

Hシリーズ
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添付資料

目次

1	リンケージエディタ	1
1.1	オブジェクトフォーマット.....	1
1.2	START オプション / サブコマンドの機能拡張.....	3
1.3	シンボルアドレス出力機能.....	4
1.4	中間ファイルのディレクトリ指定	6
1.5	仕様変更	6
1.6	エラーメッセージ	7
2	ライブラリアン	9
2.1	オブジェクトフォーマット.....	9
3	オブジェクトコンバータ	10
3.1	オブジェクトフォーマット.....	10
3.2	コンバートファイルの分割出力	10
3.3	オプションの追加	11
3.4	エラーメッセージ	12

ELF/DWARF は、the Tool Interface Standards Committee で開発された、オブジェクトフォーマットの名称です。

本添付資料は、以下のソフトウェアで追加、変更した機能の使用方法を述べたものです。

- ・H シリーズ リンケージエディタ Ver.6.0
- ・H シリーズ ライブラリアン Ver.2.0
- ・H シリーズ オブジェクトコンバータ Ver.2.0

1 リンケージエディタ

本章では、H シリーズ リンケージエディタ Ver.6.0 の追加機能を説明します。

新規に追加した機能の一覧を表 1-1 に示します。

表 1-1 新規機能一覧

No.	項 目	オプション/サブコマンド 環境変数	機 能
1	オブジェクトフォーマット (ELF/DWARF のサポート)	ELF	ELF/DWARF オブジェクトフォーマットを出力
		SYSROF	SYSROF オブジェクトフォーマットを出力 (従来オブジェクトフォーマットと互換)
		SYSROFPLUS	SYSROF/DWARF オブジェクトフォーマットを出力
2	START オプション/サブコマンド の機能拡張	START	複数セクションの同一アドレス割り付け機能
3	シンボルアドレス出力機能	FSYMBOL	解決済み外部定義シンボルをファイルに出力
4	中間ファイルのディレクトリ指定	HLNK_TMP	中間ファイル出力先のディレクトリ指定機能
5	その他	ENTRY	無効時にウォーニングメッセージ 125 出力

1.1 オブジェクトフォーマット

オブジェクト/デバッグ情報フォーマットとして、ELF/DWARF を指定できるようになりました。

(1) 指定方法

オブジェクトフォーマット出力の選択は、オプション/サブコマンドで指定します。
各オブジェクトフォーマットに対するオプション/サブコマンドは以下のとおりです。

オプション/サブコマンド名	オブジェクトフォーマット
ELF	ELF/DWARF
SYSROF	SYSROF (従来オブジェクトフォーマットと互換)
SYSROFPLUS	SYSROF/DWARF

オブジェクトフォーマットに関するオプションの指定を省略した場合、SYSROF のオブジェクトファイルを出力します。

デバッグ情報出力オプション / サブコマンド (DEBUG/SDEBUG) との組み合わせで、5 種類のオブジェクトフォーマットを選択できます。使用するデバッガに合わせて選択してください。

表 1-2 使用可能なデバッガとオプション / サブコマンドの関係

使用可能なデバッガ	オプション / サブコマンド	
	オブジェクトフォーマット	デバッグ情報出力
ELF/DWARF 形式のデバッガ	ELF *1	DEBUG
日立統合化マネージャ (Ver.4) + E8000	ELF *1	SDEBUG
日立統合化マネージャ (Ver.4) + E7000	SYSROFPLUS *1	SDEBUG
日立統合化マネージャ (Ver.3) + E7000	SYSROF	SDEBUG
日立デバッグインタフェース (Ver.2) + E6000	SYSROF	DEBUG

*1 : ELF および SYSROFPLUS オプションをデバッグ情報出力指定 (debug/sdebug) と同時に指定する場合、対応可能なコンパイラおよびアセンブラは次のとおりです。

製品名	対応バージョン
SuperH RISC engine ファミリー C/C++コンパイラ	5.0 以降
SuperH RISC engine ファミリー クロスアセンブラ	4.0 以降
H8S, H8/300 シリーズ C/C++コンパイラ	3.0 以降
H8S, H8/300 シリーズ クロスアセンブラ	3.0 以降

【注意】ELF または SYSROFPLUS 対応のコンパイラ、アセンブラで生成したオブジェクトプログラム、ライブラリと旧バージョンで生成したオブジェクトプログラム、ライブラリは、混在してリンクできます。ただし、リンク時に ELF または SYSROFPLUS オプションを指定した場合、旧バージョンで生成したオブジェクトプログラム、ライブラリのデバッグ情報は削除します。

(2) ELF オプション

本オプションは、ELF/DWARF のオブジェクトフォーマットを出力する場合に指定します。デバッグ環境が日立統合化マネージャの場合、リンク時に SDEBUG オプションを指定してください。

例

```
shc    test1.c      -debug
shc    test2.c      -debug
asmsh  test3.src    -debug
lnk     test1,test2,test3  -elf  -debug
```

(3) SYSROF オプション

本オプションは、SYSROF のオブジェクトフォーマットを出力する場合に指定します。

例

```
shc      test1.c      -debug
shc      test2.c      -debug
asmsh    test3.src    -debug
lnk      test1,test2,test3  -sysrof      -debug
```

(4) SYSROFPLUS オプション

本オプションは、SYSROF/DWARF のオブジェクトフォーマットを出力する場合に指定します。デバッグ情報出力指定では、SDEBUG オプションを指定してください。

例

```
shc      test1.c      -debug
shc      test2.c      -debug
asmsh    test3.src    -debug
lnk      test1,test2,test3  -sysrofplus  -sdebug
```

1.2 START オプション / サブコマンドの機能拡張

複数のセクションを同一アドレスに割り付けることができるようになりました。

(1) 指定方法

同一アドレスへの複数のセクションの割り付けは、START オプション / サブコマンドで同一アドレスに割り付けるセクション群をコロン (:) で区切って指定します。

オプション	PC 版	- <u>START</u> =<セクション名>[,<セクション名>...] [:<セクション名>[,<セクション名>...]...](<先頭アドレス>)
	UNIX 版	- <u>START</u> =<セクション名>[,<セクション名>...] [:<セクション名>[,<セクション名>...]...]/<先頭アドレス>
サブコマンド	<u>START</u> <セクション名>[,<セクション名>...] [:<セクション名>[,<セクション名>...]...](<先頭アドレス>)	

名称の下線部は最も短い短縮指定を示します。

(2) 使用上の制限

・ページタイプのモジュール (H8/500 シリーズ) では、複数のセクションを同一アドレスに割り付ける機能は使用できません。

・複数セクションを同一アドレスに割り付けた場合、セクション間でのシンボル参照はできません。（図1のRAM_sct1からRAM_sct2のシンボルを参照するケース）

（3）指定例

本機能を用いて、同時に存在しない複数のプログラム／データを外部ROMから高速な内部RAMに転送して実行する例を示します。

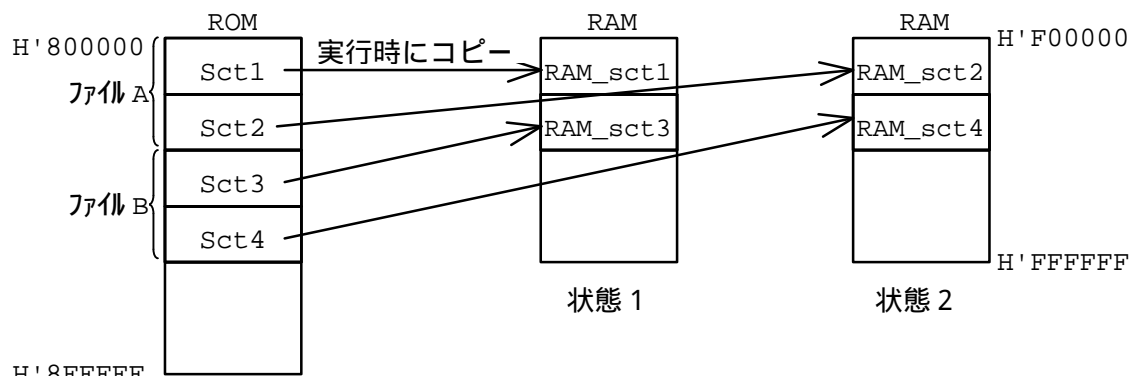


図1 同一アドレスへの複数セクション割り付け

【コマンド指定例】

```
lnk -subcommand=test.sub
```

【test.sub の内容】

```
INPUT  A,B
ROM    (Sct1, RAM_sct1), (Sct3, RAM_sct3)
ROM    (Sct2, RAM_sct2), (Sct4, RAM_sct4)
START  Sct1, Sct2, Sct3, Sct4(800000)
START  RAM_sct1, RAM_sct3:RAM_sct2, RAM_sct4(0F00000)
```

【説明】

RAM_sct1 と RAM_sct2 を同一アドレスから割り付けます。RAM_sct3 は RAM_sct1 に、RAM_sct4 は RAM_sct2 に各々連結して割り付けます。

1.3 シンボルアドレス出力機能

シンボルアドレス出力機能は、リンカージェディタで解決した外部定義シンボルをアセンブラ制御命令の形式でファイルに出力します。出力ファイルをアセンブル、リンクすることにより、当該シンボル定義を含むオブジェクトプログラムをリンクすることなく、外部参照シンボルのアドレスを解決することができます。

(1) 指定方法

シンボルアドレス出力は、FSYMBOL オプション / サブコマンドで対象となる外部定義シンボルを含むセクション名を指定します。

オプション	-FSYMBOL=<セクション名>[,<セクション名>...]
サブコマンド	FSYMBOL <セクション名>[,<セクション名>...]

名称の下線部は最も短い短縮指定を示します。

(2) 機能説明

・本オプションは、リンケージエディタで解決した外部定義シンボルをアセンブラ制御命令の形式でファイルに出力します。出力するファイル名は、ロードモジュール名にファイル拡張子".sym"を付加したファイルとなります。

(3) 使用上の制限

- ・出力するロードモジュールファイルがリロケータブル形式の場合、本オプション / サブコマンドは使用できません。
- ・外部定義シンボル名の長さが 238 文字を超えるシンボルは、ウォーニングメッセージ 126 を出力し、238 文字までが有効になります。
- ・指定したセクション名が存在しない場合、ウォーニングメッセージ 127 を表示し、処理を継続します。指定したセクションが全て存在しない場合、ファイルは出力されません。

(4) 使用例

図 2 は、製品 A の機能 A を機能 B に変更し、製品 B を開発する例です。本機能を用いて、共通 ROM 内シンボルのアドレスを解決することにより、共通 ROM が流用できます。

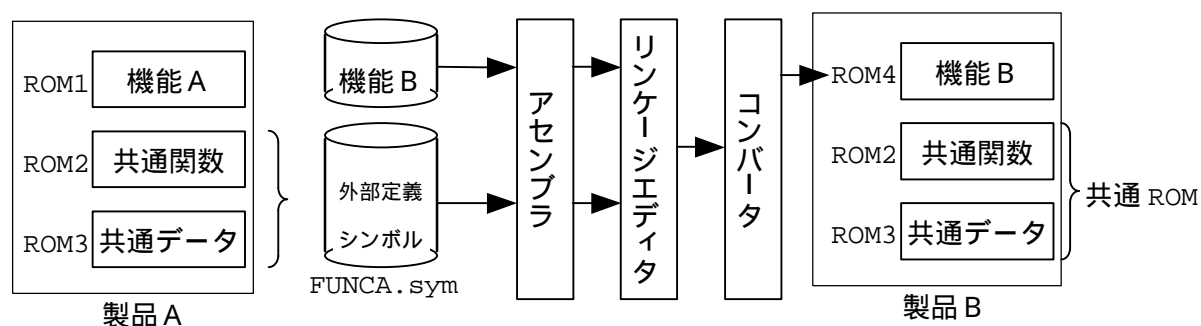


図 2 シンボルアドレス出力機能の使用例

【外部定義シンボルファイル出力の指定例】

```
lnk      ROM1,ROM2,ROM3      -output=FUNCA -fsymbol=sct2,sct3
```

sct2 と sct3 の外部定義シンボルをファイルに出力します。

【ファイル(FUNCA.sym)の出力例】

```
;H SERIES LINKAGE EDITOR GENERATED FILE    1997.10.10
;fsymbol = sct2, sct3

;SECTION NAME = sct1
.export sym1
sym1: .equ    h'00FF0080
.export sym2
sym2: .equ    h'00FF0100
;SECTION NAME = sct2
.export sym3
sym3: .equ    h'00FF0180

.end
```

【アセンブル、再リンクの指定例】

```
asmsh    ROM4
asmsh    FUNCA.sym
lnk      ROM4,FUNCA
```

ROM2,ROM3 のオブジェクトファイルをリンクすることなく、ROM4 の外部参照シンボルを解決します。

【注意】

本機能を使用する場合、共通関数から機能A内シンボルは参照できません。

1.4 中間ファイルのディレクトリ指定

環境変数により、中間ファイルの出力先のディレクトリを指定できるようになりました。

中間ファイルのディレクトリ指定は、HLNK_TMP 環境変数で指定します。

MS-DOS 版	set HLNK_TMP=<ディレクトリ名>
UNIX 版	setenv HLNK_TMP <ディレクトリ名>

1.5 仕様変更

1.5.1 ENTRY オプションの仕様変更

出力するロードモジュールがリロケータブル形式かつ実行開始アドレス（エントリポイント）が定数値の場合、ウォーニングメッセージ 125 を出力し、指定を無効とします。

1.5.2 リンケージリストの仕様変更

リンケージリストの出力フォーマットを 132 カラム / 行から 80 カラム / 行に変更しました。

1.5.3 入力ファイル数の制限拡張

リンケージエディタが 1 度のリンケージ処理で扱える入力ファイル数を 256 個から 65,535 個に拡張しました。

1.6 エラーメッセージ

表 1-3 に Ver.6.0 で追加したエラーメッセージを示します。

表 1-3 エラーメッセージ一覧(1)

エラー番号	内容
125	CANNOT SET ENTRY POINT
	内容:出力ロードモジュールがリロケータブル形式に対して、実行開始アドレスに定数の外部参照シンボルを指定した。 対策:出力ロードモジュールをアブソリュート形式にするか、実行開始アドレスの指定を削除してください。
126	TOO LONG CHARACTER NUMBER (FSYMBOL)
	内容:FSYMBOL オプション / サブコマンドで指定したセクション内のシンボルの文字数が 238 文字を超えた。 対策:該当するシンボルの文字数を 238 文字以下に変更してください。
127	EXTERNAL SYMBOL 0 (セクション名)
	内容:指定したセクション内に外部定義シンボルが存在しない。 対策:セクション名を確認してください。
128	ILLEGAL SYMBOL REFERENCE (シンボル名)
	内容:同一アドレスに割り付けたセクション間でシンボルの参照があった。 対策:同一アドレスに割り付けたセクション間でシンボルを参照しないようプログラムを変更してください。
333	CANNOT OPEN INTERNAL FILE
	内容:中間ファイルをオープンできなかった。 対策:環境変数 HLNK_TMP で指定したディレクトリ名を確認してください。 ディレクトリ名が正しい場合はディスク容量に空きがないか、またはディスクにハード的なエラーがある場合があります。確認の上、再実行してください。

表 1-3 エラーメッセージ一覧表(2)

334	CANNOT CLOSE INTERNAL FILE
	<p>内容:中間ファイルがクローズできなかった。</p> <p>対策:環境変数 HLNK_TMP で指定したディレクトリ名を確認してください。ディレクトリ名が正しい場合はディスク容量に空きがないか、またはディスクにハード的なエラーがある場合があります。確認の上、再実行してください。</p>
335	CANNOT DELETE INTERNAL FILE
	<p>内容:中間ファイルが削除できなかった。</p> <p>対策:環境変数 HLNK_TMP で指定したディレクトリ名を確認してください。ディレクトリ名が正しい場合はディスク容量に空きがないか、またはディスクにハード的なエラーがある場合があります。確認の上、再実行してください。</p>
336	CANNOT OUTPUT INTERNAL FILE
	<p>内容:中間ファイルに出力できなかった。</p> <p>対策:環境変数 HLNK_TMP で指定したディレクトリ名を確認してください。ディレクトリ名が正しい場合はディスク容量に空きがないか、またはディスクにハード的なエラーがある場合があります。確認の上、再実行してください。</p>
337	CANNOT READ INTERNAL FILE
	<p>内容:中間ファイルから入力できなかった。</p> <p>対策:環境変数 HLNK_TMP で指定したディレクトリ名を確認してください。ディレクトリ名が正しい場合はディスク容量に空きがないか、またはディスクにハード的なエラーがある場合があります。確認の上、再実行してください。</p>
338	DUPLICATE START ADDRESS SPECIFIED IN PAGE TYPE
	<p>内容:ページタイプの入力ファイルに対して、複数セクションの同一アドレス割り付けを指定した。</p> <p>対策:ページタイプは、複数セクションの同一アドレス割り付けができません。</p>

2 ライブラリアン

2.1 オブジェクトフォーマット

オブジェクトフォーマットとして、ELF/DWARF または SYSROF/DWARF を使用する場合は、H シリーズ ライブラリアン Ver.2.0 以降を使用してください。

オブジェクトフォーマットおよび対応するコンパイラ、アセンブラについては、1.1 オブジェクトフォーマットを参照してください。

3 オブジェクトコンバータ

本章では、H シリーズ オブジェクトコンバータ Ver.2.0 の追加機能を説明します。
新規に追加した機能の一覧を表 3-1 に示します。

表 3-1 新規機能一覧

No.	項 目	オ プ シ ョ ン	機 能
1	オブジェクトフォーマット	-	ELF/DWARF オブジェクトフォーマットをサポート
2	コンバートファイルの分割出力	-	任意のアドレス範囲毎にファイルを分割出力
3	出力レコードの統一	RECORD	一定のフォーマット (S1, S2, S3) で出力
4	S9 レコードの出力	S9	終端に S9 レコードを出力

3.1 オブジェクトフォーマット

リンカージェディタで ELF オプション / サブコマンドを指定した場合は、H シリーズ オブジェクトコンバータ Ver.2.0 以降を使用してください。

3.2 コンバートファイルの分割出力

S タイプに変換したオブジェクトプログラムを任意のアドレス範囲に分割してファイル出力できるようになりました。

(1) 指定方法

コンバートファイルの分割出力は、出力ファイル名とアドレス範囲を空白で区切って指定します。

分割しない場合	cnvs 入力ファイル名 [出力ファイル]
分割する場合	cnvs 入力ファイル名 出力ファイル=開始アドレス, 終了アドレス [出力ファイル=開始アドレス, 終了アドレス...]

(2) 機能説明

- ・出力ファイル名の後に開始アドレスと終了アドレスを指定すると、指定範囲内のオブジェクトを S タイプオブジェクトにコンバートします。
- ・開始アドレスおよび終了アドレスは、16 進数定数で指定します。
- ・ページタイプの場合、開始アドレスと終了アドレスの前にページアドレスをコロン (:) で区切って指定します。

(3) 使用上の制限

- ・ 開始アドレスおよび終了アドレスに負の値を指定したとき、開始アドレスより終了アドレスが小さい場合、エラーメッセージ 311 を出力します。
- ・ 指定したアドレス範囲にオブジェクトが存在しない場合、ウォーニングメッセージ 101 を出力します。

(4) 指定例

- ・ 非アドレスタイプの場合 (H8S シリーズ, H8/300 シリーズ, SuperH RISC engine ファミリ)

```
cnvs test.abs test1.mot=0,FFFF test2.mot=10000,1FFFF
```

- ・ アドレスタイプの場合 (H8/500 シリーズ)

```
cnvs test.abs test1.mot=0:0,0:FFFF test2.mot=1:0,1:FFFF
```

3.3 オプションの追加

H シリーズ オブジェクトコンバータ Ver.2.0 では、以下オプションが追加になりました。

- ・ RECORD オプション
- ・ S9 オプション

(1) コマンドフォーマット

オプションを指定する場合、コマンドフォーマットは、次のとおりです。

```
cnvs <入力ファイル名>[ <出力ファイル名>[ <出力ファイル名>... ]][[ ]-<オプション名>...]
```

(2) RECORD オプション

本オプションは、ロードアドレスに関係なく、一定のデータレコード (S1, S2, S3) で出力します。指定したデータレコードより大きいロードアドレスが存在した場合、エラーメッセージ 310 を表示し、処理を終了します。

例

```
cnvs test.abs test.mot -record=s2
```

全てのデータレコードを S2 で出力します。

(3) S9 オプション

ROM ライタには、エンドレコードが S9 でなければ正常に書込みのできない製品があります。本オプションは、エントリアドレスが H'10000 を超える場合でも S9 レコードを終端に出力します。この場合、S9 レコードのエントリアドレスは、H'0 となります。

例

```
cnvs      test.abs      test.mot      -s9
```

終端に S9 レコードを出力します。

補足)S タイプオブジェクト形式は、「H シリーズ 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」のオブジェクトコンバータ編を参照してください。

3.4 エラーメッセージ

表 3-2 に Ver.2.0 で追加したエラーメッセージを示します。

表 3-2 エラーメッセージ一覧表

エラー番号	内容
101	NO OBJECT IN SPECIFIED ADDRESS RANGE
	内容：指定したアドレスの範囲内にオブジェクトが存在しない。 対策：開始アドレスおよび終了アドレスを確認してください。
310	LOAD ADDRESS OVERFLOW
	内容：ロードアドレスが RECORD オプションで指定したデータレコードを超えた。 対策：RECORD オプションのデータレコードを変更するか、RECORD オプションを削除してください。
311	ILLEGAL ADDRESS RANGE SPECIFIED
	内容：開始アドレスまたは終了アドレスの指定が正しくない。 対策：開始アドレスまたは終了アドレスを訂正してください。