
OAKS16-62P
ユーザーズマニュアル

安全設計に関するお願い

・弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故、火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご留意ください。

本資料ご利用に際しての留意事項

- ・本資料は、お客様が用途に応じた適切な製品をご購入いただくための参考資料であり、本資料中に記載の技術情報についてオクス電子および情報を提供いただいた各社が所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
- ・本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、オクス電子は責任を負いません。
- ・本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、オクス電子は特性改良などにより予告なしに変更することがあります。
- ・本資料に記載の図、表に示す技術的な内容、及びプログラム、アルゴリズムを流用する場合、お客様の責任において実施してください。また、組み込んだプログラム、アルゴリズム単体で評価するだけでなく、システム全体で十分に評価してください。オクス電子は、一切責任を負いません。
- ・本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、オクス電子へご照会ください。
- ・本資料の転載、複製については、文書によるオクス電子の事前の承諾が必要です。
- ・本資料に関し詳細についてのお問い合わせ、その他お気づきの点がございましたらオクス電子までご照会ください。

Microsoft,MS 及び MS-DOS は、米国 Microsoft Corporation の登録商標です。
Windows95,Windows98 は、米国 Microsoft Corporation の商標です。
IBM 及び PC/AT は、米国 International Business Machines Corporation の登録商標です。
Pentium は、米国 Intel Corporation の商標です。
Adobe, Acrobat は、Adobe Systems Incorporated(アドビシステムズ社)の商標です。

はじめに

このたびは、OAKS16-62P をお買い上げいただきまして誠にありがとうございます。このマニュアルは、OAKS16-62P に含まれるハードウェアおよびソフトウェアのセットアップ方法、使用上の注意点について述べたものです。ハードウェアの内容、ソフトウェアの操作方法については各マニュアルを参照ください。

目次

目次	4
1 . 製品パッケージの内容.....	6
1 . 1 . 包装製品一覧表.....	6
1 . 2 . CDROM.....	6
1 . 3 . ソフトウェア製品	6
1 . 4 . 添付部品表.....	7
1 . 5 . 別途ご用意いただくもの	7
2 . 保証ならびにサポートについて	8
2 . 1 . 保証.....	8
2 . 2 . サポート.....	8
3 . システム構成	9
3 . 1 . ホストコンピュータ.....	9
3 . 2 . OAKS16-M30626FHPFP	10
3 . 3 . OAKS16-EXBOARD.....	10
3 . 4 . リモートデバッグ (KD30)	10
3 . 5 . コンパイラ (NC30WA)	10
3 . 6 . 統合開発環境 (TM)	10
3 . 7 . フラッシュメモリ書き込みソフト (Flashstater)	10
4 . ハードウェアのセットアップ.....	11
4 . 1 . ボードの準備.....	11
4 . 2 . CPUボードの取付け.....	14
4 . 3 . パソコンとの接続.....	15
5 . ソフトウェアのセットアップ.....	16
5 . 1 . 動作環境.....	16
5 . 2 . リモートデバッグ (KD30) のセットアップ	17
5 . 2 . 1 . KD30 のインストール方法.....	17
5 . 2 . 2 . KD30 のアンインストール方法.....	18
5 . 3 . コンパイラ (NC30WA) のセットアップ.....	19
5 . 3 . 1 . NC30WA のインストール方法.....	19
5 . 3 . 2 . NC30WA のアンインストール方法.....	23
5 . 4 . 1 . インストール方法	24
5 . 4 . 2 . アンインストール方法	24
5 . 4 . 3 . 内蔵フラッシュメモリ書き込み SW を使用する前に.....	24
5 . 4 . 4 . 内蔵フラッシュメモリ書き込み SW の使用方法.....	24
5 . 5 . TM (ツールマネージャー)	25
5 . 5 . 1 . TM のインストール方法.....	25
5 . 5 . 2 . TM のアンインストール方法.....	29
5 . 6 . 電子マニュアルの参照.....	29
6 . プログラム開発手順	30
6 . 1 . sample プログラムの内容.....	30
6 . 2 . DOS プロンプト上でコンパイルする。	31
6 . 2 . 1 . 準備.....	31
6 . 2 . 2 . AS30 の起動.....	32
6 . 2 . 3 . NC30 の起動.....	33
6 . 2 . 4 . LN30 の起動	34
6 . 3 . TMでプロジェクトを作成しビルドする。	35
6 . 3 . 1 . make について.....	35
6 . 3 . 2 . 準備.....	38
6 . 3 . 3 . TMを起動する.....	38
6 . 3 . 4 . プロジェクトを作成する	39
6 . 3 . 5 . ファイルの登録.....	41
6 . 3 . 6 . モトローラSフォーマットファイルの作成	42
6 . 4 . KD30での実行.....	44
6 . 4 . 1 . KD30 の起動.....	44

6.4.2.プログラムのダウンロード.....	46
6.4.3.プログラムの実行.....	47
6.5.フラッシュ ROM への書き込み.....	48
6.5.1.準備.....	49
6.5.2.Flashstater.exe の起動.....	49
6.5.3.書き込み.....	51
6.5.4.プログラムの実行.....	53
6.5.5.ID コードについて.....	54
6.5.5.OAKS16-62P ID コードのついで注意事項.....	55
7.使用上の注意事項.....	56
7.1.コンパイラの制限事項.....	56
7.1.1.使用できないオプション.....	56
7.1.2.スタートアップファイルの注意.....	57
7.2.リモートデバッガの制限事項.....	57
7.2.1.ストップモード、ウエイトモードに関する制限事項.....	57
7.2.2.ユーザープログラムのリアルタイム性について.....	57
7.2.3.例外的なステップ実行について.....	57
7.3.割り込みに関する制限事項.....	58
7.4.周辺機能に関する制限事項.....	59
7.5.レジスタ操作に関する制限事項.....	60
7.6.フラグレジスタに関する制限事項.....	61
7.7.メモリ拡張モード使用時の注意点.....	61
8.よくある質問.....	62
8.1.KD30 起動時に通信エラーが発生するのですが？.....	62
8.2.デバッグ中に通信エラーが発生したのですが？.....	62
8.3.「ソースファイルが見つかりません」と出たのですが？.....	62
8.4.ブレーク中の周辺 I/O の動作はどうなるの？.....	62
9.付録.....	63
9.1.メモリマップ.....	63
9.2.部品表.....	64
10.改定履歴.....	65

1. 製品パッケージの内容

OAKS16-62PFullKit 製品パッケージの包装内容を示します。開封時に包装内容をご確認下さい。

1.1. 包装製品一覧表

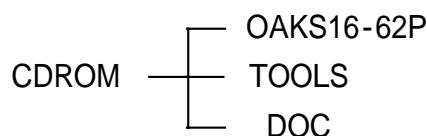
本製品の包装内容を、表 1-1 に示します。

表 1-1 OAKS16-62P-FullKit 包装内容一覧表

製品名	数量	備考
OAKS16-M30626FHPFP	1 枚	M16C/62P マイコンボード
OAKS16-EXBOARD	1 枚	学習用拡張基板（非完成品）
拡張ボード用部品	1 組	組み立て部品一式
電源	1 個	9V 500mA
包装内容説明書	1 部	
CDROM	1 枚	各種マニュアル コンパイラ NC30WA リモートデバッガ KD30 内蔵フラッシュメモリ書き込み S/W FlashStart ツールマネージャー TM 各種ドキュメント

1.2. CDROM

CDROM にはプログラム開発に必要なソフトウェア製品、電子マニュアルなどが含まれています。以下に CDROM の構成を示します。



1.3. ソフトウェア製品

本製品の電子マニュアルは PDF ファイルで提供しています。電子マニュアルを参照するためには、Acrobat Reader4.0 が必要となります。パソコン雑誌付録 CDROM 等からインストールするか、Adobe Systems 社のサイト <http://www.adobe.co.jp/> からダウンロードしてご覧ください。また図面等が見えにくい場合は拡大してご覧ください。

表 1-2 付属 CDROM 内容一覧表

ディレクトリ	内容
OAKS16-62P	OAKS16-62P キットのマニュアル類、sample プログラム等を格納しています。 最初にお読み下さい。 （PDF ファイル）
TOOLS	コンパイラ、デバッガ、ツールマネージャー、内蔵フラッシュメモリ書き込み S/W の各ファイルが格納されています。
DOC	M16C/62P に関するマニュアル等が格納されています。

1.4. 添付部品表

次のような部品が添付されています。

表 1-3 添付部品表

記号	型番	メーカー	数量	備考
PCB		オクス電子	1	OAKS16-EXBOARD-1
Q1	NJM7805FA	NJR	1	3端子レギュレータ(相当品)
LED1-3	SEL1110R	サンケン	3	LED(相当品)
C1,2	RPE132F104Z50	村田	2	積層セラミックコンデンサ(相当品)
C3	SS1C475M	エルナ	1	(16V4.7 μ)タンタルコンデンサ(相当品)
R1-3	RD16S-1K	コーア	3	炭素皮膜抵抗(相当品)
SW1-4	MS-611A	ミヤマ	4	トグルスイッチ(相当品)
SW5-7	TMEG1-01	フジソク	3	プッシュスイッチ(相当品)
SW8	KSD-08	OTAX	1	8素子ディップスイッチ(相当品)
CN1,3	OX-114-DS-40G	OAKS	2	コネクタ(相当品)
CN2	OX-114-DS-20G	OAKS	1	コネクタ(相当品)
CN4	OX-114-SS-6G	OAKS	1	コネクタ(相当品)
CN5	OX-114-SS-2G	OAKS	1	コネクタ(相当品)
J1	HEC3100	ホシデン	1	DCジャック(相当品)
J2	OX-107-09STK	OAKS	1	コネクタ(相当品)
放熱板	16P-16	吉川	1	三端子レギュレータ用放熱版(相当品)
ねじ			1	3 \times 8 + ナベ(相当品)
DC			1	9V 500mA DCアダプタ
ゴム足			4	
CN6	HIF3FC-14PA-2.54DSA	ヒロセ	1	E8用コネクタ(相当品:未実装)

部品表にあるメーカーの製品が添付されるとは限りません。その場合は相当品が添付されます。

1.5. 別途ご用意いただくもの

以下の物品については別途ご用意下さい。

- ・パーソナルコンピュータ

表 1-4 PCの内容

ホストマシン	IBM PC/AT シリーズおよびその互換機
OS	Microsoft Windows95、98、NT、me、2000、XP
CPU	486DX4-100MHz または Pentium75MHz 以上を推奨
メモリ	8M バイト以上 (推奨 16M バイト以上)

- ・RS232C ケーブル：9pin オスメス型ストレートケーブル
推奨：D9PXS-1.5m @1,450.- 神保商会

注：USB シリアルケーブルをご使用いただく場合、製品によってはうまく通信できない場合がございます。

- 推奨：Arvel 製 SRC06-USB

2 . 保証ならびにサポートについて

2 . 1 . 保証

本キットは評価用という位置付けですので、欠品、破損、初期不良時の無償交換のみのサポートとさせていただきます。それ以外の保証については行いません。

2 . 2 . サポート

本キットに関する電話によるサポートは一切お受けできません。サポートならびにOAKS16-62P-FullKitの最新情報についてはオークス電子ホームページを照会してください。

<http://www.oaks-ele.com/>

3 . システム構成

OAKS16-62P-FullKit を使用するためには、以下の装置が必要です。

- 1 . ホストコンピュータ (別途ご用意願います)
- 2 . OAKS16 -M30626FHPFP (付属)
- 3 . OAKS16-EXBOARD (付属)
- 4 . シリアルケーブル (別途ご用意願います)
- 5 . リモートデバッグ (KD30) (付属)
- 6 . コンパイラ (NC30WA) (付属)
- 7 . 統合開発環境 (TM)
- 8 . フラッシュROM書き込みソフト (flashstater)
- 9 . 電源 (付属)

OAKS16-62P-FullKit のシステム構成図を図 3-1 に示します。

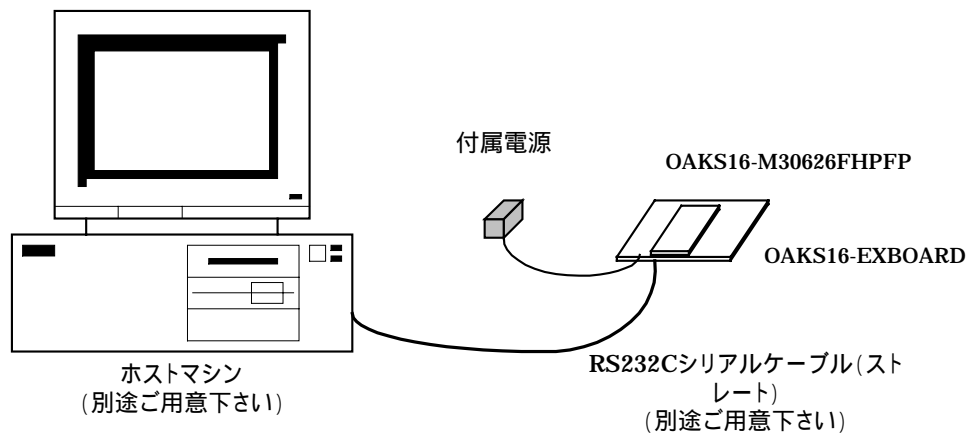


図 3-1OAKS16-62P-FullKit システム構成図

3 . 1 . ホストコンピュータ

KD30 および NC30WA は、表 3-1 のホストコンピュータ上で動作します。

表 3-1 ホストコンピュータ

ホストコンピュータ名	OS
IBM PC/AT シリーズ およびその互換機	Microsoft Windows 95 Microsoft Windows 98 Microsoft Windows NT Microsoft Windows me Microsoft Windows 2000 Microsoft Windows XP

3.2. OAKS16-M30626FHPFP

OAKS16-M30626FHPFP は、株式会社ルネサステクノロジ製 16 ビットシングルチップマイクロコンピュータ M16C/62P フラッシュメモリ内蔵版 (M30626FHPFP) を搭載した基板です。内蔵フラッシュメモリにはモニタプログラムが書き込まれています。

3.3. OAKS16-EXBOARD

OAKS16-EXBOARD は、OAKS16-M30626FHPFP 用拡張基板です。ホストコンピュータを接続するコネクタ、電源回路、I/O 制御学習のための回路を備えています。部品は半田付けされていないため、マニュアルを参照の上完成させてください。

3.4. リモートデバッガ (KD30)

KD30 は、ホストコンピュータ上で動作するソフトウェアです。KD30 はルネサス 16 ビットシングルチップマイクロコンピュータ M16C/62P フラッシュメモリ内に書き込まれているモニタプログラムと通信を行い、高機能なデバッグ環境を提供します。KD30 はルネサス 16 ビットマイコン用エミュレータコントロール用デバッグソフトウェア (PD30) と操作面で互換性を持っています。以下に KD の特徴を示します。

- 1) アセンブリ言語・構造化アセンブリ言語・C 言語のソースラインデバッグができます。
- 2) 同時に 2 点のパスカウント付きソフトウェアブレイクポイントが設定できます。
ただし、ハードウェアブレイクの設定は出来ません。
- 3) KD30 実行中に RAM の変化をウォッチできます。(RAM モニタ機能)
RAM モニタ機能は一定周期にダンプコマンドを実行しています。
- 4) C 言語の変数の参照ができます。構造体変数、共用体変数ではメンバ変数も同時に参照できます。
- 5) OAKS16-62P に実装できる外付け発振子は 6MHz です。それ以外の外付け発振子は使用できません。

3.5. コンパイラ (NC30WA)

本製品に付属しているコンパイラは NC30WA(オークス版)になります。NC30WA(オークス版)は C 言語・アセンブリ言語ソースプログラムからデバッグ情報ファイルを生成します。

NC30WA (オークス版) の簡易使用を以下に示します。

- 1) プログラムサイズに制限はありません。
- 2) コンパイルするファイルライン数に制限はありません。
- 3) 正規版のコンパイラである NC30WA に対し、一部機能制限があります。機能制限については「7. 使用上の注意事項」を参照して下さい。

3.6. 統合開発環境 (TM)

統合化開発環境は、コンパイラ/アセンブラ/デバッガ/エディタなどのツール群を共通グラフィカルユーザーインターフェース(GUI)に統合して、ソフトウェアの開発効率を改善するためのツールです。

3.7. フラッシュメモリ書き込みソフト (Flashstater)

CPU内部の書き換えブートプログラムと通信を行い、CPU内蔵のフラッシュメモリを書き換えるためのソフトウェアです。

扱う機械語ファイルはモトローラS2形式 (.mot) のファイルです。

書き換えブートプログラムは内蔵フラッシュメモリとは違う領域に、CPU出荷時に書き込まれています。

4 . ハードウェアのセットアップ

4 . 1 ボードの準備

EXBOARD(拡張基板)に部品を取付けます。
(写真 4-1) D - S u b コネクタ、電源回路は、パターン上で接続されていますので、部品を差し込んで、ハンダ付けするだけです。それ以外のスイッチ L E D は、必要に応じてポートに接続してください。

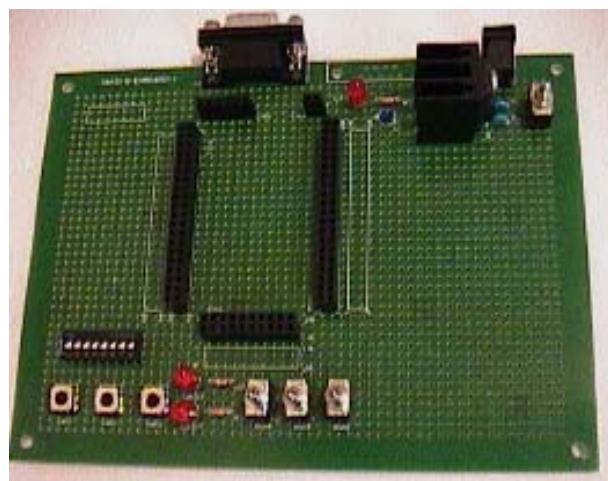


写真 4-1 部品実装済みのボード

...ご注意...

ボードに部品をハンダ付けする場合、通常は背の低い部品から取り付けていくのが基本ですが、ここでは、説明の都合上パートごとに組み立ての説明をしていきます。

(ア) C P Uボードを接続するコネクタを取付けます。
C N 1 から C N 5 まで基板の白い表示 (シルク) のところにコネクタ (写真 4-2) を取付けてください。

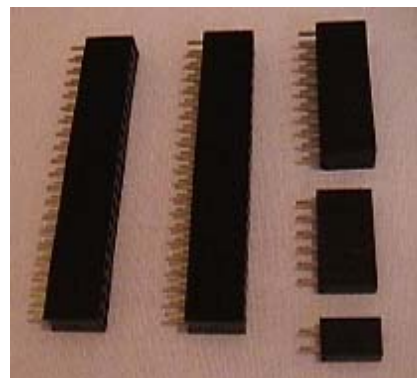


写真 4-2 コネクタ

回路図（図 4-1）を参考にして電源回路を取付けます。

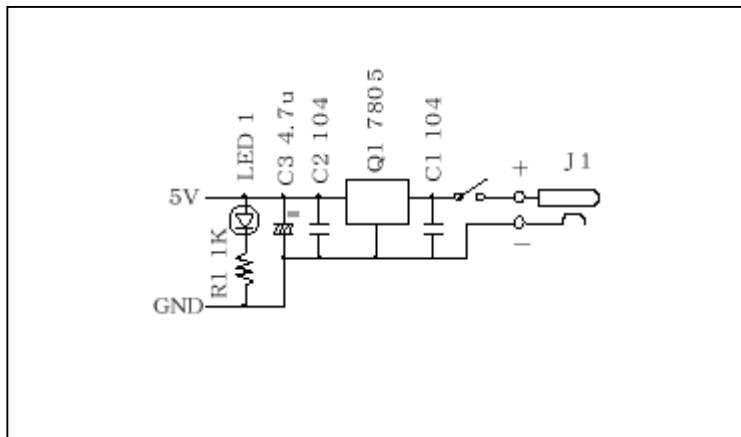


図 4-1

積層セラミックコンデンサ、抵抗（写真 4-3）を取付けます。これらは、部品の向きを気にしないで、基板の白い表示（C 1 , C 2 , R 1）の場所に取り付けます。



写真 4-3 コンデンサ、抵抗

LED、タンタル電解コンデンサ（写真 4-4）を取付けます。これらは、取付ける向きに注意が必要です。LEDは足の短い方がカソードです。足の短い方を抵抗の隣の穴に入れてください。電解コンデンサは、足の長い方が+です。足の長い方を基板の白い表示 C 3 の + のところに差し込んでください。

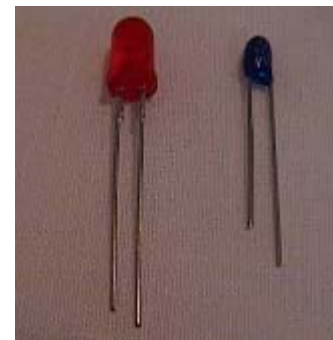


写真 4-4 LED , コンデンサ

電源スイッチ（写真 4-5）、DCジャック（写真 4-6）を取付けます。基板の白い表示（S W 1、J 1）の位置に取付けてください。



写真 4-5
電源スイッチ



写真 4-6
DCジャック

三端子レギュレータ（写真 4-7）を取付けます。基板につける前に、放熱板のねじをはずし、三端子レギュレータを放熱板に取付けます。そのあと、三端子レギュレータの足をQ 1の穴に差し込んでハンダ付けしてください（図 4-2）。



三端子レギュレータ（右）
放熱板（左） 写真 4-7

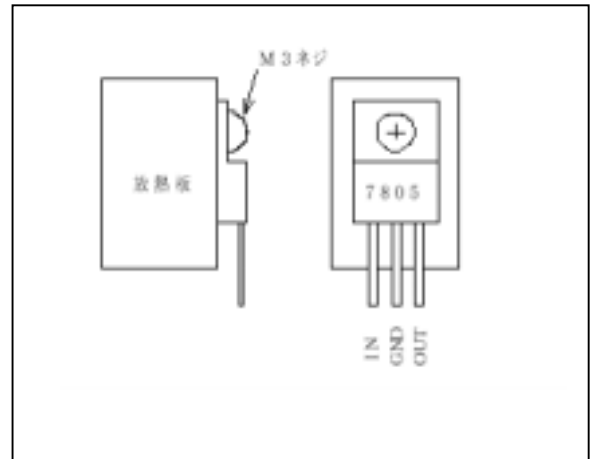


図 4-2 放熱板取り付け図

（ウ）D__SUBコネクタ（写真 4-8）をJ 2に取付けます。



写真 4-8 D__SUBコネクタ

（エ）その他のLED、スイッチ、抵抗（写真 4-9）を取付けます。
ここにはパターンでの配線はされていないので必要に応じて接続してください。

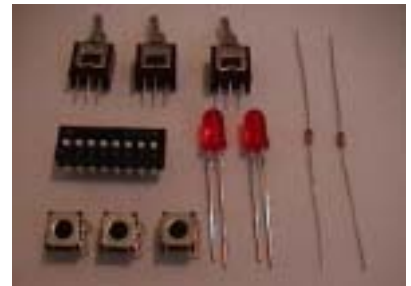


写真 4-9 その他の部品

4.2. CPUボードの取付け

コネクタの番号、向きに注意して M30626FHPFP (CPUボード) を挿し込んで下さい。(写真 4-10)



図 4-10 CPUボード取付け

4.3. パソコンとの接続

D_SUBコネクタにRS232Cシリアルケーブルを接続しパソコンに接続します。(図4-11)

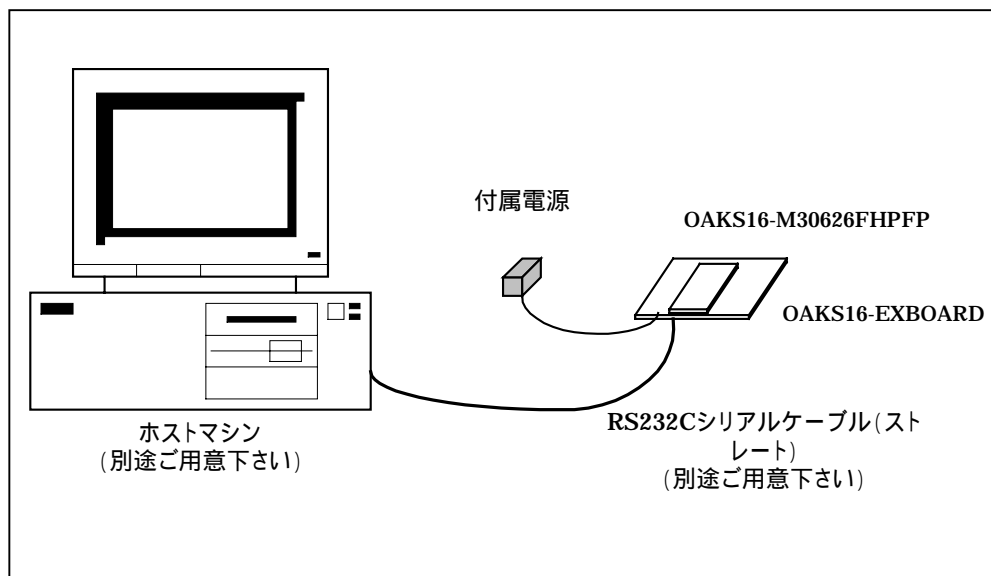


図 4-11

5 . ソフトウェアのセットアップ

5 . 1 . 動作環境

OAKS16-62P-FullKit に付属の各ソフトウェアは、表 5-1 に示すホストマシンおよび OS バージョン上で動作します。

表 5-1 動作環境

ホストマシン	IBM PC/AT シリーズおよびその互換機
OS	Microsoft Windows 95、98、NT、me、2000、XP
CPU	CPU 486DX4 -100MHz または Pentium 75MHz 以上を推奨
メモリ	メモリ 8M バイト以上（推奨 16M バイト）

NC30WA の Windows での動作について

NC30WA は Microsoft Windows の DOS プロンプトで動作します。

NC30WA および KD30 の Windows での動作について

ファイル名に特殊文字（スペース記号含む）を含めることはできません。また、ネットワークドライブ名を使用することはできませんので、ご使用の際はドライブに割り当てをおこなってください。

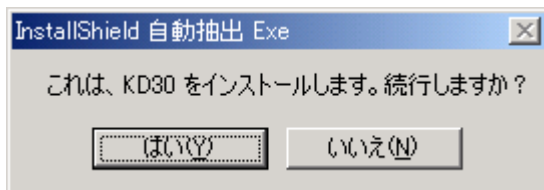
5.2. リモートデバッガ (KD30) のセットアップ

5.2.1. KD30 のインストール方法

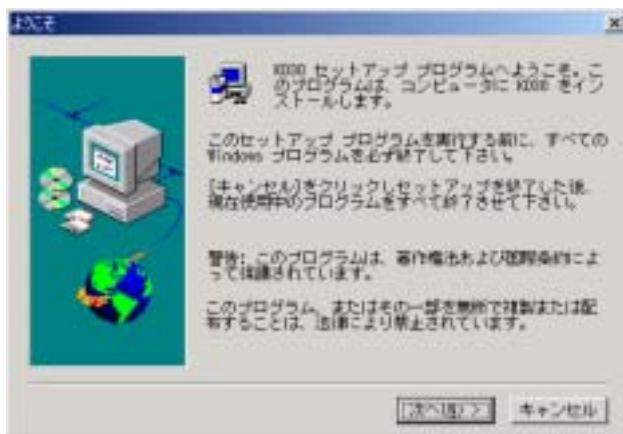
インストール

KD30 は CD-ROM の TOOLS\KD30\KD30V410R1_J_20041203.exe をダブルクリックして下さい。

2) 次頁にインストール手順の流れを明記します。



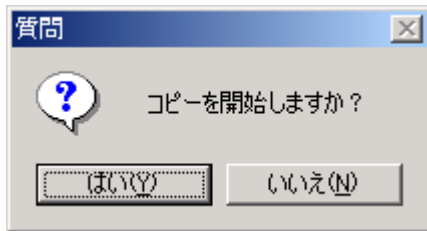
<はい> をクリックして下さい。



<次へ> をクリックして下さい。



<次へ> をクリックして下さい。



コピーする場合にのみ、<はい>をクリックして下さい。



<OK>をクリックして下さい。

3) 以上で、インストールは完了です。

5.2.2. KD30 のアンインストール方法

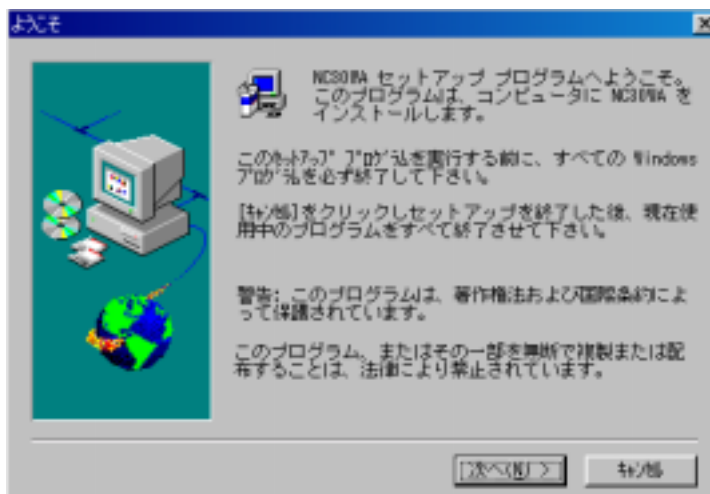
アンインストールするには「スタート」 - 「設定」 - 「コントロールパネル」を開き、「アプリケーションの追加、削除」をクリックします。プログラムリストから「KD30」を選択し、「追加と削除」ボタンをクリックします。アンインストールウィンドウが開き、KD30 がアンインストールされます。

5.3. コンパイラ (NC30WA) のセットアップ

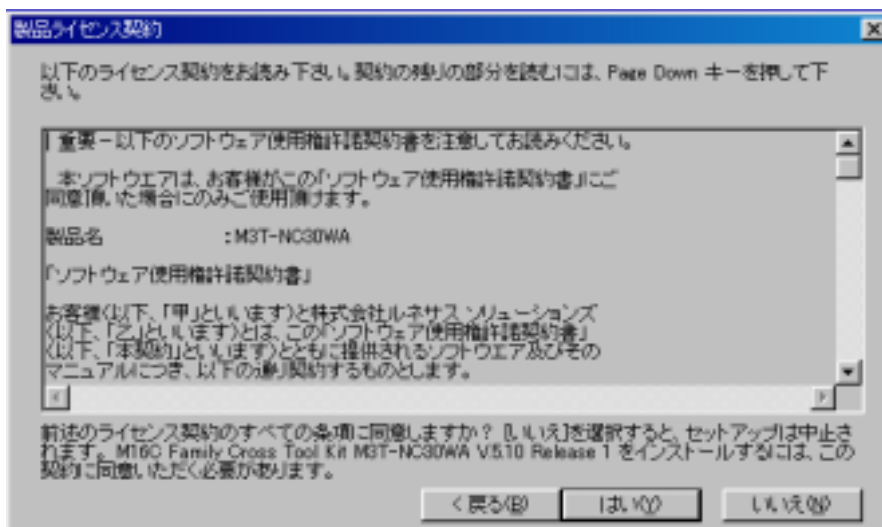
5.3.1. NC30WA のインストール方法

インストールする前に下記の点にご注意下さい。
他のNC30WA(体験版など)がインストールされているコンピュータに、NC30WA オークス版をインストールする場合は、**必ず、すでにインストールされているNC30WAをアンインストールした後で**、NC30WA オークス版をインストールしてください。(特に、同じバージョンを同じコンピュータにインストールすることは避けてください)

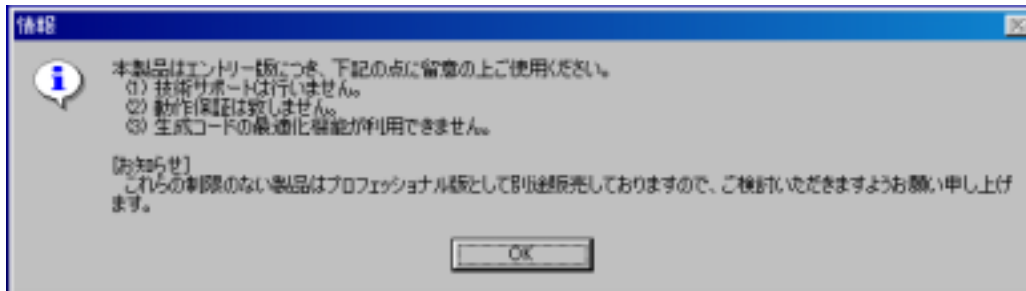
CD-ROM の TOOL\NC30WA\SETUP\setup.exe をダブルクリックして下さい。
次の画面が表示されます。



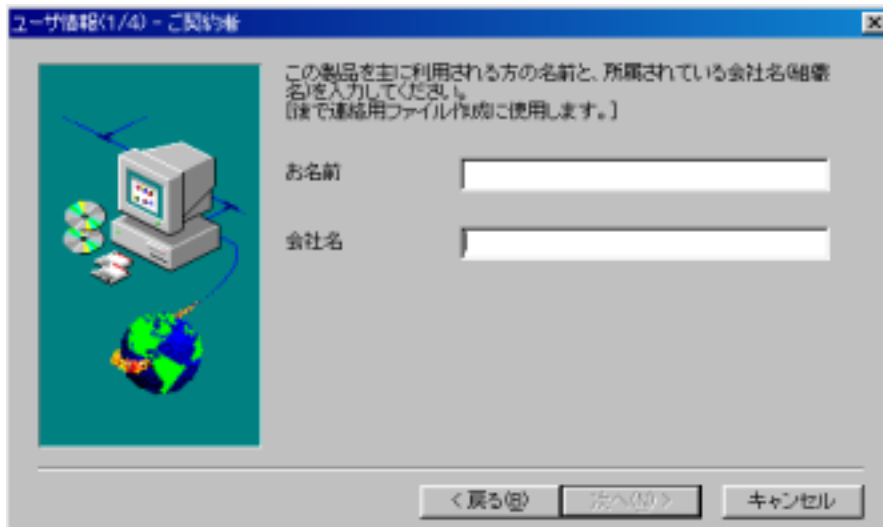
<次へ>をクリックして下さい。



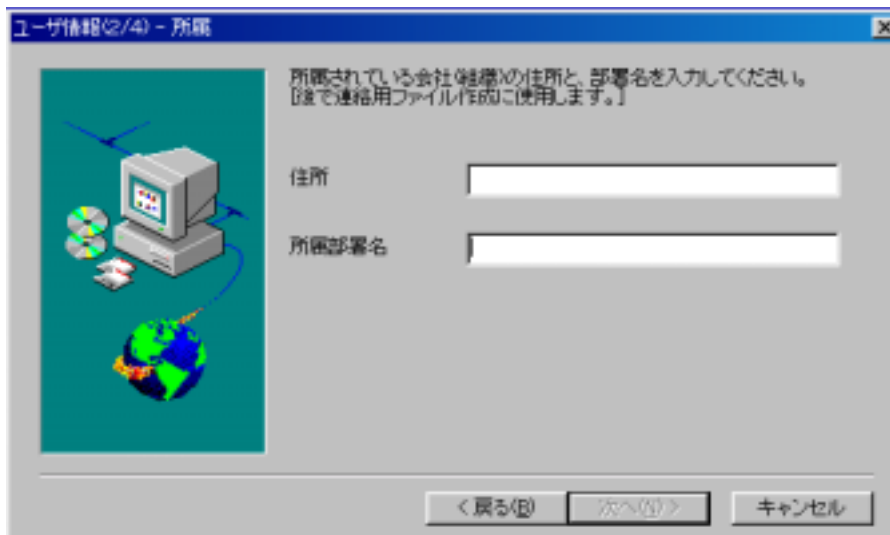
同意する場合のみ <はい> をクリックして下さい。



< OK > をクリックして下さい。



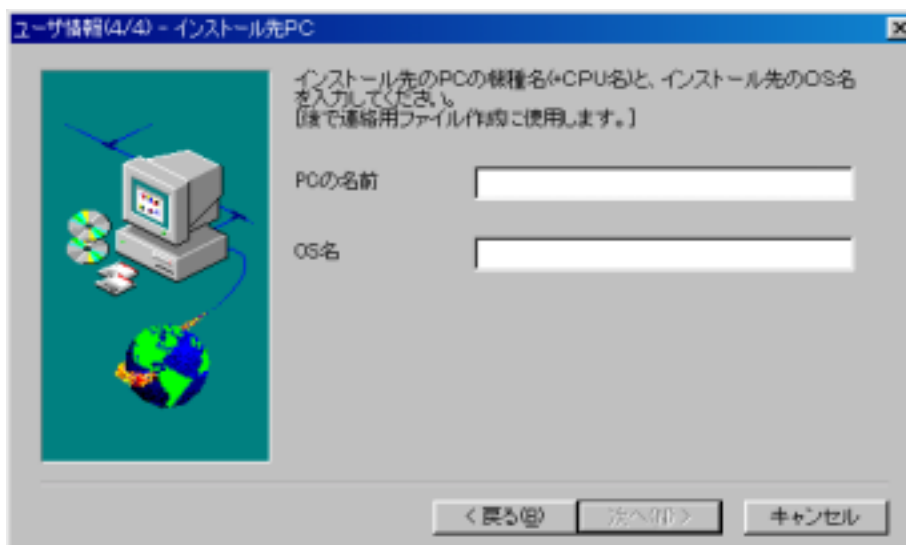
項目欄を入力して、< 次へ > をクリックして下さい。



項目欄を入力して、< 次へ > をクリックして下さい。



項目欄を入力して、<次へ>をクリックして下さい。



インストール先の PC の機種を入力して、<次へ>をクリックして下さい。



内容を十分に確認した上で、<次へ>をクリックして下さい。



インストール先を選択して、<次へ>をクリックして下さい。



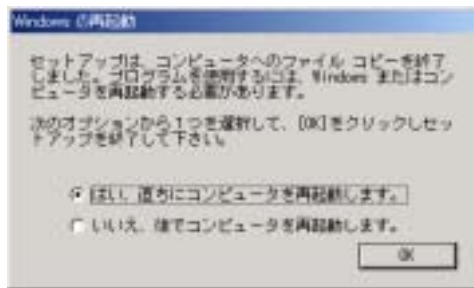
“AUTOEXEC.BAT ファイルを変更する”を選択して、<次へ>をクリックして下さい。



<はい>をクリックして下さい。



<OK>をクリックして下さい。



“はい、直ちにコンピュータを再起動します”を選択して、<OK>をクリックして下さい。

4) 以上でインストール完了です。

5.3.2. NC30WA のアンインストール方法

アンインストールするには「スタート」 - 「設定」 - 「コントロールパネル」を開き、「アプリケーションの追加、削除」をクリックします。プログラムリストから「NC30WA V.x.xx」を選択し、「追加と削除」ボタンをクリックします。アンインストールウィンドウが開き、NC30WA がアンインストールされます。

5.4. 内蔵フラッシュメモリ書き込みプログラム (FlashStart)

5.4.1. インストール方法

- 1) 任意のディレクトリを作成します。
- 2) CD-ROM の \tool\Flashsta ディレクトリ内すべてのファイルを作成したディレクトリにコピーします。

5.4.2. アンインストール方法

内蔵フラッシュメモリ書き込み S/W(flashstart)のアンインストールするには「インストール方法」で作成したディレクトリ内をすべて削除します。

5.4.3. 内蔵フラッシュメモリ書き込み SW を使用する前に

既に KD30 と通信を行うモニタプログラムがマイコンの内蔵フラッシュメモリ領域に書き込まれています。内蔵フラッシュメモリ書き込み S/W を使用し、内蔵フラッシュメモリにユーザプログラムを書き込むと、内蔵フラッシュメモリに書き込まれていたモニタプログラムが消去されます。再度 KD30 を使用してデバッグを行う場合は、モニタプログラムを書き込む必要があります。モニタプログラムは、CD-ROM の下記の場所に格納されています。

TOOL\MONITOR\Mon_Uart.mot

5.4.4. 内蔵フラッシュメモリ書き込み SW の使用方法

内蔵フラッシュメモリ書き込み S/W を使用した書き込み手順を以下に明記します。

- 1) ユーザーボード上で CNVss を”H”レベルにして下さい。
- 2) ユーザーボードに電源を投入し、Reset ボタンを押します。
- 3) 内蔵フラッシュメモリ書き込み S/W(flashstaer)を起動して下さい。この時、1) の設定がされていないと通信エラーが発生します。
- 4) 書き込むユーザプログラムを選択して下さい。(注意)
- 5) E.P.R ボタンを押して下さい。(Erase、Program、Read を行います)
- 6) 書き込みが完了したら”OK”ボタンが表示されますので、ボタンを押して内蔵フラッシュメモリ書き込み S/W(flashstart)を終了して下さい。
- 7) ユーザーボードの電源を切ります。
- 8) 1) で”H”レベルにしていた CNVss 端子を”L”レベルにして下さい。
- 9) 以上で内蔵フラッシュメモリへの書き込みは完了です。

注意) ID コードを設定する時、特定の指定 ID コードがなければ、全て”00H”を入力して下さい。モニタプログラムを消去して、ユーザプログラムを書き込む場合、ID コードは全て”00H”を入力して下さい。再度モニタプログラムを書き込む場合は、ID コードファイルが付属していますので ID コードの入力は不要です。又、操作方法については付属のマニュアルを参照下さい。また、モニタプログラムの ID コードは”00h”になります。

5.5. TM (ツールマネージャー)

TM (ツール・マネージャー) は、コンパイラ / アセンブラ / デバッガ / エディタなどのツール軍を共通グラフィカルユーザーインターフェース (GUI) に統合して、ソフトウェアの開発効率を改善する為のツールです。

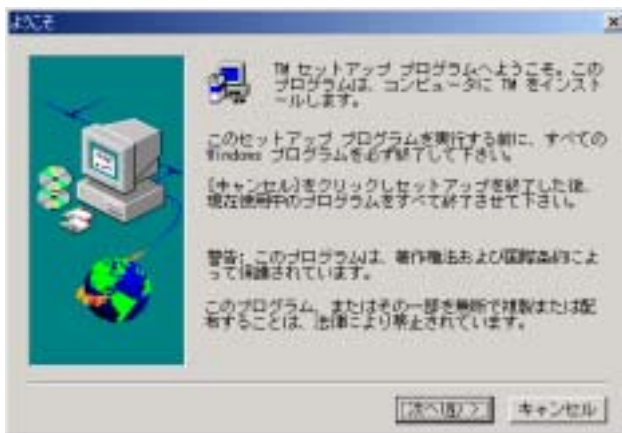
TM に、エディタ、コンパイラ、デバッガ、フラッシュROM書き込みソフトなどのツールを登録することにより、TM のツールバーから起動させることができるようになります。さらに、コンパイラの起動を make で行っているため、効率的な開発ができます。

5.5.1. TM のインストール方法

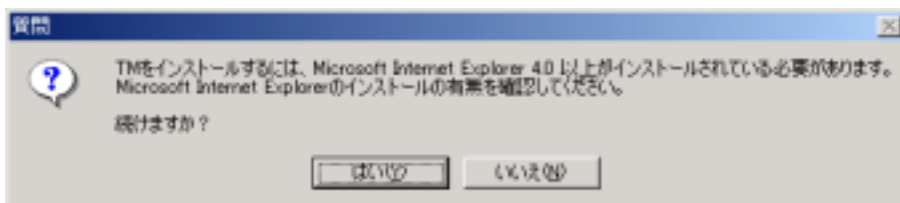
1) インストール

- ・統合化開発環境は CD-ROM の TOOLS\TM\SETUP\setup.exe をダブルクリックして下さい。
- ・漢字、空白文字を含むディレクトリ名、ファイル名は使用できません。
- ・ネットワークパス名は使用できません。ドライブ名を割り当ててご使用ください。
- ・ショートカットは使用できません。
- ・パス指定を含め 128 文字以上になるディレクトリ名、ファイル名は使用できません。
- ・インストール中の「コンポーネントの選択」画面で、「インストール先ディレクトリ」項目を「参照(R)」を使用して変更するとき、「ディレクトリの選択」画面の「ドライブ(V)」で表示されるドライブのいくつかが選択できなくなることがあります。この場合は「コンポーネントの選択」画面に戻り、その中の「ディスク容量(S)」を使ってドライブ指定を行ってください。それでもうまく行かないときは、Windows を再起動して再度インストーラを実行してください。

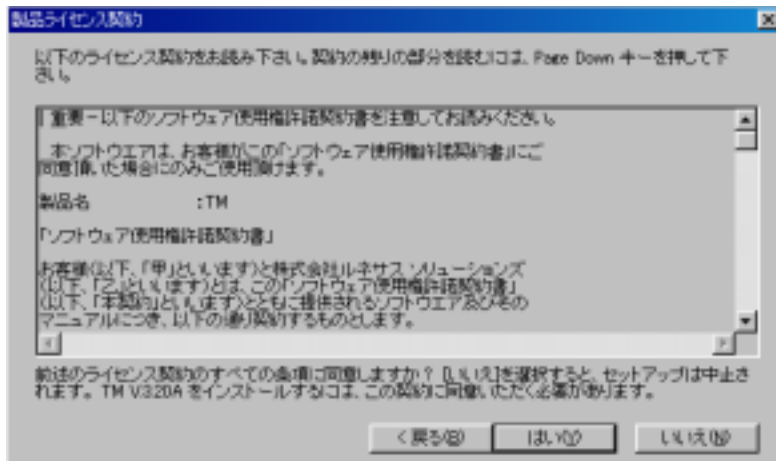
2) 次頁にインストール手順の流れを明記します。



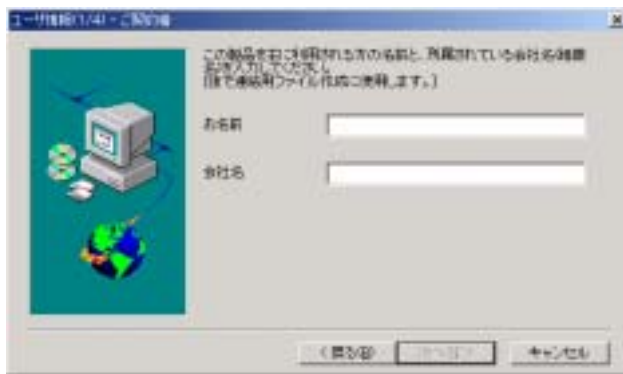
<次へ> をクリックして下さい。



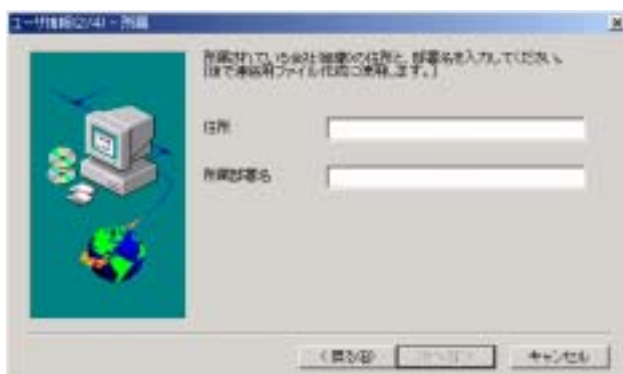
<はい> をクリックして下さい。



同意する場合のみ<はい>をクリックして下さい。



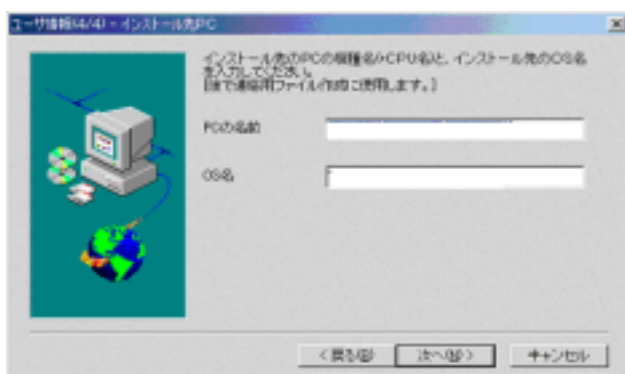
項目欄を入力して、<次へ>をクリックして下さい。



項目欄を入力して、<次へ>をクリックして下さい。



項目欄を入力して、<次へ>をクリックして下さい。



インストール先の機種を入力して
<次へ>をクリックして下さい。



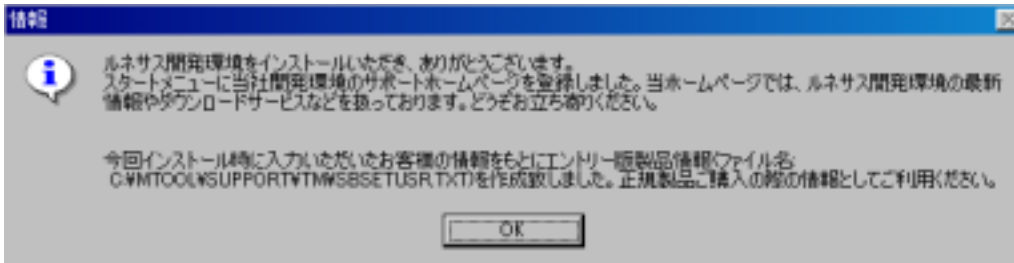
内容を十分に確認した上で、<次へ>をクリックして下さい。



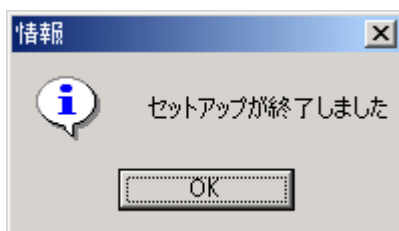
インストール先を選択して、<次へ>をクリックして下さい。



コピーする場合にのみ、<はい>をクリックして下さい。



<OK>をクリックして下さい。



<OK>をクリックして下さい。

3) 以上で、インストールは完了です。

5.5.2. TMのアンインストール方法

アンインストールするには「スタート」 - 「設定」 - 「コントロールパネル」を開き、「アプリケーションの追加、削除」をクリックします。プログラムリストから「TM」を選択し、「追加と削除」ボタンをクリックします。アンインストールウィンドウが開き、TMがアンインストールされます。

TMの詳しい使用方法は、弊社 HP

http://www.oaks-ele.com/OAKS16/text/download_new.htm

より、「OAKS16でTMをお使いになる方のために」をダウンロードしてご覧ください。

5.6. 電子マニュアルの参照

本製品の電子マニュアルはPDFファイルで提供しています。電子マニュアルを参照するためには、Acrobat Reader4.0が必要となります。パソコン雑誌付録CDROM等からインストールするか、Adobe Systems社のサイト<http://www.adobe.co.jp/>からダウンロードしてご覧ください。

6 . プログラム開発手順

ここでは Sample プログラムを使用して、各ツールの操作方法及び開発手順を説明していきます。

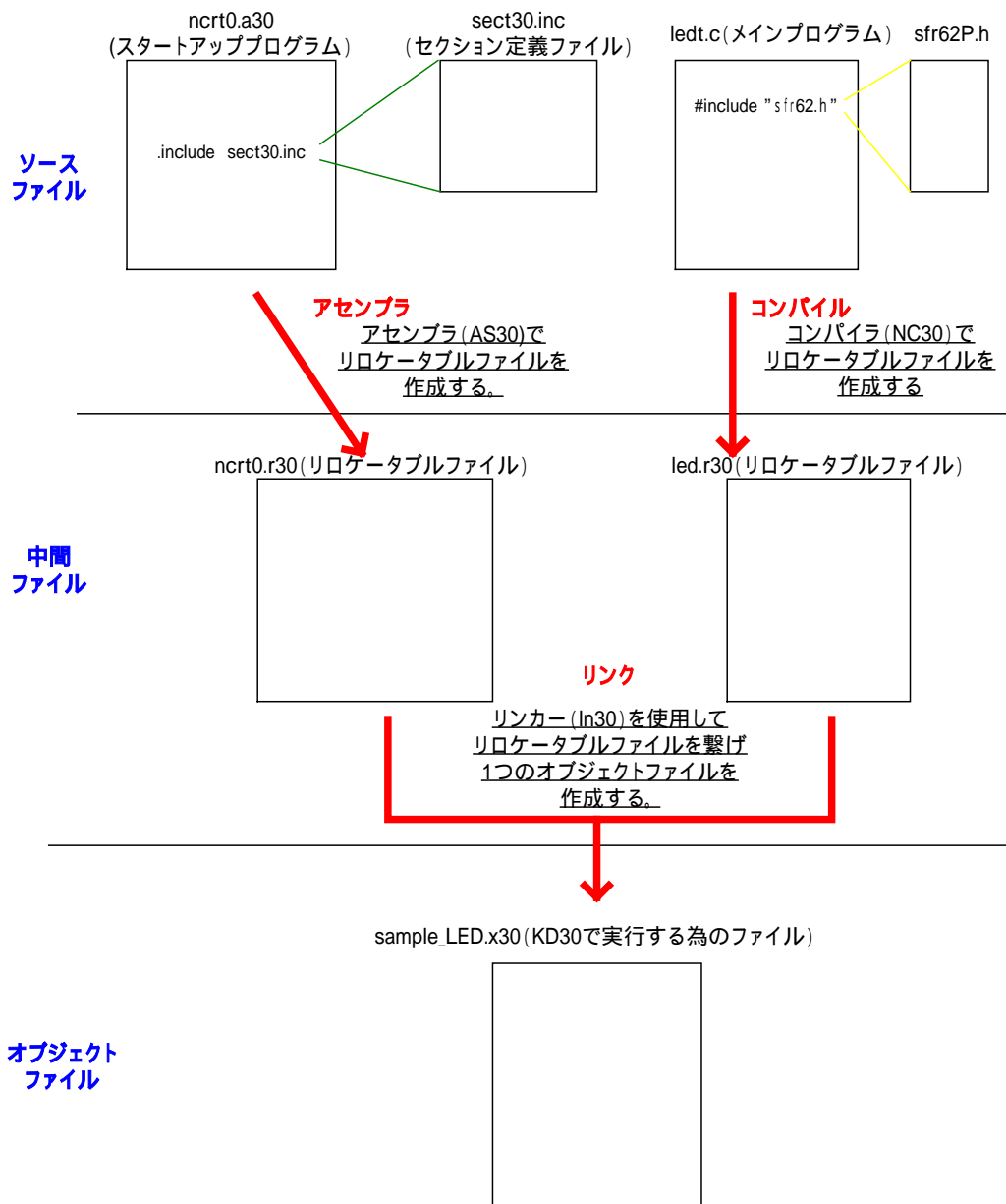
6 . 1 . sample プログラムの内容

CDROM 内の Sample_LED プログラムを使用してプログラム開発の流れを考えましょう。

Sample_LED は CPU ボード上の LED1 を点滅させるプログラムです。

OAKS16-62P でプログラムを開発する為にはスタートアッププログラム(CPU の初期設定をするためのアセンブリ言語で書かれたプログラム)と実際にターゲットを動かすための main プログラムが必要です。

ここではルネサス版スタートアッププログラムを OAKS16-62P 用に変更したものと、led.c というメインプログラムを使用します。この 2 つのプログラムから「アセンブラ」「コンパイラ」「リンク」という作業を行うことによって、最終的にボード上で動作させるオブジェクトファイルを作成します。



6.2.DOS プロンプト上でコンパイルする。

OAKS16-62P ではTMを使用してプログラム開発をすることを前提としておりますが、プログラムの生成手順を理解していただく為に、MSDOS プロンプト上でコマンド入力してオブジェクトファイルを生成する方法を説明します。

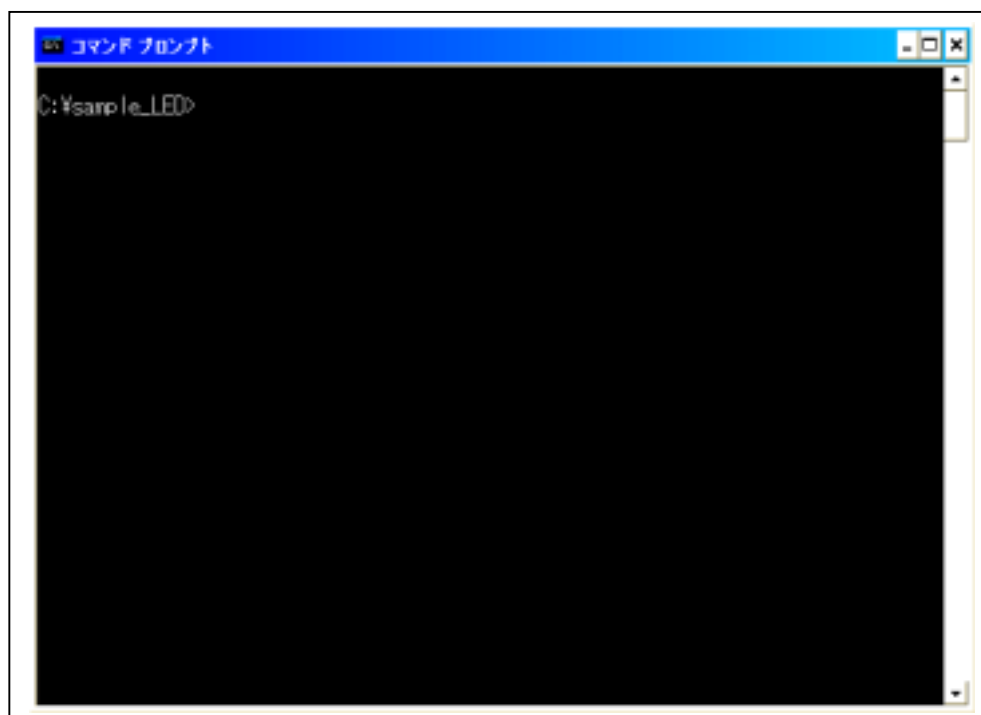
ここではコンパイラ、アセンブラ、リンカを DOS プロンプト上で起動します。

6.2.1.準備

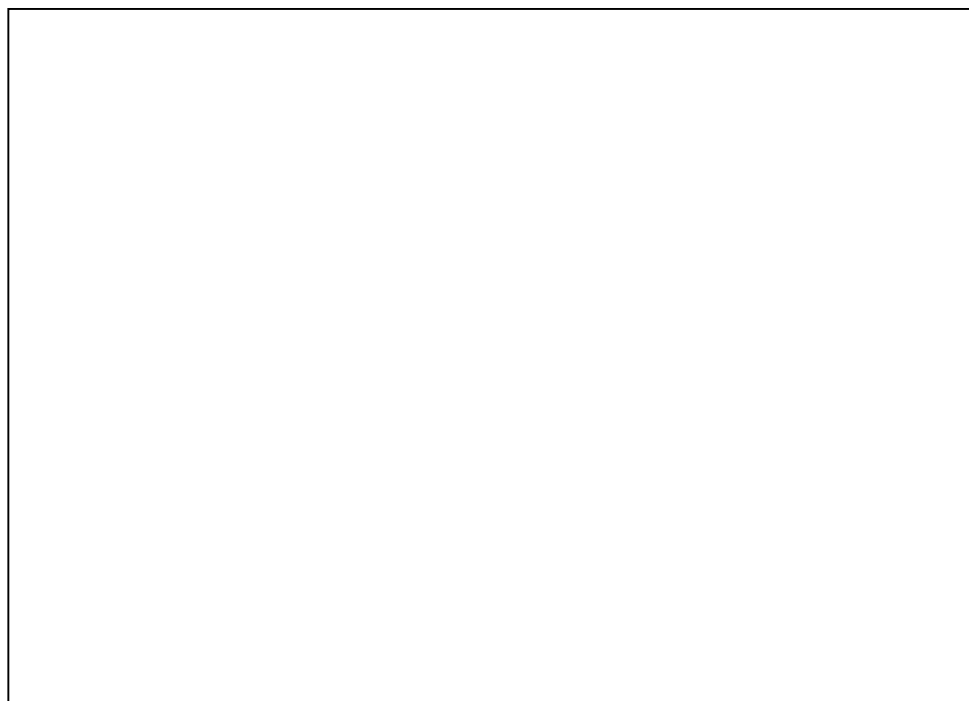
CDROM 内\OAKS16\sample\sample_LED をハードディスクのルートにコピーしてください。フォルダの中には ncr0.a30、sect30.inc、sfr62P.h、led.c の 4 つのファイルが入っています。

([6.1.sample プログラムの内容](#) 参照)

まず、スタート すべてのプログラム アクセサリ コマンドプロンプトを選択しコマンドプロンプト画面を開き、sample_LED のフォルダに移動します。

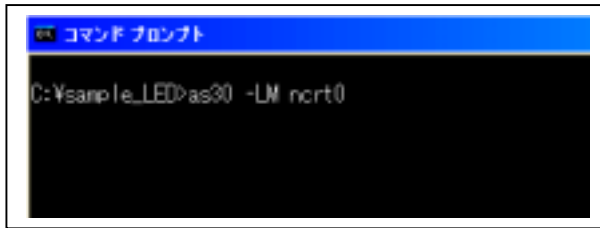


DIR コマンドでフォルダの内容を確認します。



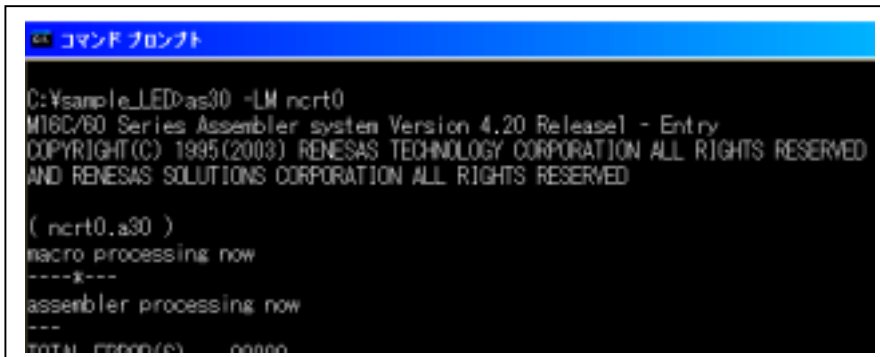
6.2.2. AS30 の起動

ncrt0.a30 をアセンブルして str0.r30 (リロケータブルファイル) を作成します。
as30 -LM ncrt0 と入力しリターンキーを押します。



```
コマンド プロンプト
C:\sample_LED>as30 -LM ncrt0
```

次の画面が出てアセンブルが実行されます。



```
コマンド プロンプト
C:\sample_LED>as30 -LM ncrt0
M16C/60 Series Assembler system Version 4.20 Release1 - Entry
COPYRIGHT(C) 1995(2003) RENESAS TECHNOLOGY CORPORATION ALL RIGHTS RESERVED
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED

( ncrt0.a30 )
macro processing now
----*----
assembler processing now
----
TOTAL FRAMES(S)      00000
```

DIR コマンドで ncrt0.r30 が作成されたことを確認します。



```
コマンド プロンプト
C:\sample_LED>dir
ドライブ C のボリューム ラベルがありません。
ボリューム シリアル番号は 94E8-1E56 です

C:\sample_LED のディレクトリ

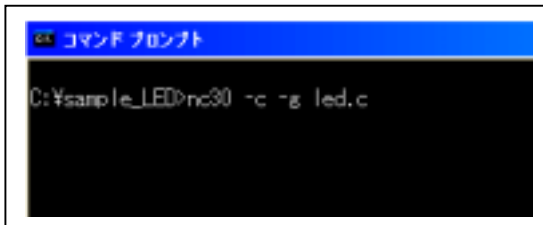
2004/08/23  21:20    <DIR>          .
2004/08/23  21:20    <DIR>          ..
2004/08/22  14:04             854 led.c
2004/08/05  07:20             7,862 ncrt0.a30
2004/08/23  21:20            24,176 NCRTO.LST
2004/08/23  21:20             5,458 NCRTO.R30
2004/08/22  14:05            10,912 sect30.inc
2003/08/06  22:51            122,360 sf+r62.h

        6 個のファイル             171,624 バイト
        2 個のディレクトリ  54,478,450,688 バイトの空き領域

C:\sample_LED>
```

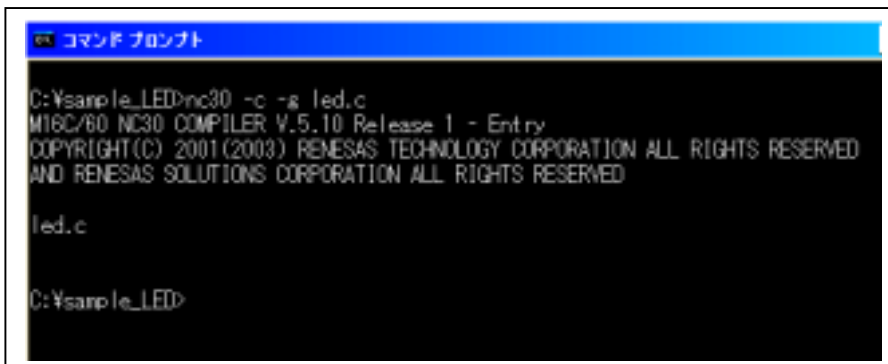
6.2.3. NC30 の起動

led.c をコンパイルして test.r30(リロケータブルファイル)を作成します。
nc30 -c -g led.c と入力しリターンキーを押します。



```
コマンド プロンプト
C:\sample_LED>nc30 -c -g led.c
```

次の画面が出てコンパイルが実行されます。

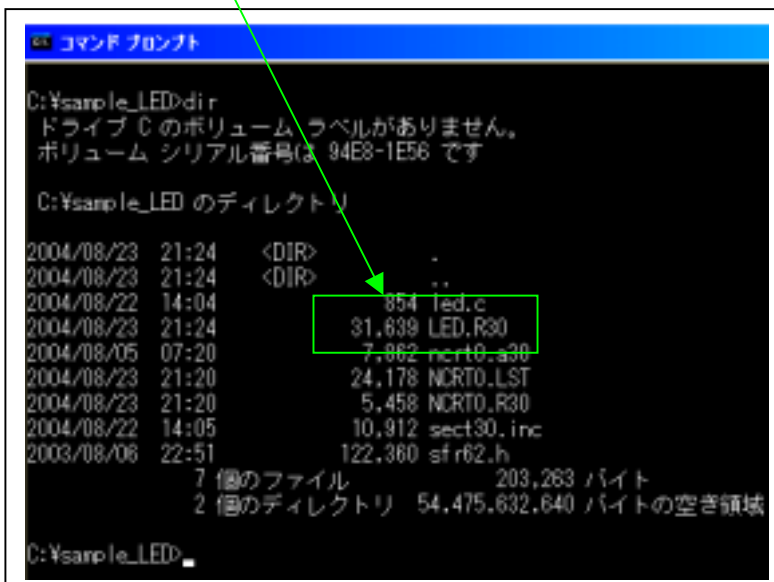


```
コマンド プロンプト
C:\sample_LED>nc30 -c -g led.c
M16C/60 NC30 COMPILER V.5.10 Release 1 - Entry
COPYRIGHT(C) 2001(2003) RENESAS TECHNOLOGY CORPORATION ALL RIGHTS RESERVED
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED

led.c

C:\sample_LED>
```

DIR コマンドで led.r30 が作成されたことを確認します。



```
コマンド プロンプト
C:\sample_LED>dir
ドライブ C のボリューム ラベルがありません。
ボリューム シリアル番号は 94E8-1E56 です

C:\sample_LED のディレクトリ

2004/08/23  21:24    <DIR>          .
2004/08/23  21:24    <DIR>          ..
2004/08/22  14:04             854 led.c
2004/08/23  21:24          31,839 LED.R30
2004/08/05  07:20           7,862 ncrto.a30
2004/08/23  21:20          24,178 NCRTO.LST
2004/08/23  21:20           5,458 NCRTO.R30
2004/08/22  14:05          10,912 sect30.inc
2003/08/06  22:51         122,360 sfr62.h
              7 個のファイル             203,263 バイト
              2 個のディレクトリ  54,475,832,640 バイトの空き領域

C:\sample_LED>
```

6.2.4. LN30の起動

In30 -G -MS -L nc30lib -o sample_LED ncrto led と入力し ncrto.r30、led.r30 をリンクして sample_LED.x30 を作成する。(ncrto がライブラリを使用しているので「-L nc30lib」が入りません。)

```
コマンドプロンプト
C:\sample_LED>In30 -G -MS -L nc30lib -o sample_LED ncrto led
Linkage Editor (In30) for M16C Family Version 4.10.00
COPYRIGHT(C) 1995(2003) RENESAS TECHNOLOGY CORPORATION ALL RIGHTS RESERVED
AND RENESAS SOLUTIONS CORPORATION ALL RIGHTS RESERVED

now processing pass 1
processing "ncrto.r30"
processing "led.r30"
processing "Libraries"
now processing pass 2
processing "ncrto.r30"
processing "led.r30"
processing "C:\MTOOLS\LIB30\nc30lib.lib (BZERO.r30)"
processing "C:\MTOOLS\LIB30\nc30lib.lib (BCOPY.r30)"

DATA      0001024(00400H) Byte(s)
ROMDATA   0000282(00124H) Byte(s)
CODE      0000400(00190H) Byte(s)

C:\sample_LED>
```

DIR コマンドで sample_LED.x30 が作成されるのを確認します。

```
コマンドプロンプト
C:\sample_LED>dir
ドライブ C のボリューム ラベルがありません。
ボリューム シリアル番号は 94E8-1E56 です

C:\sample_LED のディレクトリ

2004/08/23  21:39  <DIR>          .
2004/08/23  21:39  <DIR>          ..
2004/08/22  14:04                854 led.c
2004/08/23  21:24            31,639 LED_R30
2004/08/05  07:20             7,862 ncrto.a30
2004/08/23  21:20            24,178 NCRTO_LST
2004/08/23  21:20             5,458 NCRTO_R30
2004/08/23  21:39             4,597 sample_LED.nsp
2004/08/23  21:39            36,571 sample_LED.x30
2004/08/22  14:05            10,912 sect30.inc
2003/08/06  22:51           122,360 sfr62.h
          9 個のファイル                244,431 バイト
          2 個のディレクトリ  54,472,642,560 バイトの空き領域

C:\sample_LED>
```

以上でオブジェクトファイルが作成されました。KD30 を起動すれば実行することが出来ます。

6.3. TMでプロジェクトを作成しビルドする。

次に TM (統合開発環境) を使用してプロジェクトを作成し、オブジェクトファイルを生成するまでの手順を説明します。TM では各ファイルを登録すると、依存関係を基に `make` を作成し、ビルドコマンドの実行によって `make` が実行されオブジェクトファイルが作成されます。そのため、DOS プロンプト上でコマンドラインを入力するより簡単にオブジェクトファイルが生成できずし、`make` を使用しているため、開発時間が短縮できます。

6.3.1. `make` について

ここで、TM の説明をする前に `make` について簡単に説明します。

6.3.1.1. `make` とは

TM は、プログラムファイル作成の為にコマンド起動を、「GNUmake」を使って行っています。TM がどのようにプログラム開発をサポートしているのかを理解する為に、`make` の事を少し知っていただきたいと思います。

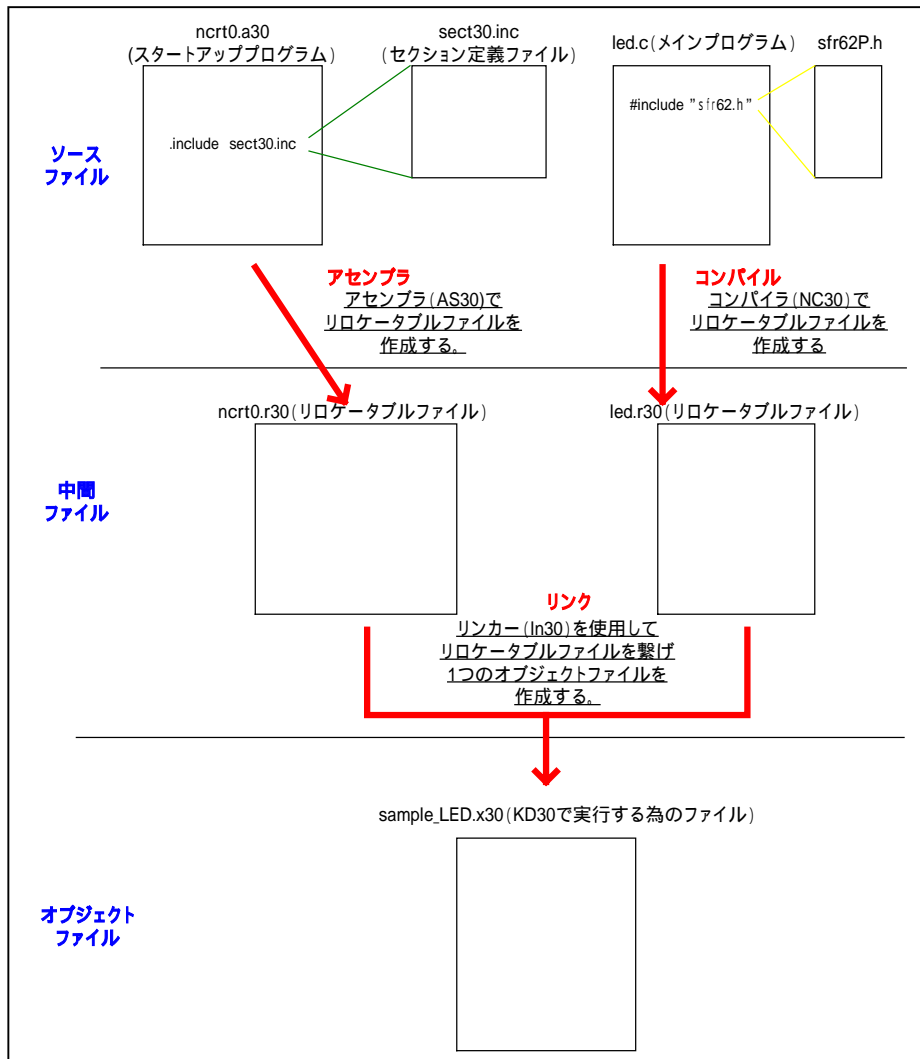
`make` は、AT&T ベル研究所で UNIX 用に開発されたプログラム保守ツールです。C 言語は UNIX オペレーティングシステムの開発のための言語ですから、C 言語プログラム開発では、多くの場合 `make` が使われます。

`make` は、コンパイラに付属されている場合もありますが、今回 TM で使っている `make` は、GNU プロジェクトで作られたものです。GNU は「GNU's Not Unix」(GNU は Unix ではない) の再起頭文字であり「グニュー」と発音されます。「GNUmake」はこの GNU プロジェクトで開発されたフリーソフトウェアで、UNIX の世界では、大変ポピュラーなものです。この `make` の簡単な使い方を、sample プログラムを使って説明していきましょう。

TM をインストールすると `make` は使えますので、特に `make.exe` だけをインストールする必要はありません。

6.3.1.2.make の考え方

sample_LED のプログラムを考えてみましょう。



ソースファイルは led.c(C 言語ソースファイル)、ncrt0.a30 (アセンブラ言語ソースファイル) の 2 つです。

ncrt0.a30 からはアセンブラ AS30 により、start0.r30 というリロケータブルなファイルが生成されます。

led.c からはコンパイラ NC30 により、led.r30 というリロケータブルなファイルが生成されます。

で作成されたりロケータブルモジュールファイルをリンクして、sample_LED.x30 というアブソリュートモジュールファイルを作成します。(デバグ KD30 用) ここまでが、開発の流れです。

ここで、全ての作業が終了してから、led.c のファイルだけに変更が発生したとします。ここで ncrt0.a30 変更していないのでそのままにし、led.c のコンパイルだけを実行した方が目的のファイルを生成するための時間が短くて済みます。これを実現するのが make です。

make では makefile にこの のような依存関係と実行コマンドを記述します。そして、それぞれのファイルの作成された日時に注目し、ソースファイルのほうに変更があった場合だけ、そのファイルに関連するコマンドを実行し、新しいオブジェクトファイルを作成します。変更されていないソースファイルに関しては、コンパイル、アセンブル等を行わず、以前に作成されたりロケータブルファイルをそのまま使います。この作業を行う実行ファイルが、make.exe です。

6.3.1.3.make の記述方法

次に簡単に make の記述について説明します。

make の記述方法

```
ターゲット：ソース 1 ソース 2 . . . ソース n  
            コマンド
```

ここでコマンド行の先頭には一つ以上のタブが必要です。

この記述の意味するところは、

「ターゲットは、ソース 1 からソース n までのファイルによって生成される。その作成手順はコマンドが示している。」

ということです。

make はこれを見て、まずターゲットという名前のファイルが存在するかどうかをチェックします。もし、存在しなければ、コマンドを実行します。また、ソースファイルのどれかがターゲットより新しければコマンドを実行します。そうでなければ、何もしません。ソースが、どこか他の行でターゲットに指定されている場合は、その行を先に実行します。

この make を記述したファイルを makefile と呼びます。ファイル名は拡張子無しの “makefile” です。

この記述にもとづいて sample_LED の makefile を作成すると次のようになります。

```
Sample_LED.x30 : ncrt0.r30 led.r30  
                ln30 -G -MS -L nc30lib -o sample_LED ncrt0 led  
start0.r30 : start0.a30  
            as30 -LM start0.a30  
test.r30 : led.c  
          nc30 -c -g led.c
```

以上が meke の考え方です。しかし、make の記述はマクロなどを使用できいろいろな記述方法があるので難しいところがあります。Make 記述をすることなく簡単に make を使えるようにしたものがこれから説明する TM (統合開発環境) というわけです。

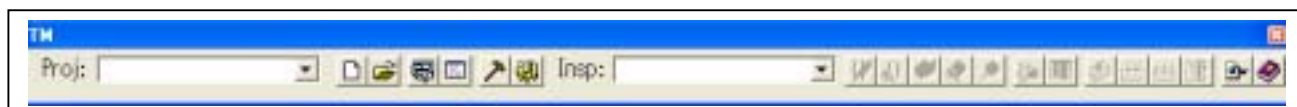
6.3.2. 準備

ハードディスクのルートに CDROM 内\OAKS62P\sample\sample_LED_TM をコピーしてください。

フォルダの中には ncrt0.a30、sect30.inc、sfr62.h、led.c の 4 つのファイルが入っています。
([6.1.sample プログラムの内容](#) 参照)

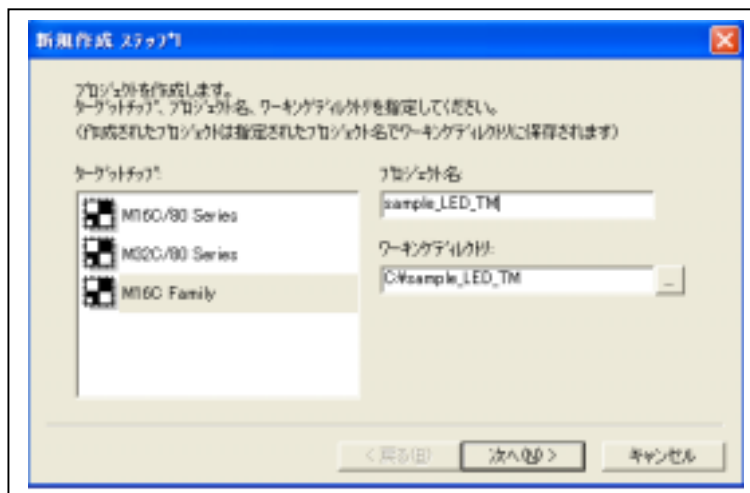
6.3.3. TMを起動する

「スタート」 「すべてのプログラム」 「RENESAS-TOOL」 「TMV3.20 A」 「TM」
をクリックして TM を起動します。

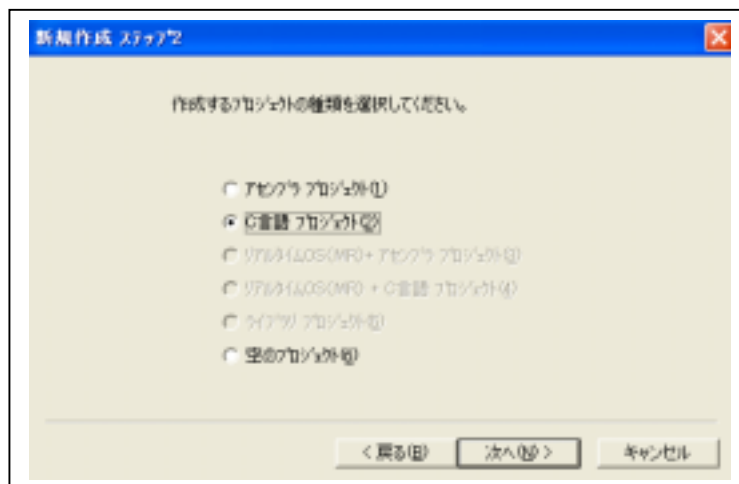


6.3.4. プロジェクトを作成する

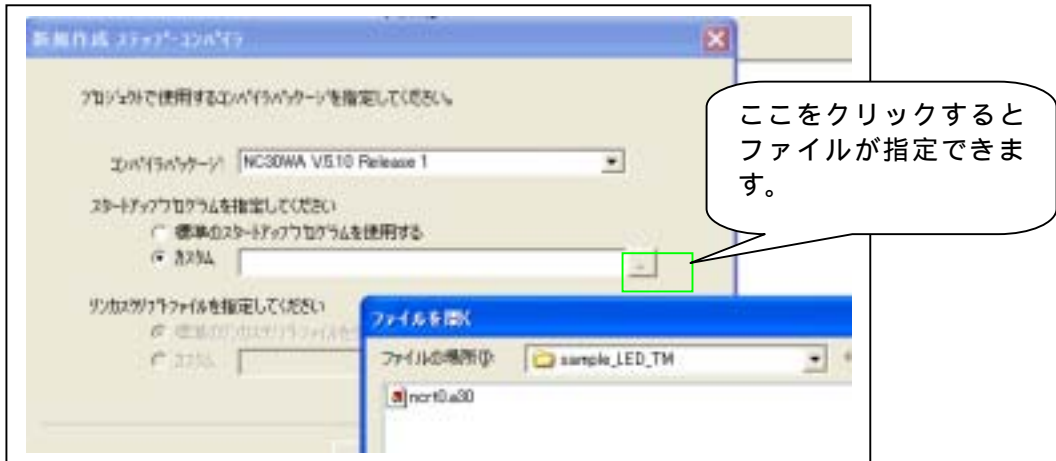
プロジェクトバーの新規プロジェクト作成ボタンをクリックします。



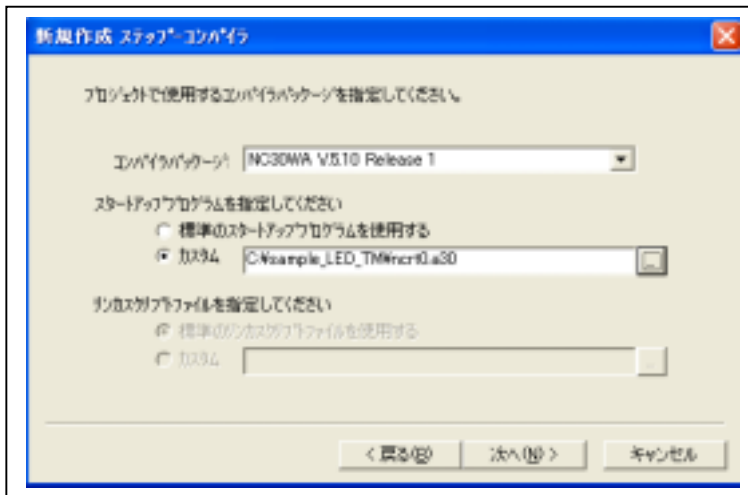
ワーキングディレクトリを選択(ソースファイルが入っている¥Sample_LED_TMを選択します)し、プロジェクト名を入力します(ここでは sample_LED_TM とします)。ターゲットチップを「M16C Family」とし「次へ」をクリックします。



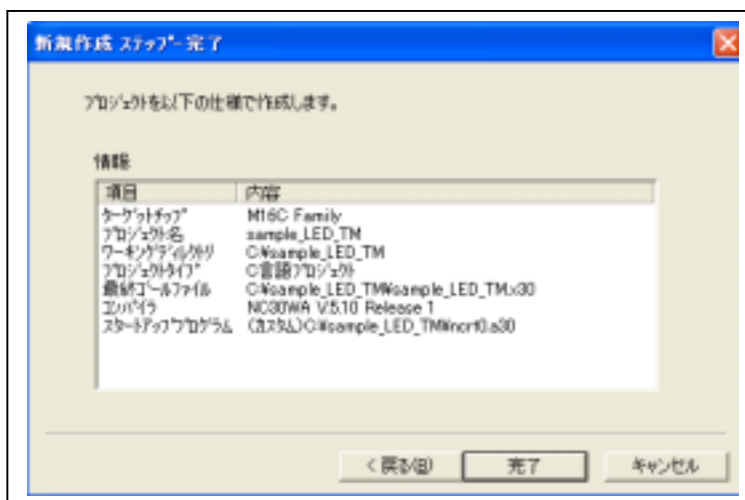
次へをクリックする。



スタートアッププログラムの指定で「カスタム」をチェックし、sample_LED_TM 内の ncr10.a30 を選択します。



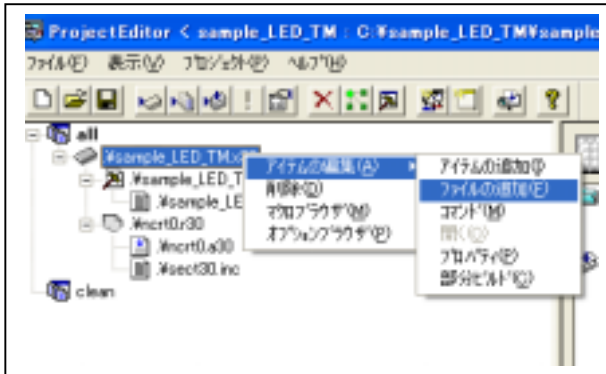
「次へ」をクリックします。



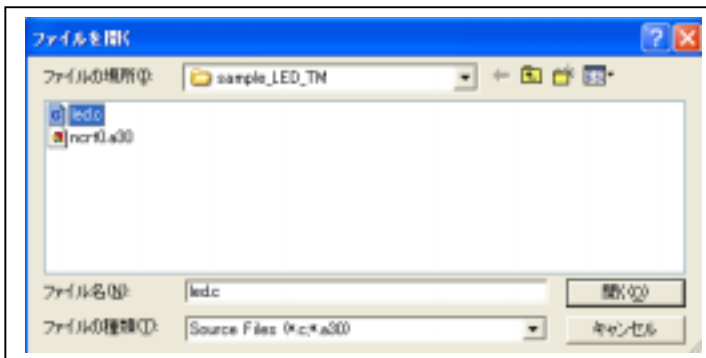
内容を確認し、「完了」をクリックする。これで、プロジェクトの基本が出来ました。

6.3.5. ファイルの登録

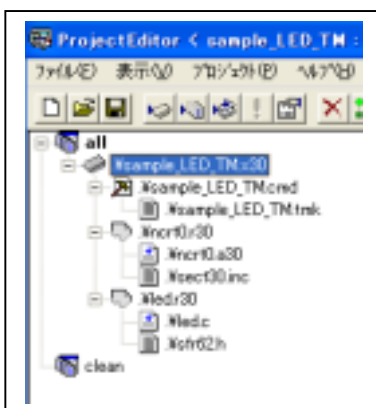
ここで、必要なファイルを登録していきます。新規プロジェクト作成の段階で指定されるのはスタートアッププログラムだけですので main プログラムやほかのモジュールわけして作成したいいくつかのプログラムをここで登録します。今回はスタートアップのほかに必要なファイルは test.c だけです。インクルードファイルはコンパイル、及びアセンブル時にインクルードされますのでここではファイルとして登録しません。



Sample_LED_TM.x30 (最終オブジェクトファイルを選択し、右クリックから「アイテムの編集」「ファイルの追加」を選択します。



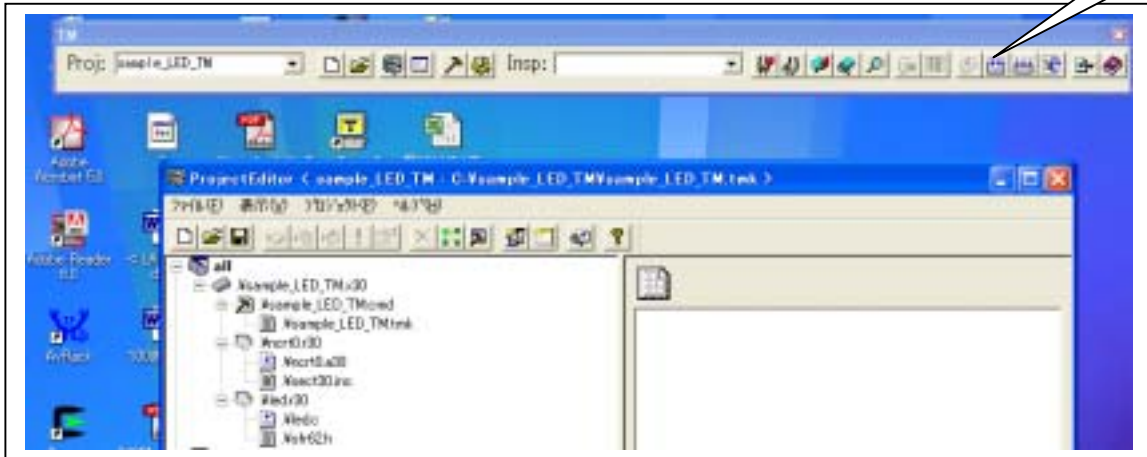
ファイルの選択画面が出ますので、test.c を選択し、「開く」をクリックします。



プロジェクトに「test.c」が追加されました。

ここでプロジェクトバーの「ビルド」ボタンをクリックすると、ビルドが実行され、コンパイラ、アセンブラ、リンカが起動しオブジェクトファイルが作成されます。

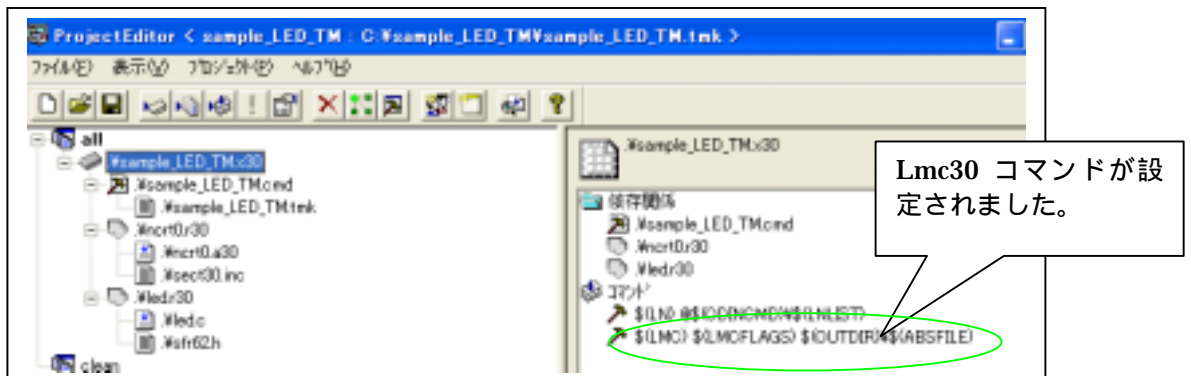
ビルド
ボタン



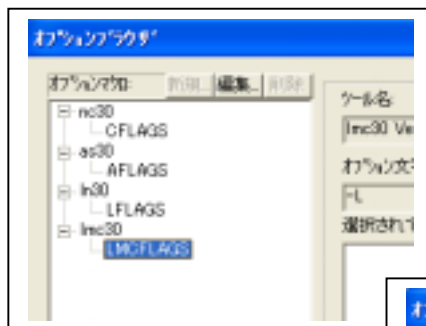
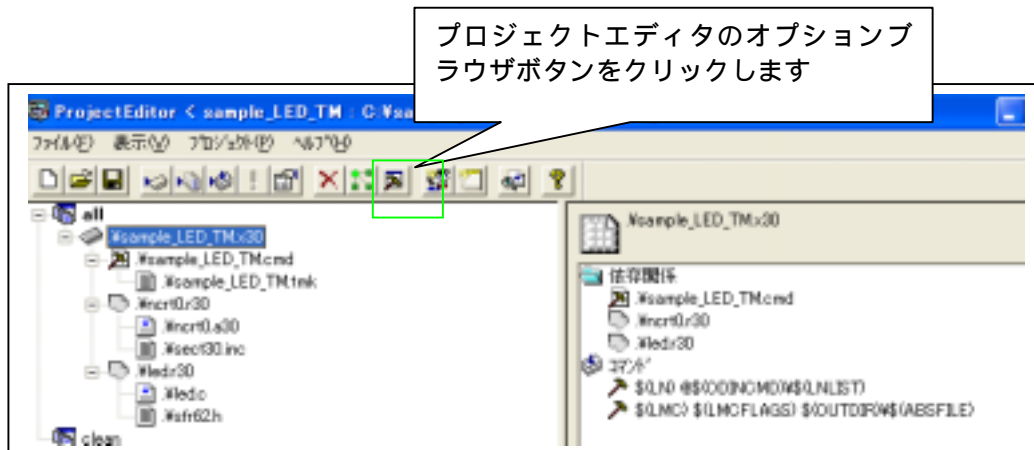
以上が、TMを使用してオブジェクトファイルを作成するまでの手順です。

6.3.6. モトローラSフォーマットファイルの作成

ここまで説明してきた sampleA.x30 は KD30 で動かすことのできるファイルです。KD30 でデバッグが終了すると M30626FHPFP 内のフラッシュメモリにプログラムを書き込んで単体で（パソコンと接続しないで）動かします。その時にフラッシュメモリに書き込めるファイルはモトローラ S フォーマットのファイルになります。拡張子.x30 のファイルから拡張子.mot（モトローラ S フォーマット形式）のファイルに変換する為には lmc30 というプログラムを使用します。TMで lmc30 を使用する手順をここで説明します。

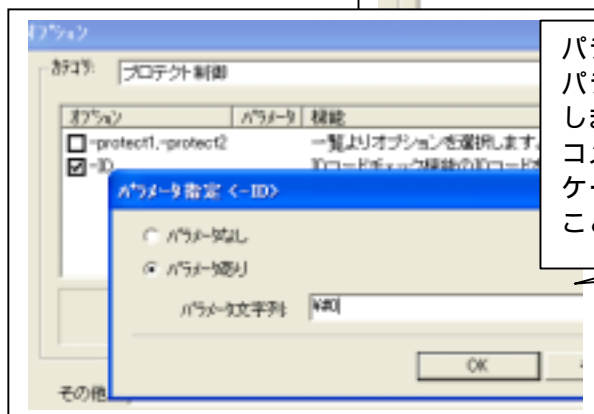
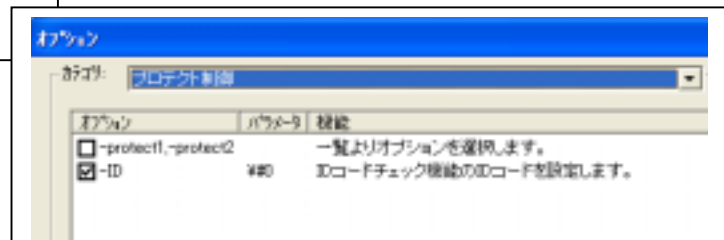


ここで、IDの設定をします。IDについては後で説明しますが、Imc30のオプションでIDを設定できますのでその方法を説明します。IDは0にしてください。



カテゴリで「プロテクト制御」を選択し「ID」にチェックします。

「LMCFLAGS」を選択し「編集」をクリックします。



パラメータ指定ウィンドが開くのでパラメータありを選択し「¥#0」を入力します。（“#”即値マークはmakeではコメントを示すマークなので前に“¥(エスケープ)”を入れ、コメントマークではないことを示します。

以上で、設定が終わりです。ビルドすると、sample_LED_TM.x30のファイルとともにsample_LED_TM.motのファイルとsample_LED_TM.IDというIDファイルが生成されます。

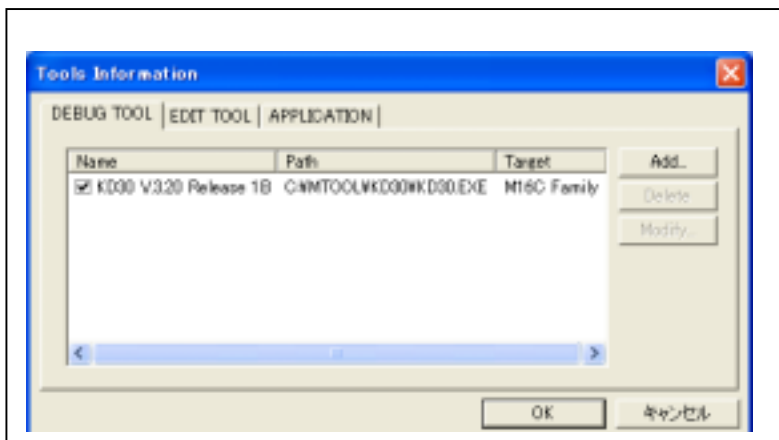
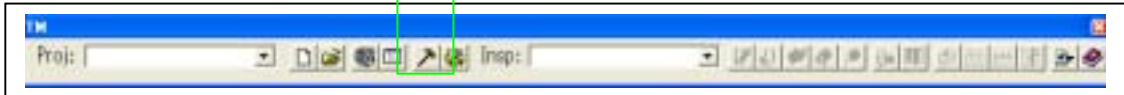
6.4. KD30での実行

出来上がったオブジェクトファイルを KD30 で動作確認します。

6.4.1. KD30 の起動

TM のプロジェクトバーに登録すれば TM から起動できます。

ツールの登録ボタンをクリックします



KD30 を選択し OK をクリックします。

プロジェクトが開いていればデバッグボタンをクリックすれば KD30 が起動します。

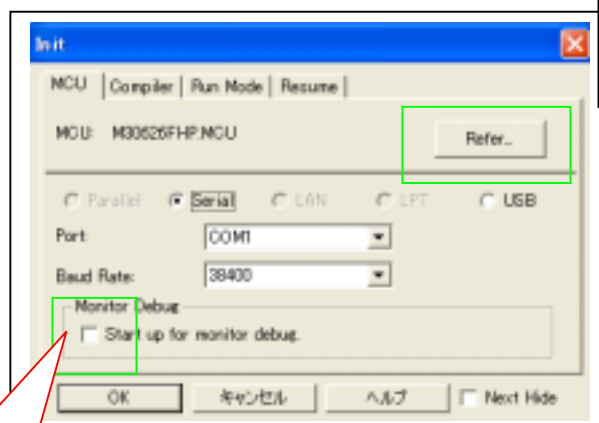


TM からでなく、KD30 を単独で起動することも出来ます。

Windows の

スタート 全てのプログラム RENRESAS-TOOL KD30 V3.20 Release1B KD30
を選択すると KD30 が起動します。

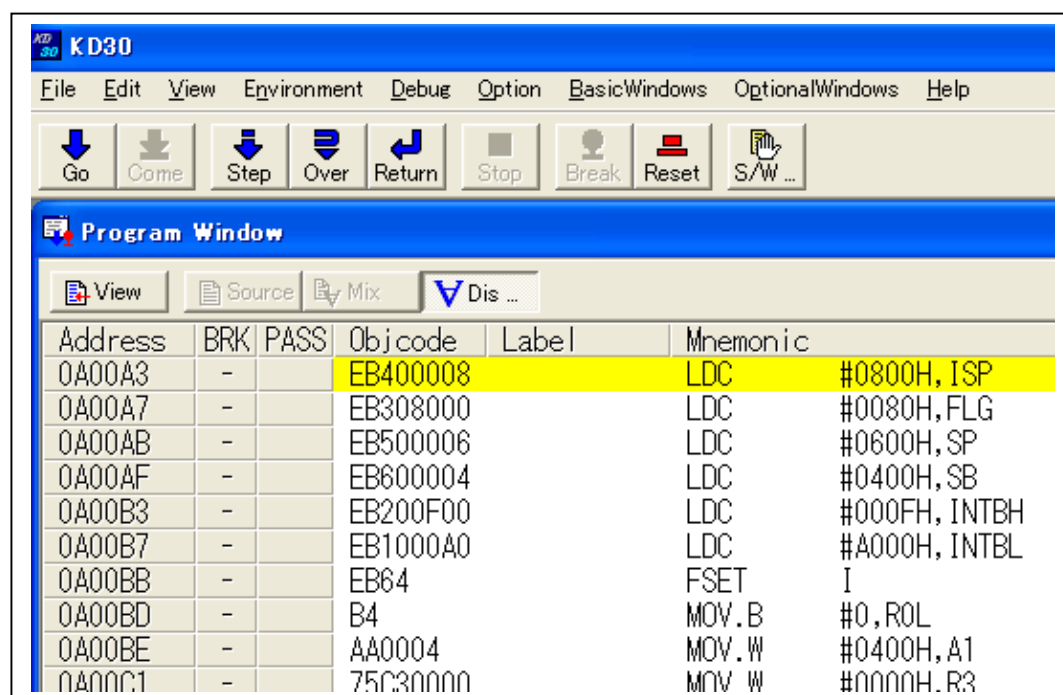
KD30 の初期画面です。MPU を選択します。



ここをクリックすると
MPU が選択できます。
“ M30626FHP.MPU ”
を選択します

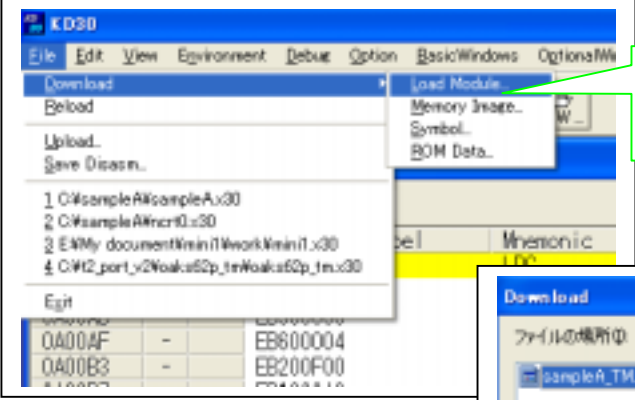
チェックしない
ください！

Port とボーレートの選択を確認し、OAKS16 ボードをリセットし、「OK」をクリックします。パソコン側の KD30 プログラムと OAKS16 上のモニタプログラムが通信を行い、接続できれば次の画面が表示されます。

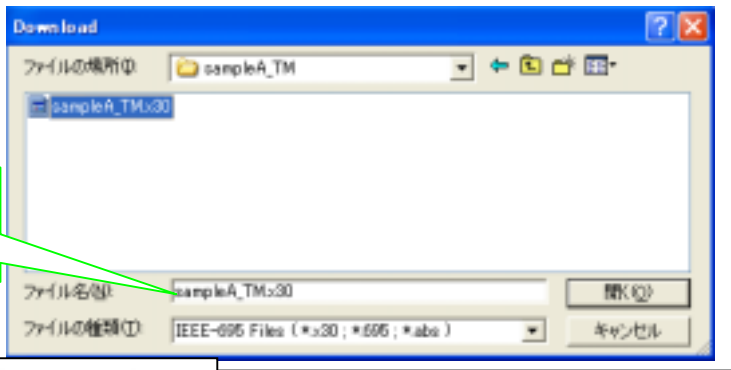


6.4.2. プログラムのダウンロード

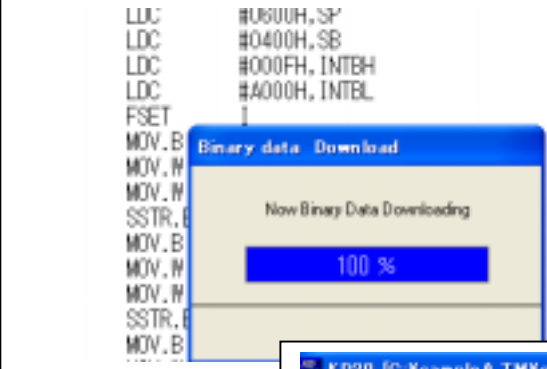
作成したプログラムを OAKS16 のメモリにダウンロードします。



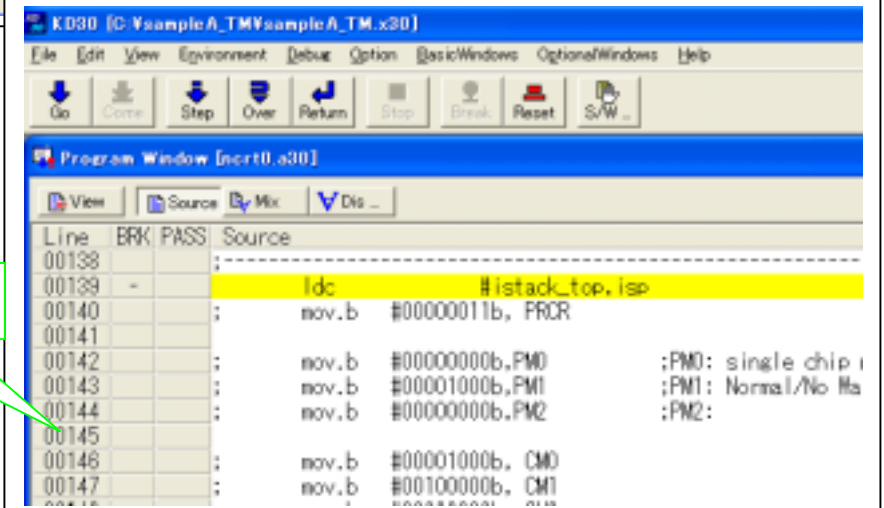
「File」「Download」「Load Module」を選択します。



SampleA_TM.x30 を選択し「開く」をクリックします。



プログラムがダウンロードされています。

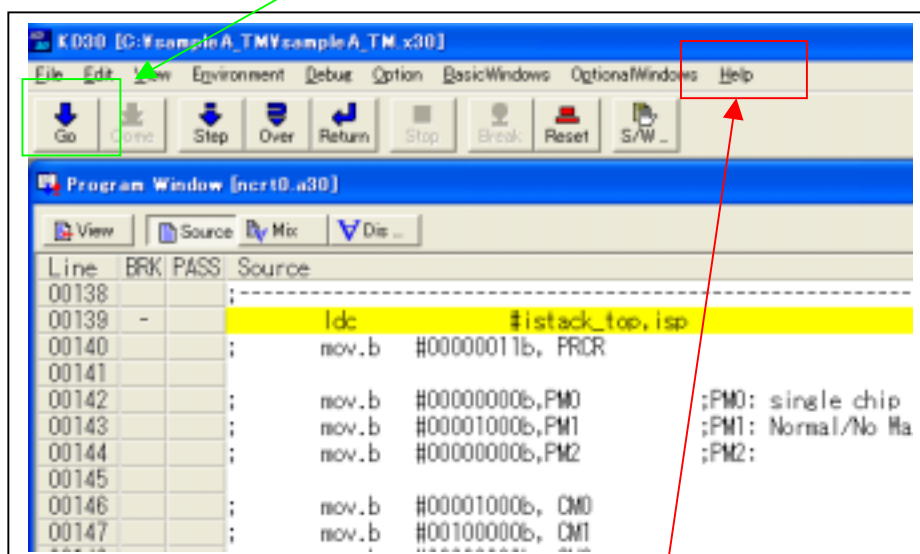


ダウンロード終了です

Line	BRK	PASS	Source
00138			
00139	-		ldc #istack_top, ispc
00140			mov.b #00000011b, PRCR
00141			
00142			mov.b #00000000b, PM0 ;PM0: single chip i
00143			mov.b #00001000b, PM1 ;PM1: Normal/No Ma
00144			mov.b #00000000b, PM2 ;PM2:
00145			
00146			mov.b #00001000b, CM0
00147			mov.b #00100000b, CM1

6.4.3. プログラムの実行

プログラムがダウンロードされたら「GO」ボタンをクリックするとプログラムが実行されます。



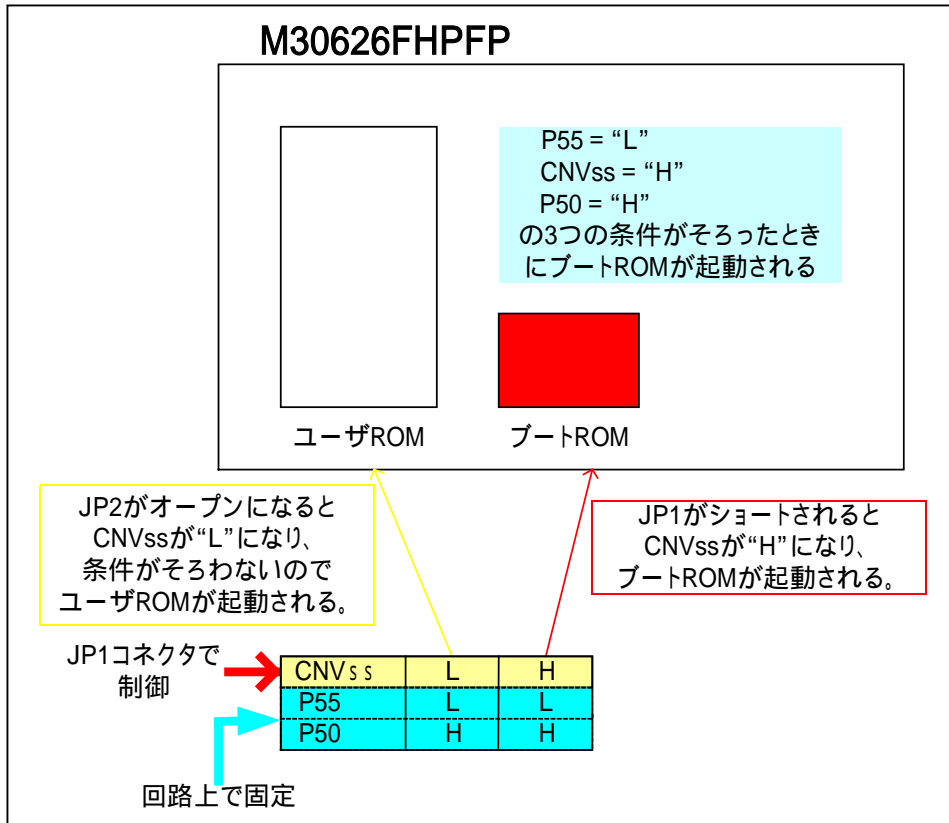
プログラムの実行を停止する場合には「STOP」ボタンをクリックします。

そのほか、ブレークポイントの設定、シングルステップ、メモリの参照等KD30の詳しい使い方はHELPを開いて確認してください。

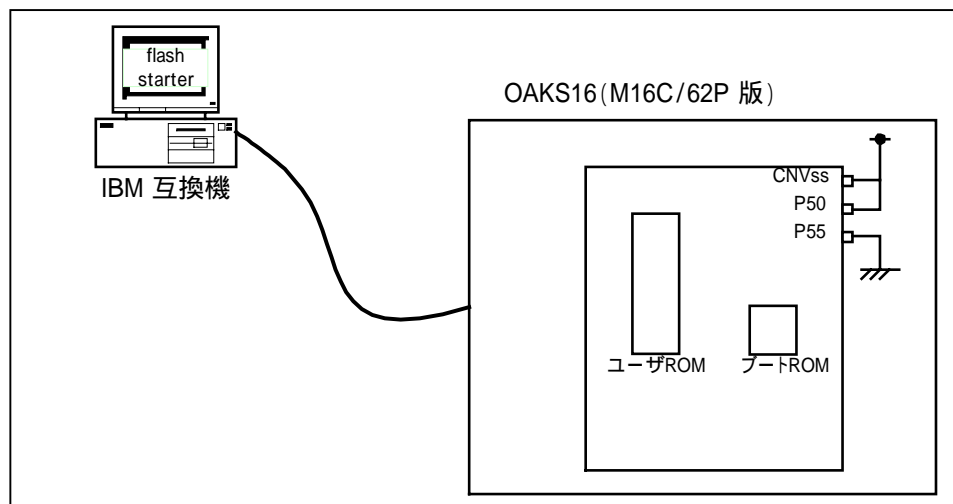
6.5. フラッシュ ROM への書き込み

KD30 でプログラムの動作が確認できたら flashROM へプログラムを書き込んで単体で動作させます。

M30626FHPFP に内蔵されているフラッシュ ROM は P55 端子に “L”、CNVss 端子に “H”、P50 端子に “H” を入力してハードウェアリセットすると、ブートモードになり、ブート ROM 領域のプログラムを実行します。



CPU 出荷時にはブートROMには標準シリアル入出力モードの書き換え制御プログラムが格納されています。OAKS16 で使用する、flashstart は、このブートROM に格納されているプログラムと通信しながらユーザROM にプログラムを書き込んでいくためのソフトです。



6.5.1. 準備

OAKS16の電源をOFFにします。
CPUボード上のJP1をショートします。
OAKS16の電源をONにしてください。

これで、M16C/62P内のブートROM内に書かれている、flashROM書き込み用のプログラムが起動しました。

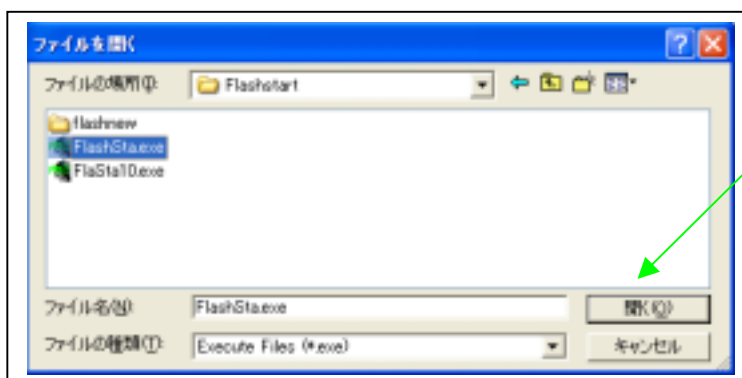
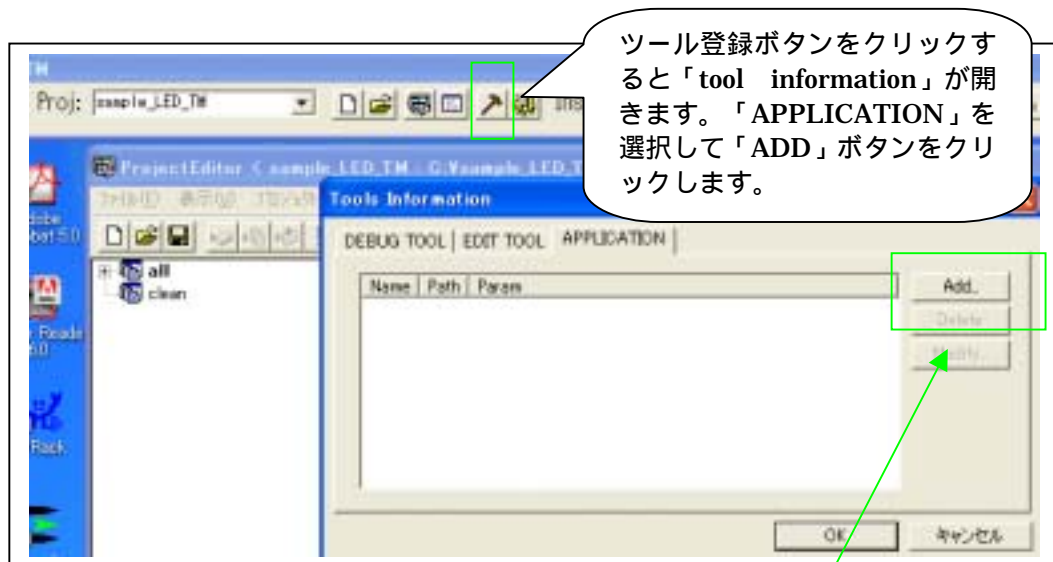
6.5.2. Flashstater.exeの起動

6.5.2.1. 直接起動する方法

インストールしたディレクトリに移動し、flashsta.exeをダブルクリックすれば、起動します。

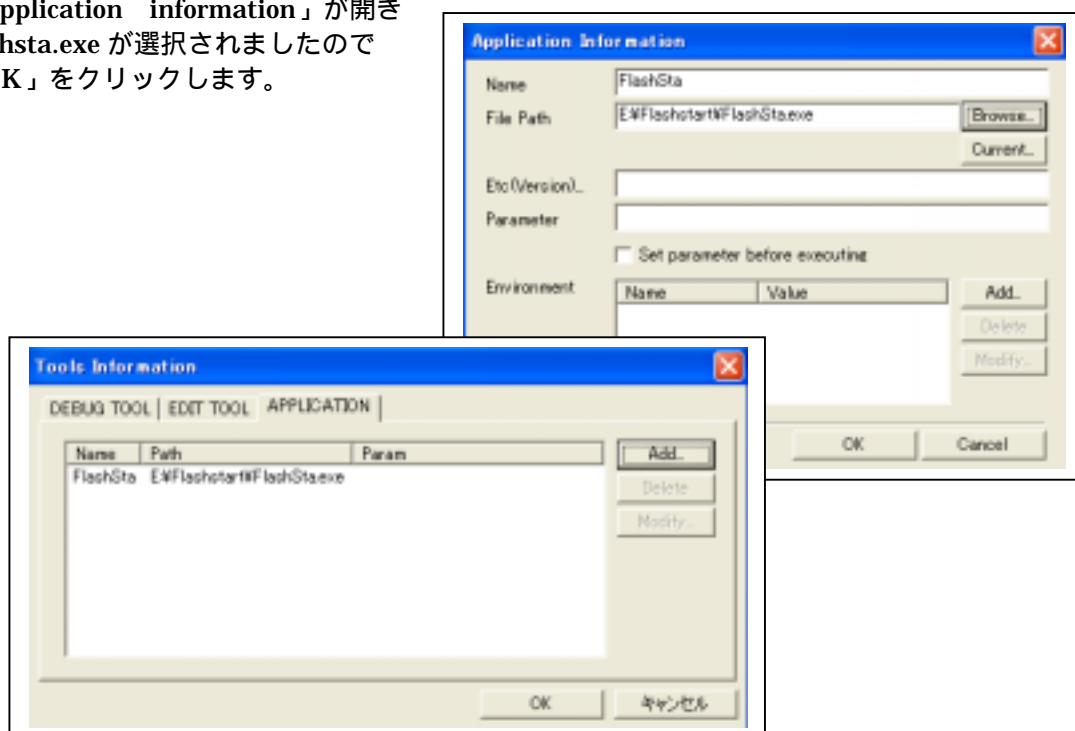
6.5.2.2. TMに登録して起動する方法

TMのツールバーに登録し、そこから起動する方法を説明します。

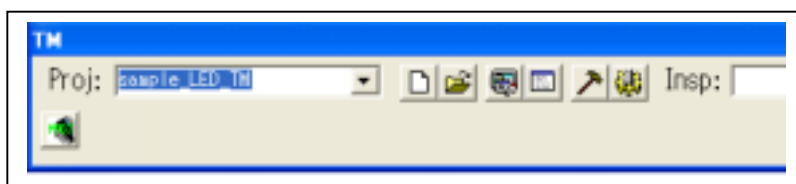


「Add」ボタンをクリックするとファイルが選択できます。
[flashsta.exe]を選択し「開く」をクリックします。

「Application information」が開き
flashsta.exe が選択されましたので
「OK」をクリックします。



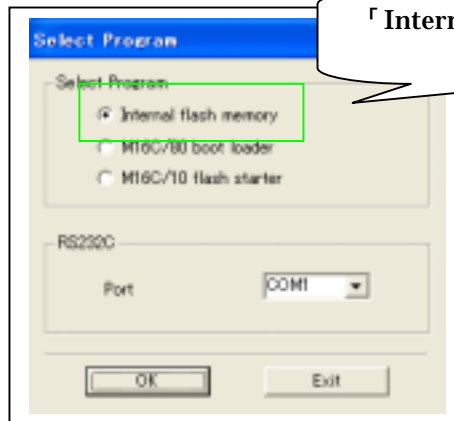
「Tool information」の画面を確認し、「OK」をクリックします。



ツールバーに flashsta のボタンが登録されました。これをクリックすれば、TM から flashsta.exe
が起動できます。

6.5.3.書き込み

flashstart が立ち上がると次の画面が表示されます。RS232C がご自身のパソコン環境で使用している port であることを確認し、「OK」をクリックします。

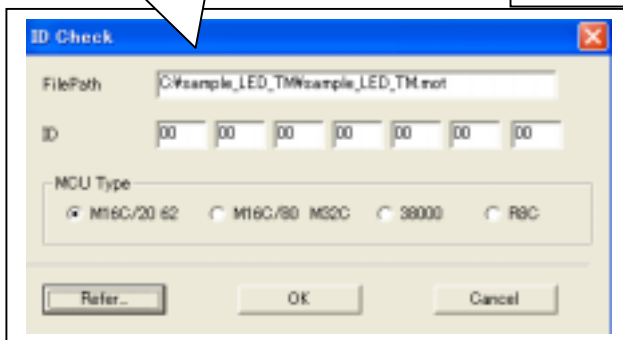
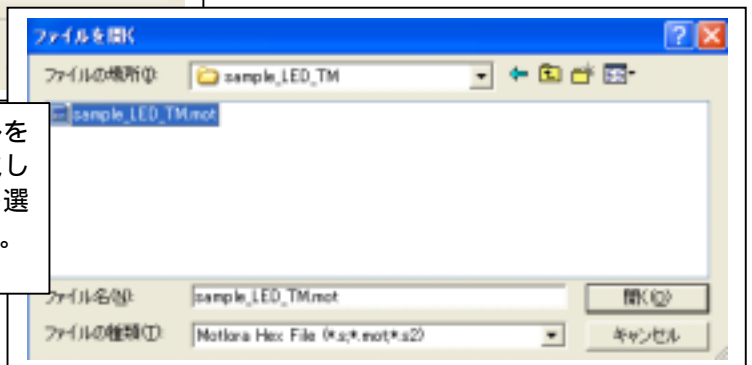


「Internal flash memory」を選択



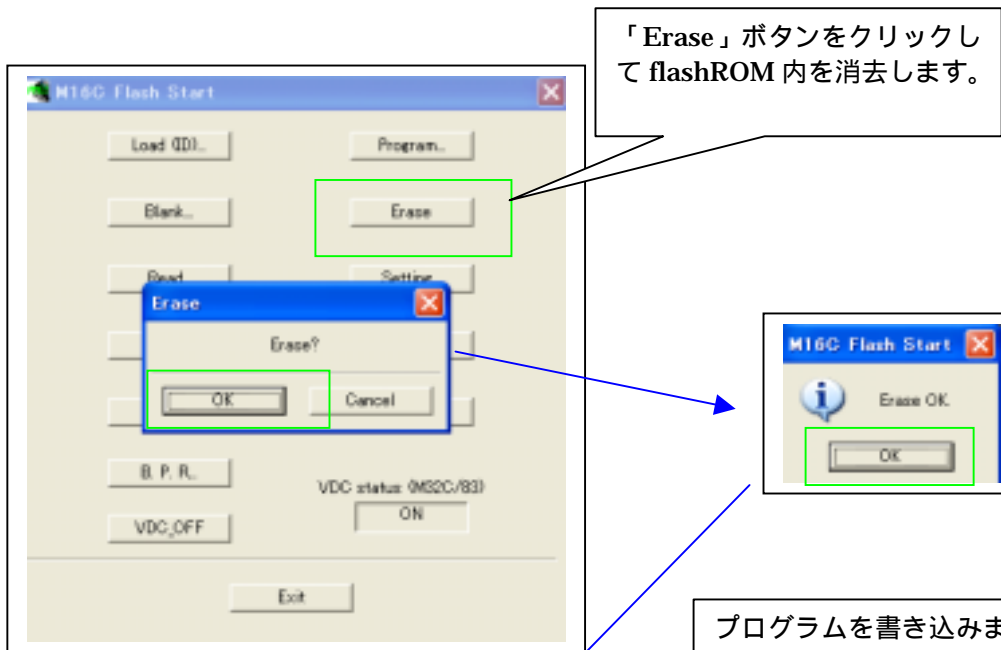
ID チェック画面が表示されます。「Refer」をクリックするとファイルが開けます。

TMでモトローラフォーマットのファイルを作成するときに、ID ファイルも一緒に作成しているので、「sample_LRD_TM.mot」を選択すると自動的に ID に“00”が入ります。

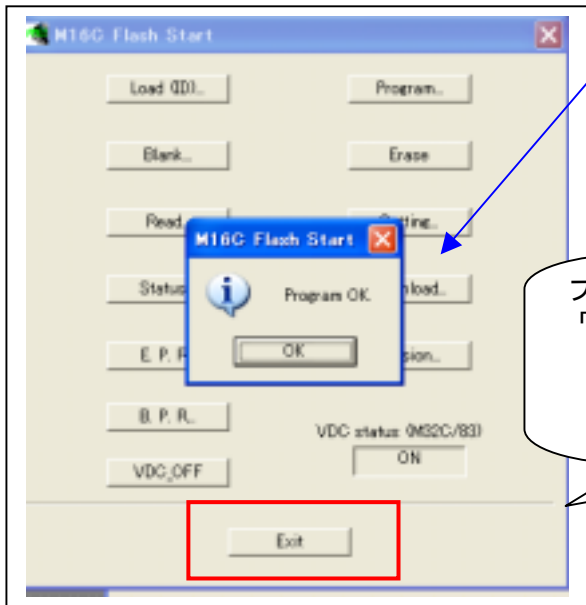
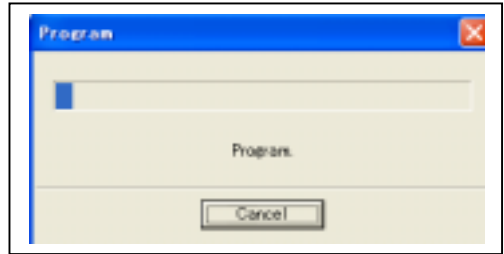
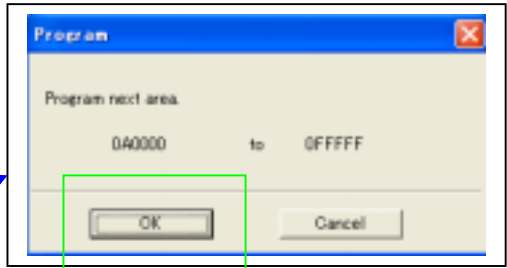
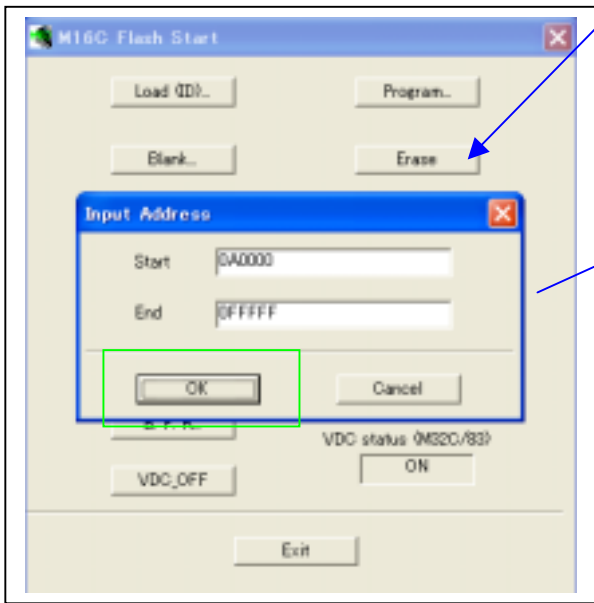


ID チェックを「OK」にすると flashstart の画面が表示されます。





プログラムを書き込みます。



プログラムの書き込み完了です。
「OK」をクリックして「Exit」します。

6.5.4. プログラムの実行

以下の手順で書き込んだプログラムが実行されます。

- OAKS16 の電源を OFF にする。
- JP1 コネクタの Pin をはずしオープンにする。
- OAKS16 の電源を ON にする。

6.5.5. IDコードについて

OAKS16-62P に使用している M30626FHFPF のフラッシュメモリには「内蔵フラッシュメモリ書き換え禁止機能」が付いています。

(CDROM /DOC/DATA_S 内、ハードウェアマニュアル内 p256 参照)

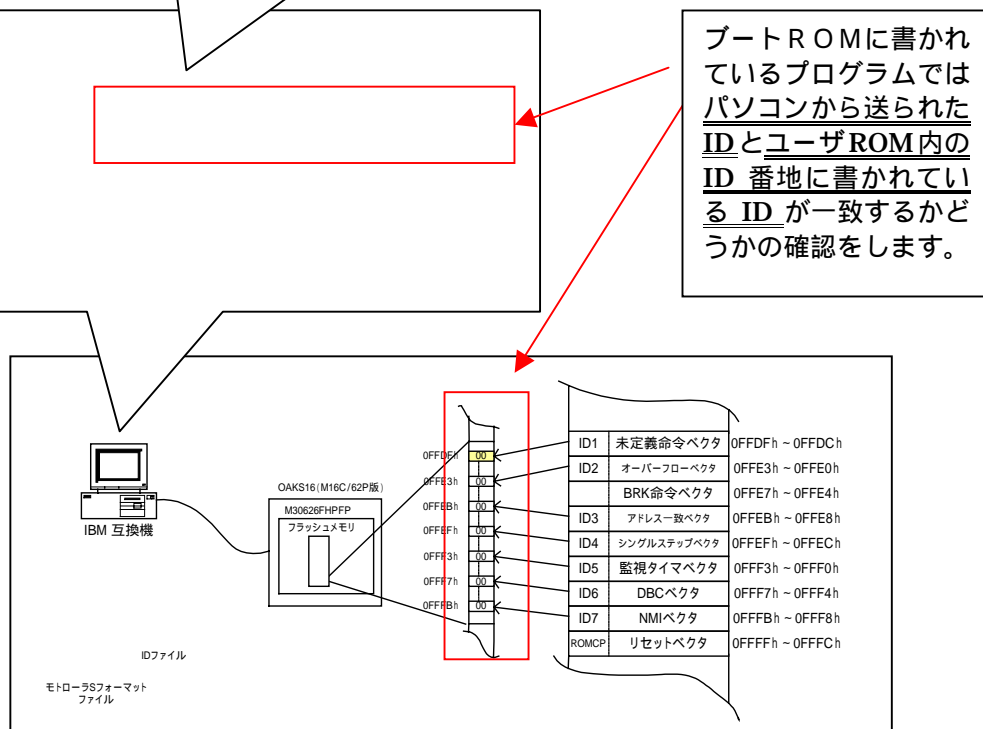
OAKS16 ではフラッシュメモリの標準入出力モードを使用してプログラムを書き換えます。このモードで使用する場合、フラッシュメモリがブランクではない場合（中にプログラムが記述されている場合）、ライターから（OAKS16 の場合パソコン上で起動している flashstart から）送られてくる ID コードとフラッシュメモリに書かれている ID コードが一致するか判定します。コードが一致しなければライターから送られてくるコマンドは受け付けられず、書き換えは出来ません。フラッシュメモリの ID コードは各 8 ビットのデータで、領域は、1 バイト目から 0FFFDh、0FFFE3h、0FFFEb、0FFFEFh、0FFFF3h、0FFFF7h、0FFFFBh 番地です。

0FFDFh ~ 0FFDC	ID1	未定義命令ベクタ
0FFFE3h ~ 0FFE0h	ID2	オーバーフローベクタ
0FFE7h ~ 0FFE4h		BRK命令ベクタ
0FFEBh ~ 0FFE8h	ID3	アドレス一致ベクタ
0FFEFh ~ 0FFEC	ID4	シングルステップベクタ
0FFF3h ~ 0FFF0h	ID5	監視タイマベクタ
0FFF7h ~ 0FFF4h	ID6	DBCベクタ
0FFFBh ~ 0FFF8h	ID7	NMIベクタ
0FFFh ~ 0FFFC	ROMCP	リセットベクタ

Flashstart の画面：

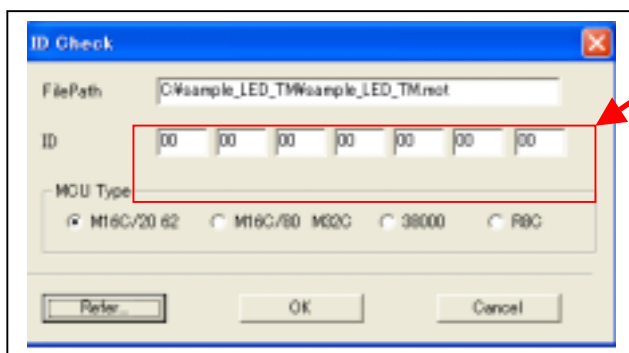
ここで入力された ID コードがボードに送られ、参照されます。

ブートROMに書かれているプログラムではパソコンから送られた ID とユーザROM内の ID 番地に書かれている ID が一致するかどうかの確認をします。



<パソコン側 : flashstart >

ID チェック画面では、書き込む対象となるファイル名を選択する必要があります。
ここで、選択したファイルを作成するときに lmc30 で ID オプションをつけて ID ファイルを作成していれば（同一フォルダ内に ID ファイルが存在すれば）ID コードに自動的に読みこまれ画面に表示されます。

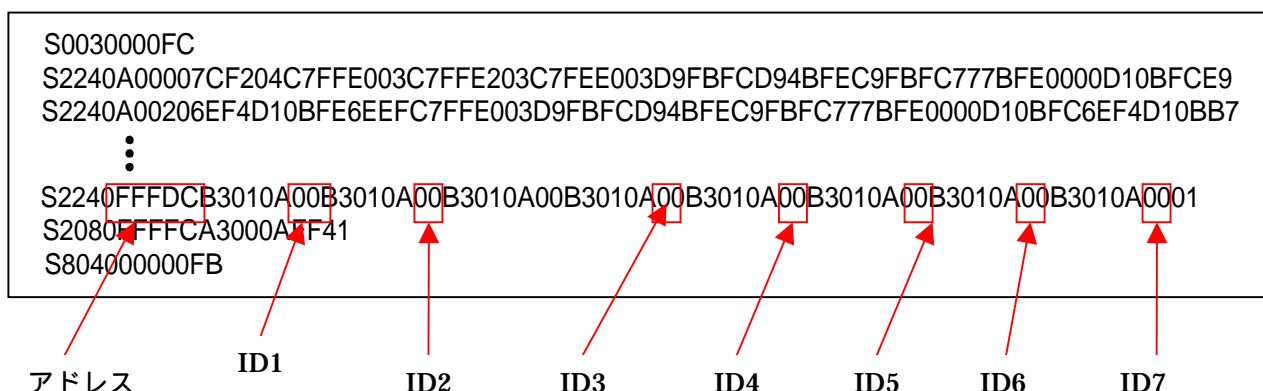


-ID#0	
FFFDf	: 00
FFFE3	: 00
FFFEb	: 00
FFFEf	: 00
FFFF3	: 00
FFFF7	: 00
FFFFb	: 00
FFFFf	: ff

< sample_LED.ID(ID ファイル) >

プログラム開発時に「lmc30 -ID #0 sample_LED_TM」と ID オプションをつけてファイルを作成した場合、ファイルの中に ID が挿入され、別ファイルとして.ID の拡張子がついた ID ファイルが作成されます。

<sample_LED_tm.mot (モトローラ S フォーマットのファイル) >



6 . 5 . 5 . 0AKS16-62P ID コードのについての注意事項

OAKS16-62P で使用している、KD30 のモニタプログラム、及び sample プログラムは全て ID を “00” としています。コンパイル/ビルド時に、LMC の ID オプションで違う値に設定されない限り、ID コードは全て 00h となります。

ID を変更する場合には、一度書き込まれてしまった ID をメモリから読み出す方法はありませんのでご注意ください。

書き込んだファイルが分かればエディタで ID 番地のデータを調べ手入力 ID を入力してください。

flashStart でイレースした後、プログラムを書き込まないで終了してしまった場合、消去後のメモリの内容は “ff” になっています。ID に手入力 “ff” を入れてください。

データ書き込み中に何らかの原因で電源が落ちてしまった場合、書き込んだ値は不定となりますので書き換え不可能になる場合があります。十分にご注意ください。

7. 使用上の注意事項

7.1. コンパイラの制限事項

7.1.1. 使用できないオプション

オークス版は、下記に示す仕様になっておりますのでご注意ください。詳細につきましては、\Tool\Rnoteに格納されているリリースノートを御参照いただけますようお願いいたします。

1) -fansiオプションについて

エントリー版では、-fansiオプションがデフォルトで指定されてコンパイルを行います。本オプションを指定しても、コンパイルに影響を与えることはありません。

-fansi, -fextend_to_int(-fETI) 、 -fnot_reserve_far_and_near(-fNRFAN) 、
-fnot_reserve_inline(-fNRI)が指定された時と同等の動作になります。

2) プログラムの記述について

下記の語を記述する際、語の前に”アンダースコア()”を付加してください。

inline	_inline
near	_near
far	_far
asm()	_asm()

3) NC30とAS30のオプションについて

マニュアルに記載されている下記のオプションは**ご使用になれません**。

C コンパイラ (NC30)	
デバッグ用オプション	-genter, -gno_reg
最適化オプション	-O[1-5], -OR, -OS, -Oconst(-OC), -Ono_bit(-ONB), -Ono_break_source_debug(-ONBSD), -Ono_float_const_fold(-ONFCF), -Ono_stdlib(-ONS), -Osp_adjust(-OSA), -Ostack_frame_align(-OSFA), -Oloop_unroll(-OLU), -Ono_asmopt(-ONA), -Ono_logical_or_combine(-ONLOC), -Ocompare_byte_to_word(-OCBTW)
生成コード変更オプション	-finfo, -fuse_DIV(-fUD), -fansi, -fnear_ROM(-fNRROM), -fsmall_array(-fSA), -fno_align(-fNA)
アセンブル、リンクオプション	-as30, -ln30
アセンブラ (AS30)	
オプション	-finfo, -P, -M

4) ソフトウェアおよびユーティリティについて

下記に示すソフトウェアおよびユーティリティは**ご使用になれません**。

ソフトウェア	STK ビューワ, MAP ビューワ, アセンブル最適化 (aopt30), ライブラリアン (lb30), 構造化記述アセンブラ (pre30), 標準関数ライブラリソースファイル
ユーティリティ	utl30(SBDATA 宣言&SPECIAL ページ宣言ユーティリティ)

5) 統合化開発環境について

統合化開発環境のインスペクタ機能は**ご使用になれません**。

ライブラリプロジェクトは**作成できません**。

7.1.2. スタートアップファイルの注意

コンパイラをインストールすると STARTUP ファイル(ncrt0.a30 と sect30.inc)と一緒にインストールされます。ユーザープログラムをKD30を使用して、OAKS16上で動作させる場合は、このSTARTUP ファイルは使用しないで下さい。

2) ユーザープログラムを使用して、OAKS16上で動作させる場合は、CD-ROM\OAKS16\SAMPLE\STARTUP に格納してある STARTUP ファイル(ncrt0.a30 と sect30.inc)を使用するようにして下さい。

7.2. リモートデバッガの制限事項

7.2.1. ストップモード、ウェイトモードに関する制限事項

ユーザープログラム上でストップモード、ウェイトモードを使用する場合、KD30はフリーランモードで起動するようにして下さい。デバッグを行う場合は、実行する前に予め、RAM ウィンドウ、C ウォッチウィンドウ、ASM ウォッチウィンドウを閉じて下さい。また、ストップモード、ウェイトモードを解除する処理部にブレークポイントを設定するなどして、ブレークポイントで止まるまで、画面の操作をしないようにして下さい。

7.2.2. ユーザープログラムのリアルタイム性について

1) SamplingRun (サンプリング) モード

サンプリングモードでは、Go 実行時および Come 実行時にユーザープログラムの実行状態を定期的に監視します。そのため、ブレークなどによるユーザープログラムの停止を検出することができます。通常のデバッグを行うときに選択してください。

2) FreeRun (フリーラン) モード

フリーランモードでは、Go 実行時および Come 実行時にユーザープログラムの実行状態を監視しません。そのため、ユーザープログラムのリアルタイム性は保証されますが、ブレークなどによるユーザープログラムの停止を検出できません。したがってユーザープログラムが停止しても、KD30はGo 実行および Come 実行動作を停止しません。KD30を停止させるには、STOP ボタンを押してください。ユーザープログラムのリアルタイム実行を行いたいときに選択してください。**注意)**フリーランモードの時、RAM ウィンドウ、C ウォッチウィンドウ、ASM ウォッチウィンドウを開いているとリアルタイム性がそこなわれますので、RAM ウィンドウ、C ウォッチウィンドウ、ASM ウォッチウィンドウを閉じた状態で使用して下さい。

7.2.3. 例外的なステップ実行について

1) ソフトウェア割り込み命令

ソフトウェア割り込みを発生させる命令(未定義命令、オーバフロー命令、BRK 命令、INT 命令の)の命令内部処理を連続して STEP 実行はできません。

<例> INT命令の場合

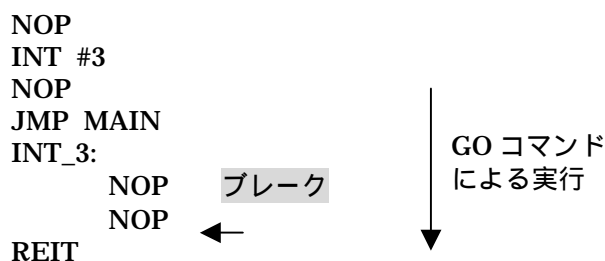
```
NOP
NOP
INT #3
NOP
JMP MAIN
INT_3:
NOP
NOP
NOP
REIT
```

STEP 実行すると
すり抜けてしまう

2) INT 命令

INT 命令を用いたプログラムのデバッグは、INT 命令内部処理にソフトウェアブレイクを設定し、GO コマンドと共に使用して下さい。

< 例 >



7.3. 割り込みに関する制限事項

1) 固定ベクタテーブルに配置している割り込み

固定ベクタテーブル上の割り込みで、アドレス一致、監視タイマ、NMI は使用禁止です。ユーザープログラム上で使用している場合は実行できないようにしてください。上記の固定ベクタテーブルにはモニタプログラムがダミー処理 (REIT 命令等) を格納しています。

表 7-1 割り込みベクタテーブル番地

割り込み要因	CPU の仕様	モニタプログラム内の仕様
未定義	FFFDC ₁₆ ~ FFFDF ₁₆	使用禁止 1
オーバーフロー	FFFE0 ₁₆ ~ FFFE3 ₁₆	使用禁止 1
BRK 命令	FFFE4 ₁₆ ~ FFFE7 ₁₆	使用禁止
アドレス一致	FFFE8 ₁₆ ~ FFE ₁₆	使用禁止
シングルステップ	FFFE ₁₆ ~ FFE ₁₆	使用禁止
監視タイマ	FFFF0 ₁₆ ~ FFFF3 ₁₆	使用禁止
DBC	FFFF4 ₁₆ ~ FFFF7 ₁₆	使用禁止
NMI	FFFF8 ₁₆ ~ FFFFB ₁₆	使用禁止
リセット	FFFFC ₁₆ ~ FFFFF ₁₆	FFFFC ₁₆ ~ FFFFF ₁₆

2) 可変ベクタテーブルに配置している割り込み

モニタプログラムは UART1 を使用しており、ユーザープログラムでは使用できません。これ以外の可変ベクタテーブルの割り込みはユーザープログラムで使用可能です。

7.4. 周辺機能に関する制限事項

1) UART 端子

UART1 の送信および受信割り込みはモニタプログラムとホストコンピュータとの通信に使用します。ユーザープログラムでは UART1 を使用しないで下さい。他の端子等と接続しないで下さい。

- ・ 100 ピン版 : TxD1(31pin)、RxD1(32pin)

2) P85/NMI 端子

NMI 割り込みは使用禁止です。OAKS16 上でプルアップしていますが、ユーザーターゲット上で Low レベルにならないように処置ください。

- ・ 100 ピン版 : 17pin

7.5. レジスタ操作に関する制限事項

表にレジスタ操作に関する制限事項を示します。なお、変更が禁止されているレジスタを変更した場合、モニタプログラムの動作は保証できません。

表 7-2 レジスタ操作に関する制限事項

レジスタ名	初期値	制限事項	ユーザープログラムによる変更
プロセッサモードレジスタ0	00 ₁₆ に初期化されます。 シグナルモード	メモリ拡張モードを使用する場合は、ユーザープログラムで、このレジスタを変更して下さい。モニタプログラムでは絶対に変更しないでください。	
プロセッサモードレジスタ1	08 ₁₆ に初期化されます。 内部予約領域拡張	このレジスタをモニタプログラムにて変更される場合は、ビット3を必ず1に設定してください	
システムクロック制御レジスタ0	08 ₁₆ に初期化されます。		
システムクロック制御レジスタ1	22 ₁₆ に初期化されます。 MCLKが分周比：分周なし	ビット1は変更しないで下さい	
ISP(割り込みスタックポイント)	07C00 ₁₆ に初期化されます。	07C00 ₁₆ 未満の値を設定してください。 07C00 ₁₆ ~07CFF ₁₆ はモニタプログラムが使用しています。	
プロテクトレジスタ		プロテクトレジスタのビット2(ポートP9方向レジスタ、SIO3,4制御レジスタの書き込み許可ビット)を“1”(許可)にした直後にモニタプログラムが動作するような場合、モニタプログラムにより任意の番地への書き込み動作が発生するため、P9方向レジスタ書き込み許可ビットが“0”(禁止)になります。したがって、以下の場合P9方向レジスタへの書き込みはできません。書き込み許可ビットを“1”にする命令へのブレーク書き込み許可ビットを“1”にする命令行への「Go」「Step」「Over」「Return」実行ダンブウィンドウなどからのP9方向レジスタの操作。	
フラグレジスタ		Dフラグ、Iフラグへの書き込みは無視されます。 (Dフラグは必ず“0”に、Iフラグは必ず“1”になります。)	
UART1送受信モードレジスタ	15 ₁₆	変更しないでください。	×
UART1転送速度レジスタ	80 ₁₆		
UART1送受信制御レジスタ0	10 ₁₆		
UART1送受信制御レジスタ1	05 ₁₆		
UART1割り込み制御レジスタ0	07 ₁₆		
UART送受信制御レジスタ2	02 ₁₆	ビット1,3,4,5,6については変更しないでください。	
UART1送信バッファレジスタ		このレジスタにデータを書き込まないでください。	×
UART1受信バッファレジスタ		このレジスタを読み出さないでください。	×
割り込みテーブルレジスタ	0FEA00 ₁₆	変更する場合はUART1送受信に“0FF900h”を設定して下さい	×
PLL制御レジスタ	9A ₁₆	変更しないで下さい	×
周辺クロック選択レジスタ	00 ₁₆	変更しないで下さい	×

ユーザープログラムによる変更可否

：変更可能

×：変更禁止

：変更可能（一部制限あり）

7.6. フラグレジスタに関する制限事項

ユーザプログラム上でフラグレジスタを操作する場合は、デバッグフラグ(Dフラグ)を変更しないように、**FSET 命令**、**FCLR 命令**で行ってください。

7.7. メモリ拡張モード使用時の注意点

1) OAKS16-62P 基板上の設定

OAKS16-62P 基板上では HOLD 端子(100PIN 版 : 41PIN)が flashstart 使用時にブート ROM を立ち上げる為、プルダウンしてあります。そのためメモリ拡張モードを使用すると MPU が HOLD 状態になってしまいます。

メモリ拡張モード使用時には OAKS16-62P 基板上の MCU の HOLD 端子(100PIN 版 : 41PIN)、RDY 端子(100PIN 版 : 39PIN)をプルアップして下さい。

2) プログラムの設定

ユーザプログラム内でプロセッサモードレジスタ 0 のプロセッサモードビットを"01 : メモリ拡張モード"に変更してください。

3) 制限事項

メモリ拡張モード時、外部メモリに対して KD30 から行える操作は「メモリ操作 (メモリの読み出し/書き込み)」のみです。外部メモリへのプログラムのダウンロード (File -- Download -- Load Module を使って外部メモリにプログラムの配置) やブレーク等のプログラム制御は行えませんので、ご注意ください。また、プロセッサモード変更後、動作確認する場合は、DUMP ウィンドウで外部メモリを参照してください。

注意) 各端子の確認は付属している回路図を参照下さい。

8. よくある質問

8.1. KD30 起動時に通信エラーが発生するのですが？

KD30 起動時に通信エラーが発生した場合は、以下の内容を確認してください。

- 1) ケーブルの接続不良がないか、電源が投入されているか。
- 2) KD30 の Init ダイアログに設定したシリアルポートと OAKS16 が接続されているシリアルポートが一致しているか。
- 3) 周波数と通信速度が一致しているか

8.2. デバッグ中に通信エラーが発生したのですが？

デバッグ中に通信エラーが発生する場合は、ユーザープログラムの影響（暴走やモニタ領域へのアクセスなど）により、OAKS16 専用にマイコンに書き込まれていた KD30 と通信を行うモニタプログラムが暴走したと考えられます。この場合、以下の手順にしたがってシステムを初期化してください。

- 1) エラーダイアログの「OK」ボタンを押して KD30 を閉じる。
- 2) OAKS16 上のリセットボタンを押す。
- 3) 再度、KD30 を立ち上げる。
- 4) ユーザープログラムをダウンロードする。

これで、デバッグを再開することができます。

デバッグ中に通信エラーが多く発生する場合は以下の点を確認してください。

- 1) 割り込みプログラムを使用しており、多重割り込みを許可していない場合
-> 割り込みプログラムの処理時間が 260 μ s を超える場合は、割り込みプログラムの先頭で I フラグを '1' にしてください。

8.3. 「ソースファイルが見つかりません」と出たのですが？

「ソースファイルが見つかりません」というエラーが発生する場合は、以下の内容を確認して下さい。

- 1) ソースファイルと実行ファイルが同一のディレクトリ（又はフォルダ）に存在しているか確認して下さい。
- 2) ソースファイルが KD30 以外のエディタで開かれていないか確認して下さい。

8.4. ブレーク中の周辺 I/O の動作はどうなるの？

ブレーク中、割り込みは受け付けませんが、周辺 I/O は動作し続けます。たとえば、タイマ等を動作させた後にブレーク等でユーザープログラムを停止させたとき、タイマはカウントし続けますが、タイマ割り込みは受け付けられません。

9 . 付録

9 . 1 . メモリマップ

9-1 に OAKS16-62P に搭載している、ルネサス 16 ビットシングルチップマイクロコンピュータ M16C/62 フラッシュメモリ内蔵版のメモリマップを示します。ユーザー使用可能領域は(RAM 約 29KB, Flash ROM 約 239.4KB) となります。

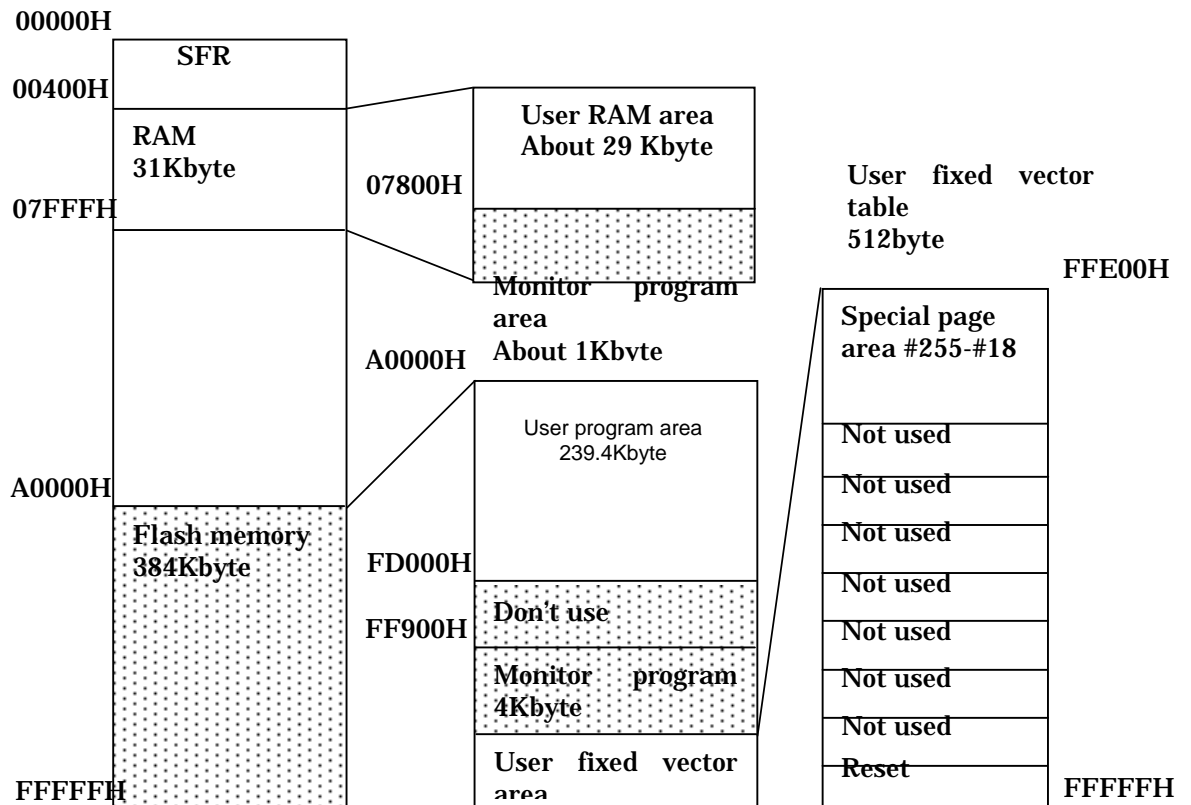


図 9-1 OAKS16 メモリマップ

9.2. 部品表

OAKS16-M30620FCAFP には次のような部品が搭載されています。

表 9-1 OAKS16-M30620FCAFP 部品表

記号	型番	メーカー	数量	備考
IC1	M30626FHPFP	ルネサス	1	FlashROM384KB SRAM31K
IC2	M51957BL	ルネサス	1	リセット IC
IC3	MAX3232	マキシム	1	RS232C ドライバ (相当品)
X1	CSTLS6M00G53-B0	ムラタ	1	負荷容量内蔵セラロック 6MHz (相当品)
C1	SME16V47 μ	日本ケミコン	1	47 μ F 電解コンデンサ (相当品)
C2-10	GRM188F11H104Z	ムラタ	9	0.1 μ F チップコンデンサ (相当品)
R1-6、 R12-17	PK73B1J10K	コーア	12	10K チップ抵抗 (相当品)
R7,8	" 15K	"	2	15K チップ抵抗 (相当品)
R9	" 2.7K	"	1	2.7K チップ抵抗 (相当品)
R10	" 150	"	1	150 チップ抵抗 (相当品)
R11	" 560	"	1	560K チップ抵抗 (相当品)
LED1	EBR3378S	スタンレー	1	小型高輝度型 LED (赤) (相当品)
SW1	SKHHAJ	アルプス	1	リセット用小型スイッチ (相当品)
CN1,3	OX112-DS40G	OX	2	40pin ピンヘッダ (20pin \times 2列) (相当品)
CN2	OX112-DS20G	OX	1	20pin ピンヘッダ (10pin \times 2列) (相当品)
CN4	OX112-SS6G	OX	1	6pin ピンヘッダ (5pin \times 1列) (相当品)
CN5	OX112-SS2G	OX	1	2pin ピンヘッダ (2pin \times 1列) (相当品)
JP1	OX112-SS2G	OX	1	
JP2	OX112-SS3G	OX	1	
JP3.4	OX112-SS6G	OX	1	
キャップ	DIC-252	本多通信工業	4	
CN6	HIF3FC14PA-2.54DSA	ヒロセ	1	オプション

注意 1)

OAKS16-ChipBoard には IC1 と IC3 のみが搭載されています。他の部品に関しては別途ご用意下さい。

10. 改定履歴

日付	ページ	改定内容	旧	新
H18/7		E8 対応		

OAKS16-62P-FullKit ユーザーズマニュアル Rev 2.0

2004年9月初版発行

2006年7月第二版発行

編集 オークス電子株式会社

発行 オークス電子株式会社

禁無断転載

本説明書の一部又は全部を、当社に断りなく、いかなる形でも転載又は複製することを堅くお断りします。