間計測処理指定領域

性能解析対象関数の実行時間計測処理に関する設定を行う領域です。

(a) [時間計測用ファイルを用いて関数の実行時間を計測する (E)] チェック・ボックス

TW850 のウィザード処理 (性能解析処理: 性能解析対象関数の実行時間計測処理) として、性能解析対象関数ごとの実行時間計測処理を行うか否かを指定します。

性能解析対象関数の実行時間計測処理を行うには、時間計測用ファイル (タイマ制御コード部) を用意する必要があります。時間計測用ファイル (タイマ制御コード部) については、「4.4 タイマ制御コード部」を参照してください。

デフォルトでは、非チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“性能解析対象関数の実行時間計測処理を行う”となります。

(b) [ファイル名 (E)] 指定領域

TW850 がウィザード処理 (性能解析処理: 性能解析対象関数の実行時間計測処理) を実行する際に必要となる時間計測用ファイル (タイマ制御コード部) のファイル名を指定します。

デフォルトでは、何も指定されません。

なお、時間計測用ファイル (タイマ制御コード部) については「[4.4 タイマ制御コード部](#)」を参照してください。

注意 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- [時間計測用ファイルを用いて関数の実行時間を計測する (E)] チェック・ボックスが非チェック状態

(c) <参照 (R)... > ボタン

[ファイル選択ダイアログ](#)をオープンします。

注意 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- [時間計測用ファイルを用いて関数の実行時間を計測する (E)] チェック・ボックスが非チェック状態

(5) [総実行時間を取得するためにトレースをオンにする (G)] チェック・ボックス

トレース・データのみから総実行時間を取得できる場合に、トレースをオンにして総実行時間を取得するか否かを指定します。

なお、本チェック・ボックスがチェック状態の場合、“総実行時間を取得するためにトレースをオンにして総実行時間を取得する”となります。

注意 本チェック・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- NEC エレクトロニクス・ツール・チェーンを使用している

実行解析範囲選択パネル

ユーザ・プログラム、実行モジュールの実行開始位置、および、ウィザード処理（性能解析処理）における性能解析範囲を指定するためのパネルです。

図 6-14 実行解析範囲選択パネル



ここでは、次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [\[実行開始位置 \(S\) \] コンボ・ボックス](#)
- [性能解析対象関数指定領域](#)
- [性能解析対象範囲指定領域](#)
- [\[トレースのオン / オフを制御するイベントを指定する \(O\) \] ラジオ・ボタン](#)

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) [実行開始位置 (S)] コンボ・ボックス

ユーザ・プログラム、実行モジュールの実行開始位置を指定します。

選択項目は、エントリ・ポイントのアドレス、および、オリジナル実行モジュール情報を解析して得られた全シンボルとなります。

また、テキスト入力も可能です。入力可能な値の範囲は、0x0 ~ プロジェクトで指定されているチップの最大アドレスとなります。

デフォルトでは、エントリ・ポイントのアドレスが指定されます。ただし、エントリ・ポイントのアドレスと同じ値のシンボルが存在する場合、そのシンボルが指定されます。

性能解析範囲に依存せずに、ここで指定した位置からユーザ・プログラム、実行モジュールの実行を開始するため、通常はリセット番地を指定します。

注意 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

(3) 性能解析対象関数指定領域

ウィザード処理 (性能解析処理) における性能解析対象関数を指定します。

性能解析情報の獲得方式により、以下に示した特徴があります。

【トレース・データ解析方式の場合】

指定した実行開始位置と終了位置、および、性能解析範囲内で、トレース・バッファがいっぱいになるごとにプログラムを停止します。

【プローブ・コード挿入方式の場合】

指定した実行開始位置と終了位置で、プログラムを停止します。

ただし、ターゲット上で、ID850、ID850NW を使用して実行時間計測を行う場合、指定した実行開始位置と終了位置を含む関数の実行時間計測を行うことはできません。

なお、本指定領域は、以下に示したラジオ・ボタン、コンボ・ボックスから構成されています。

(a) [解析する関数を指定する (E)] ラジオ・ボタン

ウィザード処理 (性能解析処理) における性能解析対象範囲を関数名で指定するか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“性能解析対象範囲を関数名で指定する” となります。

注意 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

(b) 性能解析対象関数名指定コンボ・ボックス

ウィザード処理 (性能解析処理) における性能解析対象範囲 (関数名) を選択します。

選択項目は、オリジナル実行モジュール情報を解析して得られた関数名となります。

デフォルトでは、何も選択されません。

注意 1 同じ名前の関数が複数存在する場合、“関数名 (ファイル名)” のように表示されます。

また、関数名、および、ファイル名が同じ名前である関数が複数存在する場合、“関数名 (絶対パス名)” のように表示されます。

注意 2 タスク名を選択した場合は、指定したタスクの開始から任意のタスクの終了までの性能解析を行います。

注意 3 TW850 では、関数の戻りアドレスがレジスタ LP に設定されるものとして処理を行います。したがって、本コンボ・ボックスにおいて、戻りアドレスがレジスタ LP に設定されない関数を指定した場合、その動作を保証していません。

注意 4 処理が終了する関数を指定してください。処理が終了しない関数を指定した場合、性能解析処理が終了しません。

注意 5 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態
- [解析する関数を指定する (E)] ラジオ・ボタンが非チェック状態

(4) 性能解析対象範囲指定領域

ウィザード処理 (性能解析処理) における性能解析対象範囲を指定します。
性能解析情報の獲得方式により、以下に示した特徴があります。

【トレース・データ解析方式の場合】

指定した実行開始位置と終了位置、および、性能解析範囲内で、トレース・バッファがいっぱいになるごとにプログラムを停止します。

【プローブ・コード挿入方式の場合】

指定した実行開始位置と終了位置で、プログラムを停止します。

ただし、ターゲット上で、ID850, ID850NW を使用して実行時間計測を行う場合、指定した実行開始位置と終了位置を含む関数の実行時間計測を行うことはできません。

なお、本指定領域は、以下に示したラジオ・ボタン、コンボ・ボックス、指定領域から構成されています。

(a) [解析する範囲を指定する (R)] ラジオ・ボタン

ウィザード処理 (性能解析処理) における性能解析対象範囲をファイル名、シンボル名、行番号で指定するか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“性能解析対象範囲をファイル名、シンボル名、行番号で指定する”となります。

注意 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

(b) [開始位置 (I)] コンボ・ボックス

ウィザード処理 (性能解析処理) における性能解析開始位置 (ファイル名、シンボル名) を選択します。

選択項目は、C 言語ソース・ファイル名、および、オリジナル実行モジュール情報を解析して得られたテキスト属性領域の全シンボル名となります。

デフォルトでは、何も選択されません。

注意 1 本コンボ・ボックスは、以下に示した何れかの条件が成立した場合に無効状態となります。

- [解析する範囲を指定する (R)] ラジオ・ボタンが非チェック状態
- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

注意 2 同じ名前の関数が複数存在する場合、“関数名 (ファイル名)” のように表示されます。

また、関数名、および、ファイル名が同じ名前である関数が複数存在する場合、“関数名 (絶対パス名)” のように表示されます。

注意 3 本コンボ・ボックスの選択項目に“ファイル名”を表示するには、ディバグ・オプション (NEC エレクトロニクス製コンパイラでは -g) を用いてオリジナル実行モジュールが生成されている必要があります。

注意 4 本コンボ・ボックスの選択項目には、必ず実行されるファイル名、または、シンボル名を指定してください。

(c) [行番号 (L)] 指定領域

ウィザード処理 (性能解析処理) における性能解析開始位置 (行番号) を指定します。

入力可能な値の範囲は、1 ~ 4,294,967,295 となります。

デフォルトでは、何も指定されません。

注意 1 本指定領域は、以下に示した何れかの条件が成立した場合に無効状態となります。

- [開始位置 (I)] コンボ・ボックスが未入力状態
- [開始位置 (I)] コンボ・ボックスにソース・ファイル以外が指定されている状態
- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

注意 2 ディバグ上で、ブレーク・ポイントを設定でき、実行される行番号を指定してください。

注意 3 指定した行番号が、コンパイラの最適化によりユーザの意図しない位置を示す可能性がある場合、行番号は赤色表示されます。

(d) [終了位置 (E)] コンボ・ボックス

ウィザード処理 (性能解析処理) における性能解析終了位置 (ファイル名, シンボル名) を選択します。
 選択項目は, C 言語ソース・ファイル名, および, オリジナル実行モジュール情報を解析して得られたテキスト属性領域の全シンボル名となります。
 デフォルトでは, 何も選択されません。

注意 1 本コンボ・ボックスは, 以下に示した何れかの条件が成立した場合に無効状態となります。

- [解析する範囲を指定する (R)] ラジオ・ボタンが非チェック状態
- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

注意 2 同じ名前の関数が複数存在する場合, “関数名 (ファイル名)” のように表示されます。
 また, 関数名, および, ファイル名が同じ名前である関数が複数存在する場合, “関数名 (絶対パス名)” のように表示されます。

注意 3 本コンボ・ボックスの選択項目に “ファイル名” を表示するには, デバッグ・オプション (NEC エレクトロニクス製コンパイラでは -g) を用いてオリジナル実行モジュールが生成されている必要があります。

注意 4 本コンボ・ボックスの選択項目には, 必ず実行されるファイル名, または, シンボル名を指定してください。

(e) [行番号 (I)] 指定領域

ウィザード処理 (性能解析処理) における性能解析終了位置 (行番号) を指定します。
 入力可能な値の範囲は, 1 ~ 4,294,967,295 となります。
 デフォルトでは, 何も指定されません。

注意 1 本指定領域は, 以下に示した何れかの条件が成立した場合に無効状態となります。

- [終了位置 (E)] コンボ・ボックスが未入力状態
- [終了位置 (E)] コンボ・ボックスにソース・ファイル以外が指定されている状態
- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

注意 2 デバッグ上で, ブレーク・ポイントを設定でき, 実行される行番号を指定してください。

注意 3 指定した行番号が, コンパイラの最適化によりユーザの意図しない位置を示す可能性がある場合, 行番号は赤色表示されます。

(5) [トレースのオン/オフを制御するイベントを指定する (Q)] ラジオ・ボタン

トレース・データ解析方式において, プログラム中のデータ書き込みをトレースのオン/オフ (性能解析開始/終了) 制御のためのイベント (トレース・イベント) として使用するかどうかを指定します。

デフォルトでは, 非チェック状態となります。

なお, 本ラジオ・ボタンがチェック状態の場合, “トレース・データ解析方式において, プログラム中のデータ書き込みをトレース・イベントとして使用する” となります。

また, [解析する関数を指定する (E)] ラジオ・ボタン, または, [解析する範囲を指定する (R)] ラジオ・ボタンがチェック状態の場合と比較すると, 以下に示した特徴があります。

- トレース・イベントが発生した位置でプログラムを停止するが (性能解析終了位置を指定した場合は, 性能解析終了位置でも停止), トレース・バッファがいっぱいになってもプログラムは停止しない
- トレース・バッファがいっぱいになった場合, トレース・バッファ容量のトレース・データのみを取得
- プログラム中に, トレース・イベントを制御するデータ書き込みが必要
- トレース・イベントが発生することによる性能解析開始から終了までを複数回解析して, 累計することが可能
- ユーザ・プログラム内で, 性能解析開始/終了を制御できるため, ユーザの意図したトレース・データを取得することが可能

注意 本ラジオ・ボタンは, 以下に示した条件が成立した場合に無効状態となります。

- デバッグに ID850NW を使用していない, または, デバッグにアクセス・ブレーク・イベントが設定不可能
- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

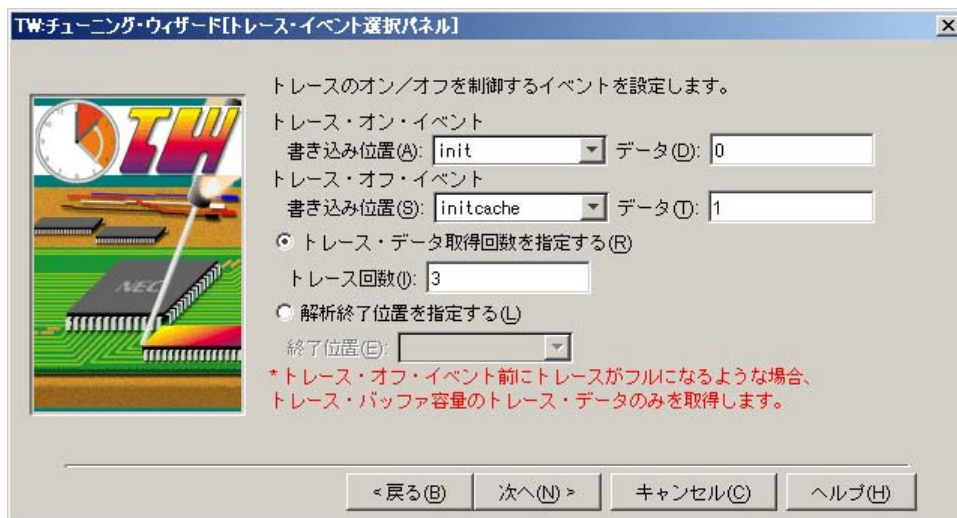
トレース・イベント選択パネル

TW850 のウィザード処理 (性能解析処理) における性能解析範囲のトレース・イベントを選択するためのパネルです。

注意 本パネルは、以下に示した条件が成立した場合はオープンされません。

- デバッグに ID850NW を使用していない

図 6-15 トレース・イベント選択パネル



ここでは、次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [トレース・オン・イベント指定領域](#)
- [トレース・オフ・イベント指定領域](#)
- [トレース・データ取得回数指定領域](#)
- [解析終了位置指定領域](#)

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) トレース・オン・イベント指定領域

トレースをオン (性能解析開始) にするイベントを指定する領域です。

なお、本指定領域は、以下に示したコンボ・ボックス、指定領域から構成されています。

(a) [書き込み位置 (A)] コンボ・ボックス

トレースをオンにするイベントとなる書き込み先シンボル名、または、アドレスを指定します。

選択項目は、オリジナル実行モジュール情報を解析して得られた全グローバル・シンボル名となります。

また、テキスト入力も可能です。入力可能な値の範囲は、0x0 ~ プロジェクトで指定されているチップの最大アドレスとなります。

デフォルトでは、何も指定されません。

注意 1 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

注意 2 チューニング処理によって、書き込みアドレスが変更となる可能性がある場合は、シンボル名を指定するようにしてください。

注意 3 本コンボ・ボックスは、トレース・オフ・イベント指定領域の [書き込み位置 (S)] コンボ・ボックスと同一位置を指定することが可能です。

(b) [データ (D)] 指定領域

トレースをオンにするイベントとなる 4 バイトの書き込みデータを指定します。

入力可能な値の範囲は、0 ~ 4294967295(0xffffffff) の 4 バイト非負整数となります。

注意 1 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

注意 2 本コンボ・ボックスは、トレース・オフ・イベント指定領域の [データ (I)] 指定領域と同一位置を指定することが可能です。

注意 3 TW850 は、実行モジュールの実行時に、トレースがオフ状態でトレース・オン・イベントを検出 (指定した書き込み位置に、指定した書き込みデータが書き込まれるのをアクセス・ブレイク・イベントとして検出) した場合、イベント発生位置で実行 (CPU) を停止し、トレースをオンにした後に実行を再開します。このトレース・オン・イベントは、ユーザ・プログラム中に記述されている必要があります。

なお、トレースがオン状態でトレース・オン・イベントを検出した場合、何も行いません。

注意 4 N-Wire トレース・データを取得可能なターゲットを使用している場合、以下に示したアドレス範囲のみ指定できます。

64MB モード : 0x0 ~ 0x3ffffff

256MB モード : 0x0 ~ 0xffffffff

注意 5 ターゲットにより、実際にデータを書き込むタイミングに対してアクセス・ブレイク・イベントの検出が遅れる場合があります。そのようなターゲットにおいて、トレース・オン・イベントを設定する場合は、その遅れを誤差として認識するか、または、対策を講じる必要があります。アクセス・ブレイク・イベントの発生タイミングが遅れるかどうかの判断、および、対策は、各ターゲットのユーザーズ・マニュアルを参照してください。

(3) トレース・オフ・イベント指定領域

トレースをオフ (性能解析終了) にするイベントを指定する領域です。

なお、本指定領域は、以下に示したコンボ・ボックス、指定領域から構成されています。

(a) [書き込み位置 (S)] コンボ・ボックス

トレースをオフにするイベントとなる書き込み先シンボル名、または、アドレスを指定します。

選択項目は、オリジナル実行モジュール情報を解析して得られた全グローバル・シンボル名となります。

また、テキスト入力も可能です。入力可能な値の範囲は、0x0 ~ プロジェクトで指定されているチップの最大アドレスとなります。

デフォルトでは、何も指定されません。

注意 1 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

注意 2 チューニング処理によって、書き込みアドレスが変更となる可能性がある場合は、シンボル名を指定するようにしてください。

注意 3 本コンボ・ボックスは、トレース・オン・イベント指定領域の [書き込み位置 (A)] コンボ・ボックスと同一位置を指定することが可能です。

(b) [データ (I)] 指定領域

トレースをオフにするイベントとなる 4 バイトの書き込みデータを指定します。

入力可能な値の範囲は、0 ~ 4294967295(0xffffffff) の 4 バイト非負整数となります。

注意 1 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

注意 2 本コンボ・ボックスは、トレース・オフ・イベント指定領域の [データ (D)] 指定領域と同一位置を指定することが可能です。

注意 3 TW850 は、実行モジュールの実行時に、トレースがオン状態でトレース・オフ・イベントを検出 (指定した書き込み位置に、指定した書き込みデータが書き込まれるのをアクセス・ブレイク・イベントとして検出) した場合、イベント発生位置で実行 (CPU) を停止し、トレースをオフにした後に実行を再開します。このトレース・オフ・イベントは、ユーザ・プログラム中に記述されている必要があります。

なお、トレースがオン状態でトレース・オン・イベントを検出した場合、何も行いません。

注意 4 N-Wire トレース・データを取得可能なターゲットを使用している場合、以下に示したアドレス範囲のみ指定できます。

64MB モード : 0x0 ~ 0x3ffffff

256MB モード : 0x0 ~ 0xffffffff

注意 5 ターゲットにより、実際にデータを書き込むタイミングに対してアクセス・ブレイク・イベントの検出が遅れる場合があります。そのようなターゲットにおいて、トレース・オン・イベントを設定する場合は、その遅れを誤差として認識するか、または、対策を講じる必要があります。アクセス・ブレイク・イベントの発生タイミングが遅れるかどうかの判断、および、対策は、各ターゲットのユーザーズ・マニュアルを参照してください。

(4) トレース・データ取得回数指定領域

ウィザード処理 (性能解析処理) におけるトレース・データを取得する回数に関する設定を行う領域です。

指定したトレース・データ取得回数で性能解析 (プログラムの実行) を終了させます。

なお、本指定領域は、以下に示したラジオ・ボタン、指定領域から構成されています。

(a) [トレース・データ取得回数を指定する (R)] ラジオ・ボタン

トレース・データを取得する回数を指定して解析を行うか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、「トレース・データ取得回数を指定して解析を行う」となります。

注意 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

(b) [トレース回数 (I)] 指定領域

トレース・データを取得する回数を指定します。

入力可能な値の範囲は、0 ~ 2,147,483,647 となります。

注意 1 トレース・データを取得できる回数 (トレース・オン・イベント発生後にトレース・オフ・イベントが発生する回数) より多い回数を指定すると、性能解析処理が終了しなくなりますので、トレース・オフ・イベントが発生する回数を超えないようにしてください。

注意 2 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態
- [トレース・データ取得回数を指定する (R)] ラジオ・ボタンが非チェック状態

(5) 解析終了位置指定領域

ウィザード処理 (性能解析処理) における解析終了位置に関する設定を行う領域です。

なお、本指定領域は、以下に示したラジオ・ボタン、コンボ・ボックスから構成されています。

(a) [解析終了位置を指定する (L)] ラジオ・ボタン

終了位置を指定して解析を行うか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“解析終了位置を指定する” となります。

注意 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態

(b) [終了位置 (E)] コンボ・ボックス

解析終了位置のシンボル名、または、アドレスを指定します。

選択項目は、オリジナル実行モジュール情報を解析して得られた全シンボルとなります。

また、テキスト入力も可能です。入力可能な値の範囲は、0x0 ~ プロジェクトで指定されているチップの最大アドレスとなります。

デフォルトでは、何も指定されません。

注意 1 チューニング処理によって、書き込みアドレスが変更となる可能性がある場合は、シンボル名を指定するようにしてください。

注意 2 本コンボ・ボックスの選択項目には、必ず実行されるファイル名、または、シンボル名を指定してください。

注意 3 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態
- [解析終了位置を指定する (L)] ラジオ・ボタンが非チェック状態

命令キャッシュ選択パネル

TW850 のウィザード処理 (性能解析処理 , チューニング処理) において , 使用する命令キャッシュの種別 , および , パラメータを指定するためのパネルです。

図 6-16 命令キャッシュ選択パネル



ここでは , 次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは , 次の項目から構成されています。

- [メッセージ表示領域](#)
- [\[命令キャッシュ \(I\) \] コンボ・ボックス](#)
- [命令キャッシュ・パラメータ指定領域](#)
- [\[オリジナル実行モジュールの性能解析時に命令キャッシュを無効にする \(D\) \] チェック・ボックス](#)

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) [命令キャッシュ (I)] コンボ・ボックス

ウィザード処理 (性能解析処理, チューニング処理) において使用する命令キャッシュの種別を選択します。
 選択項目は, “ デバイスの初期値 ”, “ なし ” (命令キャッシュ・サイズをキャッシュ・マクロの組み合わせから指定可能な実行エンジンの場合), “ NB85E212(4K バイト, 2 ウエイ・セット・アソシアティブ) ” (V850 シリーズの場合), “ NB85E213(8K バイト, 2 ウエイ・セット・アソシアティブ) ” (V850 シリーズの場合), “ ユーザ指定 ” (命令キャッシュ・サイズをキャッシュ・マクロの組み合わせ以外から指定可能な実行エンジンの場合) となります。
 デフォルトでは, “ デバイスの初期値 ” が指定されます。

注意 命令キャッシュが使用可能な場合, TW850 は, 本コンボ・ボックスで選択した命令キャッシュを実行エンジンに設定しますが, 命令キャッシュを使用するための CPU 初期化処理 (ユーザーズ・マニュアル「命令キャッシュ, データ・キャッシュ NB85E, NB85ET 編」に記述されているような命令キャッシュの設定) は行ないません。そのため, 命令キャッシュを使用するためには, CPU によりスタート・アップ・ルーチンに命令キャッシュの設定処理を追加する必要があります。

なお, NEC エレクトロニクス製シミュレータでは, スタート・アップ・ルーチンに命令キャッシュの設定は必要ありません。

(3) 命令キャッシュ・パラメータ指定領域

ウィザード処理 (性能解析処理, チューニング処理) において使用する命令キャッシュのパラメータを指定します。
 なお, 本指定領域は, 以下に示した指定領域から構成されています。

(a) [サイズ (S)] 指定領域

ウィザード処理 (性能解析処理, チューニング処理) において使用する命令キャッシュのサイズ (単位: バイト) を指定します。

なお, 本指定領域に指定可能な値は,

“ ライン・サイズ指定領域で指定する値 ” * “ ウエイ数指定領域で指定する値 ”

の算出結果の整数倍値に限られます。

デフォルトでは, プロジェクトで指定されているチップの命令キャッシュ・サイズが指定されます。

注意 1 本指定領域は, 以下に示した条件が成立した場合に無効状態となります。

- [命令キャッシュ (I)] コンボ・ボックスにおいて, “ ユーザ指定 ” 以外を選択

注意 2 命令キャッシュのサイズを 0 にすると, ICC レジスタの値が 0x0 にならなくなり, タグの初期化の完了を検出できない場合があります。このような場合, ICC レジスタの最下位 2 ビットが 0 になるのを待つことでタグの初期化の完了を検出することが可能です。

なお, NEC エレクトロニクス製シミュレータでは, この記述は関係ありません。

(b) [ライン・サイズ (L)] 指定領域

ウィザード処理 (性能解析処理, チューニング処理) において使用する命令キャッシュのライン・サイズ (単位: バイト) を指定します。

入力可能な値の範囲は, 実行エンジンに依存します。

デフォルトでは, プロジェクトで指定されているチップの命令キャッシュ・ライン・サイズが指定されます。

注意 本指定領域は, 以下に示した条件が成立した場合に無効状態となります。

- [命令キャッシュ (I)] コンボ・ボックスにおいて, “ ユーザ指定 ” 以外を選択

(c) [ウエイ数 (W)] 指定領域

ウィザード処理 (性能解析処理, チューニング処理) において使用する命令キャッシュのウエイ数を指定します。

入力可能な値の範囲は, 実行エンジンに依存します。

デフォルトでは, プロジェクトで指定されているチップのウエイ数が指定されます。

注意 1 本指定領域は, 以下に示した条件が成立した場合に無効状態となります。

- [命令キャッシュ (I)] コンボ・ボックスにおいて, “ ユーザ指定 ” 以外を選択

注意 2 命令キャッシュのウエイ数を 1 以下にすると、ICC レジスタの値が 0x0 にならなくなり、タグの初期化の完了を検出できない場合があります。このような場合、ICC レジスタの最下位 2 ビットが 0 になるのを待つことでタグの初期化の完了を検出することが可能です。
なお、NEC エレクトロニクス製シミュレータでは、この記述は関係ありません。

(4) [オリジナル実行モジュールの性能解析時に命令キャッシュを無効にする (D)] チェック・ボックス

オリジナル実行モジュールを性能解析する際、命令キャッシュを無効にするか否かを指定します。
デフォルトでは、非チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“オリジナル実行モジュールを性能解析する際、命令キャッシュを無効にする”となります。

注意 本チェック・ボックスは、以下に示した何れかの条件が成立した場合に無効状態となります。

- 実行項目選択パネルにおいて、[オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態
- 命令キャッシュのサイズが 0
- キャッシュを無効にできないデバイスを指定

高速アクセス・メモリ領域選択パネル

TW850 のウィザード処理（チューニング処理）において、使用する高速アクセス・メモリ領域の種別、および、パラメータを指定するためのパネルです。

図 6-17 高速アクセス・メモリ領域選択パネル



ここでは、次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [\[高速アクセス・メモリ領域を使用する \(F\) \] チェック・ボックス](#)
- [\[高速アクセス・メモリ領域 \(M\) \] コンボ・ボックス](#)
- [高速アクセス・メモリ領域パラメータ指定領域](#)

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) [高速アクセス・メモリ領域を使用する (E)] チェック・ボックス

ウィザード処理 (チューニング処理) において、実行モジュールの実行性能を落とす原因となっている関数を優先的に高速アクセス・メモリ領域 (内蔵 ROM など) へと配置するか否かを指定します。

プロジェクトで指定されているチップに内蔵 ROM が存在する場合、デフォルトでチェック状態となります。なお、本チェック・ボックスがチェック状態の場合、“実行性能を落とす原因となっている関数を優先的に高速アクセス・メモリ領域 (内蔵 ROM など) へと配置する” となります。

(3) [高速アクセス・メモリ領域 (M)] コンボ・ボックス

ウィザード処理 (チューニング処理) において使用する高速アクセス・メモリ領域の種別を選択します。

選択項目は、“内蔵 ROM” (V850 シリーズの場合)、“ユーザ指定” となります。

デフォルトでは、内蔵 ROM が存在する場合は“内蔵 ROM”、内蔵 ROM が存在しない場合は“ユーザ指定”が指定されます。

注意 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- [高速アクセス・メモリ領域を使用する (E)] チェック・ボックスが非チェック状態

(4) 高速アクセス・メモリ領域パラメータ指定領域

ウィザード処理 (チューニング処理) において使用する高速アクセス・メモリ領域のパラメータを指定します。

なお、本指定領域は、以下に示した指定領域から構成されています。

(a) [開始アドレス (A)] 指定領域

ウィザード処理 (チューニング処理) において使用する高速アクセス・メモリ領域の開始アドレスを指定します。

入力可能な値の範囲は、0x0 ~ プロジェクトで指定されているチップの最大アドレスとなります。

デフォルトでは、内蔵 ROM が存在する場合はプロジェクトで指定されているチップの内蔵 ROM の開始アドレス、内蔵 ROM が存在しない場合は 0x0 が指定されます。

注意 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- [高速アクセス・メモリ領域を使用する (E)] チェック・ボックスが非チェック状態

(b) [サイズ (S)] 指定領域

ウィザード処理 (チューニング処理) において使用する高速アクセス・メモリ領域のサイズ (単位: K バイト) を指定します。

入力可能な値の範囲は、[高速アクセス・メモリ領域 (M)] コンボ・ボックスにおいて、“内蔵 ROM” を指定した場合は実行エンジンに依存し、“ユーザ指定” を指定した場合は 1 ~ プロジェクトで指定されているアドレス空間の最大値となります。

デフォルトでは、内蔵 ROM が存在する場合はプロジェクトで指定されているチップの内蔵 ROM のサイズ、内蔵 ROM が存在しない場合は 0 が指定されます。

注意 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- [高速アクセス・メモリ領域を使用する (E)] チェック・ボックスが非チェック状態
- 内蔵 ROM を指定した ID850, ID850NW を使用している

チューニング方法選択パネル

ユーザ・プログラム、実行モジュールのチューニング方法（標準パターン、ユーザ指定パターン）を指定するためのパネルです。

なお、チューニング機能についての詳細は、「[4.2 チューニング機能](#)」を参照してください。

図 6-18 チューニング方法選択パネル



ここでは、次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [チューニング方法指定領域](#)

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) チューニング方法指定領域

ユーザ・プログラム、実行モジュールのチューニング方法に関する設定を行う領域です。

なお、本指定領域は、以下に示したラジオ・ボタンから構成されています。

(a) [標準 (D)] ラジオ・ボタン

ユーザ・プログラム、実行モジュールのチューニングを標準パターン (TW850 が推奨するチューニング・パターン) で行うか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“チューニングを標準パターン (TW850 が推奨するチューニング・パターン) で行う” となります。

標準パターンでは、複数のチューニング・オプションを組み合わせたチューニングを数回行い、最も高速化できたチューニング・オプションの組み合わせによってチューニングを行った結果を報告します。チューニングの詳細な理解を必要とせず、簡単に複数のチューニングを自動的に実行することができます。

注意 数回に渡りチューニングとプログラムの実行をするため、チューニングには時間がかかる場合があります。

以下に、TW850 が試行するチューニング・オプションの組み合わせを示します。

【命令キャッシュが存在する場合】

命令キャッシュが存在する場合、以下に示すパターン 1 ~ 3 が試行されます。

チューニング・オプション	パターン 1	パターン 2	パターン 3
高速アクセス・メモリ領域の使用を指定している場合、高速アクセス・メモリ領域のチューニングを行う	○	○	○
高速アクセス・メモリ領域チューニングにおいて、トレース・データ解析方式を用いる場合、実行に時間がかかる関数を優先する プローブ・コード解析方式を用いる場合、コールペアの呼び出し回数が多い関数を優先する	○	○	○
命令キャッシュのチューニングを行う	○	○	○
命令キャッシュのチューニングにおいて、カラリング・アルゴリズムを用い、配置優先度補正オプションとしてコールペア呼び出し回数を優先する	○	-	-
命令キャッシュのチューニングにおいて、カラリング・アルゴリズムを用い、配置優先度補正オプションとしてコールペアの深さを優先する	-	○	-
命令キャッシュのチューニングにおいて、グルーピング・アルゴリズムを用い、配置優先度補正オプションとしてコールペア呼び出し回数を優先する ウエイ数が 2 以上の場合は、ウエイ数を考慮する	-	-	○
コールペア補正オプションは何も指定しない	○	-	○
トレース・データ解析方式を用いる場合、コールペア補正オプションとしてホット・パスにあたる関数を優先する	-	○	-

【命令キャッシュが存在しない場合】

命令キャッシュが存在しない場合、以下に示すパターン 1 ~ 3 が試行されます。

チューニング・オプション	パターン 1	パターン 2	パターン 3
高速アクセス・メモリ領域のチューニングを行う	○	○	○
高速アクセス・メモリ領域チューニングにおいて、コールベアの呼び出し回数が多い関数を優先する	○	○	-
高速アクセス・メモリ領域チューニングにおいて、実行に時間がかかる関数を優先する	-	-	○
コールベア補正オプションは何も指定しない	○	-	○
コールベア補正オプションとして、float ランタイム・ライブラリの関数を優先 (TW850 用システム・ライブラリを使用し、ユーザ・プログラムに float ランタイム・ライブラリ関数が含まれている場合) し、ループ処理内のコールベアを優先する	-	○	-

注意 パターン 3 は、トレース・データ解析方式、あるいは、プローブ・コード解析方式において、関数の実行時間計測を指定する場合のみ試行されます。

【総実行時間が取得できない場合】

総実行時間が取得できない場合、以下に示すパターンが試行されます。

- 高速アクセス・メモリ領域の使用を指定している場合、高速アクセス・メモリ領域のチューニングを行う
- 高速アクセス・メモリ領域のチューニングにおいて、プローブ・コード解析方式を用いる場合、実行に時間がかかる関数を優先する
プローブ・コード解析方式を用いる場合、コールベアの呼び出し回数が多い関数を優先する
- 命令キャッシュが存在する場合、命令キャッシュのチューニングを行う
- 命令キャッシュのチューニングにおいて、カラリング・アルゴリズムを用い、配置優先度補正オプションとしてコールベア呼び出し回数を優先する
- コールベア補正オプションは何も指定しない

(b) [ユーザ指定 (U)] ラジオ・ボタン

ユーザ・プログラム、実行モジュールのチューニングをユーザ指定パターンで行うか否かを指定します。

デフォルトでは、非チェック状態となります。

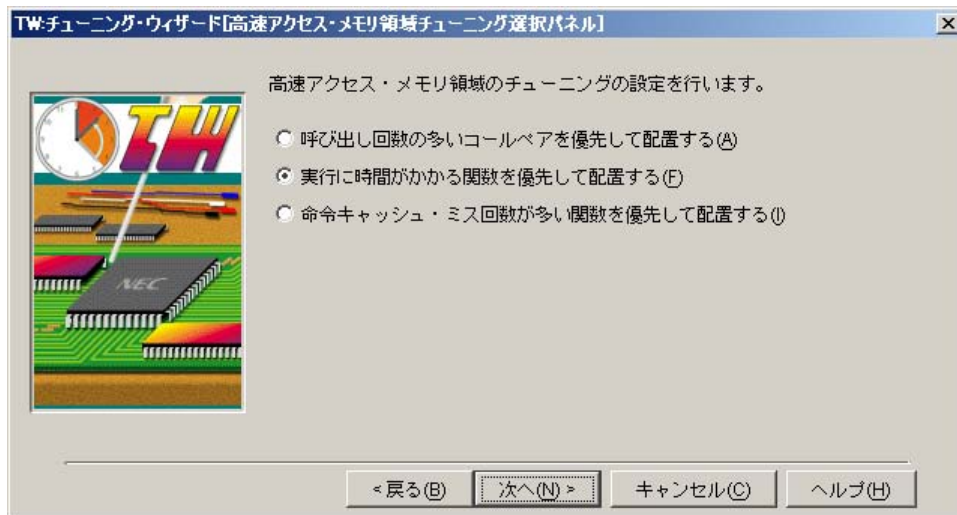
なお、本ラジオ・ボタンがチェック状態の場合、“チューニングをユーザ指定パターンで行う”となります。ユーザ指定パターンでは、一度標準パターンで最適なチューニング・オプションが確定している場合に再度チューニングを行ったり、標準パターンにはない様々なチューニング・オプションを設定してチューニングを行うことができます。

注意 本ラジオ・ボタンがチェック状態の場合に必要な“各種チューニング項目の設定操作”は、<次へ (N)> ボタンを押下することにより継続実行されるウィザード処理 (高速アクセス・メモリ領域チューニング選択パネル、命令キャッシュ・チューニング選択パネル、コールベア補正オプション選択パネルなど) において行います。

高速アクセス・メモリ領域チューニング選択パネル

ユーザ・プログラム、実行モジュールの性能解析時に獲得したコールペア情報をもとに、高速アクセス・メモリ領域（内蔵 ROM など）に対する関数配置方式を指定するためのパネルです。

図 6-19 高速アクセス・メモリ領域チューニング選択パネル



ここでは、次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [関数配置方式指定領域](#)

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) 関数配置方式指定領域

ユーザ・プログラム、実行モジュールの性能解析時に獲得した情報 (コールペア情報、実行時間情報など) をもとに、高速アクセス・メモリ領域 (内蔵 ROM など) に対する関数配置方式に関する設定を行う領域です。

なお、本指定領域は、以下に示したラジオ・ボタンから構成されています。

(a) [呼び出し回数の多いコールペアを優先して配置する (A)] ラジオ・ボタン

ユーザ・プログラム、実行モジュールの性能解析時に獲得したコールペア情報 (関数間の呼び出し関係) をもとに、呼び出し回数の多い関数 (呼び出し関数、呼び出され関数) のうち、コード・サイズの小さい関数を優先的に高速アクセス・メモリ領域に配置するか否かを指定します。

[実行に時間がかかる関数を優先して配置する (E)] ラジオ・ボタンが非チェック状態の場合、デフォルトでチェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“呼び出し回数の多い関数 (呼び出し関数、呼び出され関数) のうち、コード・サイズの小さい関数を優先的に高速アクセス・メモリ領域に配置” となります。

呼び出し回数が多いコード・サイズの小さい関数が多数ある場合や、命令キャッシュが存在し、高速アクセス・メモリがアンキャッシュャブル領域である場合、特にチューニング効果を発揮します。

注意 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 高速アクセス・メモリ領域選択パネルにおいて、“[高速アクセス・メモリ領域を使用する (E)] チェック・ボックス” が非チェック状態

(b) [実行に時間がかかる関数を優先して配置する (E)] ラジオ・ボタン

ユーザ・プログラム、実行モジュールの性能解析時に獲得した各種関数の実行時間情報をもとに、1 命令当たりの実行に時間を要する関数を優先的に高速アクセス・メモリ領域に配置するか否かを指定します。

本ラジオ・ボタンが有効状態の場合、デフォルトでチェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“1 命令当たりの実行に時間を要する関数を優先的に高速アクセス・メモリ領域に配置” となります。

関数の中で複雑な処理をしたり、ループ処理で関数の一部を繰り返し実行するなど、負荷の高い関数に対し、特にチューニング効果を発揮します。

注意 1 本ラジオ・ボタンは、以下に示した全ての条件が成立した場合に無効状態となります。

- 実行項目選択パネルにおいて、“[ブローブ解析を行う (オリジナル実行モジュールのみ) (B)] ラジオ・ボタン” がチェック状態
- ブローブ解析オプション選択パネルにおいて、“[時間計測用ファイルを用いて関数の実行時間を計測する (E)] チェック・ボックス” が非チェック状態

注意 2 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 高速アクセス・メモリ領域選択パネルにおいて、“[高速アクセス・メモリ領域を使用する (E)] チェック・ボックス” が非チェック状態

(c) [命令キャッシュ・ミス回数が多い関数を優先して配置する (I)] ラジオ・ボタン

ユーザ・プログラム、実行モジュールの性能解析時に獲得した各種関数の命令キャッシュ情報 (命令キャッシュ・ヒットの回数、命令キャッシュ・ミスの回数) をもとに、命令キャッシュ・ミスの回数が多い関数を優先的に高速アクセス・メモリ領域に配置するか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“命令キャッシュ・ミスの回数が多い関数を優先的に高速アクセス・メモリ領域に配置” となります。

命令キャッシュ・ミスにより実行速度の低下した関数を高速化するだけでなく、高速アクセス・メモリ領域がアンキャッシュャブル領域である場合は、命令キャッシュ・ミスの多い関数を高速アクセス・メモリに配置することにより、その関数が原因による命令キャッシュ・ミスを無くするというチューニング効果も発揮します。

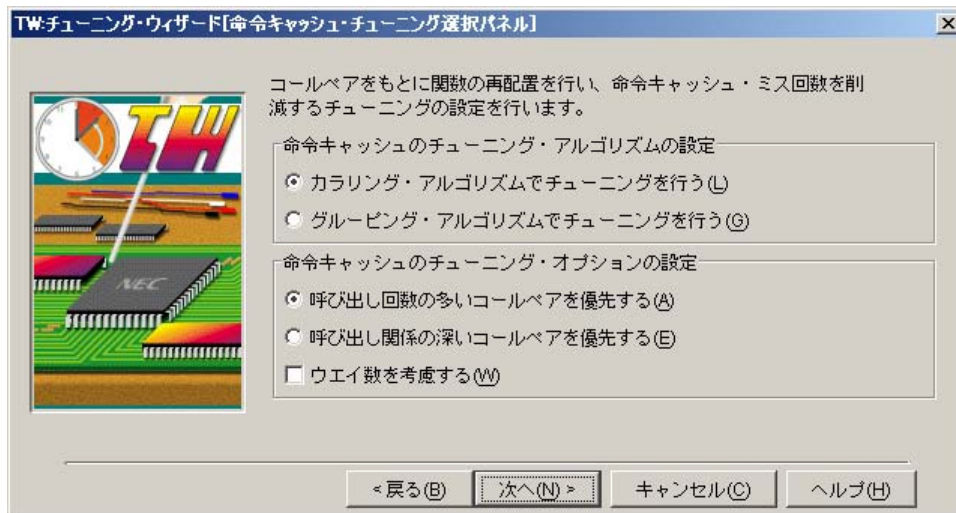
注意 本ラジオ・ボタンは、以下に示した何れかの条件が成立した場合に無効状態となります。

- 実行項目選択パネルにおいて、“[トレース・データ解析を行う (I)] ラジオ・ボタン” が非チェック状態
- 命令キャッシュのサイズが 0
- 命令キャッシュ選択パネルにおいて、“[オリジナル実行モジュールの性能解析時に命令キャッシュを無効にする (D)] チェック・ボックス” がチェック状態
- 高速アクセス・メモリ領域選択パネルにおいて、“[高速アクセス・メモリ領域を使用する (E)] チェック・ボックス” が非チェック状態

命令キャッシュ・チューニング選択パネル

ユーザ・プログラム 実行モジュールの性能解析時に獲得したコールペア情報をもとに、命令キャッシュに対するチューニング方法 (チューニング・アルゴリズム、配置優先度補正オプション) を指定するためのパネルです。

図 6-20 命令キャッシュ・チューニング選択パネル



ここでは、次の項目について説明しています。

- エリア説明

エリア説明

本パネルは、次の項目から構成されています。

- メッセージ表示領域
- チューニング・アルゴリズム指定領域
- 配置優先度補正オプション指定領域

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) チューニング・アルゴリズム指定領域

ユーザ・プログラム、実行モジュールの性能解析時に獲得したコールペア情報をもとに、命令キャッシュに対するチューニング・アルゴリズムに関する設定を行う領域です。

なお、本指定領域は、以下に示したラジオ・ボタンから構成されています。

(a) [カラリング・アルゴリズムでチューニングを行う(L)] ラジオ・ボタン

命令キャッシュに対する関数配置をカラリング・アルゴリズムで行うか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“関数配置をカラリング・アルゴリズムで行う”となります。

注意 1 カラリング・アルゴリズムでは、各種関数のコールペア情報（関数間の呼び出し関係）をもとに、可能な範囲で全てのコールペア同士の命令キャッシュ・ミスが減らすように関数配置を行います。そのため、より広い範囲で命令キャッシュ・ミスが減らすことができますが、チューニング処理に時間がかかる場合があります。また、関数と関数の間にホールを挿入し、すき間を作ることがあるため、ホール・サイズが増加する場合があります。

注意 2 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 命令キャッシュのサイズが 0

(b) [グルーピング・アルゴリズムでチューニングを行う (G)] ラジオ・ボタン

命令キャッシュに対する関数配置をグルーピング・アルゴリズムで行うか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“関数配置をグルーピング・アルゴリズムで行う”となります。カラリング・アルゴリズムよりもアルゴリズムが単純であるため、チューニング処理にかかる時間が短くなります。また、サイズの小さい多くの関数を局所的に何度も呼び出すようなプログラムについて、特にチューニング効果を発揮します。

注意 1 グルーピング・アルゴリズムでは、各種関数のコールペア情報（関数間の呼び出し関係）をもとに、命令キャッシュのサイズをグループ・サイズの上限值として関数間の呼び出し関係が強い関数をグループ化し、グループ内での命令キャッシュ・ミスが減らすように配置を行います。また、グループ内での関数の順序を替えることにより、異なるグループに配属された関数同士の命令キャッシュ・ミスが減らすように配置を行います。

注意 2 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 命令キャッシュのサイズが 0

(3) 配置優先度補正オプション指定領域

ユーザ・プログラム、実行モジュールの性能解析時に獲得したコールペア情報をもとに、命令キャッシュに対する配置優先度補正オプションに関する設定を行う領域です。

なお、本指定領域は、以下に示したラジオ・ボタン、チェック・ボックスから構成されています。

(a) [呼び出し回数の多いコールペアを優先する (A)] ラジオ・ボタン

呼び出し回数の多いコールペアを優先的に命令キャッシュに配置するか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“呼び出し回数の多いコールペアを優先的に命令キャッシュに配置”となります。

注意 1 コールペア補正オプション選択パネル、および、優先配置関数選択パネルの設定項目は、コールペアの呼び出し回数に影響するため、これらのパネルで設定を行う場合は、本ラジオ・ボタンをチェックしておくことでチューニング効果を発揮します。

注意 2 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 命令キャッシュのサイズが 0

(b) [呼び出し関係の深いコールペアを優先する (E)] ラジオ・ボタン

呼び出し関係の末端に近いコールペアを優先的に命令キャッシュに配置するか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“呼び出し関係の末端に近いコールペアを優先的に命令キャッシュに配置”となります。

呼び出し関係の末端の部分で多くの実行時間を消費するようなプログラムについて、特にチューニング効果を発揮します。

注意 本ラジオ・ボタンは、以下に示した条件が成立した場合に無効状態となります。

- 命令キャッシュのサイズが 0

(c) [ウエイ数を考慮する (W)] チェック・ボックス

命令キャッシュに対する関数配置を行う際、キャッシュ機構（ウエイ数）を考慮するか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“キャッシュ機構（ウエイ数）を考慮”となります。

注意 本チェック・ボックスは、以下に示した何れかの条件が成立した場合に無効状態となります。

- 命令キャッシュのサイズが 0
- 命令キャッシュのキャッシュ機構（ウエイ数）が 2 未満

コールペア補正オプション選択パネル

ユーザ・プログラム 実行モジュールの性能解析時に獲得したコールペア情報をもとに、再配置対象関数に対するチューニング方法 (高速アクセス・メモリ・チューニングと命令キャッシュ・チューニングにおけるコールペアの優先度の補正方式) を指定するためのパネルです。

図 6-21 コールペア補正オプション選択パネル



ここでは、次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [コールペア補正オプション指定領域](#)

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) コールペア補正オプション指定領域

ユーザ・プログラム、実行モジュールの性能解析時に獲得したコールペア情報をもとに、再配置対象関数に対するチューニング方法 (高速アクセス・メモリ・チューニングと命令キャッシュ・チューニングにおけるコールペアの優先度の補正方式) に関する設定を行う領域です。

なお、本指定領域は、以下に示したチェック・ボックスから構成されています。

(a) [float ランタイム・ライブラリの関数を優先して配置する (R)] チェック・ボックス

float ランタイム・ライブラリ関数 (浮動小数点演算などを行った場合に、コンパイラによって使用されるライブラリ関数) を優先的に命令キャッシュ、または、高速アクセス・メモリ領域に配置するか否かを指定します。デフォルトでは、非チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“ float ランタイム・ライブラリに含まれている関数 (expf、cosf など) を優先的に命令キャッシュ、または、高速アクセス・メモリ領域に配置 ” となります。

float ランタイム・ライブラリ関数を特に高速化したい場合、float ランタイム・ライブラリ関数に関わるコールペアの呼び出し回数を補正し、実行を高速に行えるように関数を配置することができます。

注意 1 本チェック・ボックスは、以下に示した全ての条件が成立した場合に無効状態となります。

- 高速アクセス・メモリ領域チューニング選択パネルにおいて、[呼び出し回数の多いコールペアを優先して配置する (A)] ラジオ・ボタン以外を選択
- 命令キャッシュのサイズが 0

注意 2 本チェック・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- float ランタイム・ライブラリに含まれている関数 (expf、cosf など) がユーザ・プログラム、実行モジュールに含まれていない

(b) [ホット・パスにあたる関数を優先して配置する (Q)] チェック・ボックス

ホット・パスにあたる関数 (プログラム中で、最も実行時間のかかる関数遷移のパス) を優先的に命令キャッシュ、または、高速アクセス・メモリ領域に配置するか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“ ホット・パスにあたる関数を優先的に命令キャッシュ、または、高速アクセス・メモリ領域に配置 ” となります。

ホット・パスにあたる関数を特に高速化したい場合、関数の子関数を含めた実行時間 (GROSS 時間) の情報をもとに、その関数に関わるコールペアの呼び出し回数を補正し、実行を高速に行えるように関数を配置することができます。

注意 1 本チェック・ボックスは、以下に示した全ての条件が成立した場合に無効状態となります。

- 高速アクセス・メモリ領域チューニング選択パネルにおいて、[呼び出し回数の多いコールペアを優先して配置する (A)] ラジオ・ボタン以外を選択
- 命令キャッシュのサイズが 0

注意 2 本チェック・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルにおいて、[プローブ解析を行う (オリジナル実行モジュールのみ) (E)] ラジオ・ボタンを選択

(c) [ループ処理内のコールペアを優先して配置する (L)] チェック・ボックス

同一ループ処理に関わるコールペア関数を優先的に命令キャッシュ、または、高速アクセス・メモリ領域に配置するか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“ 同一ループ処理に関わるコールペア関数を優先的に命令キャッシュ、または、高速アクセス・メモリ領域に配置 ” となります。

ループ内での関数呼び出し箇所が少なく、その箇所を特に高速化したい場合、同一ループ処理に関わるコールペアの呼び出し回数を補正し、ループ内の関数の実行を高速に行えるように関数を配置することができます。

注意 1 ループ内での関数呼び出し箇所が多い場合、チューニングに時間がかかる場合があります。

注意 2 本チェック・ボックスは、以下に示した全ての条件が成立した場合に無効状態となります。

- 高速アクセス・メモリ領域チューニング選択パネルにおいて、[呼び出し回数の多いコールペアを優先して配置する (A)] ラジオ・ボタン以外を選択
- 命令キャッシュのサイズが 0

注意 TW850 では、複数のチェック・ボックスが同時に指定された場合、以下に示した順序でコールペア情報の補正を行います。

float ランタイム・ライブラリ > ホット・パスにあたる関数 > ループ処理内関数

ただし、すべてのコールペア補正オプションを指定することによって、必ずしも良い結果となるわけではありません。優先度を補正すると、補正されたコールペアは優先度が上がるため、優先して配置されやすくなりますが、逆に、補正されなかったコールペアは優先度が下がることになります。

優先配置関数選択パネル

TW850 のウィザード処理（チューニング処理）において、高速アクセス・メモリ・チューニングと命令キャッシュ・チューニングにおけるコールペアの優先度を補正するために、優先的に再配置対象とする関数を指定するためのパネルです。

図 6-22 優先配置関数選択パネル



ここでは、次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [\[優先して配置する関数を指定する\(F\)\] チェック・ボックス](#)
- [\[優先した関数間の優先度をリスト順で指定する\(P\)\] チェック・ボックス](#)
- [再配置対象関数指定領域](#)
- [再配置対象関数表示領域](#)

(1)メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2)[優先して配置する関数を指定する(E)]チェック・ボックス

ウィザード処理(チューニング処理)において、高速アクセス・メモリ・チューニングと命令キャッシュ・チューニングにおけるコールペアの優先度を補正するために、優先的に再配置対象とする関数を指定するか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“優先的に再配置対象とする関数を指定する”となります。特に優先して高速化したい関数がある場合、指定した関数の実行を高速に行えるように関数を配置することができます。

注意 1 指定した関数は、高速アクセス・メモリ・チューニングで最優先(すべてのコールペア補正オプションよりも優先)に配置の対象となり、配置しなかった場合は命令キャッシュ・チューニングで最優先に配置の対象となります。

注意 2 優先的に再配置対象とする関数は、任意の関数を指定することができますが、多数の関数を指定すると、チューニングの効果が薄れてしまう場合があります。

(3)[優先した関数間の優先度をリスト順で指定する(P)]チェック・ボックス

ウィザード処理(チューニング処理)において、本パネルの再配置対象関数表示領域の表示順(上側:優先度高,下側:優先度低)に関数再配置処理を実行するか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“再配置対象関数表示領域の表示順(上:優先度高,下:優先度低)に関数再配置処理を実行する”となります。

特に高速化したい関数の順序を指定したい場合、指定した関数リスト順の優先度で関数を配置することができます。

本チェック・ボックスが非チェック状態の場合、高速アクセス・メモリ・チューニングは[高速アクセス・メモリ領域選択パネル](#)で指定した設定項目をもとに優先度を決定し、命令キャッシュ・チューニングはすべて同じ優先度としてチューニングを行います。

注意 本チェック・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- [優先して配置する関数を指定する(E)]チェック・ボックスが非チェック状態

(4)再配置対象関数指定領域

ウィザード処理(チューニング処理)において、優先的に再配置対象とする関数を指定します。

なお、本指定領域は、以下に示したコンボ・ボックス、ボタン群から構成されています。

(a)再配置対象関数名指定コンボ・ボックス

ウィザード処理(チューニング処理)において、優先的に再配置対象とする関数の関数名を選択します。

選択項目は、オリジナル実行モジュール情報を解析して得られた関数名となります。

デフォルトでは、何も指定されません。

注意 1 同じ名前の関数が複数存在する場合、“関数名(ファイル名)”のように表示されます。

また、関数名、および、ファイル名が同じ名前である関数が複数存在する場合、“関数名(絶対パス名)”のように表示されます。

注意 2 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- [優先して配置する関数を指定する(E)]チェック・ボックスが非チェック状態

(b) <追加(A)>ボタン

再配置対象関数名指定コンボ・ボックスにおいて選択した関数を“優先的に再配置対象とする関数”として登録します。

なお、本ボタンの押下により、再配置対象関数表示領域の最下位に再配置対象関数名指定コンボ・ボックスで指定した関数名が追加されます。

注意 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- [優先して配置する関数を指定する(E)]チェック・ボックスが非チェック状態

(c) <削除 (D) > ボタン

再配置対象関数名指定コンボ・ボックスにおいて選択した関数を“優先的に再配置対象とする関数”から登録解除します。

なお、本ボタンの押下により、再配置対象関数名指定コンボ・ボックスで選択した関数名が再配置対象関数表示領域から削除されます。

注意 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- [優先して配置する関数を指定する (E)] チェック・ボックスが非チェック状態

(d) <上へ (U) > ボタン

再配置対象関数名指定コンボ・ボックスにおいて選択した関数の優先順位を1レベル上位に変更します。

なお、本ボタンの押下により、再配置対象関数名指定コンボ・ボックスで選択した関数名が再配置対象関数表示領域において、1レベル上位側に移動します。

注意 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- [優先して配置する関数を指定する (E)] チェック・ボックスが非チェック状態

(e) <下へ (W) > ボタン

再配置対象関数名指定コンボ・ボックスにおいて選択した関数の優先順位を1レベル下位に変更します。

なお、本ボタンの押下により、再配置対象関数名指定コンボ・ボックスで選択した関数名が再配置対象関数表示領域において、1レベル下位側に移動します。

注意 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- [優先して配置する関数を指定する (E)] チェック・ボックスが非チェック状態

(5) 再配置対象関数表示領域

ウィザード処理 (チューニング処理) において、優先的に再配置対象とする関数の関数名を一覧表示しています。デフォルトでは、何も表示されません。

注意 本表示領域は、以下に示した条件が成立した場合に無効状態となります。

- [優先して配置する関数を指定する (E)] チェック・ボックスが非チェック状態

配置先選択パネル

チューニング時に再配置対象となった関数の再配置用メモリ領域、および、性能解析時に中間ファイルとして出力されるプロファイル・ルーチン、プロファイル・テーブルの配置用メモリ領域を指定するためのパネルです。

図 6-23 配置先選択パネル



ここでは、次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [再配置対象関数指定領域](#)
- [プロファイル・ルーチン指定領域](#)
- [プロファイル・テーブル指定領域](#)

(1) メッセージ表示領域

本パネルに関する解説メッセージを表示しています。

(2) 再配置対象関数指定領域

チューニング時に再配置対象となった関数の再配置用メモリ領域に関する設定を行う領域です。

ただし、高速アクセス・メモリ領域に配置される関数は対象外となります。

なお、本指定領域は、以下に示したコンボ・ボックス、および、指定領域から構成されています。

(a) [挿入位置 (I)] コンボ・ボックス

チューニング時に再配置対象となった関数の再配置用メモリ領域の挿入位置を選択します。

選択項目は、“指定領域へ”、“指定領域の後”、“指定領域の前”、“指定アドレス”となります。

デフォルトでは、“指定領域へ”が選択状態となります。

注意 1 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの[チューニングとチューニング後実行モジュールの解析を行う(Q)]チェック・ボックスが非チェック状態

注意 2 本コンボ・ボックスは、以下に示した条件が成立した場合に“指定領域へ”以外の項目が無効状態となります。

- 対象リンク・ディレクティブ・ファイルに TWTEXT セグメントを定義

(b) [領域 (P)] コンボ・ボックス

チューニング時に再配置対象となった関数の再配置用メモリ領域の挿入対象領域(セグメント名)を選択します。

選択項目は、リンク・ディレクティブ・ファイルに指定されている text 属性のセグメントすべて、リンク・ディレクティブ・ファイルに指定されているセクションすべてとなります。

デフォルトでは、“TWTEXT”があれば“TWTEXT”、次に“TEXT”があれば“TEXT”が選択状態となります。

通常は、“TEXT セグメント領域”、または、“.text セクションの後”を指定します。

注意 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの[チューニングとチューニング後実行モジュールの解析を行う(Q)]チェック・ボックスが非チェック状態
- 対象リンク・ディレクティブ・ファイルに TWTEXT セグメントを定義

(c) [アドレス (A)] 指定領域

チューニング時に再配置対象となった関数の再配置用メモリ領域の挿入対象領域(アドレス)を指定します。

入力可能な値の範囲は、0x0 ~ プロジェクトで指定されているチップの最大アドレスとなります。

デフォルトでは、何も指定されません。

注意 1 本指定領域に指定できるのは、4 の倍数のみです。

注意 2 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの[チューニングとチューニング後実行モジュールの解析を行う(Q)]チェック・ボックスが非チェック状態
- [挿入位置 (I)] コンボ・ボックスにおいて、“指定アドレス”以外を選択

(3) プロファイル・ルーチン指定領域

性能解析時、チューニング時に中間ファイルとして出力されるプロファイル・ルーチンの配置用メモリ領域に関する設定を行う領域です。

注意 プロファイル・ルーチンを解析対象のプログラムと異なる性能のメモリに配置した場合、誤差が大きくなります。プロファイル・ルーチンは、解析対象のプログラムと同じ性能のメモリに配置してください。

なお、本指定領域は、以下に示したコンボ・ボックス、および、指定領域から構成されています。

(a) [挿入位置 (S)] コンボ・ボックス

性能解析時、チューニング時に中間ファイルとして出力されるプロファイル・ルーチンの配置用メモリ領域の挿入位置を選択します。

選択項目は、“指定領域へ”、“指定領域の後”、“指定領域の前”、“指定アドレス”となります。

デフォルトでは、“指定領域へ”が選択状態となります。

注意 1 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態
- 実行項目選択パネルにおいて、[プローブ解析を行う (オリジナル実行モジュールのみ) (E)] ラジオ・ボタン が非チェック状態

注意 2 本コンボ・ボックスは、以下に示した条件が成立した場合に“指定領域へ”以外の項目が無効状態となります。

- 対象リンク・ディレクティブ・ファイルに TWROUTINE セグメントを定義

(b) [領域 (Q)] コンボ・ボックス

性能解析時、チューニング時に中間ファイルとして出力されるプロファイル・ルーチンの配置用メモリ領域の挿入対象領域 (セグメント名) を選択します。

選択項目は、リンク・ディレクティブ・ファイルに指定されている text 属性のセグメントすべて、リンク・ディレクティブ・ファイルに指定されているセクションすべてとなります。

デフォルトでは、“TWROUTINE” があれば “TWROUTINE”、次に “TEXT” があれば “TEXT” が選択状態となります。

通常は、「TEXT セグメント領域」、または、「.text セクションの後」を指定します。

注意 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態
- 実行項目選択パネルにおいて、[プローブ解析を行う (オリジナル実行モジュールのみ) (E)] ラジオ・ボタン が非チェック状態
- 対象リンク・ディレクティブ・ファイルに TWROUTINE セグメントを定義

(c) [アドレス (D)] 指定領域

性能解析時、チューニング時に中間ファイルとして出力されるプロファイル・ルーチンの配置用メモリ領域の挿入対象領域 (アドレス) を指定します。

入力可能な値の範囲は、0x0 ~ プロジェクトで指定されているチップの最大アドレスとなります。

デフォルトでは、何も指定されません。

注意 1 本指定領域に指定できるのは、4 の倍数のみです。

注意 2 本指定領域は、以下に示した何れかの条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態
- 実行項目選択パネルにおいて、[プローブ解析を行う (オリジナル実行モジュールのみ) (E)] ラジオ・ボタン が非チェック状態
- [挿入位置 (S)] コンボ・ボックスにおいて、“指定アドレス” 以外を選択

(4) プロファイル・テーブル指定領域

性能解析時、チューニング時に中間ファイルとして出力されるプロファイル・テーブルの配置用メモリ領域に関する設定を行う領域です。

十分に領域のある、書き込み可能な領域を指定してください。また、配置先が RAM 領域になるように指定してください。

なお、本指定領域は、以下に示したコンボ・ボックス、指定領域、表示領域、および、ボタン群から構成されています。

(a) [挿入位置 (E)] コンボ・ボックス

性能解析時、チューニング時に中間ファイルとして出力されるプロファイル・テーブルの配置用メモリ領域の挿入位置を選択します。

選択項目は、“指定領域へ”、“指定領域の後”、“指定領域の前”、“指定アドレス” となります。

デフォルトでは、“指定領域へ” が選択状態となります。

注意 本コンボ・ボックスは、以下に示した条件が成立した場合に“指定領域へ”以外の項目が無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態
- 実行項目選択パネルにおいて、“[プローブ解析を行う (オリジナル実行モジュールのみ) (P)] ラジオ・ボタン” が非チェック状態
- 対象リンク・ディレクティブ・ファイルに TWTABLE セグメントを定義

(b) [領域 (I)] コンボ・ボックス

性能解析時、チューニング時に中間ファイルとして出力されるプロファイル・テーブルの配置用メモリ領域の挿入対象領域 (セグメント名) を選択します。

選択項目は、リンク・ディレクティブ・ファイルに指定されている data 属性のセグメントすべて、リンク・ディレクティブ・ファイルに指定されているセクションすべてとなります。

デフォルトでは、“TWTABLE”があれば“TWTABLE”、次に“DATA”があれば“DATA”が選択状態となります。

通常は、「DATA セグメント領域」を指定します。

注意 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態
- 実行項目選択パネルにおいて、“[プローブ解析を行う (オリジナル実行モジュールのみ) (P)] ラジオ・ボタン” が非チェック状態
- 対象リンク・ディレクティブ・ファイルに TWTABLE セグメントを定義

(c) [アドレス (R)] 指定領域

性能解析時、チューニング時に中間ファイルとして出力されるプロファイル・テーブルの配置用メモリ領域の挿入対象領域 (アドレス) を指定します。

入力可能な値の範囲は、0x0 ~ プロジェクトで指定されているチップの最大アドレスとなります。

デフォルトでは、何も指定されません。

注意 1 本指定領域に指定できるのは、4 の倍数のみです。

注意 2 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- 実行項目選択パネルの [オリジナル実行モジュールの性能解析を行う (A)] チェック・ボックスが非チェック状態
- 実行項目選択パネルにおいて、“[プローブ解析を行う (オリジナル実行モジュールのみ) (P)] ラジオ・ボタン” が非チェック状態
- [挿入位置 (E)] コンボ・ボックスにおいて、“指定アドレス” 以外を選択

(d) サイズ表示領域

< 計算する (L) > ボタンを押下することにより得られるプロファイル・テーブルのサイズ (単位: バイト) を表示します。

ただし、実行項目選択パネルにおいて、“[プローブ解析を行う (オリジナル実行モジュールのみ) (P)] ラジオ・ボタン” が非チェック状態の場合、“なし” が表示されます。

デフォルトでは、“未計算” が表示されます。

注意 計算によるサイズにはホール・サイズが含まれていないため、実際に占有するメモリ容量はホール・サイズ分増加します。

(e) < 計算する (L) > ボタン

プロファイル・テーブルのサイズ (単位: バイト) を算出した後、算出結果をサイズ表示領域に表示します。ただし、計算には時間がかかる場合があります。

注意 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- プロファイル・テーブルのサイズが計算済

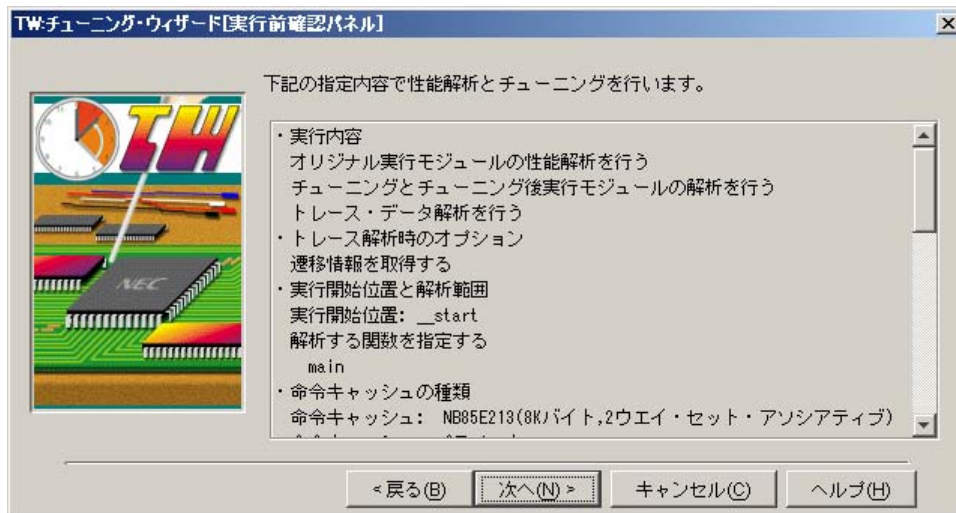
(f) < 中止 (L) > ボタン

プロファイル・テーブルのサイズの計算を中断します。

実行前確認パネル

TW850 がウィザード処理（性能解析処理，チューニング処理）を実行するうえで必要となる性能解析項目，チューニング項目の設定操作が完了したことを通知するためのパネルです。

図 6-24 実行前確認パネル



ここでは，次の項目について説明しています。

- ・ [エリア説明](#)

エリア説明

本パネルは，次の項目から構成されています。

- ・ [メッセージ表示領域](#)

(1) メッセージ表示領域

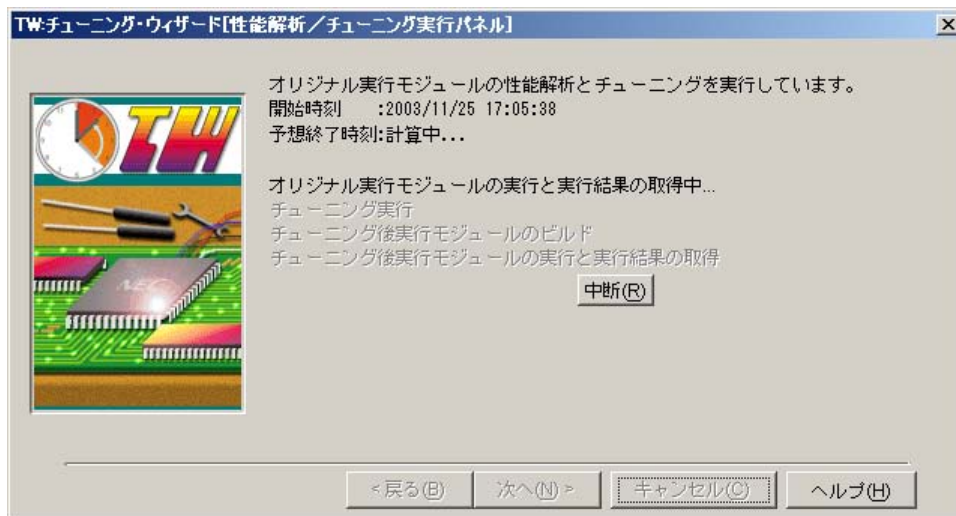
TW850 がウィザード処理（性能解析処理，チューニング処理）を実行するうえで必要となる性能解析項目／チューニング項目の設定操作が完了したことを通知するためのメッセージを表示します。

性能解析 / チューニング実行パネル

TW850 のウィザード処理 (性能解析処理, チューニング処理) が実行中であることを通知するためのパネルです。以下に, TW850 の実行内容を示します。

- オリジナル解析用実行モジュールの性能解析処理を行うとき, オリジナル解析用実行モジュールの実行, 性能解析結果の取得
- プローブ・コード挿入方式を用いて性能解析処理を行うとき, プロファイル用実行モジュールのビルド, 実行, 性能解析結果の取得
- チューニング, および, チューニング後実行モジュールの性能解析処理を行うとき, チューニング, ビルド, チューニング後実行モジュールの実行, 性能解析結果, および, チューニング結果の取得

図 6-25 性能解析 / チューニング実行パネル



ここでは, 次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは, 次の項目から構成されています。

- [メッセージ表示領域](#)

(1) メッセージ表示領域

TW850 のウィザード処理 (性能解析処理, チューニング処理) が実行中であることを通知するためのメッセージを表示します。

メッセージの内容は, 現在行っている処理の実行状況, および, 開始時刻, 予想終了時刻となります。また, プロファイル用実行モジュールを生成する場合, プロファイル・テーブルのサイズも表示されます。

注意 計算によるプロファイル・テーブルのサイズには, ホール・サイズが含まれていないため, 実際に占有するメモリ容量はホール・サイズ分増加します。

(a) <中断 (R) > ボタン

ウィザード処理 (性能解析処理, チューニング処理) を中止します。

注意 本ボタンを押下してから実際にウィザード処理が中止されるまでの間には, 時間を要する場合があります。

(b) <再試行 (R) > ボタン

ウィザード処理 (性能解析処理, チューニング処理) を再実行します。

実行完了確認パネル

TW850 のウィザード処理 (性能解析処理 , チューニング処理) が完了したことを通知するためのパネルです。

図 6-26 実行完了確認パネル



ここでは、次の項目について説明しています。

- [エリア説明](#)

エリア説明

本パネルは、次の項目から構成されています。

- [メッセージ表示領域](#)

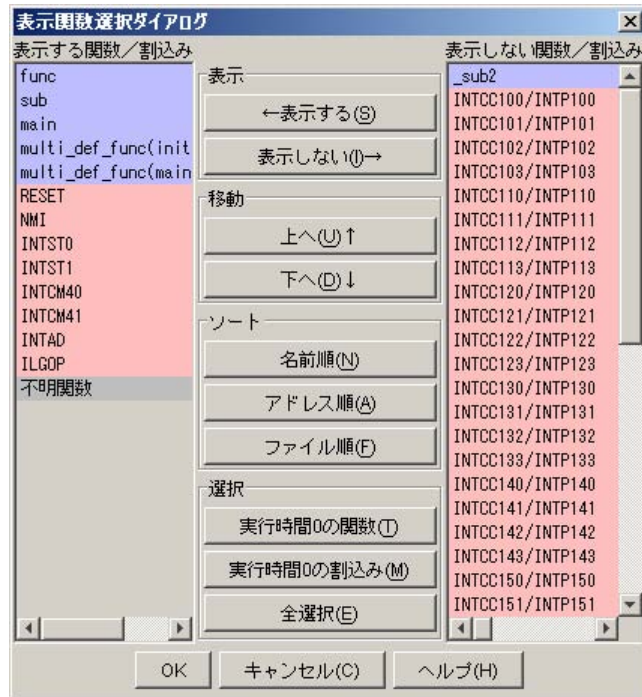
(1) メッセージ表示領域

TW850 のウィザード処理 (性能解析処理 , チューニング処理) が正常終了したことを通知するためのメッセージ (出力ファイルの格納フォルダ名 , および , 出力ファイル名など) を表示します。

表示関数選択ダイアログ

関数遷移情報ウィンドウに表示する関数，または，割り込みを選択するためのダイアログです。

図 6-27 表示関数選択ダイアログ



ここでは，次の項目について説明しています。

- オープン方法
- エリア説明

オープン方法

本ダイアログは，次の方法でオープンすることができます。

- メイン・ウィンドウにおいて，関数遷移情報ウィンドウをアクティブにした後，
 - 右クリック → [表示関数選択 (D)...] を選択したとき
 - [表示 (V)] メニュー → [表示関数選択 (D)...] を選択したとき

エリア説明

本ダイアログは、次の項目から構成されています。

- [表示する関数 / 割り込み] 領域
- 関数表示指定領域
- 関数移動指定領域
- 関数ソート指定領域
- 関数選択指定領域
- [表示しない関数 / 割り込み] 領域
- 機能ボタン

(1) [表示する関数 / 割り込み] 領域

関数遷移情報ウィンドウに表示する関数、および、割り込みを選択する領域です。

デフォルトでは、その時点で関数遷移情報ウィンドウに表示されている関数、および、割り込みが同じ順番で表示されます。

注意 1 関数、および、割り込みをクリックした場合、その項目は選択状態となります。また、関数、および、割り込みは、Ctrl キーを押下しながらクリックすることにより、複数選択することが可能です。

注意 2 関数、および、割り込みをドラッグすることにより、本領域内において、関数、および、割り込みを移動することができます。
ただし、関数と割り込みを混在して複数選択している場合、移動することはできません。

(2) 関数表示指定領域

関数遷移情報ウィンドウに関数、および、割り込みを表示するか否かを指定する領域です。

なお、本指定領域は、以下に示したボタンから構成されています。

(a) < ← 表示する (S) > ボタン

[表示しない関数 / 割り込み] 領域で選択された関数、または、割り込みを [表示する関数 / 割り込み] 領域の最下位に追加します。

注意 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- [表示しない関数 / 割り込み] 領域において、関数、または、割り込みが非選択状態

(b) < 表示しない (I) → > ボタン

[表示する関数 / 割り込み] 領域で選択された関数、または、割り込みを [表示しない関数 / 割り込み] 領域の最下位に追加します。

注意 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- [表示する関数 / 割り込み] 領域において、関数、または、割り込みが非選択状態

(3) 関数移動指定領域

[表示する関数 / 割り込み] 領域、および、[表示しない関数 / 割り込み] 領域で選択された関数、および、割り込みをひとつ上位、または、下位に移動する領域です。

なお、本指定領域は、以下に示したボタンから構成されています。

(a) < 上へ (U) ↑ > ボタン

[表示する関数 / 割り込み] 領域、または、[表示しない関数 / 割り込み] 領域で選択された関数、または、割り込みをひとつ上位に移動します。

注意 1 本ボタンは、以下に示した条件が成立した場合に無効状態となります。

- [表示する関数 / 割り込み] 領域、または、[表示しない関数 / 割り込み] 領域において、関数、または、割り込みが非選択状態

注意 2 割り込みは、関数よりも上位には移動できません。

(b) <下へ (D) ↓ > ボタン

[表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域で選択された関数, または, 割り込みをひとつ下位に移動します。

注意 1 本ボタンは, 以下に示した条件が成立した場合に無効状態となります。

- [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域において, 関数, または, 割り込みが非選択状態

注意 2 関数は, 割り込みよりも下位には移動できません。

(4) 関数ソート指定領域

[表示する関数 / 割り込み] 領域, および, [表示しない関数 / 割り込み] 領域において, 関数, および, 割り込みをソートする領域です。

なお, 本指定領域は, 以下に示したボタンから構成されています。

(a) <名前順 (N) > ボタン

選択されている [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域において, 関数, および, 割り込みを名前順でソートします。

注意 1 本ボタンは, 以下に示した条件が成立した場合に無効状態となります。

- [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域が非選択状態

注意 2 関数は, 割り込みよりも常に上位に表示されます。

(b) <アドレス順 (A) > ボタン

選択されている [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域において, 関数, および, 割り込みをアドレス順でソートします。

注意 1 本ボタンは, 以下に示した条件が成立した場合に無効状態となります。

- [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域が非選択状態

注意 2 関数は, 割り込みよりも常に上位に表示されます。

(c) <ファイル順 (F) > ボタン

選択されている [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域において, 関数を関数が所属しているファイル順でソートします。

注意 1 本ボタンは, 以下に示した条件が成立した場合に無効状態となります。

- [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域が非選択状態

注意 2 同じファイルに所属する関数が複数存在する場合, それらの関数についてはソート前の表示順のままとなります。

(5) 関数選択指定領域

[表示する関数 / 割り込み] 領域, および, [表示しない関数 / 割り込み] 領域において, 実行時間が 0 である関数, または, 割り込み, または, 関数, および, 割り込みをすべて選択する領域です。

なお, 本指定領域は, 以下に示したボタンから構成されています。

(a) <実行時間 0 の関数 (I) > ボタン

選択されている [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域において, 実行時間が 0 である関数を選択状態にします。

注意 本ボタンは, 以下に示した条件が成立した場合に無効状態となります。

- [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域が非選択状態

(b) <実行時間 0 の割り込み (M) > ボタン

選択されている [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域において, 実行時間が 0 である割り込みを選択状態にします。

注意 本ボタンは, 以下に示した条件が成立した場合に無効状態となります。

- [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域が非選択状態

(c) <全選択 (E) > ボタン

選択されている [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域において, 関数, および, 割り込みをすべて選択状態にします。

注意 本ボタンは, 以下に示した条件が成立した場合に無効状態となります。

- [表示する関数 / 割り込み] 領域, または, [表示しない関数 / 割り込み] 領域が非選択状態

(6) [表示しない関数 / 割り込み] 領域

関数遷移情報ウィンドウに表示しない関数, および, 割り込みを選択する領域です。

デフォルトでは, その時点で関数遷移情報ウィンドウに表示されていない関数, および, 割り込みが表示されます。

注意 1 関数, および, 割り込みをクリックした場合, その項目は選択状態となります。また, 関数, および, 割り込みは, 複数選択することが可能です。

注意 2 関数, および, 割り込みをドラッグすることにより, 本領域内において, 関数, および, 割り込みを移動することができます。
ただし, 関数と割り込みを混在して複数選択している場合, 移動することはできません。

(7) 機能ボタン

機能ボタンは, 以下に示したボタン群から構成されています。

(a) < OK > ボタン

表示関数リストの関数および割り込みを表示するよう関数遷移情報ウィンドウに設定した後, 本ダイアログを閉じます。

(b) < キャンセル (C) > ボタン

本ダイアログを閉じます。

(c) < ヘルプ (H) > ボタン

ヘルプを起動します。

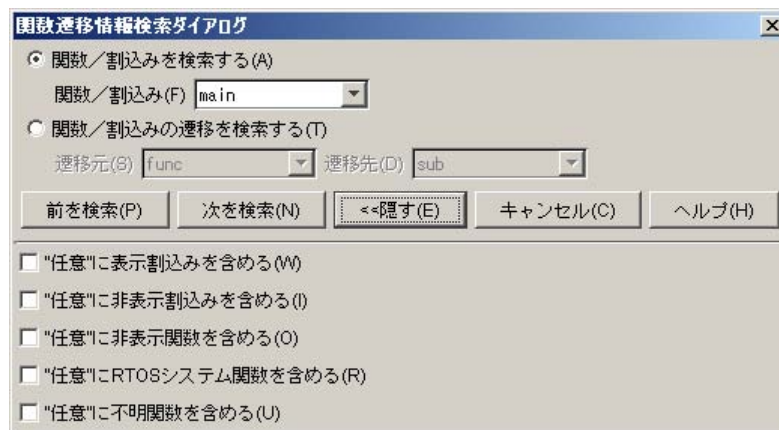
関数遷移情報検索ダイアログ

関数遷移情報ウィンドウにおいて、関数実行線、または、関数遷移線を検索するためのダイアログです。

図 6-28 関数遷移情報検索ダイアログ



図 6-29 関数遷移情報検索ダイアログ (拡張検索条件表示時)



ここでは、次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ダイアログは、次の方法でオープンすることができます。

- [メイン・ウィンドウ](#)において、[関数遷移情報ウィンドウ](#)を選択した後、
 - [編集(E)]メニュー→[検索(S)...]を選択したとき
- [関数遷移情報ウィンドウ](#)において、
 - 検索ボタンを押下したとき

エリア説明

本ダイアログは、次の項目から構成されています。

- 関数検索条件指定領域
- 関数遷移検索条件指定領域
- 拡張検索条件指定領域
- 機能ボタン

(1) 関数検索条件指定領域

関数遷移情報ウィンドウで検索するために、関数に関する検索条件を指定する領域です。

なお、本指定領域は、以下に示したラジオ・ボタン、コンボ・ボックスから構成されています。

(a) [関数 / 割り込みを検索する (A)] ラジオ・ボタン

関数、または、割り込みを検索するか否かを指定します。

関数遷移情報ウィンドウにおいて、何も選択されていない場合、または、関数ボックス、割り込みボックス、関数実行線の何れかが選択されている場合、デフォルトでチェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“関数、または、割り込みを検索する”となります。

(b) [関数 / 割り込み (E)] コンボ・ボックス

検索する関数名、または、割り込み名を指定します。

選択項目は、性能解析処理を行って得られた関数名、および、プロジェクトで指定されているチップのすべての割り込み名となります。また、任意の関数や割り込みなど意味する項目“任意”もリストに表示されます。

関数遷移情報ウィンドウにおいて、関数ボックス、割り込みボックス、関数実行線の何れかが選択されている場合、デフォルトで選択されている関数名、または、割り込み名が表示されます。

注意 1 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- [関数 / 割り込みを検索する (A)] ラジオ・ボタンが非チェック状態

注意 2 同じ名前の関数が複数存在する場合、“関数名 (ファイル名)” のように表示されます。

また、関数名、および、ファイル名が同じ名前である関数が複数存在する場合、“関数名 (絶対パス名)” のように表示されます。

(2) 関数遷移検索条件指定領域

関数遷移情報ウィンドウで検索するために、関数遷移に関する検索条件を指定する領域です。

なお、本指定領域は、以下に示したラジオ・ボタン、コンボ・ボックスから構成されています。

(a) [関数 / 割り込みの遷移を検索する (I)] ラジオ・ボタン

関数、または、関数遷移線を検索するか否かを指定します。

関数遷移情報ウィンドウにおいて、関数遷移線が選択されている場合、デフォルトでチェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、“関数、または、関数遷移線を検索する”となります。

(b) [遷移元 (S)] コンボ・ボックス

検索条件の遷移元となる関数名、または、割り込み名を指定します。

選択項目は、性能解析処理を行って得られた関数名、および、プロジェクトで指定されているチップのすべての割り込み名となります。また、任意の関数や割り込みなど意味する項目“任意”もリストに表示されます。

関数遷移情報ウィンドウにおいて、関数遷移線が選択されている場合、デフォルトで選択されている関数遷移線の遷移元である関数名、または、割り込み名が表示されます。

注意 1 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- [関数 / 割り込みの遷移を検索する (I)] ラジオ・ボタンが非チェック状態

注意 2 同じ名前の関数が複数存在する場合、“関数名 (ファイル名)” のように表示されます。

また、関数名、および、ファイル名が同じ名前である関数が複数存在する場合、“関数名 (絶対パス名)” のように表示されます。

(c) [遷移先 (D)] コンボ・ボックス

検索条件の遷移先となる関数名、または、割り込み名を指定します。

選択項目は、性能解析処理を行って得られた関数名、および、プロジェクトで指定されているチップのすべての割り込み名となります。また、任意の関数や割り込みなど意味する項目“任意”もリストに表示されます。[関数遷移情報ウィンドウ](#)において、関数遷移線が選択されている場合、デフォルトで選択されている関数遷移線の遷移先である関数名、または、割り込み名が表示されます。

注意 1 本コンボ・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- [関数 / 割り込みの遷移を検索する (I)] ラジオ・ボタンが非チェック状態

注意 2 同じ名前の関数が複数存在する場合、“関数名 (ファイル名) ” のように表示されます。

また、関数名、および、ファイル名が同じ名前である関数が複数存在する場合、“関数名 (絶対パス名) ” のように表示されます。

(3) 拡張検索条件指定領域

[関数遷移情報ウィンドウ](#)で検索するために、関数、および、関数遷移に関する拡張検索条件を指定する領域です。なお、本指定領域は、以下に示したチェック・ボックスから構成されています。

(a) [“任意” に表示割り込みを含める (W)] チェック・ボックス

[関数 / 割り込み (E)] コンボ・ボックス、[遷移元 (S)] コンボ・ボックス、および、[遷移先 (D)] コンボ・ボックスに“任意”が指定された場合、[関数遷移情報ウィンドウ](#)に表示されている割り込みを検索対象とすることを指定するか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“[関数遷移情報ウィンドウ](#)に表示されている割り込みを検索対象とする”となります。

(b) [“任意” に非表示割り込みを含める (I)] チェック・ボックス

[関数 / 割り込み (E)] コンボ・ボックス、[遷移元 (S)] コンボ・ボックス、および、[遷移先 (D)] コンボ・ボックスに“任意”が指定された場合、[関数遷移情報ウィンドウ](#)に表示されていない割り込みを検索対象とすることを指定するか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“[関数遷移情報ウィンドウ](#)に表示されていない割り込みを検索対象とする”となります。

(c) [“任意” に非表示関数を含める (Q)] チェック・ボックス

[関数 / 割り込み (E)] コンボ・ボックス、[遷移元 (S)] コンボ・ボックス、および、[遷移先 (D)] コンボ・ボックスに“任意”が指定された場合、[関数遷移情報ウィンドウ](#)に表示されていない関数を検索対象とすることを指定するか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“[関数遷移情報ウィンドウ](#)に表示されていない関数を検索対象とする”となります。

(d) [“任意” に RTOS システムを含める (R)] チェック・ボックス

[関数 / 割り込み (E)] コンボ・ボックス、[遷移元 (S)] コンボ・ボックス、および、[遷移先 (D)] コンボ・ボックスに“任意”が指定された場合、RTOS システムを検索対象とすることを指定するか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“RTOS システムを検索対象とする”となります。

(e) [“任意” に不明関数を含める (U)] チェック・ボックス

[関数 / 割り込み (E)] コンボ・ボックス、[遷移元 (S)] コンボ・ボックス、および、[遷移先 (D)] コンボ・ボックスに“任意”が指定された場合、不明関数を検索対象とすることを指定するか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“不明関数を検索対象とする”となります。

注意 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- < << 隠す (E) > ボタンを押下

(4) 機能ボタン

機能ボタンは、以下に示したボタン群から構成されています。

(a) <前を検索 (P)> ボタン

選択状態である関数実行線、または、関数遷移線を基準に、検索条件指定領域で指定された条件の関数実行線、または、関数遷移線を左方向に検索します。

注意 本ボタンは、以下に示した何れかの条件が成立した場合に無効状態となります。

- [関数 / 割り込みを検索する (A)] ラジオ・ボタンがチェック状態、かつ、[関数 / 割り込み (E)] コンボ・ボックスの指定内容が不正
- [関数 / 割り込みの遷移を検索する (I)] ラジオ・ボタンがチェック状態、かつ、[遷移元 (S)] コンボ・ボックス、または、[遷移先 (D)] コンボ・ボックスの指定内容が不正

(b) <次を検索 (N)> ボタン

選択状態である関数実行線、または、関数遷移線を基準に、検索条件指定領域で指定された条件の関数実行線、または、関数遷移線を右方向に検索します。

注意 本ボタンは、以下に示した何れかの条件が成立した場合に無効状態となります。

- [関数 / 割り込みを検索する (A)] ラジオ・ボタンがチェック状態、かつ、[関数 / 割り込み (E)] コンボ・ボックスの指定内容が不正
- [関数 / 割り込みの遷移を検索する (I)] ラジオ・ボタンがチェック状態、かつ、[遷移元 (S)] コンボ・ボックス、または、[遷移先 (D)] コンボ・ボックスの指定内容が不正

(c) <詳細 (E)>> ボタン

拡張検索条件指定領域を表示します。

(d) <<< 隠す (E)> ボタン

拡張検索条件指定領域を非表示にします。

(e) <キャンセル (C)> ボタン

本ダイアログを閉じます。

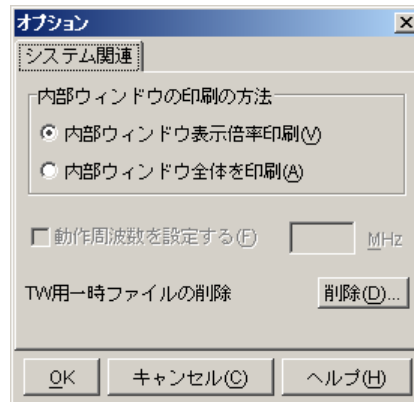
(f) <ヘルプ (H)> ボタン

ヘルプを起動します。

オプション・ダイアログ

メイン・ウィンドウの内部ウィンドウ表示領域に表示されているコールグラフ・ウィンドウ、関数遷移情報ウィンドウの印刷方法、TW850 が性能解析処理 / チューニング処理を実行した際に出力した中間ファイル（一時ファイル）を削除するか否かなどを指定するためのダイアログです。

図 6-30 オプション・ダイアログ



ここでは、次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ダイアログは、次の方法でオープンすることができます。

- [メイン・ウィンドウ](#)において、
 - [ツール (I)] メニュー → [オプション (O)...] を選択したとき

エリア説明

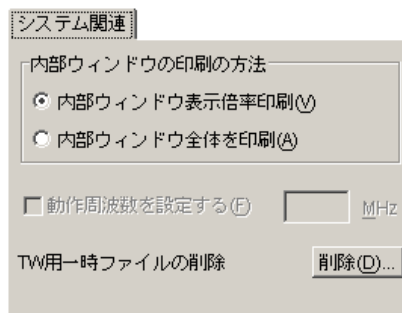
本ダイアログは、次の項目から構成されています。

- [システム関連タブ](#)
- [機能ボタン](#)

(1) システム関連タブ

TW850 の全体処理に関わるオプション設定を行うタブです。

図 6-31 [システム関連] タブ



なお、本タブは、以下に示した指定領域、および、ボタン群から構成されています。

(a) 印刷方法指定領域

- [内部ウィンドウ表示倍率印刷 (V)] ラジオ・ボタン

メイン・ウィンドウの内部ウィンドウ表示領域に表示されている **コールグラフ・ウィンドウ**、**関数遷移情報ウィンドウ**の印刷方法を“表示イメージでの印刷”とするか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、**コールグラフ・ウィンドウ**、**関数遷移情報ウィンドウ**の印刷方法は、“表示イメージでの印刷”となります。

- [内部ウィンドウ全体を印刷 (A)] ラジオ・ボタン

メイン・ウィンドウの内部ウィンドウ表示領域に表示されている **コールグラフ・ウィンドウ**、**関数遷移情報ウィンドウ**の印刷方法を“**コールグラフ・ウィンドウ**、**関数遷移情報ウィンドウ**全体の印刷”とするか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本ラジオ・ボタンがチェック状態の場合、**コールグラフ・ウィンドウ**、**関数遷移情報ウィンドウ**の印刷方法は、“**コールグラフ・ウィンドウ**、**関数遷移情報ウィンドウ**全体の印刷”となります。

(b) [動作周波数を設定する (E)] チェック・ボックス

動作周波数指定領域に対する情報 (実行エンジンの動作周波数) の指定を可能とするか否かを指定します。

なお、本チェック・ボックスがチェック状態の場合、動作周波数指定領域に対する情報 (実行エンジンの動作周波数) の指定が可能となります。

デフォルトでは、非チェック状態となります。

注意 本チェック・ボックスは、以下に示した条件が成立した場合に無効状態となります。

- TW850 が実行エンジン (ディバッガ、シミュレータ) の動作周波数を自動的に検出可能

(c) 動作周波数指定領域

実行エンジン (ディバッガ、シミュレータなど) の動作周波数 (単位: MHz) を指定します。

デフォルトでは、何も指定されません。

注意 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- [動作周波数を設定する (E)] チェック・ボックスが非チェック状態

(d) <削除 (D)...> ボタン

TW850 が性能解析処理 / チューニング処理を実行した際に出力した中間ファイル (一時ファイル) を削除します。

なお、削除対象は、TW850 が出力した中間ファイルのうち、以下に示すファイル以外のすべてのファイルとなります。

- TW850 用プロジェクト・ファイル (オリジナル・プロジェクト・ファイル名).pdb
- チューニング後実行モジュール (オリジナル実行モジュール名)_o.out
- チューニング後実行モジュールのヘキサ・フォーマット・ファイル (オリジナル実行モジュール名)_o.hex (ヘキサ・ファイルの生成を指定している場合)
- チューニング後実行モジュールの Tcl ファイル (オリジナル実行モジュール名)_o.tcl

(2) 機能ボタン

機能ボタンは、以下に示したボタン群から構成されています。

(a) <OK> ボタン

システム関連タブで指定された情報に対応した処理を実行した後、本ダイアログを閉じます。

(b) <キャンセル (C)> ボタン

システム関連タブで指定された情報に対応した処理を実行することなく、本ダイアログを閉じます。

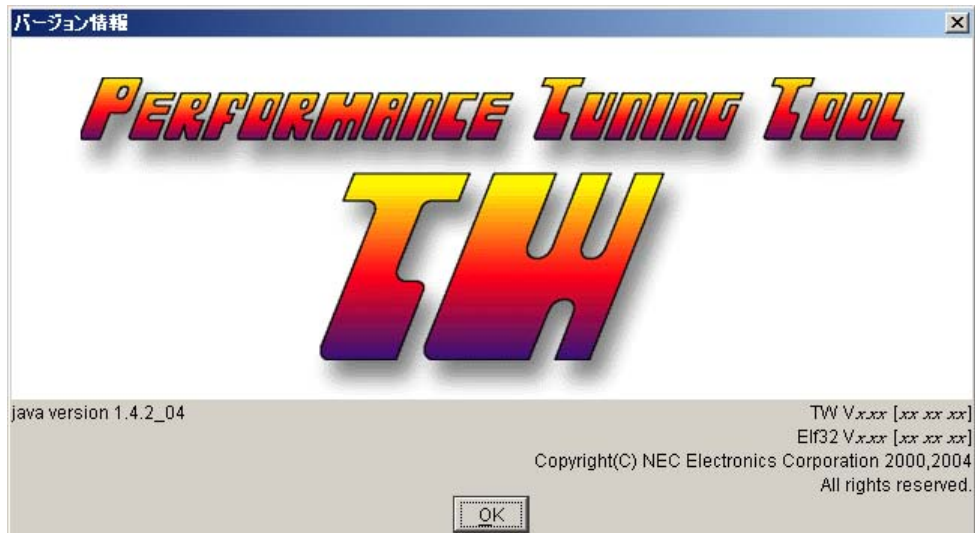
(c) <ヘルプ (H)> ボタン

ヘルプを起動します。

バージョン情報ダイアログ

TW850 のバージョン情報 (バージョン, 著作権など) を表示するためのダイアログです。

図 6-32 バージョン情報ダイアログ



ここでは、次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ダイアログは、次の方法でオープンすることができます。

- [メイン・ウィンドウ](#)において、
 - [ヘルプ (H)] メニュー → [バージョン情報 (A)...] を選択したとき

エリア説明

本ダイアログは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [機能ボタン](#)

(1) メッセージ表示領域

TW850 のバージョン情報 (バージョン, 著作権など) を表示します。

(2) 機能ボタン

機能ボタンは, 以下に示したボタン群から構成されています。

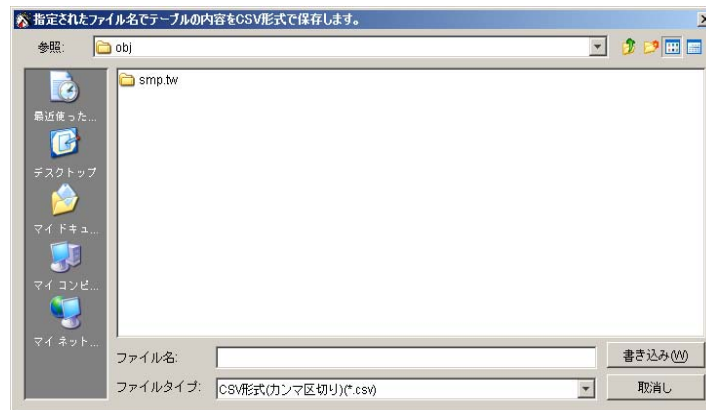
(a) < QK > ボタン

本ダイアログを閉じます。

ファイル選択ダイアログ

TW850 が性能解析処理 / チューニング処理を実行する際に必要となるファイル (タイマ制御コード部など) を指定するためのダイアログです。

図 6-33 ファイル選択ダイアログ



ここでは、次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ダイアログは、次の方法でオープンすることができます。

- [プローブ解析オプション選択パネル](#)において、
 - <参照 (R)... > ボタンを押下したとき
- [メイン・ウィンドウ](#)において、
 - [ファイル (E)] メニュー → [ファイルから入力 (I)...] を選択したとき
- [関数情報テーブル・ウィンドウ](#)、[関数遷移情報ウィンドウ](#)、[実行モジュール情報ウィンドウ](#)、または、[関数詳細情報ウィンドウ](#)にフォーカスを合わせた後、[メイン・ウィンドウ](#)において、
 - [ファイル (E)] メニュー → [ファイルに出力 (E)...] を選択したとき

エリア説明

本ダイアログは、次の項目から構成されています。

- [開く場所指定コンボ・ボックス](#)
- [ファイル名表示領域](#)
- [ファイル名指定領域](#)
- [ファイル・タイプ指定コンボ・ボックス](#)
- [機能ボタン](#)

(1) 開く場所指定コンボ・ボックス

該当ファイル(タイマ制御コード部など)が格納されているフォルダ名を選択します。
フォルダ名が長すぎて表示できない場合、末尾は“...”と表示されます。

(2) ファイル名表示領域

開く場所指定コンボ・ボックス、および、ファイル・タイプ指定コンボ・ボックスで指定された情報をもとに、該当フォルダに格納されているフォルダ、および、ファイルを一覧表示しています。

注意 1 ファイル名をクリックした場合、ファイルは選択状態となり、ファイル名指定領域に表示されます。

注意 2 ファイル名をダブル・クリックした場合、ファイルは選択状態となり、<開く(O)>ボタン、または、<書き込み(W)>ボタンを押下した際と同じ状態となります。

なお、フォルダ名をダブル・クリックした際には、対象フォルダ内の全フォルダ、および、全ファイルがファイル名表示領域内に一覧表示されます。

(3) ファイル名指定領域

TW850 が性能解析処理 / チューニング処理を実行する際に必要となるファイル(タイマ制御コード部など)のファイル名を指定します。

(4) ファイル・タイプ指定コンボ・ボックス

ファイル名表示領域に表示させるファイルのファイル・タイプを選択します。

(5) 機能ボタン

機能ボタンは、以下に示したボタン群から構成されています。

(a) <1 レベル上へ> ボタン



ファイル名表示領域で表示するフォルダを 1 レベル上のフォルダに変更します。

ただし、本ボタンは、ファイル名表示領域においてルート・フォルダを表示している際には、無効状態となります。

(b) <フォルダの新規作成> ボタン



ファイル名表示領域で現在表示しているフォルダに、新しいフォルダを作成します。

(c) <選択(S)>、<開く(O)>、<書き込み(W)> ボタン

TW850 が性能解析処理 / チューニング処理を実行する際に必要となるファイルがファイル名指定領域で指定されたファイルであることを TW850 に通知した後、または、ファイル名指定領域で指定されたファイルを開いた後、あるいは、TW850 の実行状況をファイル名指定領域で指定されたファイルに保存した後、本ダイアログを閉じます。

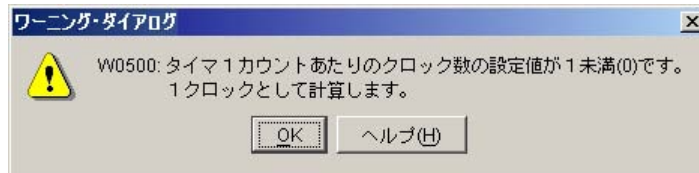
(d) <取消し> ボタン

TW850 が性能解析処理 / チューニング処理を実行する際に必要となるファイルがファイル名指定領域で指定されたファイルであることを TW850 に通知することなく、または、ファイル名指定領域で指定されたファイルを開くことなく、あるいは、TW850 の実行状況をファイル名指定領域で指定されたファイルに保存することなく、本ダイアログを閉じます。

メッセージ・ダイアログ

TW850 が性能解析処理 / チューニング処理を実行中に検出した “ ユーザに通知すべき情報 (処理の経過状況 , 警告事項の発生など) ” を表示するためのダイアログです。

図 6-34 メッセージ・ダイアログ



ここでは、次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ダイアログは、TW850 が性能解析処理 / チューニング処理を実行中にユーザに通知すべき情報 (処理の経過状況 , 警告事項の発生など) を検出した際、自動的にオープンします。

エリア説明

本ダイアログは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [機能ボタン](#)

(1) メッセージ表示領域

TW850 が性能解析処理 / チューニング処理を実行中に検出したユーザに通知すべき情報 (処理の経過状況 , 警告事項の発生など) を表示します。

なお、ユーザに通知すべき情報 (処理の経過状況 , 警告事項の発生など) についての詳細は、「[A.2 情報メッセージ](#)」、および、「[A.3 警告メッセージ](#)」を参照してください。

(2) 機能ボタン

機能ボタンは、以下に示したボタン群から構成されています。

(a) < OK > ボタン

本ダイアログを閉じます。

(b) < ヘルプ (H) > ボタン

ヘルプを起動します。

エラー・ダイアログ

TW850 の起動時や、TW850 が性能解析処理 / チューニング処理を実行中に検出したユーザに通知すべき情報 (致命的エラーの発生など) を表示するためのダイアログです。

図 6-35 エラー・ダイアログ (TW850 起動時)

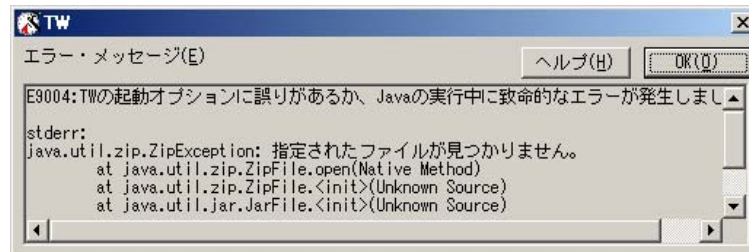
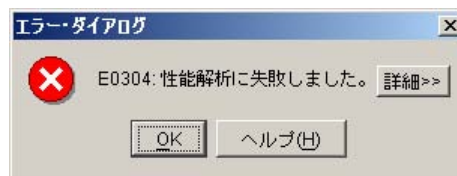


図 6-36 エラー・ダイアログ (TW850 実行中)



ここでは、次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ダイアログは、TW850 の起動時や TW850 が処理を実行中にユーザに通知すべき情報 (致命的エラーの発生など) を検出した際、自動的にオープンします。

エリア説明

本ダイアログは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [機能ボタン](#)

(1) メッセージ表示領域

TW850 の起動時や TW850 が性能解析処理 / チューニング処理を実行中に検出したユーザに通知すべき情報 (致命的エラーの発生など) を表示します。

なお、ユーザに通知すべき情報 (致命的エラーの発生など) についての詳細は、「[A.4 エラー・メッセージ](#)」を参照してください。

(2) 機能ボタン

機能ボタンは、以下に示したボタン群から構成されています。

(a) < 詳細 >> > ボタン

本ダイアログ内に該当情報の詳細内容を表示します。

また、詳細内容のメッセージをダブル・クリックすると、そのメッセージに対応するヘルプを起動します。

(b) < OK > ボタン

本ダイアログを閉じます。

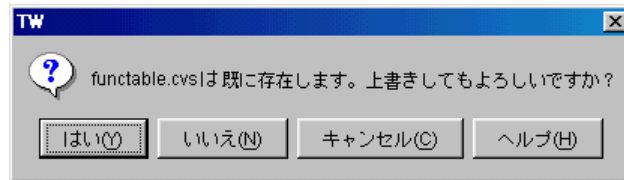
(c) < ヘルプ (H) > ボタン

表示しているメッセージに対応するヘルプを起動します。

確認ダイアログ

TW850 が性能解析処理 / チューニング処理を実行中に検出した問い合わせ情報 (処理を続行するうえでの問題点など) を表示するためのダイアログです。

図 6-37 確認ダイアログ



ここでは、次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ダイアログは、次の方法でオープンすることができます。

- [メイン・ウィンドウ](#)において、
 - [ファイル (E)] メニュー → [終了 (X)...] を選択したとき
 - [編集 (E)] メニュー → [履歴から復帰 (R)...] を選択したとき
 - ツール・バー → 履歴から復帰ボタンを押下したとき
- チューニング・ウィザード ([初期画面パネル](#) , [実行モジュール情報獲得パネル](#) , [実行項目選択パネル](#) など) において、
 - < 閉じる > ボタンを押下したとき
 - < キャンセル (C) > ボタンを押下したとき
- [ファイル選択ダイアログ](#)において、
 - 指定したファイルが既に存在し、< 書き込み (W) > ボタンを押下したとき
- [オプション・ダイアログ](#)の [システム関連] タブにおいて、
 - < 削除 (D)... > ボタンを押下したとき

エリア説明

本ダイアログは、次の項目から構成されています。

- [メッセージ表示領域](#)
- [機能ボタン](#)

(1) メッセージ表示領域

TW850 が性能解析処理 / チューニング処理を実行中に検出した問い合わせ情報 (処理を続行するうえでの問題点など) を表示します。

(2) 機能ボタン

機能ボタンは、以下に示したボタン群から構成されています。

(a) < はい (Y) > ボタン

表示メッセージの内容に対応した処理を実行した後、本ダイアログを閉じます。

(b) < いいえ (N) > ボタン

表示メッセージの内容に対応した処理を実行することなく、本ダイアログを閉じます。

(c) < キャンセル (C) > ボタン

[ファイル選択ダイアログ](#)において指定されたファイルへの保存処理を実行することなく、本ダイアログを閉じます。

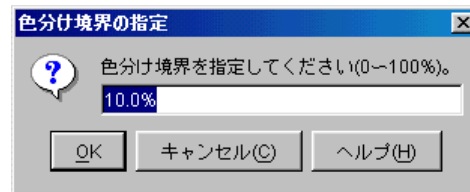
(d) < ヘルプ (H) > ボタン

ヘルプを起動します。

入力ダイアログ

TW850 から要求されている情報 (色分け境界 , 表示倍率 , ルート関数など) を指定するためのダイアログです。

図 6-38 入力ダイアログ



ここでは , 次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ダイアログは , 次の方法でオープンすることができます。

- [コールグラフ・ウィンドウ](#)にフォーカスを合わせた後 , [メイン・ウィンドウ](#)において ,
 - [表示 (V)] メニュー → [コールグラフ表示方法 (C)] → [色分け境界設定 (I)...] を選択したとき
 - [表示 (V)] メニュー → [コールグラフ表示方法 (C)] → [ズーム (Z)...] を選択したとき
 - [表示 (V)] メニュー → [コールグラフ表示方法 (C)] → [ルート関数変更 (R)...] を選択したとき

エリア説明

本ダイアログは , 次の項目から構成されています。

- [メッセージ表示領域](#)
- [要求情報指定領域](#) , [要求情報指定コンボ・ボックス](#)
- [機能ボタン](#)

(1) メッセージ表示領域

本ダイアログに関する解説メッセージを表示しています。

(2) 要求情報指定領域, 要求情報指定コンボ・ボックス

TW850 から要求されている情報 (色分け境界, 表示倍率, ルート関数など) を指定します。

(3) 機能ボタン

機能ボタンは, 以下に示したボタン群から構成されています。

(a) < OK > ボタン

TW850 から要求されている情報が要求情報指定領域, 要求情報指定コンボ・ボックスで指定された情報であることを TW850 に通知した後, 本ダイアログを閉じます。

(b) < キャンセル (C) > ボタン

TW850 から要求されている情報が要求情報指定領域, 要求情報指定コンボ・ボックスで指定された情報であることを TW850 に通知することなく, 本ダイアログを閉じます。

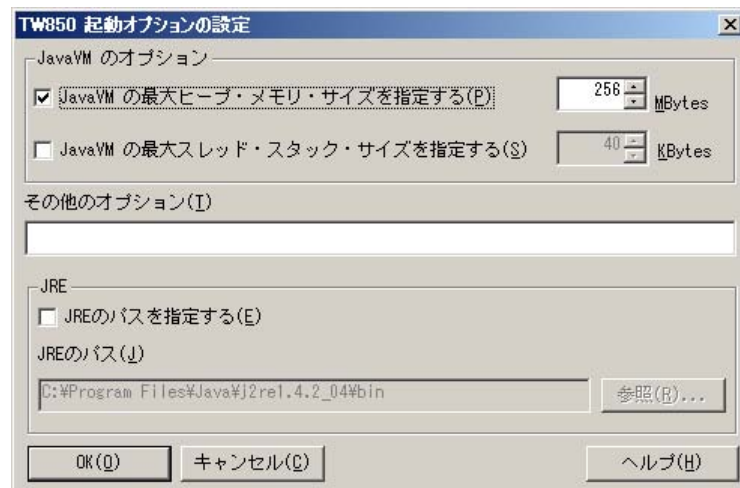
(c) < ヘルプ (H) > ボタン

ヘルプを起動します。

TW850 起動オプション設定ダイアログ

TW850 の起動オプションを指定するためのダイアログです。

図 6-39 TW850 起動オプション設定ダイアログ



ここでは、次の項目について説明しています。

- [オープン方法](#)
- [エリア説明](#)

オープン方法

本ダイアログは、次の方法でオープンすることができます。

- プロジェクト・マネージャ PM plus の[メイン・ウィンドウ](#)において、
 - [ツール (I)] メニュー → [TW850 起動オプションの設定 (W)...] を選択したとき
 - < TW850 起動オプションの設定 > ボタンを押下したとき

エリア説明

本ダイアログは、次の項目から構成されています。

- [JavaVM オプション指定領域](#)
- [\[その他のオプション \(T\) \] 指定領域](#)
- [JRE 指定領域](#)
- [機能ボタン](#)

(1) JavaVM オプション指定領域

JavaVM に関わるオプション設定を行う領域です。

なお、本指定領域は、以下に示したチェック・ボックス、指定領域から構成されています。

(a) [JavaVM の最大ヒープ・メモリ・サイズを指定する (P)] チェック・ボックス

JavaVM がヒープ用メモリ領域として使用可能な最大サイズを指定するか否かを指定します。

デフォルトでは、チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“ヒープ用メモリ領域として使用可能な最大サイズを指定する”となります。

プロジェクトを解析するために必要なメモリが不足している場合、使用するヒープ用メモリ領域を増やすことができます。

(b) 最大ヒープ・メモリ・サイズ指定領域

JavaVM がヒープ用メモリ領域として使用可能な最大サイズ (単位: M バイト) を指定します。

デフォルトでは、256 が指定されます。

注意 1 プロジェクトを解析するために必要なメモリが不足している場合、65M バイト以上を指定してください。目安として、ソース・ファイル数が 500 個、関数数が 5000 個の場合は 128M バイト以上、実装メモリが多い場合は 256M バイト以上を指定することを推奨します。

注意 2 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- [JavaVM の最大ヒープ・メモリ・サイズを指定する (P)] チェック・ボックスが非チェック状態

(c) [JavaVM のスレッド・スタック・サイズを指定する (S)] チェック・ボックス

JavaVM がスレッド・スタック用メモリ領域として使用可能な最大サイズを指定するか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“スレッド・スタック用メモリ領域として使用可能な最大サイズを指定する”となります。

(d) スレッド・スタック・サイズ指定領域

JavaVM がスレッド・スタック用メモリ領域として使用可能な最大サイズ (単位: K バイト) を指定します。

デフォルトでは、40 が指定されます。

注意 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- [JavaVM のスレッド・スタック・サイズを指定する (S)] チェック・ボックスが非チェック状態

(2) [その他のオプション (I)] 指定領域

TW850 のウィザード処理 (性能解析処理、チューニング処理) に関わるオプション設定を行う領域です。

デフォルトでは、何も指定されません。

注意 オプションについての詳細は、「[3.1.2 起動オプション](#)」を参照してください。

(3) JRE 指定領域

JRE (Java 実行環境) のパスを指定する領域です。

なお、本指定領域は、以下に示したチェック・ボックス、指定領域、ボタンから構成されています。

(a) [JRE のパスを指定する (E)] チェック・ボックス

TW850 で使用する JRE のパスを指定するか否かを指定します。

デフォルトでは、非チェック状態となります。

なお、本チェック・ボックスがチェック状態の場合、“TW850 で使用する JRE のパスを指定する”となります。

非チェック状態の場合は、TW850 がサポートするバージョンの JRE のパスが使用されます。

(b) [JRE のパス (J)] 指定領域

TW850 で使用する JRE のフルパスを指定します (例: C:\Program Files\Java\j2re1.4.2_04\bin)。

デフォルトでは、TW850 で対応する JRE がインストールされているパスが指定されます。

TW850 を起動する際、ここで指定したパスが環境変数 PATH の先頭に追加されます。

注意 1 JRE のパスは、以下のレジストリキーの “JavaHome” から取得されます。

HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment\バージョン

注意 2 本指定領域は、以下に示した条件が成立した場合に無効状態となります。

- [JRE のパスを指定する (E)] チェック・ボックスが非チェック状態

(c) <参照 (R)...> ボタン

JRE のパスを選択するためのダイアログをオープンします。

(4) 機能ボタン

機能ボタンは、以下に示したボタン群から構成されています。

(a) <OK(O)> ボタン

JavaVM オプション指定領域、および、他オプション指定領域で指定された情報を PM plus に対して通知した後、本ダイアログを閉じます。

(b) <キャンセル (C)> ボタン

JavaVM オプション指定領域、および、他オプション指定領域で指定された情報を PM plus に対して通知することなく、本ダイアログを閉じます。

(c) <ヘルプ (H)> ボタン

ヘルプを起動します。

付録 A メッセージ

本付録では、TW850 のメイン・ウィンドウのメッセージ表示領域、または、メッセージ・ダイアログ、エラー・ダイアログで表示されるメッセージについて解説しています。

A.1 表示形式

メッセージは、TW850 が処理を実行中に、「ユーザに通知すべき情報（処理の経過状況、警告事項の発生、致命的エラーの発生など）」を検出した際に生成され、メイン・ウィンドウのメッセージ表示領域、または、メッセージ・ダイアログ、エラー・ダイアログに出力されます。

図 A-1 メイン・ウィンドウのメッセージ表示領域

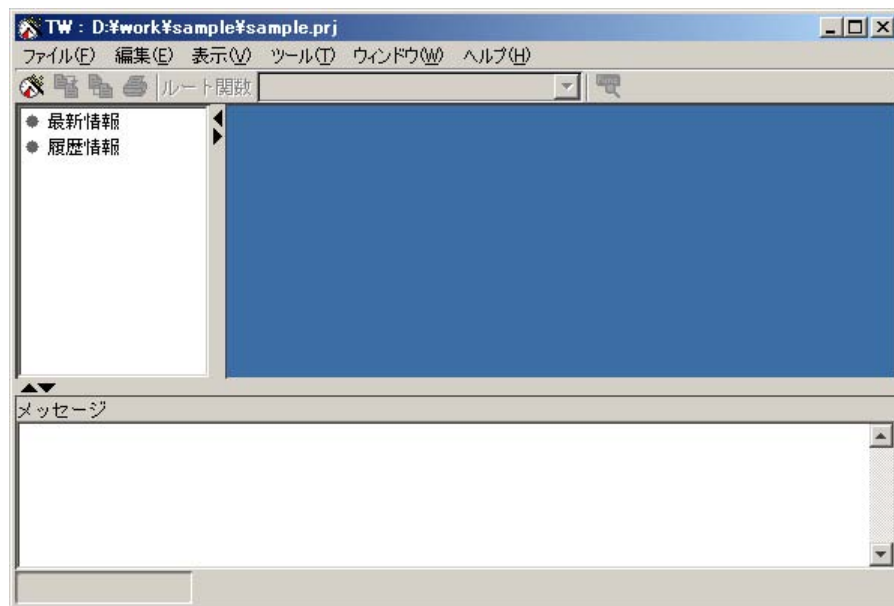


図 A-2 メッセージ・ダイアログ

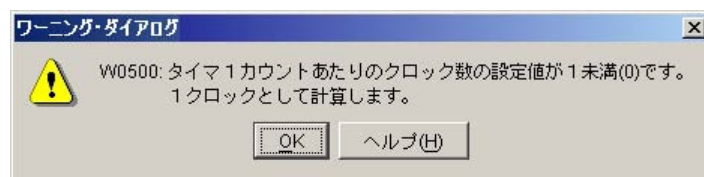
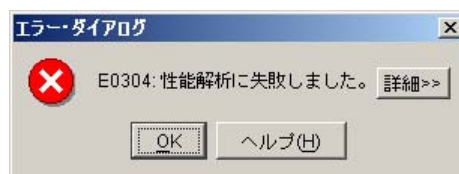


図 A-3 エラー・ダイアログ



なお、TW850 では、メッセージを 3 種類（情報メッセージ、警告メッセージ、エラー・メッセージ）に分けており、メッセージを出力する際には、その先頭に種類を示す英字を付与しています。

I : 情報メッセージ

TW850 では、情報（処理の経過状況など）を検出した際、情報メッセージをメイン・ウィンドウのメッセージ表示領域、または、メッセージ・ダイアログに出力した後、処理を続行します。

W : 警告メッセージ

TW850 では、情報（警告事項の発生など）を検出した際、警告メッセージをメイン・ウィンドウのメッセージ表示領域、または、メッセージ・ダイアログに出力した後、処理を続行します。

E : エラー・メッセージ

TW850 では、情報（致命的エラーの発生など）を検出した際、エラー・メッセージをメイン・ウィンドウのメッセージ表示領域、または、エラー・ダイアログに出力した後、処理を中止します。

A.2 情報メッセージ

以下に、TW850 が処理を実行中に、情報（処理の経過状況など）を検出した際に出力される情報メッセージの一覧（メッセージ番号順）を示します。

なお、情報メッセージ内の %s については、情報（処理の経過状況など）を検出した際に決定されます。

I0200: 指定された条件が見つかりませんでした。

- 1) 最後まで検索しましたが、指定された条件に該当する情報が見つかりませんでした。

I0500: シンボル (%s) を代入できませんでしたが、無視します (%s)。

- 1) シンボル %s の代入処理が異常終了しています。現時点における代入処理の異常終了は TW850 が処理を続行するうえで問題とならないため、TW850 は代入処理の異常終了を無視して、処理を続行します。

I0501: ファイル・シンボル指定に誤りがありますが、無視します (%s)。

- 1) 不正なファイル・シンボル宣言を記述したアセンブリ言語ソース・ファイルの可能性があります。現時点におけるファイル・シンボル宣言の異常検出は TW850 が処理を続行するうえで問題とならないため、TW850 はファイル・シンボル宣言の異常検出を無視して、処理を続行します。

I0502: オプションによるアセンブラ・シンボル (%s) の定義 (%s) に指定に誤りがありますが、無視します (%s)。

- 1) アセンブラ・オプションにおいて、不正なアセンブラ・シンボル宣言 %s を指定した可能性があります。TW850 はアセンブラ・シンボル宣言 %s の設定値に 0 が設定されていたものとして処理を続行します。

I0503: 関数解析中 ... (%s) ソース・ファイル (%s)

- 1) TW850 はソース・ファイル %s 内の性能解析対象関数に対する性能解析処理の経過状況 %s を通知して、処理を続行します。

I0504: 関数解析を終了しました。

- 1) TW850 は全ての性能解析対象関数に対する解析処理が完了した旨を通知して、処理を続行します。

I0505: コールペア解析中 ... (%s) ソース・ファイル (%s)

- 1) TW850 はソース・ファイル %s 内の性能解析対象関数に対するコールペア情報（関数間の呼び出し関係）解析処理の経過状況 %s を通知して、処理を続行します。

I0506: コールペア解析を終了しました。

- 1) TW850 は全ての性能解析対象関数に対するコールペア情報（関数間の呼び出し関係）解析処理が完了した旨を通知して、処理を続行します。

I0507: コールペア解析準備中 ...

- 1) コールペア解析のための前処理中のメッセージです。

I0508: オリジナル解析用実行モジュール情報解析中 ...

1) オリジナル解析用実行モジュール情報解析処理中のメッセージです。

I0509: プロファイル用実行モジュール情報解析中 ...

1) プロファイル用実行モジュール情報解析処理中のメッセージです。

I0510: チューニング後実行モジュール情報解析中 ...

1) チューニング後実行モジュール情報解析処理中のメッセージです。

I0511: 実行モジュール情報解析を終了しました。

1) 実行モジュール情報解析処理の終了メッセージです。

I0800: 古いバージョン (%s) の TW 用プロジェクト・ファイルを読み込みました。

1) 現在使用している TW850 よりも古いバージョンの TW850 で保存された TW850 用プロジェクト・ファイルを読み込んだ場合に表示されるメッセージです。

I1401: トレース・データ解析中 ...

1) トレース・データ解析処理中のメッセージです。

I1402: トレース・データ解析を終了しました。

1) トレース・データ解析処理の終了メッセージです。

I1500: 既存の領域とオーバーラップするため領域 (%s) の先頭アドレスを %s から %s に変更しました。

1) リンク・ディレクティブ・ファイルで指定された領域 (セグメント, または, セクション) %s のサイズではチューニング後実行モジュールのメモリ割り付け処理において, メモリ領域のオーバーラップが発生する可能性があります。TW850 はリンク・ディレクティブ・ファイルで指定された先頭アドレス %s を %s が指定されていたものとして処理を続行します。

I1600: %s

1) TW850 は他ツール (C コンパイラ, アセンブラなど) が出力したメッセージを通知して, 処理を続行します。

I1800: デバイス名を取得できませんでしたが, 無視します。

1) デバイス情報 (デバイス名など) を検出することができませんでした。デバイス情報検出処理の異常終了は TW850 が処理を続行するうえで問題とならないため, TW850 はデバイス情報検出処理の異常終了を無視して, 処理を続行します。

A.3 警告メッセージ

以下に、TW850 が処理を実行中に、情報（警告事項の発生など）を検出した際に出力される警告メッセージの一覧（メッセージ番号順）を示します。

なお、警告メッセージ内の %s については、情報（警告事項の発生など）を検出した際に決定されます。

W0200: 現在の地域とプロジェクト情報保存時の地域が異なるので、文字が正常に表示されない可能性があります。

- 1) 現在とプロジェクト情報保存時では、Windows の “ 地域オプションの設定 ” が異なっている可能性があります。TW850 は現在の設定値のまま、処理を開始します。

W0201: Java 実行環境が TW でサポートしているバージョン (%s) より新しくなっています。

TW でサポートしているバージョンの Java 実行環境を使用することを推奨します。

- 1) TW850 でサポートしているバージョンより新しい Java 実行環境です。サポート外の Java 実行環境となるため、正常な動作を保証できません。TW850 のサポートしている Java 実行環境 (Java2 Runtime Environment Standard Edition V1.4.2_04) を使用してください。

W0400: 指定した RTOS に必要なシンボルが見つかりませんでした。

RTOS(%s) を指定しているとみなして動作します。

- 1) チューニング・ウィザードの初期画面パネルで、誤った RTOS を指定している可能性があります。RTOS 情報が正常に解析できていない可能性があるため、チューニング・ウィザードの初期画面パネルに戻って正しい RTOS を指定し、解析をやり直してください。

W0500: タイマ 1 カウントあたりのクロック数の設定値が 1 未満 (%s) です。

1 クロックとして計算します。

- 1) タイマ制御コード部の “ タイマ 1 カウントあたりの CPU クロック数 tw_clock_count ” に 1 未満の値 %s が設定されている可能性があります。TW850 は “ タイマ 1 カウントあたりの CPU クロック数 ” に 1 が設定されていたものとして処理を開始します。

W0700: TW 用プロジェクト・ファイル (%s) を無視します。

- 1) プロジェクト・ファイル %s の内容が更新 / 破壊されている可能性があります。TW850 は “ 過去の解析情報 ” の読み込み処理は行わず、 “ 過去の解析情報 ” を削除した後、処理を開始します。

W0701: プロジェクト (%s) を復帰できませんでした。

- 1) プロジェクト・ファイル %s の内容が破壊されている可能性があります。TW850 はプロジェクト・ファイル %s の環境復元処理は行わず、デフォルトの環境で処理を開始します。
- 2) プロジェクト・ファイル %s を検出することができませんでした。TW850 はプロジェクト・ファイル %s の環境復元処理は行わず、デフォルトの環境で処理を開始します。

W0800: 新しいバージョン (%s) の TW 用プロジェクト・ファイルを読み込みました。

プロジェクトの初期化を行います。

- 1) 現在使用している TW850 よりも新しいバージョンの TW850 で保存された TW850 用プロジェクト・ファイルを読み込みました。TW850 用プロジェクト・ファイルからの読み込みを中断し、プロジェクトの初期化を行います。

W1000: デバイス・ファイル (%s) 中に不明な割り込み情報が含まれています。

- 1) デバイス・ファイルに TW850 の認識できない情報が含まれています。TW850 はこの情報を無視して動作を進めます。TW850 がサポートしていないバージョンのデバイス・ファイルを使用している可能性があります。ユーザーズ・マニュアルを参照し、デバイス・ファイルのバージョンが、TW850 がサポートしているバージョンであることを確認してください。

W1500: 関数を正確に配置できない可能性のあるセクション (%s) 配置指定があります (%s)。

- 1) リンク・ディレクティブ・ファイル中のセクション配置指定の入力セクションに、TW850 の予約セクション名 ("TW_" から始まるセクション名) が含まれるような表現がないか確認してください。そのような配置指定がある場合、チューニングが正確にできない場合があります。入力セクションは、TW850 の予約セクション名を含まないように記述してください。
- 2) リンク・ディレクティブ・ファイル中のオブジェクト・ファイルの指定に、正規表現を使わない具体的なオブジェクト・ファイル、または、ワイルド・カード ("*") のみを使用した任意のオブジェクト・ファイルを指定しているか確認してください。それら以外の正規表現を用いてオブジェクト・ファイルを指定した場合、正常に実行モジュールを生成できない場合があります。オブジェクト・ファイルには、正規表現を使わない具体的なオブジェクト・ファイル、または、ワイルド・カードのみを使用した任意のオブジェクト・ファイルを指定してください。

A.4 エラー・メッセージ

以下に、TW850 が処理を実行中に、情報（致命的エラーの発生など）を検出した際に出力されるエラー・メッセージの一覧（メッセージ番号順）を示します。

なお、エラー・メッセージ内の %s については、情報（致命的エラーの発生など）を検出した際に決定されます。

E0000: ファイル (%s) が見つかりませんでした。

- 1) ファイル %s を検出することができませんでした。ファイル %s が存在しているか否かを確認してください。
- 2) ファイル %s, または, ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。ファイル %s, および, ファイル %s が格納されているフォルダの属性を確認してください。
- 3) アセンブリ言語ソース・ファイルで指定されたインクルード・ファイル %s を検出することができませんでした。プロジェクト・ファイルにおいて, “インクルード・ファイル %s への検索パス” が設定されているか否かを確認してください。

E0001: ファイルを読み書きできませんでした (%s)。

- 1) ファイルの読み込み処理, または, 書き込み処理が異常終了しています。表示メッセージ %s の内容を確認してください。
- 2) ファイル, または, ファイルが格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。表示メッセージ %s の内容にしたがって, ファイル, および, ファイルが格納されているフォルダの属性を確認してください。
- 3) Windows が不安定な状態となり, ファイルがロックされている可能性があります。表示メッセージ %s の内容にしたがって, Windows を再起動してください。

E0002: ファイル (%s) を開けませんでした。

- 1) ファイル %s, または, ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。ファイル %s, および, ファイル %s が格納されているフォルダの属性を確認してください。

E0003: ファイル (%s) を読み込めませんでした。

- 1) ファイル %s, または, ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。ファイル %s, および, ファイル %s が格納されているフォルダの属性を確認してください。

E0004: ファイル (%s) に書き込めませんでした。

- 1) ファイル %s, または, ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。ファイル %s, および, ファイル %s が格納されているフォルダの属性を確認してください。
- 2) ファイル %s に書き込む際に必要なディスク領域が不足している可能性があります。不要ファイルの削除などを行い, 十分なディスク領域を確保してください。
- 3) Windows が不安定な状態となり, ファイル %s がロックされている可能性があります。Windows を再起動してください。

E0005: ファイル (%s) を削除できませんでした。

- 1) ファイル %s, または, ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。ファイル %s, および, ファイル %s が格納されているフォルダの属性を確認してください。
- 2) Windows が不安定な状態となり, ファイル %s がロックされている可能性があります。Windows を再起動してください。

E0006: 致命的なエラーが発生しました。

- 1) Windows が不安定な状態となり, ファイルがロックされている可能性があります。Windows を再起動してください。
- 2) Java 実行環境が正常にインストールされていない可能性があります。TW850 のサポートしている Java 実行環境 (Java2 Runtime Environment Standard Edition V1.4.2_04) をインストールしてください。
- 3) TW850 の不具合である可能性があります。弊社問い合わせ窓口まで御連絡ください。

E0007: プロジェクト (%s) を読み込めませんでした。

- 1) PM plus プロジェクト・ファイル %s の読み込み処理が異常終了しています。[エラー・ダイアログ](#)内の <詳細>> ボタンを押下し, 発生原因, および, 対処方法を確認してください。

E0100: DLL の読み込みに失敗しました。

- 1) Windows が不安定な状態となり、DLL ファイルがロックされている可能性があります。Windows を再起動してください。
- 2) TW850 が正常にインストールされていない可能性があります。TW850 をアンインストールした後、TW850 を再インストールしてください。

E0103: TW の初期化処理に失敗したので終了します。

- 1) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。
- 2) Java 実行環境が正常にインストールされていない可能性があります。TW850 のサポートしている Java 実行環境 (Java2 Runtime Environment Standard Edition V1.4.2_04) をインストールしてください。
- 3) TW850 が正常にインストールされていない可能性があります。TW850 をアンインストールした後、TW850 を再インストールしてください。

E0105: 致命的なエラーが発生しました。TW を強制終了します。

- 1) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。
- 2) Java 実行環境が正常にインストールされていない可能性があります。TW850 のサポートしている Java 実行環境 (Java2 Runtime Environment Standard Edition V1.4.2_04) をインストールしてください。
- 3) TW850 が正常にインストールされていない可能性があります。TW850 をアンインストールした後、TW850 を再インストールしてください。
- 4) TW850 の不具合である可能性があります。弊社問い合わせ窓口まで御連絡ください。

E0106: メモリが不足しているため TW を終了します。

- 1) ヒープ用メモリ領域、または、スレッド・スタック用メモリ領域が不足している可能性があります。
[TW850 起動オプション設定ダイアログ](#)において、該当メモリ領域のサイズを増やしてください。
- 2) JavaVM が動作する際に必要なメモリ領域が不足している可能性があります。不要アプリケーションの終了などを行い、十分な空きメモリ領域を確保してください。
- 3) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E0107: スタックが不足しているため TW を終了します。

- 1) プロジェクトが必要とするスタック領域が不足している可能性があります。
[TW850 起動オプション設定ダイアログ](#)において、該当スタック領域のサイズを増やしてください。
- 2) JavaVM が動作する際に必要なスレッド・スタック用メモリ領域が不足している可能性があります。不要アプリケーションの終了などを行い、十分な空きメモリ領域を確保してください。
- 3) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E0200: Java 実行環境が TW でサポートしているバージョン (%s) ではありません。

- 1) Java 実行環境が TW850 のサポートしていないバージョンである可能性があります。Java 実行環境が TW850 のサポートしているバージョン %s であるか否か、また、その Java 実行環境を使用しているか否かを確認してください。

E0202: TW 用プロジェクト・ファイル (%s) を保存できませんでした。

- 1) TW850 用プロジェクト・ファイル %s、または、TW850 用プロジェクト・ファイル %s を格納するための TW850 用プロジェクト・データ・フォルダの属性がアクセス制限された状態となっている可能性があります。TW850 用プロジェクト・ファイル %s、および、TW850 用プロジェクト・ファイル %s を格納するための TW850 用プロジェクト・データ・フォルダの属性を確認してください。
- 2) TW850 用プロジェクト・ファイル %s を保存する際に必要なディスク領域が不足している可能性があります。不要ファイルの削除などを行い、十分なディスク領域を確保してください。

E0204: TW 用一時ファイル (%s) を削除できませんでした。

- 1) TW850 用一時ファイル %s、または、TW850 用一時ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。TW850 用一時ファイル %s、および、TW850 用一時ファイル %s が格納されているフォルダの属性を確認してください。

E0206: 印刷できませんでした。

- 1) プリンタ・ドライバが正常に動作していない可能性があります。Windows から正常に印刷することができるか否かを確認してください。

E0207: ヘルプを起動できませんでした。

- 1) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。
- 2) TW850 が正常にインストールされていない可能性があります。TW850 をアンインストールした後、TW850 を再インストールしてください。

E0208: ファイル名が入力されていません。

- 1) ファイル名の入力が行われていない可能性があります。ファイル名の入力が必要条件となっている[ファイル選択ダイアログ](#)において、入力漏れがあるか否かを確認してください。

E0210: メモリが不足しています。

- 1) ヒープ用メモリ領域、または、スレッド・スタック用メモリ領域が不足している可能性があります。[TW850 起動オプション設定ダイアログ](#)において、該当メモリ領域のサイズを増やしてください。
- 2) JavaVM が動作する際に必要なメモリ領域が不足している可能性があります。不要アプリケーションの終了などを行い、十分な空きメモリ領域を確保してください。
- 3) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E0211: スタックが不足しています。

- 1) プロジェクトが必要とするスタック領域が不足している可能性があります。[TW850 起動オプション設定ダイアログ](#)において、該当スタック領域のサイズを増やしてください。
- 2) JavaVM が動作する際に必要なスレッド・スタック用メモリ領域が不足している可能性があります。不要アプリケーションの終了などを行い、十分な空きメモリ領域を確保してください。
- 3) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E0213: 指定したファイル(%s)はWindowsの予約デバイス名です。
別の名前を指定してください。

- 1) ファイル名 %s が Windows で予約されているデバイス名と一致している可能性があります。ファイル名 %s を Windows で予約されているデバイス名 (AUX, COM1, COM2, COM3, COM4, CON, LPT1, LPT2, LPT3, LPT4, PRN, NUL など) とは異なるものに変更してください。

E0214: アクティブ・プロジェクトの取得に失敗しました。

- 1) アクティブ・プロジェクトの設定が行われていない可能性があります。PM plus の [アクティブ・プロジェクトの選択] メニューにおいて、アクティブ・プロジェクトを設定してください。
- 2) PM plus との通信処理が異常終了しています。[エラー・ダイアログ](#)内の <詳細>> ボタンを押下し、発生原因、および、対処方法を確認してください。

E0215: PM plus の状態復帰に失敗しました。

- 1) PM plus, または, ToolLinkManager が起動していないか正常に動作していない可能性があります。PM plus, または, ToolLinkManager が正常に起動しているか確認してください。
- 2) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。
- 3) PM plus, または, ToolLinkManager が正常にインストールされていない可能性があります。PM plus, または, ToolLinkManager をアンインストール後、再インストールしてください。

E0216: 指定したファイル(%s)は関数遷移情報ファイルではありません。

- 1) 関数遷移情報ファイルではないファイルを関数遷移情報ファイルとして指定しています。正しい関数遷移情報ファイルを指定してください。

E0217: 指定されたファイル(%s)はサポート外の形式です。%s以降のTWを使用してください。

- 1) 新しいバージョンのTW850で保存したファイルである可能性があります。このファイルを保存したバージョンと同じか、それより新しいバージョンのTW850で開いてください。

E0218: 指定したファイル (%s) は壊れています。

- 1) ファイルが壊れているか、編集されている可能性があります。正しい関数遷移情報ファイルを指定してください。

E0300: プロファイル・テーブル・サイズを計算できませんでした。

- 1) プロファイル・テーブルのサイズ算出処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し、発生原因、および、対処方法を確認してください。

E0301: オリジナル解析用実行モジュールのビルドまたは情報取得に失敗しました。

- 1) 不正なプロジェクトを指定した可能性があります。正常にオリジナル解析用実行モジュールを生成することが可能なプロジェクトであるか否かを確認してください。
- 2) TW850 がサポートしていないバージョンの C コンパイラ・パッケージを使用している可能性があります。TW850 がサポートするバージョンか否かを確認してください。

E0303: TW の解析可能な関数が 1 つも存在しません。

- 1) 不正なリンク・ディレクティブ・ファイルを指定した可能性があります。TW850 では、TEXT セグメント、または、.text セクションに関数の割り付けが行われていないリンク・ディレクティブ・ファイルを性能解析対象 / チューニング対象とすることはできません。TW850 が解析できるよう、定められたセグメント、または、セクションに性能解析対象 / チューニング対象とする関数が存在するソース・ファイルを指定してください。または、性能解析対象 / チューニング対象とする関数を定められたセグメント / セクションにマッピングするリンク・ディレクティブを作成してください。
- 2) 不正なフォーマットの C 言語ソース・ファイル、または、アセンブリ言語ソース・ファイルを指定した可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能なフォーマットに C 言語ソース・ファイル、または、アセンブリ言語ソース・ファイルの内容を修正してください。

E0304: 性能解析に失敗しました。

- 1) 不正なプロジェクトを指定した可能性があります。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し、発生原因、および、対処方法を確認してください。

E0305: チューニングに失敗しました。

- 1) 不正なプロジェクトを指定した可能性があります。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し、発生原因、および、対処方法を確認してください。

E0306: チューニング・ウィザードを起動できませんでした。

- 1) 不正なプロジェクトを指定した可能性があります。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し、発生原因、および、対処方法を確認してください。

E0308: チューニング結果の復帰ができませんでした。

- 1) TW850 用プロジェクト・ファイルを格納するための TW850 用プロジェクト・データ・フォルダの属性がアクセス制限された状態となっている可能性があります。TW850 用プロジェクト・ファイルを格納するための TW850 用プロジェクト・データ・フォルダの属性を確認してください。
- 2) チューニング結果を復帰する際に必要なディスク領域が不足している可能性があります。不要ファイルの削除などを行い、十分なディスク領域を確保してください。

E0309: 一時ファイルを %s に保存できませんでした。

- 1) 一時ファイル、または、一時ファイルを格納するためのフォルダ %s の属性がアクセス制限された状態となっている可能性があります。一時ファイル、および、一時ファイルを格納するためのフォルダの属性を確認してください。
- 2) 一時ファイルを保存する際に必要なディスク領域が不足している可能性があります。不要ファイルの削除などを行い、十分なディスク領域を確保してください。

E0310: 一時ファイル (%s) の復帰ができませんでした。

- 1) 一時ファイル %s、または、一時ファイル %s を格納するためのフォルダの属性がアクセス制限された状態となっている可能性があります。一時ファイル %s、および、一時ファイル %s を格納するためのフォルダの属性を確認してください。

E0311: 一時ファイル (%s) の削除ができませんでした。

- 1) 一時ファイル %s, または, 一時ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。一時ファイル %s, および, 一時ファイル %s が格納されているフォルダの属性を確認してください。

E0314: 起動中のディバग्ガ (%s) はプロジェクトで指定されたものではありません。

- 1) プロジェクトで未定義のディバग्ガ %s(プロジェクトで ID850 を指定している場合の SM850 等) が起動している可能性があります。TW850 では, 未定義のディバग्ガが起動している場合, 正常な処理を実行することはできません。

E0400: オリジナル解析用実行モジュールのビルドに失敗しました。

- 1) オリジナル解析用実行モジュールの生成処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し, 発生原因, および, 対処方法を確認してください。
- 2) TW850 がサポートしていないバージョンの C コンパイラ・パッケージを使用している可能性があります。TW850 がサポートするバージョンか否かを確認してください。

E0401: プロファイル用実行モジュールのビルドに失敗しました。

- 1) プロファイル用実行モジュールの生成処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し, 発生原因, および, 対処方法を確認してください。
- 2) プローブ・コードを挿入, または, プロファイル・テーブルを保存する際に必要なメモリ領域が不足している可能性があります。リンク・ディレクティブ・ファイルの指定変更などを行い, 十分な空きメモリ領域を確保してください。
- 3) TW850 がサポートしていないバージョンの C コンパイラ・パッケージを使用している可能性があります。TW850 がサポートするバージョンか否かを確認してください。

E0402: チューニング後実行モジュールのビルドに失敗しました。

- 1) チューニング後実行モジュールの生成処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し, 発生原因, および, 対処方法を確認してください。
- 2) チューニング対象関数を再配置する際に必要なメモリ領域が不足している可能性があります。リンク・ディレクティブ・ファイルの指定変更などを行い, 十分な空きメモリ領域を確保してください。
- 3) TW850 がサポートしていないバージョンの C コンパイラ・パッケージを使用している可能性があります。TW850 がサポートするバージョンか否かを確認してください。

E0403: アセンブリ・ソース・ファイルの生成に失敗しました。

- 1) アセンブリ言語ソース・ファイルの生成処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し, 発生原因, および, 対処方法を確認してください。
- 2) TW850 がサポートしていないバージョンの C コンパイラ・パッケージを使用している可能性があります。TW850 がサポートするバージョンか否かを確認してください。

E0404: コールベア情報の解析に失敗しました。

- 1) コールベア情報の解析処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し, 発生原因, および, 対処方法を確認してください。

E0405: オリジナル解析用実行モジュールの実行に失敗しました。

- 1) オリジナル解析用実行モジュールの実行処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し, 発生原因, および, 対処方法を確認してください。

E0406: プロファイル用実行モジュールの実行に失敗しました。

- 1) プロファイル用実行モジュールの実行処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し, 発生原因, および, 対処方法を確認してください。
- 2) 不正なプロファイル用実行モジュールが出力された可能性があります。チューニング後実行モジュールを生成する際にメッセージ (警告メッセージなど) が表示されているか否かを確認してください。
- 3) 不正なフォーマットのアセンブリ言語ソース・ファイルを指定した可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能なフォーマットにアセンブリ言語ソース・ファイルの内容を修正してください。

E0407: チューニング後実行モジュールの実行に失敗しました。

- 1) チューニング後実行モジュールの実行処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し、発生原因、および、対処方法を確認してください。
- 2) 不正なチューニング後実行モジュールが出力された可能性があります。チューニング後実行モジュールを生成する際にメッセージ(警告メッセージなど)が表示されているか否かを確認してください。
- 3) 不正なフォーマットのアセンブリ言語ソース・ファイルを指定した可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能なフォーマットにアセンブリ言語ソース・ファイルの内容を修正してください。

E0408: 関数情報の解析に失敗しました。

- 1) 関数情報の解析処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し、発生原因、および、対処方法を確認してください。

E0409: チューニングに失敗しました。

- 1) 実行モジュールのチューニング処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し、発生原因、および、対処方法を確認してください。

E0411: アセンブリ・ソース・ファイルの解析に失敗しました。

- 1) アセンブリ・ソース・ファイルの解析に失敗しました。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し、発生原因、および、対処方法を確認してください。

E0412: オリジナル解析用実行モジュールの解析に失敗しました。

- 1) オリジナル解析用実行モジュールの解析に失敗しました。[エラー・ダイアログ](#)内の<詳細>> ボタンを押下し、発生原因、および、対処方法を確認してください。

E0500: 解析可能な関数の最大数(%s)を超えました。

- 1) 性能解析対象/チューニング対象となる関数が最大数 %s を超えている可能性があります。性能解析対象/チューニング対象とする必要性が少ない関数を TEXT セグメント、または、.text セクションから他のセグメント、または、セクションに割り付けるなどの処置を行うことにより、性能解析対象/チューニング対象となる関数の総数を %s 以下に削減してください。

E0501: システム・ライブラリ情報ファイル(%s)のフォーマットが不正です。

- 1) システム・ライブラリ情報ファイル %s の内容が破壊されている可能性があります。TW850 をアンインストールした後、TW850 を再インストールするなどの処置を行うことにより、システム・ライブラリ情報ファイルをオリジナルの状態に復元してください。
- 2) システム・ライブラリ情報ファイル %s が TW850 のサポートしていないバージョンである可能性があります。システム・ライブラリ情報ファイル %s が TW850 のサポートしているバージョンであるか否かを確認してください。

E0502: システム・ライブラリ情報ファイル(%s)に不整合なデータが存在します。

- 1) システム・ライブラリ情報ファイル %s の内容が破壊されている可能性があります。TW850 をアンインストールした後、TW850 を再インストールするなどの処置を行うことにより、システム・ライブラリ情報ファイルをオリジナルの状態に復元してください。

E0503: アセンブラの構文エラーまたは算術エラーです(%s)。

- 1) 不正な構文/評価式を記述したアセンブリ言語ソース・ファイルの可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能な構文/評価式にアセンブリ言語ソース・ファイルの該当箇所 %s を修正してください。

E0504: 条件アセンブル終了疑似命令が余分にあります(%s)。

- 1) 条件アセンブル終了疑似命令に対応した条件アセンブル開始疑似命令が記述されていないアセンブリ言語ソース・ファイルの可能性があります。アセンブリ言語ソース・ファイルの該当箇所 %s に対応した条件アセンブル開始疑似命令の記述追加、または、該当箇所の記述削除を行ってください。

E0505: システム予約マクロ名 (%s) を使用しています。

- 1) マクロ名 %s が TW850 で予約されているシステム予約マクロ名と一致している可能性があります。マクロ名 %s を TW850 で予約されているシステム予約マクロ名とは異なるものに変更してください。

E0506: 評価式中に未解決シンボルがあるため評価できません (%s)。

- 1) シンボル情報の静的解析処理が異常終了しています。アセンブリ言語ソース・ファイル %s の評価式中に含まれている該当シンボルの定義を行ってください。

E0507: 実行モジュールが読み込まれていません。

- 1) 実行モジュール, または, 実行モジュールが格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。実行モジュール, および, 実行モジュールが格納されているフォルダの属性を確認してください。

E0508: プロファイル・メモリ領域から十分なデータを取得できません。

- 1) プロファイル・メモリ領域として必要なメモリ領域が不足している可能性があります。リンク・ディレクティブ・ファイルにおいて, プロファイル・メモリ領域が割り付けられるセグメント, または, セクションのサイズを増やしてください。
- 2) TW850 が生成したファイルに対する操作 (移動, 削除, 属性変更など) が行われたために, プロファイル・メモリ領域のデータを実行エンジン (ディバッガ, シミュレータなど) から読み込むことができない環境となっている可能性があります。性能解析処理 / チューニング処理を再実行するなどの処置を行うことにより, TW850 が生成したファイルをオリジナルの状態に復元してください。

E0509: シンボル (%s) が実行モジュールに見つかりません。

- 1) TW850 が生成した実行モジュールに対する操作 (移動, 削除, 属性変更など) が行われたために, シンボル %s の検出処理が異常終了しています。性能解析処理 / チューニング処理を再実行するなどの処置を行うことにより, TW850 が生成した実行モジュールをオリジナルの状態に復元してください。
- 2) 実行モジュールに該当関数シンボルと同名の非関数シンボルが存在します。リンク・ディレクティブで実行モジュールにシンボルを埋め込むことができる場合, リンク・ディレクティブで実行モジュールに埋め込むシンボル名は関数名と同一にならないようにしてください。ただし, プログラム中のシンボルと同名のシンボルをリンク・ディレクティブで宣言するための表記がある場合 (例えば, GCC のリンカ・スクリプト・コマンド PROVIDE など) は, その表記方法で記述する方法でも, エラーを回避することができます。

E0510: ラベル (%s) が二重定義されています (%s)。

- 1) 同一ラベル名 %s が複数定義されているアセンブリ言語ソース・ファイルの可能性があり。アセンブリ言語ソース・ファイル %s の該当ラベルを修正してください。

E0511: マクロ定義の終了疑似命令または繰り返しの終了疑似命令が多すぎます (%s)。

- 1) マクロ定義 / 繰り返し終了疑似命令に対応したマクロ定義 / 繰り返し開始疑似命令が記述されていないアセンブリ言語ソース・ファイルの可能性があり。アセンブリ言語ソース・ファイルの該当箇所 %s に対応したマクロ定義 / 繰り返し開始疑似命令の記述追加, または, 該当箇所の記述削除を行ってください。

E0512: マクロのネスト展開回数が限界値 (%s) を超えました (%s)。

- 1) マクロのネスト展開回数が限界値 %s を超えている可能性があります。アセンブリ言語ソース・ファイル %s のマクロのネスト展開回数が限界値 %s 以下となるように修正してください。

E0513: マクロ定義の終了命令と繰り返しの終了命令とで不整合があります (%s)。

- 1) マクロ定義 / 繰り返し開始疑似命令とマクロ定義 / 繰り返し終了疑似命令が不整合のアセンブリ言語ソース・ファイルの可能性があり。アセンブリ言語ソース・ファイル %s のマクロ定義 / 繰り返し開始疑似命令とマクロ定義 / 繰り返し終了疑似命令が整合性のとれた状態となるように修正してください。

E0514: 使用しているマクロ名がアセンブラ命令ニーモニック (%s) と一致しています。

- 1) アセンブラ命令ニーモニックと一致するマクロ名を使用することはできません。アセンブラ命令ニーモニックと異なるマクロ名に変更するか, そのマクロ名を使用しているソース・ファイルを解析対象外としてください。

E0600: プロファイル情報に不正なデータを発見したためチューニングを中断しました。

- 1) チューニング後実行モジュールを生成する際に必要なプロファイル情報の内容が破壊されている可能性があります。オリジナル解析用実行モジュールの性能解析処理が正常に行われているか否かを確認してください。

E0700: チューニング結果を保存できませんでした (%s)。

- 1) チューニング後実行モジュール,または,チューニング後実行モジュールを格納するためのフォルダの属性がアクセス制限された状態となっている可能性があります。表示メッセージ %s の内容にしたがって,チューニング後実行モジュール,および,チューニング後実行モジュールを格納するためのフォルダの属性を確認してください。

E0701: チューニング結果を復帰できませんでした (%s)。

- 1) チューニング後実行モジュール,または,チューニング後実行モジュールを格納するためのフォルダの属性がアクセス制限された状態となっている可能性があります。表示メッセージ %s の内容にしたがって,チューニング後実行モジュール,および,チューニング後実行モジュールを格納するためのフォルダの属性を確認してください。

E0705: TW 用プロジェクト・バックアップ・ファイル (%s) の生成に失敗しました。

- 1) TW850 用プロジェクト・ファイル,または,その格納フォルダのアクセス許可,TW850 用プロジェクト・バックアップ・ファイルの書き込み許可がない可能性があります。プロジェクト・ファイル,または,その格納フォルダ,および,プロジェクト・バックアップ・ファイルの属性を確認してください。
- 2) プロジェクト・ファイルを保存する際に必要なディスク領域が不足している可能性があります。十分なディスク領域を確保してからもう一度操作を実行してください。

E0706: TW 用プロジェクト・バックアップ・ファイル (%s) からの復元に失敗しました。

- 1) TW850 用プロジェクト・バックアップ・ファイルのアクセス許可がない可能性があります。プロジェクト・バックアップ・ファイルの属性を確認してください。TW850 用プロジェクト・バックアップ・ファイルからの復元に失敗した場合,TW850 用プロジェクト・ファイルから情報を復元します。
- 2) TW850 用プロジェクト・バックアップ・ファイルが破壊されたか,互換性のないファイルを読み込んだ可能性があります。ファイルを削除してください。TW850 用プロジェクト・バックアップ・ファイルからの復元に失敗した場合,TW850 用プロジェクト・ファイルから情報を復元します。

E0802: オプション (%s) はサポートしていません。

- 1) TW850 のサポートしていないオプション %s が指定されている可能性があります。オプション %s の指定をプロジェクトから削除してください。

E0803: プロジェクト・ファイル用フォルダ (%s) にアクセスできませんでした。

- 1) プロジェクト・ファイルが格納されているフォルダ %s の属性がアクセス制限された状態となっている可能性があります。プロジェクト・ファイルが格納されているフォルダ %s の属性を確認してください。

E0804: TW 用プロジェクト・ファイル (%s) を読み込めませんでした。

- 1) TW850 用プロジェクト・ファイル %s,または,TW850 用プロジェクト・ファイル %s が格納されている TW850 用プロジェクト・データ・フォルダの属性がアクセス制限された状態となっている可能性があります。TW850 用プロジェクト・ファイル %s,および,プロジェクト・ファイル %s が格納されている TW850 用プロジェクト・データ・フォルダの属性を確認してください。
- 2) TW850 用プロジェクト・ファイル %s の内容が破壊されている可能性があります。正常に読み込むことが可能なプロジェクト・ファイルであるか否かを確認してください。

E0805: TW 用プロジェクト・ファイル (%s) に書き込めませんでした。

- 1) TW850 用プロジェクト・ファイル %s,または,TW850 用プロジェクト・ファイル %s が格納されている TW850 用プロジェクト・データ・フォルダの属性がアクセス制限された状態となっている可能性があります。TW850 用プロジェクト・ファイル %s,および,プロジェクト・ファイル %s が格納されている TW850 用プロジェクト・データ・フォルダの属性を確認してください。

E0806: TW 用プロジェクト・データ・フォルダ (%s) を作成できませんでした。

- 1) TW850 用プロジェクト・データ・フォルダを格納するためのフォルダの属性がアクセス制限された状態となっている可能性があります。TW850 用プロジェクト・データ・フォルダを格納するためのフォルダの属性を確認してください。
- 2) 既存フォルダ名が TW850 用プロジェクト・データ・フォルダ名 %s と一致している可能性があります。既存フォルダ名を TW850 用プロジェクト・データ・フォルダ名 %s とは異なるものに変更してください。

E0807: TW 用一時ファイル (%s) を削除できませんでした。

- 1) TW850 用一時ファイル %s, または, TW850 用一時ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。TW850 用一時ファイル %s, および, TW850 用一時ファイル %s が格納されているフォルダの属性を確認してください。

E0808: プロジェクト・ファイル (%s) が不正です。

- 1) プロジェクト・ファイル %s の内容が破壊されている可能性があります。正常に実行モジュールを生成することが可能なプロジェクト・ファイルであるか否かを確認してください。

E0810: リンク・ディレクティブ・ファイルが指定されていません。

- 1) 不正なプロジェクト・ファイルを指定した可能性があります。TW850 では, リンク・ディレクティブ・ファイルの指定が行われていないプロジェクト・ファイルを性能解析対象 / チューニング対象とすることはできません。

E0811: 実行エンジンが不明です。

- 1) 実行エンジン (ディバッガ, シミュレータなど) を検出することができませんでした。PM plus の [ディバッガの設定] メニューにおいて, TW850 のサポートしている実行エンジンが設定されているか否かを確認してください。

E0812: 環境変数 (%s) が設定されていません。

- 1) 環境変数 %s が記述されていない可能性があります。環境変数設定ファイル (autoexec.bat など) において, ホスト・マシンの環境変数として %s が設定されているか否かを確認してください。

E0816: アクティブ・プロジェクトは TW のサポート対象外です。
%s

- 1) 不正なプロジェクト (アクティブ・プロジェクト) を指定した可能性があります。TW850 では, C 言語ソース・ファイル, および, アセンブリ言語ソース・ファイルの指定が行われていないプロジェクトを性能解析対象 / チューニング対象とすることはできません。
- 2) アクティブ・プロジェクトが TW850 のサポートしていない品種である可能性があります。アクティブ・プロジェクトが TW850 のサポートしている品種であるか否かを確認してください。

E0817: TW で必要な情報 (%s) がプロジェクト・ファイル (%s) 中に存在しませんでした。

- 1) TW850 が性能解析処理 / チューニング処理を実行するうえで必要な情報 %s をプロジェクト・ファイル %s から検出することができませんでした。PM plus において, TW850 が性能解析処理 / チューニング処理を実行するうえで必要な情報 %s を設定してください。

E0818: ディバッガに関するプロジェクト情報が存在しません。

- 1) ディバッガに関する情報をプロジェクト・ファイルから検出することができませんでした。ディバッガにおいて, プロジェクト・ファイルに対するディバグ環境の保存処理を実行してください。

E0819: ディバッガが指定されていません。

- 1) ディバッガの設定が行われていない可能性があります。PM plus の [ディバッガの設定] メニューにおいて, ディバッガを設定してください。

E0820: ソース・ファイルがひとつも登録されていません。

- 1) TW850 が性能解析処理 / チューニング処理を実行するうえで必要な情報 (ソース・ファイル名) をプロジェクト・ファイルから検出することができませんでした。PM plus において, TW850 が性能解析処理 / チューニング処理を実行するうえで必要なソース・ファイル名を設定してください。

E0821: デバッグ・ターゲット・ファイルが指定されていません。

- 1) TW850 が性能解析処理 / チューニング処理を実行するうえで必要な情報 (デバッグ・ターゲット・ファイル名) をプロジェクト・ファイルから検出することができませんでした。PM plus において、リンカ・オプションで出力ファイル名を設定したのち、ワーク・スペースの保存処理を実行してください。

E0824: コンパイラのデフォルト・インクルード・パスを取得できませんでした。

- 1) 他ツール (PM plus / CA850) が正常にインストールされていない可能性があります。他ツールをアンインストールしたのち、再インストールしてください。
- 2) Windows が不安定になっている可能性があります。Windows を再起動してください。

E0900: 実行モジュール (%s) はサポート外の形式です。

- 1) 不正なフォーマットの実行モジュール %s を指定した可能性があります。実行モジュール %s が TW850 のサポートしている C コンパイラ・パッケージによって生成されたものであるか否かを確認してください。

E0901: 行番号 (%s) をアドレス値に変換できませんでした。

- 1) 不正な行番号 %s を指定した可能性があります。TW850 では、存在しない行番号、アドレスに変換することができない関数外の行番号を性能解析対象 / チューニング対象とすることはできません。

E0902: アドレス情報 (%s) をアドレス値に変換できませんでした。

- 1) 不正なシンボル %s を指定した可能性があります。TW850 では、存在しないシンボルを性能解析対象 / チューニング対象とすることはできません。

E1001: デバイス・ファイル (%s) が不正です。

- 1) 不正なフォーマットのデバイス・ファイル %s を指定した可能性があります。正常に実行モジュールを生成することが可能なデバイス・ファイルであるか否かを確認してください。
- 2) デバイス・ファイルが正常にインストールされていない可能性があります。デバイス・ファイルをアンインストールした後、デバイス・ファイルを再インストールしてください。

E1100: 対になる文字列定数記号 (%s) がありません。

- 1) 文字列定数を宣言するための終了記号条件が記述されていないアセンブリ言語ソース・ファイルの可能性があります。アセンブリ言語ソース・ファイルの該当箇所 %s に対応した終了記号の記述追加、または、該当箇所の記述削除を行ってください。

E1101: 文字列に変換できないオペランドです。

- 1) 文字列定数を宣言するオペランドにおいて、不正な文字列定数を記述したアセンブリ言語ソース・ファイルの可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能な “ 文字列定数の宣言 ” にアセンブリ言語ソース・ファイルの該当箇所を修正してください。

E1102: 式が評価できません。

- 1) 不正な評価式を記述したアセンブリ言語ソース・ファイルの可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能な評価式にアセンブリ言語ソース・ファイルの該当箇所を修正してください。

E1103: マクロ定義にマクロ名の宣言がありません。

- 1) マクロ定義において、マクロ名が記述されていないアセンブリ言語ソース・ファイルの可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能なマクロ名にアセンブリ言語ソース・ファイルの該当箇所を修正してください。

E1104: マクロ使用で引数の数に不整合があります。

- 1) マクロを定義した際の引数の数とマクロを使用した際の引数の数が異なっているアセンブリ言語ソース・ファイルの可能性があります。引数の数が同じになるようにアセンブリ言語ソース・ファイルの該当箇所を修正してください。

E1105: ファイル・シンボル指定が不正です。

- 1) 不正なファイル・シンボル宣言を記述したアセンブリ言語ソース・ファイルの可能性があります。アセンブリ言語ソース・ファイルのファイル・シンボル宣言が正常に行われているか否かを確認してください。

E1200: 対になるカッコがありません。

- 1) 評価式中の括弧が不整合のアセンブリ言語ソース・ファイルの可能性があります。アセンブリ言語ソース・ファイルの評価式中に含まれている括弧が整合性のとれた状態となるように修正してください。

E1201: 評価する式がありません。

- 1) 不正な評価式を記述したアセンブリ言語ソース・ファイルの可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能な評価式にアセンブリ言語ソース・ファイルの該当箇所を修正してください。

E1202: 評価できない数値表現 (%s) があります。

- 1) 不正な数値表現 %s が含まれている評価式を記述したアセンブリ言語ソース・ファイルの可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能な数値表現にアセンブリ言語ソース・ファイルの該当箇所を修正してください。

E1203: 評価できないシンボル (%s) があります。

- 1) 不正なシンボル %s が含まれている評価式を記述したアセンブリ言語ソース・ファイルの可能性があります。TW850 では、値の定義が行われていないシンボルを評価することはできません。
- 2) ラベル %s が含まれている評価式を記述したアセンブリ言語ソース・ファイルの可能性があります。TW850 では、アセンブリ言語ソース・ファイルの評価式中に記述されたラベルを評価することはできません。

E1204: 演算と数値の数に不整合があります。

- 1) 不正な項数の演算子が含まれている評価式を記述したアセンブリ言語ソース・ファイルの可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能な評価式にアセンブリ言語ソース・ファイルの該当箇所を修正してください。

E1205: 演算子 (%s) の使い方に誤りがあります。

- 1) 不正な演算子が含まれている評価式を記述したアセンブリ言語ソース・ファイルの可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能な評価式にアセンブリ言語ソース・ファイルの該当箇所を修正してください。

E1206: 評価式において 0 で除算しています。

- 1) 不正な評価式 (0 除算) を記述したアセンブリ言語ソース・ファイルの可能性があります。TW850 のサポートしている C コンパイラ・パッケージが対応可能な評価式にアセンブリ言語ソース・ファイルの該当箇所を修正してください。

E1300: 実行モジュールのビルドに失敗しました (%s)。

- 1) オリジナル解析用実行モジュールの生成処理が異常終了しています。表示メッセージ %s の内容にしたがって、ビルド・ツールをオリジナルのプロジェクト・ファイルから単独起動し、オリジナル解析用実行モジュールの生成処理が正常に行われているか否かを確認してください。
- 2) プロファイル用実行モジュールの生成処理が異常終了しています。表示メッセージ %s の内容にしたがって、ビルド・ツールをプロファイル用実行モジュールのプロジェクト・ファイルから単独起動し、プロファイル用実行モジュールの生成処理が正常に行われているか否かを確認してください。
- 3) チューニング後実行モジュールの生成処理が異常終了しています。表示メッセージ %s の内容にしたがって、ビルド・ツールをチューニング後実行モジュールのプロジェクト・ファイルから単独起動し、チューニング後実行モジュールの生成処理が正常に行われているか否かを確認してください。
- 4) 不正なフォーマットのリンク・ディレクティブ・ファイルを指定した可能性があります。表示メッセージ %s の内容にしたがって、正常に実行モジュールを生成することが可能なリンク・ディレクティブ・ファイルであるか否かを確認してください。

E1302: ビルド時に生成する一時ファイルを削除できませんでした (%s)。

- 1) 一時ファイル,または,一時ファイルが格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。表示メッセージ %s の内容にしたがって,一時ファイル,および,一時ファイルが格納されているフォルダの属性を確認してください。
- 2) 一時ファイルを他のアプリケーション(エディタなど)から参照している可能性があります。表示メッセージ %s の内容にしたがって,一時ファイルを他のアプリケーションから参照しているか否かを確認してください。

E1303: 実行モジュールに TW が正常に動作しない可能性のあるシンボルがあります。

- 1) 不正な実行モジュールを指定した可能性があります。TW850 では,システム・パフォーマンス・アナライザ AZ850 用モニタ・プログラムが含まれている実行モジュールを性能解析対象/チューニング対象とすることはできません。

E1400: 実行モジュール (%s) の実行に失敗しました。

- 1) 実行モジュール %s の実行処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>>ボタンを押下し,発生原因,および,対処方法と共に「[4.6.3 アセンブリ言語ソース・ファイルの記述形式](#)」を確認してください。
- 2) 不正なプロジェクトを指定した可能性があります。ディバッガをオリジナルのプロジェクト・ファイルから単独起動し,指定範囲の実行処理が正常に行われているか否かを確認してください。
- 3) 再配置関数,プロファイル・ルーチン,または,プロファイル・テーブルの割り付けが正常に割り付けられていない可能性があります。再配置関数,プロファイル・ルーチン,および,プロファイル・テーブルが[配置先選択パネル](#)で指定したメモリ領域に割り付けられているか否かを確認してください。

E1401: 実行モジュール (%s) の実行停止に失敗しました。

- 1) 実行モジュール %s の実行停止処理が異常終了しています。[エラー・ダイアログ](#)内の<詳細>>ボタンを押下し,発生原因,および,対処方法を確認してください。

E1403: プロファイル情報の解析に失敗しました。

- 1) TW850 の不具合である可能性があります。弊社問い合わせ窓口まで御連絡ください。

E1404: トレース・データの解析に失敗しました。

- 1) トレース・データを検出することができませんでした。[エラー・ダイアログ](#)内の<詳細>>ボタンを押下し,発生原因,および,対処方法を確認してください。

E1407: TW 用のタイマ初期化処理実行中に異常が発生しました。

- 1) TW850 用のタイマ初期化処理実行中に,想定外の要因で実行が停止しました。タイマ制御コード部の tw_init_timer の処理,または,tw_init_timer_timing_address の指定に,TW850 のサポート外の指定をしている可能性があります。ユーザーズ・マニュアルを参照して,TW850 のサポート内の範囲で動作をしているか確認してください。
- 2) プロファイル・ルーチン,プロファイル・テーブルの配置先に誤りがある可能性があります。ユーザーズ・マニュアルを参照して,TW850 のサポートしている範囲内で指定しているか確認してください。
- 3) ターゲット・システムが不安定になっている可能性があります。ターゲット・システムを確認して,リセット後再度実行してください。
- 4) ディバッガが不安定になっている可能性があります。解析対象の実行モジュールをディバッガで正常に実行できるか確認してください。
- 5) Windows が不安定になっている可能性があります。Windows を再起動してください。

E1408: 実行開始位置から解析開始位置までの実行中に異常が発生しました。

- 1) 実行開始位置から解析開始位置までの実行中に,想定外の要因で実行が停止しました。実行開始位置,または,解析関数/解析開始位置/トレース・オン・イベントの指定に,TW850 のサポート外の箇所を指定している可能性があります。ユーザーズ・マニュアルを参照して,TW850 のサポート内の範囲で指定しているか確認してください。
- 2) プロファイル・ルーチン,プロファイル・テーブルの配置先に誤りがある可能性があります。マニュアルを参照して,TW のサポートしている範囲内で指定しているか確認してください。
- 3) ターゲット・システムが不安定になっている可能性があります。ターゲット・システムを確認して,リセット後再度実行してください。
- 4) ディバッガが不安定になっている可能性があります。解析対象の実行モジュールをディバッガで正常に実行できるか確認してください。
- 5) Windows が不安定になっている可能性があります。Windows を再起動してください。

E1409: 解析開始位置から解析終了位置までの実行中に異常が発生しました。

- 1) 解析開始位置から解析終了位置までの実行中に、想定外の要因で実行が停止しました。解析関数 / 解析開始位置 / トレース・オン・イベントの指定、または、解析終了位置 / トレース・オフ・イベントの指定に、TW850 のサポート外の個所を指定している可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポート内の範囲で指定しているか確認してください。
- 2) プロファイル・ルーチン、プロファイル・テーブルの配置先に誤りがある可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポートしている範囲内で指定しているか確認してください。
- 3) ターゲット・システムが不安定になっている可能性があります。ターゲット・システムを確認して、リセット後再度実行してください。
- 4) デバッグが不安定になっている可能性があります。解析対象の実行モジュールをデバッグで正常に実行できるか確認してください。
- 5) Windows が不安定になっている可能性があります。Windows を再起動してください。

E1410: 実行開始位置からタイマ初期化位置 (%s) までの実行中に異常が発生しました。

- 1) 実行開始位置からタイマ初期化位置までの実行中に、想定外の要因で実行が停止しました。実行開始位置の指定に、TW850 のサポート外の個所を指定している可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポート内の範囲で指定しているか確認してください。
- 2) タイマ制御コード部でタイマ初期化位置を指定している場合、その指定に誤りがある可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポートしている範囲内で指定しているか確認してください。
- 3) プロファイル・ルーチン、プロファイル・テーブルの配置先に誤りがある可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポートしている範囲内で指定しているか確認してください。
- 4) ターゲット・システムが不安定になっている可能性があります。ターゲット・システムを確認して、リセット後再度実行してください。
- 5) デバッグが不安定になっている可能性があります。解析対象の実行モジュールをデバッグで正常に実行できるか確認してください。
- 6) Windows が不安定になっている可能性があります。Windows を再起動してください。

E1411: タイマ初期化位置 (%s) から解析開始位置までの実行中に異常が発生しました。

- 1) タイマ初期化位置から解析開始位置までの実行中に、想定外の要因で実行が停止しました。解析関数 / 解析開始位置の指定に、TW850 のサポート外の個所を指定している可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポート内の範囲で指定しているか確認してください。
- 2) タイマ制御コード部でタイマ初期化位置を指定している場合、その指定に誤りがある可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポートしている範囲内で指定しているか確認してください。
- 3) ターゲット・システムが不安定になっている可能性があります。ターゲット・システムを確認して、リセット後再度実行してください。
- 4) デバッグが不安定になっている可能性があります。解析対象の実行モジュールをデバッグで正常に実行できるか確認してください。
- 5) Windows が不安定になっている可能性があります。Windows を再起動してください。

E1412: TW の時間プロファイル前処理実行中に異常が発生しました。

- 1) TW850 の時間プロファイル前処理の実行中に、想定外の要因で実行が停止しました。解析関数 / 解析開始位置の指定、または、解析終了位置の指定に、TW850 のサポート外の個所を指定している可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポート内の範囲で指定しているか確認してください。
- 2) タイマ制御コード部でタイマ初期化位置を指定している場合、その指定に誤りがある可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポートしている範囲内で指定しているか確認してください。
- 3) プロファイル・ルーチン、プロファイル・テーブルの配置先に誤りがある可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポートしている範囲内で指定しているか確認してください。
- 4) ターゲット・システムが不安定になっている可能性があります。ターゲット・システムを確認して、リセット後再度実行してください。
- 5) デバッグが不安定になっている可能性があります。解析対象の実行モジュールをデバッグで正常に実行できるか確認してください。
- 6) Windows が不安定になっている可能性があります。Windows を再起動してください。

E1413: TW の時間プロファイル後処理実行中に異常が発生しました。

- 1) TW850 の時間プロファイル後処理の実行中に、想定外の要因で実行が停止しました。解析関数 / 解析開始位置の指定、または、解析終了位置の指定に、TW850 のサポート外の個所を指定している可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポート内の範囲で指定しているか確認してください。
- 2) タイマ制御コード部でタイマ初期化位置を指定している場合、その指定に誤りがある可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポートしている範囲内で指定しているか確認してください。
- 3) プロファイル・ルーチン、プロファイル・テーブルの配置先に誤りがある可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポートしている範囲内で指定しているか確認してください。
- 4) ターゲット・システムが不安定になっている可能性があります。ターゲット・システムを確認して、リセット後再度実行してください。
- 5) デバッグが不安定になっている可能性があります。解析対象の実行モジュールをデバッグで正常に実行できるか確認してください。
- 6) Windows が不安定になっている可能性があります。Windows を再起動してください。

E1414: 解析終了位置から解析開始位置までの実行中に異常が発生しました。

- 1) 解析終了位置から解析開始位置までの実行中に、想定外の要因で実行が停止しました。トレース・オン・イベントの指定、または、解析終了位置 / トレース・オフ・イベントの指定に、TW850 のサポート外の個所を指定しているか、トレース回数の指定に誤りがある可能性があります。ユーザーズ・マニュアルを参照して、TW850 のサポート内の範囲で指定しているか確認してください。
- 2) ターゲット・システムが不安定になっている可能性があります。ターゲット・システムを確認して、リセット後再度実行してください。
- 3) デバッグが不安定になっている可能性があります。解析対象の実行モジュールをデバッグで正常に実行できるか確認してください。
- 4) Windows が不安定になっている可能性があります。Windows を再起動してください。

E1500: リンク・ディレクティブ・ファイル (%s) が不正です。

- 1) リンク・ディレクティブ・ファイル %s の解析処理が異常終了しています。ビルド・ツールをオリジナルのプロジェクト・ファイルから単独起動し、実行モジュールの生成処理が正常に行われているか否かを確認してください。
- 2) リンク・ディレクティブ・ファイル %s、または、リンク・ディレクティブ・ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。リンク・ディレクティブ・ファイル %s、および、リンク・ディレクティブ・ファイル %s が格納されているフォルダの属性を確認してください。
- 3) TW850 がサポートしていないフォーマットで記述されたリンク・ディレクティブ・ファイルである可能性があります。C コンパイラ・パッケージのバージョンが TW がサポートするバージョンか否かを確認してください。また、リリース・ノートなどを参照して、リンク・ディレクティブに関する注意事項を確認してください。

E1501: リンク・ディレクティブ・ファイルを生成できませんでした。

- 1) リンク・ディレクティブ・ファイルを格納するための TW850 用プロジェクト・データ・フォルダの属性がアクセス制限された状態となっている可能性があります。リンク・ディレクティブ・ファイルを格納するための TW850 用プロジェクト・データ・フォルダの属性を確認してください。
- 2) 不正なフォーマットのリンク・ディレクティブ・ファイルを指定した可能性があります。正常に実行モジュールを生成することが可能なリンク・ディレクティブ・ファイルであるか否かを確認してください。

E1502: リンク・ディレクティブ・ファイルを新規に生成するための情報が取得できませんでした。

- 1) リンク・ディレクティブ・ファイルを新規に生成するための情報の取得に失敗しました。生成する実行モジュールが、リンク・ディレクティブ・ファイルを作成することができないリロケータブル形式になっていないか確認してください。TW850 はリロケータブル形式を生成するプロジェクトから、リンク・ディレクティブ・ファイルを新規に生成することはできません。アドレスが解決する実行モジュールを生成するプロジェクトに変更してください。

E1503: セクション配置を指定するオプション (%s) が存在しません。

- 1) リンク・ディレクティブ・ファイル中には、セクション配置を指定するオプションを指定する必要があります。セクション配置を指定するオプションを指定してください。

E1504: セクション配置を指定するオプション (%s) が複数指定されています (%s)。

- 1) リンク・ディレクティブ・ファイル中に、セクション配置を指定するオプションを複数指定することはできません。セクション配置を指定するオプションは 1 回のみ指定してください。

E1505: (%s) 情報の存在する領域が、不正な文字 (%s) から始まっています (%s)。

- 1) リンク・ディレクティブ・ファイル中に、構文エラーがあります。構文エラーを修正してください。

E1506: 記号 (%s) の使い方に誤りがあります (%s)。

- 1) リンク・ディレクティブ・ファイル中に、構文エラーがあります。構文エラーを修正してください。

E1507: 不正な指定文字 (%s) があります (%s)。

- 1) リンク・ディレクティブ・ファイル中に、構文エラーがあります。構文エラーを修正してください。

E1508: 数値変換できない表現 (%s) があります (%s)。

- 1) リンク・ディレクティブ・ファイル中に、数値に変換できない文字を指定している構文エラーがあります。数値に変換できるように記述してください。

E1602: ターゲットへ実行モジュール (%s) をダウンロードできませんでした。

- 1) 実行モジュールのダウンロード処理が異常終了しています。実行モジュールのマッピング・サイズ指定が正常に行われているか否かを確認してください。
- 2) 実行モジュール %s, または、実行モジュール %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。実行モジュール %s, および、実行モジュール %s が格納されているフォルダの属性を確認してください。
- 3) 実行モジュール %s をアクセス制限されているメモリ領域に対してダウンロードしている可能性があります。実行モジュールのダウンロード処理を行うメモリ領域の割り付け、および、周辺 I/O レジスタの設定が RAM 領域に対して行われていないか確認してください。
- 4) 不正なフォーマットの実行モジュール %s を指定した可能性があります。実行モジュール %s が TW850 のサポートしている C コンパイラ・パッケージによって生成されたものであるか否かを確認してください。
- 5) 他ツールが TW850 のサポートしていないバージョンである可能性があります。他ツールが TW850 のサポートしているバージョンであるか否かを確認してください。
- 6) 他ツールが正常に動作していない可能性があります。他ツールを再起動してください。
- 7) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E1603: レジスタ (%s) から値を読み込めませんでした。

- 1) 他ツールが TW850 のサポートしていないバージョンである可能性があります。他ツールが TW850 のサポートしているバージョンであるか否かを確認してください。
- 2) 他ツールが正常に動作していない可能性があります。他ツールを再起動してください。
- 3) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E1604: レジスタ (%s) に値を書き込めませんでした。

- 1) 他ツールが TW850 のサポートしていないバージョンである可能性があります。他ツールが TW850 のサポートしているバージョンであるか否かを確認してください。
- 2) 他ツールが正常に動作していない可能性があります。他ツールを再起動してください。
- 3) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E1605: メモリ (%s) から読み込めませんでした。

- 1) 読み込み処理をアクセス制限されているメモリ領域に対して行っている可能性があります。読み込み処理を行うメモリ領域の割り付け、および、周辺 I/O レジスタの設定が RAM 領域に対して行われていないか確認してください。
- 2) 他ツールが TW850 のサポートしていないバージョンである可能性があります。他ツールが TW850 のサポートしているバージョンであるか否かを確認してください。
- 3) 他ツールが正常に動作していない可能性があります。他ツールを再起動してください。
- 4) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E1606: メモリ (%s) に書き込めませんでした。

- 1) 書き込み処理をアクセス制限されているメモリ領域に対して行っている可能性があります。書き込み処理を行うメモリ領域の割り付け、および、周辺 I/O レジスタの設定が RAM 領域に対して行われていないか確認してください。
- 2) 他ツールが TW850 のサポートしていないバージョンである可能性があります。他ツールが TW850 のサポートしているバージョンであるか否かを確認してください。
- 3) 他ツールが正常に動作していない可能性があります。他ツールを再起動してください。
- 4) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E1607: メモリ (%s) にフィルできませんでした。

- 1) フィル処理をアクセス制限されているメモリ領域に対して行っている可能性があります。フィル処理を行うメモリ領域の割り付け、および、周辺 I/O レジスタの設定が RAM 領域に対して行われていないか確認してください。
- 2) 他ツールが TW850 のサポートしていないバージョンである可能性があります。他ツールが TW850 のサポートしているバージョンであるか否かを確認してください。
- 3) 他ツールが正常に動作していない可能性があります。他ツールを再起動してください。
- 4) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E1608: 以下の他ツールとの連携動作に失敗しました。

%s

- 1) 他ツールが TW850 のサポートしていないバージョンである可能性があります。他ツールが TW850 のサポートしているバージョンであるか否かを確認してください。
- 2) 他ツールが正常に動作していない可能性があります。他ツールを再起動してください。
- 3) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E1631: プロジェクト・ファイル (%s) を読み込めませんでした。

- 1) プロジェクト・ファイル %s, または, プロジェクト・ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。プロジェクト・ファイル %s, および, プロジェクト・ファイル %s が格納されているフォルダの属性を確認してください。
- 2) 他ツールが TW850 のサポートしていないバージョンである可能性があります。他ツールが TW850 のサポートしているバージョンであるか否かを確認してください。
- 3) 他ツールが正常に動作していない可能性があります。他ツールを再起動してください。
- 4) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E1632: カレント・プロジェクト・ファイル名を取得できませんでした。

- 1) 他ツールが TW850 のサポートしていないバージョンである可能性があります。他ツールが TW850 のサポートしているバージョンであるか否かを確認してください。
- 2) 他ツールが正常に動作していない可能性があります。他ツールを再起動してください。
- 3) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E1636: ヘルプ・ファイル (%s) を開けませんでした。

- 1) ヘルプ・ファイル %s を検出することができませんでした。ヘルプ・ファイル %s が存在しているか否かを確認してください。
- 2) TW850 が正常にインストールされていない可能性があります。TW850 をアンインストールした後, TW850 を再インストールしてください。

E1637: レジストリ値の一覧を取得できませんでした (%s)。

- 1) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。
- 2) TW850, または, 他ツールが正常にインストールされていない可能性があります。TW850, または, 他ツールをアンインストールした後, TW850, または, 他ツールを再インストールしてください。

E1638: レジストリ値の文字列を取得できませんでした (%s)。

- 1) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。
- 2) TW850, または, 他ツールが正常にインストールされていない可能性があります。TW850, または, 他ツールをアンインストールした後, TW850, または, 他ツールを再インストールしてください。

E1639: レジストリ値の数値を取得できませんでした (%s)。

- 1) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。
- 2) TW850, または, 他ツールが正常にインストールされていない可能性があります。TW850, または, 他ツールをアンインストールした後, TW850, または, 他ツールを再インストールしてください。

E1640: 外部コマンドを実行できませんでした。

- 1) 外部コマンドの起動処理が異常終了しています。[エラー・ダイアログ](#)内の <詳細>> ボタンを押下し, 発生原因, および, 対処方法を確認してください。

E1646: デバッグを指定時間 (%s 秒) 内に起動できませんでした。

- 1) デバッグが指定時間内に操作可能な状態になりませんでした。TW850 の起動オプション -Dtw.DebuggerTimeout で指定されたタイムアウト値 %s を適切な値に変更してください。
- 2) デバッグが正常に動作していない可能性があります。デバッグを単独起動し、正常に動作するか否かを確認してください。
- 3) TW850 が対応していないバージョンのデバッグを指定している可能性があります。デバッグのバージョンを確認してください。
- 4) Windows が不安定な状態になっている可能性があります。表示メッセージ %s の内容にしたがって、Windows を再起動してください。

E1700: 外部コマンドの実行に失敗しました (%s)。

- 1) 外部コマンドの起動処理が異常終了しています。該当外部コマンド %s をコマンド・プロンプトから起動した際、正常に起動しているか否かを確認してください。
- 2) 外部コマンドを検出することができませんでした。環境変数設定ファイル (autoexec.bat など) において、ホスト・マシンの環境変数として “ 該当外部コマンドへ %s の検索パス ” が設定されているか否かを確認してください。
- 3) 外部コマンドが動作する際に必要なメモリ領域が不足している可能性があります。不要アプリケーションの終了などを行い、充分な空きメモリ領域を確保してください。

E1801: デバイス情報 (%s) がレジストリに登録されていません。

- 1) デバイス・ファイルが正常にインストールされていない可能性があります。デバイス・ファイルをアンインストールした後、デバイス・ファイルを再インストールしてください。

E1803: デバイス・ファイル (%s) が存在しません。

- 1) デバイス・ファイル %s を検出することができませんでした。環境変数設定ファイル (autoexec.bat など) において、ホスト・マシンの環境変数として “ デバイス・ファイル %s への検索パス ” が設定されているか否かを確認してください。
- 2) デバイス・ファイル %s、または、デバイス・ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。デバイス・ファイル %s、および、デバイス・ファイル %s が格納されているフォルダの属性を確認してください。

E1804: デバイス情報が起動時に指定されていません。

- 1) デバッグを指定しないときのデバイスの指定が正確にされていません。[TW850 起動オプション設定ダイアログ](#) で正確なデバイスを指定してください。また、デバイス・ファイルを使用している場合は、そのデバイス・ファイルが正常にインストールされているか確認してください。

E1900: ToolLinkManager に接続できません。

- 1) TW850 が不安定な状態になっている可能性があります。TW850 を再起動してください。
- 2) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。
- 3) ToolLinkManager が正常にインストールされていない可能性があります。ToolLinkManager をアンインストールした後、ToolLinkManager を再インストールしてください。

E1901: ToolLinkManager が起動していません。

- 1) ToolLinkManager が正常に動作していない可能性があります。NEC エレクトロニクス製ツール (PM plus, ID850, ID850NW, SM850, SM plus for V850, TW850) を終了した後、ToolLinkManager を再起動してください。

E9000: TW を起動できませんでした。

- 1) Java 実行環境が正常にインストールされていない可能性があります。TW850 のサポートしている Java 実行環境 (Java2 Runtime Environment Standard Edition V1.4.2_04) をインストールしてください。
- 2) Java 実行環境を検出することができませんでした。環境変数設定ファイル (autoexec.bat など) において、ホスト・マシンの環境変数として “ Java 実行環境への検索パス ” が設定されているか否かを確認してください。

E9001: オプション (%s) が不正です。

- 1) プロジェクト・ファイルを検出することができませんでした。プロジェクト・ファイルが存在しているか否かを確認してください。
- 2) プロジェクト・ファイル, または, プロジェクト・ファイルが格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。プロジェクト・ファイル, および, プロジェクト・ファイルが格納されているフォルダの属性を確認してください。
- 3) プロジェクトに TW850 のサポートしていないオプションが指定されている可能性があります。オプション %s の指定をプロジェクトから削除してください。

E9002: 致命的なエラーが発生しました。TW を起動できませんでした。

- 1) JavaVM が動作する際に必要なメモリ領域が不足している可能性があります。不要アプリケーションの終了などを行い, 十分な空きメモリ領域を確保してください。
- 2) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E9003: Java アプリケーション起動コマンド (%s) が見つかりませんでした。

- 1) Java 実行環境が正常にインストールされていない可能性があります。TW850 のサポートしている Java 実行環境 (Java2 Runtime Environment Standard Edition V1.4.2_04) をインストールしてください。
- 2) Java アプリケーション起動コマンド %s を検出することができませんでした。環境変数設定ファイル (autoexec.bat など) において, ホスト・マシンの環境変数として “Java アプリケーション起動コマンドへの検索パス” が設定されているか否かを確認してください。
- 3) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E9004: TW の起動オプションに誤りがあるか, Java の実行中に致命的なエラーが発生しました。

- 1) TW850 の起動オプションのその他のオプションにオプションを指定している場合, そのオプションの指定に誤りがある可能性があります。「[3.1.2 起動オプション](#)」, Java のユーザズ・マニュアルを参照して, オプションの指定が正しいか確認してください。
- 2) TW850 と関係する他ツールが正しく動作していない可能性があります。一旦, 他ツールを終了し, 再度起動してください。
- 3) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。
- 4) Java が正しく動作していない可能性があります。Java をアンインストール後, 再インストールしてください。また, Java のバージョンが, TW850 のサポートするバージョンか確認してください。
- 5) TW850 と関係する他ツールが正しくインストールされていない可能性があります。他ツールをアンインストール後, 再インストールしてください。また, 関係する他ツールのバージョンが, TW850 のサポートするバージョンか確認してください。

E9100: %s から %s にファイルをコピーできませんでした。

- 1) コピー元ファイル %s を検出することができませんでした。コピー元ファイル %s が存在しているか否かを確認してください。
- 2) ファイル %s, または, ファイル %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。ファイル %s, および, ファイル %s が格納されているフォルダの属性を確認してください。
- 3) ファイルをコピーする際に必要なディスク領域が不足している可能性があります。不要ファイルの削除などを行い, 十分なディスク領域を確保してください。

E9101: ラッパー・コマンド (%s) の検査に失敗しました。

- 1) TW850 が提供するアセンブラ用ラッパー・コマンド AS850.EXE の属性がアクセス制限された状態となっている可能性があります。TW850 が提供するアセンブラ用ラッパー・コマンド AS850.EXE の属性を確認してください。
- 2) ラッパー・コマンド %s, または, ラッパー・コマンド %s が格納されているフォルダの属性がアクセス制限された状態となっている可能性があります。ラッパー・コマンド %s, および, ラッパー・コマンド %s が格納されているフォルダの属性を確認してください。
- 3) ラッパー・コマンド %s が動作する際に必要なメモリ領域が不足している可能性があります。不要アプリケーションの終了などを行い, 十分な空きメモリ領域を確保してください。

E9102: レジストリのアクセスに失敗しました。

- 1) レジストリ情報の属性がアクセス制限された状態となっている可能性があります。Administrator 権限を有するユーザで TW850 を起動してください。
- 2) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。

E9103: TWCommon.dll あるいはレジストリが不正です。

- 1) Windows が不安定な状態になっている可能性があります。Windows を再起動してください。
- 2) TW850 が提供している外部コマンド共通ライブラリ TWCommon.dll, または, レジストリ情報の内容が破壊されている可能性があります。TW850 をアンインストールした後, TW850 を再インストールしてください。

付録 B 制限値・限界値

本付録では、TW850 の制限値、および、限界値について解説しています。

B.1 制限値

TW850 の制限値を超えるような場合、性能解析処理、および、チューニング処理を実行することはできません。
表 B-1 に、TW850 の制限値を示します。

表 B-1 TW850 の制限値

項目	制限値
ユーザ関数の数	64,511
ユーザ関数のコールペア数とシステム・ライブラリ関数のコールペア数の合計	2,147,483,647
アセンブリ言語ソース・ファイルにおけるマクロのネスト使用	32 段階

B.2 限界値

表 B-2 に、TW850 の限界値を示します。

表 B-2 TW850 の限界値

項目	限界値
コールペアの呼び出し回数 (トレース・データ解析方式)	9,223,372,036,854,775,807
コールペアの呼び出し回数 (プローブ・コード挿入方式)	2,147,483,646
関数の実行回数	9,223,372,036,854,775,8072
プローブ・コード挿入方式を利用した時間計測処理における関数の実行時間計測クロック	2,147,483,646 * タイマ1カウントあたりのCPUクロック数

注意 タイマ 1 カウントあたりの CPU クロック数は、時間計測用ファイル (タイマ制御コード部) においてユーザが指定した値となります。時間計測用ファイル (タイマ制御コード部) については、「[4.4 タイマ制御コード部](#)」を参照してください。

付録 C 中間ファイル

本付録では、TW850 がユーザ・プログラム、実行モジュールの性能解析時、チューニング時に出力される中間ファイルについて解説しています。

以下に、出力される中間ファイルの一覧を示します。

なお、最終的には、チューニング後実行モジュール (オリジナル実行モジュール名)_o.out が出力されます。

ただし、デバッグ・ターゲットとしてヘキサ・ファイルを指定している場合は、チューニング後実行モジュールのヘキサ・フォーマット・ファイル“(オリジナル実行モジュール名)_o.hex”が出力されます。

また、(オリジナル・プロジェクト・ファイル名).tcl が存在する場合は、プロジェクト・ファイル用 Tcl ファイル“(オリジナル・プロジェクト・ファイル名)_o.tcl”が出力され、tw.tcl、または、(オリジナル実行モジュール名).tcl が存在する場合は、チューニング後実行モジュールの Tcl ファイル“(オリジナル実行モジュール名)_o.tcl”が出力されます。

twerrlog.txt ,	(オリジナル・プロジェクト・ファイル名)_s.prj ,
(オリジナル・プロジェクト・ファイル名)_i.prj ,	(オリジナル・プロジェクト・ファイル名)_i.tcl ,
(オリジナル・プロジェクト・ファイル名)_p.prj ,	(オリジナル・プロジェクト・ファイル名)_p.tcl ,
(オリジナル・プロジェクト・ファイル名)_o.prj ,	(オリジナル・プロジェクト・ファイル名)_o.tcl ,
(オリジナル・ユーザ・プログラム名).s ,	(オリジナル・ユーザ・プログラム名).stw ,
(オリジナル・ユーザ・プログラム名)_i.c2s ,	(オリジナル・ユーザ・プログラム名)_i.s ,
(オリジナル・ユーザ・プログラム名)_p.c2s ,	(オリジナル・ユーザ・プログラム名)_p.s ,
(オリジナル・ユーザ・プログラム名)_o.c2s ,	(オリジナル・ユーザ・プログラム名)_o.s ,
(オリジナル・ユーザ・プログラム名)_i.o ,	(オリジナル・ユーザ・プログラム名)_p.o ,
(オリジナル・ユーザ・プログラム名)_o.o ,	twprftbl.s ,
twprftbl.o ,	(タイマ制御コード部プログラム・ファイル名).o ,
(オリジナル・プロジェクト・ファイル名).pdb ,	(オリジナル・プロジェクト・ファイル名)_*.pdb ,
(オリジナル実行モジュール名)_i.out ,	(オリジナル実行モジュール名)_i.xxx ,
(オリジナル実行モジュール名)_i.tcl ,	(オリジナル実行モジュール名)_i.tft ,
(オリジナル実行モジュール名)_p.out ,	(オリジナル実行モジュール名)_p.xxx ,
(オリジナル実行モジュール名)_p.tcl ,	(オリジナル実行モジュール名)_o.out ,
(オリジナル実行モジュール名)_o.xxx ,	(オリジナル実行モジュール名)_o.tcl ,
(オリジナル実行モジュール名)_o.tft ,	(オリジナル・リンク・ディレクティブ・ファイル名)

- twerrlog.txt

エラー・メッセージを格納したログ・ファイルです。

なお、本ファイルは、TW850 が致命的なエラーを検出した際、オリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

注意 プロジェクトが確定していない場合には、本ファイルは TW850 の実行形式ファイルと同一フォルダに出力されます。

- (オリジナル・プロジェクト・ファイル名)_s.prj

C 言語ソース・プログラムをアセンブリ言語ソース・プログラムに変換するためのプロジェクト・ファイルです。

なお、本ファイルは、性能解析時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

- (オリジナル・プロジェクト・ファイル名)_i.prj

オリジナル状態の実行モジュール (オリジナル解析用実行モジュール) を生成するためのプロジェクト・ファイルです。

なお、本ファイルは、性能解析時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

- (オリジナル・プロジェクト・ファイル名)_i.tcl

オリジナル状態の実行モジュール (オリジナル解析用実行モジュール) を実行するためのプロジェクト・ファイル用 Tcl ファイルです。

なお、本ファイルは、性能解析時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

注意 (オリジナル・プロジェクト・ファイル名).tcl が存在する場合のみ出力されます。

- (オリジナル・プロジェクト・ファイル名)_p.prj
 プローブ・コードが挿入された実行モジュール (プロファイル用実行モジュール) を生成するためのプロジェクト・ファイルです。
 なお、本ファイルは、プローブ・コード挿入方式による性能解析時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。
- (オリジナル・プロジェクト・ファイル名)_p.tcl
 プローブ・コードが挿入された実行モジュール (プロファイル用実行モジュール) を実行するためのプロジェクト・ファイル用 Tcl ファイルです。
 なお、本ファイルは、プローブ・コード挿入方式による性能解析時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。
 注意 (オリジナル・プロジェクト・ファイル名).tcl が存在する場合のみ出力されます。
- (オリジナル・プロジェクト・ファイル名)_o.prj
 TW850 によるチューニング処理が完了した実行モジュール (チューニング後実行モジュール) を生成するためのプロジェクト・ファイルです。
 なお、本ファイルは、チューニング時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。
- (オリジナル・プロジェクト・ファイル名)_o.tcl
 TW850 によるチューニング処理が完了した実行モジュール (チューニング後実行モジュール) を実行するためのプロジェクト・ファイル用 Tcl ファイルです。
 なお、本ファイルは、チューニング時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。
 注意 (オリジナル・プロジェクト・ファイル名).tcl が存在する場合のみ出力されます。
- (オリジナル・ユーザ・プログラム名).s
 オリジナル・ユーザ・プログラム (C 言語ソース・ファイル) を性能解析情報獲得用アセンブリ言語ソース・プログラムに変換した中間出力ファイルです。
 なお、本ファイルは、性能解析時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。
 注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。
- (オリジナル・ユーザ・プログラム名).stw
 オリジナル・ユーザ・プログラム (アセンブリ言語ソース・ファイル) を性能解析情報獲得用アセンブリ言語ソース・プログラムに変換した中間出力ファイルです。
 なお、本ファイルは、性能解析時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。
 注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。
- (オリジナル・ユーザ・プログラム名)_i.c2s
 オリジナル・ユーザ・プログラム (C 言語ソース・ファイル) を性能解析情報獲得用アセンブリ言語ソース・プログラムに変換した中間出力ファイルです。
 なお、本ファイルは、実行モジュール (オリジナル解析用実行モジュール) 生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。
 注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。
- (オリジナル・ユーザ・プログラム名)_i.s
 オリジナル・ユーザ・プログラム (アセンブリ言語ソース・ファイル) を性能解析情報獲得用アセンブリ言語ソース・プログラムに変換した中間出力ファイルです。
 なお、本ファイルは、実行モジュール (オリジナル解析用実行モジュール) 生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。
 注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。

- (オリジナル・ユーザ・プログラム名)_p.c2s

オリジナル・ユーザ・プログラム (C 言語ソース・ファイル) を性能解析情報獲得用アセンブリ言語ソース・プログラムに変換した中間出力ファイルです。

なお、本ファイルは、実行モジュール (プロファイル用実行モジュール) 生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。

- (オリジナル・ユーザ・プログラム名)_p.s

オリジナル・ユーザ・プログラム (アセンブリ言語ソース・ファイル) を性能解析情報獲得用アセンブリ言語ソース・プログラムに変換した中間出力ファイルです。

なお、本ファイルは、実行モジュール (プロファイル用実行モジュール) 生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。

- (オリジナル・ユーザ・プログラム名)_o.c2s

オリジナル・ユーザ・プログラム (C 言語ソース・ファイル) を性能解析情報獲得用アセンブリ言語ソース・プログラムに変換した中間出力ファイルです。

なお、本ファイルは、実行モジュール (チューニング後実行モジュール) 生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。

- (オリジナル・ユーザ・プログラム名)_o.s

オリジナル・ユーザ・プログラム (アセンブリ言語ソース・ファイル) を性能解析情報獲得用アセンブリ言語ソース・プログラムに変換した中間出力ファイルです。

なお、本ファイルは、実行モジュール (チューニング後実行モジュール) 生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。

- (オリジナル・ユーザ・プログラム名)_i.o

オリジナル・ユーザ・プログラム (C 言語ソース・ファイル、アセンブリ言語ソース・ファイル) を性能解析情報獲得用オブジェクト・ファイルに変換した中間出力ファイルです。

なお、本ファイルは、実行モジュール (オリジナル解析用実行モジュール) 生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。

- (オリジナル・ユーザ・プログラム名)_p.o

オリジナル・ユーザ・プログラム (C 言語ソース・ファイル、アセンブリ言語ソース・ファイル) を性能解析情報獲得用オブジェクト・ファイルに変換した中間出力ファイルです。

なお、本ファイルは、実行モジュール (プロファイル用実行モジュール) 生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。

- (オリジナル・ユーザ・プログラム名)_o.o

オリジナル・ユーザ・プログラム (C 言語ソース・ファイル、アセンブリ言語ソース・ファイル) を性能解析情報獲得用オブジェクト・ファイルに変換した中間出力ファイルです。

なお、本ファイルは、実行モジュール (チューニング後実行モジュール) 生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。

- twprftbl.s

プロファイル情報獲得用アセンブリ言語ソース・プログラムです。

なお、本ファイルは、**配置先選択パネル**のプロファイル・テーブル指定領域の<計算する(L)>ボタンを選択時、または、実行モジュール(プロファイル用実行モジュール)生成前にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

- twprftbl.o

プロファイル情報獲得用アセンブリ言語ソース・プログラムをプロファイル情報獲得用オブジェクト・ファイルに変換した中間出力ファイルです。

なお、本ファイルは、実行モジュール(プロファイル用実行モジュール)生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

注意 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。

- (タイマ制御コード部プログラム・ファイル名).o

タイマ制御コード部プログラム・ファイル(アセンブリ言語ソース・プログラム)をタイマ制御コード部プログラム・ファイル(オブジェクト・ファイル)に変換した中間出力ファイルです。

なお、本ファイルは、実行モジュール(プロファイル用実行モジュール)生成時にオリジナル・プロジェクト・ファイルと同一フォルダに出力されます。

ただし、

注意 1 本ファイルは、性能解析項目として“プローブ・コード解析方式”、および、“関数の実行時間を計測”を指定した場合に限り出力されます。

注意 2 PM plus で中間出力ファイルの出力先フォルダが定義されている場合には、本ファイルは該当フォルダに出力されます。

- (オリジナル・プロジェクト・ファイル名).pdb

TW850 のプロジェクト情報を格納したファイルです。

なお、本ファイルは、TW850 の**メイン・ウィンドウ**の[ファイル(E)]メニュー→[プロジェクト情報保存(S)]を選択した際、または、TW850 を終了した際、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。

- (オリジナル・プロジェクト・ファイル名)_*.pdb

TW850 用プロジェクト・ファイル(TW850 のプロジェクト情報を格納したファイル)のバックアップ・ファイルです。

なお、本ファイルは、使用中の TW850 と異なるバージョンの TW850 用プロジェクト・ファイルを読み込んだ際、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。

- (オリジナル実行モジュール名)_i.out

オリジナル解析用実行モジュールです。

なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。

- (オリジナル実行モジュール名)_i.xxx

オリジナル解析用実行モジュールのヘキサ・フォーマット・ファイルです。

なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。

注意 本ファイルの拡張子“.xxx”は、オリジナル実行モジュールのヘキサ・フォーマット・ファイルの拡張子と同一になります。

- (オリジナル実行モジュール名)_i.tcl

オリジナル解析用実行モジュールの Tcl ファイルです。

なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。

注意 (オリジナル実行モジュール名).tcl が存在する場合のみ出力されます。

- (オリジナル実行モジュール名)_i.tft

オリジナル解析用実行モジュールの関数遷移情報を格納するデータ・ファイルです。

なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。

注意 関数遷移情報を取得した場合のみ出力されます。

- (オリジナル実行モジュール名)_p.out
プロファイル用実行モジュールです。
なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。
 - (オリジナル実行モジュール名)_p.xxx
プロファイル用実行モジュールのヘキサ・フォーマット・ファイルです。
なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。
注意 本ファイルの拡張子“.xxx”は、オリジナル実行モジュールのヘキサ・フォーマット・ファイルの拡張子と同一になります。
 - (オリジナル実行モジュール名)_p.tcl
プロファイル用実行モジュールの Tcl ファイルです。
なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。
注意 (オリジナル実行モジュール名).tcl が存在する場合のみ出力されます。
 - (オリジナル実行モジュール名)_o.out
チューニング後実行モジュールです。
なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。
 - (オリジナル実行モジュール名)_o.xxx
チューニング後実行モジュールのヘキサ・フォーマット・ファイルです。
なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。
注意 本ファイルの拡張子“.xxx”は、オリジナル実行モジュールのヘキサ・フォーマット・ファイルの拡張子と同一になります。
 - (オリジナル実行モジュール名)_o.tcl
チューニング後実行モジュールの Tcl ファイルです。
なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。
注意 (オリジナル実行モジュール名).tcl が存在する場合のみ出力されます。
 - (オリジナル実行モジュール名)_o.tft
チューニング後実行モジュールの関数遷移情報を格納するデータ・ファイルです。
なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。
注意 関数遷移情報を取得した場合のみ出力されます。
 - (オリジナル・リンク・ディレクティブ・ファイル名)
実行モジュール(オリジナル解析用実行モジュール, プロファイル用実行モジュール, チューニング後実行モジュール)を生成するためのリンク・ディレクティブ・ファイルです。
なお、本ファイルは、“(オリジナル・プロジェクト・ファイル名).tw”フォルダに出力されます。
- 注意 1 TW850 は、ソース・ファイル名をベースとして中間ファイルの名前を決定するため、プロジェクト内に、同じベース名を持つソース・ファイルがある場合 (source.c と source.s, source.850 など)、チューニング対象とすることができません。
- 注意 2 ソース・ファイル中で、中間ファイルと同名のファイルをインクルードしている場合 (source.c 中で、xxx/source.s をインクルードしている場合など) 中間ファイルを優先してインクルードする可能性があるため、チューニング対象とすることができません。
- 注意 3 複数のプロジェクトが同じソース・ファイル(アセンブリ言語ソース・ファイルを除く)を含む場合、それらのプロジェクトに対して交互に TW850 を用いてチューニング処理を行った場合、正常にビルドできない、または、不正なモジュールが生成される場合があります。
このような場合、プロジェクトを変更した後、[オプション・ダイアログ](#)で中間ファイルを削除することにより、正常にビルドを行えるようになります。
- 注意 4 TW850 が生成したプロジェクト、または、TW850 が生成したファイルを含むプロジェクトは、チューニング対象とすることができません。

索引

C

CPU	17
C コンパイラ・パッケージ	17

F

far jump 呼び出し	15
---------------------	----

G

GUI 機能	16, 37
--------------	--------

J

Java 実行環境	17
-----------------	----

R

RTOS(リアルタイム OS)	17
------------------------	----

T

TW850	15
GUI 機能	16, 37
インストレーション	18
ウィンドウ・レファレンス	90
概説	15
起動オプション	24
起動と終了	23
起動方法	24
機能	26
終了方法	25
制限値・限界値	215
性能解析機能	16, 26
操作方法	54
中間ファイル	216
チューニング機能	16, 33
動作環境	17
特徴	16
メッセージ	191
TW850 起動オプション設定ダイアログ	188
TW850 用システム・ライブラリ	42

あ

アンインストール手順	20
------------------	----

い

インストール手順	19
インストレーション	18, 21

アンインストール手順	20
インストール手順	19
フォルダ構成	21

う

ウィンドウ・レファレンス	90
TW850 起動オプション設定ダイアログ	188
エラー・ダイアログ	182
オプション・ダイアログ	174
解析対象ソース・ファイル選択パネル	123
確認ダイアログ	184
関数詳細情報ウィンドウ	116
関数情報テーブル・ウィンドウ	106
関数遷移情報ウィンドウ	109
関数遷移情報検索ダイアログ	170
高速アクセス・メモリ領域選択パネル	144
高速アクセス・メモリ領域チューニング選択パネル	149
コールグラフ・ウィンドウ	103
コールペア補正オプション選択パネル	153
実行解析範囲選択パネル	133
実行完了確認パネル	165
実行項目選択パネル	126
実行前確認パネル	162
実行モジュール情報ウィンドウ	114
実行モジュール情報獲得パネル	125
初期画面パネル	120
性能解析/チューニング実行パネル	163
チューニング方法選択パネル	146
トレース・イベント選択パネル	137
トレース解析オプション選択パネル	128
入力ダイアログ	186
配置先選択パネル	158
表示関数選択ダイアログ	166
ファイル選択ダイアログ	179
プロープ解析オプション選択パネル	130
プロジェクト情報獲得パネル	122
命令キャッシュ選択パネル	141
命令キャッシュ・チューニング選択パネル	151
メイン・ウィンドウ	95
メッセージ・ダイアログ	181
優先配置関数選択パネル	155
バージョン情報ダイアログ	177

え

エラー・ダイアログ	182
エラー・メッセージ	192, 196

お

オプション・ダイアログ	174
-------------------	-----

か

解析対象ソース・ファイル選択パネル	123
-------------------------	-----

確認ダイアログ	184
関数詳細情報ウィンドウ	116
関数情報テーブル・ウィンドウ	106
関数遷移情報ウィンドウ	109
関数遷移情報検索ダイアログ	170

き

起動オプション	24
起動方法	24
機能	26
基本ソフトウェア	17

け

警告メッセージ	192, 194
限界値	215

こ

高速アクセス・メモリ領域	15
高速アクセス・メモリ領域選択パネル	144
高速アクセス・メモリ領域チューニング選択パネル	149
コールグラフ・ウィンドウ	103
コールペア	15
コールペア補正オプション選択パネル	153

し

実行エンジン	15
実行解析範囲選択パネル	133
実行完了確認パネル	165
実行項目選択パネル	126
実行前確認パネル	162
実行モジュール	15
実行モジュール情報ウィンドウ	114
実行モジュール情報獲得パネル	125
シミュレータ	17
終了方法	25
情報メッセージ	192
初期画面パネル	120

せ

制限値	215
制限値・限界値	215
性能解析機能	16, 26
性能解析 / チューニング実行パネル	163

そ

操作方法	54
ソフトウェア環境	17
C コンパイラ・パッケージ	17
Java 実行環境	17
基本ソフトウェア	17
シミュレータ	17
ディバッガ	17
デバイス・ファイル	17
プロジェクト・マネージャ	17
RTOS(リアルタイム OS)	17

た

タイマ制御コード部	38
-----------------	----

ち

中間ファイル	216
チューニング機能	16, 33
チューニング方法選択パネル	146

て

ディバッガ	17
デバイス・ファイル	17

と

動作環境	17
ソフトウェア環境	17
ハードウェア環境	17
特徴	16
トレース・イベント選択パネル	137

に

入力ダイアログ	186
---------------	-----

は

バージョン情報ダイアログ	177
ハードウェア環境	17
CPU	17
ホストマシン	17
メイン・メモリ	17
配置先選択パネル	158

ひ

表示関数選択ダイアログ	166
-------------------	-----

ふ

ファイル選択ダイアログ	179
フォルダ構成	21
プロジェクト情報獲得パネル	122
プロジェクト・マネージャ	17

ほ

ホスト・マシン	17
ホット・パス	15

め

命令キャッシュ選択パネル	141
命令キャッシュ・チューニング選択パネル	151
命令キャッシュ・ミス	15
メイン・ウィンドウ	95
メイン・メモリ	17
メッセージ	191
エラー・メッセージ	192, 196
警告メッセージ	192, 194
情報メッセージ	192
表示形式	191
メッセージ・ダイアログ	181

ゆ

優先配置関数選択パネル	155
-------------------	-----

【発 行】

NECエレクトロニクス株式会社

〒211-8668 神奈川県川崎市中原区下沼部1753

電話（代表）：044(435)5111

お問い合わせ先

【ホームページ】

NECエレクトロニクスの情報がインターネットでご覧になれます。

URL（アドレス） <http://www.necel.co.jp/>

【営業関係，技術関係お問い合わせ先】

半導体ホットライン

（電話：午前 9:00～12:00，午後 1:00～5:00）

電 話 ： 044-435-9494

E-mail ： info@necel.com

【資料請求先】

NECエレクトロニクスのホームページよりダウンロードいただくか，NECエレクトロニクスの販売特約店へお申し付けください。