

DT - 300 開発ガイド

Rev 1.02

カシオ計算機株式会社

目次

1. はじめに	1
1-1 環境	1
1-2 開発の流れ	1
2. SHCコンパイラ開発環境	4
2-1 必要な開発環境	4
2-2 コンパイラのインストール	4
2-2-1 ハードディスクへのコピー	4
2-2-2 PATHの設定	6
2-2-3 環境変数の設定	6
2-2-4 CONFIG.SYSファイルの設定	7
2-3 コンパイル	9
2-3-1 漢字コード変換 (KJ_CNVRT.EXE)	9
2-3-2 コンパイルの実行	10
2-3-3 コンパイルの実行例	10
2-4 リンク	10
2-4-1 リンク環境の設定	10
2-4-2 リンクの実行	11
2-4-3 B, Rセクション先頭アドレス補正ツール (SUBMK.EXE)	11
2-4-4 再リンクの実行 (LNK.EXE)	11
2-4-5 リンクの実行例	11
2-5 オブジェクト変換	12
2-5-1 オブジェクト変換 (OVS.EXE)	12
2-5-2 ダウンロードファイルへの変換 (APCOMY.COM)	12
2-6 MAKE	13
2-7 アプリケーション容量の確認	14
3. プログラムダウンロード	16
3-1 動作環境	16
3-1-1 HIOWINIによるダウンロード	16
3-1-2 LMWINIによるダウンロード	19
3-2 ダウンロードファイル	20
3-2-1 システム環境ファイル (CONFIG.HTS)	20
3-2-2 自動起動ファイル (ASTART.HTS)	21
3-3 ダウンロード	22
3-3-1 HIOWINIによるダウンロード	22
3-3-2 LMWINIによるダウンロード	26
4. アプリケーション開発上の留意事項	34
5. Q&A集	35

1. はじめに

本冊子は、DT - 300 のアプリケーションプログラム開発を初めて行う方を対象に、その概要を説明するものです。

プログラムの開発、ダウンロード等の操作方法を、一連の処理の流れに沿って説明します。

なお、各処理の詳細については、各解説書を随時参照して下さい。

1 - 1 環境

DT - 300 のソフト開発にあたっては、以下のものが必要になります。

プログラム作成 / ダウンロード用 PC

PC は、PC / AT 互換機です。

開発キット

ソースコードのコンパイル、リンク、及びダウンロードファイルの作成を行います。

既に以下の開発キットをお持ちの方には、バージョンアップキットが用意されています。

DT - 700 用、DT - 750 用、DT - 800 用、

DT - 830 用、DT - 850 用、DT - 900 用

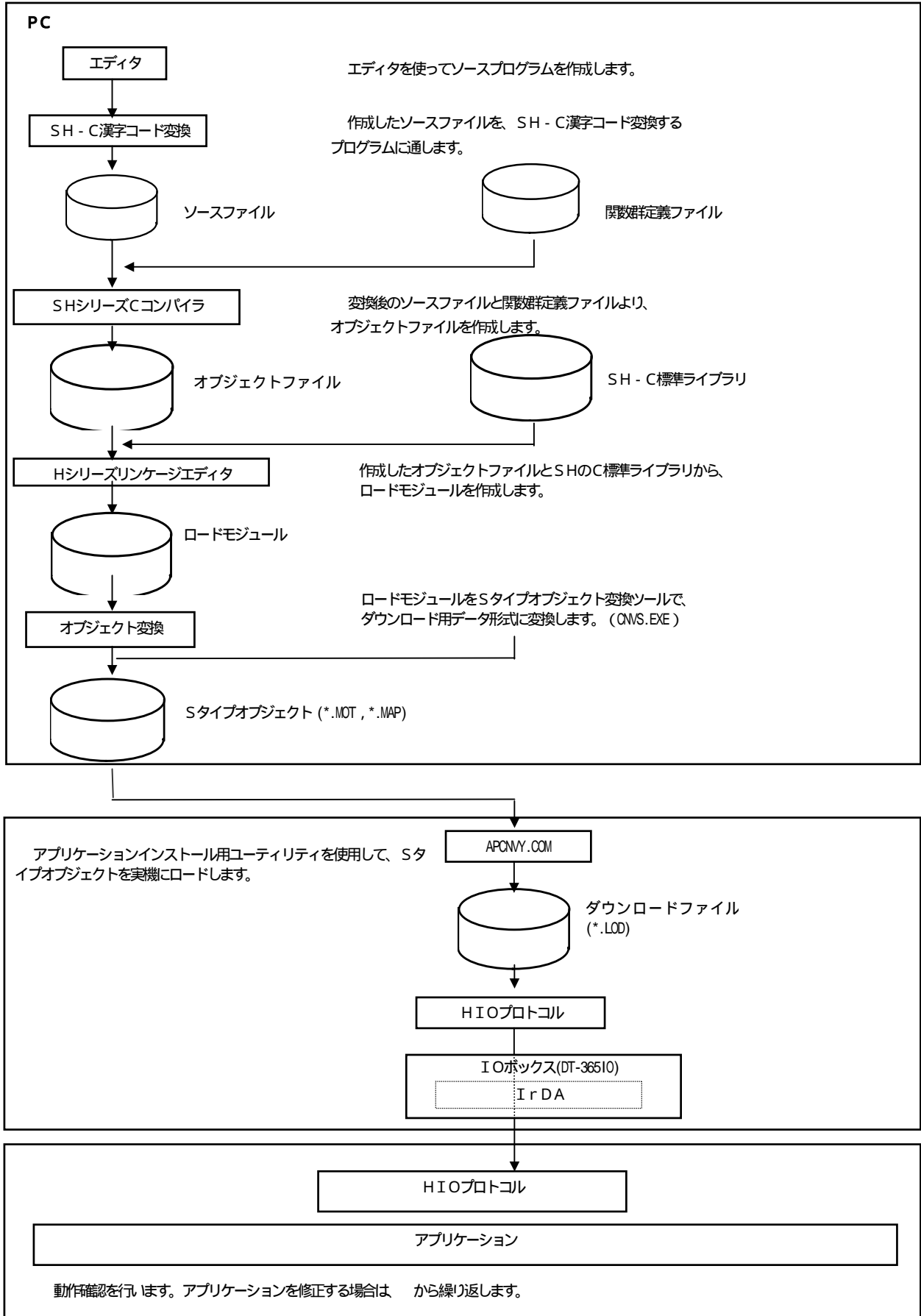
通信キット

ダウンロードファイルの PC から DT - 300 へのダウンロードに使用します。

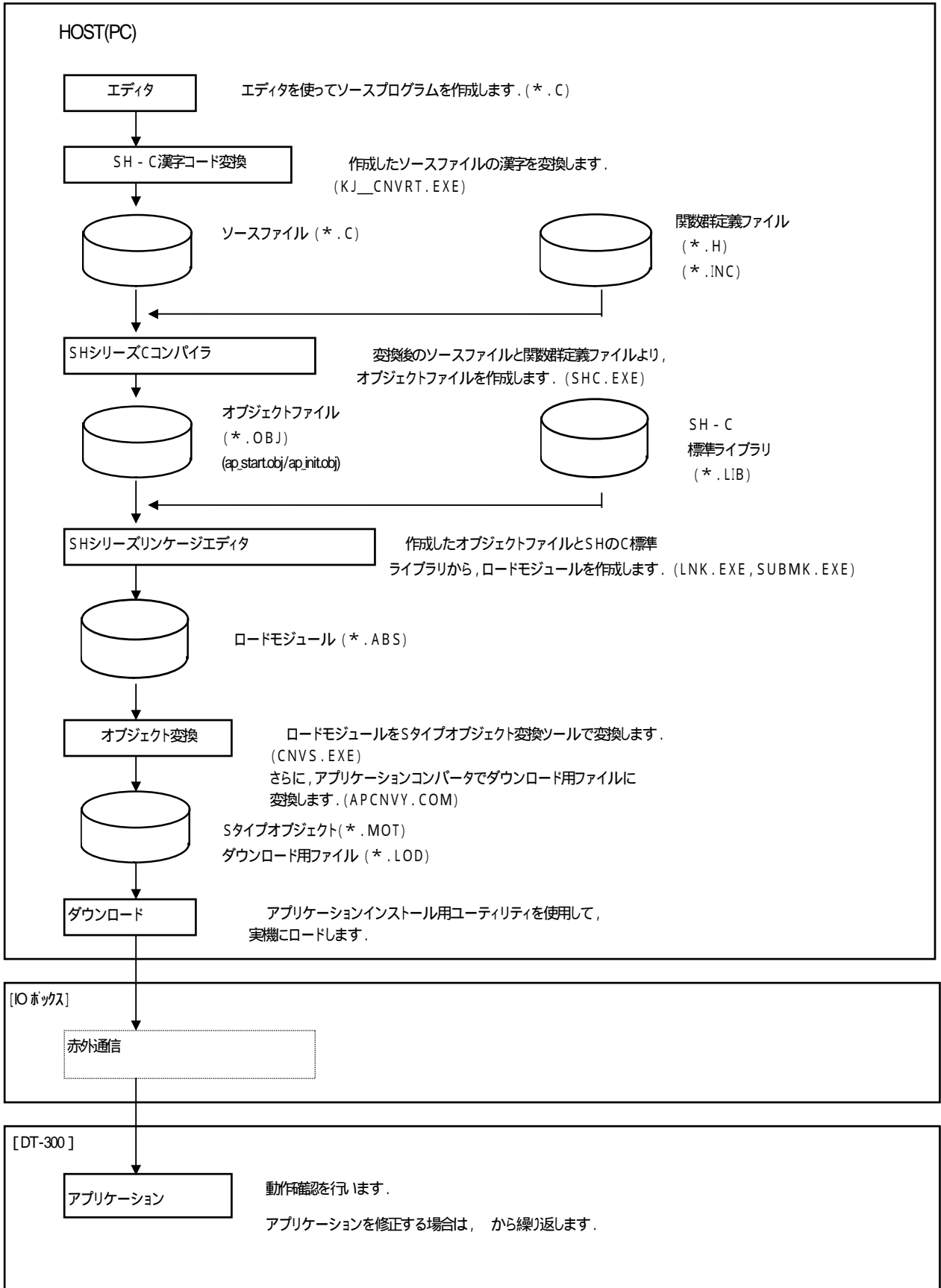
1 - 2 開発の流れ

処理の流れと各処理で必要となるプログラム名を次ページに記載します。

(1)ダウンロードにHIOWINを使用する場合



(2)ダウンロードにLMWINを使用する場合



2 . SHCコンパイラ開発環境

2 - 1 必要な開発環境

DT - 300 のアプリケーションプログラムを開発するには、以下の環境が必要になります。ただし、ユーザーズメモリについては、コンパイルするプログラムの大きさによって変動しますので、ここではあくまでも目安としてご参照ください。

ハードウェア	PC/AT互換機
CPU	80386SX以上
OS	IBM-DOS Ver J4.05/V 以上または同等以上のMS-DOS
ユーザーズメモリ	コンベンショナル 640KB + プロテクトメモリ 3MB 以上
空きディスク容量	3MB以上

表2 - 1 開発用機種と開発環境

2 - 2 コンパイラのインストール

2 - 2 - 1 ハードディスクへのコピー

開発環境の各ファイルを、任意のディレクトリへコピーしてください。
一般的には、次のようにするとよいと思われます。

¥	SHC¥	BIN	SHC コンパイラ本体エラーメッセージファイル
		LIB	標準ライブラリ関数群実行時ルーチン群
		INCLUDE	標準インクルードファイル
		LINK	リンク環境
DT300¥		OBJ	DT-300 専用オブジェクトファイル
		TOOLS	DT-300 専用ツール

表2 - 2 推奨ディレクトリ構成

なお、C言語開発キットに添付されているフロッピーディスクの内容は次の通りです。
また、バージョンアップキットには のみ添付されています。

コンパイラディスク 1

¥	BIN¥	DOS4G.EXE	SHC コンパイラ
		SHC.EXE	SHC コンパイラ
		SHCERR.MSG	エラーメッセージファイル
		SHCERR.OFF	エラーメッセージファイル
		SHCFRT.EXE	SHC コンパイラ
		SHCHLP.MSG	エラーメッセージファイル
		SHCPRM.EXE	SHC コンパイラ
		SHCTIL.EXE	SHC コンパイラ

表2 - 3 SHC コンパイラディスク 1 の内容

コンパイラディスク 2

¥	BIN¥	SHCASM.EXE	SHC コンパイラ
		SHCGEN.EXE	SHC コンパイラ
		SHCMDL.EXE	SHC コンパイラ

コンパイラディスク 3

¥	LIB¥		
		SHCLIB.LIB	SHC 標準ライブラリ
		SHCNPIC.LIB	SHC 標準ライブラリ
		SHCPIC.LIB	SHC 標準ライブラリ
		ASSERT.H	SHC 標準インクルードファイル
		CTYPE.H	SHC 標準インクルードファイル
		ERRNO.H	SHC 標準インクルードファイル
		FLOAT.H	SHC 標準インクルードファイル
		LIMITS.H	SHC 標準インクルードファイル
		MACHINE.H	SHC 標準インクルードファイル
		MATH.H	SHC 標準インクルードファイル
		MATHF.H	SHC 標準インクルードファイル
		SETJMP.H	SHC 標準インクルードファイル
		STDARG.H	SHC 標準インクルードファイル
		STDDEF.H	SHC 標準インクルードファイル
		STDIO.H	SHC 標準インクルードファイル
		STDLIB.H	SHC 標準インクルードファイル
		STRING.H	SHC 標準インクルードファイル

表 2 - 5 SHC コンパイラディスク 3 の内容

リンカディスク

¥		
	CNVS.EXE	S タイプオブジェクトコンバータ
	LNK.EXE	リンカ

表 2 - 6 リンカディスクの内容

専用ファイルディスク

¥	LIB¥		
		HICIF.LIB	ITORN ライブラリ
		H10LIB.LIB	H10 プロトコライブラリ (DT-360I0 通信ライブラリ)
	INC¥	BIOS1DEF.H	BIOS 関数定義ヘッダファイル
		BIOS1MAC.H	BIOS マクロ定義ヘッダファイル
		CMNDEF.H	BIOS 共通定義ヘッダファイル
		ITRON.H	ITRON ヘッダファイル
		H10LIB.H	H10 プロトコル定義ファイル
	OBJ¥	AP_START.OBJ	AP 初期化モジュール
		AP_INIT.OBJ	AP 使用変数初期化モジュール
		PATCH001.LOD	パッチファイル (注 5)
	TOOLS¥	KJ_CNVRT.EXE	漢字コード変換ツール
		SUBMK.EXE	ワークセッション先頭アドレス設定ツール
		APCNVY.COM	ダウンロードファイル作成ツール
	SAMPLE¥DEMO¥	APSMP.C	サンプルアプリケーション
		APSMP.MOT	
		APSMP.MAP	
		APSMP.LOD	
		APSMP.MAK	
		APSMP.SUB	
		CONFIG.HTS	環境設定ファイル
		CONFIG.PAS	代理店 ID 設定ファイル
		CONFIG.ID	機器 ID 設定ファイル
		ASTART.HTS	自動実行プログラム設定ファイル
		CONFIG.H10	通信用スクリプトファイル
	SAMPLE¥KEY¥	smp_key.c	キー入力サンプルソースファイル
		smp_key.h	キー入力サンプル定義ファイル

表 2 - 7 専用ファイルディスクの内容

専用ファイルディスクの内容は、DT-300 と他の機種では内容が異なります。
上記中の「 ¥ 」印の付いたファイルが実際に異なるファイルです。
複数の機種の開発環境を共存させる場合は混在させないように注意してください。

2 - 2 - 2 PATHの設定

コンパイラやリンカ、DT-300用ツールが格納されているディレクトリにPATHを設定します。

以下の例は「2 - 2 - 1 ハードディスクへのコピー」の冒頭で示した推奨するディレクトリ構成になっている場合のもので、

例 (既存のPATHに追加する場合)

```
SET PATH=%PATH%;C:¥SHC¥BIN;C:¥SHC¥LNK;C:¥DT300¥TOOLS
```

2 - 2 - 3 環境変数の設定

Cコンパイラシステム格納場所の設定 (SHC_LIB)

Cコンパイラシステム (Cコンパイラメッセージファイル) を格納した場所のパスを環境変数名「SHC_LIB」にセットします。

例 SET SHC_LIB=C:¥SHC¥BIN

ここで設定するのは、ライブラリファイルのある場所ではなく、コンパイラのある場所です。環境変数名が「SHC_LIB」となっている為、ライブラリファイルのある場所と間違えやすいので注意してください。

中間ファイル作成ディレクトリの設定 (SHC_TMP)

コンパイラは、プログラムのコンパイル処理中にいくつかの中間ファイルを生成します。環境変数「SHC_TMP」は中間ファイルの作成ディレクトリを設定します。

例 SET SHC_TMP=C:¥SHC

各環境変数名に設定できるパス名は、それぞれ1つずつだけです。

プロテクトメモリ領域指定

SHCコンパイラは増設されている拡張メモリを使用するため、コンパイラが使用するプロテクトメモリ領域を環境変数「DOS16M」に設定します。

ただし、設定が無ければコンパイラはVCP規格または、DPM規格でドライバにメモリを要求します。

PC/AT 互換機のメモリドライバは、基本的に上記規格に対応している為、大方の場合、環境変数「DOS16M」は設定しなくても構いません。
同様に Windows の DOS 窓で動作させる場合も、設定しなくてかまいません。

<書式 1 >

SET DOS16M=1[@<先頭アドレス>] [:<サイズ>]

<書式 2 >

SET DOS16M=1[@<先頭アドレス>] [-<終了アドレス>]

[] 内は省略可能を意味します。

アドレス及びサイズは 1 ~ 16 MB まで指定可能です。

- ・先頭アドレス：コンパイラが使用するメモリ領域の先頭アドレスを指定します。
- ・終了アドレス：コンパイラが使用するメモリ領域の狩猟アドレスを指定します。
- ・サイズ：コンパイラが使用するプロテクトメモリの容量を指定します。

設定項目は 10 進または、16 進で行います。

ただし、16 進の場合はプリフィックスの「0x」が必要です。

また数字は K バイト / M バイト 単位で指定できますが、サフィックスが無い場合は K バイトであると見なします。

例 1 (拡張メモリを 4 MB 実装し、1 MB から 4 MB の 3 MB を使用する場合)

SET DOS16M=1@1M-4M

或いは、

SET DOS16M=1@1M:3M

例 2 (拡張メモリを 4096 K 使用する場合)

SET DOS16M=1:4096K

VCPI (Virtual Control Program Interface)

メモリ管理規格の 1 つで、仮想 86 モードで使う EMS ドライブとプロテクトメモリと使う DOS エクステンダを調整し、メモリを効率よく配分する為の規格。

DPMI (Dos Protect Mode Interface)

メモリ管理規格の 1 つで、MS-DOS をベースにマルチタスク環境でのアプリケーションプログラムがプロテクトメモリを効率よく使用できるようにした規格。

2 - 2 - 4 CONFIG.SYS ファイルの設定**FILES**

プログラムの処理中に、少なくとも 20 個のファイルを同時にオープンできるように設定する必要があります。

FILES のオペランドには 20 以上の値を設定して下さい。

例 FILES=20

EMS ドライバ

EMS ドライバを組み込んだ状態で拡張メモリを使用する際は、EMS ドライバが VCPI / DPMI 規格に対応している場合のみ、SHC コンパイラと共存することができます。

例 (CONFIG.SYSファイル)

```
FILES = 20  
BUFFERS = 20  
DEVICE = C:\¥TOOL¥HSB.EXE V -X  
DEVICE = C:\¥DOS¥VMM386.EXE /I /U  
SHELL = C:\¥COMMAND.COM A:¥ /P /E:1024  
DEVICEHIGH = C:\¥SYS¥RSDRV.SYS  
DEVICEHIGH = C:\¥SYS¥MOUSE.SYS  
DEVICEHIGH = C:\¥SYS¥PRINT.SYS
```

STACKS

STACKSは外して下さい。

WindowsのDOS窓を使用する場合には、CONFIG.SYSを変更する必要はありません。

2 - 3 コンパイル

コンパイルには次の2つの作業が必要になります。

- ・漢字コード変換 (KJ_CNVRT.EXE)
- ・コンパイル (SHC.EXE)

コンパイルが完了するとオブジェクトファイル (*.OBJ) が生成されます。

2 - 3 - 1 漢字コード変換 (KJ_CNVRT.EXE)

漢字変換ツールは、変換元プログラムの漢字コードを16進数値データに変換して、結果を変換先ファイルに出力します。

《書式》 KJ_CNVRT <変換元ファイル> <変換先ファイル>

例 C:\¥DT300¥SAMPLE>KJ_CNVRT SAMPLE.SRC SAMPLE.C

SAMPLE.SRC (変換前のファイル)

```
char dsp_data[ ]="か";
void sample(){          /*コメント*/
    lcd_string(LCD_ANK_STANDARD,LCD_ATTR_NORMAL,(UB *)dsp_data[0],LCD_LF_OFF);
}
```

SAMPLE.C (変換後のファイル)

```
char dsp_data[ ]="¥xb6¥xc5";
void sample(){          /*コメント*/
    lcd_string(LCD_ANK_STANDARD,LCD_ATTR_NORMAL,(UB *)dsp_data[0],LCD_LF_OFF);
}
```

本ツールは、シフト JIS で記述された C プログラムを対象にしています。
文字 (' ') 内の 2 バイト全角文字を、対応する 1 バイト半角文字コードに変換することはありません。環境変数「DOS16M」は設定しなくても構いません。
アプリケーション用のヘッダファイルに漢字変換が必要な文字列が存在する場合も、C ソースコードと同様に変換処理を行ってください。

2 - 3 - 2 コンパイルの実行

《書式》 SHC <オプション> <ソースファイル名>

オプション

/ I = パス名 : インクルードファイル取込先パス名設定

/ L : リストファイル出力先指定

/ OP = n : n = 0 は最適化無し n = 1 は最適化あり

コンパイルオプションの詳細については「SHシリーズCコンパイラユーザーズマニュアル」の「コンパイラオプション」の項をご参照ください。

2 - 3 - 3 コンパイルの実行例

```
C:¥DT300¥SAMPLE>KJ_CNVRT SAMPLE.SRC SAMPLE.C          . . . a
C:¥DT300¥SAMPLE>SHC /i=.,C:¥DT800¥INCLUDE;C:¥SHC¥INCLUDE SAMPLE.C . . . b
```

- a . 漢字変換を実行します。
- b . コンパイルを実行します。

2 - 4 リンク

リンクの作業には次の3つがあります。

- ・リンク (LNK . EXE)
- ・ B , R セクションの先頭アドレスの補正 (SUBMK . EXE)
- ・再リンク (LNK . EXE)

リンク完了時にはロードモジュール (*.ABS) とマップファイル (*.MAP) が生成されます。

2 - 4 - 1 リンク環境の設定

まず最初に「SUBCOMMAND」ファイルを作成します。

これはリンク時の環境を設定するものです。テキストエディタ等で作成してください。拡張子は「.SUB」と付けます。

主な設定内容は次の通りです。

```

a {
  INPUT C:¥SHC¥OBJ¥AP_START.OBJ,&
        C:¥SHC¥OBJ¥AP_INIT.OBJ,&
        C:¥DT800¥AP¥SAMPLE.OBJ

  LIBRARY C:¥SHC¥LIB¥SHCLIB.LIB
  LIBRARY C:¥SHC¥LIB¥HICIF.LIB

b →
c →
  ROM (D,R)
  START P,C,D,hicif(09048030)
  OUTPUT .¥SAMPLE
  PRINT .¥SAMPLE
  FORM A
  EXIT

```

- a . リンク対象オブジェクトファイルの指定
DT - 300 専用ファイル (AP_START.OBJ , AP_INIT.OBJ) と作成したアプリケーションプログラムを指定します。
- b . ROM化支援機能の指定
初期値付き変数領域を初期化コードと変数領域に分割するために、D , Rセッションを指定して下さい。
- c . アプリケーション開始アドレスの指定
次の値を指定して下さい。
・ 09048030

SUBCOMMANDファイルについては、「SH - Cリンケージエディタ ライブラリアンオブジェクトコンバータ ユーザーズマニュアル」を参照して下さい。

2 - 4 - 2 リンクの実行

《書式》 LNK -SUBCOMMAND=<SUBCOMMANDファイル名>

2 - 4 - 3 B , Rセッション先頭アドレス補正ツール (SUBMK.EXE)

B , Rセッション先頭アドレス補正ツールは、マップファイルよりCセッションの終了アドレスを求め、SUBCOMMANDファイルの書き換えを行います。
そこでこのツールを実行後、再度リンクを行わなければなりません。

本ツールはサブコマンドファイルを直接書き換えてしまう為、事前に対象ファイルのバックアップを取る必要があります。
補正はバックアップしたサブコマンドファイルに対して行ってください。

2 - 4 - 4 再リンクの実行 (LNK.EXE)

《書式》 LNK -SUBCOMMAND=<SUBCOMMANDファイル名>

2 - 4 - 5 リンクの実行例

```
C:¥DT300¥SAMPLE>COPY SAMPLE.SUB SUBFILE.SUB      . . . . . a
C:¥DT300¥SAMPLE>LNK -SUBCOMMAND=SUBFILE.SUB      . . . . . b
C:¥DT300¥SAMPLE>SUBMK SUBFILE.SUB                . . . . . c
C:¥DT300¥SAMPLE>LNK -SUBCOMMAND=SUBFILE.SUB      . . . . . d
```

- a . cで実行する「SUBMK . EXE」はSUBCOMMANDファイルを更新する為、サブコマンドファイルを補正用サブコマンドファイルにコピーします。
- b . B , Rセッション先頭アドレス補正ツール実行用のマップファイル (*.MAP) を作成する為に1度リンクを実行します。
- c . B , Rセッション先頭アドレス補正ツールを実行します。
- d . 補正されたサブコマンドファイルで再度リンクします。

2 - 5 オブジェクト変換

オブジェクト変換の作業には以下の2つがあります。

オブジェクト変換を行うことにより、Sタイプオブジェクト (*.MOT) が生成されます。

Sタイプオブジェクト (*.MOT) を、さらにダウンロードファイルへ変換することで、実際にDT-300へダウンロードするファイル (*.LOD) が作成されます。

オブジェクト変換	(CNVS.EXE)
ダウンロードファイルへの変換	(APCNVY.COM)

2 - 5 - 1 オブジェクト変換 (CNVS.EXE)

Sタイプオブジェクト (*.MOT) の生成を行います。

《書式》 CNVS <ロードモジュール名(*.ABS)>
拡張子「*.ABS」は省略できます。

例 C:¥DT300¥SAMPLE>CNVS SAMPLE

2 - 5 - 2 ダウンロードファイルへの変換 (APCNVY.COM)

実際にDT-300にダウンロードするダウンロードファイルを作成します。

《書式》 APCNVY <Sタイプオブジェクト名(*.MOT)> <ダウンロードファイル名(*.LOD)>

例 C:¥DT300¥SAMPLE>APCNVY SAMPLE.MOT SAMPLE.LOD

ファイル変換を行う時には、同一ファイル名の「*.MAP」ファイルが必要です。
リンク実行時に作成されたMAPファイルを同一ディレクトリに用意してください。

実際にDT-300に格納されるアプリケーションプログラムの容量は、ダウンロードファイルの
ファイル容量ではありません。
容量の計算は「2-7 アプリケーション容量の確認」を参照してください。

2 - 6 MAKE

今まで説明した作業は、バッチファイルまたは、MAKEファイルに登録しておくことで便利です。これまでの操作をMAKEファイル記述した例を以下に述べます。ただし、本製品にはMAKEプログラムは添付されておりません。MAKEファイルを使用する場合には別途「MASM」か、「MS - C Ver. 7.0A」のMAKEプログラムをご用意ください。なお、「MASM」の“MAKE.EXE”と「MS - C」の“NMAKE.EXE”では書式が異なりますので注意してください。以下のサンプルは、MAKEの記述例です。

```

=====
#                               DT - 300 makefile sample                               #
=====
# Macro #
HEADER      =c:%dt300%include%cmndef.h  c:%dt300%include%bios1mac.h
AP          =SAMPLE

# Kanji #
$(AP).c     :$(AP).src
            kj_cnvrt      $(AP).src  $(AP).c

# Compile #
$(AP).obj   :$(AP).c  $(HEADER)
            shc /i=.,c:%dt300%include;c:%shc%bin  $(AP).c > err.c

# Link #
$(AP).abs   : $(AP).obj  $(HEADER)
copy        $(AP).sub  subfile.sub
lnk         -SUBCOMMAND=subfile.sub
submk       subfile.sub
lnk         -SUBCOMMAND=subfile.sub

# Stype #
$(AP).mot   : $(AP).abs
            cnvs  $(AP).abs

# Download #
$(AP).lod   : $(AP).mot
            apcnvy  $(AP).mot

```

2 - 7 アプリケーション容量の確認

リンク完了後、マップファイル(*.MAP)を参照することにより、アプリケーションの開始アドレスと終了アドレスが確認でき、その差の値がアプリケーション容量となります。

単に、容量が足りるかを確認するだけであれば、終了アドレスがアプリケーション領域内に入っているかを確認する方法もあります。ただし、メモリサイズ変更処理を行った場合は、終了アドレスが変わりますので注意が必要です。

製 品 名	アプリケーション領域の容量	アプリケーション領域
DT-300	約336Kbyte	09048030H ~ 0909C000H

表2 - 8 デフォルトのアプリケーション領域

万一、容量がオーバーした場合はメモリサイズの変更処理が必要です。
DT-300 はシステムメニューからアプリケーション領域の容量が指定できます。

【マップファイル参照方法】

マップファイルでは、各セクションの情報が次の表のように表示されます。
アプリケーションの開始アドレスと最終アドレスを調べるには、マップファイルの「*SECTION NAME*」と「*TOTAL ADDRESS*」の2ヶ所を参照します。

SECTION NAME	START	-	END	LENGTH	UNIT NAME
P	H'????????	-	H'????????	H'????????	AP_START
	⋮		⋮	⋮	⋮
	⋮		⋮	⋮	⋮
* TOTAL ADDRESS *	<u>H'????????</u>		H'????????	H'????????	
	⋮				
	⋮				
SECTION NAME	START	-	END	LENGTH	UNIT NAME
hicif	H'????????	-	H'????????	H'????????	AP_START
	⋮		⋮	⋮	⋮
	⋮		⋮	⋮	⋮
* TOTAL ADDRESS *	H'????????	-	H'????????	H'????????	
	⋮				
	⋮				
SECTION NAME	START	-	END	LENGTH	UNIT NAME
ATTRIBUTE: DATA NOSHR RAM					
R	H'????????	-	H'????????	H'????????	AP_START
	⋮		⋮	⋮	⋮
	⋮		⋮	⋮	⋮
* TOTAL ADDRESS *	H'????????	-	<u>H'????????</u>	H'????????	

図2 - 1 マップファイル

アプリケーションの先頭アドレスは、「SECTION NAME」が「P」の「TOTAL ADDRESS」行の「START」部分のアドレスです（ここでは09048030になっているはず）。

アプリケーションの最終アドレスは、「SECTION NAME」が「R」の「TOTAL ADDRESS」行の「END」部分のアドレスです。

この2つのアドレスの差がアプリケーションのファイル容量となります。

3. プログラムダウンロード

3-1. 動作環境

3-1-1 HIOWINによるダウンロード

プログラムをダウンロードするためのツールとして通信キット「HIOWIN」と「LMWIN」を用意しています。（LMWINについては、次項で説明します）
HIOWINはPC～IOボックス間でファイル転送を行うためのツールです。IOボックス～DT-300間のファイル転送は、DT-300のシステムメニューで用意されている通信機能を利用します。

ハードウェア	PC/AT互換機
CPU	Pentium133MHz以上
OS	WindowsNT4.0SP5以上 Windows2000

表3-1 HIOWINの動作環境

HIOWINでダウンロードを行う場合のシステム構成を次に図示します。

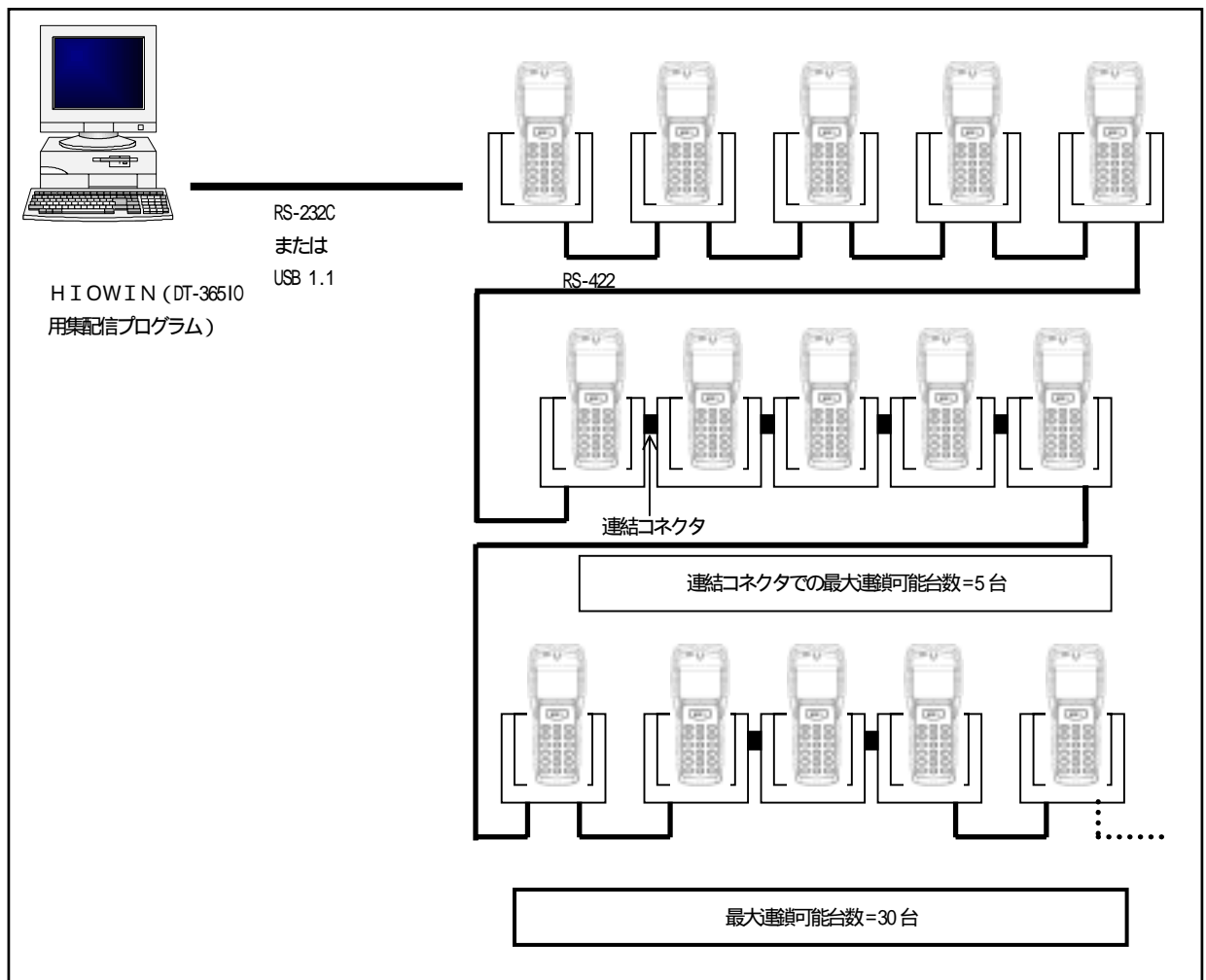


図3-1 HIOWINのシステム構成

- 1 RS-232Cケーブルは、下記製品をご用意ください。
DT-887AX
- 2 USBケーブルは、下記製品をご用意ください。

DT - 380USB

3 RS - 422ケーブルは、下記の製品をご使用ください。

DT - 888RSC

また、接続は上位側I/OボックスのC - OUTから下位側I/OボックスのC - INIにつなげて下さい。

I/Oボックスのディップスイッチ内容及び設定例は下記の通りです。

Sw-1	IrDA 通信速度	Sw-2	IrDA 通信速度
Sw-3	動作モード	Sw-4	動作モード
Sw-5	終端	Sw-6	RS232C 通信速度
Sw-7	RS232C 通信速度	Sw-8	RS232C 通信速度
Sw-9		Sw-10	
Sw-11	動作モード	Sw-12	

IrDA通信速度

Sw-1	Sw-2	
OFF	OFF	38400 bps
ON	OFF	115200 bps
OFF	ON	予約
ON	ON	4M bps

RS232C通信速度

Sw-6	Sw-7	Sw-8	
OFF	OFF	OFF	2400 bps
ON	OFF	OFF	4800 bps
OFF	ON	OFF	9600 bps
ON	ON	OFF	19200 bps
OFF	OFF	ON	38400 bps
ON	OFF	ON	57600 bps
OFF	ON	ON	115200 bps

動作モード

Sw-3	Sw-4	Sw-11	
OFF	OFF	ON	Master
ON	OFF	ON	Slave
OFF	ON	ON	予約
ON	ON	ON	予約

終端

b5	
OFF	無効(非終端)
ON	有効(終端)

以下に例を記述します

1. マスター&終端の場合 (I/Oボックスが一つの場合)

	Sw-1	Sw-2	Sw-3	Sw-4	Sw-5	Sw-6	Sw-7	Sw-8	Sw-9	Sw-10	Sw-11
ON											
OFF											

2. マスター&終端でない場合 (I/Oボックスが複数の場合)

	Sw-1	Sw-2	Sw-3	Sw-4	Sw-5	Sw-6	Sw-7	Sw-8	Sw-9	Sw-10	Sw-11
ON											
OFF											

3. スレーブ&終端の場合 (I/Oボックスが複数の場合)

	Sw-1	Sw-2	Sw-3	Sw-4	Sw-5	Sw-6	Sw-7	Sw-8	Sw-9	Sw-10	Sw-11
ON											
OFF											

4. スレーブ&終端でない場合 (I/Oボックスが複数の場合)

	Sw-1	Sw-2	Sw-3	Sw-4	Sw-5	Sw-6	Sw-7	Sw-8	Sw-9	Sw-10	Sw-11
ON											

OFF												
-----	--	--	--	--	--	--	--	--	--	--	--	--

マスター：上位側がPCと接続しているIOボックス
スレーブ：上位側がIOボックスと接続しているIOボックス

3 - 1 - 2 LMWINによるダウンロード

LMWINは、I/Oボックスを経由してホスト（パソコン）と通信を行うツールです。動作環境は次の通りです。

ハードウェア	PC/AT互換機
CPU	i486以上
OS	MS-Windows95
ユーザースメモリ	16MB以上(コンベンショナル 640KB)

表3 - 2 LMWINの動作環境

LMWINでダウンロードを行う場合のシステム構成を次に図示します。

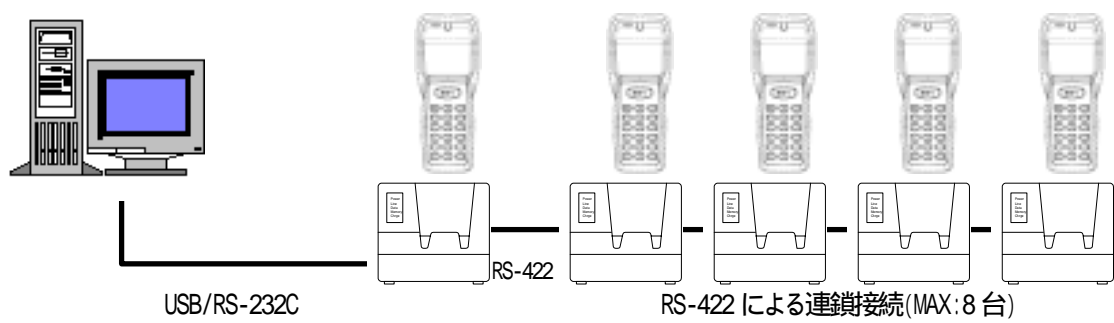


図3 - 2 LMWINのシステム構成

3 - 2 ダウンロードファイル

DT - 300でプログラムを動作させるために必要なファイルは、以下のファイルです。
これらのファイルをAドライブ(RAM)またはBドライブ(FROM)にダウンロードします。
アプリケーション起動時に、CONFIG.HTSの内容にしたがってシステム環境が設定され、ASTART.HTS
で指定されているプログラムが起動します。

プログラムファイル (xxxxx.LOD)	ユーザー作成プログラム
システム環境ファイル (CONFIG.HTS)	このファイルのダウンロードは任意
自動起動ファイル (ASTART.HTS)	必須
パッチファイル (PATCH001.LOD)	必須

3 - 2 - 1 システム環境ファイル (CONFIG.HTS)

DT - 300のシステム環境設定を登録したファイルで、テキスト形式で設定値を記述します。
ファイル名は「CONFIG.HTS」固定で、先頭に識別子「CONFIG.HTS」を持ちます。

項目	位置	サイズ	設定範囲	既定値		
ID	00	10	CONFIG.HTS 固定	CONFIG.HTS		
電源	APO 時間	+10	2	00-59(分)	10	
	ABO 時間	+12	2	00,10-59(秒)	15	
	ビューム	+14	2	00:OFF / 01:ON	01	
KEY	外音	+16	2	00:OFF / 01:ON	01	
OBR	読取回数	+18	2	01-09(回)	01	
	照合回数	+20	2	01-09(回)	03	
	スキャン時間	+22	2	01-09(秒)	04	
表示	MODE	+24	2	00:6dot / 01:8dot / 02:10dot	01	
	日/英	+26	2	00:日本語 / 01:英語	00	
	種別	+28	2	00:NORMAL / 01:BOLD	00	
	エントリ	+30	2	00-15(段)	07	
通信	共通	プロトコル	+32	2	01:FLINK / 03:高速 I O (H I O)	01
	個別	速度 (IR)	+36	2	02:2400 / 03:4800 / 04:9600 / 05:19200 / 06:38400 07:57600 / 08:115200(bps)	08
		データ (IR)	+38	2	07:7bit / 08:8bit	08
		パリティ (IR)	+40	2	00:NON / 01:EVEN / 02:ODD	00
		STOP (IR)	+42	2	00:1bit / 01:2bit	00
タイマ	音量	+60	2	00:OFF / 01:小 / 02:中 / 03:大	02	
プロトコル	下記参照	+62				
サイズ	AP サイズ	+79	5	M0128-M0992(kbyte)	M0336	

→ プロトコル関連：高速 I O (H I O)

項目	位置	サイズ	設定範囲	既定値
タイムアウト	+62	4	0-9000(ミリ秒)	3000
リトライ回数	+66	4	0-99(回)	0005
リザーブ	+70	6	000000	000000

プロトコル関連：FLINK

項目	位置	サイズ	設定範囲	既定値
セッション確立タイムアウト	+62	4	0000-3600(秒)	1800
受信タイムアウト	+66	4	0000-0600(秒)	0300
セッション終了タイムアウト	+70	4	0000-0600(秒)	0010
リザーブ	+74	2	00	00

表 3 - 3 システム環境ファイルの構造

ファイルの内容は DT-700、DT-750、DT-800、DT-800、DT-850、DT-900 とは異なりますが、ファイル名は同一ですので混在しないように注意してください。

3 - 2 - 2 自動起動ファイル (ASTART.HTS)

DT - 3 0 0 のシステムメニューで “ A Pキドウ ” にて起動するアプリケーションを記述します。ここに記述されているプログラムファイルをアプリケーションエリアに展開して起動します。

<書式> パス名+ファイル名

<記述例> A:¥SAMPLE.LOD

3 - 3 ダウンロード

3 - 3 - 1 HIOWINによるダウンロード

3 - 3 - 1 - 1 ダウンロード手順

PCよりDT - 300にプログラムをダウンロードするまでの手順は以下の通りです。

HIOWINを使用して、ダウンロードファイルをIOボックスにコピーします。

DT - 300よりシステムメニューのファイル受信機能を使用して、ファイルを受信します。

3 - 3 - 1 - 2 HIOWINの操作

HIOWINをインストールすると以下のような階層でフォルダが作成されます。(デフォルトではC:\Program Files以下に作成されます。)

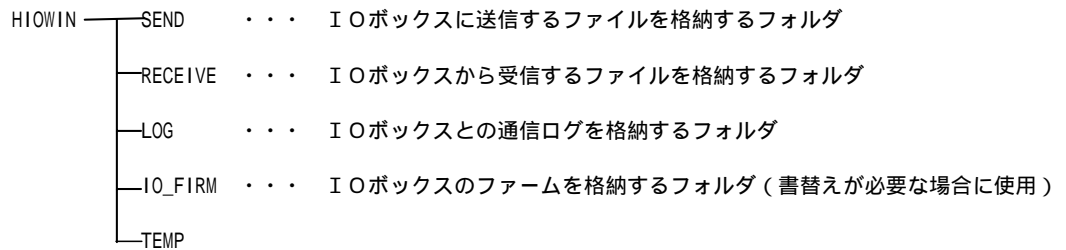


図3 - 3 フォルダ構成

3 - 3 - 1 - 2 - 1 ファイルの準備

(1) ダウンロードファイルの準備

DT - 300にダウンロードする必要のあるファイルを SEND フォルダにコピーします。

(2) スクリプトファイルの準備

IOボックスからDT - 300にダウンロードするファイルを記述したスクリプトファイル (CONFIG.HIO) を作成し、SEND フォルダに格納します。

<記述例>

```

/R A:¥CONFIG.HTS A:¥
/R A:¥ASTART.HTS A:¥
/R A:¥APSMP.LOD A:¥
/R A:¥PATCH001.LOD A:¥
  
```

上記例では、IOボックスのAドライブ(RAM)に入っている各ファイルをDT - 300のAドライブ(RAM)にコピーします。

詳細は、HIOWIN操作説明書をご参照ください。

(3) HIOWIN初期設定の設定

HIOWIN フォルダ内の初期設定ファイル (HIOWIN.INI) の[FILES]の設定でIOボックスにダウンロードするファイルを記述します。

<記述例>

```

[FILE]
#RAM(A:) FROM(B:)
  
```


SendFile01=A:¥
 SendFile02=
 SendFile03=
 SendFile04=
 SendFile05=

上記例では、SEND フォルダに格納されている全ファイルを I Oボックスの Aドライブ (RAM) へコピーします。

詳細は、HIOWIN 操作説明書をご参照ください。

3 - 3 - 1 - 2 - 2 HIOWINの起動

HIOWIN フォルダの HIOWIN.EXE を起動すると次のような画面が起動します。

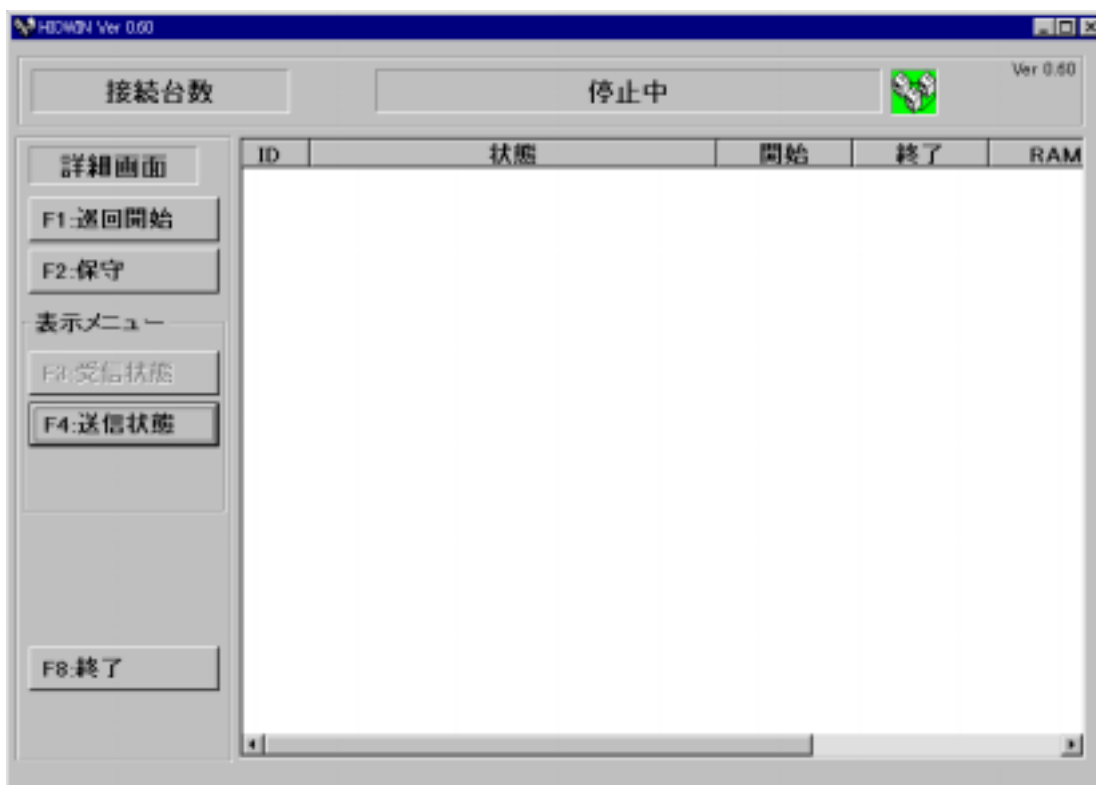


図3 - 4 HIOWIN初期画面

初めて、PCとI Oボックスを繋いでHIOWINを使用する場合、「保守」機能により、以下の手順でI Oボックスの設定を行ってください。

I Oボックスのディップスイッチの確認

PC・I Oボックス間がUSB又はRS 2 3 2 Cで正しく接続されているか確認

HIOWINの「保守」機能により、COM設定を行います。

HIOWINの「保守」機能により、ID設定を行います。

HIOWINの「保守」機能により、RAMクリアを行います。

HIOWINの「保守」機能により、FROMクリアを行います。

I Oボックスの設定は以上です。詳細はHIOWIN操作解説書をご参照ください。

3 - 3 - 1 - 2 - 3 PCからI/Oボックスへの送信

初期画面から“保守”を選択します。

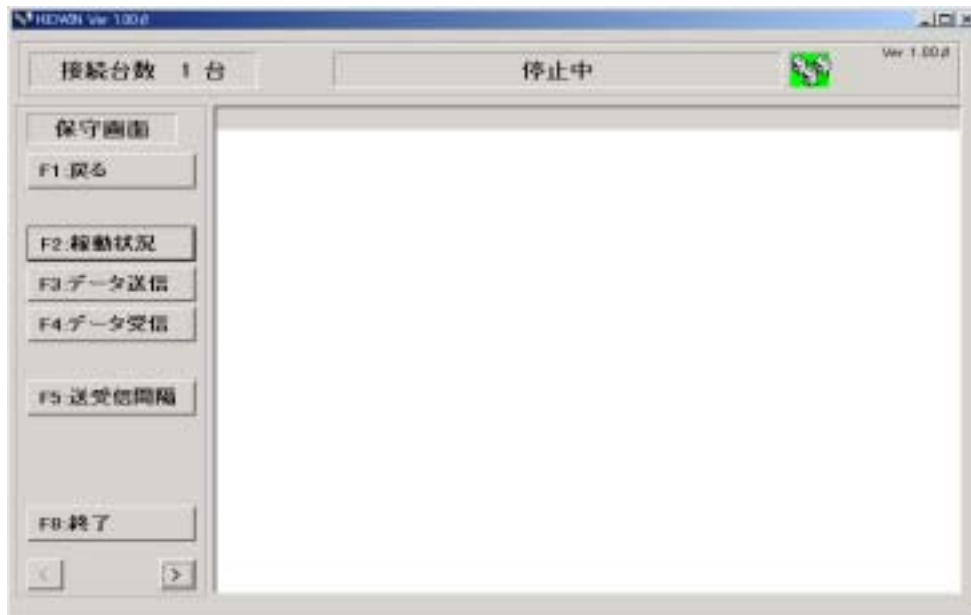


図3 - 4 保守画面

“データ送信”を押下することにより HIOWIN.INI で設定しているファイルをI/Oボックスに送信します。送信が完了するとI/Oボックスの状態が表示されます。

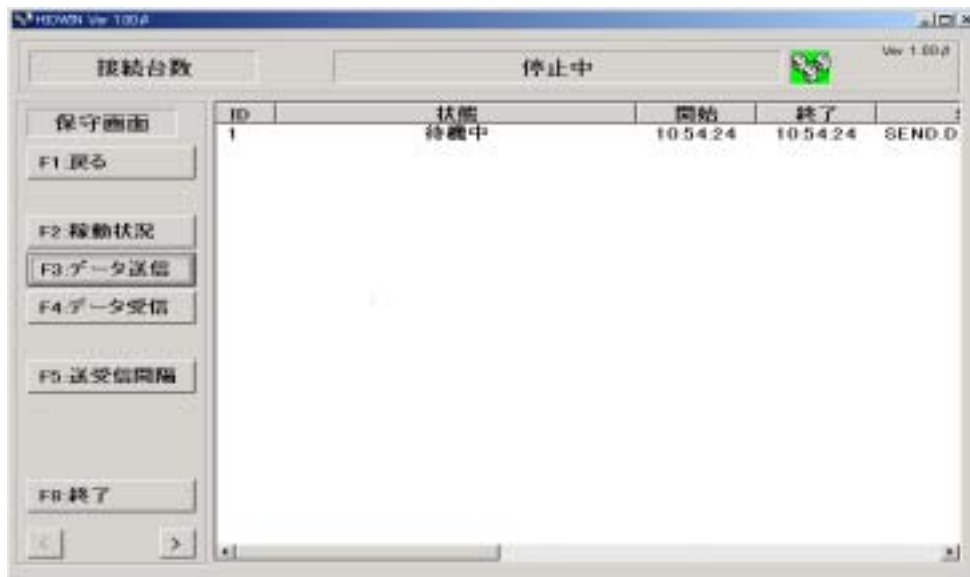


図3 - 5 保守画面 (データ送信完了時)

3 - 3 - 1 - 3 DT - 300 操作

IOボックスに格納されているファイルをDT - 300で受信するには、DT - 300をI/Oボックスに載せて、次の手順でDT - 300本体側の準備を行い、受信を開始します。

- a . 「F」キーと「決定」キーを押しながら電源をオンします（システムメニュー起動）。
- b . 「4：テンソウ」を選択します。
- c . 「4：プロトコル」を選択します。
「2：コウソクIO」を選択して「決定」キーにて確定します。
- d . 「3：ユーティリティ」を選択します。
「1：ファイルテンソウ」を選択します。

以上で通信待機状態になります。

DT - 300がIOボックスに乗っていれば通信が開始されます。

ダウンロード完了後は「クリア」キーを押して、「システムメニュー」のトップページに戻り、「1：APキドウ」を行うと、アプリケーションプログラムが起動します。

3 - 3 - 2 LMWINによるダウンロード

3 - 3 - 2 - 1 LMWINの操作

LMWINを起動すると、次の画面が表示されます。

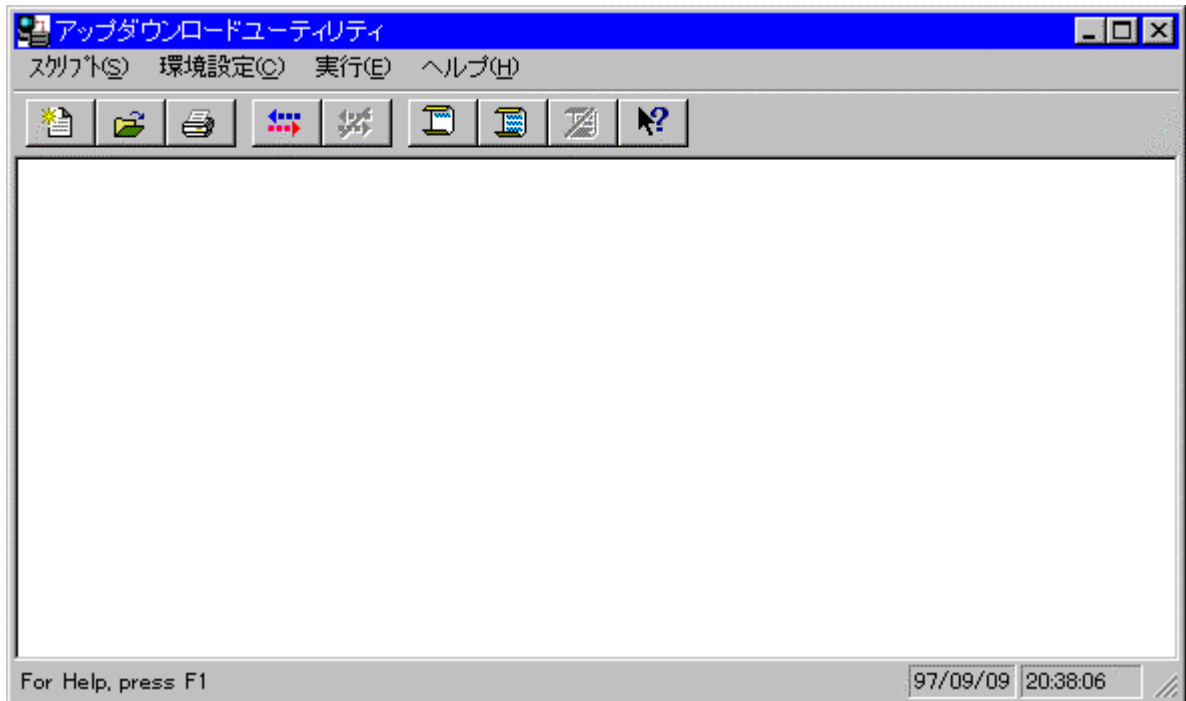


図3 - 17 LMWINのメインメニュー画面

3 - 3 - 2 - 2 LMWINのスクリプト作成

まず、スクリプトメニューから新しいスクリプトを作成します。

「新規作成サブメニューを」選択してください。

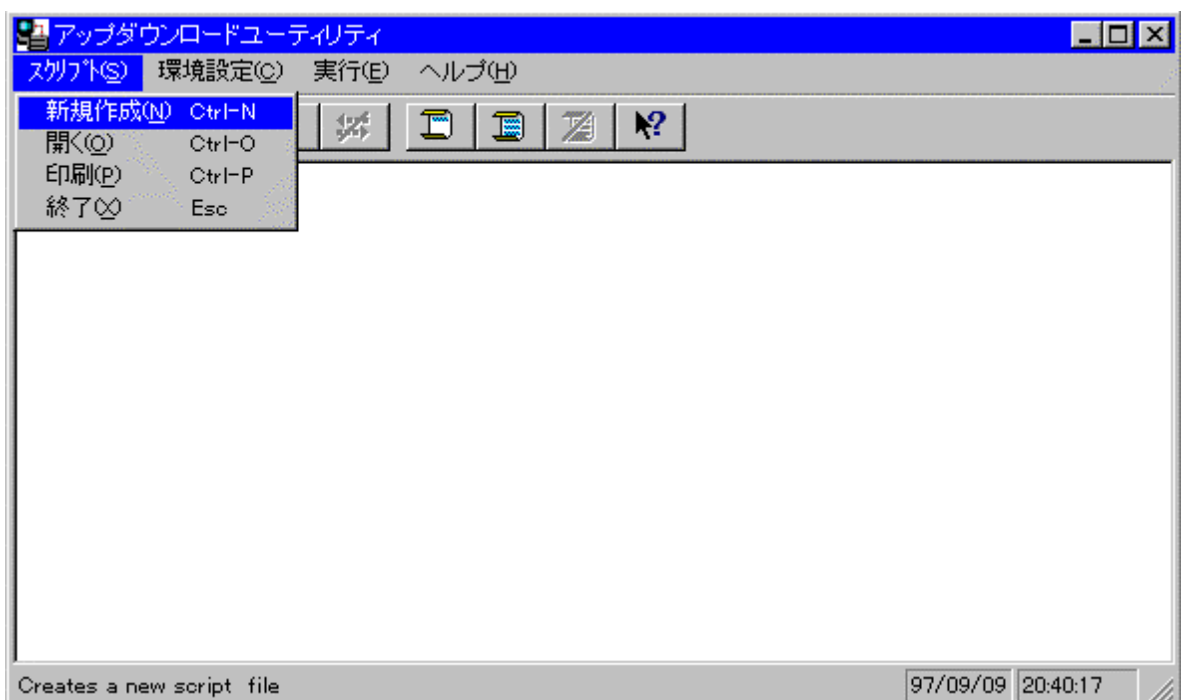


図3 - 18 LMWINのスクリプトメニュー画面

3 - 3 - 2 - 3 LMWINのスクリプト新規作成

ここで「新規作成」サブメニューオプションを選ぶと、下の画面になります。
(この画面は「追加」を押す前のコマンドとオプションが選択された画面です)

この画面よりコマンドボックスから実行するコマンドを選びます。
コマンドを選択すると、コマンドと一致しているオプションを入力/選択しなければなりません。例えば、もし選ばれたコマンドがsendならば、「ファイル」、「格納ディレクトリ」のオプションを入力し、必要であれば「オプション」を選択します。
その後、[追加]ボタンを押す事によりスクリプトに追加されます。



図3 - 19 LMWINのスクリプトファイル新規作成画面

この画面上のボタンの意味は次の通りです。

- | | |
|------------|------------------------------------|
| [追加] | ... コマンドを「スクリプト」に追加します。 |
| [結合] | ... 前のスクリプトコマンドに結合します |
| [戻る] | ... メイン画面に戻ります。スクリプトファイルのセーブはしません。 |
| [保存] | ... スクリプトファイルを上書き保存します。 |
| [名前を付けて保存] | ... 名前を付けて保存します。 |
| [参照] | ... ユーザがHTに送るファイルを選択できます。 |

3 - 3 - 2 - 4 LMWINのスク립トを開く

新規作成の「開く」サブメニューオプションを選ぶとスク립トファイル選択画面が表示されます。

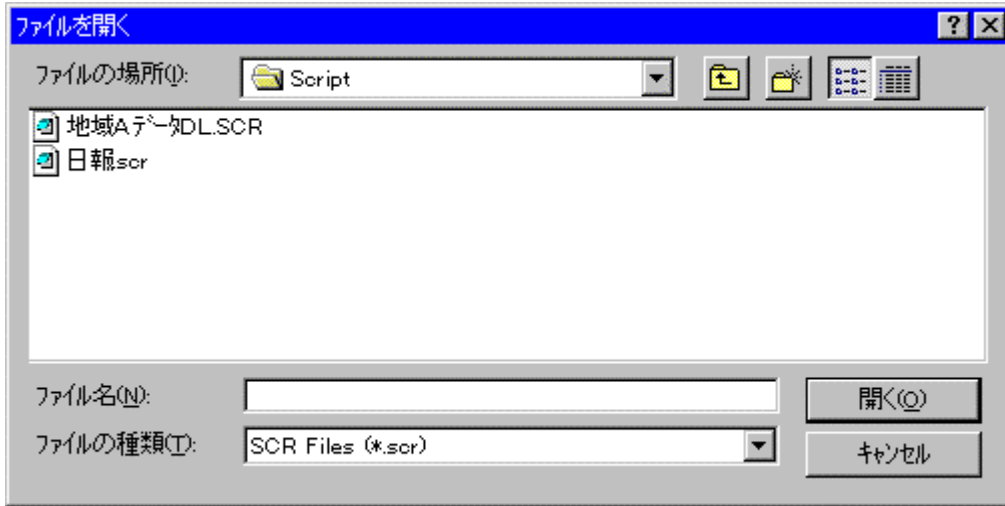


図3 - 20 スクリプトファイルを開く画面その1

スク립ト選択画面でスク립トファイルを選択すると次の画面になります。

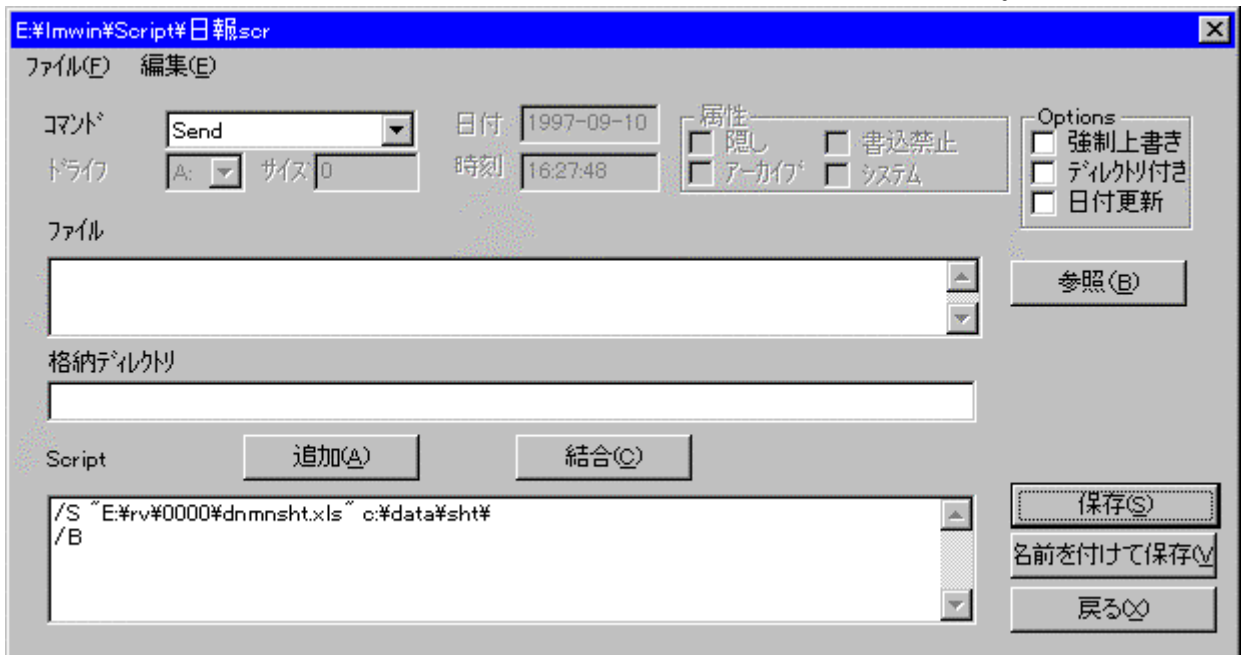


図3 - 21 スクリプトファイルを開く画面その2

「スク립トファイル」に選択したスク립トファイルの内容が記述されます。

3 - 3 - 2 - 5 LMWINの環境設定

次に環境設定メニューの「通信設定」サブメニューを選択します。
これは通信設定画面を表示するものです。

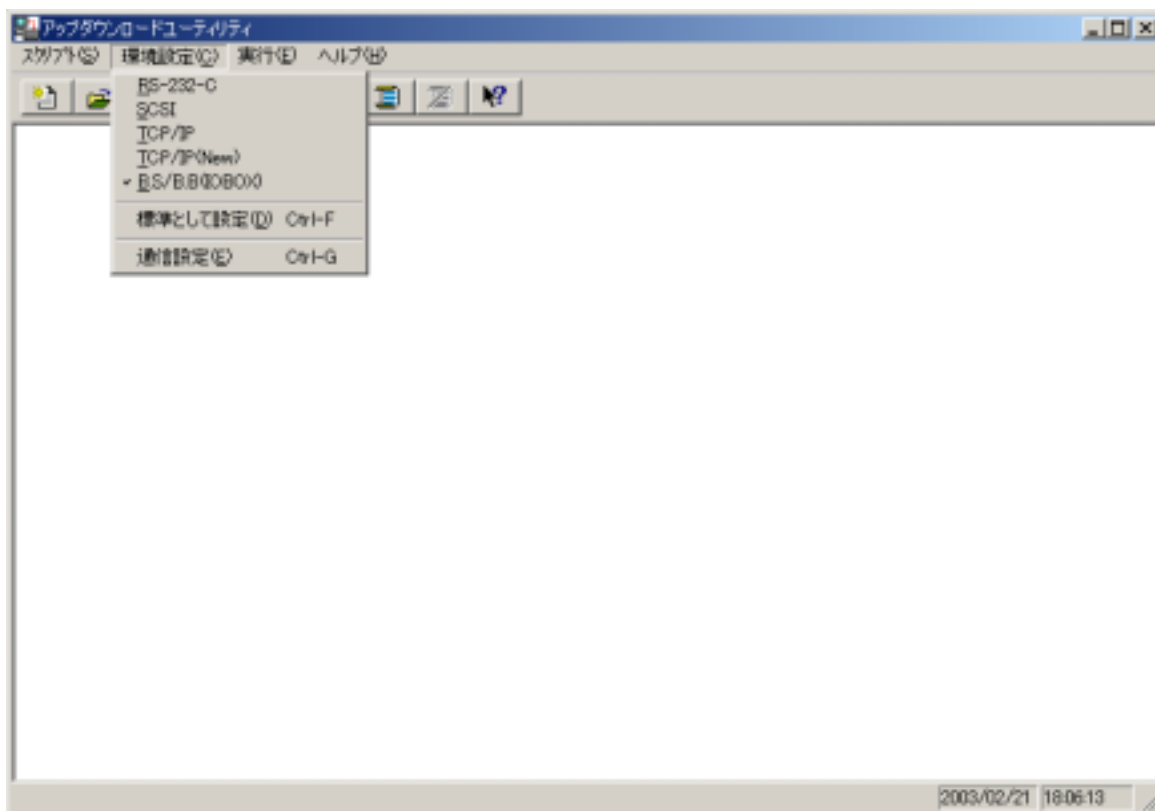
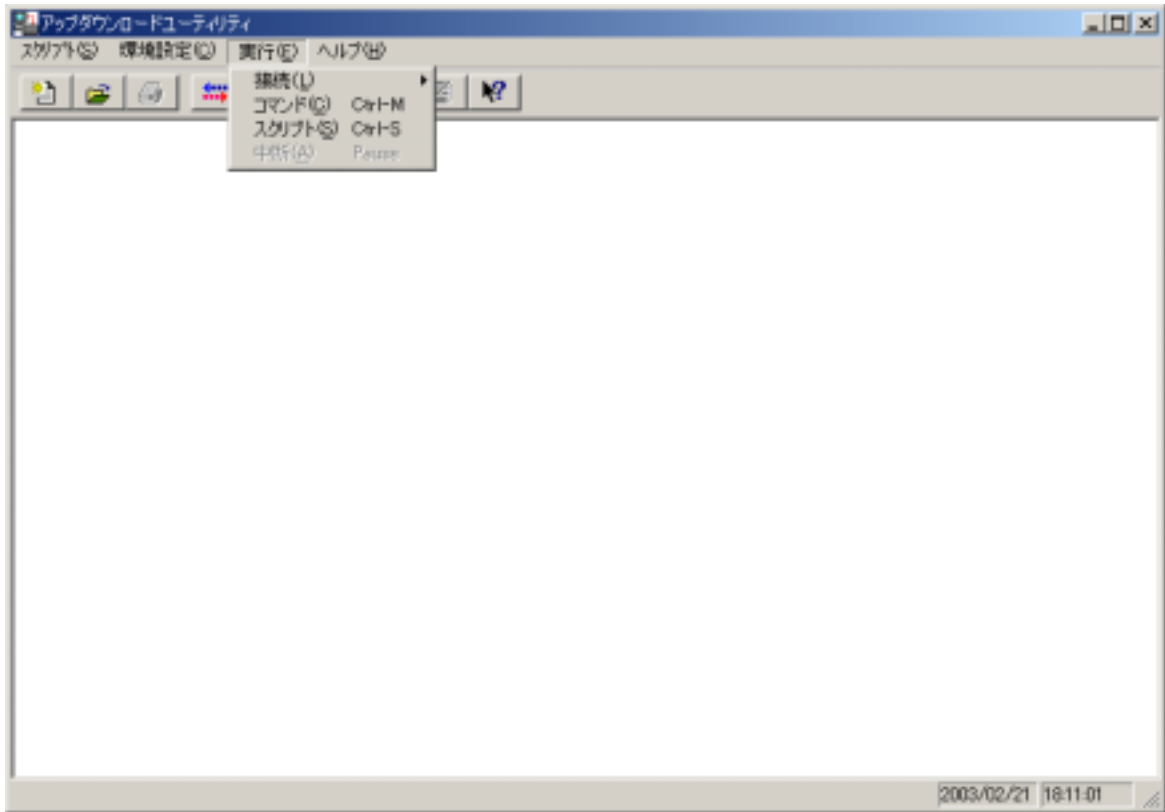


図3 - 2 2 LMWIN環境設定メニュー

DT-36410 で通信を行うには、<B.S./B.B.IOBXX>を選択して下さい。

3 - 3 - 2 - 6 LMWINの実行



3 - 3 - 2 - 6 - 1 スクリプト実行

スクリプト選択画面では1種類のスクリプトを選択できます。
選択したスクリプトを実行します。



図3 - 27 LMWINサーバーモード実行スクリプト選択画面

3 - 3 - 2 - 7 中断

中断サブメニューは選択すると通信を中断し、スクリプトファイルの作成 / 修正と環境設定を行なうことが出来るようになります。

通信実行中は「環境設定」メニューは使用できません。
その為、通信中は通信設定を行うことができません。また、スクリプトメニューの「新規作成」サブメニューも使用できません。さら通信中はスクリプトファイルの作成と編集はできません。ご注意ください。

3 - 3 - 2 - 8 ブリッジサテライトIO - BOXの設置

サテライトIO - BOXのインターフェースは、次のようになっています。次に進む前に、ケーブルはきちんと接続されているかどうか、I/Oボックスには電源を入れてあるか、確認してください。

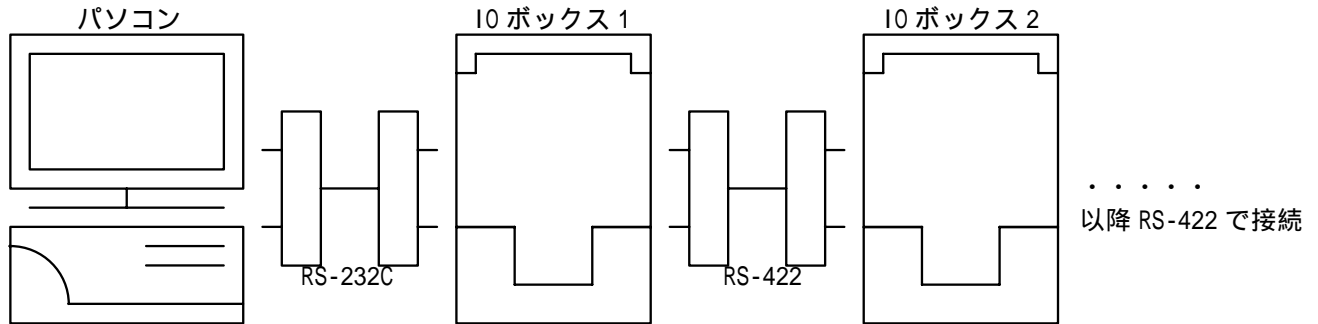


図3 - 28 パソコンとサテライトIOボックスの接続

10ボックス最大接続台数	ハンディ本体最大認識数
8台	8種類

表3 - 6 サテライトIOボックスの最大接続台数と最大認識数

3 - 3 - 2 - 9 DT - 300操作

パソコンのLMWINとDT - 300で通信をするには、パソコン側を「3 - 3 - 2 - 6 - 1 スクリプト実行」で通信待機にしておき、DT - 300をI/Oボックスに乗せて、次の手順でDT - 300本体側の準備を行い、通信を開始します。

- a . 「F」キーと「決定」キーを押しながら電源をオンします（システムニュー起動）。
- b . 「4 : テンソウ」を選択します。
- c . 「4 : プロトコル」を選択します。
「1 : FLINK」を選択して「決定」キーにて確定します。
- d . 「3 : ユーティリティ」を選択します。
「1 : ファイルテンソウ」を選択します。

以上で通信待機状態になります。

DT - 300がI/Oボックスに乗っていれば通信が開始されます。

ダウンロード完了後は「クリア」キーを押して、「システムメニュー」のトップページに戻り、「1 : APキドウ」を行うと、アプリケーションプログラムが起動します。

パソコン側は、スクリプトではなく「実行」メニューから「コマンド」を選び、コマンドメニューの“send”を選んで実行しても構いません。

4 . アプリケーション開発上の留意事項

コントラストの調整について

画面のコントラスト調整及び、バックライトのオン/オフは、初期状態においては次の操作で調整することが可能です。

- ・ (F1) : コントラストを1段階濃くする
- ・ (F4) : コントラストを1段階薄くする
- ・ バックライト (F2) : バックライトのオン/オフ

コントラスト調整は、環境の変化や個体のばらつきにより必要になる場合があります。ファンクションキーの設定を変更する場合も、アプリケーションでコントラスト調整が可能になるようにしてください。

LB0の通知機能について

主電池 (乾電池) の電圧低下状態には、LB1 (警告レベル) とLB0 (強制電源断) があります。

LB0発生時には、システム側で強制的に電源断処理を行います。従って、通知機能によるフラグをアプリケーションが認識するのは、次回の電源オン時です。

一般的に、LB0からの復旧状態では、電池交換等により電源が確保されているわけですから、LB0発生時に通知機能を使い電源断処理などを行うと、電池交換しても一回は電源が切れてしまう、という動きになります。操作上、混乱しないようにして下さい。

一般的にLB0の通知機能が必要となるのは、単なるレジューム処理では現状を復帰できない場合や、復旧する必要のない場合です。たとえば、通信中のLB0による電源断の場合などです。

Bドライブ (FROM) 使用上の注意点

Bドライブ (FROM) には書込み制限がありますので、頻繁に書込み処理を行うファイルを格納することは避け、プログラムファイルやマスタファイルなど読み込み専用のファイルを格納するようにしてください。

5 . Q & A 集

Q1 : `fread`関数を使った時に異常終了となった場合、`fclose`関数にて異常終了したファイルをクローズすると、異常終了してしまうのですが？

A1 : Cライブラリの仕様です。

`fclose`関数は、エラーフラグの立っているファイルはエラーと判断し、クローズしません。

このような場合の異常終了を回避するには、`fread`関数の後に次のような処理を行ってください。

```
ret = fread( buf, size, n, fp );
if( ferror( fp ) )
{
```

ここにエラー時の処理を記述してください

```
clearerr( fp );    エラーフラグをクリアします
}
```

Q2 : `lcd_gaiji`関数にて外字登録を行うと、その外字はどの領域に登録されるのですか？

A2 : ファイルデータを外字フォントエリアにメモリ展開します。

このエリアは、あらかじめ確保されているエリアなので、ワークエリアを圧迫することはありません。

Q3 : `KJ_CNVRT`で、2バイトコードとエスケープ文字列を同一文字列に入れて変換すると、エスケープ文字列も変換されてしまうのですが？

A3 : 制約事項です。

文字列内に2バイトコードがあった場合、それ以降の文字を全て変換しているためです。同一文字列内で2バイトコードとエスケープ文字列を混在して使用しないでください。

例)

変換前

```
char dsp_data[ ]="漢字¥n";
```

変換後

```
char dsp_data[ ]="¥x8a¥xbf¥x8e¥x9a¥x5c¥x6e";
```

Q4 : キー入力関数で、リターン条件に「LB発生終了」を設定しておけばLB通知モードを設定していなくても入力処理を脱出することができるのですか？

(`key_read`, `key_num`, `key_string`)

A4 : 脱出することはできません。

LB通知モードを設定し、さらにキー入力関数のリターン条件に「LB発生終了」を設定しなければなりません。

Q5 : コントラスト調整や、バックライトのON/OFFをプログラムで実行することは

できますか？

A5：コントラスト調整は `dat_system` 関数で、バックライトの ON/OFF は `lcd_el` 関数で実行できます。

Q6：通知モードのフラグ状態取得関数「`flg_sts`」の第1パラメータである「`p_flgpid`」は何のために使用するのですか？

A6：将来の拡張用ですので現行の DT-300 では使用できません。

Q7：`key_num` 関数を反転表示モードで呼び出し、キー入力後に脱出した後も入力エリアは反転したままですが、ノーマル表示にする方法はありませんか？

A7：ありません。

ノーマル表示モードで表示し直してください。

Q8：読み取り可能バーコードで、WPC (UPC - E 以外) にチェックデジットの実行有りを指定した場合、UPC - B コードは読み取り可能ですか？

A8：読めません。

UPC - B コードを読み取りたい場合は、チェックデジット実行無しに設定してください。

Q9：メモリ領域のサイズを変更した場合、ソフトリセット或いは、ハードリセットによってその値はデフォルトに戻るのですか？

A9：ハードリセットを行ってもデフォルト値に戻る事はありません。

但し、主電池及び副電池がローバッテリーでメモリを保持できなくなった場合には、デフォルトに戻ります。

Q10：システムメニューで設定値を変更後、アプリケーションすると設定値が変わってしまっているようです。なぜでしょうか？

A10：アプリケーション起動時には、「`CONFIG.HTS`」の設定値が有効となります。システムメニューで設定した値をアプリケーションで使用したい場合には「`CONFIG.HTS`」はダウンロードしないで下さい。

Q11：長期保管により、主電池、副電池がなくなってしまうメモリが初期化されてしまいました。メモリが初期化されてもアプリケーションが消えないようにすることはできませんか？

A11：プログラムファイルや環境ファイルを RAM (Aドライブ) ではなく FROM (Bドライブ) に格納しておくことにより、メモリが初期化された場合でもアプリケーションを起動することが可能です。

DT - 300 開発ガイド
平成15年5月 Rev 1.02 発行
カシオ計算機株式会社