

**HITACHI**

ソフトウェアマニュアル  
オペレーション  
**S10V CPMSデバッグ**  
**For Windows®**

---

**S10V**

Programmable Controller

ソフトウェアマニュアル  
オペレーション  
**S10V CPMSデバツガ**  
**For Windows®**

---

**S10V**  
Programmable Controller

この製品を輸出される場合には、外国為替及び外国貿易法の規制並びに米国輸出管理規則など外国の輸出関連法規をご確認のうえ、必要な手続きをお取りください。  
なお、不明な場合は、弊社担当営業にお問い合わせください。

2004年 2月 (第1版) SVJ-3-126 (A) (廃版)  
2012年 9月 (第2版) SVJ-3-126 (B) (廃版)  
2017年 6月 (第3版) SVJ-3-126 (C)

- このマニュアルの一部または全部を無断で転写したり複製したりすることは、固くお断りいたします。
- このマニュアルの内容を、改良のため予告なしに変更することがあります。

## 安全上のご注意

- システムの構築やプログラムの作成などは、このマニュアルの記載内容をよく読み、書かれている指示や注意を十分理解してから行ってください。誤操作により、システムが故障することがあります。
- このマニュアルは、必要なときすぐに参照できるよう、手近なところに保管してください。
- このマニュアルの記載内容について疑問点または不明点がございましたら、最寄りの弊社営業またはSEまでお知らせください。
- お客様の誤操作に起因する事故発生や損害については、弊社は責任を負いかねますのでご了承ください。
- 弊社提供ソフトウェアを改変して使用した場合に発生した事故や損害については、弊社は責任を負いかねますのでご了承ください。
- 弊社提供以外のソフトウェアを使用した場合の信頼性については、弊社は責任を負いかねますのでご了承ください。
- ファイルのバックアップ作業を日常業務に組み入れてください。ファイル装置の障害、ファイルアクセス中の停電、誤操作、その他何らかの原因によりファイルの内容を消失することがあります。このような事態に備え、計画的にファイルのバックアップを取っておいてください。
- 弊社製品が故障や誤動作したりプログラムに欠陥があった場合でも、使用されるシステムの安全が十分に確保されるよう、保護・安全回路は外部に設け、人身事故や重大な災害に対する安全対策が十分確保できるようなシステム設計としてください。
- 非常停止回路、インターロック回路などはPLCの外部で構成してください。PLCの故障により、機械の破損や事故の恐れがあります。
- 運転中のプログラム変更、強制出力、RUN、STOPなどは十分安全を確認してから行ってください。誤操作により、機械の破損や事故の恐れがあります。
- このマニュアルでは、安全上の注意事項のランクを潜在危険の重大度によって、「危険」、「警告」、「注意」、「通知」と区分しています。

### 警告表示の定義



：この表示を無視して誤った取り扱いをすると、死亡または重大な傷害を引き起こす危険の存在を示す。



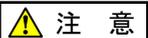
：この表示を無視して誤った取り扱いをすると、死亡または重大な傷害を引き起こすおそれのある危険の存在を示す。



：この表示を無視して誤った取り扱いをすると、軽度の傷害または中程度の傷害を引き起こすおそれのある危険の存在を示す。



：この表示を無視して誤った取り扱いをすると、人身傷害とは関係のない損害を引き起こすおそれのある危険の存在を示す。

なお、 **注意**、 **通知** に記載した事項でも、状況によっては重大な結果に結びつく可能性があります。どれも重要な内容を記載していますので必ず守ってください。

「重大な傷害」、「軽度の傷害または中程度の傷害」、「人身傷害とは関係のない損害」について、具体的な内容を以下に示します。

**重大な傷害**

失明、けが、やけど（高温、低温）、感電傷害、骨折、中毒などで、後遺症が残るものおよび治療のために入院、長期の通院を要するもの

**軽度の傷害または中程度の傷害**

治療のために入院や長期の通院を必要としないけが、やけど、感電傷害など

**人身傷害とは関係のない損害**

周囲の財物の損傷、弊社製品の故障や破損、データの損失など、人身傷害以外の損害

安全上の注意事項は、安全性を確保するための原則に基づいた、弊社製品における各種対策を補完する重要なものです。弊社製品やマニュアルに表示されている安全上の注意事項は、十分に検討されたものですが、それでも、予測を超えた事態が起こることが考えられます。操作するときは指示に従うだけでなく、常に自分自身でも注意するようにしてください。また、弊社製品の安全な運転および保守のために、各種規格、基準に従って安全施策を確立してください。

1.  このマニュアル内の警告表示

このマニュアルの中に書かれている警告とその記載箇所を、以下にまとめて示します。

1. 1 「通知」と表示されているもの

(1章、1-4ページ)

| 通 知  |
|--|
| <ul style="list-style-type: none"><li>● この製品は、PCsがRUN中にプログラム、内部レジスタ値の書き換えができますが、安易に書き換えると設備の破損などの重大な事故を引き起こす場合があります。書き換えに際しては、設備の状態などをチェックし、問題がないことを確認してから実施してください。</li><li>● CPMSデバッグは、タスクのデータをCMUモジュールにローディングした場合、CMU内蔵フラッシュに書き込みます。CMU内蔵フラッシュの同一アドレスに同時に書き込みを行った場合、データが壊れる可能性がありますので、CPMSデバッグとCMUモジュールに対して書き込みを行う他のツール（HI-FLOWシステム、RPDP、NX/Tools-S10Vシステムなど）で同時にデータの送信を行わないでください。</li></ul> |

(2章、2-2ページ)

| 通 知   |
|---|
| CPMSデバッグをインストールする前に、すべてのWindows®上で作動するプログラムを必ず終了してください。ウイルス監視ソフトウェアなどメモリに常駐しているプログラムも必ず終了してください。終了せずにインストールした場合、インストールでエラーが発生する場合があります。その場合は、「2. 2 アンインストール」で一度アンインストールし、すべてのWindows®上で作動するプログラムを終了してから、再度インストールしてください。 |

(2章、2-4ページ)

| 通 知   |
|---|
| <ul style="list-style-type: none"><li>● Windows®でアンインストール中に「共有ファイルを削除しますか？」のウィンドウが表示された場合は、<input type="button" value="いいえ"/> ボタンをクリックして共有ファイルを削除しないでください。</li><li>● CPMSデバッグシステムを再インストールする場合は、必ずアンインストールしてから再インストールしてください。</li></ul> |

このページは白紙です。

このマニュアルは、以下のプログラムプロダクトの説明をしたものです。

<プログラムプロダクト>

S-7895-07 「S10V CPMSデバッガシステム」 (01-03-/A)

S-7895-62 「S10V CPMSデバッガシステム」 (01-00-/A)

## 来歴一覧表

| 改訂No. | 来歴（改訂内容および改訂理由）              | 発行年月   | 備考 |
|-------|------------------------------|--------|----|
| A     | 新規作成                         | 2004.2 |    |
| B     | サポートOSにWindows® 7（32bit）を追加  | 2012.9 |    |
| C     | サポートOSにWindows® 10（32bit）を追加 | 2017.6 |    |

上記追加変更の他に、記述不明瞭な部分、単なる誤字・脱字などについては、お断りなく訂正しました。

# はじめに

S10V CPMSデバッグシステムをお買い上げいただき誠にありがとうございます。

このシステムは、パーソナルコンピュータ上で動作し、PCsで動作するタスクの登録や起動、削除、また各アドレスのビット状態のモニタおよびシステムの状態表示を実行します。

このマニュアルは、S10V CPMSデバッグシステムにおける操作方法について記述してあります。

このマニュアルは、下記バージョンのシステムに対応しています。

| PP型式      | システム名称およびバージョン                          | 対応OS                  |
|-----------|---|-----------------------|
| S-7895-07 | S10V CPMSデバッグシステム For Windows® 01-03-/A | Windows® 2000/XP      |
| S-7895-62 | S10V CPMSデバッグシステム For Windows® 01-00-/A | Windows® 7/10 (32ビット) |

## <関連マニュアル>

S10V ソフトウェアマニュアル CPMS概説&マクロ仕様 (マニュアル番号 SVJ-3-201)

S10V ソフトウェアマニュアル オペレーション RPDP/S10V For Windows®  
(マニュアル番号 SVJ-3-133)

## <商標について>

- Microsoft®, Windows®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。
- Ethernet®は米国Xerox Corp.の登録商標です。
- SuperHは、ルネサス エレクトロニクス株式会社の登録商標です。

## <用語の定義>

PCs : Programmable Controllersの略です。

S10VおよびS10α、S10miniシリーズ等のPLCの総称です。

## <記憶容量の計算値についての注意>

- 2<sup>n</sup>計算値の場合 (メモリ容量・所要量、ファイル容量・所要量など)
  - 1KB (キロバイト) = 1,024バイトの計算値です。
  - 1MB (メガバイト) = 1,048,576バイトの計算値です。
  - 1GB (ギガバイト) = 1,073,741,824バイトの計算値です。
- 10<sup>n</sup>計算値の場合 (ディスク容量など)
  - 1KB (キロバイト) = 1,000バイトの計算値です。
  - 1MB (メガバイト) = 1,000<sup>2</sup>バイトの計算値です。
  - 1GB (ギガバイト) = 1,000<sup>3</sup>バイトの計算値です。

# 目 次

|      |                          |      |
|------|--------------------------|------|
| 1    | ご使用にあたり                  | 1-1  |
| 1.1  | システムの概要                  | 1-2  |
| 1.2  | 必要なハードウェアとソフトウェア         | 1-2  |
| 1.3  | CPMSデバッガとRPDP/S10V混在時の注意 | 1-3  |
| 1.4  | NX/HOST-S10V使用時の注意       | 1-3  |
| 2    | システムインストール               | 2-1  |
| 2.1  | インストール                   | 2-2  |
| 2.2  | アンインストール                 | 2-4  |
| 2.3  | システム立ち上げ                 | 2-5  |
| 2.4  | システム終了                   | 2-7  |
| 3    | コマンド                     | 3-1  |
| 3.1  | コマンド体系                   | 3-2  |
| 3.2  | タスク転送／登録                 | 3-3  |
| 3.3  | タスク削除                    | 3-6  |
| 3.4  | タスク状態表示                  | 3-7  |
| 3.5  | タスクRELEASE               | 3-12 |
| 3.6  | タスクQUEUE                 | 3-13 |
| 3.7  | タスクABORT                 | 3-14 |
| 3.8  | タスク周期起動                  | 3-15 |
| 3.9  | タスク実行抑止                  | 3-19 |
| 3.10 | タスク実行抑止解除                | 3-20 |
| 3.11 | ブレークポイント                 | 3-21 |
| 3.12 | エラーログ                    | 3-29 |
| 3.13 | PCs状態表示                  | 3-33 |
| 3.14 | 時刻設定                     | 3-35 |
| 3.15 | ADTの設定／解除                | 3-36 |
| 3.16 | ADT発生監視                  | 3-39 |
| 3.17 | DHP記録許可／禁止／表示            | 3-40 |
| 3.18 | スタックの初期化                 | 3-42 |
| 3.19 | スタック使用量表示                | 3-44 |
| 3.20 | マトリクスモニタ                 | 3-47 |
| 3.21 | MCS                      | 3-49 |

|      |                           |      |
|------|---------------------------|------|
| 3.22 | 接続PCs変更 .....             | 3-51 |
| 3.23 | タスク環境初期化 .....            | 3-52 |
| 付 録  | .....                     | A-1  |
| 付録A  | プログラムで使用できる名称 .....       | A-2  |
| 付録B  | ライブラリの使用するスタックサイズ一覧 ..... | A-4  |
| 付録C  | タスク作成手順 .....             | A-6  |
| 付録D  | エラーメッセージ .....            | A-26 |

## 目 次

|          |                         |      |
|----------|-------------------------|------|
| 図 2 - 1  | [CPMSデバッグ] 画面           | 2-5  |
| 図 3 - 1  | [タスク転送／登録] 画面           | 3-3  |
| 図 3 - 2  | [ファイルを開く] 画面            | 3-4  |
| 図 3 - 3  | [エラーメッセージ] 画面           | 3-5  |
| 図 3 - 4  | [タスク削除] 画面              | 3-6  |
| 図 3 - 5  | [タスク状態表示] 画面            | 3-7  |
| 図 3 - 6  | [タスク状態表示] (詳細) 画面       | 3-9  |
| 図 3 - 7  | [タスクRELEASE] 画面         | 3-12 |
| 図 3 - 8  | [タスクQUEUE] 画面           | 3-13 |
| 図 3 - 9  | [タスクABORT] 画面           | 3-14 |
| 図 3 - 10 | [タスク周期起動] 画面            | 3-15 |
| 図 3 - 11 | [タスク周期起動] 画面 (1 : 時間指定) | 3-17 |
| 図 3 - 12 | [タスク周期起動] 画面 (2 : 時刻指定) | 3-17 |
| 図 3 - 13 | [タスク実行抑止] 画面            | 3-19 |
| 図 3 - 14 | [タスク実行抑止解除] 画面          | 3-20 |
| 図 3 - 15 | [ブレークポイント] 画面           | 3-21 |
| 図 3 - 16 | [ブレークポイントアドレス] 画面       | 3-22 |
| 図 3 - 17 | [浮動小数点レジスタ] 画面          | 3-27 |
| 図 3 - 18 | [浮動小数点レジスタ] 画面 (FLOAT)  | 3-27 |
| 図 3 - 19 | [エラーログ情報] 画面            | 3-29 |
| 図 3 - 20 | [エラーログ詳細] 画面            | 3-31 |
| 図 3 - 21 | [名前を付けて保存] 画面           | 3-32 |
| 図 3 - 22 | [PCs状態表示] 画面            | 3-33 |
| 図 3 - 23 | [時刻設定] 画面               | 3-35 |
| 図 3 - 24 | [ADTの設定／解除] 画面          | 3-36 |
| 図 3 - 25 | [ADT設定内容] 画面            | 3-37 |
| 図 3 - 26 | ADTが設定できる範囲             | 3-38 |
| 図 3 - 27 | [ADT発生監視] 画面            | 3-39 |
| 図 3 - 28 | ADT発生画面                 | 3-39 |
| 図 3 - 29 | [DHPの記録許可／禁止／表示] 画面     | 3-40 |
| 図 3 - 30 | [DHPトレース情報] 画面          | 3-41 |
| 図 3 - 31 | [スタックの初期化] 画面           | 3-42 |
| 図 3 - 32 | スタックの初期化範囲              | 3-43 |
| 図 3 - 33 | [スタック使用量表示] 画面          | 3-44 |

|        |                       |      |
|--------|-----------------------|------|
| 図 3-34 | [スタック使用量表示] (内容表示) 画面 | 3-45 |
| 図 3-35 | スタック使用量表示が表示する内容      | 3-46 |
| 図 3-36 | [マトリクスモニタ] 画面         | 3-47 |
| 図 3-37 | [入出力設定] 画面            | 3-48 |
| 図 3-38 | [MCS] 画面              | 3-49 |
| 図 3-39 | [通信種類] 画面             | 3-51 |
| 図 3-40 | [タスク環境初期化] 画面         | 3-52 |
| 図 3-41 | CPMSデバッガのデフォルトの環境     | 3-54 |
| 図 C-1  | タスク構成                 | A-17 |
| 図 C-2  | マップ構成                 | A-21 |
| 図 C-3  | 関数呼び出しの関係とスタック使用量     | A-23 |

# 表 目 次

|       |                             |      |
|-------|-----------------------------|------|
| 表 C-1 | 設定環境変数 .....                | A-9  |
| 表 C-2 | 浮動小数点数の扱い制御オプション .....      | A-10 |
| 表 C-3 | 浮動小数点数の扱いと対応する標準ライブラリ ..... | A-10 |
| 表 C-4 | ライブラリ内容 .....               | A-11 |
| 表 C-5 | S10V CPMSデバグ提供ファイル .....    | A-12 |
| 表 C-6 | optlnkの指定形式 .....           | A-16 |
| 表 C-7 | スタックサイズの計算例 .....           | A-23 |

# 1 ご使用にあたり

## 1 ご使用にあたり

このマニュアルは、Windows®パソコンプログラミングのユーザを対象としています。

### 1.1 システムの概要

S10V CPMSデバッグシステムFor Windows®（以下、CPMSデバッグと略します。）は、一般的なWindows®上で作動するアプリケーションと等価なオペレーションによりS10V用タスクの登録や起動、モニタを行います。

### 1.2 必要なハードウェアとソフトウェア

CPMSデバッグを使用するためには、以下のハードウェアおよびソフトウェアが必要です。

- Pentium300MHz以上のCPUを搭載したパーソナルコンピュータ（以下、パソコンと略します）本体
- 1GHz以上のCPUを搭載したパソコン本体（Windows® 7（32bit）、Windows® 10（32bit）使用時）
- 800×600ドット（SVGA）以上の解像度を持つディスプレイ
- Microsoft® Windows® 2000 operating system、Microsoft® Windows® XP operating system、Microsoft® Windows® 7（32bit） operating system、またはMicrosoft® Windows® 10（32bit） operating system
- 64MB以上のRAM（Windows® 2000使用時）
- 128MB以上のRAM（Windows® XP使用時）
- 1GB以上のRAM（Windows® 7（32bit）、Windows® 10（32bit）使用時）
- 10MB以上の空きハードディスク容量
- S10VシリーズLPU、CMU
- S10Vシリーズ電源、バックボード
- パソコンとCMUモジュール、またはET.NETモジュール（LQE720）間の接続ケーブル（RJ-45モジュラコネクタを持つ100BASE-TXツイストペアクロスケーブル）
- 必要に応じたりモートI/Oステーション、電源、バックボード、カードおよび配線ケーブル

### 1. 3 CPMSデバッガとRPDP/S10V混在時の注意

CPMSデバッガは、RPDP/S10Vで作成したタスクの状態参照やabort等に使用します。プログラムの作成・修正はRPDP/S10Vで行ってください。

CPMSデバッガでタスクを登録すると、RPDP/S10Vの開発環境と不整合となります。そのため、RPDP/S10Vでsvrplを実行すると、CPMSデバッガで登録したタスクは無効となります。

CPMSデバッガで新規登録したタスクは、RPDP/S10Vで管理できませんので注意してください。

### 1. 4 NX/HOST-S10V使用時の注意

従来のNX/HOST-S10 (S10mini) は、NX/HOST-S10のシステムファイルをPCsに送信した場合、CPMSデバッガで作成・登録したユーザタスクがPCs上に存在しても、エリアが重複していなければ、ユーザタスクは削除されません。しかし、S10VのNX/HOST-S10Vでは、NX/HOST-S10VのシステムファイルをPCsに送信した場合、CPMSデバッガで作成・登録したユーザタスクがPCs上に存在すると、エリアが重複していなくても、ユーザタスクは削除されますので注意してください。

### 通 知

- この製品は、PCsがRUN中にプログラム、内部レジスタ値の書き換えができますが、安易に書き換えると設備の破損などの重大な事故を引き起こす場合があります。書き換えに際しては、設備の状態などをチェックし、問題がないことを確認してから実施してください。
- CPMSデバッグは、タスクのデータをCMUモジュールにローディングした場合、CMU内蔵フラッシュに書き込みます。CMU内蔵フラッシュの同一アドレスに同時に書き込みを行った場合、データが壊れる可能性がありますので、CPMSデバッグとCMUモジュールに対して書き込みを行う他のツール（HI-FLOWシステム、RPDP、NX/Tools-S10Vシステムなど）で同時にデータの送信を行わないでください。

この製品を使用するユーザは、Windows®環境およびユーザインタフェースについての知識が必要です。このシステムは、Windows®標準に従っています。このマニュアルは、基本となるWindows®の使用法を習得しているユーザを対象にして記述されています。

- サスペンド機能を持つパソコンを使用する場合は、サスペンド機能をOFFにしてください。このシステムを実行中にサスペンド機能が動作すると正常に動作しないことがあります。
- RAMの空きメモリ容量が不足すると、アプリケーションエラーが発生する場合があります。アプリケーションエラーが発生する場合は、空きメモリ容量をチェックし、少ない場合はRAMを増設してください。

## 2 システムインストール

## 2 システムインストール

---

### 2.1 インストール

CPMSデバッガをインストールするには、CPMSデバッガCDのDISK1フォルダに格納されている“setup.exe”をダブルクリックし、セットアッププログラムを実行してください。

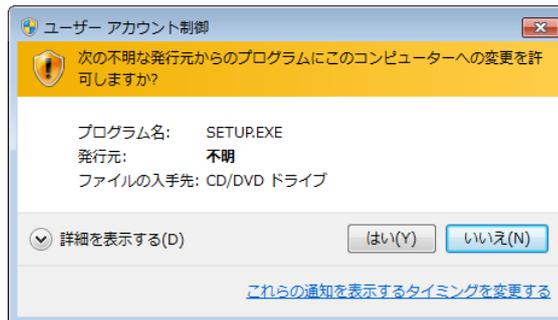
インストール後、インストールしたプログラムの画面は表示されませんので、必要に応じてデスクトップにショートカットを貼り付けてください。

#### 通 知

CPMSデバッガをインストールする前に、すべてのWindows®上で作動するプログラムを必ず終了してください。ウイルス監視ソフトウェアなどメモリに常駐しているプログラムも必ず終了してください。終了せずにインストールした場合、インストールでエラーが発生する場合があります。その場合は、「2.2 アンインストール」で一度アンインストールし、すべてのWindows®上で作動するプログラムを終了してから、再度インストールしてください。

### <Windows® 7 (32bit) 、Windows® 10 (32bit) でのインストール時の留意事項>

Windows® 7 (32bit) またはWindows® 10 (32bit) 搭載のパソコンへCPMSデバッグシステムをインストールする場合は、パソコンの初期状態から最初に作成した管理者アカウントでWindows®にログオンし、CPMSデバッグシステムCDのDISK1フォルダに格納されている“setup.exe”をダブルクリックします。“setup.exe”を起動すると、以下のダイアログボックスが表示される場合がありますので、 はい ボタンをクリックしてセットアッププログラムを実行してください。



このシステムはユーザー別アプリケーションには対応していないため、必ず管理者アカウントでログオンしてからインストールしてください。

標準アカウントからユーザーアカウント制御(\*)を使用してインストールしたり、標準アカウントからユーザーアカウント制御を使用して作成した管理者アカウントでログオンしてからは、正しくインストールされない場合があります。

パソコンの初期状態から最初に作成した管理者アカウントでログオンしてからインストールしてください。

インストールしたユーザーアカウントとは別のユーザーアカウントでログオンした際に、プログラムメニューの中にインストールしたプログラムが表示されない場合は、パソコンの初期状態から最初に作成した管理者アカウントでログオンし直し、プログラムを一度アンインストールしてから、再度インストールしてください。

また、新規にアカウントを作成する場合は、ユーザーアカウント制御を使用せずに管理者アカウントでログオンしてください。

(\*) ユーザーアカウント制御は、標準アカウントに一時的に管理者権限を与えることができる機能です。

再インストールする際に、読み取り専用ファイルの検出メッセージが表示される場合は、 はい ボタンをクリックして、上書きしてください。

### 2.2 アンインストール

CPMSデバッガシステムのバージョンアップ時には、以下の手順でアンインストールしてください。

#### (1) Windows® 2000からのアンインストール

Windows®の [コントロールパネル] を開いてください。 [アプリケーションの追加と削除] をダブルクリックし、 [プログラムの変更と削除] タブで “S10V CPMSデバッガシステム” を選択し、  ボタンをクリックしてください。

[ファイル削除の確認] 画面が表示されますので、  ボタンをクリックしてください。

#### (2) Windows® XPからのアンインストール

Windows®の [コントロールパネル] を開いてください。 [プログラムの追加と削除] をダブルクリックし、 [プログラムの変更と削除] タブで “S10V CPMSデバッガシステム” を選択し、  ボタンをクリックしてください。

[ファイル削除の確認] 画面が表示されますので、  ボタンをクリックしてください。

#### (3) Windows® 7 (32bit) 、Windows® 10 (32bit) からのアンインストール

Windows®の [コントロールパネル] を開いてください。 [プログラムのアンインストール] をクリックし、 “S10V CPMSデバッガシステム” を選択し、  ボタンをクリックしてください。

[ファイル削除の確認] 画面が表示されますので、  ボタンをクリックしてください。

### 通 知

- Windows®でアンインストール中に [共有ファイルを削除しますか?] のウィンドウが表示された場合は、  ボタンをクリックして共有ファイルを削除しないでください。
- CPMSデバッガシステムを再インストールする場合は、必ずアンインストールしてから再インストールしてください。

## 2.3 システム立ち上げ

CPMSデバッグの立ち上げ操作手順を示します。Windows®画面の **スタート** ボタンから [Hitachi S10V] - [S10V CPMSデバッグシステム] をクリックし、S10V CPMSデバッグを起動すると、下記の画面が表示されます。各機能を実行する場合、各ボタンをクリックしてください。



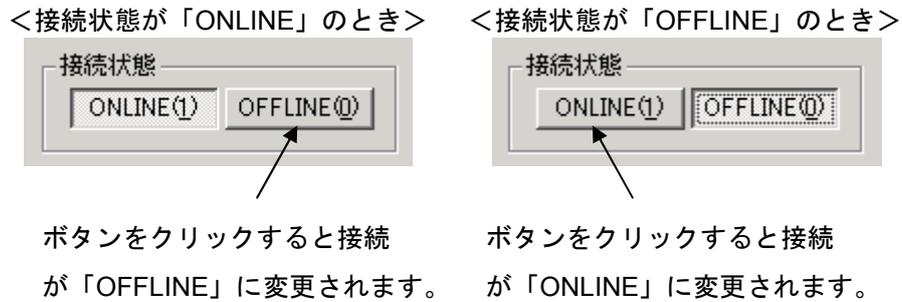
図 2-1 [CPMSデバッグ] 画面

- (1) [イーサネット] ボックス  
接続先PCsのIPアドレスが表示されます。
- (2) **接続PCs変更** ボタン  
PCsとパソコンの通信種類を設定するPCs接続変更画面が表示されます。  
設定終了後、PCsの接続状態が「ONLINE」になります。

## 2 システムインストール

### (3) [接続状態] グループ

現在のPCsとの接続状態が表示されます。この接続状態を切り替える場合は、「ONLINE」または「OFFLINE」のプッシュボタンをクリックして切り替えてください。



### (4) ヘルプ ボタン

このボタンをクリックまたはキーボードの [F1] キーを押したとき、CPMSデバッガのオンラインヘルプが表示されます。

### (5) バージョン情報

CPMSデバッガ画面の左隅のアイコン、またはタイトルバーを右クリックすると、以下の [バージョン情報] 画面が表示されます。この画面を閉じる場合は、**OK** ボタンをクリックしてください。

(PP型式 : S-7895-07の場合)



(PP型式 : S-7895-62の場合)



## 2.4 システム終了

[CPMSデバッガ] 画面 (図 2-1) で、 または  ボタンをクリックしてください。

このページは白紙です。

# 3 コマンド

### 3 コマンド

#### 3.1 コマンド体系

CPMSデバッガのコマンド体系を以下に示します。

コマンドの概要を3.2節以降に示します。コマンドの詳細は、次ページ以降またはヘルプを参照してください。

|      |   |               |
|------|---|---------------|
| コマンド | — | タスク転送／登録      |
|      | — | タスク削除         |
|      | — | タスク状態表示       |
|      | — | タスクRELEASE    |
|      | — | タスクQUEUE      |
|      | — | タスクABORT      |
|      | — | タスク周期起動       |
|      | — | タスク実行抑止       |
|      | — | タスク実行抑止解除     |
|      | — | ブレークポイント      |
|      | — | エラーログ         |
|      | — | PCs状態表示       |
|      | — | 時刻設定          |
|      | — | ADTの設定／解除     |
|      | — | ADT発生監視       |
|      | — | DHP記録許可／禁止／表示 |
|      | — | スタックの初期化      |
|      | — | スタック使用量表示     |
|      | — | マトリクスモニタ      |
|      | — | MCS           |
|      | — | 接続PCs変更       |
|      | — | タスク環境初期化      |

## 3. 2 タスク転送／登録

機能：ユーザが作成したタスクをPCsに転送および登録します。

操作：[CPMSデバッガ]画面より、**タスク転送／登録** ボタンをクリックして表示します。

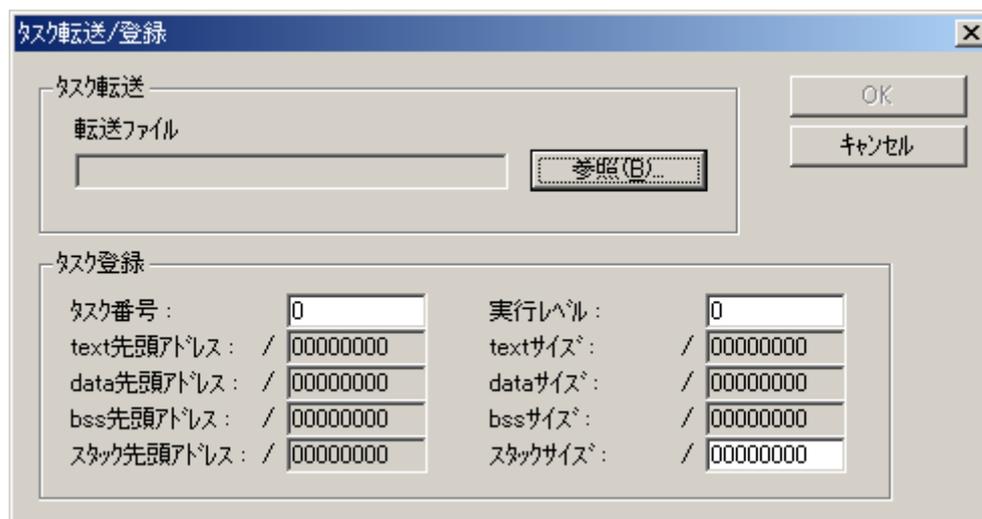


図 3 - 1 [タスク転送／登録]画面

### (1) [転送ファイル]ボックス

PCsに転送するタスクファイルが表示されます。タスクファイルの指定は **参照** ボタンから行ってください。

### 3 コマンド

(2) **参照** ボタン

このボタンをクリックしたとき、以下の画面が表示されます。この画面でPCsに転送するタスクファイルを指定してください。

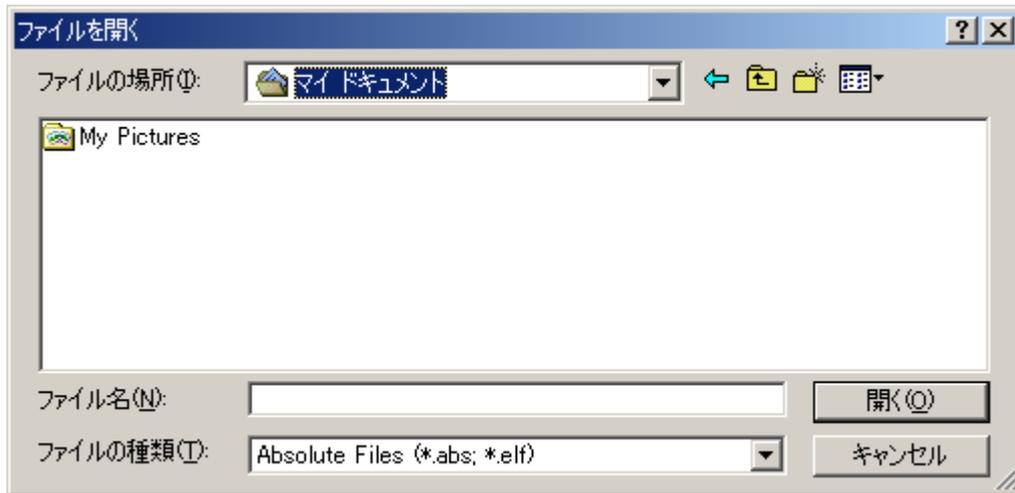


図 3-2 「ファイルを開く」画面

(3) [タスク番号] ボックス

登録するタスクのタスク番号を指定してください。ユーザタスクを登録する場合は1～224、システムタスクを登録する場合は225～229の範囲で指定してください。

(4) [実行レベル] ボックス

登録するタスクの初期起動時の実行レベルを指定してください。ユーザタスクを登録する場合は4～27、システムタスクを登録する場合は0～31の範囲で指定してください。

(5) [text先頭アドレス] ボックス

[参照] ボタンで指定したタスクファイルのtext部が転送される先頭アドレスが表示されます。

(6) [textサイズ] ボックス

[参照] ボタンで指定したタスクファイルのtext部のサイズがバイト長で表示されます。

(7) [data先頭アドレス] ボックス

[参照] ボタンで指定したタスクファイルのdata部が転送される先頭アドレスが表示されます。

(8) [dataサイズ] ボックス

[参照] ボタンで指定したタスクファイルのdata部のサイズがバイト長で表示されます。

- (9) [bss先頭アドレス] ボックス  
[参照] ボタンで指定したタスクファイルが使用するbss部の先頭アドレスが表示されます。
- (10) [bssサイズ] ボックス  
[参照] ボタンで指定したタスクファイルが使用するbss部のサイズがバイト長で表示されます。
- (11) [スタック先頭アドレス] ボックス  
[参照] ボタンで指定したタスクファイルが使用するスタック部の先頭アドレスが表示されます。
- (12) [スタックサイズ] ボックス  
登録するタスクが使用するスタック部のサイズをバイト長で指定してください。  
スタックサイズは、0x1000の倍数で、0x00001000～0x00800000まで指定してください。0x1000の倍数以外を指定した場合、0x1000の倍数に切り上げて処理します。
- (13)  ボタン  
ユーザが指定した内容でタスクの転送および登録が行われます。タスクの転送および登録が正常に終了した後、[CPMSデバッガ] 画面に戻ります。
- (14)  ボタン  
タスクの転送および登録を行わず、[CPMSデバッガ] 画面に戻ります。

#### <注意事項>

CPMSデバッガは、ELF形式のタスクファイルのみサポートしています。そのため  ボタンでELF形式以外のファイルを指定した場合、以下のエラーメッセージが表示されます。



図 3-3 [エラーメッセージ] 画面

また、タスクの先頭アドレスはページ境界上に存在する必要があります。そのため  ボタンでタスク先頭アドレス (text先頭アドレス) がページ境界をまたがっているファイルを指定した場合、タスクファイルを転送できないため  ボタンがシャドウ表示になります。

#### 3.3 タスク削除

機能：タスクを削除します。

操作：[CPMSデバッグ]画面より、**タスク削除** ボタンをクリックして表示します。



図 3-4 [タスク削除]画面

- (1) [タスク番号] ボックス  
削除するタスク番号を入力してください。
- (2) **OK** ボタン  
指定したタスク番号が削除されます。削除後、[CPMSデバッグ]画面に戻ります。
- (3) **キャンセル** ボタン  
タスクの削除を行わず、[CPMSデバッグ]画面に戻ります。

### 3. 4 タスク状態表示

< [タスク状態表示] 画面 >

機能：指定したタスク番号の状態を表示します。

操作：[CPMSデバッガ] 画面より、タスク状態表示 ボタンをクリックして表示します。

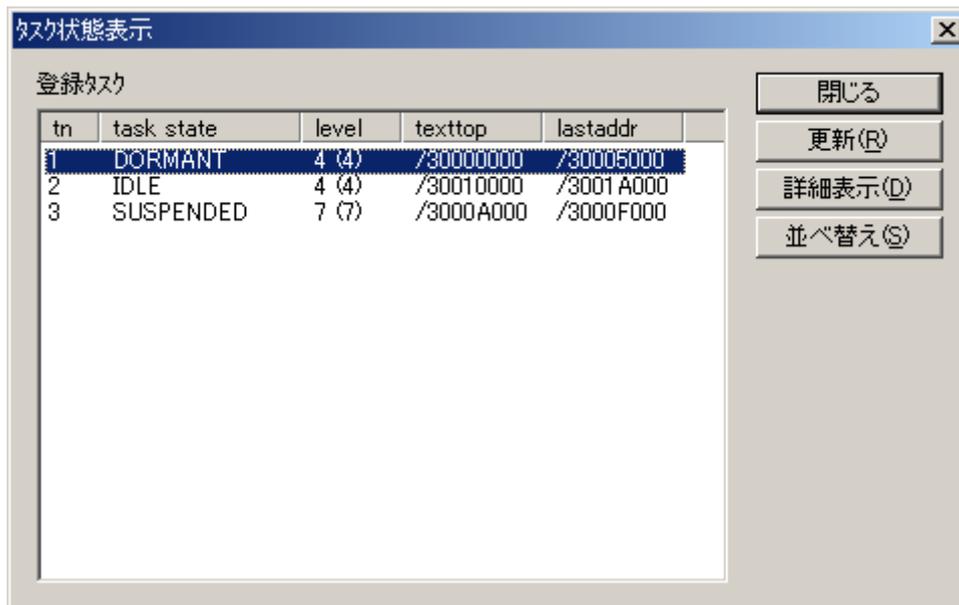


図 3-5 [タスク状態表示] 画面

### 3 コマンド

(1) [登録タスク] ボックス

登録されているタスクの情報が一覧で表示されます。

表示される内容は以下のとおりです。

| No. | 表示         | 内容              |
|-----|------------|-----------------|
| 1   | tn         | タスク番号           |
| 2   | task state | タスク状態 (*)       |
| 3   | level      | 実行レベル (初期実行レベル) |
| 4   | texttop    | タスク先頭アドレス       |
| 5   | lastaddr   | タスク最終アドレス       |

(\*) タスク状態

| No. | 表示        | 内容           |
|-----|-----------|--------------|
| 1   | DORMANT   | 起動抑止状態       |
| 2   | IDLE      | 起動待ち状態       |
| 3   | READY     | 実行中または実行待ち状態 |
| 4   | WAIT      | イベント待ち状態     |
| 5   | SUSPENDED | 実行抑止状態       |

(2)  ボタン

[タスク状態表示] 画面を終了し、[CPMSデバッグ] 画面に戻ります。

(3)  ボタン

[登録タスク] ボックスに表示されている内容を最新の状態で再表示します。

(4)  ボタン

[登録タスク] ボックスで指定したタスク番号のタスク情報の詳細を表示する [タスク状態表示] 画面が表示されます。

(5)  ボタン

[登録タスク] ボックスに表示されている内容をタスク先頭アドレス順に並び替えます。このボタンをクリックするごとに表示順番が、昇順、降順に入れ替わります。

< [タスク状態表示] (詳細) 画面 >

機能：タスクの詳細状態を表示します。

操作：[タスク状態表示] 画面 (図 3-5) より、**詳細表示** ボタンをクリックして表示します。



図 3-6 [タスク状態表示] (詳細) 画面

(1) [タスク番号] ボックス

現在表示されているタスク番号が10進表示されます。

指定したタスク番号の状態を表示する場合は、このボックスにタスク番号を入力し、[Enter] キーを押すか、**更新** ボタンをクリックしてください。

タスク番号の表示可能範囲は1~229です。

(2) **次タスク** ボタン

指定したタスクの次に登録されているタスクの最新の詳細情報が表示されます。

未登録のタスク状態表示はスキップされます。

(3) **前タスク** ボタン

指定したタスクの前に登録されているタスクの最新の詳細情報が表示されます。

未登録のタスク状態表示はスキップされます。

### 3 コマンド

(4) タスク起動要因

指定したタスクの起動要因が表示されます。

指定したタスクが登録されていない場合、“/00000000”が表示されます。

(5) 実行レベル

指定したタスクの実行レベル（カッコ内は実行レベルの初期値）が表示されます。

指定したタスクが登録されていない場合、“0 (0)”が表示されます。

(6) TCB先頭アドレス

指定したタスクのTCB情報の先頭アドレス（物理アドレス）が表示されます。

(7) タスク先頭アドレス

指定したタスクの先頭アドレスが表示されます。指定したタスクが登録されていない場合、“/00000000”が表示されます。

(8) タスク状態

指定したタスクの状態が表示されます。

表示される内容は以下のとおりです。

| No. | 表示           | 内容           |
|-----|--------------|--------------|
| 1   | DORMANT      | 起動抑止状態       |
| 2   | IDLE         | 起動待ち状態       |
| 3   | READY        | 実行中または実行待ち状態 |
| 4   | WAIT         | イベント待ち状態     |
| 5   | SUSPENDED    | 実行抑止状態       |
| 6   | NON-EXISTENT | 未登録          |

(9) [タスク状態の詳細] グループ

指定したタスク状態の詳細情報（ステータスビットの情報）がチェックボックスのON/OFFで表示されます。

このパラメータは、複数のステータスが同時にONになる場合があります。

(10) [アドレス情報] グループ

指定したタスクが確保しているアドレスとサイズが表示されます。また、指定タスクが登録されていない場合、アドレスおよびサイズには“/00000000”が表示されます。

- (11)  ボタン

タスク状態の表示を終了し、[CPMSデバッグ] 画面に戻ります。

- (12)  ボタン

[タスク番号] ボックスで入力したタスク番号の最新状態が表示されます。

## 3.5 タスクRELEASE

機能：タスクの起動抑止を解除します。

操作：[CPMSデバッガ]画面より、**タスクRELEASE** ボタンをクリックして表示します。



図 3-7 [タスクRELEASE] 画面

(1) [タスク番号] ボックス

起動抑止を解除するタスク番号を入力してください。[範囲指定] チェックボックスがONのとき、起動抑止を解除するタスク番号を範囲指定することができます。

(2) [範囲指定] チェックボックス

タスク番号の範囲指定を許可するかを選択してください。範囲指定時、エラーが発生してもエラーメッセージは表示されません。実行結果は[タスク状態表示]画面で指定範囲のタスク状態が「READY」になっているかを確認してください。

(3) **OK** ボタン

指定したタスク番号の起動抑止が解除されます。[範囲指定] チェックボックスがONの場合は、指定した範囲のタスクの起動抑止が解除されます。

起動抑止解除後、[CPMSデバッガ]画面に戻ります。

(4) **キャンセル** ボタン

タスクの起動抑止解除を行わず、[CPMSデバッガ]画面に戻ります。

## 3.6 タスクQUEUE

機能：タスクの起動要求をします。

操作：[CPMSデバッガ]画面より、**タスクQUEUE** ボタンをクリックして表示します。

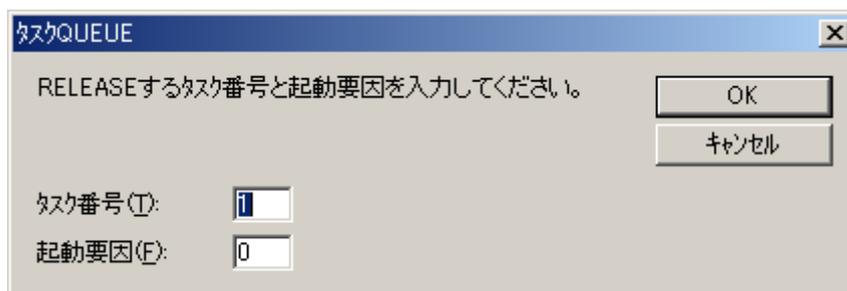


図 3-8 [タスクQUEUE] 画面

- (1) [タスク番号] ボックス  
起動要求するタスク番号を入力してください。
- (2) [起動要因] ボックス  
起動要求するタスクの起動要因を入力してください。
- (3) **OK** ボタン  
指定したタスク番号が起動要求されます。起動要求後、[CPMSデバッガ]画面に戻ります。
- (4) **キャンセル** ボタン  
タスクの起動要求を行わず、[CPMSデバッガ]画面に戻ります。

## 3.7 タスクABORT

機能：タスクの起動を抑止します。

操作：[CPMSデバッガ]画面より、**タスクABORT** ボタンをクリックして表示します。

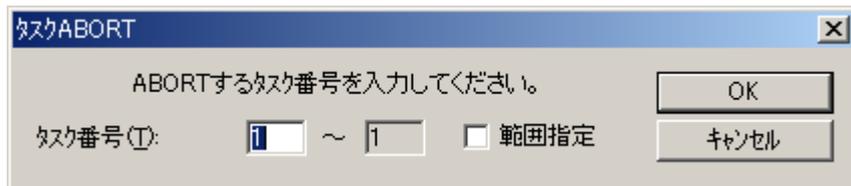


図3-9 [タスクABORT]画面

(1) [タスク番号] ボックス

起動を抑止するタスク番号を入力してください。[範囲指定]チェックボックスがONのとき、起動を抑止するタスク番号を範囲指定することができます。

(2) [範囲指定] チェックボックス

タスク番号の範囲指定を許可するかを選択してください。範囲指定時、エラーが発生してもエラーメッセージは表示されません。実行結果は[タスク状態表示]画面で指定範囲のタスク状態が「DORMANT」になっているかを確認してください。

(3) **OK** ボタン

指定したタスク番号の起動が抑止されます。[範囲指定]チェックボックスがONの場合は、指定した範囲のタスクの起動を抑止されます。

起動抑止後、[CPMSデバッガ]画面に戻ります。

(4) **キャンセル** ボタン

タスクの起動抑止を行わず、[CPMSデバッガ]画面に戻ります。

### 3. 8 タスク周期起動

機能：現在設定されているタスクの周期起動設定の一覧と操作を行います。

操作：[CPMSデバッガ]画面より、タスク周期起動 ボタンをクリックして表示します。

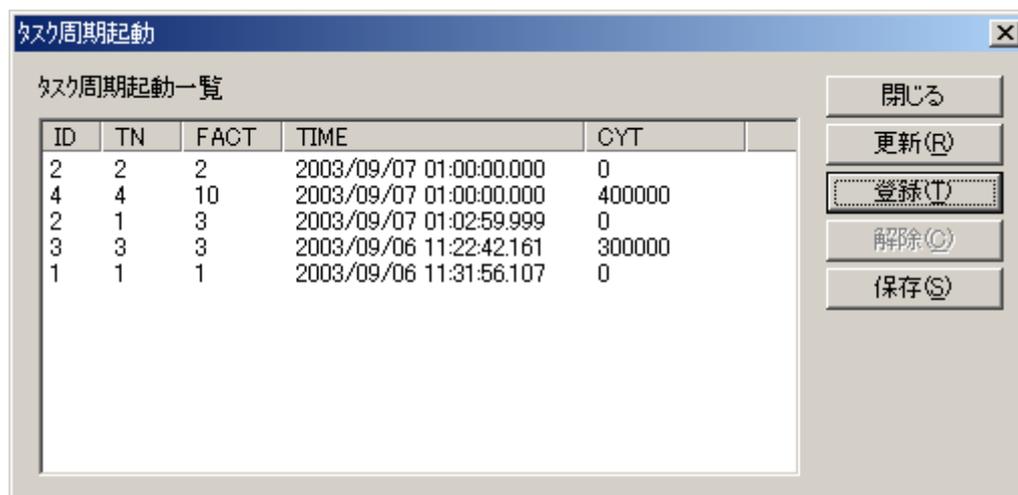


図 3-10 [タスク周期起動]画面

(1) [タスク周期起動一覧]ボックス

現在設定されている周期起動の一覧が表示されます。

表示される内容は、RPDP/S10Vのshtコマンドと同様です。

表示される内容は以下のとおりです。

| No. | 項目   | 内容                     |
|-----|------|------------------------|
| 1   | ID   | タイマ種別 (*)              |
| 2   | TN   | タスク番号                  |
| 3   | FACT | 起動要因                   |
| 4   | TIME | 起動時刻 (年/月/日 時:分:秒.ミリ秒) |
| 5   | CYT  | 周期時間 (ミリ秒)             |

### 3 コマンド

---

(\*) タイマ種別

| ID | 内容               |
|----|------------------|
| 1  | 時間起動 (timer)     |
| 2  | 時刻起動 (timer)     |
| 3  | 時間指定周期起動 (timer) |
| 4  | 時刻指定周期起動 (timer) |

(2)  ボタン

タスク周期起動表示を終了し、[CPMSデバッグ] 画面に戻ります。

(3)  ボタン

[タスク周期起動一覧] ボックスに表示されている内容が最新の情報に更新されます。

(4)  ボタン

タスク周期起動の設定を行う [タスク周期起動] 画面が表示されます。

(5)  ボタン

[タスク周期起動一覧] ボックスで選択しているタスク周期起動設定が解除されます。

処理内容はRPDP/S10Vのctコマンドと同様です。

<注意> 起動要因が0の周期起動タスクを解除すると、同じタスク番号の周期起動タスクも解除されます。

(6)  ボタン

[タスク周期起動一覧] ボックスに表示している内容をテキスト形式で保存することができます。

< [タスク周期起動] 画面 >

機能：タスクの周期起動を設定します。

操作：[タスク周期起動] 画面（図 3-10）より、登録 ボタンをクリックして表示します。

図 3-11 [タスク周期起動] 画面（1：時間指定）

図 3-12 [タスク周期起動] 画面（2：時刻指定）

- (1) タスク番号  
周期起動をかけるタスク番号を入力してください。
- (2) 初期起動時間、初期起動時刻  
タイマイベントの初期起動時間、初期起動時刻を指定してください。  
初期起動時間は、現時刻から起動までの相対時間で1～86400000の範囲で指定できます。  
初期起動時刻は、初回のタイマイベントの時刻として、00:00:0.000～23:59:59.999の範囲で指定できます。
- (3) 周期時間  
周期的にイベントを発生させる場合の周期時間を指定してください。  
周期時間は、ミリ秒単位で1～86400000を指定できます。

### 3 コマンド

---

(4) 起動要因

タスクの起動要因を指定してください。

(5)  ボタン

指定した内容でタスク周期起動を設定します。設定後、[タスク周期起動] 画面に戻ります。

(6)  ボタン

タスク周期起動の設定を行わず、[タスク周期起動] 画面に戻ります。

### 3.9 タスク実行抑止

機能：タスクの実行を抑止します。

操作：[CPMSデバッガ]画面より、タスク実行抑止 ボタンをクリックして表示します。

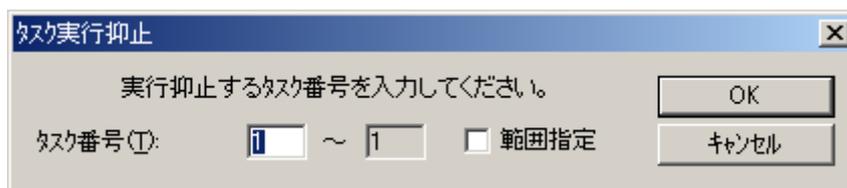


図 3-13 [タスク実行抑止]画面

(1) [タスク番号] ボックス

実行を抑止するタスク番号を入力してください。[範囲指定] チェックボックスがONのとき、実行を抑止するタスク番号を範囲指定することができます。

(2) [範囲指定] チェックボックス

タスク番号の範囲指定を許可するかを選択してください。範囲指定時、エラーが発生してもエラーメッセージは表示されません。実行結果は[タスク状態表示]画面で指定範囲の[タスク状態の詳細]のグループボックス内の[SUSPによる実効抑止中]のチェックボックスがONになっているかを確認してください。

(3) OK ボタン

指定したタスク番号の実行が抑止されます。[範囲指定] チェックボックスがONの場合は、指定した範囲のタスクの実行を抑止します。

実行抑止後、[CPMSデバッガ]画面に戻ります。

(4) キャンセル ボタン

タスクの実行抑止を行わず、[CPMSデバッガ]画面に戻ります。

## 3.10 タスク実行抑止解除

機能：タスクの実行抑止状態を解除します。

操作：[CPMSデバッガ]画面より、**タスク実行抑止解除** ボタンをクリックして表示します。



図 3-14 [タスク実行抑止解除]画面

(1) [タスク番号] ボックス

実行抑止を解除するタスク番号を入力してください。[範囲指定] チェックボックスがONのとき、実行抑止を解除するタスク番号を範囲指定することができます。

(2) [範囲指定] チェックボックス

タスク番号の範囲指定を許可するかを選択してください。範囲指定時、エラーが発生してもエラーメッセージは表示されません。実行結果は[タスク状態表示]画面で指定範囲の[タスク状態の詳細]のグループボックス内の[SUSPによる実効抑止中]のチェックボックスがOFFになっているかを確認してください。

(3) **OK** ボタン

指定したタスク番号の実行抑止が解除されます。[範囲指定] チェックボックスがONの場合は、指定した範囲のタスクの実行抑止を解除します。

実行抑止を解除後、[CPMSデバッガ]画面に戻ります。

(4) **キャンセル** ボタン

タスクの実行抑止の解除を行わず、[CPMSデバッガ]画面に戻ります。

### 3.11 ブレークポイント

< [ブレークポイント] 画面 >

機能：ブレークポイントを設定および解除します。

操作：[CPMSデバッガ] 画面より、ブレークポイント ボタンをクリックして表示します。

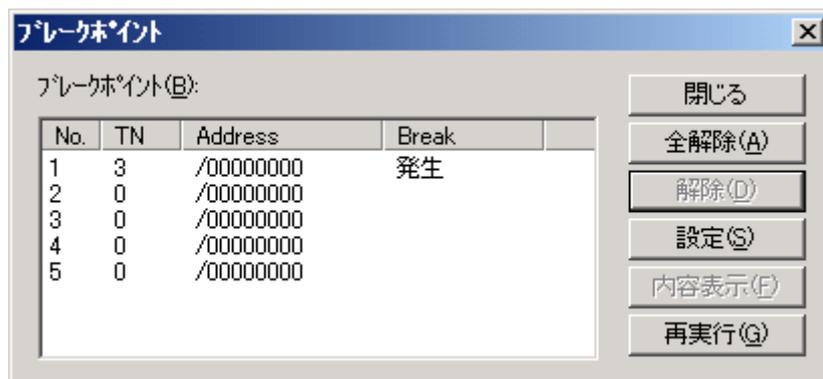


図 3-15 [ブレークポイント] 画面

(1) [ブレークポイント] ボックス

現在のブレーク状態が表示されます。ブレークポイントは5つまで登録できます。表示される内容は以下のとおりです。

| No. | 項目      | 内容                          |
|-----|---------|-----------------------------|
| 1   | TN      | タスク番号                       |
| 2   | Address | ブレークポイントアドレス (プログラム内相対アドレス) |
| 3   | Break   | ブレーク発生状態                    |

(2) 閉じる ボタン

ブレークポイントの設定を終了し、[CPMSデバッガ] 画面に戻ります。

(3) 全解除 ボタン

現在設定されているブレークポイントがすべて解除されます。ただし、ブレーク発生中のタスクは、解除されません。

(4) 解除 ボタン

[ブレークポイント] ボックスで指定したNo.のブレークポイントが解除されます。

### 3 コマンド

- (5) **設定** ボタン  
ブレークポイントの設定を行う [ブレークポイントアドレス] 画面が表示されます。
- (6) **内容表示** ボタン  
レジスタ内容の表示・設定を行うレジスタ内容画面が表示されます。  
CPU (PCs) 内のプログラムがブレーク状態のときのみ使用できます。
- (7) **再実行** ボタン  
ブレーク中状態のタスクが再起動されます。  
CPU (PCs) 内のプログラムがブレーク状態のときのみ使用できます。

< [ブレークポイントアドレス] 画面 >

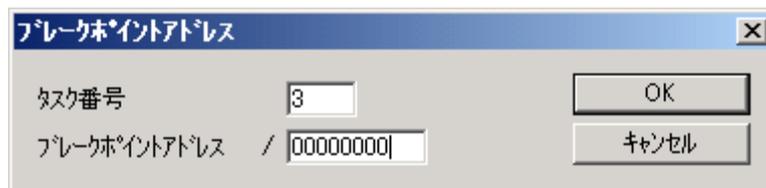


図 3-16 [ブレークポイントアドレス] 画面

- (1) [タスク番号] ボックス  
ブレークポイントを設定するタスク番号を指定してください。
- (2) [ブレークポイントアドレス] ボックス  
ブレークポイントアドレス (プログラム内相対アドレス) を指定してください。
- (3) **OK** ボタン  
指定したアドレスでブレークポイントが設定されます。設定後、[ブレークポイント] 画面に戻ります。
- (4) **キャンセル** ボタン  
ブレークポイントの設定を行わず、[ブレークポイント] 画面に戻ります。

## &lt;ブレークポイント設定手順&gt;

下記のtask1というタスクプログラムを生成した場合のポイント (A) にブレークポイントを設定する場合の手順を示します。

## &lt;task1.c&gt;

```
int b1;
int d1 = 11;
static int b2 ;
static int d2 = 101 ;

main()
{
    unsigned short *fw010;
    unsigned short *dw000;

    static int b3 ;
    static int d3 = 1001 ;
    int s1;
    int s2 = 21 ;

    fw010 = (unsigned short *)0xE2020;
    dw000 = (unsigned short *)0x61000;

    *fw010 = *fw010 + *dw000;
    exit(0);
}
```

← (A)

shcコマンドでソースコンパイル時に、“-listfile -show=source,object” オプションを指定することで、アセンブラソースにC言語のソースファイルを挿入することができます。task1.cのソース内にブレークポイントを設定するため、task1.lstを参照してください。

ブレークポイントを設定するC言語のソースは (1) なので、その直後のアセンブラ命令 (2) がC言語のソースに対応しています。該当する命令のソース内 (task1.c) でのオフセットは、0x0000000Aとなります。

### 3 コマンド

<prog1.lst>

```

***** OBJECT LISTING *****
FILE NAME: task1.c

SCT OFFSET   CODE      C LABEL      INSTRUCTION OPERAND      COMMENT
task1.c      1          int b1;
task1.c      2          int d1 = 11;
task1.c      3          static int b2 ;
task1.c      4          static int d2 = 101 ;
task1.c      5
task1.c      6          main()
P 00000000   _main:                                ; function: main
                                           ; frame size=0
task1.c      7          {
task1.c      8          unsigned short *fw010;
task1.c      9          unsigned short *dw000;
task1.c     10
task1.c     11          static int b3 ;
task1.c     12          static int d3 = 1001 ;
task1.c     13          int s1;
task1.c     14          int s2 = 21 ;
task1.c     15
task1.c     16          fw010 = (unsigned short *)0xE2020;
00000000 D504          MOV.L      L11+2,R5      ; H'000E2020
00000002 E400          MOV        #0,R4        ; H'00000000
task1.c     17          dw000 = (unsigned short *)0x61000;
task1.c     18
task1.c     19          *fw010 = *fw010 + *dw000; (1)
00000004 D204          MOV.L      L11+6,R2      ; H'00061000
00000006 6621          MOV.W      @R2,R6
00000008 6251          MOV.W      @R5,R2
0000000A 362C          ADD        R2,R6 (2)
task1.c     20          exit(0);
0000000C D203          MOV.L      L11+10,R2     ; _exit
0000000E 422B          JMP        @R2
00000010 2561          MOV.W      R6,@R5
00000012          L11:
00000012 00000002          .RES.W      1
00000014 000E2020          .DATA.L     H'000E2020
00000018 00061000          .DATA.L     H'00061000
0000001C <00000000>          .DATA.L     _exit
task1.c     21
task1.c     22          }

```

optlnkコマンドで実行ファイル生成時に、“-list” オプションを設定することで、マップファイルを生成します。

マップファイルのSECTION=Pを確認すると、task1.objは、0x30000000から始まることが解ります。

ブレークポイントを設定するソース内相対アドレスは0x0000000Aなので、ブレークポイントを設定するアドレスは0x3000000A (=0x30000000 + 0x0000000A) となります。

プログラム内相対アドレスは、ブレークポイント設定アドレスとプログラムの先頭アドレスから求めればよいので、0x0000000A (=3000000A - 0x30000000) となります。

<prog.map>

| *** Mapping List *** |          |          |      |       |
|----------------------|----------|----------|------|-------|
| SECTION              | START    | END      | SIZE | ALIGN |
| P                    | 30000000 | 3000006f | 70   | 4     |

### 3 コマンド

#### <レジスタ内容画面>

機能：レジスタ内容の表示および設定をします。

操作：[ブレークポイント]画面より、 ボタンをクリックして表示します。

#### (1) レジスタ内容

以下のレジスタ内容が表示されます。これらのボックスに変更したい値を入力し、[Enter]キーを押すか、 ボタンをクリックすると、指定した内容でレジスタ内容を変更することができます（指定は16進数で行います）。

| No. | 項目     | 内容                        |
|-----|--------|---------------------------|
| 1   | SR     | ステータスレジスタ                 |
| 2   | PC     | プログラムレジスタ                 |
| 3   | GBR    | グローバルベースレジスタ              |
| 4   | PR     | プロシジャレジスタ                 |
| 5   | MACH   | システムレジスタ（積和上位レジスタ）        |
| 6   | MACL   | システムレジスタ（積和下位レジスタ）        |
| 7   | FPUL   | 浮動小数点通信レジスタ               |
| 8   | FPSCR  | 浮動小数点ステータス/コントロールレジスタ     |
| 9   | R0~R15 | 汎用レジスタ（R15はスタックポインタとして使用） |

#### (2) 浮動小数点レジスタ

ボタンをクリックしたときに、以下の浮動小数点レジスタの内容を表示・設定を行う[浮動小数点レジスタ]画面が表示されます。

| No. | 項目 | 内容             |
|-----|----|----------------|
| 1   | FR | 単精度浮動小数点レジスタ   |
| 2   | XF | 単精度浮動小数点拡張レジスタ |
| 3   | DR | 倍精度浮動小数点レジスタ   |
| 4   | XD | 倍精度浮動小数点拡張レジスタ |

#### (3) ボタン

指定したレジスタ内容および汎用レジスタ内容で、ブレーク中のレジスタが変更されます。変更後、[ブレークポイント]画面に戻ります。

#### (4) ボタン

レジスタ内容の変更を行わず、[ブレークポイント]画面に戻ります。

< [浮動小数点レジスタ] 画面 >

機能：浮動小数点レジスタの内容を表示および設定します。

操作：[レジスタ内容] 画面より、内容表示 ボタンをクリックして表示します。



図 3-17 [浮動小数点レジスタ] 画面

#### (1) HEX/FLOAT

表示内容の表示形式をHEX（16進数）形式またはFLOAT（実数）形式に切り替えることができます。

この切り替えによって表示内容のボックスが図 3-18のように変化します。

<単精度浮動小数点（FR）での表示例>



図 3-18 [浮動小数点レジスタ] 画面 (FLOAT)

#### (2) [レジスタ] 表示

[レジスタ内容] 画面で指定したレジスタの項目が0~15の範囲で表示されます。

### 3 コマンド

---

(3) [レジスタ内容] 表示

表示レジスタの内容が表示されます。表示および設定形式は、HEX/FLOAT形式に切り替え可能です。

(4)  ボタン

指定した浮動小数点レジスタの内容で、ブレーク中のレジスタが変更されます。変更後、[レジスタ内容] 画面に戻ります。

(5)  ボタン

浮動小数点レジスタの変更を行わず、[レジスタ内容] 画面に戻ります。

## 3.12 エラーログ

機能：エラーログ情報の一覧を表示します。

操作：[CPMSデバッガ]画面より、エラーログ ボタンをクリックして表示します。

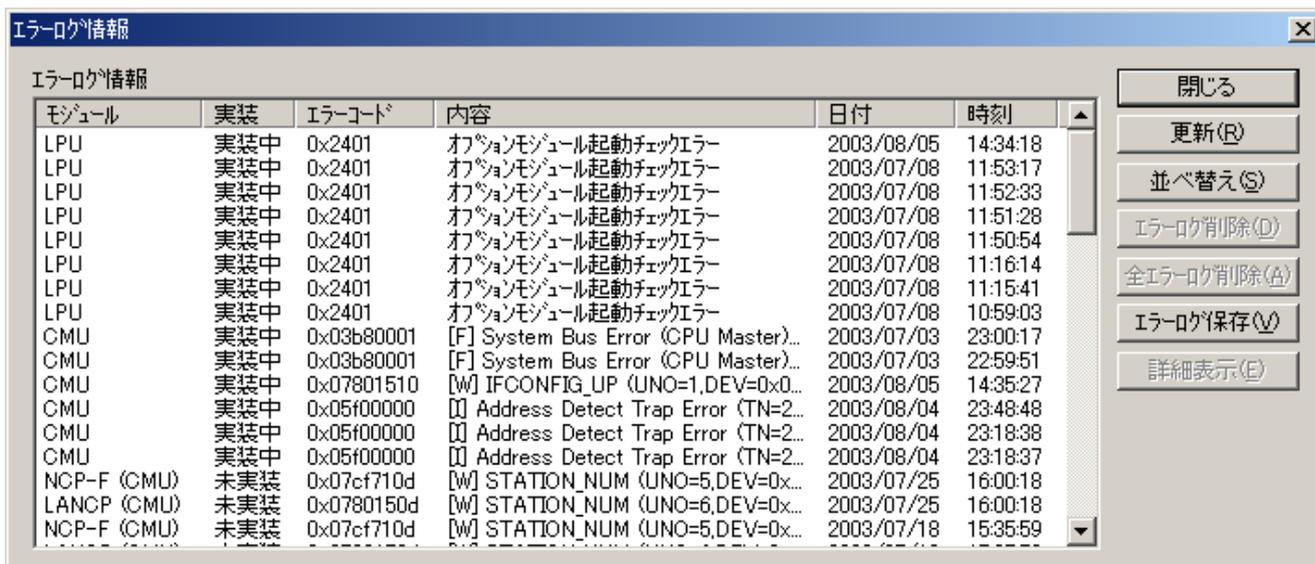


図 3-19 [エラーログ情報]画面

### (1) [エラーログ情報]ボックス

現在のエラーログ情報の一覧が表示されます。

表示される内容は以下のとおりです。

| No. | 項目     | 内容                      |
|-----|--------|-------------------------|
| 1   | モジュール  | エラーログが存在するオプションモジュールの名称 |
| 2   | 実装     | モジュールの実装状態              |
| 3   | エラーコード | 発生したエラーのエラーコード          |
| 4   | 内容     | 発生したエラーのエラーコード内容        |
| 5   | 日付     | エラー発生日                  |
| 6   | 時刻     | エラー発生時刻                 |

### (2) 閉じる ボタン

エラーログ情報の表示を終了し、[CPMSデバッガ]画面に戻ります。

### 3 コマンド

---

(3) **更新** ボタン

最新のエラーログ情報が表示されます。

(4) **並べ替え** ボタン

エラー情報を発生時間順に並べ替えます。**並べ替え** ボタンをクリックするたびに発生時間の昇順、降順に入れ替わります。

(5) **エラーログ削除** ボタン

[エラーログ情報] ボックスで指定したモジュールのエラーログ情報が削除されます。

(6) **全エラーログ削除** ボタン

LPU、CMU、オプションモジュールのエラーログ情報がすべて削除されます。

(7) **エラーログ保存** ボタン

LPU、CMU、オプションモジュールのエラーログ情報をテキストファイルに保存することができます。このボタンのクリックで保存ファイルを選択する [名前を付けて保存] 画面が表示されます。

(8) **詳細表示** ボタン

[エラーログ情報] ボックスで指定したエラーログの詳細を表示する [エラーログ詳細] 画面が表示されます。CMUエラーログのみ表示可能です。

< [エラーログ詳細] 画面 >

機能：CMUエラーログ情報の詳細を表示します。

操作：[エラーログ情報] 画面より、**詳細表示** ボタンをクリックして表示します。

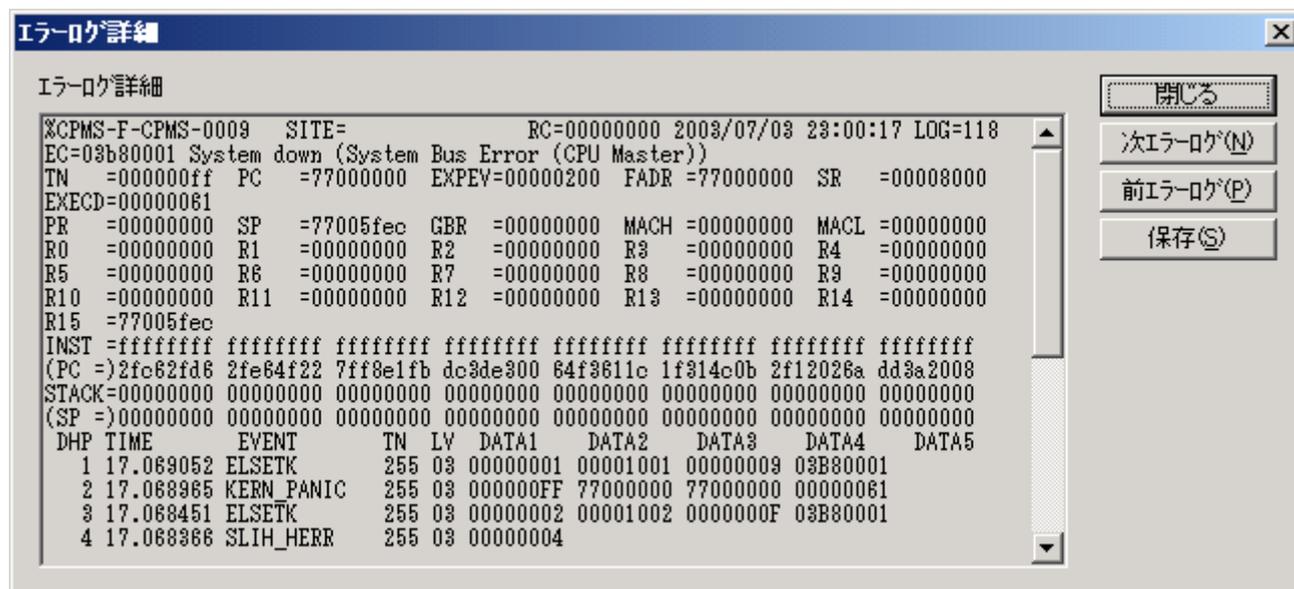


図 3-20 [エラーログ詳細] 画面

(1) [エラーログ詳細] ボックス

[エラーログ情報] 画面で指定したエラーログの詳細が表示されます。

表示される内容は、RPDP/S10Vのsvelogコマンドと同様です。

(2) **閉じる** ボタン

エラーログ詳細の表示を終了し、[エラーログ情報] 画面に戻ります。

(3) **次エラーログ** ボタン

次のCMUのエラーログが表示されます。

(4) **前エラーログ** ボタン

前のCMUのエラーログが表示されます。

(5) **保存** ボタン

[エラーログ詳細] ボックスに表示している内容をテキスト形式で保存することができます。先頭行に記録されるサイト名は、[DEBUGGER] 固定です。

### 3 コマンド

< [名前を付けて保存] 画面 >

機能：エラーログ情報をテキスト形式で保存します。

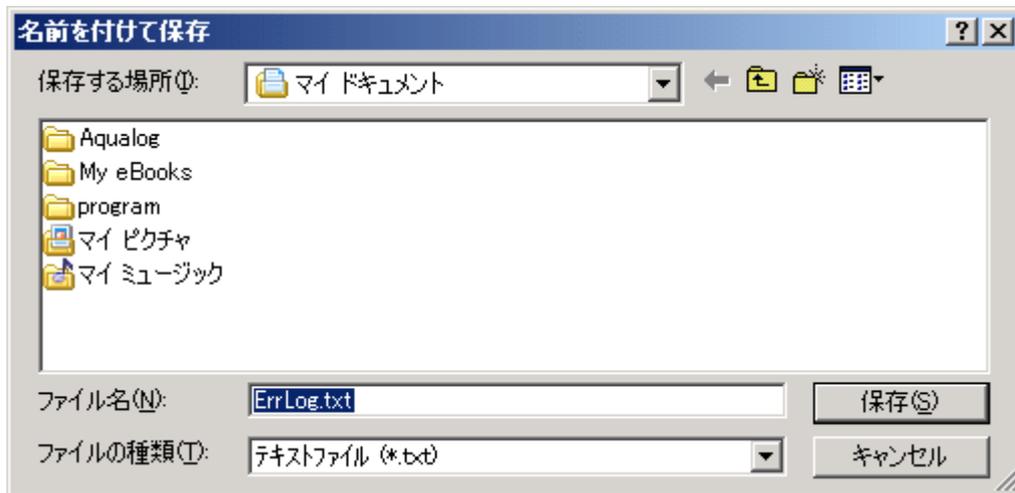


図 3-21 [名前を付けて保存] 画面

(1) [保存する場所] ボックス

コンボボックスや入力ボックスの選択でファイルを保存するフォルダの移動や、保存ファイルの選択を行ってください。

(2) [ファイル名] ボックス

選択された保存ファイルの表示や、保存ファイル名を入力してください。デフォルトで“ErrLog.txt”が表示されます。

(3) [ファイルの種類] コンボボックス

保存するファイルの種類を選択してください。“テキストファイル (\*.txt)”と“すべてのファイル (\*.\*)”の選択が行えます。

(4) **保存** ボタン

指定したファイルにエラーログ情報を保存し、前の画面に戻ります。  
先頭行に記録されるサイト名は、[DEBUGGER] 固定です。

(5) **キャンセル** ボタン

指定したファイルの保存を行わず、前の画面に戻ります。

### 3.13 PCs状態表示

機能：PCsの状態を表示します。

操作：[CPMSデバッグ]画面より、**PCs状態表示** ボタンをクリックして表示します。

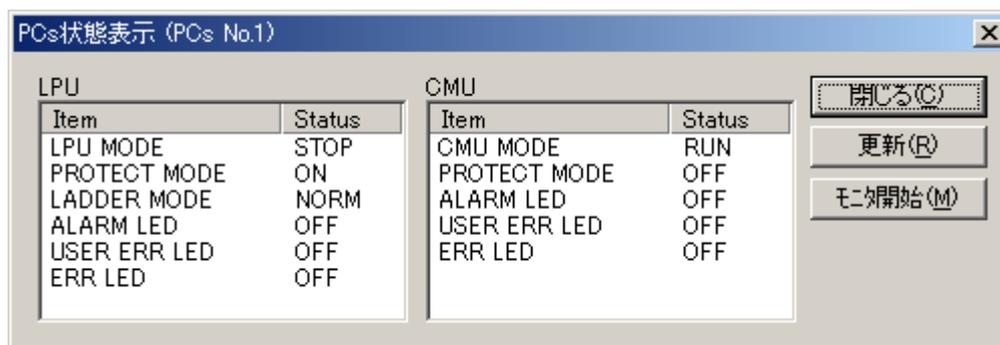


図 3-22 [PCs状態表示]画面

(1) [LPU] ボックス、[CMU] ボックス

現在のLPUおよびCMUの状態が表示されます。

表示される内容は以下のとおりです。

| No. | Item         | Status | 内容                    |
|-----|--------------|--------|-----------------------|
| 1   | LPU/CMU MODE | RUN    | LPU/CMUが動作状態          |
|     |              | STOP   | LPU/CMUが停止状態          |
| 2   | PROTECT MODE | ON     | LPU/CMUがプロテクションモード状態  |
|     |              | OFF    | LPU/CMUが非プロテクションモード状態 |
| 3   | LADDER MODE  | NORM   | LPUが通常動作状態            |
|     |              | SIMU   | LPUがシミュレート状態          |
| 4   | ALARM LED    | ON     | [ALARM] インディケータ点灯中    |
|     |              | OFF    | [ALARM] インディケータ消灯中    |
| 5   | USER ERR LED | ON     | [USER] インディケータ点灯中     |
|     |              | OFF    | [USER] インディケータ消灯中     |
| 6   | ERR LED      | ON     | [ERR] インディケータ点灯中      |
|     |              | OFF    | [ERR] インディケータ消灯中      |

(2) **閉じる** ボタン

PCs状態表示を終了し、[CPMSデバッグ]画面に戻ります。

### 3 コマンド

---

(3)  ボタン

最新のPCs状態を表示します。

(4)  ボタン、  ボタン

システム状態のモニタのON/OFFを切り替えることができます。

### 3.14 時刻設定

機能：CPU（PCs）の現在時刻の表示と設定をします。

操作：[CPMSデバッグ] 画面より、**時刻設定** ボタンをクリックして表示します。



図 3-23 [時刻設定] 画面

- (1) **時刻** ボックス  
現在のCPU（PCs）に登録されている時刻が表示されます。
- (2) **時刻設定** ボタン  
[時刻設定] ボックスに入力した内容で、CPU（PCs）の時刻を更新します。
- (3) **PC時刻取得** ボタン  
パソコンの時刻で [時刻設定] ボックスの内容を更新することができます。  
設定を有効にする場合は、**時刻設定** ボタンをクリックしてください。
- (4) **キャンセル** ボタン  
時刻の設定を行わずに、[CPMSデバッグ] 画面に戻ります。

## 3.15 ADTの設定／解除

< [ADTの設定／解除] 画面 >

機能：ADT（アドレスディテクトトラップ）を設定します。

操作：[CPMSデバッガ] 画面より、**ADTの設定／解除** ボタンをクリックして表示します。

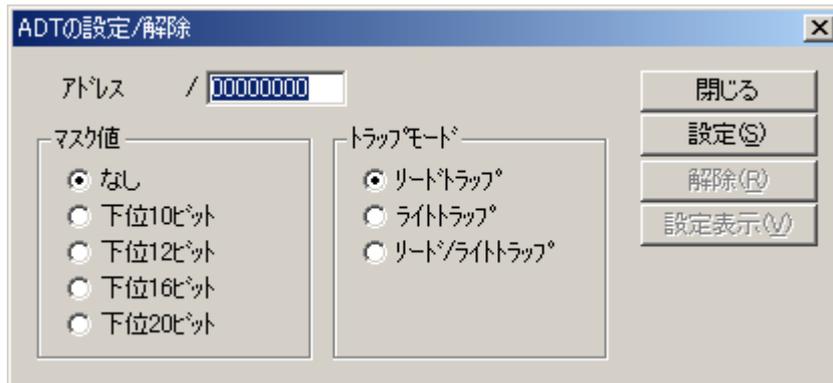


図 3-24 [ADTの設定／解除] 画面

- (1) [アドレス] ボックス  
トラップを設定する論理アドレスをバイト単位で指定してください。
- (2) [マスク値] グループ  
論理アドレスをマスクするマスク値を選択してください。
- (3) [リード／ライトトラップ] グループ  
ADTに設定するモードを選択してください。
- (4) **閉じる** ボタン  
ADTの設定／解除処理を終了し、[CPMSデバッガ] 画面に戻ります。
- (5) **設定** ボタン  
指定した内容でADTが設定されます。  
ADTの対象となるのは、[アドレス] ボックスに設定した論理アドレスに対し、バイトアクセス、ワードアクセス、ロングアクセスとなります。設定後、[ADT設定内容] 画面を表示し、[ADTの設定／解除] 画面に戻ります。

- (6)  ボタン

ADTの設定が解除されます。

- (7)  ボタン

ADTの設定内容を表示する [ADT設定内容] 画面が表示されます。

< [ADT設定内容] 画面 >

機能：ADTの設定内容を表示します。

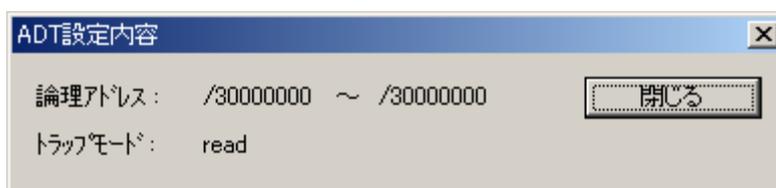


図 3-25 [ADT設定内容] 画面

- (1) 論理アドレス

ADTを設定したアドレスの範囲が表示されます。

- (2) モード

ADTに設定した以下のモードが表示されます。

read : リードトラップ

write : ライトトラップ

access: リード/ライトトラップ

- (3)  ボタン

ADT設定内容の表示を終了し、[ADTの設定/解除] 画面に戻ります。

### 3 コマンド

<注意事項>

- ADTは、1箇所だけ設定できます。
- ADTは、CPUのリスタートでは解除されません。
- ADTで検出できるのは、プロセッサのMMUを経由したアクセスのみです。このため、物理アドレスでアクセスするデバッガのメモリアクセスサブコマンドは、ADTでは検出できません。
- ADTが設定できる範囲は、図3-26の網掛けで示される範囲内になりますが、未マッピングエリアには設定できません。

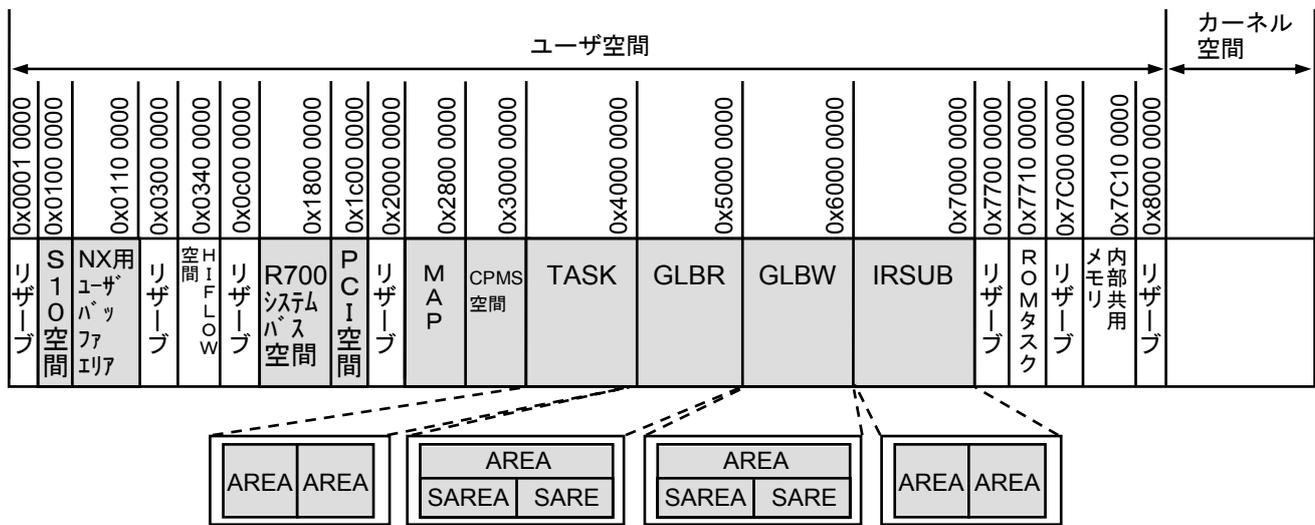


図3-26 ADTが設定できる範囲

### 3.16 ADT発生監視

機能：ADTの発生を監視します。

操作：[CPMSデバッガ]画面より、**ADT発生監視** ボタンをクリックして表示します。

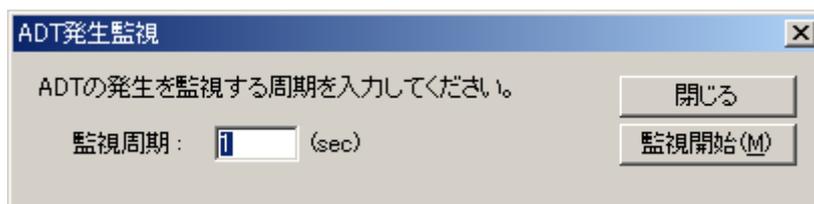


図 3-27 [ADT発生監視]画面

- (1) [監視周期] ボックス  
ADTの発生を監視する周期を、1～60secの範囲で指定してください。
- (2) **閉じる** ボタン  
ADT発生監視を終了し、[CPMSデバッガ]画面に戻ります。
- (3) **監視開始** ボタン  
ADTの発生を[監視周期]ボックスで指定した間隔で監視することができます。  
ADTの発生を確認した場合、以下のメッセージを表示し、監視処理が終了されます。



図 3-28 ADT発生画面

## 3.17 DHP記録許可／禁止／表示

< [DHPの記録許可／禁止／表示] 画面 >

機能：DHPの記録モードを設定します。

操作：[CPMSデバッグ] 画面より、**DHP記録許可／禁止／表示** ボタンをクリックして表示します。

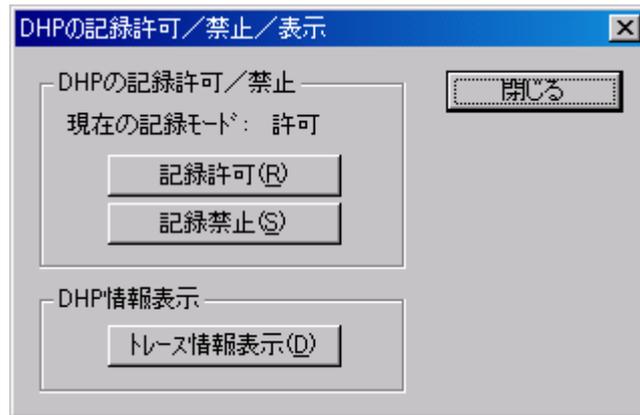


図 3-29 [DHPの記録許可／禁止／表示] 画面

- (1) [現在の記録モード] メッセージ  
DHPの記録モードが表示されます。
- (2) **記録許可** ボタン  
DHPの記録を許可モードに設定することができます。  
設定が正常終了した場合、[現在の記録モード] メッセージが更新されます。
- (3) **記録禁止** ボタン  
DHPの記録を禁止モードに設定することができます。  
設定が正常終了した場合、[現在の記録モード] メッセージが更新されます。
- (4) **トレース情報表示** ボタン  
DHPトレース情報を表示する [DHPトレース情報] 画面が表示されます。
- (5) **閉じる** ボタン  
DHPの記録モードの設定を終了し、[CPMSデバッグ] 画面に戻ります。

## &lt; [DHPトレース情報] 画面 &gt;

機能：DHPのトレース情報を表示します。

操作：[DHPの記録許可/禁止/表示] 画面より、トレース情報表示 ボタンをクリックして表示します。

| DHP | TIME      | EVENT      | TN  | LV | DATA1    | DATA2    | DATA3    | DATA4    | DATA5    |
|-----|-----------|------------|-----|----|----------|----------|----------|----------|----------|
| 1   | 46.610546 | DHPREAD    | 244 | 03 | 7C0D0000 | 7C000DA8 |          |          |          |
| 2   | 46.610527 | DISPATCH_E | 244 | 03 | 000000F4 | 0000002B | 8468F000 | 00000001 |          |
| 3   | 46.610517 | RUNQ       | 244 | 03 | 000000F4 |          |          |          |          |
| 4   | 46.610509 | DISPATCH   | 244 | 03 | 000000F4 | 0000002B | 8468F000 |          |          |
| 5   | 46.610435 | DISPATCH_E | 244 | 03 | 000000F4 | 0000002B | 8468F000 | 00000001 |          |
| 6   | 46.610350 | RUNQ       | 244 | 03 | 000000F4 |          |          |          |          |
| 7   | 46.610337 | WAKEUP     | 244 | 03 | 8492343C |          |          |          |          |
| 8   | 46.609993 | NET_ATEN   | 244 | 03 | 01040800 | 061804A8 | 04331B5B | C0C0C080 | C0C0C001 |
| 9   | 46.609677 | NET_TERM   | 244 | 03 | 0104FFFF | 0000B05F | 00009003 | 00004AFB | 00000000 |
| 10  | 46.609497 | IDLE       | 244 | 03 |          |          |          |          |          |
| 11  | 46.609467 | DISPATCH   | 244 | 03 | 000000F4 | 0000001A | 8468F000 |          |          |
| 12  | 46.609449 | SLEEP      | 244 | 03 | 8492343C | 0000001A |          |          |          |
| 13  | 46.609435 | RECV       | 244 | 03 | 0104C02E | 7C000D80 | 04800000 |          |          |
| 14  | 46.609396 | NET_START  | 244 | 03 | 01040800 | 000604A8 | 1B5B0433 | 50181000 | B402D4E4 |
| 15  | 46.609160 | SEND       | 244 | 03 | 0104C02E | 7C000D80 | 04800000 |          |          |
| 16  | 46.608978 | RECV       | 244 | 03 | 0104C02E | 7C000D80 | 04800000 |          |          |

図 3-30 [DHPトレース情報] 画面

## (1) [DHPトレース情報] ボックス

現在のDHPトレース情報の一覧を表示します。

表示される内容は、RPDP/S10Vのsvdhpコマンドと同様です。

表示される内容は以下のとおりです。

| No. | 項目          | 内容  |
|-----|-------------|---|
| 1   | DHP         | DHPトレース番号の表示番号                            |
| 2   | TIME        | トレース時刻<br>tt.tttttt<br>↑↑<br>秒 1マイクロ秒まで出力 |
| 3   | EVENT       | トレースポイント種別                                |
| 4   | TN          | タスク番号                                     |
| 5   | LV          | 優先レベル                                     |
| 6   | DATA1~DATA5 | トレースデータ (16進数で出力)                         |

(2) 閉じる ボタン

DHPトレース情報表示を終了し、[DHPの記録許可/禁止/表示] 画面に戻ります。

(3) 保存 ボタン

[DHPトレース情報] ボックスに表示している内容をテキスト形式で保存することができます。

## 3.18 スタックの初期化

機能：タスクのスタックを固定パターンで初期化します。

操作：[CPMSデバッガ]画面より、スタックの初期化 ボタンをクリックして表示します。



図 3-31 「スタックの初期化」画面

- (1) [タスク番号] ボックス  
スタック使用量を表示するタスク番号を1~229の範囲で入力してください。
- (2) [初期化パターン] ボックス  
スタックを初期化する固定パターンを0x0~0xFの範囲で入力してください。
- (3) OK ボタン  
指定タスクのスタックが初期化されます。
- (4) キャンセル ボタン  
スタックの初期化を行わず、[CPMSデバッガ]画面に戻ります。
- (5) [範囲指定] チェックボックス  
タスク番号の範囲指定を許可するかを選択してください。

## &lt;注意事項&gt;

- 初期化の対象となるタスクは、DORMANT状態にしておいてください。
- タスク範囲指定ありの場合、タスク範囲指定でスタック初期化を実行する前に、指定範囲中にDORMANT状態以外のタスクが存在しないかチェックします。チェックの結果DORMANT状態以外のタスクが見つかった場合、「タスク番号 \*\* がDORMANT状態ではありません」というエラーメッセージを表示し異常終了します。異常終了した場合は、DORMANT状態のタスクに対してもスタックの初期化は実行されません (\*\*:タスク番号)。
- スタックの初期化は、指定したタスクのスタックが存在するページ領域のうち、ページ先頭アドレスからスタック最終アドレスまでの範囲を対象とします (図3-32を参照)。

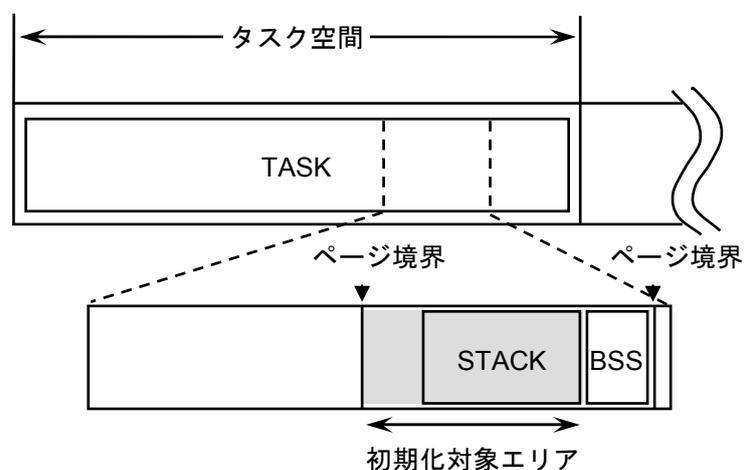


図3-32 スタックの初期化範囲

## 3.19 スタック使用量表示

機能：タスクのスタック使用量を表示します。

操作：[CPMSデバッガ] 画面より、スタック使用量表示 ボタンをクリックして表示します。



図 3-33 [スタック使用量表示] 画面

(1) [タスク番号] ボックス

スタック使用量を表示するタスク番号を1~229の範囲で入力してください。

(2) [チェックパターン] ボックス

スタック使用量を算出するチェックパターンを0x0~0xFの範囲で入力してください。

(3) [範囲指定] チェックボックス

タスク番号の範囲指定を許可するかを選択してください。範囲指定時、エラーメッセージは表示されません。

(4) キャンセル ボタン

スタック使用量の内容表示を行わず、[CPMSデバッガ] 画面に戻ります。

(5) **OK** ボタン

指定タスクのスタック使用量を表示する [スタック使用量表示] (内容表示) 画面が表示されます。

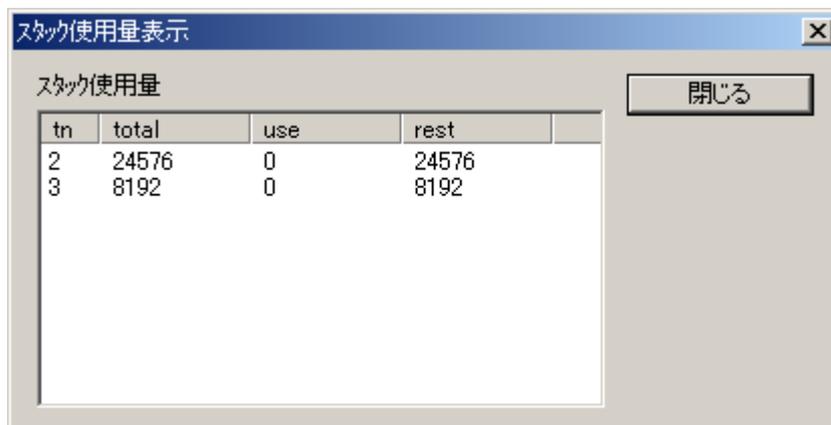


図 3-34 [スタック使用量表示] (内容表示) 画面

<表示項目>

| No. | 項目    | 内容          |
|-----|-------|-------------|
| 1   | tn    | タスク番号       |
| 2   | total | スタック使用可能サイズ |
| 3   | use   | スタック使用サイズ   |
| 4   | rest  | スタック未使用サイズ  |

(1) **閉じる** ボタン

[CPMSデバッガ] 画面に戻ります。

### 3 コマンド

#### <注意事項>

- チェックパターンに指定する値は、[スタックの初期化]画面で指定した初期化パターンと同じ値を指定してください。
- スタック使用量は、指定したタスクが使用するスタックのページ領域に対して、チェックパターンで指定した値と、異なるパターンが現れたアドレスをもとに算出します。このため、スタックの先頭が、チェックパターンと同一のパターンであった場合、正しいスタック使用量の表示はできなくなります。
- スタック使用量表示が表示する内容と、タスクの動作空間との対応を図3-35に示します。

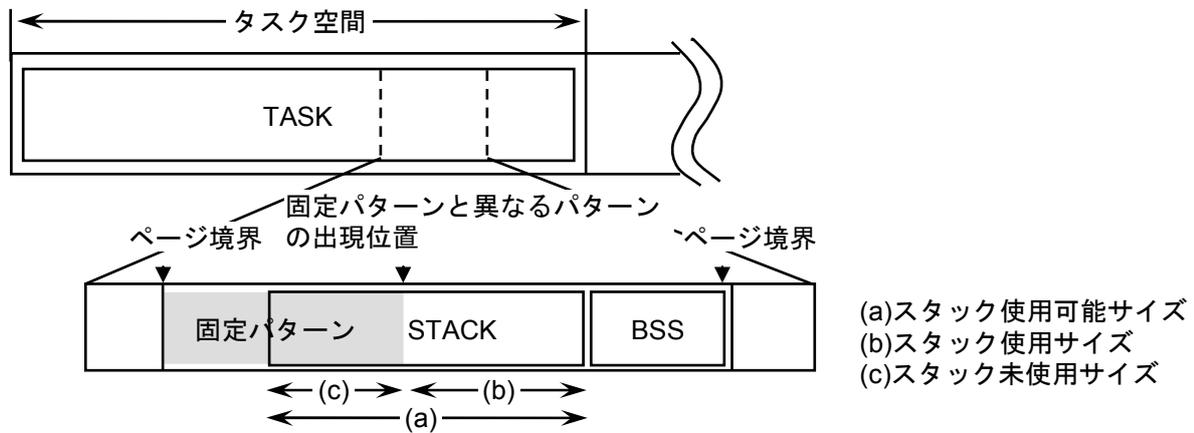


図3-35 スタック使用量表示が表示する内容

## 3.20 マトリクスモニタ

機能：マトリクスモニタを表示します。

操作：[CPMSデバッガ]画面より、マトリクスモニタ ボタンをクリックして表示します。

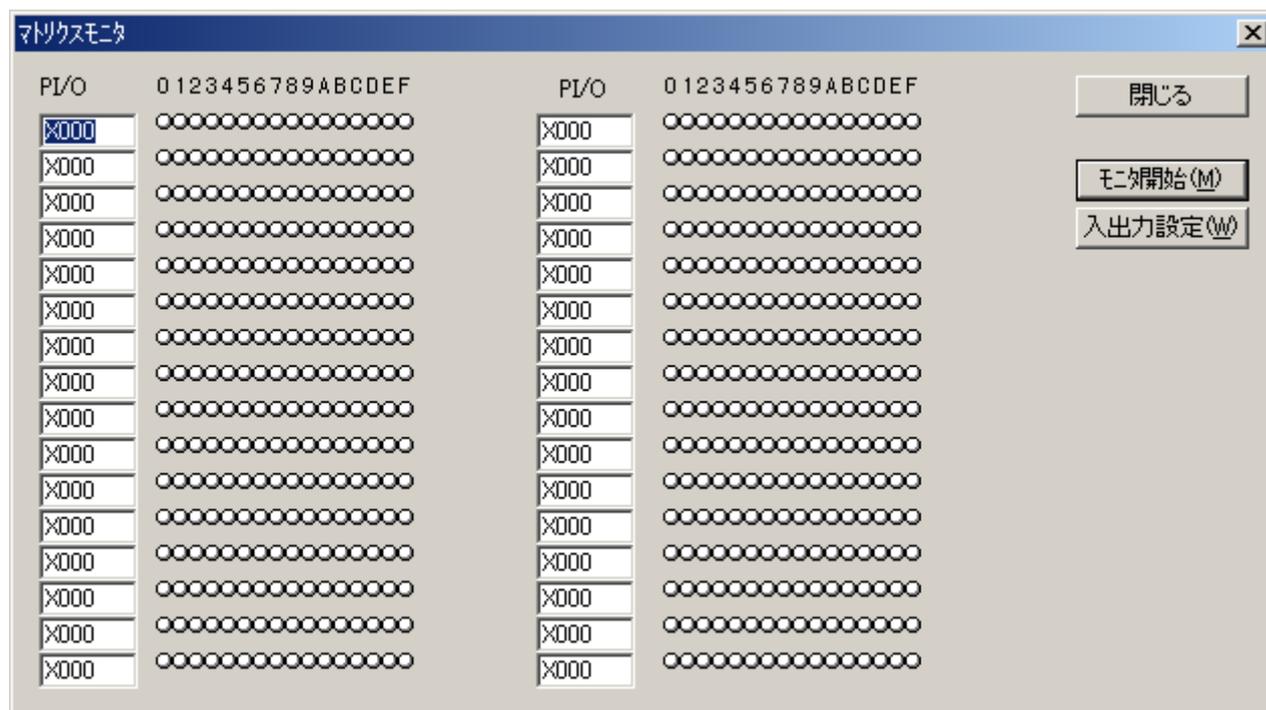


図 3-36 [マトリクスモニタ]画面

(1) モニタ開始 ボタン

入力したPI/Oのモニタが開始されます。

### 3 コマンド

---

(2) **入出力設定** ボタン

設定したPI/Oに値を設定する場合は **入出力設定** ボタンをクリックしてください。図3-37の画面が表示されます。PI/Oのビット状態を設定し、**OK** ボタンをクリックしてください。



図3-37 [入出力設定] 画面

(3) **閉じる** ボタン

マトリクスモニタを終了し、[CPMSデバッガ] 画面に戻ります。

## 3.21 MCS

機能：MCSを表示します。

操作：[CPMSデバッグ]画面より、**[MCS]** ボタンをクリックして表示します。

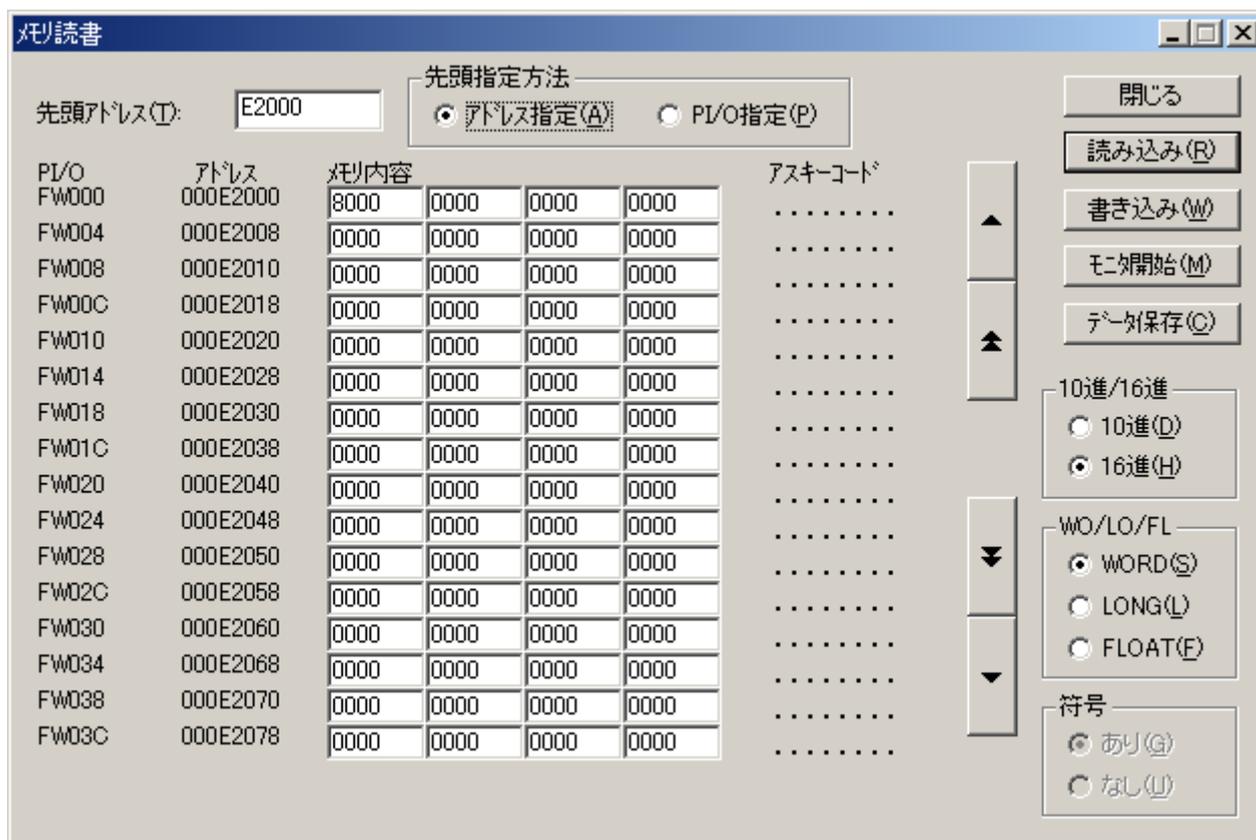


図 3-38 [MCS] 画面

(1) [先頭アドレス] ボックス

メモリ内容を表示するアドレスを指定してください。アドレスの指定方法は[先頭指定方法]グループボックスから、「アドレス指定」または「PI/O指定」を選択してください。

(2) **[読み込み]** ボタン

表示しているアドレスのメモリ内容をPCsから読み込みます。

(3) **[書き込み]** ボタン

表示しているアドレスのメモリ内容をPCsに書き込みます。

### 3 コマンド

---

- (4)  ボタン  
表示しているアドレスのモニターが開始されます。
- (5)  ボタン  
表示しているアドレスのモニターが停止されます。
- (6)  ボタン  
MCSを終了し、[CPMSデバッガ]画面に戻ります。
- (7)  ボタン  
MCSの表示内容をテキストファイルに保存することができます。
- (8) [10進/16進] ラジオボタン  
メモリ内容ボックスの表示形式を「10進/16進」に切り替えることができます。
- (9) [WO/LO/FL] ラジオボタン  
メモリ内容ボックスの表示形式を「WORD/LONG/FLOAT」に切り替えることができます。
- (10) [符号] ラジオボタン  
メモリ内容ボックスの表示形式を「符号あり/符号なし」に切り替えることができます。

## 3.22 接続PCs変更

機能：PCsとパソコンの通信種類（\*）を設定します。

操作：[CPMSデバッガ]画面より、**接続PCs変更** ボタンをクリックして表示します。



図 3-39 [通信種類] 画面

- (1) [IPアドレス] ボックス  
接続先PCsのIPアドレスを入力してください。
  - (2) **OK** ボタン  
指定したIPアドレスでPCsとの接続が行われます。接続後、[CPMSデバッガ]画面に戻ります。
  - (3) **キャンセル** ボタン  
通信種類の変更を行わず、[CPMSデバッガ]画面に戻ります。
- (\*) 通信種類  
CPMSデバッガとPCsの通信は、CMUのイーサネットのみサポートしています。  
CMUのIPアドレスはS10V基本システムから設定してください。

### 3.23 タスク環境初期化

機能：タスク環境 (\*) を初期化します。

操作：[CPMSデバッガ] 画面より、タスク環境初期化 ボタンをクリックして表示します。

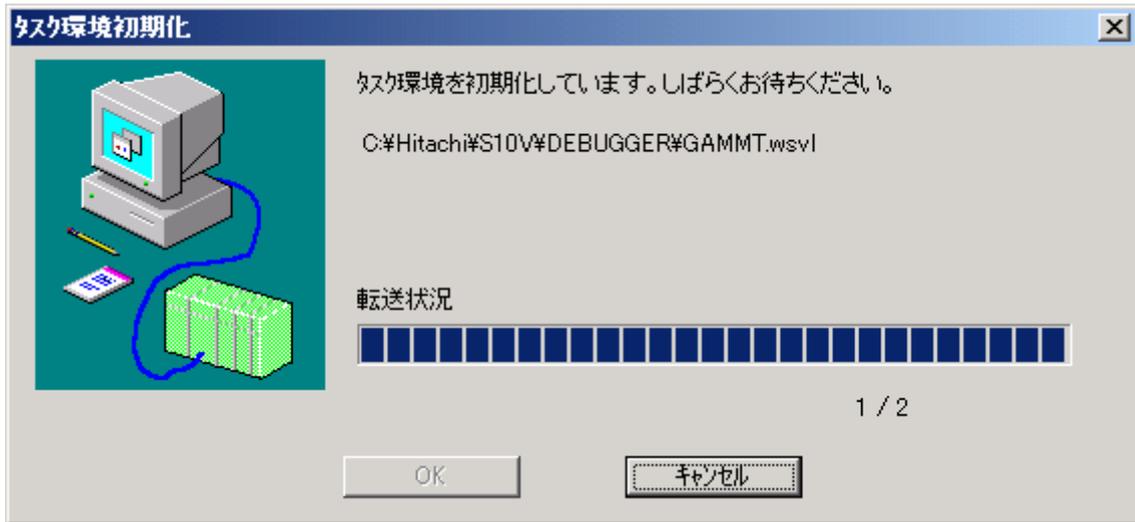
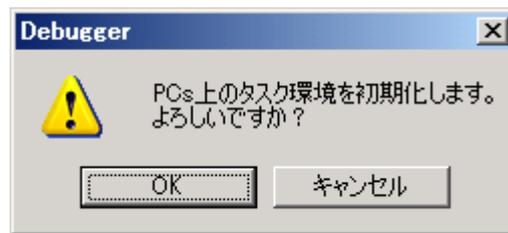


図 3-40 [タスク環境初期化] 画面

<確認メッセージ1>

この画面で キャンセル ボタンをクリックした場合、処理を中止し [CPMSデバッガ] 画面に戻ります。

OK ボタンをクリックした場合、確認メッセージ2が表示されます。

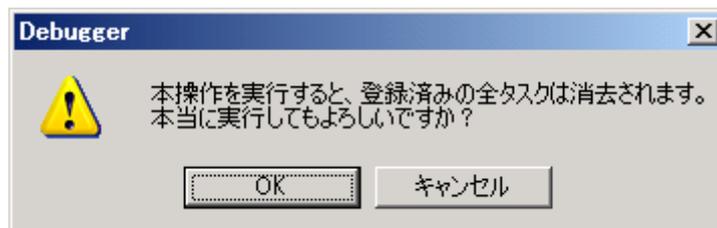


## &lt;確認メッセージ2&gt;

この画面で **キャンセル** ボタンをクリックした場合、処理を中止し [CPMSデバッガ] 画面に戻ります。

**OK** ボタンをクリックした場合、全タスクをABORT後、タスク環境が初期化されます。

タスク初期化終了後、以下のメッセージを表示し、 [CPMSデバッガ] 画面に戻ります。



### 3 コマンド

(\*) タスク環境

PCs上のタスク環境がCPMSデバッガのデフォルト環境に初期化されます。

CPMSデバッガのデフォルト環境（エリア定義MAP）を以下に示します。

|             |                   |
|-------------|-------------------|
| 0x2000 0000 | \$MAP (792KB固定)   |
| 0x200c 6000 | 空き                |
| 0x3000 0000 | \$TASK (4096KB)   |
| 0x3040 0000 | 空き                |
| 0x4000 0000 | 予約                |
| 0x4008 0000 | 空き                |
| 0x5000 0000 | \$GLBRW (10240KB) |
| 0x50a0 0000 | 空き                |
| 0x6000 0000 | 予約                |
| 0x6008 0000 | 空き                |

\$MAP : PCsメモリ上の管理情報を格納するエリアです。

\$TASK : タスク（プログラム）を格納するエリアです。

\$GLBRW : 読み書き両用グローバルを格納するエリアです。

図 3-41 CPMSデバッガのデフォルトの環境

# 付 録

## 付録A プログラムで使用できる名称

あらかじめシステムで用意されたサブルーチンの名称と同一名称のプログラムを使用する場合、注意が必要です。システムで用意されたサブルーチンは、すべてライブラリファイルに納められています。optlnkにおいてlibraryオプションを指定すると簡単に結合できます。しかし、システムのサブルーチンと同一名称である場合、そのサブルーチンを定義するオブジェクトファイルをoptlnkで引数に指定しないとライブラリファイルより同一名称のサブルーチンが結合されます。

以下に各システム用のライブラリファイルとそこで定義されている名称を示します。名称が重複しないようにプログラミングしてください。

重複した名称を使用する場合、ライブラリファイルの指定順序を結合したいオブジェクトファイルの後にすればライブラリファイルから結合されません。

以降にシステムで用意されたサブルーチンをライブラリ別に示します。

なお、\_ (アンダーライン) で始まる名称はシステムで予約した名称ですので使用しないでください。

ライブラリの構成は下記のようになっています。

| ライブラリ名          | ライブラリの内容                                  | 備考   |
|-----------------|---|--|
| libsh4nbmzz.lib | C言語用のサブルーチン群<br>非正規化数：非正規化数<br>値の丸め方：切り捨て | 詳細はshcコンパイラ付属のマニュアルを参照してください。                          |
| libsh4nbmdn.lib | C言語用のサブルーチン群<br>非正規化数：0<br>値の丸め方：切り捨て     |  |
| libcpms.lib     | CPMSマクロリンケージサブルーチン群                       | 詳細は「CPMS概説&マクロ仕様<br>(マニュアル番号 SVJ-3-201)」を<br>参照してください。 |

## &lt;libsh4nbmzz.lib&gt;

## &lt;libsh4nbmdn.lib&gt;

|             |         |          |          |         |      |
|-------------|---------|----------|----------|---------|------|
| atof        | memcpy  | strlen   | fpcheck  | fmod    | tan  |
| fpgetmask   | memset  | strncat  | fpchecko | log     | tanh |
| fpgetround  | modf    | strncmp  | acos     | log10   |      |
| fpgetsticky | sscanf  | strncpy  | asin     | matherr |      |
| fpsetmask   | sprintf | strpbrk  | atan     | pow     |      |
| fpsetround  | strcat  | strrchr  | atan2    | cos     |      |
| fpsetsticky | strchr  | strspn   | ceil     | sin     |      |
| frexp       | strcmp  | strtod   | exp      | cosh    |      |
| ldexp       | strcpy  | strtol   | fabs     | sinh    |      |
| memchr      | strcspn | vsprintf | floor    | sqrt    |      |

## &lt;libcpms.lib&gt;

|           |            |            |           |             |
|-----------|------------|------------|-----------|-------------|
| abort     | dhpread    | printf     | stime     | elset       |
| arsum     | exit       | prsrv      | susp      | geterrno    |
| asusp     | free       | queue      | timer     | gettimebase |
| chap      | gfact      | read       | usrdhpset | gtkmem      |
| chml      | cpms_ginfo | resume_env | usrelset  | getsysinfo  |
| close     | gtime      | rleas      | wait      | gettaskinfo |
| cpms_copy | ioctl      | rserv      | write     | usrdhp      |
| ctime     | open       | rsum       | chkbmem   | usrel       |
| delay     | pfree      | save_env   | chaktaer  | wrtmem      |
| dhpctl    | post       | sfact      | dhpset    |             |

## &lt;libsysctl.lib&gt;

|            |          |         |            |             |
|------------|----------|---------|------------|-------------|
| cardoff    | cardstat | dsuctl  | ptnwrite   | sysRegWrite |
| slotewrite | dcmctl   | dsustat | regLRread  | sysdo       |
| cardon     | dcmrb    | ledctl  | hdctl      | wdtset      |
| sloteclear | dcmstat  | pioctl  | sysRegRead |             |

## 付録B ライブラリの使用するスタックサイズ一覧

ライブラリが使用するスタックサイズの一覧を以下に示します。

(1/2)

| ライブラリ  | 関数名      | スタックサイズ |
|--|----------|---------|
| C標準ライブラリ<br>(libsh4nbmdn.lib<br>libsh4nbmzz.lib) | atof     | 408     |
|  | freexp   | 8       |
|  | ldexp    | 20      |
|  | memchr   | 0       |
|  | memset   | 12      |
|  | modf     | 40      |
|  | sscanf   | 528     |
|  | sprintf  | 752     |
|  | strcat   | 0       |
|  | strchr   | 0       |
|  | strcmp   | 20      |
|  | strcpy   | 24      |
|  | strcspn  | 4       |
|  | strlen   | 0       |
|  | strncat  | 4       |
|  | strncmp  | 4       |
|  | strncpy  | 0       |
|  | strpbrk  | 4       |
|  | strchr   | 12      |
|  | strspn   | 4       |
|  | strtod   | 408     |
|  | strtol   | 68      |
|  | vsprintf | 752     |
|  | acos     | 196     |
|  | asin     | 184     |
|  | atan     | 156     |
|  | atan2    | 176     |
|  | ceil     | 28      |
|  | exp      | 92      |
|  | fabs     | 0       |
| floor  | 28       |         |
| fmod   | 40       |         |

(2/2)

| ライブラリ  | 関数名         | スタックサイズ |
|--|-------------|---------|
| C標準ライブラリ<br>(libsh4nbmdn.lib<br>libsh4nbmzz.lib) | log         | 60      |
|  | log10       | 72      |
|  | pow         | 132     |
|  | cos         | 84      |
|  | sin         | 84      |
|  | cosh        | 112     |
|  | sinh        | 144     |
|  | sqrt        | 8       |
|  | tan         | 132     |
|  | tanh        | 156     |
| libers.lib                                       | fpgetmask   | 0       |
|  | fpgetround  | 0       |
|  | fpgetsticky | 0       |
|  | fpsetmask   | 0       |
|  | fpsetround  | 0       |
|  | fpsetsticky | 0       |
|  | fpcheck     | 0       |
|  | fpchecko    | 0       |
| libfirad.lib                                     | irglbad     | 0       |
|  | irsubad     | 0       |
| libcpms.lib                                      | memcpy (*)  | 28      |

(\*) リンクしたプログラムにおいて、memcpy()はC標準ライブラリではなく、CPMSライブラリのmemcpy()が使用されます。

付録C タスク作成手順

<RPDP/S10Vありの場合（標準手順）>

(1) 前提ソフトウェア

前題OSがWindows® 2000, Windows® XPの場合

SHコンパイラ（SuperH RISC engine C/C++コンパイラ Ver.7）

RPDP/S10V（型式：S-7895-10）

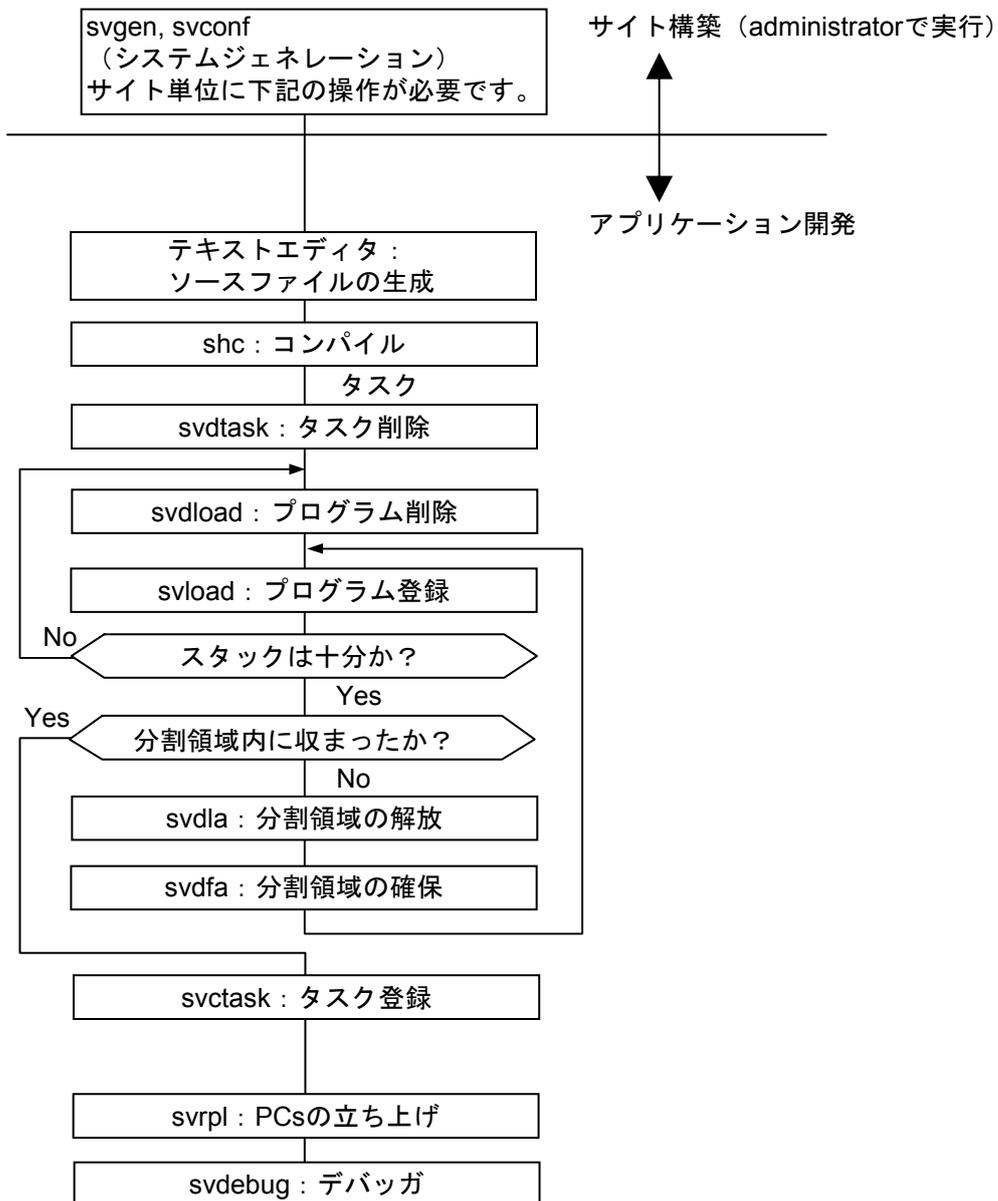
前題OSがWindows® 7（32bit）, Windows® 10（32bit）の場合

SHコンパイラ（SuperH RISC engine C/C++コンパイラ Ver.9）

RPDP/S10V（型式：S-7895-63）

(2) プログラム開発手順

RPDP/S10Vの操作手順に従い、タスクを作成してください。



## &lt;RPDP/S10Vなしの場合（非標準手順）&gt;

## (1) 前提ソフトウェア

前題OSがWindows® 2000, Windows® XPの場合

SHコンパイラ（SuperH RISC engine C/C++コンパイラ Ver.7）

S10V CPMSデバッガ（型式：S-7895-07）

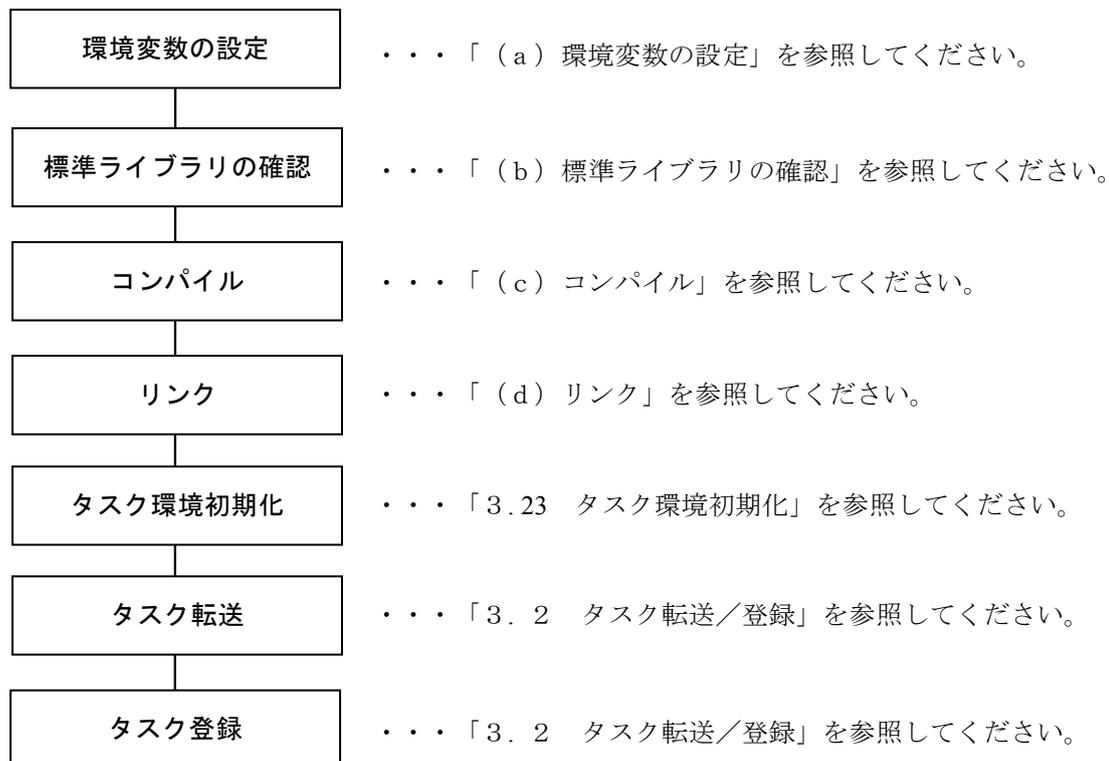
前題OSがWindows® 7（32bit）, Windows® 10（32bit）の場合

SHコンパイラ（SuperH RISC engine C/C++コンパイラ Ver.9）

S10V CPMSデバッガ（型式：S-7895-62）

## (2) プログラム開発手順

下記の操作手順に従い、タスクを作成してください。



(a) 環境変数の設定

以下の設定例に従って、環境変数を設定してください。

shcコンパイラパッケージ実行環境設定例 (S-7895-07P)

```
SHCPU=SH4
SHC_INC=C:\Hew2\Tools\Hitachi\Sh7_1_1\include;C:\users\subsystmp\include;
C:\Hitachi\S10V\Debug\include\cpms
SHC_LIB=C:\Hew2\Tools\Hitachi\Sh7_1_1\BIN ①
SHC_TMP=C:\Hitachi\S10V\Debug\tmp
HLNK_DIR=C:\Hitachi\S10V\Debug\lib;C:\users\subsystmp\lib ②
HLNK_TMP=C:\Hitachi\S10V\Debug\tmp
PATH=%PATH%;C:\Hew2\Tools\Hitachi\Sh7_1_1\BIN ③
```

- ① shcコンパイラパッケージ (Ver7.1.01) をデフォルトの設定でインストールした場合のコンパイラの格納ディレクトリC:\Hew2\Tools\Hitachi\Sh7\_1\_1\binを設定しています。
- ② 標準ライブラリ格納ディレクトリC:\Hitachi\S10V\Debug\libと、ユーザ独自のライブラリ格納ディレクトリC:\users\subsystmp\libを設定しています。
- ③ shcコンパイラの格納ディレクトリC:\Hew2\Tools\Hitachi\Sh7\_1\_1\binを設定しています。

shcコンパイラパッケージ実行環境設定例 (S-7895-62P)

```
SHCPU=SH4
SHC_INC=C:\Program Files\Renesas\Hew\Tools\Renesas\Sh9_4_0\include; ⑦
C:\users\subsystmp\include;C:\Hitachi\S10V\Debug\include;
C:\Hitachi\S10V\Debug\include\cpms
SHC_LIB=C:\Program Files\Renesas\Hew\Tools\Renesas\Sh9_4_0\BIN ④ ⑦
SHC_TMP=C:\Hitachi\S10V\Debug\tmp ⑦
HLNK_DIR=C:\Hitachi\S10V\Debug\lib;C:\users\subsystmp\lib ⑤
HLNK_TMP=C:\Hitachi\S10V\Debug\tmp
PATH=%PATH%;C:\ProgramFiles\Renesas\Hew\Tools\Renesas\Sh9_4_0\BIN ⑥ ⑦
```

- ④ shcコンパイラパッケージ (Ver9) をデフォルトの設定でインストールした場合のコンパイラの格納ディレクトリC:\Program Files\Renesas\Hew\Tools\Renesas\Sh9\_4\_0\binを設定しています。
- ⑤ 標準ライブラリ格納ディレクトリC:\Hitachi\S10V\Debug\libとユーザ独自のライブラリ格納ディレクトリC:\users\subsystmp\libを設定しています。
- ⑥ shcコンパイラの格納ディレクトリC:\Program Files\Renesas\Hew\Tools\Renesas\Sh9\_4\_0\binを設定しています。
- ⑦ C:\Program Files\Renesas\Hew\Tools\Renesas\Sh9\_4\_0\shv9400env.batを起動するとSHC\_INC, SHC\_LIB, SHC\_TMP, PATHの環境変数設定を行います (ただしSHC\_TMPは環境変数TEMPの設定値が設定されます)。

表 C-1 設定環境変数

| No. | 環境変数     | 設定内容   |
|-----|----------|--|
| 1   | path     | <p>環境変数pathにインストールしたコンパイラパッケージの実行ファイルの格納ディレクトリを追加してください。</p> <p>コンパイラ (shc) 、アセンブラ (asmsh) 、最適化リンケージエディタ (optlnk) のパスの設定が必要です。この環境変数の指定は必須です。</p> <p>指定フォーマット：<br/>path=&lt;実行ファイルパス名&gt;[;&lt;既存パス名&gt;;...]</p>  |
| 2   | SHC_LIB  | <p>コンパイラのロードモジュールおよびシステムインクルードファイルを格納したディレクトリを指定してください。この環境変数の指定は必須です。</p> <p>指定フォーマット：<br/>set SHC_LIB=&lt;実行ファイルパス名&gt;</p>  |
| 3   | SHCPU    | <p>対象とするCPU種別を指定します。</p> <p>このシステムでは、SHCPU=SH4を指定してください。</p> <p>この指定を省略した場合、CPU種別はSH1として扱われるので注意してください。</p> <p>CPU種別は-cpuオプションでも指定できます。</p> <p>指定フォーマット：<br/>set SHCPU=&lt;CPU&gt;</p>   |
| 4   | SHC_INC  | <p>コンパイラのインクルードファイル格納ディレクトリを指定してください。システムインクルードファイルの検索順序は、includeオプション指定ディレクトリ、SHC_INC指定ディレクトリ、システムディレクトリ (SHC_LIB) となります。</p> <p>ユーザインクルードファイルの検索順序はカレントディレクトリ、includeオプション指定ディレクトリ、SHC_INC指定ディレクトリとなります。</p> <p>指定フォーマット：<br/>set SHC_INC=&lt;インクルードパス名&gt; [;&lt;インクルードパス名&gt;;...]</p> |
| 5   | HLNK_DIR | <p>最適化リンケージエディタの入力ファイル格納ディレクトリを指定します。</p> <p>Inputオプション、libraryオプションで指定したファイルの検索順序は、カレントディレクトリ、HLNK_DIR指定ディレクトリになります。</p> <p>ローダのライブラリサーチパスもHLNK_DIRの設定に従います。</p> <p>指定フォーマット：<br/>set HLNK_DIR=&lt;入力ファイルパス名&gt;[;&lt;入力ファイルパス名&gt;;...]</p>  |
| 6   | SHC_TMP  | <p>コンパイラがテンポラリファイルを作成するディレクトリを設定します。</p> <p>set SHC_TMP=&lt;ディレクトリ&gt;</p>   |
| 7   | HLNK_TMP | <p>リンケージエディタがテンポラリファイルを作成するディレクトリを設定します。</p> <p>set HLNK_TMP=&lt;ディレクトリ&gt;</p>  |

(b) 標準ライブラリの確認

shcを使用してコンパイルする場合

● 浮動小数点数の扱い

shcでは浮動小数点数の非正規化数と丸めの扱いをコンパイルオプションで制御できます。

ただし、それぞれの扱いによって標準ライブラリが異なりますので注意してください。以下に非正規化数の扱いと丸め方を制御するオプションと対応する標準ライブラリの対応を示します。

リンク時に使用するライブラリを指定してください。ライブラリの指定方法は、「(d) リンク」を参照してください。

表 C-2 浮動小数点数の扱い制御オプション

|          | 仕様                | オプション                | デフォルト  |
|----------|-------------------|----------------------|--------|
| 非正規化数の扱い | 0として扱う            | -denormalization=off | 0として扱う |
|          | 非正規化数として扱う (*)    | -denormalization=on  |        |
| 結果の値の丸め方 | 有効数字を超える部分を切り捨てる  | -round=zero          | 切り捨て   |
|          | 有効数字を超える部分を四捨五入する | round=nearest        |        |

(\*) S10VのCPUであるSH4 (SH7751) は非正規化数をサポートしていないため、実行時には0として扱われません。

表 C-3 浮動小数点数の扱いと対応する標準ライブラリ

|         | -denormalization | -round  | 標準ライブラリ         |
|---------|------------------|---------|-----------------|
| 指定オプション | off              | zero    | libsh4nbmzz.lib |
|         | on               | zero    | 非サポート           |
|         | off              | nearest | 非サポート           |
|         | on               | nearest | libsh4nbmdn.lib |

表C-4 ライブラリ内容

| ライブラリ名          | ライブラリの内容                                  | 備考   |
|-----------------|---|--|
| libsh4nbmzz.lib | C言語用のサブルーチン群<br>非正規化数：非正規化数<br>値の丸め方：切り捨て | 詳細はshcコンパイラ付属のマニュアルを参照してください。                  |
| libsh4nbmdn.lib | C言語用のサブルーチン群<br>非正規化数：0<br>値の丸め方：切り捨て     |  |
| libcpms.lib     | CPMSマクロリンケージサブルーチン群                       | 詳細は「CPMS概説&マクロ仕様（マニュアル番号 SVJ-3-201）」を参照してください。 |

RPDP/S10VとS10V CPMSデバッグシステムにて提供するファイルを下記に示します。

表 C-5 S10V CPMSデバッグ提供ファイル

(1/2)

| ファイル名           | S10V CPMSデバッグ インストール時の格納先 (デフォルト) (*1) | 内容                          |
|-----------------|--|-----------------------------|
| libcpms.lib     | C:\HITACHI\S10V\DEBUG\LIB              | CPMSマクロ リンケージサブルーチン群        |
| libers.lib      | C:\HITACHI\S10V\DEBUG\LIB              | IEEE浮動小数点処理環境制御サブルーチン群      |
| libsh4nbmdn.lib | C:\HITACHI\S10V\DEBUG\LIB              | SHコンパイラC言語用のサブルーチン群         |
| libsh4nbmzz.lib | C:\HITACHI\S10V\DEBUG\LIB              | SHコンパイラC言語用のサブルーチン群         |
| libsysctl.lib   | C:\HITACHI\S10V\DEBUG\LIB              | CPMSマクロ システム管理用サブルーチン群 (*2) |
| cpms_dhp.h      | C:\HITACHI\S10V\DEBUG\INCLUDE          | DHP用インクルードファイル              |
| cpms_elog.h     | C:\HITACHI\S10V\DEBUG\INCLUDE          | ユーザエラーログ用インクルードファイル         |
| cpms_errno.h    | C:\HITACHI\S10V\DEBUG\INCLUDE          | エラー番号定義用インクルードファイル          |
| cpms_macro.h    | C:\HITACHI\S10V\DEBUG\INCLUDE          | CPMSマクロ使用のためのインクルードファイル     |
| cpms_table.h    | C:\HITACHI\S10V\DEBUG\INCLUDE          | タスク実行環境テーブル用インクルードファイル      |
| cpms_types.h    | C:\HITACHI\S10V\DEBUG\INCLUDE          | マクロ使用の型宣言用インクルードファイル        |
| cpms_ulsub.h    | C:\HITACHI\S10V\DEBUG\INCLUDE          | 組み込みサブルーチン用インクルードファイル       |
| ieeefp.h        | C:\HITACHI\S10V\DEBUG\INCLUDE          | 浮動小数点レジスタ用インクルードファイル        |
| sdio.h          | C:\HITACHI\S10V\DEBUG\INCLUDE          | SDIOドライバ用インクルードファイル         |
| stdio.h         | C:\HITACHI\S10V\DEBUG\INCLUDE          | 標準入出力用インクルードファイル            |
| stdlib.h        | C:\HITACHI\S10V\DEBUG\INCLUDE          | 標準ライブラリ用インクルードファイル          |
| cpms_adbbf.h    | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | ADBテーブル用インクルードファイル          |
| cpms_cardctl.h  | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | (未使用)                       |
| cpms_dcm.h      | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | (未使用)                       |
| cpms_log.h      | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | エラーログ番号用インクルードファイル          |
| cpms_oscbl.h    | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | OSCBテーブル用インクルードファイル         |
| cpms_pucomt.h   | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | (未使用)                       |
| cpms_rserv.h    | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | 共有リソース用インクルードファイル           |
| cpms_syscbl.h   | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | SYSCBテーブル用インクルードファイル        |
| cpms_taskenv.h  | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | タスクの実行環境データテーブル用インクルードファイル  |
| cpms_tcb.h      | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | TCBテーブル用インクルードファイル          |

(\*1) S10V CPMSデバッグのインストール先ドライブがCの場合

(\*2) wdtset, TimebaseToSecs等

表 C-5 S10V CPMSデバッグ提供ファイル

(2/2)

| ファイル名            | S10V CPMSデバッグ インストール時の格納先 (デフォルト) (*1) | 内容                       |
|------------------|--|--------------------------|
| cpms_timer.h     | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | Timer関連用インクルードファイル       |
| cpms_ucb.h       | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | UCBテーブル用インクルードファイル       |
| cpms_ulsubln.h   | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | 組み込みサブルーチンリンク用インクルードファイル |
| cpms_usr_param.h | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | 各種パラメータ定義用インクルードファイル     |
| dhpinfo.h        | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | DHP情報用インクルードファイル         |
| errno.h          | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | エラー番号定義用インクルードファイル       |
| macro_base_p.h   | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | CPMSマクロ使用のためのインクルードファイル  |
| macro_id.h       | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | CPMSマクロ番号定義インクルードファイル    |
| macro_rpd_p.h    | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | RPDP用インクルードファイル          |
| R700_parameter.h | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | テーブルアドレス定義用インクルードファイル    |
| R700_usr_param.h | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | システム定義用インクルードファイル        |
| sysctl.h         | C:\HITACHI\S10V\DEBUG\INCLUDE\CPMS     | システム制御用インクルードファイル        |

(\*1) S10V CPMSデバッグのインストール先ドライブがCの場合

## (c) コンパイル

このCPMSデバッグではコンパイラ・アセンブラを使用する場合には、「日立マイクロコンピュータ開発環境システムSuperH RISC engine C/C++コンパイラパッケージ Ver.7以降」（以下、shcコンパイラと表記します。）を使用することを前提としています。

## Cコンパイラオプション詳細

以下にshcを使用したコンパイルの方法と、コンパイル時の注意点を示します。

shcの仕様の詳細は、shcコンパイラ付属のマニュアルを参照してください。

## ● コマンド形式

shc [ $\Delta$ <オプション>...][ $\Delta$ <ファイル名>[ $\Delta$ <オプション>...]...]

(例) shc $\Delta$ test1.c  $\Delta$ test2.c [Enter]

## ● 環境変数の設定

shc使用時には表C-1に示す環境変数を設定してください。

## ● コンパイルリストの生成と保存 (shc)

タスクの使用するスタックサイズの算出等で必要となるため、コンパイルリストを生成して保存しておいてください。コンパイルリストを生成するためには、以下に示すオプションを指定します。

-listfileオプションはコンパイルするCソースファイルより前に指定してください。

Cソースファイルより後ろに指定した場合、最後の1ファイルのみコンパイルリストが生成されます。

## ■ コンパイルリスト生成指定

-listfile [=<リストファイル名>] -show=source,object

リストファイル名の指定を省略した場合、ソースファイル名と同じファイル名に拡張子“lst”を付加したファイルを生成します。

(例)

◆ shc  $\Delta$ -listfile  $\Delta$ test1.c  $\Delta$ test2.c [Enter]

「test1.c」、「test2.c」ともlistfile オプションが有効となります。

◆ shc  $\Delta$ test1.c  $\Delta$ test2.c  $\Delta$ -listfile [Enter]

listfile オプションは「test2.c」だけに対して有効になります。

[Enter] を押してください。

## (d) リンク

コンパイル済みのオブジェクトファイルをリンクし、PCsに送信可能な形式のファイル（アブソリュートモジュール）を生成します。

optlnkコマンドを使用してリンクします。以下にリンクの起動方法を示します。

MS-DOSのコマンドプロンプトより下記を実行します。

## ● コマンド形式

```
optlnk -noprelink -form=absolute -start=P,D,C/アドレス1,S/アドレス2,B/アドレス3  
-output=生成するアブソリュートモジュールのファイル名 -library=libcpms.lib  
-library=libsh4nbmdn.lib (-list=マップファイル -show=symbol) (-stack)  
ユーザオブジェクト,ライブラリ
```

## (例)

```
c:\tmp>optlnk -noprelink -form=absolute -start=P,D,C/30004000,S/30005000,B/30006000  
-output=task1.elf -library=libcpms.lib -library=libsh4nbmdn.lib -list=task1.map -stack task1.obj  
アドレス1 : 30004000  
アドレス2 : 30005000  
アドレス3 : 30006000  
生成するアブソリュートモジュールのファイル名 : task1.elf  
マップファイル : task1.map  
ユーザオブジェクト : task1.obj
```

表 C-6 optlinkの指定形式

| No. | 項目          | 指定形式                       | 機能   |
|-----|-------------|----------------------------|--|
| 1   | ライブラリファイル   | library=ファイル名              | ライブラリ指定  |
| 2   | プレリンカ起動抑止   | noprelink                  | 高速化のためにプレリンカの起動を抑止。C++のテンプレート機能および実行時型検査機能未使用の場合はプレリンク不要。  |
| 3   | 出力形式        | form=absolute              | リンカ出力ファイル形式指定。ロードモジュールの生成に使用。指定必須。   |
| 4   | 出力ファイル      | output=ファイル名               | リンカ出力ファイル指定。ロードモジュールの生成に使用。  |
| 5   | リストファイル     | list=ファイル名                 | リストファイル出力。マップファイル生成に使用。  |
| 6   | リスト内容       | show=symbol                | リストファイル出力。マップファイル生成に使用。  |
| 7   | セクション開始アドレス | start=セクション/<br>アドレス[,...] | セクションを指定アドレスに配置。プログラムの配置に使用。次に示すマップの構成とするため、必ず指定要。アドレス1、アドレス2は、プログラムのサイズかスタックサイズを考慮し、ユーザが決定してください。 |
| 8   | スタック情報ファイル  | stack                      | スタック使用量情報ファイルの出力。  |

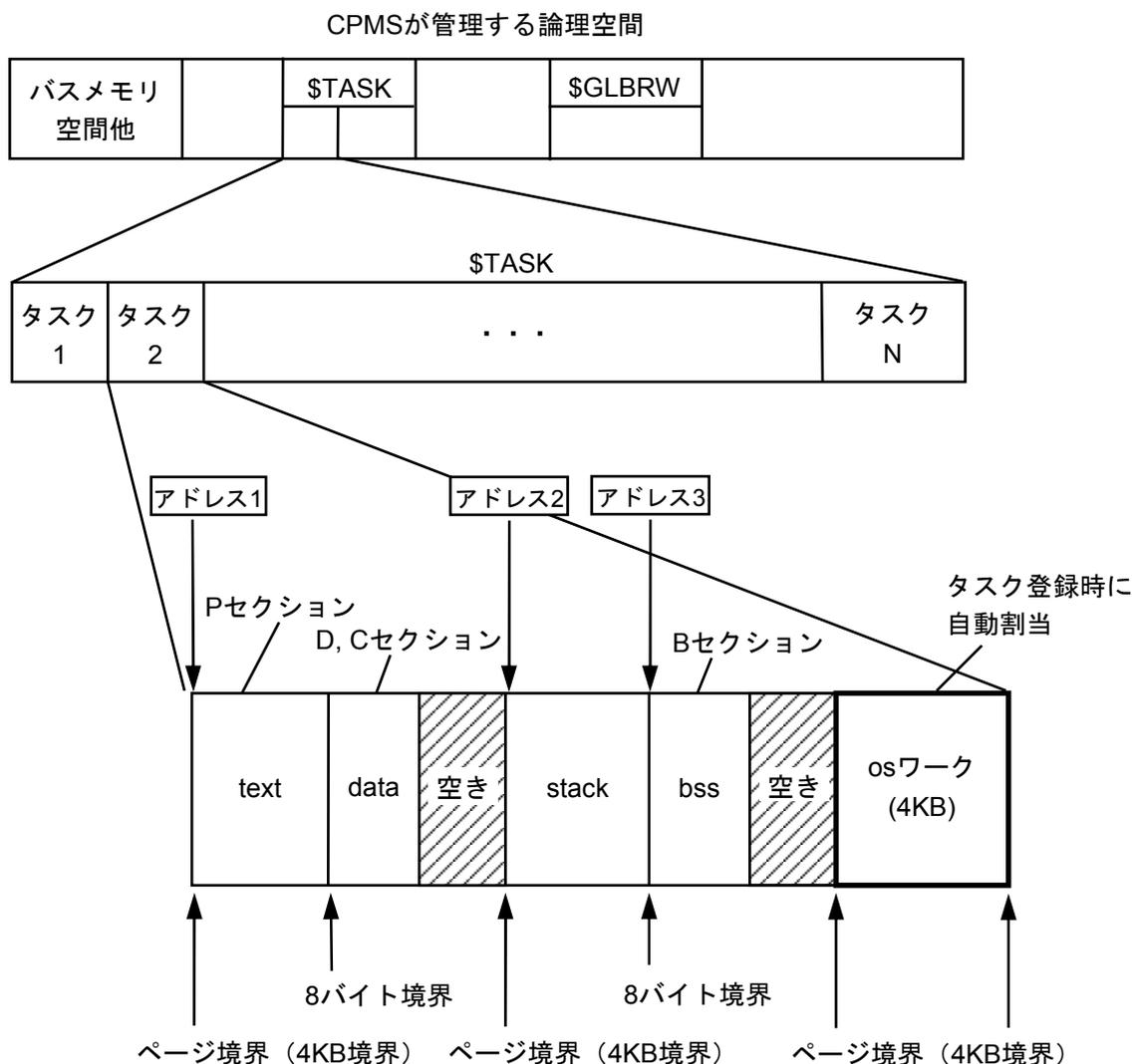


図 C-1 タスク構成

\$TASKは、タスク（プログラム）を格納するエリアです。

\$GLBRWは、読み書き両用グローバルを格納するエリアです。

textは、プログラムのプロシジャが入るエリアです。

dataは、プログラムの初期データが入るエリアです。

stackは、タスクが使う動的なワークエリアです。

bssは、プログラムが使う静的なワークエリアです。

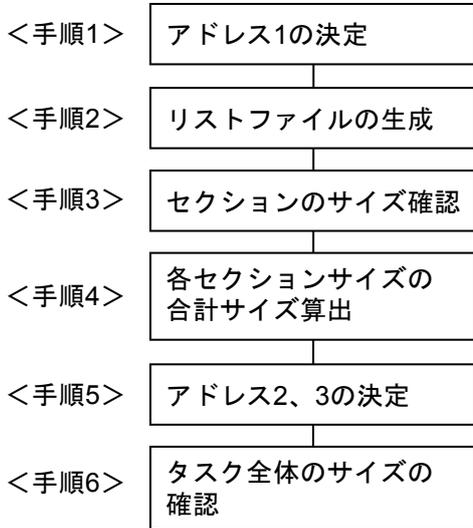
osワークは、osが使う動的なワークエリアです。タスク登録時にbssの次の1ページとして自動割り当てされます。

textはページ境界（4KB境界）に、data/bssとstackは8バイト境界に配置してください。

stackはページ単位に配置してください。

アドレス1、アドレス2、アドレス3はユーザが指定してください。

アドレス1、アドレス2の求め方について下記の手順で行います。



```

(例)
(ソースファイル) ファイル名 : task1.c

int b1;
int d1 = 10;
static int b2 ;
static int d2 = 100 ;
main()
{
    static int b3 ;
    static int d3 = 1000 ;

    int s1;
    int s2 = 20 ;
    exit(0);
}
    
```

必ず最終行にexit(0);を追加してください。

<手順1> (アドレス1の決定)

textの先頭アドレス (アドレス1) をユーザが任意に決定します。アドレス1は、他タスクで使用しているアドレス範囲と重複しないよう考慮し、ページ境界 (4KB) 単位としてください。アドレス1は、「(d) リンク」のoptlnkコマンド実行時に指定してください。

<手順2> (リストファイルの生成)

コンパイル時に-listオプションを付け、リストファイルを生成します。

```
F: %tmp>shc task1.c -list
```

<手順3> (セクションのサイズ確認)

生成されたリストファイル (ファイル名 : task1.lst) のframe sizeにてスタック容量を確認し、SECTION SIZE INFORMATIONにて各セクションのサイズを確認します。

スタック容量算出方法の詳細は、次ページ以降を参照してください。

(生成されたリストファイル) ファイル名 : task1.lst

| SCT OFFSET                           | CODE         | C LABEL       | INSTRUCTION | OPERAND  | COMMENT                                   |
|--------------------------------------|--------------|---------------|-------------|----------|---|
| P 00000000                           |              | <u>_main:</u> |             |          | ; function: main<br>; <u>frame size=0</u> |
| 00000000                             | D201         |               | MOV.L       | L11+2,R2 | ; _exit                                   |
| 00000002                             | 422B         |               | JMP         | @R2      |   |
| ~~~~                                 |              |               |             |          |   |
| ***** SECTION SIZE INFORMATION ***** |              |               |             |          |   |
| PROGRAM                              | SECTION (P): |               |             |          | 0000000C Byte(s)                          |
| CONSTANT                             | SECTION (C): |               |             |          | 00000000 Byte(s)                          |
| DATA                                 | SECTION (D): |               |             |          | 0000000C Byte(s)                          |
| BSS                                  | SECTION (B): |               |             |          | 0000000C Byte(s)                          |

スタックのサイズ ←

<手順4> (各セクションサイズの合計サイズ算出)

プログラムセクション(P)+コンスタントセクション(C)+データセクション(D)のサイズを算出します。下記の場合24バイトとなります。このサイズよりtextとdataが1ページ(4KB)内に収まることが分かります。

| <各セクションのサイズ> |              |                         |
|--------------|--------------|-------------------------|
| PROGRAM      | SECTION (P): | <u>0000000C Byte(s)</u> |
| CONSTANT     | SECTION (C): | <u>00000000 Byte(s)</u> |
| DATA         | SECTION (D): | <u>0000000C Byte(s)</u> |

(P) (C) (D)

$0xC + 0x0 + 0xC = 0x18$ バイト (10進で24バイト)

<手順5> (アドレス2、3の決定)

stackの先頭アドレス(アドレス2)とbssの先頭アドレス(アドレス3)を決定します。

stack/bssは、text/dataと異なるページとします。

手順3で求めたstackのサイズは0ですが、将来用として1ページ分(4KB)確保します。

そのため、stackの先頭アドレスはtext/dataの次(2ページ目の先頭)とします。

bssの先頭アドレスは、stackの次(3ページ目の先頭)とします。

手順3よりbssのサイズは12バイトですので、3ページ目に収まること分かります。

アドレス2とアドレス3は、「(d)リンク」のoptlnkコマンド実行時に指定してください。

<手順6> (タスク全体のサイズの確認)

OSワークはページ境界(4KB境界)に配置する必要があり、サイズは4KB固定ですので、4ページ目に配置されることになり、このタスク全体のサイズは16KBとなります。

別タスクを配置する場合、OSワークの最終アドレスから配置可能となります。

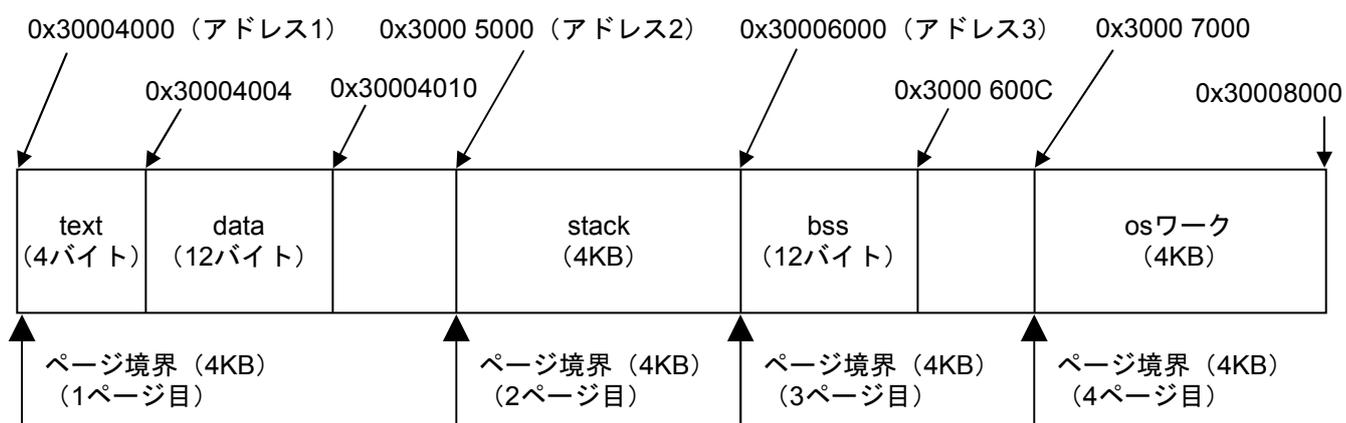


図 C-2 マップ構成

<スタック容量算出方法>

スタックエリアの使用量は、プログラムを構成する各関数ごとのスタック使用量を算出し、関数の呼び出し関係から全体のスタック使用量を算出します。

(1) 各関数の使用するスタック領域の算出

各関数の使用するスタック領域の大きさは、コンパイラが出力するオブジェクトリストのframe sizeから分かります。

以下に具体例を示します。

■ ソースコード

```
extern int h(char , int *, double);
int h(char a, register int *b, double c)
{
    char *d;

    d = &a;
    h(*d,b,c);
    {
        register int i;

        i = *d;
        return i;
    }
}
```

### ■ オブジェクトリスト

| SCT | OFFSET   | CODE | C LABEL | INSTRUCTION | OPERAND   | COMMENT                                 |
|-----|----------|------|---------|-------------|-----------|---|
| P   | 00000000 |      | _h:     |             |           | ; function: h<br>; <b>frame size=12</b> |
|     | 00000000 | 2FE6 |         | MOV.L       | R14,@-R15 |   |
|     | 00000002 | 4F22 |         | STS.L       | PR,@-R15  |   |

上記の例では関数hの使用するスタック領域のサイズは、オブジェクトリスト中の項目“COMMENT”の“frame size”の値12バイトとなります。

#### (2) 呼び出し関係からの全体スタック容量の算出

関数呼び出しの関係から使用するスタック領域のサイズを算出します。

関数呼び出しの関係からのスタック使用量算出方法の例を図C-3に示します。



図C-3 関数呼び出しの関係とスタック使用量

上記の場合、関数fを介して関数gが呼ばれた場合のスタック領域のサイズは、表C-7に示すようになります。

表C-7 スタックサイズの計算例

| 呼び出し経路                   | スタックサイズ (バイト) |
|--------------------------|---------------|
| main(24) → f(32) → g(24) | 80            |
| main(24) → g(24)         | 48            |

このように、呼び出しレベルの一番深い関数についてスタック容量のサイズを計算し、その最大サイズのスタック領域を最低制限り当てなければなりません。

標準ライブラリ関数を使用する場合には、ライブラリ関数が使用するスタックサイズも考慮する必要があります。標準ライブラリ関数の使用するスタックサイズは、「付録B ライブラリの使用するスタックサイズ一覧」を参照してください。

再帰的に呼び出される関数のスタックサイズは、「関数のスタックサイズ×再帰的に呼び出される回数の最大値」で算出してください。

また、ソースプログラム上ライブラリ関数を使用していなくても、プログラムの実行に必要な実行時ルーチンがリンクされている場合があります。実行時ルーチンの使用するスタックサイズは、次ページのスタック使用量確認方法に示すスタック解析ツールで確認できます。

### <スタック使用量確認方法>

プログラムのリンク時に`-stack`オプションを指定すると、スタック使用量情報ファイルを生成することができます。

リンク時に生成したスタック使用量情報ファイルをコンパイラパッケージ付属のスタック解析ツールで解析することにより、プログラムの全体スタック使用量を求めることができます。

#### ● スタック使用量情報ファイルの生成

プログラムのリンク時に`-stack`オプションを指定すると、スタック使用量情報ファイルを生成します。

スタック使用量情報ファイルは、リンク実行時のカレントディレクトリ下に、拡張子`“.sni”`のファイルが生成されます。

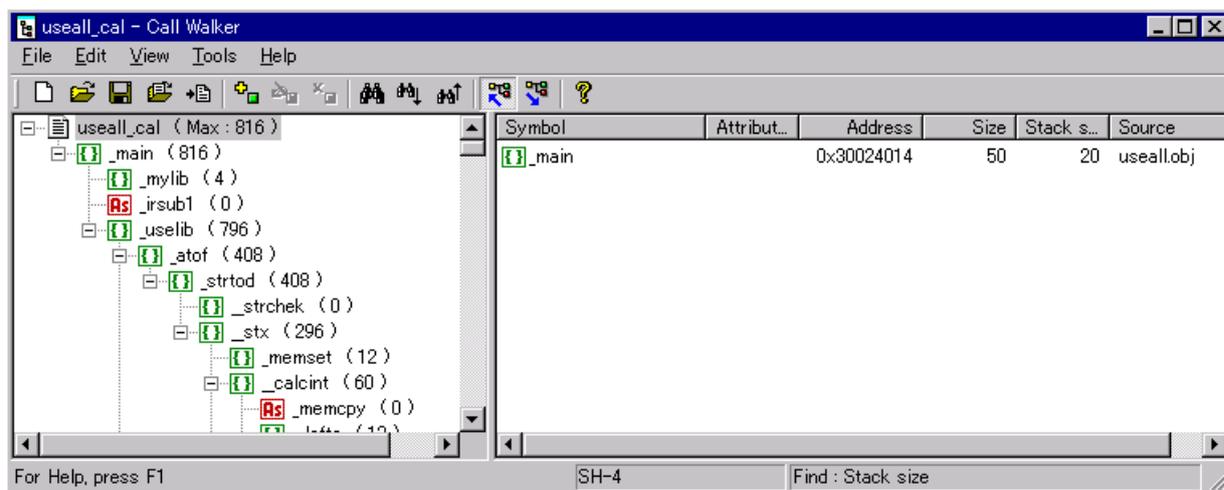
#### ● スタック解析ツール使用方法

以下に示す操作を行うことによりスタック解析ツールを起動し、プログラム、サブプログラムのスタック使用量を表示することができます。

スタック解析ツールの使用方法の詳細は、shcコンパイラ付属のマニュアルおよびスタック解析ツールのヘルプを参照してください。

- ① SHCのVerが7の場合には、Windows®の  ボタンから [Hitachi Embedded Workshop 2] - [Hitachi Call Walker] をクリックし、スタック解析ツールを起動します。SHCのVerが9の場合には、Windows®の  ボタンから [Renesas] - [High-performance Embedded Workshop] - [Call Walker] を選択し、スタック解析ツールを起動します。
- ② スタック解析ツールの [File] メニューから [Import Stack file...] を選択し、表示されたダイアログボックスの“ファイル名”にローダで生成したスタック使用量情報ファイルを指定し、  ボタンをクリックします。

■ スタック解析ツールの表示例



● ロードしたプログラム、サブプログラムのスタック使用量解析時の注意点

スタック解析ツールで算出するスタック使用量では、アセンブラで記述されたプログラム、サブプログラムのスタックサイズは0バイトとして表示されます。このため、RPDPでロードしたプログラム、サブプログラムのスタック使用量をスタック解析ツールで解析する場合には、以下に示す注意事項があります。

■ memcpy()のスタックサイズ

ロードしたプログラム、サブプログラムでは、memcpy()関数はC標準ライブラリではなくCPMSライブラリのmemcpy()関数がリンクされます。スタック解析ツールではCPMSライブラリのmemcpy()が使用するスタックサイズは0として表示されますが、CPMSライブラリのmemcpy()はスタックを28バイト使用します。

スタック解析ツールの [Edit] メニューから [Modify] コマンドを使用してmemcpy()関数のスタックサイズを28バイトに変更し、スタック使用量を再計算してください。

ロードしたプログラム、サブプログラムがmemcpy()を使用しているか否かは、スタック解析ツールの検索機能でmemcpyを検索することで識別することができます。

■ その他アセンブラで記述されたプログラム、サブプログラムのスタックサイズ

アセンブラで記述されたプログラム、サブプログラムのスタックサイズも同様に、スタック解析ツールの [Edit] メニューから [Modify] コマンドを使用してスタックサイズを変更し、スタック使用量を再計算してください。

## 付録D エラーメッセージ

(1/3)

| No. | エラーメッセージ                                    | ユーザの対応   |
|-----|---|--|
| 1   | 指定されたタスクはDORMANT状態でした。                      | タスクの状態を確認し、再試行してください。  |
| 2   | 指定されたタスクはDORMANT状態ではありません。                  | タスクの状態を確認し、再試行してください。  |
| 3   | 指定されたタスクはすでに実行抑止状態です。                       | タスクの状態を確認し、再試行してください。  |
| 4   | 指定されたタスクは実行抑止状態ではありません。                     | タスクの状態を確認し、再試行してください。  |
| 5   | 指定されたタスクは未登録です。                             | タスク番号を見直してください。  |
| 6   | ユーザタスクがRUNNING状態ではありません。                    | ユーザタスクがRUNNING状態ではありません。   |
| 7   | アクセス不可能なアドレスをアクセスしようとしました。                  | CMUのエラーログ内容を確認してください。 (*3)   |
| 8   | アクセス不可能なアドレスを指定しています。                       | ADTの設定アドレスを見直してください。   |
| 9   | プロセッサ間連絡テーブルが満杯でした。他ユーザの使用終了を待ち、再試行してください。  | 他ユーザの使用終了を待ち、再試行してください。  |
| 10  | 指定したタスクはIDLE状態ではありません。タスクの状態を確認し、再試行してください。 | タスクの状態を確認し、再試行してください。  |
| 11  | レジスタの取り出しに失敗しました。                           | システム管理者に連絡してください。 (*1)<br>CMUのエラーログ内容を確認してください。 (*3)                                   |
| 12  | タイマイベントの登録に失敗しました。                          | タイマの登録数が最大数を超えました。タイマの登録数を減らしてください。  |
| 13  | TEXT空間内のアドレスを指定してください。                      | ブレークポイントの設定アドレスを見直してください。  |
| 14  | TCBの取り出しに失敗しました。                            | システム管理者に連絡してください。 (*1)<br>CMUのエラーログ内容を確認してください。 (*3)                                   |
| 15  | ADTはすでに設定されています。ADTを解除してから実行してください。         | ADTを解除してから実行してください。  |
| 16  | ブレーク中のタスクが見つかりませんでした。                       | 他のPCよりブレークポイントが解除されている可能性があります。DHPトレース内容を確認してください。 (*4)<br>タスク状態とブレークポイントの設定を確認してください。 |
| 17  | システムコールのエラーが発生しました。(システムコール名、エラーコード)        | システム管理者に連絡してください。 (*1)<br>CMUのエラーログ内容を確認してください。 (*3)                                   |

| No. | エラーメッセージ                               | ユーザの対応   |
|-----|--|--|
| 18  | DHPトレースデータ読み出し時にエラーを検出しました。            | CMUのエラーログ内容を確認してください。 (*3)<br>システム管理者に連絡してください。 (*1)                           |
| 19  | DHPトレース制御時にエラーを検出しました。                 | CMUのエラーログ内容を確認してください。 (*3)<br>システム管理者に連絡してください。 (*1)                           |
| 20  | タイムアウトが発生しました。                         | ネットワークの接続状態 (IPアドレス、通信ケーブル) を確認し、再試行してください。 (*2)                               |
| 21  | RPCサーバが不在です。                           | ネットワークの接続状態 (IPアドレス、通信ケーブル) を確認し、再試行してください。 (*2)<br>CMUのエラーログ内容を確認してください。 (*3) |
| 22  | RPCサーバが切断されています。                       | ネットワークの接続状態 (IPアドレス、通信ケーブル) を確認し、再試行してください。 (*2)<br>CMUのエラーログ内容を確認してください。 (*3) |
| 23  | コネクションがリセットされています。                     | ネットワークの接続状態 (IPアドレス、通信ケーブル) を確認し、再試行してください。 (*2)<br>CMUのエラーログ内容を確認してください。 (*3) |
| 24  | RPCサーバがクローズしています。                      | ネットワークの接続状態 (IPアドレス、通信ケーブル) を確認し、再試行してください。 (*2)<br>CMUのエラーログ内容を確認してください。 (*3) |
| 25  | 回線ポートがbusy状態です。他ユーザの使用終了を待ち、再試行してください。 | 他ユーザの使用終了を待ち、再試行してください。  |
| 26  | ソケットの生成に失敗しました。                        | CMUのエラーログ内容を確認してください。 (*3)   |
| 27  | メモリ確保に失敗しました。                          | CMUのエラーログ内容を確認してください。 (*3)   |
| 28  | ネットワークが未接続状態です。                        | ネットワークの接続状態 (IPアドレス、通信ケーブル) を確認し、再試行してください。 (*2)                               |
| 29  | ネットワーク接続インタフェースがダウンしています。              | CMUのネットワーク接続インタフェースに異常が発生しました。CMUのエラーログ内容を確認してください。 (*3)                       |
| 30  | ポート番号の取り込みに失敗しました。                     | システム管理者に連絡してください。 (*1)<br>CMUのエラーログ内容を確認してください。 (*3)                           |
| 31  | IPアドレスの取り込みに失敗しました。                    | システム管理者に連絡してください。 (*1)<br>CMUのエラーログ内容を確認してください。 (*3)                           |
| 32  | 共用メモリのアタッチに失敗しました。                     | システム管理者に連絡してください。 (*1)<br>CMUのエラーログ内容を確認してください。 (*3)                           |
| 33  | 致命的なエラーが発生しました。                        | システム管理者に連絡してください。 (*1)<br>CMUのエラーログ内容を確認してください。 (*3)                           |

| No. | エラーメッセージ                         | ユーザの対応   |
|-----|----------------------------------|--|
| 34  | RPCライブラリエラー (rc=RPCライブラリのエラーコード) | ネットワークの接続状態 (IPアドレス、通信ケーブル) を確認し、再試行してください。 (*2)<br>CMUのエラーログ内容を確認してください。 (*3) |

(\*1) ユーザの対応欄が「システム管理者に連絡してください。」となっているエラーに関しては、調査用のデータとDHPトレースを収集してください。収集するデータファイルは、以下のファイルです。

<調査用のデータ>

C:\Hitachi\S10V\DEBUG\Debugger.log (S10V CPMSデバッグのインストール先ドライブがCの場合)

<DHPトレース>

「3.17 DHP記録許可/禁止/表示 < [DHPトレース情報] 画面> (3) 保存 ボタン」にて保存したファイル

(\*2) パソコンの「Ping機能」によるネットワーク接続状態の確認方法

汎用のWindows® 2000, Windows® XP, Windows® 7 (32bit) , Windows® 10 (32bit) パソコンなどを使用して、対象となるCMUのイーサネット接続およびIPアドレス設定の確認をしてください。

IP接続の場合、[Ping] (ピング) コマンドを使って接続に問題がないか確認してください。

- ① Windows®のスタートメニューからコマンドプロンプトを立ち上げます。
- ② [Ping] コマンドを入力し、リンクユニットとパソコン間の基本的な通信テストを実行します。  
[Ping] コマンドはPing [IPアドレス] またはPing [ホスト名] と入力します。

<例 : IPアドレス> Ping 192.192.192. 13

対象のCMUとパソコンとのイーサネット接続が正しく設定されている場合は、以下のメッセージが表示されます。

```
Pinging 192.192.192. 13 with 32 bytes of data
Reply from 192.192.192. 13: bytes=32 time=2ms TTL=32
Reply from 192.192.192. 13: bytes=32 time=1ms TTL=32
Reply from 192.192.192. 13: bytes=32 time=1ms TTL=32
Reply from 192.192.192. 13 : bytes=32 time=1ms TTL=32
C:\WINDOWS>
```

- ③ 正しく接続できていない場合、下記のような表示（タイムアウト）になります。

```
Pinging 192.192.192. 13 with 32 bytes of data:  
Request timed out.  
Request timed out.  
Request timed out.  
Request timed out.  
C:\WINDOWS>
```

(\*3) エラーログの確認方法は、「3.12 エラーログ」を参照してください。

(\*4) DHPトレースの確認方法は、「3.17 DHP記録許可／禁止／表示」を参照してください。

このページは白紙です。