

SUPER FAMICOM DEBUGGER
S F D
ユーザーズ・マニュアル

1991年 9月 作成

株式会社リコー
電子デバイス事業部

- MS-DOSは、Microsoft Corp.の商標です.
- DOS/16Mは、Rational System, Inc.の商標です.
- 本製品は、Microsoft C Compilerを使用しています.
- 本製品は、DOS/16Mを使用しています.

目 次

第1章	はじめに	1
第2章	動作環境	2
2.1	S F Dの動作環境	2
2.2	ハードウェア設定	3
2.3	S F Dの使用するメモリについて	3
2.4	S F Dのインストール	3
第3章	使用方法	5
3.1	S F Dで扱うファイル	5
3.2	S F Dの起動と起動オプション	6
3.3	画面構成	8
3.4	データ入力の基数	9
3.5	アドレスの表記	9
3.6	パラメータの演算	10
3.7	S F Dで扱うシンボル	10
3.7.1	グローバルシンボル	10
3.7.2	ローカルシンボル	10
3.7.3	行番号シンボル	11
3.7.4	ユーザー定義シンボル	11
3.8	マップファイル、シンボルファイルの作成	12
第4章	ウィンドウコマンド	13
4.1	ウィンドウ	13
	コードウィンドウ	14
	コマンドウィンドウ	15
	レジスタウィンドウ	16
	ステータスウィンドウ	17
	ウォッチウィンドウ	18
	ダイアログボックス	19
4.2	コントロールキー	20
4.2.1	画面制御	20
4.2.2	行編集制御	20
4.2.3	コマンド実行制御	20

4.3	ファンクションキー	22
	f・1 File	22
	f・2 Regs	22
	f・3 Watch	22
	f・4 Code	22
	f・5 Go	22
	f・6 Cursor	22
	f・7 Come	23
	f・8 Trace	23
	f・9 Break	23
	f・10 Step	23
	SHIFT+f・1 Find	23
	SHIFT+f・2 Next	23
	SHIFT+f・3 Config	24
	SHIFT+f・4 B.Comp	25
	SHIFT+f・5 PC set	31
	SHIFT+f・6 DMA	31
	SHIFT+f・7 PPU	33
	SHIFT+f・8 RTTStt	34
	SHIFT+f・9 RTTDsp	35
	CTRL+f・1 Main	37
	CTRL+f・2 CG-RAM	39
	CTRL+f・3 OAM	40
	CTRL+f・4 VRAM	40

第5章 ダイアログコマンド

4 1

5.1	ダイアログコマンドについて	41
5.2	コマンド入力書式	41
5.3	ダイアログコマンド	41
	A (アセンブル)	42
	A D R (アドレス表示書式変更)	43
	B (ブレークスイッチ)	44
	B C (バスサイクルコンパレータ)	45
	B S (シンプルバスサイクルブレーク)	47
	C (割り込みコントロール)	48
	C M D (コマンドファイル作成)	50
	C P (画面コピー)	51
	C S (チェックサム)	52

D	(メモリ表示)	53
DG/DO/DV	(指定メモリ表示)	54
DMA	(DMAレジスタ)	55
ER	(エミュレーションCPUリセット)	56
EX	(コマンドファイル実行)	57
F	(メモリフィル)	58
G	(プログラム実行)	59
H	(データの演算)	60
L	(逆アセンブル表示)	61
LOG	(ログファイル作成)	62
M	(メモリ転送)	63
MM	(メモリマップ)	64
MS	(メモリ検索)	65
MV	(メモリベリファイ)	66
N	(データ入力基数の変更)	67
PI	(I/Oポート入力)	68
PO	(I/Oポート出力)	69
PPU	(PPUレジスタ)	70
Q	(SFD終了)	71
RD	(ファイルリード)	72
S	(メモリ変更)	73
SG/SO/SV	(指定メモリの変更)	74
ST	(ICEステータス)	75
TIM	(インターバルタイマ)	76
TM	(リアルタイムトレース)	77
TG	(ターゲット)	78
T	(トレース/ステップ実行)	79
U	(トレース/ステップ実行)	80
VER	(バージョン番号の表示)	81
VMM	(ROMエリアのメモリマッピング)	82
W	(ウォッチ)	83
WT	(ファイルライト)	84
X	(レジスタ)	85
Y	(シンボル表示)	87
#	(タブ)	89
≡	(MS-DOSコマンド実行)	90
!	(ヒストリ)	91

6.1	マクロ機能	92
6.2	マクロコマンドの実行	92
6.3	マクロコマンドの登録	92
	MACRO (マクロ登録)	93
	MLIST (マクロ表示)	94
	KILL (マクロ削除)	95
6.4	マクロ制御文と式評価	96
	REPEAT {} (繰り返し)	97
	WHILE {} (条件付き繰り返し)	98
	DO {} WHILE (条件付き繰り返し)	99
	IF {} (条件制御)	100
	BREAK (ループからの抜け出し)	101
付録A	アセンブラの書式	102
	A.1 ニーモニック	102
	A.2 アドレッシングモード	103
	A.3 擬似命令	103

第 1 章 はじめに

SFDは、スーパーファミコン用ソフトウェアの開発デバッグツールです。SFBOX-IIと接続することでデバッグに必要な様々な機能を使うことができます。

SFDは、ウィンドウ指向のデバッガで、ソースコード、逆アセンブルコード、レジスタの内容を見ながらデバッグすることができます。また、サブウィンドウを開き面倒な設定が必要な機能も、簡単なキー操作で設定できます。

第 2 章 動作環境

2. 1 SFDの動作環境

ホストコンピュータ

NEC製	PC-9801シリーズ
EPSON製	PC-286/386シリーズ
CPU:	80286/80386
メモリ:	MB以上
モード:	ノーマルモード
O S:	MS-DOS Version 3.30 以降

インサーキットエミュレータ	SFBOX-II
---------------	----------

GP-IBインターフェースボード	EP-98GPA
------------------	----------

2. 2 ハードウェア設定

S F B O X - II

S F B O X - II 背面のファンクションスイッチを、次のように設定して下さい。

ファンクションスイッチ	No.1	No.2	No.3	No.4	No.5	No.6	No.7	No.8
	ON	ON	ON	ON	OFF	ON	ON	ON

G P - I B インターフェースボード

G P - I B インターフェースボードには、ディップスイッチやジャンパプラグがあります。これらを次のように設定して下さい。

ディップスイッチ1	No.1	No.2	No.3	No.4	No.5	No.6	No.7	No.8
(SW1)	ON	ON	ON	ON	ON	ON	ON	ON

ディップスイッチ2	No.1	No.2	No.3	No.4	No.5	No.6
(SW2)	OFF	OFF	OFF	OFF	OFF	OFF

ジャンパプラグ (J1) 1 番ピンをショートします。

2. 3 S F D の使用するメモリについて

S F D は、コンピュータ本体のアドレス 1 0 0 0 0 0 番地から 2 F F F F F 番地のプロテクトモードのメモリを使用します。

仮想 8 6 モードとの共存は出来ません。但し、V C P I 対応の仮想 8 6 ドライバとの共存は出来ます。

2. 4 S F D のインストール

添付のフロッピーディスクに入っているファイルは、すべて同じディレクトリにコピーして下さい。また、AUTOEXEC.BAT に PATH と COMSPEC を指定して下さい。

例：ルートの下に実行用ディレクトリ "S F D" を作成し、フロッピーディスクに入っているファイルをコピーします。なお、実行用ディレクトリを作成するドライブを "A"、フロッピーディスクを挿入するドライブを "B" とします。

```
A:>MKDIR ¥SFD  
A:>COPY B:*. * SFD
```

AUTOEXEC.BAT に次の文を追加します。

PATH=A:¥SFD;

COMSPEC=A:¥COMMAND.COM

第 3 章 使用方法

3. 1 SFDで扱うファイル

SFDでは、以下のファイルを扱います。()内は、基本的なファイル名拡張子です。

ソースファイル	(.ASM)
ヘキサファイル	(.HEX)
マップファイル	(.MAP)
シンボルファイル	(.SYM)
バイナリファイル	(.BIN)
コマンドファイル	(.CMD)
ログファイル	(.LOG)
画面コピーファイル	

SFDは、ユーザが作成したソースファイル、ヘキサファイル、マップファイル、シンボルファイルを読み込みます。これらのファイルは、SFD起動時またはダイアログコマンドによって読み込む事が出来ます。

SFDは、メモリの内容をバイナリ形式で、ファイルを読み書きすることが出来ます。このファイルをバイナリファイルと呼びます。

コマンドファイルは、ダイアログコマンドの集まりで出来ています。あらかじめダイアログコマンドをファイルに登録しておけば、そのファイルを呼び出すことで一連のコマンドを実行することが出来ます。コマンドファイルは、SFD動作中に作成することも出来ます。

SFDは、入力されたダイアログコマンドや、そのコマンドの実行結果をファイルに書き込む機能があります。このファイルをログファイルと呼びます。

SFDは、現在の画面のイメージをファイルに出力することが出来ます。このファイルを、画面コピーファイルと呼び、画面コピーした日付、時刻を付けてテキスト形式で作成されます。但し、画面の表示属性はコピーされません。

3.2 SFDの起動と起動オプション

SFDの起動は、MS-DOSコマンドレベルで次のように入力します。

A>SFD [module] [option] 

モジュール名`module`を指定すると、ヘキサファイル、マップファイル、シンボルファイル、ソースファイルを読み込みます。

起動オプション`option`には、次のものがあります。システムに合わせて設定して下さい。システム既定値で良い場合は、指定しなくても構いません。

- `n` GP-IBデバイスアドレスの指定 (システム既定値: `n=0`)
`n`には、GP-IBのデバイスアドレスの番号を指定します。この値は、GP-IBインターフェースボードのディップスイッチ2の値と等しくして下さい。
- `Dn` DMAチャンネルの指定 (システム既定値: `n=`)
使用するDMAチャンネルに合わせて`n`を決めます。
`n=0`…チャンネル0、`1`…DMAを使わない、`2`…チャンネル3
- `In` 割り込みチャンネルの指定 (システム既定値: `n=`)
使用する割り込みチャンネルに合わせて`n`を決めます。
`n=0`…`INT0`、`1`…`INT2`、`2`…`INT5`、`3`…`INT6`
- `Pn` I/Oポートアドレスの指定 (システム既定値: `n=`)
使用するI/Oポートアドレスを指定します。`n`は、16進数2桁で指定します。
- `?` ヘルプ
SFD起動に関する簡単な説明が表示されます。このオプションが指定された場合、SFDは起動しません。

SFDの起動例を示します。

例) A>SFD SAMPLE -2 -D0 -I3 -PFF

モジュール名SAMPLE、GP-IBデバイスアドレスが2、DMAチャンネルはチャンネル0、割り込みチャンネルはINT6、I/OポートアドレスはFFHに設定し、SFDを起動します。

SFD起動時に、カレントディレクトリにファイル"SFD_INIT.CMD"があると、このファイルに書かれたダイアログコマンドを実行します。特に、SFD起動時の各種設定を記述しておくことで、SFD起動後すぐにデバッグ作業に入ることが出来ます。

3. 3 画面構成

ウォッチ		Interval: OFF Time= 3sec.		レジスタ	
⑤	1: 000000 EF			A= 00	D =0000
	2: 000001 20			B= 00	S =01FF
	3: 000002 28			C=0000	PC =8000
	4: 000003 AF			X=0000	DBR= 00
	コード [] ASM	SFD Version 1.00		Y=0000	PBR= 00 ③
	> 008000 0053	BRK	53	P : Env-bdIzc	
	008002 0000	BRK	00	Memory : 8-bit	
	008004 EB	XBA		Index : 8-bit	
	008005 A90000	LDA	#0000	ステータス	
①	008008 EE7D00	INC	!007D	T.Power : ON	
	00800B 00E4	BRK	E4	T.Clock : READY	
	00800D E90000	SBC	#0000	RTT : STOP	
	008010 7CF300	JMP	(!00F3,X)	H-DMA : OFF ④	
	008013 00B3	BRK	B3	V-BLANK : OFF	
	008015 F000	BEQ	008017	Port Wv : OFF	
	008017 0068	BRK	68	Mem Wv : OFF	
	008019 D400	PEI	00		
	00801B 003C	BRK	3C		
	コマンド	Timer:OFF	0s000m000.0u	Bcmp:OFF	STOP
②	>W2				
	>W3				
	>CP ABC				
⑥	File	Regs	Watch	Code	Go
				Cursor	Come
				Trace	Break
				Step	

①コードウィンドウ

ソースコード、逆アセンブルコードを表示します。

②コマンドウィンドウ

ダイアログコマンドの入力や、コマンドの実行結果を表示します。

③レジスタウィンドウ

CPUのレジスタの値を表示します。

④ステータスウィンドウ

ICEの状態を表示します。

⑤ウォッチウィンドウ

指定されたメモリの値を表示します。

⑥ファンクションキー表示ライン

ファンクションキーで実行するウィンドウコマンドの内容を表示します。

3.4 データ入力 of 基数

SFDでは、16進数、10進数、8進数、2進数を扱うことができ、数値の前に基数を示す記号を付けます。基数記号がない場合、原則的に16進数として扱います。しかし、コマンドパラメータによっては（個数入力等）、10進数入力の場合もありますので注意して下さい。

基 数	記号	扱える文字
16進数	\$	'0'～'9'、'A'～'F'/'a'～'f'
10進数	#	'0'～'9'
8進数	×	'0'～'7'
2進数	%	'0'～'1'

3.5 アドレス of 表記

アドレスの入力では、24ビットアドレスの他に、バンクアドレスとバンク内アドレスを（:）で分けて表現することが出来ます。

表 記	解 説
XXXXXX	24ビットアドレス
BB:XXXX	BB バンクアドレス XXXX バンク内アドレス
DBR:XXXX	DBR CPUのデータバンクレジスタの値 XXXX バンク内アドレス
PBR:XXXX	PBR CPUのプログラムバンクレジスタの値 XXXX バンク内アドレス
:XXXX	バンクアドレスは、直前のコマンドパラメータの バンクアドレスに同じ値となります。 XXXX バンク内アドレス

3. 6 パラメータの演算

コマンドパラメータの演算では、加減算のみ行えます。

+	加算
-	減算

3. 7 SFDで扱うシンボル

SFDでは、各種シンボルが扱えます。

1. グローバルシンボル
2. ローカルシンボル
3. 行番号シンボル
4. CPUモードシンボル
5. ユーザ定義シンボル

グローバルシンボル、ローカルシンボル、行番号シンボル、CPUモードシンボルは、ユーザが作成したシンボルファイルを読み込むことで、参照のみが行えます。

ユーザ定義シンボルは、SFD内でユーザにより定義されたシンボルで、参照および変更が可能です。

3. 7. 1 グローバルシンボル

グローバルシンボルは、ソースファイルでグローバル定義されているシンボルで、シンボルファイルが読み込まれているとき参照のみ行え、シンボル名の前にアンダーバー（`_`）を付けて指定します。

`_symbol`

3. 7. 2 ローカルシンボル

ローカルシンボルは、モジュール内でのみ有効なシンボルで、シンボル名とモジュール名で指定

します。シンボル名の前にはアンダーバー (_) を付け、モジュール名の前にはシャープ (#) を付けます。

`_symbol#module`

3. 7. 3 行番号シンボル

シンボルファイルに、行番号シンボルが付加されているときに有効で、モジュール内の行番号とそのモジュール名を指定します。行番号は10進数で指定し、番号の前にアンダーバー (_) を付けます。モジュール名の前には、シャープ (#) を付けます。

`_行番号#module`

3. 7. 4 CPUモードシンボル

シンボルファイルに、CPUモードシンボルが付加されているときに有効で、アセンブルまたは逆アセンブル表示のとき参照します。CPUモードシンボルの直接的な指定は出来ません。

3. 7. 5 ユーザ定義シンボル

ユーザ定義シンボルは、SFD内でユーザが自由に設定、変更が行えるシンボルです。

シンボルは、シンボル名の前にピリオド (.) を付けて指定します。

シンボル名は、最大8文字で英字の大文字と小文字を区別します。

`.symbol`

シンボルへの値の代入は、ダイアログコマンド入力時とA (アセンブル) コマンド実行時に行えます。

3. 8 マップファイル、シンボルファイルの作成

SFX用アセンブラシステムを使い、マップファイルとシンボルファイルは以下の方法で作成して下さい。これらのファイルが無い場合、ソースコードデバッグは出来ません。

マップファイルの作成は、「リンカ」のオプション"-ls"をつけ、ファイル名拡張子".MAP"として下さい。

```
XLINK -ls FILE.MAP
```

シンボルファイルの作成は、「オブジェクトインスペクタ」のオプション"-f"および"-s"を指定し、行番号シンボルとCPUモードシンボルを入れるようにして下さい。

```
INSP -f -s FILE.SYM
```

マップファイル名とシンボルファイル名は、ヘキサファイルと同じ名前にして下さい。

第 4 章 ウィンドウコマンド

4. 1 ウィンドウ

SFDは、ウィンドウ指向のデバッガで、次のウィンドウがあります。

1. コードウィンドウ
2. コマンドウィンドウ
3. レジスタウィンドウ
4. ステータスウィンドウ
5. ウォッチウィンドウ
6. サブウィンドウ
7. データ入力ウィンドウ
8. 検索ウィンドウ
9. メッセージウィンドウ
10. ヘルプウィンドウ

SFD起動時は、コードウィンドウとコマンドウィンドウが表示されます。レジスタウィンドウ、ステータスウィンドウ、ウォッチウィンドウは、ファンクションキーによりオープン/クローズします。

コードウィンドウ

コードウィンドウは、ソースコードや逆アセンブルコードを表示します。

ウィンドウは、行単位での拡大／縮小ができ、カーソルキーを使ってのロールアップ／ロールダウンができます。

コードウィンドウカーソルが有効なとき、ウィンドウコマンドによりブレークポイントの設定／解除、プログラムの実行等が行えます。

①	②	③	
コード	[]	ASM	SFD Version 1.00
④ >	008000 0053	BRK 53	
	008002 0000	BRK 00	
	008004 EB	XBA	
⑤	008005 A90000	LDA #0000	
	008008 EE7D00	INC !007D	
	00800B 00E4	BRK E4	
⑥	00800D E90000	SBC #0000	
	008010 7CF300	JMP (!00F3,X)	
	008013 00B3	BRK B3	
	008015 F000	BEQ 008017	
	008017 0068	BRK 68	
	008019 D400	PEI 00	
	00801B 003C	BRK 3C	

① "コード"が白色反転表示されているとき、コードウィンドウ内でのカーソルキー操作が可能です。上下カーソルキーやロールアップ、ロールダウンキーの行単位でロールアップ／ダウンします。

② 現在、参照可能なモジュール名を表示します。

③ 表示されているコードの種類を表示します。

ASM	逆アセンブルコード表示
SRC	ソースコード表示
SRC/ASM	ソース、逆アセンブル表示

④ コードウィンドウカーソル。カーソルキーがコードウィンドウで有効なとき、このマーク(>)が反転表示します。

⑤ CPUのPBR:PCが示すアドレスと、行の先頭のアドレスが等しいとき、その行は黄色反転表示されます。(PBRはプログラムバンクレジスタ、PCはプログラムカウンタを表します。)

⑥ アンダーライン表示されている行は、その行の先頭アドレスにシンプルバスサイクルブレークのオペコードリードが設定されていることを示します。

コマンドウィンドウ

コマンドウィンドウは、ダイログコマンドの入力やコマンドの実行結果を表示します。

	①	②	③	④	⑤
	コマンド	Timer:OFF	0s000m000.0u	Bcmp:OFF	STOP
⑦	>D 0,10	00 00 00 00 00 00 00 00	00-00 00 00 00 00 00 00 00		
⑥	>D 10,20				

- ① "コマンド"が白色反転表示されているとき、コードウィンドウ内でのカーソルキー操作が可能です。上下カーソルキーやロールアップ、ロールダウンキーで行単位のロールアップ／ダウンします。
- ② インターバルタイマ動作状態
ON…動作中 OFF…停止
- ③ インターバルタイマ値
タイマ値は、0秒から429.4967295秒までを秒(s)単位、ミリ秒(m)単位、マイクロ秒(u)単位に区切って表示します。また、タイマ値が最大秒数以上になったとき、タイマ値右横に"OVER"を表示します。
- ④ バスコンパレータ出力制御
ON…許可 OFF…禁止
- ⑤ エミュレーション動作状態
RUN…エミュレーション中 STOP…エミュレーション停止
- ⑥ ダイアログコマンド入力行
- ⑦ コマンド実行結果表示

レジスタウィンドウ

レジスタウィンドウは、CPUのレジスタの値を表示します。

レジスタ	
A= 00	D =0000
B= 00	S =01FF
C=0000	PC =8000
X=0000	DBR= 00
Y=0000	PBR= 00
P : Env-bdIzc	
Memory :	8-bit
Index :	8-bit

A : アキュムレータA

B : アキュムレータB

C : アキュムレータC (アキュムレータA、B)

X : インデックスX

Y : インデックスY

P : ステータスレジスタ

D : ダイレクトページレジスタ

S : スタックポインタ

PC : プログラムカウンタ

DBR : データバンクレジスタ

PBR : プログラムバンクレジスタ

ステータスレジスタの各ビットは、状態により英字の大文字と小文字で表します。大文字は、そのビットが'1'であることを示し、小文字は、そのビットが'0'であることを表します。

E : エミュレーション

N : ネガティブ

V : オーバーフローフラグ

M : メモリ/アキュムレータ選択

X : インデックスレジスタ選択

D : デシマルモード

I : IRQディスエーブル

Z : ゼロフラグ

C : キャリーフラグ

その他、レジスタウィンドウでは、ステータスレジスタのMフラグ、Xフラグの状態によりビット幅を表示します。

ICEの各種状態を表示します。

ステータス	
T.Power	: ON
T.Clock	: READY
RTT	: STOP
H-DMA	: OFF
V-BLANK	: OFF
Port Wv	: OFF
Mem. Wv	: OFF

ステータスウィンドウの表示および意味は、次の通りです。

T.Power	ターゲットパワー	ON...ON	OFF...OFF
T.Clock	ターゲットクロック	READY...レディ	FAULT...フォルト
RTT	リアルタイムトレース	RUN...動作中	STOP...停止
H-DMA	H-DMA動作	ON...許可	OFF...禁止
V-BLANK	V-BLANK同期動作	ON...許可	OFF...禁止
Port Wv	ポートライトベリファイ	ON...許可	OFF...禁止
Mem. Wv	メモリライトベリファイ	ON...許可	OFF...禁止

ウォッチウィンドウ

プログラム実行中、または実行後の特定メモリの内容を表示します。

ウォッチウィンドウは、ダイアログコマンドでウォッチ文を登録されていれば、ファンクションキーでオープン/クローズが行えます。

データや検索文字列を入力するためのウィンドウです。

1. データ入力用ダイアログボックス

データ入力用ダイアログボックスでは、次の様なウィンドウが画面に表示されます。



データを入力し、リターンキーを押すとウィンドウはクローズします。

データ入力中、ESCキーでこの入力をキャンセルすることが出来ます。

2. 検索文字列入力用ダイアログボックス

検索文字列を入力するためのダイアログボックスです。



ダイアログボックス上の矢印（↑↓）は、検索方向を示します。検索方向の切り替えは、このダイアログボックス表示中に、カーソルキー（↑↓）を押します。

↑ 後方検索を行います。

↓ 前方検索を行います。

4. 2 コントロールキー

コントロールキーを使い、コードウィンドウやコマンドウィンドウでの各種制御について説明します。

4. 2. 1 画面制御

CTRL+W	SHIFT+CLR	コマンドウィンドウを1行拡大し、コードウィンドウを1行縮小します。
CTRL+Z		コマンドウィンドウを1行縮小し、コードウィンドウを1行拡大します。
CTRL+E	↑	1行ロールダウンします。
CTRL+X	↓	1行ロールアップします。
CTRL+N	ROLLUP	半画面ロールアップします。
CTRL+U	ROLLDOWN	半画面ロールダウンします。

4. 2. 2 行編集制御

CTRL+A		カーソルを行の先頭に移動します。
CTRL+F		カーソルを行の終わりに移動します。
CTRL+S	←	カーソルを1文字左に移動します。
CTRL+D	→	カーソルを1文字右に移動します。
CTRL+G	DEL	カーソル上の1文字を消去します。
CTRL+H	BS	カーソルの一つ前の文字を消去します。
CTRL+Y		1行すべてを消去します。
CTRL+M	☒	行入力を終了します。
CTRL+V	INS	挿入／上書きモードを切り換えます。どのモードにあるかは、カーソルの形状で知ることが出来ます。挿入モードでは半高 (▬) のカーソル、上書きモードでは全高 (█) のカーソルが表示されます。
CTRL+I	TAB	タブは、いくつかのスペースに変換されます。
CTRL+R		行履歴内の一つ前の行をコピーします。
CTRL+T		行履歴内の一つ後の行をコピーします。
CTRL+Q		直前の入力行をコピーし、行入力を終了します。

4. 2. 3 コマンド実行制御

ダイアログコマンド実行中に、次の文字コードを入力することによりコマンド実行を制御するこ

とが出来ます。

- CTRL+S 文字出力を一時中断します。
- CTRL+Q 文字出力の一時中断を解除し、文字出力を再開します。
- CTRL+C コマンド実行を強制終了します。
- [ス^°-ス] 文字出力の中断／再開を繰り返します。

4. 3 ファンクションキー

ファンクションキーを使うことにより、プログラムの実行、ブレークポイントの設定、メモリの変更などが出来ます。

f・1 File ソースファイル表示

サブウィンドウがオープンし、表示可能なソースファイル（モジュール）を表示します。カーソルキー（↑↓）を使い、反転表示を目的のモジュール名に移動し、リターンキーを押します。

f・2 Regs レジスタウィンドウのオープン／クローズ

レジスタウィンドウのオープン／クローズを行います。レジスタウィンドウが画面に表示されていないとき、レジスタウィンドウをオープンします。また、レジスタウィンドウが画面に表示されているとき、レジスタウィンドウをクローズします。

レジスタウィンドウと共に、ステータスウィンドウもオープン／クローズします。

f・3 Watch ウォッチウィンドウのオープン／クローズ

ウォッチウィンドウのオープン／クローズを行います。ウォッチ文が登録されているとき有効で、ウォッチウィンドウが画面に表示されていないとき、ウォッチウィンドウをオープンします。また、ウォッチウィンドウが画面に表示されているとき、ウォッチウィンドウをクローズします。ウォッチ文を登録したとき、ウォッチウィンドウは自動的にオープンします。また、登録されている全てのウォッチ文を削除すると、ウォッチウィンドウは自動的にクローズします。

f・4 Code コードウィンドウの表示切り換え

コードウィンドウに表示されているコードを、ソースコード、逆アセンブルコード、ソース／逆アセンブルコードの順に切り換えます。

f・5 Go プログラムの実行

現在のプログラムカウンタの値からプログラムを実行します。

ESCキーでブレークします。

f・6 Cursor カーソル移動

カーソルキーが有効となるウィンドウを切り換えます。

f・7 Come 指定アドレスまでの実行

現在のプログラムカウンタの値から、コードウィンドウのカーソルの示す行までプログラムを実行します。

f・8 Trace トレース実行

現在のプログラムカウンタの値からプログラムを1命令実行します。

f・9 Break ブレークポイントの設定

コードウィンドウのカーソル行のアドレスに、シングルバスサイクルブレークポイントの設定／解除を行います。バスサイクルは、"オペコードリード"が設定されます。

f・10 Step ステップ実行

現在のプログラムカウンタの値からプログラムを1命令実行します。

サブルーチン (JSR / JSL) 命令は、1命令として実行します。

SHIFT+f・1 Find ソースコードの文字列検索 (検索文字列入力)

ダイアログボックスからソースコードを検索する文字列を入力します。

検索は行単位に行い、検索文字列を含む行が見つかったら、その行にカーソルが移動します。

SHIFT+f・2 Next ソースコードの文字列検索

「SHIFT+f・1 Find」で入力された文字列を再度検索します。

ブレークスイッチおよびICEステータスの設定を行います。

```

Break
Read only area write BREAK | OFF
No mount area access BREAK | OFF
Trace memory full BREAK | OFF
PPU illegal access BREAK | OFF
ICE status
H-DMA in system mode | OFF
V-BLANK synchronize | OFF
Port write verify | OFF
Memory write verify | OFF

```

コンフィグレーション設定では、以下の項目の設定を行います。

Read only area write BREAK	リードオンリエリアにライトした時、プログラムをブ レークするか否かを設定します。
No mount area access BREAK	メモリ未実装エリアをアクセスした時、プログラムをブ レークするか否かを設定します。
Trace memory full BREAK	トレースメモリが一杯になった時、プログラムをブ レークするか否かを設定します。
PPU illegal access BREAK	P P U不正アクセスがあった時、プログラムをブ レークするか否かを設定します。
H-DMA in system mode	システムモード時のH-DMA動作の許可／禁止を設定 します。
V-BLANK synchronize	V B L A N K同期動作の許可／禁止を設定します。
Port write verify	ポートライトベリファイの許可／禁止を設定します。
Memory write verify	メモリライトベリファイの許可／禁止を設定します。

1. 設定

カーソルキー（↑↓）を使い、設定する項目にカーソルを移動します。スペースキーを使い ON/OFF 表示を切り換えます。

2. キー操作

f・6	Init	すべての状態を"OFF"にします。
f・10	Cancel	設定した状態を無効にし、ウィンドウをクローズします。
ESC		現在表示されている値を設定し、ウィンドウをクローズします。

Bus cycle comparator												
#	Address	Data	Status	LC	H1	H2	PC	H1=1 HF1	H2=0 HF2	ITM=E ITM	ITM CBP	RTT
1	>00:0FFF	=0*000000	=WIOBB*--	X	*	*	H1	>	-	-	X	-
2	<00:2000	=0*000000	=WIOBB*--	*	*	*	H1	>	-	-	-	-
3	=00:80**	=5A	=W10*****	*	*	*	H1	>	-	-	-	X
4	**:*	**	*****	*	*	*	H1	-	-	-	-	-
5	**:*	**	*****	*	*	*	H1	-	-	-	-	-
6	**:*	**	*****	*	*	*	H1	-	-	-	-	-
7	**:*	**	*****	*	*	*	H1	-	-	-	-	-
8	**:*	**	*****	*	*	*	H1	-	-	-	-	-

Status DMA : Don't Care

Address	Data	Status	Pass.C	En/Dis	Init			Cancel
---------	------	--------	--------	--------	------	--	--	--------

バスサイクルコンパレータ設定では、以下の項目を設定します。

Address	アドレスの設定値・一致モード・マスクを設定します。
Data	データの設定値・一致モード・マスクを設定します。
Status	ステータスの設定値・一致モード・マスクを設定します。
LC	次のバスコンパレータの出力をAND条件で入れるか否かを設定します。
H1	ホールドフラグ1の設定値を設定します。
H2	ホールドフラグ2の設定値を設定します。
PC	パスカウントの設定値・動作モードを設定します。
HF1	ホールドフラグ1のセット／リセット条件を設定します。
HF2	ホールドフラグ2のセット／リセット条件を設定します。
ITM	インターバルタイマーのイネーブル／ディセーブル条件を設定します。
CBP	コンプレックスブレイクポイントの発生条件を設定します。
RTT	リアルタイムトレースのトリガ条件を設定します。

1. 設定方法

カーソルキーを使い項目を選択し、設定値の選択または設定値入力をします。ウィンドウ下には、現在選択されている項目のガイドメッセージが表示されます。

設定値の選択は、スペースキーまたはリターンキーを使います。

設定値の入力は、データキーを押しダイアログボックスで入力します。

以下に各項目の設定方法について説明します。

□ Address

スペースキーまたはリターンキーでアドレスの一致モードを選択します。アドレス一致モードの表示文字とその意味は次の通りです。

- = 設定値のとき条件が真
- > 設定値より大のとき条件が真
- < 設定値より小のとき条件が真
- ! 設定値と違うとき条件が真

データキーを押すとダイアログボックスが表示され、アドレスの設定値を入力することが出来ます。データ入力が16進数のときのみ、ワイルドカード”*”を使うことが出来ます。また、アドレスの文字列の先頭に先の一致モード表示文字を付けることで、一致モードをこの状態からも設定することが出来ます。

□Data

スペースキーまたはリターンキーでデータの一致モードを選択します。データ一致モードの表示文字とその意味は次の通りです。

- = 設定値のとき条件が真
- > 設定値より大のとき条件が真
- < 設定値より小のとき条件が真
- ! 設定値と違うとき条件が真

データキーを押すとダイアログボックスが表示され、データの設定値を入力することが出来ます。データ入力が16進数または2進数のときのみ、ワイルドカード”*”を使用することが出来ます。また、データの文字列の先頭に先の一致モード表示文字を付けることで、一致モードをこの状態からも設定することが出来ます。ワイルドカードを使用した場合、16進数の場合は2桁、2進数の場合は8桁で表示されます。

□Status

スペースキーまたはリターンキーを押下することによりステータスの一致モードを選択します。ステータス一致モードの表示文字とその意味は次の通りです。

- = 設定値のとき条件が真
- ! 設定値と違うとき条件が真

ステータスの設定値8ビットは、カーソル移動によりすべてビット単位で設定します。各ビットの表示文字およびその意味を左から順に説明します。

R/W	R	リード。R/W信号が1のとき真
	W	ライト。R/W信号が0のとき真
	*	不定。R/W信号が1または0のとき真
VDA	1	VDA信号が1のとき真
	0	VDA信号が0のとき真
	*	不定。VDA信号が1または0のとき真
VPA	1	VPA信号が1のとき真
	0	VPA信号が0のとき真
	*	不定。VPA信号が1または0のとき真

M	B	バイト。M信号が1のとき真
	W	ワード。M信号が0のとき真
	*	不定。M信号が1または0のとき真
X	B	バイト。X信号が1のとき真
	W	ワード。X信号が0のとき真
	*	不定。X信号が1または0のとき真
DMA	D	DMA中。DMA信号が0のとき真
	—	非DMA中。DMA信号が1のとき真
	*	不定。DMA信号が1または0のとき真
HBLANK	B	水平ブランキング中。HBLANK信号が1のとき真
	—	非水平ブランキング中。HBLANK信号が0のとき真
	*	不定。HBLANK信号が1または0のとき真
VBLANK	B	垂直ブランキング中。VBLANK信号が1のとき真
	—	非垂直ブランキング中。VBLANK信号が0のとき真
	*	不定。VBLANK信号が1または0のとき真

□LC(Link Control)

スペースキーまたはリターンキーで次のバスコンパレータの条件をカーソルがあるバスコンパレータに加える（AND条件として算入）か否かを設定します。表示文字およびその意味は次の通りです。

- X 次のバスコンパレータ条件を加える。
- * 次のバスコンパレータ条件を加えない。

□H1/H2

スペースキーまたはリターンキーでホールドフラグ1/ホールドフラグ2それぞれの設定値を選択します。表示文字およびその意味は次の通りです。

- X ホールドフラグ1/ホールドフラグ2が1のとき真
- * 不定。ホールドフラグ1/ホールドフラグ2が1または0のとき真

□PC(Pass Count)

スペースキーまたはリターンキーでパスカウン트의動作モードを選択します。表示文字および意味は次の通りです。

- R リピートモード。一致出力がパスカウント回数に1回出力される。
- H ホールドモード。一致出力がパスカウント回数後1回のみ出力される。

データキーを押すとダイアログボックスが表示され、パスカウント値を入力することが出来ます。値は10進数（1～65536）で入力します。パスカウント値の文字列の先頭に先の動作モード表示文字を付加することで、動作モードをこの状態からも設定することが出来ます。

□ H F 1 (Hold flag1)

スペースキーまたはリターンキーでホールドフラグ1のセット／リセット条件に当コンパレータ出力を加える（OR条件として算入）か否かを選択します。その表示文字および意味は次の通りです。

- S 当コンパレータ出力をホールドフラグ1のセット条件に加える。
- R 当コンパレータ出力をホールドフラグ1のリセット条件に加える。
- 条件に加えない。

□ H F 2 (Hold flag2)

スペースキーまたはリターンキーでホールドフラグ2のセット／リセット条件に当コンパレータ出力を加える（OR条件として算入）か否かを選択します。その表示文字および意味は次の通りです。

- S 当コンパレータ出力をホールドフラグ1のセット条件に加える。
- R 当コンパレータ出力をホールドフラグ1のリセット条件に加える。
- 条件に加えない。

□ I T M (Interval timer)

スペースキーまたはリターンキーでインターバルタイマーのイネーブル／ディセーブル条件に当コンパレータ出力を加える（OR条件として算入）か否かを選択します。その表示文字および意味は次の通りです。

- E 当コンパレータ出力をインターバルタイマーのイネーブル条件に加える。
- D 当コンパレータ出力をインターバルタイマーのディセーブル条件に加える。
- 条件に加えない。

□ C B P (Complex break point)

スペースキーまたはリターンキーでコンプレックスブレイクポイントの発生条件に当コンパレータ出力を加える（OR条件として算入）か否かを選択します。その表示文字および意味は次の通りです。

- X 当コンパレータ出力をコンプレックスブレイクポイントの発生条件に加える。
- 発生条件に加えない。

□ R T T (Real time trace)

スペースキーまたはリターンキーでリアルタイムトレースのトリガ条件に当コンパレータ出力を加える（OR条件として算入）か否かを選択します。その表示文字および意味は次の通りです。

- X 当コンパレータ出力をリアルタイムトレースのトリガ条件に加える。
- トリガ条件に加えない。

2. ファンクションキー

ファンクションキーにより次のコマンドが実行できます。実行は、カーソルがどの項目にあっても構いません。

- f・1 **Address** カーソルのあるバスコンパレータのアドレス設定値を直接入力します。
- f・2 **Data** カーソルのあるバスコンパレータのデータ設定値を直接入力します。
- f・3 **Status** カーソルのあるバスコンパレータのステータス設定値の内、R/W、VPA、VDAについてCPUのステート表現で値を設定します。この時、次のようなサブウィンドウが表示され、カーソルキーでステートを選択し、リターンキーで決定することが出来ます。

```
CPU State
Data read
Data write
Data read/write
OP-code read
Operand read
Instruction read
Data/instruction read/write
```

- f・4 **Pass.C** カーソルのあるバスコンパレータのパスカウント設定値を直接入力します。
- f・5 **En/Dis** カーソルのあるバスコンパレータの出力を許可／禁止を設定します。現在の状態は、パスカウント設定値右の文字で示します。その表示文字および意味は次の通りです。
 - > 出力は許可されている。
 - | 出力は禁止されている。
- f・6 **Init** カーソルのあるバスコンパレータのすべての設定値を既定値に初期化します。
- f・10 **Cancel** 現在までの設定を無効にしてウィンドウをクローズします。

3. 状態表示

ホールドフラグ1、ホールドフラグ2、インターバルタイマフラグの現在の状態をウィンドウ右上に表示します。表示文字および意味は次の通りです。

H 1	ホールドフラグ 1	0	ホールドフラグ 1 が"0"
		1	ホールドフラグ 1 が"1"
H 2	ホールドフラグ 2	0	ホールドフラグ 2 が"0"
		1	ホールドフラグ 2 が"1"
I T M	インターバルタイマ	E	イネーブル
		D	ディセーブル

4. 設定終了

ESCキーにより設定値が実際に設定され、ウィンドウはクローズします。またこの時、ホールドフラグ1／2とパスカウント値は、"0"にリセットされます。

コードウィンドウのカーソルのある行のアドレスを、レジスタのプログラムカウンタ（PBR：PC）にセットします。

DMAレジスタの表示、設定を行います。

H-DMA/DMA		Ch.7	Ch.6	Ch.5	Ch.4	Ch.3	Ch.2	Ch.1	Ch.0
H-DMA	OFF	ON	OFF	ON	OFF	ON	OFF	ON	ON
DMA	ON	OFF	ON	OFF	ON	OFF	ON	OFF	OFF
		Ch.7	Ch.6	Ch.5	Ch.4	Ch.3	Ch.2	Ch.1	Ch.0
43X0	FF	FF	FF	FF	FF	FF	FF	FF	FF
43X1	FF	FF	FF	FF	FF	FF	FF	FF	FF
43X2	FF	FF	FF	FF	FF	FF	FF	FF	FF
43X3	FF	FF	FF	FF	FF	FF	FF	FF	FF
43X4	FF	FF	FF	FF	FF	FF	FF	FF	FF
43X5	FF	FF	FF	FF	FF	FF	FF	FF	FF
43X6	FF	FF	FF	FF	FF	FF	FF	FF	FF
43X7	FF	FF	FF	FF	FF	FF	FF	FF	FF
43X8	FF	FF	FF	FF	FF	FF	FF	FF	FF
43X9	FF	FF	FF	FF	FF	FF	FF	FF	FF
43XA	FF	FF	FF	FF	FF	FF	FF	FF	FF

Init Cancel

1. 表示

DMAおよびH DMAチャンネルと、DMAレジスタの現在の値を表示します。

2. チャンネルの設定

カーソルキー（↑↓←→）を使い、設定する項目を選びます。スペースキーでON/OFFの表示が切り替わります。

3. DMAレジスタの変更

カーソルキー（↑↓←→）を使い、変更する項目を選びます。データキーを押し、ダイアログボックスから変更するデータを入力します。

4. 終了

次のキーで設定を終了します。

ESC

設定された値を実際のレジスタに書き込みます。

f・10 Cancel 設定された値を無効にし、ウィンドウをクローズします。

5. その他のファンクションキー

f・6 Init すべての状態を"OFF", "FF"の状態にします。

現在のPPUレジスタ、およびPPU内部レジスタの表示、変更を行います。

Current PPU internal register															
INIDISP	2100	00	BG1VOFS*H	210E	00	VMDATAH	2119	00	W34SEL	2124	00				
OBJSEL	2101	00	BG2HOFS*L	210F	00	M7SEL	211A	00	WOBJSEL	2125	00				
OAMADDL	2102	00	BG2HOFS*H	210F	00	M7A*L	211B	00	WH0	2126	00				
OAMADDH	2103	00	BG2VOFS*L	2110	00	M7A*H	211B	00	WH1	2127	00				
OAMADDL*	2102	00	BG2VOFS*H	2110	00	M7B*L	211C	00	WH2	2128	00				
OAMADDH*	2103	00	BG3HOFS*L	2111	00	M7B*H	211C	00	WH3	2129	00				
OAMDATA*L	2104	00	BG3HOFS*H	2111	00	M7C*L	211D	00	WBGLOG	212A	00				
OAMDATA*H	2104	00	BG3VOFS*L	2112	00	M7C*H	211D	00	WOBJLOG	212B	00				
BGMODE	2105	00	BG3VOFS*H	2112	00	M7D*L	211E	00	TM	212C	00				
MOZAIKU	2106	00	BG4HOFS*L	2113	00	M7D*H	211E	00	TS	212D	00				
BG1SC	2107	00	BG4HOFS*H	2113	00	M7X*L	211F	00	TMW	212E	00				
BG2SC	2108	00	BG4VOFS*L	2114	00	M7X*H	211F	00	TSW	212F	00				
BG3SC	2109	00	BG4VOFS*H	2114	00	M7Y*L	2120	00	CGWSEL	2130	00				
BG4SC	210A	00	VMAINC	2115	00	M7Y*H	2120	00	CGADDSUB	2131	00				
BG12NBA	210B	00	VMADDL	2116	00	CGADD	2121	00	COLDATA	2132	00				
BG34NBA	210C	00	VMADDH	2117	00	CGADD*	2121	00	COLDATA*R	2132	00				
BG1HOFS*L	210D	00	VMADDL*	2116	00	CGDATA*L	2122	00	COLDATA*G	2132	00				
BG1HOFS*H	210D	00	VMADDH*	2117	00	CGDATA*H	2122	00	COLDATA*B	2132	00				
BG1VOFS*L	210E	00	VMDATAL	2118	00	W12SEL	2123	00	SETINI	2133	00				

Switch

Restor

1. 表示切り替え

次のファンクションキーで、PPUレジスタの表示が切り替わり、PPUレジスタとPPU内部レジスタの値が交互に表示されます。

f・1 Switch

2. データの変更

カーソルキー（↑↓←→）で設定項目を選択します。データキーを押しダイアログボックスから変更データを入力します。入力データは、すべて1バイトです。

3. PPU内部レジスタのリセット

次のファンクションキーで、PPU内部レジスタの値をブレイクした時点の状態に戻すことが出来ます。

f・10 Restor

4. 終了

ESCキーでウィンドウはクローズし、PPUレジスタ設定を終了します。

リアルタイムトレースの動作モードの設定、およびステータスの表示を行います。

RTT Status			
Trace: ON			
Mode : Only(Cont.)			
Step : 2048			
Rstart	Start	Stop	Mode

1. 表示項目

Trace 現在のR T Tの状態を表示します。

ON 開始

OFF 停止

Mode 現在のR T Tの動作モードを表示します。

Before トリガポイント前2048ステップをトレース

About トリガポイント前後1024ステップをトレース

After トリガポイント後2048ステップをトレース

Only(Cont.) トリガポイントのみ連続してトレース

Only(Stop) トリガポイントのみ2048ステップまでトレース

Step 現在のR T Tメモリにトレースされているバスサイクルのステップ数を表示します。0～2048の10進数で表示します。

2. ファンクションキー

ファンクションキーにより次のコマンドが実行できます。

f・1 Rstart R T TメモリをクリアしてR T Tを開始します。

f・2 Start 項目"Mode"に表示されている動作モードで、R T Tを開始します。

f・3 Stop R T Tを停止します。

f・4 Mode R T Tの動作モードを設定します。 本キーを押すとウィンドウ内の項目"Mode"の表示が順次変わります。

リアルタイムトレースメモリの内容を表示します。

RTT Memory					
Line	Address	Data	Stat	RDPMDHV	Instruction
>0000	00:8000	18	OC	R11BB---	CLC
0001	00:8001	18	IO	R00BB---	
0002	00:8001	FB	OC	R11BB---	XCE
0003	00:8002	FB	IO	R00BB---	
0004	00:8002	C2	OC	R11BB---	REP #10
0005	00:8003	10	OR	R01BB---	
0006	00:8003	10	IO	R00BB---	
0007	00:8004	E2	OC	R11BB---	SEP #20
0008	00:8005	20	OR	R01BW---	
0009	00:8005	20	IO	R00BW---	
0010	00:8006	A2	OC	R11BW---	LDX #0100
0011	00:8007	00	OR	R01BW---	
0012	00:8008	01	OR	R01BW---	
0013	00:8009	9A	OC	R11BW---	TXS
0014	00:800A	9A	IO	R00BW---	
0015	00:800A	A9	OC	R11BW---	LDA #00

Find Next

1. 表示内容

ウィンドウ内には次の項目を表示します。

Line	トレースメモリ内で最も古いバスサイクルを0として番号付けしたステップ番号です。トリガポイントのステップには、">"を先頭に表示します。
Address	アドレス値を表示します。
Data	データの値を表示します。
Stat	バスステータスの内、R/W、VDA、VPAの3ビットでデコードされるCPUのステートを表示します。そのステート、表示文字、および意味は次の通りです。

R/W	VDA	VPA	Stat	意味
0	0	0	? 1	不定状態1
0	0	1	? 2	不定状態2
0	1	0	DW	データライト
0	1	1	? 3	不定状態3
1	0	0	IO	内部動作
1	0	1	OR	オペランドリード
1	1	0	DR	データリード
1	1	1	OC	オペコードリード

RDPMXDHV バスステータスを表示します。その表示順序は、次の通りです。

R	R/W
V	VDA
P	VPA
M	M
X	X
D	DMA
H	HBLANK
V	VBANK

Instruction 実行した命令のニーモニックを表示します。

2. カーソル移動

カーソルキー（↑↓）やロールキー（ROLLUP,ROLLDOWN）により表示画面がスクロールします。

3. ファンクションキー

ファンクションキーにより次のコマンドが実行できます。

f・6 Find 文字列の検索を行います。ダイアログボックスから検索する文字列を入力します。入力後、文字列の検索を行い、その文字列を含む行にカーソルが移動します。

f・7 Next "f・6 Find" で入力した文字列を再度検索します。

Main memory [00:0033]																
Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
00:0000	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Byte	Word	Long			Find	Next			Jump
------	------	------	--	--	------	------	--	--	------

メインメモリの内容の表示、変更、検索を行います。

1. メモリの表示

ウィンドウには、データをバイト（1バイト）、ワード（2バイト）、ロングワード（3バイト）単位で表示させることができます。

カーソルキー（↑↓←→）やロールアップ／ダウンキー（ROLLUP/ROLLDOWN）を使い、ウィンドウのスクロールが行えます。現在カーソルのあるアドレス値はウィンドウ上部に表示されます。

次のファンクションキーを使い、表示するデータのサイズを変更することができます。

- f・1 Byte バイト表示に切り換えます。
- f・2 Word ワード表示に切り換えます。
- f・3 Long ロングワード表示に切り換えます。

2. メモリの変更

カーソルキーを使い、カーソルを目的のアドレスに移動します。データキーを押すとダイアログボックスがオープンしますので、変更するデータを入力して下さい。リターンキーで確定、ESCキーでキャンセルします。

3. メモリの検索

次のファンクションキーを使い、データを検索することができます。

- f・6 Find 検索データの入力およびデータ検索
- f・7 Next 次の候補を検索します。

検索データは、カンマで区切って複数指定ができます。

データが見つかったら、カーソルがそのアドレス上に移動します。

4. アドレス入力によるカーソル移動

次のファンクションキーを使い、アドレス入力によるカーソルの移動が行えます。

f・10 Jump 指定アドレスへのジャンプ

ダイアログボックスでアドレスを入力します。

5. ウィンドウの表示

Main memory [00:0033]																
Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
00:0000	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00:0070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

バイト表示画面

Main memory [00:0011]									
Address	+0	+2	+4	+6	+8	+A	+C	+E	
00:0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
00:0010	0000	0000	0000	0000	0000	0000	0000	0000	0000
00:0020	0000	0000	0000	0000	0000	0000	0000	0000	0000
00:0030	0000	0000	0000	0000	0000	0000	0000	0000	0000
00:0040	0000	0000	0000	0000	0000	0000	0000	0000	0000
00:0050	0000	0000	0000	0000	0000	0000	0000	0000	0000
00:0060	0000	0000	0000	0000	0000	0000	0000	0000	0000
00:0070	0000	0000	0000	0000	0000	0000	0000	0000	0000

ワード表示画面

Main memory [00:0018]									
Address	+0	+3	+6	+9	+C	+F	+12	+15	
00:0000	000000	000000	000000	000000	000000	000000	000000	000000	000000
00:0018	000000	000000	000000	000000	000000	000000	000000	000000	000000
00:0030	000000	000000	000000	000000	000000	000000	000000	000000	000000
00:0048	000000	000000	000000	000000	000000	000000	000000	000000	000000
00:0060	000000	000000	000000	000000	000000	000000	000000	000000	000000
00:0078	000000	000000	000000	000000	000000	000000	000000	000000	000000

ロングワード表示画面

CG-RAM Address	+0	+1	+2	+3	+4	+5	+6	+7
0000	0000	0000	0000	0000	0000	0000	0000	0000
0010	0000	0000	0000	0000	0000	0000	0000	0000
0020	0000	0000	0000	0000	0000	0000	0000	0000
0030	0000	0000	0000	0000	0000	0000	0000	0000
0040	0000	0000	0000	0000	0000	0000	0000	0000
0050	0000	0000	0000	0000	0000	0000	0000	0000
0060	0000	0000	0000	0000	0000	0000	0000	0000
0070	0000	0000	0000	0000	0000	0000	0000	0000

Find Next Jump

CG-RAMの内容の表示、変更、検索を行います。

1. メモリの表示

ウィンドウには、ワード（2バイト）単位でデータを表示します。カーソルキー（↑↓←→）やロールアップ/ダウンキー（ROLLUP/ROLLDOWN）を使い、ウィンドウのスクロールが行えます。現在カーソルのあるアドレス値はウィンドウ上部に表示されます。

2. メモリの変更

カーソルキーを使い、カーソルを目的のアドレスに移動します。データキーを押すとダイアログボックスがオープンしますので、変更するデータを入力して下さい。リターンキーで確定、ESCキーでキャンセルします。

3. メモリの検索

次のファンクションキーを使い、データを検索することが出来ます。

f・6 Find 検索データの入力およびデータ検索

f・7 Next 次の候補を検索します。

検索できるデータは1つだけです。

データが見つかり、カーソルがそのアドレス上に移動します。

4. アドレス入力によるカーソル移動

次のファンクションキーを使い、アドレス入力によるカーソルの移動が行えます。

f・10 Jump 指定アドレスへのジャンプ

ダイアログボックスでアドレスを入力します。

OAM		[0021]							
Address		+0	+1	+2	+3	+4	+5	+6	+7
0000		0000	0000	0000	0000	0000	0000	0000	0000
0010		0000	0000	0000	0000	0000	0000	0000	0000
0020		0000	0000	0000	0000	0000	0000	0000	0000
0030		0000	0000	0000	0000	0000	0000	0000	0000
0040		0000	0000	0000	0000	0000	0000	0000	0000
0050		0000	0000	0000	0000	0000	0000	0000	0000
0060		0000	0000	0000	0000	0000	0000	0000	0000
0070		0000	0000	0000	0000	0000	0000	0000	0000

Find Next Jump

OAMの内容の表示、変更、検索を行います。

操作方法は、「CTRL+f・2 CG-RAM CG-RAMの表示、変更」を参照して下さい。

V-RAM		[0021]							
Address		+0	+1	+2	+3	+4	+5	+6	+7
0000		0000	0000	0000	0000	0000	0000	0000	0000
0010		0000	0000	0000	0000	0000	0000	0000	0000
0020		0000	0000	0000	0000	0000	0000	0000	0000
0030		0000	0000	0000	0000	0000	0000	0000	0000
0040		0000	0000	0000	0000	0000	0000	0000	0000
0050		0000	0000	0000	0000	0000	0000	0000	0000
0060		0000	0000	0000	0000	0000	0000	0000	0000
0070		0000	0000	0000	0000	0000	0000	0000	0000

Find Next Jump

V-RAMの内容の表示、変更、検索を行います。

操作方法は、「CTRL+f・2 CG-RAM CG-RAMの表示、変更」を参照して下さい。

第5章 ダイアログコマンド

5.1 ダイアログコマンドについて

ダイアログコマンドは、コマンドウィンドウにプロンプト">"が表示されているときに入力できます。

5.2 コマンド入力書式

以下にコマンドの入力書式を示します。

>コマンド名 パラメータ [; コマンド名 パラメータ ; ...]

コマンド名とパラメータの間は、原則的にスペースで区切ります。但し、コマンド名が一意に特定できる場合は続けて書くことが出来ます。

例1 : >D B アドレス000B番地からのメモリの内容を表示します。

例2 : >DB 前回のDコマンドの続きから表示します。

パラメータが複数ある場合は、各パラメータをカンマ", "で区切って指定します。

例 : >D 0,10

各コマンドをセミコロン";"で区切ると、1行に複数のコマンドを記述することが出来ます。

例 : >D 0,10;D 20,30 アドレス0番地から10番地までのメモリを表示後、20番地から30番地までのメモリを表示します。

5.3 ダイアログコマンド

次に、各ダイアログコマンドについて説明します。

A (アセンブル)

65C816ニーモニックをマシンコードに変換し、メモリへ書き込みます。

A [adr][,cpu]

ロードアドレスadr番地からCPUモードcpuに従いアセンブルし、マシンコードをメモリへ書き込みます。ロードアドレス省略時は、直前のAコマンドの続きから始まります。CPUモード省略時は、直前のAコマンドのCPUモードとなります。

CPUモードには、次の文字を指定します。

- | | |
|----|-----------------------------------|
| m | アセンブラ内の仮想Mフラグを"0"とします。 |
| M | アセンブラ内の仮想Mフラグを"1"とします。 |
| x | アセンブラ内の仮想Xフラグを"0"とします。 |
| X | アセンブラ内の仮想Xフラグを"1"とします。 |
| mx | アセンブラ内の仮想Mフラグを"0"、仮想Xフラグを"0"とします。 |
| mX | アセンブラ内の仮想Mフラグを"0"、仮想Xフラグを"1"とします。 |
| Mx | アセンブラ内の仮想Mフラグを"1"、仮想Xフラグを"0"とします。 |
| MX | アセンブラ内の仮想Mフラグを"1"、仮想Xフラグを"1"とします。 |

Aコマンドを実行すると、入力モードになります。ピリオド"."の入力で、Aコマンドは終了します。リターンキーのみの入力では、メモリの変更は行われません。

次の擬似命令が使えます。

- | | |
|-------|--|
| ORG | ロードアドレスを変更します。 |
| IDX8 | アセンブラ内の仮想Xフラグを"1" (X, Y: 8ビット幅) にし、これ以降のイミディエート・アドレッシングのオペランドを1バイトで生成します。 |
| IDX16 | アセンブラ内の仮想Xフラグを"0" (X, Y: 16ビット幅) にし、これ以降のイミディエート・アドレッシングのオペランドを2バイトで生成します。 |
| MEM8 | アセンブラ内の仮想Mフラグを"1" (Acc: 8ビット幅) にし、これ以降のイミディエート・アドレッシングのオペランドを1バイトで生成します。 |
| MEM16 | アセンブラ内の仮想Mフラグを"0" (Acc: 16ビット幅) にし、これ以降のイミディエート・アドレッシングのオペランドを2バイトで生成します。 |
| DS | 指定された大きさの領域を確保します。 |
| DB | バイト (1バイト) 単位でデータをメモリに書き込みます。 |
| DW | ワード (2バイト) 単位でデータをメモリに書き込みます。 |
| DL | ロングワード (3バイト) 単位でデータをメモリに書き込みます。 |

ADR (アドレス表示)

アドレスの表示書式を切り換えます。

ADR

アドレスの表示書式には、次の2通りがあります。

XXXXXX 24ビットアドレス

XX:XXXX バンクアドレス、バンク内アドレス。

B (ブレークスイッチ)

ブレークスイッチの表示、設定を行います。

B

現在設定されているブレークスイッチを表示します。

B switch

スイッチswitchに、次の文字を指定することで、ブレークスイッチが"ON"になります。

R...リードオンリエリア

N...ノーマウントエリアアクセス

P...PPUイリーガルアクセス

T...トレースメモリフル

*...すべてのブレークスイッチ

スイッチは、複数指定が出来ます。

-B switch

スイッチswitchに、次の文字を指定することで、ブレークスイッチが"ON"になります。

R...リードオンリエリア

N...ノーマウントエリアアクセス

P...PPUイリーガルアクセス

T...トレースメモリフル

*...すべてのブレークスイッチ

スイッチは、複数指定が出来ます。

BC (バスサイクルコンパレータ)

バスサイクルコンパレータの設定状況の表示、設定を行います。

BC

現在のバスサイクルコンパレータの設定状況と、制御フラグの内容を表示します。

制御フラグには、ホールドフラグ1、ホールドフラグ2とインターバルタイマ動作があります。

BC num,adr,data,stat,link,hflg,pcnt,act

バスサイクルコンパレータを設定します。

num 設定するコンパレータ番号(1~8)を指定します。

adr アドレスの設定値とアドレス一致モードを指定します。

設定値が16進数のときのみワイルドカード(*)が使えます。一致モードは、設定値の前に次の記号を置きます。一致モードを省略すると直前のモードが設定されます。

=...設定値と同じ、 !...設定値と違う

>...設定値より大きい、 <...設定値より小さい

data データの設定値とデータ一致モードを指定します。

設定値が16進数または2進数のときのみワイルドカード(*)が使えます。一致モードを指定することが出来ます。一致モードは、設定値の前に次の記号を置きます。

一致モードを省略すると、直前のモードが設定されます。

=...設定値と同じ、 !...設定値と違う

>...設定値より大きい、 <...設定値より小さい

stat バスステータスにおける条件を次の順序で9文字で指定します。

一致モード: =...設定値と同じ、 !...設定値と違う

>...設定値より大きい、 <...設定値より小さい

R/W: R...リード、 W...ライト、 *...不定

VDA: 0...OFF、 1...ON、 *...不定

VPA: 0...OFF、 1...ON、 *...不定

M : B...メモリアキュムレータバイトモード、 W...ワードモード、 *...不定

X : B...インデックスバイトモード、 W...ワードモード、 *...不定

DMA: D...DMA中、 -...非DMA中、 *...不定

HBLANK: B...水平フランク中、 -...非水平フランク中、 *...不定

VBLANK: B...垂直フランク中、 -...非垂直フランク中、 *...不定

link 現在設定しているバスコンパレータ条件に、次のバスコンパレータ条件をAND条件として加えるか否かを指定します。ただし、次のコンパレータ出力のバスカウント条件は含まれません。

X...次のバスコンパレータ条件を加える

*...次のバスコンパレータ条件を加えない

hflg ホールドフラグの状態をコンパレータ条件に加えるか否かを指定します。

ホールドフラグが”1”のとき、条件成立となります。ホールドフラグ1、2の順に次の文字を指定します。

X…コンパレータ条件に加える

*…コンパレータ条件に加えない

pent パスカウントの設置値とパスカウント動作モードを指定します。

設定値は、10進数（1～65536）で指定します。動作モードは、設定値の前に次の文字を置きます。

H…ホールドモード

R…リピートモード

act コンパレータ条件が成立したときの動作を決めます。動作は、次の順に5文字で指定します。

ホールドフラグ1 : S…セット（1）、 R…リセット（0）、 -…未使用

ホールドフラグ2 : S…セット（1）、 R…リセット（0）、 -…未使用

インターバルタイ : E…許可、 D…禁止、 -…未使用

コンプレックスブレイク : X…発生、 -…未使用

リアルタイムトレース : X…トリガ、 -…未使用

BCE num

指定した番号のコンパレータ条件をイネーブルにします。

番号numには、バスコンパレータ番号（1～8）を指定します。複数指定可能です。

BCD num

指定した番号のコンパレータ条件をディセーブルにします。

番号numには、バスコンパレータ番号（1～8）を指定します。複数指定可能です。

BCR

バスコンパレータをリセットします。

BS (シンプルバスサイクルブ레이크)

シンプルバスサイクルブ레이크ポイントの表示、設定を行います。

BS

現在設定されているブ레이크ポイントを表示します。

BS [cycle=]adr1[, [cycle=]adr2,...]

ブ레이크ポイントを設定します。ブ레이크アドレスadr番地に、ブ레이크条件cycleを設定します。ブ레이크条件cycleには、次の文字を指定します。

R...データリード

W...データライト

D...データ リード/ライト

C...オペコードリード

P...オペランドリード

O...オペコード/オペランド リード

*...すべての条件

複数のブ레이크ポイントを1度に設定する場合、ブ레이크条件が同じ場合は2つ目以降のcycleは省略できます。

-BS adr1[,adr2,...]

adr番地に設定されたブ레이크ポイントを解除します。

C (割り込みコントロール)

割り込みコントロールの表示、設定を行います。

C

現在の割り込みコントロールの設定状況を表示します。

C int

intで指定した割り込みを許可にします。

割り込みintには、次の文字を指定します。1度に複数指定できます。

A...ABORT

R...RESET

N...NMI

I...IRQ

*...すべての割り込み

-C int

intで指定した割り込みを禁止にします。

割り込みintには、次の文字を指定します。1度に複数指定できます。

A...ABORT

R...RESET

N...NMI

I...IRQ

*...すべての割り込み

CTint cnt (cntは10進数入力)

intで指定した割り込みを、cnt回おきに割り込みリクエストを受け付けるようにします。

割り込みintには、次の文字を指定します。

A...ABORT

N...NMI

I...IRQ

CSint step

intで指定した割り込みを、step回の割り込みでその割り込みを禁止するようにします。

割り込みintには、次の文字を指定します。

A...ABORT

N...NMI

I...IRQ

CCint

intで指定した割り込みを、カウント機能に設定し、カウント値をクリアします。

割り込みintには、次の文字を指定します。

A...ABORT

N...NMI

I...IRQ

CMD (コマンドファイル)

コマンドファイルを作成します。

CMD file

このコマンド以降に入力されたダイアログコマンドを、ファイルfileに出力します。ファイル名拡張子は、". CMD"になります。ファイルは新規作成となり、同じ名前のファイルが存在する場合、そのファイルは削除されます。

CMD +file

このコマンド以降に入力されたダイアログコマンドを、ファイルfileに追加書き込みします。ファイル名拡張子は、". CMD"になります。

COFF

コマンドファイル作成を終了します。このコマンドは、コマンドファイルに出力されません。

CP (画面コピー)

画面のイメージをファイルに出力します。

CP file

ファイル名fileに画面イメージを出力します。ファイルは、新規ファイルとして作成され、既に同じ名前のファイルが存在する場合、既存のファイルは消去されます。

CP +file

ファイル名fileに画面イメージを追加出力します。ファイルfileが存在しない場合は、新規ファイルとして作成されます。

※画面に表示されている文字の属性（反転、アンダーライン等）は、出力されません。

CS (チェックサム)

メモリのチェックサムを算出し、表示します。

CS adr1,adr2

アドレスadr1番地からadr2番地までのチェックサムを算出し、表示します。

D (メモリ表示)

指定サイズ、表示基数でメモリの内容を表示します。

D[size] adr1,adr2,radix

アドレスadr1番地からadr2番地までをサイズsizeバイト、表示基数radixで表示します。

サイズsizeには、次の文字を指定します。

B…バイト表示 (1バイト)

W…ワード表示 (2バイト)

L…ロングワード表示 (3バイト)

サイズ省略時は、バイト表示となります。

表示基数radixには、次の文字を指定します。

H…16進数表示

D…10進数表示

O…8進数表示

B…2進数表示

表示基数省略は、直前のDコマンドの表示基数となります。

D[size] adrs,&cnt,radix

アドレスadrs番地から、cnt個分を表示基数radixで表示します。

DG/DO/DV (指定メモリ表示)

指定メモリの内容を表示します。

DG adr1,adr2,radix

CG-RAMのアドレスadr1番地からadr2番地までをワード単位(2バイト)、表示基数radixで表示します。

表示基数radixには、次の文字を指定します。

H...16進数表示

D...10進数表示

O...8進数表示

B...2進数表示

表示基数省略は、直前のDコマンドの表示基数となります。

DG adrs,&cnt,radix

CG-RAMのアドレスadrs番地から、cnt個分を表示基数radixで表示します。

DO adr1,adr2,radix

OAMのアドレスadr1番地からadr2番地までをワード単位(2バイト)、表示基数radixで表示します。

DO adrs,&cnt,radix

OAMのアドレスadrs番地から、cnt個分を表示基数radixで表示します。

DV adr1,adr2,radix

VRAMのアドレスadr1番地からadr2番地までをワード単位(2バイト)、表示基数radixで表示します。

DV adrs,&cnt,radix

VRAMのアドレスadrs番地から、cnt個分を表示基数radixで表示します。

DMA (DMA/HDMAレジスタ)

DMAレジスタの表示、設定を行います。

DMA

現在のDMAレジスタの設定状況を表示します。

DMA ch

chで指定したDMAのチャンネルをONにします。

チャンネル番号は0～7までの内、1度に複数指定できます。

-DMA ch

chで指定したDMAのチャンネルをOFFにします。

チャンネル番号は0～7までの内、1度に複数指定できます。

DMA adr,dat

HDMA ch

chで指定したHDMAのチャンネルをONにします。

チャンネル番号は0～7までの内、1度に複数指定できます。

-HDMA ch

chで指定したHDMAのチャンネルをOFFにします。

チャンネル番号は0～7までの内、1度に複数指定できます。

ER (エミュレーションCPUリセット)

エミュレーションCPUをリセットします。

ER

エミュレーションCPUをリセットします。

EX (コマンドファイル実行)

コマンドファイルを実行します。

EX file

ファイル名fileをコマンドファイルとして実行します。ファイル名拡張子".CMD"が、自動的に付加されます。

F (メモリフィル)

指定範囲のメモリにデータ列を書き込みます。

F[size] adr1,adr2,data

アドレスadr1番地からadr2番地までを、データ列dataでフィルします。データは、指定したサイズsize単位で書き込まれます。

データが複数ある場合は、それぞれのデータをカンマ", "で区切って指定します。

サイズsizeには、次の文字を指定します。

B…バイト単位 (1バイト)

W…ワード単位 (2バイト)

L…ロングワード単位 (3バイト)

サイズ省略時は、バイト単位となります。

F[size] adrs,&cnt,data

アドレスadrs番地からcnt個分を、データ列dataでフィルします。

G (プログラム実行)

プログラムを実行します。

G [adr1][,adr2...]

アドレスadr1番地からプログラムを実行します。アドレスadr1を省略すると、現在のCPUレジスタのプログラムカウンタが示すアドレスから実行します。

一時ブレークアドレスadr2を指定すると、プログラム実行中にプログラムカウンタがadr2と等しいとき、プログラムがブレークします。一時ブレークアドレスを複数指定する場合は、各アドレスの間をカンマ", "で区切ります。

プログラム実行中は、次のキーを受け付けます。

ESC…プログラムを強制ブレークします。

Q……プログラムを実行したまま、SFDを終了します。

H (データの演算)

データの各種演算を行い、結果を表示します。

H data

データdataを、16進数、符号無し10進数、符号付き10進数、アスキーコードで表示します。

H data1,data2

データdata1とdata2を演算（加減乗除、剰余、AND、OR、XOR）し、結果を16進数、符号無し10進数、符号付き10進数、アスキーコードで表示します。

L (逆アセンブル表示)

コードウィンドウ、コマンドウィンドウに逆アセンブルコードを表示します。

L adrs

コードウィンドウに、アドレスadrs番地から逆アセンブルコードを表示します。

L L adrs,cpu

コマンドウィンドウに、アドレスadrs番地からCPUモードcpuに従い、逆アセンブルコードを表示します。

CPUモードcpuには、次の文字を指定します。

- m アセンブラ内の仮想Mフラグを"0"とします。
- M アセンブラ内の仮想Mフラグを"1"とします。
- x アセンブラ内の仮想Xフラグを"0"とします。
- X アセンブラ内の仮想Xフラグを"1"とします。
- m x アセンブラ内の仮想Mフラグを"0"、仮想Xフラグを"0"とします。
- m X アセンブラ内の仮想Mフラグを"0"、仮想Xフラグを"1"とします。
- M x アセンブラ内の仮想Mフラグを"1"、仮想Xフラグを"0"とします。
- M X アセンブラ内の仮想Mフラグを"1"、仮想Xフラグを"1"とします。

※CPUモードmodeを設定しても、CPUモードシンボルが読み込まれている場合、その値に従って表示します。

LOG (ログファイル作成)

ログファイルを作成します。

LOG file

ファイル名fileでログファイルを作成します。ファイル名には、ファイル名拡張子". LOG"が付加されます。ファイルは、新規作成となります。ファイル名fileと同じ名前のファイルが存在する場合、そのファイルは削除されます。

LOG +file

ファイル名fileにログ情報を追加出力します。ファイルが存在しない場合は、新たに作成されます。

LOFF

ログファイル作成を終了します。

M (メモリ転送)

指定範囲のメモリブロックを、他のメモリブロックに転送します。

M adr1,adr2,adr3

アドレスadr1番地からadr2番地までのメモリブロックを、アドレスadr3番地からのメモリに転送します。

M adr1,&cnt,adr2

アドレスadr1番地からcntバイト分のメモリブロックを、アドレスadr2番地からのメモリに転送します。

MM (メモリマップ)

メモリマップの表示、設定を行います。

MM

メモリのマッピング状況と、システムエリアとして使えるメモリのKB（キロバイト）数を表示します。

MM adr1,adr2,attr

アドレスadr1番地からadr2番地までを属性attrに設定します。

属性attrには、次の文字を指定します。

UA…ユーザーRAMエリア

UO…ユーザーROMエリア

SA…システムRAMエリア

SO…システムROMエリア

N…メモリ未実装エリア

MS (メモリ検索)

データ列を検索します。

MS[size] adr1,adr2,data

アドレスadr1番地からadr2番地までの範囲で、データ列dataを捜します。データ列が見つかり、その先頭アドレスを表示します。

データ列dataはサイズsize単位として指定し、複数指定する場合は各データの間をカンマ", "で区切ります。サイズsizeには、次の文字を指定します。

B…バイト単位 (1バイト)

W…ワード単位 (2バイト)

L…ロングワード単位 (3バイト)

サイズ省略は、バイト単位となります。

MV (メモリベリファイ)

メモリブロックを比較し、データの違うアドレスを表示します。

MV adr1,adr2,adr3

アドレスadr1番地からadr2番地までのデータと、アドレスadr3番地からのデータとを比較し、データの違うアドレスを表示します。

MV adrs,&byte,adr2

アドレスadrs番地からbyteバイト分のデータと、アドレスadr2番地からのデータとを比較し、データの違うアドレスを表示します。

N (データ入力基数の変更)

データの入力基数の表示、変更を行います。

N

現在のデータの入力基数を表示します。

Nradix

このコマンド以降のデータの入力基数を、基数radixに変更します。SFD起動時の入力基数は16進です。基数を変更した場合、変更した基数でのデータ入力時の基数記号は省略することが出来ます。基数radixには、次の文字を指定します。

16…16進数入力とします。

10…10進数入力とします。

8…8進数入力とします。

2…2進数入力とします。

PI (I/Oポート入力)

I/Oポートのデータの表示、変更を行います。

PI size adrs

アドレスadrs番地のI/Oポートにサイズsizeバイト単位でデータを書き込みます。

サイズsizeには、次の文字を指定します。

B…バイト(1バイト)単位

W…ワード(2バイト)単位

このコマンド実行すると入力モードになります。データは、カンマ", "で区切ると複数指定することが出来ます。入力モードでは、次の制御文字を受け付けます。

=adrs…アドレスをadrs番地に変更します。

リターンキー…アドレスをsize分インクリメントします。

^ …アドレスをsize分インクリメントします。

; …アドレスをsize分デクリメントします。

. …このコマンドを終了します。

PO (I/Oポート出力)

I/Oポートにデータを書き込みます。

POsize adr,data

アドレス`adr`番地のI/Oポートにサイズ`size`バイト単位でデータ列`data`を書き込みます。データ列`data`は、カンマ", "で区切ると複数指定することが出来ます。

サイズ`size`には、次の文字を指定します。

B…バイト(1バイト)単位

W…ワード(2バイト)単位

PPU (PPUレジスタ)

PPUレジスタの内容を表示します。

PPU

現在のPPUレジスタの内容を表示します。

PPUI

ブレーク時のPPU内部レジスタの内容を表示します。

PPUR

ブレーク時のPPU内部レジスタの値をPPUレジスタにライトします。

Q (SFD終了)

SFDを終了します。

Q

SFDを終了します。

RD (ファイルリード)

ヘキサファイル、シンボルファイル、マップファイル、バイナリファイルを読み込みます。

RD file

ファイル名fileを読み込みます。ファイル名拡張子を省略した場合、ヘキサファイル、シンボルファイル、マップファイルを読み込みます。

ファイル名に次の拡張子を付けることで、特定のファイルのみを読み込むことができます。

".HEX"...ヘキサファイルを読み込みます。

".SYM"...シンボルファイルを読み込みます。

".MAP"...マップファイルを読み込みます。

RDB file,adrs,attr

バイナリファイルfileを読み、データを指定したメモリattrのアドレスadrs番地に書き込みます。メモリattrには、次の文字を指定します。

M...メインメモリ

C...CG-RAM

O...OAM

V...VRAM

※メモリattr省略時は、メインメモリに読み込みます。

S (メモリ変更)

メモリの内容を変更します。

S size adrs

アドレスadrs番地からサイズsizeバイト単位でメモリの内容を変更します。

サイズsizeには、次の文字を指定します。

B…バイト（1バイト）単位

W…ワード（2バイト）単位

L…ロングワード（3バイト）単位

このコマンド実行すると入力モードになります。データは、カンマ”，”で区切ると複数指定することが出来ます。入力モードでは、次の制御文字を受け付けます。

=adrs…アドレスをadrs番地に変更します。

リターンキー…アドレスをsize分インクリメントします。

^ …アドレスをsize分インクリメントします。

; …アドレスをsize分デクリメントします。

. …このコマンドを終了します。

S size adrs,data

アドレスadrs番地からサイズsizeバイト単位でメモリの内容をデータ列dataに変更します。データ列dataは、カンマ”，”で区切ると複数指定することが出来ます。

サイズsizeには、次の文字を指定します。

B…バイト（1バイト）単位

W…ワード（2バイト）単位

L…ロングワード（3バイト）単位

SG/SO/SV (指定メモリの変更)

指定メモリの内容を変更します。

SG adrs

アドレスadrs番地からワード(2バイト)単位でCG-RAMの内容を変更します。

このコマンド実行すると入力モードになります。データは、カンマ", "で区切ると複数指定することが出来ます。入力モードでは、次の制御文字を受け付けます。

=adrs…アドレスをadrs番地に変更します。

リターン…アドレスをsize分インクリメントします。

^ …アドレスをsize分インクリメントします。

; …アドレスをsize分デクリメントします。

. …このコマンドを終了します。

SG adrs,data

アドレスadrs番地からワード(2バイト)単位でCG-RAMの内容をデータ列dataに変更します。データ列dataは、カンマ", "で区切ると複数指定することが出来ます。

SO adrs

アドレスadrs番地からワード(2バイト)単位でOAMの内容を変更します。

このコマンド実行すると入力モードになります。

SO adrs,data

アドレスadrs番地からワード(2バイト)単位でOAMの内容をデータ列dataに変更します。

データ列dataは、カンマ", "で区切ると複数指定することが出来ます。

SV adrs

アドレスadrs番地からワード(2バイト)単位でVRAMの内容を変更します。

このコマンド実行すると入力モードになります。

SV adrs,data

アドレスadrs番地からワード(2バイト)単位でVRAMの内容をデータ列dataに変更します。

データ列dataは、カンマ", "で区切ると複数指定することが出来ます。

ST (ICEステータス)

ICEステータスの表示、設定を行います。

ST

現在のICEステータスの設定状況を表示します。

ST switch

スイッチswitchで指定されたICEステータスの動作を許可にします。

スイッチswitchには、次の文字を指定します。

M…メモリライトベリファイ

P…ポートライトベリファイ

V…VBLANK同期動作

H…システムモード時のHDMA動作

*…すべての動作

-ST switch

スイッチswitchで指定されたICEステータスの動作を禁止にします。

スイッチswitchには、次の文字を指定します。

M…メモリライトベリファイ

P…ポートライトベリファイ

V…VBLANK同期動作

H…システムモード時のHDMA動作

*…すべての動作

T I M (インターバルタイマ)

インターバルタイマの設定、表示を行います。

T I M

現在のインターバルタイマ値を表示します。

T I M switch

インターバルタイマをONまたはOFFにします。

スイッチswitchには、次の文字を指定します。

ON……タイマをクリアし、タイマ動作を許可にします。

OFF…タイマ動作を禁止します。

TM (リアルタイムトレース)

リアルタイムトレースの開始、トレースメモリの表示を行います。

TM

現在設定されているトレースモードと、トレースメモリの記憶サイクル数を表示します。

TMS mode

指定したトレースモードmodeによりリアルタイムトレースを開始します。

トレースモードmodeには、次の文字を指定します。

BE…トリガポイント前

AB…トリガポイント前後

AF…トリガポイント後

OC…トリガポイントのみ (継続モード)

OS…トリガポイントのみ (停止モード)

TMR

リアルタイムトレースを再開します。

TMP

リアルタイムトレースを停止します。

TMD line,step

line行からstep行分のトレースメモリを表示します。

TG (ターゲット)

ターゲットパワーとターゲットクロックの状態を表示します。

TG

現在のターゲットパワーとターゲットパワーの状態を表示します。

T (トレース/ステップ実行)

プログラムをトレースまたはステップ実行します。

T step

現在のプログラムカウンタの値から1命令ずつstep命令分のプログラムをトレース実行し、1命令終了ごとにレジスタの内容を表示します。

TS step

現在のプログラムカウンタの値から1命令ずつstep命令分のプログラムをステップ実行し、1命令終了ごとにレジスタの内容を表示します。サブルーチン命令(JSR/JSR)やブロックムーブ命令(MVN/MVP)は、1命令として実行されます。

※すべての割り込み入力禁止されます。

※実時間実行ではありません。

U (トレース/ステップ実行)

プログラムをトレースまたはステップ実行します。

U step

現在のプログラムカウンタの値から1命令ずつstep命令分のプログラムをトレース実行します。
実行終了後、レジスタの内容を表示します。

US step

現在のプログラムカウンタの値から1命令ずつstep命令分のプログラムをステップ実行します。
実行終了後、レジスタの内容を表示します。サブルーチン命令(JSR/JSL)やブロック
ムーブ命令(MVN/MVP)は、1命令として実行されます。

※すべての割り込み入力は禁止されます。

※実時間実行ではありません。

VER (バージョン番号の表示)

SFDおよびコントロールプログラムのバージョン番号の表示。

VER

SFDおよびコントロールプログラムのバージョン番号を表示します。

VMM (ROMエリアのメモリマッピング)

指定されたバンクのROMエリアの属性を設定します。

VMM bank1, bank2, attr

バンクアドレスbank1からbank2までのROMエリアを、属性attrに設定します。

属性attrには、次の文字を指定します。

U...ユーザーROMエリア

S...システムROMエリア

N...メモリ未実装エリア

W (ウォッチ)

ウォッチ文の登録、削除およびウォッチ制御の設定を行います。

Wsize adr1,adr2,radix

ウォッチ文を登録します。アドレスadr1番地からadr2番地までのデータを、サイズsize単位で表示基数radixで表示します。

Wsize adrs,&cnt,radix

ウォッチ文を登録します。アドレスadrs番地からcnt個分のデータを、サイズsize単位で表示基数radixで表示します。

WD num

numで指定された番号のウォッチ文を削除します。

WI

現在のウォッチインターバルの設定状況を表示します。

WI time,switch

ウォッチインターバルを設定します。スイッチswitchを"ON"にすると、エミュレーション中に指定時間time(秒)ごとにウォッチウィンドウを更新します。スイッチが"OFF"の場合、更新は行いません。時間timeは、1から255秒までの範囲で設定できます。

※ウォッチデータは、ウォッチウィンドウに表示されます。

WT (ファイルライト)

メモリのデータをファイルに書き込みます。

WT file,adr1,adr2 (ヘキサファイル作成)

アドレスadr1番地からadr2番地までのデータを、ヘキサファイルとしてファイル名fileに書き込みます。ファイル名拡張子". H E X"が自動的に付加されます。

WTB file,adr1,adr2,attr,mode (バイナリファイル作成)

attrで指定したメモリのアドレスadr1番地からadr2番地までのデータを、ファイルfileに書き込みます。ファイルは、バイナリファイルとして作成されます。

モードmodeに"A"を指定すると、追加書き込みとなります。モード省略時は、ファイルは新規作成されます。メモリattrには、次の文字を指定します。

C...CG-RAM

O...OAM

V...VRAM

※モードを省略するとメインメモリとなります。

X (レジスタ)

CPUレジスタの表示、変更を行います。

X

現在のCPUレジスタの内容を表示します。

X reg=data

レジスタregの値を、データdataに変更します。

X reg

レジスタregから順に変更します。このコマンドを実行すると入力モードになります。変更するデータを入力し、リターンキーを押すとレジスタの値が変更されます。入力モードでリターンキーのみの場合、レジスタの値は変更されず次のレジスタに移ります。ピリオド". "を入力すると、このコマンドを終了します。

レジスタ名"reg"には、次の文字を指定します。

P	ステータスレジスタ
A	アキュムレータA
B	アキュムレータB
C	アキュムレータC (アキュムレータA, B)
X	インデックスX
Y	インデックスY
D	ダイレクトレジスタ
S	スタックポインタ
PC	プログラムポインタ
DBR	データバンクレジスタ
PBR	プログラムバンクレジスタ

ステータスレジスタでは、各フラグを次の文字で指定します。大文字はフラグ"1"、小文字はフラグ"0"の状態を示します。

Eフラグ	エミュレーション	E = 1、	e = 0
Nフラグ	ネガティブ	N = 1、	n = 0
Vフラグ	オーバーフロー	V = 1、	v = 0
Mフラグ	メモリ/アキュムレータ	M = 1、	m = 0
Xフラグ	インデックスレジスタ選択	X = 1、	x = 0
Dフラグ	デシマルモード	D = 1、	d = 0
Iフラグ	IRQディセーブル	I = 1、	i = 0
Zフラグ	ゼロ	Z = 1、	z = 0

Cフラグ キャリー
Bフラグ ブレーク

$C = 1$ 、 $c = 0$

$B = 1$ 、 $b = 0$

Y (シンボル表示)

シンボルを表示します。

Y

グローバルシンボル、ローカルシンボル、モジュール名、CPUモードシンボル、ユーザー定義シンボルを表示します。

Y name

文字列nameを先頭を含む、グローバルシンボル、ローカルシンボル、モジュール名、ユーザー定義シンボルを表示します。

YG

すべてのグローバルシンボルを表示します。

YG name

文字列nameを先頭を含むグローバルシンボルを表示します。

YL

すべてのローカルシンボルを表示します。

YL name

文字列nameを先頭を含むローカルシンボルを表示します。

YC

CPUモードシンボルの設定状況を表示します。

YM

すべてのモジュール名を表示します。

YM name

文字列nameを先頭を含むモジュール名を表示します。

YU

すべてのユーザー定義シンボルを表示します。

YU name

文字列nameを先頭を含むユーザー定義シンボルを表示します。

Y D name

シンボル名nameのユーザー定義シンボルを削除します。

(タブ)

タブの値を設定します。

tab

ダイアログコマンド入力時、またはソースコード表示時にタブはいくつかのスペースに変換されて表示されます。#コマンドでは、tab桁ごとにタブが設定されます。

※ (MS-DOSコマンド実行)

MS-DOSのコマンドを実行します。

※

SFDを起動したまま、MS-DOSコマンドレベルに戻ります。

MS-DOSコマンドの"EXIT"、またはファンクションキーの【f・10】でSFDに戻ります。

※ cmd

MS-DOSコマンドcmdを実行します。コマンド実行後、何れかのキーを押すとSFDに戻ります。

! (ヒストリ)

入力された行のヒストリを表示、実行します。

!

入力された行のヒストリを表示します。

!!

一つ前に入力された行をコマンドとして実行します。

! num

numで指定されたヒストリ行をコマンドとして実行します。

第6章 マクロコマンド

6.1 マクロ機能

マクロ機能は、複数のダイログコマンドを組み合わせ、新しいコマンドを作成する機能です。登録されたマクロコマンドは、ダイアログコマンドと同じように扱うことができます。また、マクロ機能では、マクロ制御文やマクロ変数を使うことにより、自由度のあるマクロコマンドを作成することができます。

6.2 マクロコマンドの実行

マクロコマンドの実行は、ダイアログコマンドと同じように扱えます。マクロコマンドの実行書式は、次のようになります。

>マクロコマンド パラメータ

パラメータは、カンマ", "で区切って最大10個まで指定できます。指定したパラメータは左から、マクロコマンド内の"&0"から"&9"までに対応します。

例：マクロコマンドm tを登録します。このマクロコマンドを実行すると、アドレス0番地からマクロパラメータ0で指定したアドレスまでのメモリ内容を表示します。

>MACRO z ←マクロコマンドの登録

* D 0,&0

*ENDM

>z 7 ←アドレス0番地から7番地までのメモリを表示

>z 100 ←アドレス0番地から100番地までのメモリを表示

6.3 マクロコマンドの登録

次に、マクロコマンドの登録、削除などを行うためのダイアログコマンドについて説明します。

MACRO (マクロ登録)

マクロコマンドの登録

MACRO name

Command

ENDM

マクロ名nameで、マクロコマンドをSFDに登録します。KILLコマンドで削除するか、SFD終了まで有効です。登録中に"CANM"を入力すると、登録はキャンセルされます。

※マクロ名は、ダイアログコマンド名と同じ名前を付けることはできません。

MACRO

Commnad

ENDM

一時的なマクロコマンドとしてSFDに登録します。登録終了後、直ちに個のマクロコマンドが実行されます。実行終了後、このマクロコマンドは消去されます。

※一時マクロでは、パラメータの指定は出来ません。

MLIST (マクロ表示)

登録されたマクロコマンドを表示します。

MLIST

登録されたすべてのマクロコマンドの一覧を表示します。

MLIST name

マクロ名nameのマクロコマンドを表示します。

K I L L (マクロ削除)

マクロコマンドを削除します。

K I L L macro

指定されたマクロ名macroを削除します。

6. 4 マクロ制御文と式評価

マクロ制御文には次のものがあり、マクロコマンド内でのみ使うことができます。

REPEAT {} 文	繰り返し
WHILE {} 文	条件付き繰り返し
DO {} WHILE 文	条件付き繰り返し
IF {} 文	条件制御
BREAK 文	ループからの抜け出し

マクロ制御文によっては、式評価の結果で処理を分岐する場合があります。
式評価には、次の記号を使います。

式1 == 式2	式1と式2が等しいとき"真"。
式1 != 式2	式1と式2が等しくないとき"真"。
式1 >= 式2	式1が式2より大きいか、等しいとき"真"。
式1 <= 式2	式1が式2より小さいか、等しいとき"真"。
式1 > 式2	式1が式2より大きいとき"真"。
式1 < 式2	式1が式2より小さいとき"真"。

次に、マクロ制御文について説明します。

REPEAT {} (繰り返し)

指定回数の繰り返し実行

```
REPEAT { cnt      (cntは10進数入力)
  Command
}
```

コマンドCommandをcnt回実行します。

WHILE {} (条件付き繰り返し)

条件によりコマンドを繰り返し実行します。

```
WHILE { 式  
    Command  
}
```

式評価の結果が”真”の間、コマンドCommandを実行します。

`DO {} WHILE` (条件付き繰り返し)

条件によりコマンドを繰り返し実行します。

```
DO {  
    Command  
} WHILE 式
```

式評価の結果が”真”の間、コマンドCommandを実行します。

I F {} (条件制御)

式評価の結果により分岐します。

```
I F { 式1
      Command1
} E L S E I F { 式2
      Command2
} E L S E {
      Command3
}
```

式評価の結果が”真”ならば、I Fに続くコマンドを実行します。”偽”ならば、I Fに対応する”{”または”E L S E”を実行します。

”} E L S E I F {”または”} E L S E {”は、省略できます。

BREAK (ループからの抜け出し)

ループから抜け出します。

BREAK

BREAK文が繰り返し制御文の中にある時は、直近のループから抜け出します。また、繰り返し制御文の外にあるときは、マクロコマンドの実行を終了します。

付録A アセンブラの書式

A. 1 ニーモニック

ニーモニック：

ADC	AND	ASL	BCC	BCS	BEQ	BIT	BMI
BNE	BPL	BRA	BRK	BRL	BVC	BVS	CLC
CLD	CLI	CLV	CMP	COP	CPX	CPY	DEC
DEX	DEY	EOR	INC	INX	INY	JML	JMP
JSL	JSR	LDA	LDX	LDY	LSR	MVN	MVP
NOP	ORA	PEA	PEI	PER	PHA	PHB	PHD
PHK	PHP	PHX	PHY	PLA	PLB	PLD	PLP
PLX	PLY	REP	ROL	ROR	RTI	RTL	RTS
SBC	SEC	SED	SEI	SEP	STA	STP	STX
STY	STZ	TAX	TAY	TCD	TCS	TDC	TRB
TSB	TSC	TSX	TXA	TXS	TXY	TYA	TYX
WAI	WDM	XBA	XCE				

ニーモニックのエイリアス：

ニーモニック		エイリアス
BCC	...	BLT
BCS	...	BGE
DEC A	...	DEA
INC A	...	INA
TCD	...	TAD
TCS	...	TAS
TDC	...	TDA
TSC	...	TSA
XBA	...	SWA

A. 2 アドレッシングモード

イミディエート	#
アブソリュート	! a
アブソリュート・ロング	> a l
ダイレクト	< d
アキュムレータ	A
インプライド	
ダイレクト・インダイレクト・インデックスド・Y	(< d), Y
ダイレクト・インダイレクト・ロング・インデックスド・Y	[< d], Y
ダイレクト・インデックスド・インダイレクト・X	(< d, X)
ダイレクト・X	< d, X
ダイレクト・Y	< d, Y
アブソリュート・X	! a, X
アブソリュート・ロング・X	> a, X
アブソリュート・Y	! a, Y
リラティブ	
リアラティブ・ロング	
アブソリュート・インダイレクト	(! a)
ダイレクト・インダイレクト	(< d)
ダイレクト・インダイレクト・ロング	[< d]
アブソリュート・インデックスド・インダイレクト	(! a, X)
スタック	
スタック・リラティブ	< d, S
スタック・リラティブ・インダイレクト・インデックスド	(< d, S), Y
ブロック・ムーブ	src, dst

A. 3 擬似命令

```

ORG    MEM8    MEM16    IDX8    IDX16
DS     DB      DW      DL

```


SUPER FAMICM DEBUGGER SFD ユーザーズ・マニュアル
Copyright (C) 1991, 株式会社リコー 電子デバイス事業部

- ★本書の内容の一部又は全部を、株式会社リコーの同意無しに転載することは禁止されています。
- ★本資料は、工業所有権その他権利の実施に対する保証、又は実施権の許諾をおこなうものではありません。
- ★本品は外国為替及び外国貿易管理法（外為法）に定められる戦略物資等（貨物又は役務）に該当します。本品を輸出又は日本国外へ持ち出す場合は、外為法及び関連法規に基づく輸出手続きが必要です。

