Universal Probe

ソフトウェア・ユーザーズ・マニュアル

SPI Writer

Copyright © 2014 Sohwa & Sophia Technologies Inc.

No. J090961-02



目 次

注	意事項6
	使用上の注意
略	語・用語・記載ルール8
1.	本ソフトウェアの概要10
	1.1. システム要件
	1.2. 特徴
	1.2.1. SPI フラッシュメモリ操作機能11
	1.3. 電源の入れ方/切り方12
	1.3.1. ターゲットとの接続方法と電源の入れ方12
	1.3.2. 電源の切り方とターゲットからの外し方12
2.	操作方法13
	2.1. 作業フロー
	2.2. 本ソフトウェアの起動14
	2.2.1. 起動方法
	2.3. プロジェクトファイルの選択17
	2.3.1. プロジェクトファイルを新規作成する17
	2.3.2. プロジェクトファイルを開く20
	2.4. プローブの選択
	2.5. License Code の登録
	2.6. SPI フラッシュメモリ設定24
	2.7. プローブ環境の設定
	2.8. SPI フラッシュメモリのクリア
	2.9. ユーザーデータの書き込み
	2.10. SPI フラッシュメモリのフィルアップ
	2.11. プロジェクトファイルの保存32
	2.12. 本ソフトウェアの終了
3.	本ソフトウェアでのデータ表現34
	3.1. 数值表現
	3.2. アドレス表現
	3.3. データ表現
	3.4. SPI フラッシュメモリの読み込み
4.	メニュー
2/	131



	4.1. ファイル	36
	4.1.1. 閉じる	36
	4.1.2. プロジェクトの新規作成	36
	4.1.3. プロジェクトを開く	38
	4.1.4. プロジェクトの保存	38
	4.1.5. プロジェクトを名前をつけて保存	38
	4.1.6. プロジェクトを閉じる	39
	4.1.7. アプリケーションの終了	39
	4.2. リソース	40
	4.2.1. ダウンロード	40
	4.2.2. アップロード	42
	4.2.3. プローブ環境の設定	44
	4.2.4. バッチファイルの自動実行設定	48
	4.2.5. メモリサーチ	49
	4.2.6. SPI フラッシュメモリ	51
	4.3. 実行	52
	4.3.1. キャンセル	52
	4.3.2. ベリファイ(ライト時に比較)	52
	4.3.3. ベリファイオンリー(ライトせずに比較)	52
	4.3.4. チェックサム値の計算	53
	4.4. SPI フラッシュメモリ	54
	4.4.1. SPI フラッシュメモリ設定ダイアログ	54
	4.4.2. デバイスタブ	54
	4.4.3. クリアタブ	56
	4.4.4. フィルアップタブ	56
_		
5.	ワインドワ	7
	5.1. ウィンドウのドッキングについて	57
	5.1.1. ウィンドウの状態	58
	5.1.2. 子ウィンドウ状態の変更方法	60
	5.1.3. ドラッグ&ドロップによるドッキング状態の変更方法	62
	5.1.4. 同じ辺に複数の子ウィンドウをドッキングさせた場合	64
	5.1.5. ドッキングウィンドウのサイズ変更	65
	5.2. ダンプウィンドウ	66
	5.2.1. ダンプウィンドウのコンテキストメニュー	68
	5.2.2. ダンプウィンドウの複数表示	72
	5.3. フマンドウィンドウ	73
~		
6.	キーホード	'4
	6.1. ダンプウィンドウ	74
	6.2. コマンドウィンドウ	74
3/	131	
Uni	versal Probe ソフトウェア・ユーザーズ・マニュアル - SPI Writer	



7.	. コマンドラインインターフェース	75
	7.1. コマンドラインでの操作方法	75
	7.1.1. コマンドの入力方法	75
	7.2.2. 連想選択方式によるコマンド入力	75
	7.2.3. 履歴から入力する	76
	7.2. コマンド解説	77
	7.2.1. ASSIGN とドット(.)(式の評価)	77
	7.2.2. BATCH(バッチプログラムの実行)	78
	7.2.3. CD(フォルダの変更、ドライブの変更、カレントパス表示)	79
	7.2.4. CLOSE(プロジェクトファイルを閉じる)	80
	7.2.5. CLS(コマンドウィンドウのクリア)	81
	7.2.6. DIR(フォルダ内容の参照)	82
	7.2.7. DUMP(メモリダンプ)	83
	7.2.8. ENV(プローブ環境設定)	85
	7.2.9. ERROR_ECHO(エラーメッセージ表示の設定)	86
	7.2.10. EXIT(本ソフトウェアの終了)	87
	7.2.11. FM(SPI フラッシュメモリの設定)	88
	7.2.12. FMCLEAR(SPI フラッシュメモリのクリア)	90
	7.2.13. FMFILL(SPI フラッシュメモリのフィルアップ)	91
	7.2.14. LOAD(オブジェクトデータのロード)	92
	7.2.15. LOADPARAM(LOAD コマンドのパラメータ補足)	94
	7.2.16. LOG(コマンドウィンドウのロギング制御)	95
	7.2.17. MKDIR(フォルダの作成)	96
	7.2.18. NEWBATCH(バッチファイルの作成)	97
	7.2.19. OPTION(コマンドウィンドウのオプション設定)	98
	7.2.20. QUERY(各種設定状態の参照)	99
	7.2.21. RADIX(入力基数の設定)	
	7.2.22. SAVEWIN(コマンドウィンドウのファイル出力)	
	7.2.23. SEARCH(メモリサーチ)	
	7.2.24. SHELLEXE(ファイルの実行)	
	7.2.25. UPLOAD(オブジェクトデータのアップロード)	
	7.2.26. VERIFY(ベリファイの設定)	106
8.	. バッチ機能	108
	8.1. ワーク変数	
	8.2. ラベル	110
	8.3. コメント	
	8.4. 数値演算で使用可能な演算子	112
	8.4.1. 演算子	
	8.4.2. 優先度と評価順序	113



8.5. SPI フラッシュメモリの読み込み	
8.6. 実行制御	
8.6.1. FOR,FBREAK,NEXT(カウンタ付き繰り返し実行)	
8.6.2. WHILE,WBREAK,WEND(繰り返し実行)	
8.6.3. GOTO(無条件分岐)	
8.6.4. IF,ELSEIF,ELSE,ENDIF(条件判断)	
8.6.5. END(全てのバッチプログラムの終了)	
8.6.6. QUIT(現在のバッチプログラムの終了)	
8.7. ECHO(バッチコマンドの表示/非表示の切り替え)	
8.8. KEYIN(キーボードからの入力)	
8.9. PRINT(文字列の表示)	
8.10. BEEP(BEEP 音を鳴らす)	
8.11. WAIT(バッチプログラムの一時停止)	
9. スタンドアロン機能	126
9.1. 手順の記録	
9.2. 記録の終了	
9.3. 記録した手順のバックアップとリストア	
改訂履歴	130
製造者情報	



注意事項

このたびは株式会社 Sohwa & Sophia Technologies 製「Universal Probe」をお買い上げいただき、誠にありがとう ございます。本書に記載されている注意事項などを正しくご理解のうえ、お使いいただきますようお願い申し上げます。

- 1. 本書に記載の製品及び技術で、『外国為替及び外国貿易法』に該当するものを輸出する時、又は、国外に持ち出す時は、日本政府の許可が必要です。
- 2. 本書に記載されている製品は、一般電子機器(事務機器、通信機器、計測機器、家電製品など)に使用されることを意図しております。特別な品質、信頼性が要求され、その故障や誤動作が直接人命を脅かしたり、人体に危害を及ぼしたりする恐れのある特定用途機器(自動車・鉄道・船舶・航空・宇宙用機器、交通機器、燃焼機器、安全装置、医療機器、インフラ機器、原子力など)には使用しないでください。もしこれらの機器でご使用になる場合は、お客様の責任のもとでご使用ください。
- 3. 本書の内容の一部または全部を当社の文書による承諾なしに、無断で転載することは固くお断りいたします。
- 4. 本書に記載の内容は、将来予告なしに変更される場合があります。
- 5. 本書に記載の仕様は、お客様の環境、測定条件によって異なる結果が得られる場合があります。
- 6. 運用した結果の影響について、一切の責任を負いかねますのでご了承ください。
- 7. 本書に記載の「使用上のご注意」は、使用者や他者への危害と財産の損害を未然に防ぎ、安全に正しくお使いいただくための重要な注意事項です。ご使用になる前に必ずお読みください。
- 8. 本書に記載されている製品名および商品名は、各社の商標または登録商標です。



連絡先は 株式会社 Sohwa & Sophia Technologies のホームページでご確認ください。 URL > <u>http://www.ss-technologies.co.jp</u>



使用上の注意

	下記の注意を守らないと人が死亡する、または重傷を負う可能性があります。
1 強制	本製品に仕様で規定した範囲外の電源電圧を加えないでください。 範囲外ので電源電圧を加えると、破損・火災の恐れがあります。
设制	アース端子が付いているターゲットに使用する場合は、ターゲットや周辺機器のアースを確実に接続してくださ い。機器の故障や感電の恐れがあります。 また、ガス管にアース端子をつながないでください。火災や爆発の原因になります。
公 禁止	本製品に接続した機器を取り付けたまま持ち運ばないでください。 特にケーブルはプラグを持って抜き差ししてください。ケーブルが破損し、火災・感電の恐れがあります。
秋	ケーブルを取り扱う場合は次の点を守ってください。「傷つけない」「加工しない」「無理に曲げない」「ねじらない」「引っ張らない」「物を載せない」「加熱しない」「熱器具に近づけない」「濡れた手で触らない」。 これらを守らないと火災・感電の恐れがあります。 もしケーブルが破損した場合、そのケーブルの使用を中止してください。
公 禁止	雷が鳴りだしたら、電源プラグに触れないでください。感電の原因となります。 落雷により製品が破損したと思われる場合は、本製品の使用を中止してください。
会	ステープラの針、クリップなどの金属を内部に入れないでください。火災・故障の恐れがあります。
	直射日光の当たる場所、熱器具の近く、極端な高温環境、極端な低温環境、振動の激しいところ、金属 や油を含むほこりの多い場所、スパイク系のノイズが発生する場所で使用したり、放置しないでください。 また、強い衝撃を与えないでください。
分解禁止	分解・改造・修理しないでください。火災・感電の恐れがあります。
水濡れ禁止	風呂場やコップの近くなど、液体のある場所、湿気の多い場所では使用しないでください。 感電する恐れがあります。 液体が本製品内部に入った場合はすぐに電源を切り、使用を中止してください。
注意	通電中の本製品に長時間触れていると低温やけどになる恐れがあります。 また、本製品を布団などで覆った状態で使用しないでください。
プラグを抜く	もし、異常なにおい・異常な音・発煙・発火した場合、または落としたり、強い衝撃を与えたりして破損、破 損した恐れのある場合は、すぐに電源を切ってください。そのまま使うと重大な事故を起こす可能性があります ので、使用を中止してください。



略語・用語・記載ルール

本書で使用する略語・用語や記載ルールについて説明します。

- … 特に記載がない限り、数値はすべてプラスの値とします。 数値について ٠
- ··· 2¹⁰=1024 を表します。(例:16K=16384) K(大文字) •
- k(小文字)
 … 1000を表します。(例:1kHz=1000Hz)

 [xxxxx]
 … xxxxx というウィンドウタイトルを示します。

 <xxxxx>
 … xxxxx というウィンドウ内の項目名を示します。

 •
- •

本書で使用する注釈・注意点などについては Figure 1 の通りです。



Figure 1



略語・用語の解説は Table 1 の通りです。

略語・用語	説明
本製品	Universal Probe 本体・付属品を含むもの。
プローブ	Universal Probe 本体のこと。
本ソフトウェア	SPI Writer のこと。
フラッシュメモリ	Flash メモリ、EEPROM などの総称です。
Serial No.	Serial Number の略。プローブの ID のこと。
Software Code	各ソフトウェアの License Code を発行するために必要なライセンス。無償版は不要。
License Code	本機で使用できる機能を追加するためのコード。
モジュール	フラッシュメモリに書き込むデータのこと。オブジェクトデータとシンボルデータを含むもの。
オブジェクトデータ	実際にフラッシュメモリに書き込む、バイナリデータのこと。
シンボルデータ	変数名や関数名(=シンボル)とアドレスの対応関係を表すデータのこと。
ホスト PC	本ソフトウェアを動作させる PC のこと。
ターゲット	Universal Probeによって制御、計測する対象のこと。
スタンドアロン機能	ホスト PC と接続しなくても動作すること。(電源は必要)
(N/A)	Not Assigned の略。割り当てられた情報がないことを表す。
PC	単に PC と記載した場合はプログラムカウンタの略。

Table 1



1. 本ソフトウェアの概要

本ソフトウェアの概要を説明します。

本ソフトウェアは、General Connector に接続された「SPI フラッシュメモリ」へデータを書き込むソフトウェアです。 別ソフトウェアとして「ARM Writer」がありますが、機能・対応する構成の違いを Figure 2 に示します。



Figure 2



1.1. システム要件

本ソフトウェアを動作させるためには、以下のシステムが必要です。

- Microsoft Windows 7 以降が動作する PC
- CPU: 1GHz 以上 (使用する OS の要件に準拠します)
- Memory: 1GB 以上 (使用する OS の要件に準拠します)
- HDD:空き容量 500MB 以上
- OS: Windows 7 以降 (32bit または 64bit)
- USB 2.0 の空きポート1つ以上

1.2. 特徴

本ソフトウェアには、以下の特徴があります。

- プロジェクトファイルによる動作環境の保存と復元
- 書き込みデータのフォーマットを自動認識するダウンロード機能
- 複数のデータ書き込みをサポート
- バッチ機能により柔軟な書き込み処理を実現

1.2.1. SPI フラッシュメモリ操作機能

- 指定されたアドレス範囲へ特定データを書き込むフィルアップ機能
- 指定されたアドレス範囲の文字列やデータの検索するサーチ機能
- メモリデータのファイル保存機能



1.3. 電源の入れ方/切り方

1.3.1. ターゲットとの接続方法と電源の入れ方

以下の手順でプローブとターゲットを接続します。

- 1) ターゲットの電源が切れている事を確認します。
- 2) プローブの USB ケーブルを接続します。
- 3) ターゲットとプローブを接続します。
- 4) ターゲットの電源を入れます。
- 5) 本ソフトウェアを起動し、データの書き込み作業などを行います。

1.3.2. 電源の切り方とターゲットからの外し方

以下の手順でプローブとターゲットを切り離します。

- 1) 本ソフトウェアを終了します。
- 2) ターゲットの電源を切ります。
- 3) ターゲットからプローブを外します。

プローブのハードウェア仕様、ターゲットの制限、接続などの詳細については、『ハードウェア・ユーザーズ・マニュアル』を 参照してください。



2. 操作方法

本ソフトウェアの操作方法について説明します。

2.1. 作業フロー

本ソフトウェアを起動してから、ユーザーのモジュールを書き込み、プロジェクトファイルを保存するまでの、主な工程を Figure 3 に示します。ここに記載されていない操作については「2.操作方法」もしくは「4.メニュー」の項を参照ください。



Figure 3



2.2. 本ソフトウェアの起動

本ソフトウェアの起動には、いくつかの方法があります。

2.2.1. 起動方法

●起動方法 1

「デスクトップ」にある本ソフトウェアのアイコンをダブルクリックします。

●起動方法 2



Figure 4



●起動方法 3

エクスプローラなどで、SPIWriter.exe やプロジェクトファイル(.spiwpj)をダブルクリックします。

						- • ×
	▶ SP	I_Writer	▼ 47	SPI_Writer	の検索	٩
ファイル(E) 編集(E) 表示(<u>V</u>) ツール(<u>T</u>)		プ(<u>H</u>)				
整理 ▼ ライブラリに追加 ▼ 共有 ▼	書	き込む 新しいフォルダー				• 🔟 🔞
📔 NVIDIA	~	名前	更新日日	÷.	種類	サイズ
NVIDIA Corporation		test.bak	2014/0	9/26 21:15	BAK ファイル	5 KB
\mu nxp		o test.spiwpj	2014/0	9/26 21:17	UniversalProbe	5 KB
i openocd-0.8.0						
퉬 Program Files						
퉬 ProgramData						
\mu SPI_Writer						
System Volume Information						
🔐 temp	-					
2 個の項目					🌉 コンピューター	

Figure 5

●起動方法 4

「スタート」→「ファイル名を指定して実行」をクリックし、SPIWriter.exe やプロジェクトファイル(.spiwpj)を指定します。

▶ 検索結果の続きを表示	
SPIWriter.exe ×	シャットダウン 🕨
	🖉 🍐 🖺 😼
F : C	





以上の操作で下図のように、本ソフトウェアが起動します。

O Universal Probe	
ファイル(E) 表示(⊻) 実行(⊻) ヘルプ(且)	
b7*1	

Figure 7



2.3. プロジェクトファイルの選択

本ソフトウェアは以下の設定を保存するために、必ずプロジェクトファイル(.spiwpj)を使用します。

- 書き込むモジュールの情報
- 書き込み環境の設定状態

新規で書き込みを行うには、プロジェクトファイルを作成する必要があります。 既存のプロジェクトファイルを開くと、保存した状態を復元することができます。

2.3.1. プロジェクトファイルを新規作成する

例として、C:¥SPI_Writer フォルダに TEST フォルダを作成し、test.spiwpj というプロジェクトファイルを作成します。 以下のツールバーボタンまたは、メニューバーをクリックします。([プロジェクトの新規作成]ダイアログ詳細)



[プロジェクトの新規作成]ダイアログが開きます。

ר״ר״ר״ר		<u> </u>		
プローブ種類: 選択されていません				
ターケットCPU: 選択されていません				
ライセンス: 選択されていません				
位果 (D)・				
C¥		参照(<u>R</u>)		
	Figure 8	/	I	



プロジェクトファイルを保存するフォルダを作成するか、既存のフォルダを選択します。

①フォルダを作成する

参照 ボタンをクリックします。

[フォルダの選択]ダイアログが開きます。

フォルダの選択	x
フォルタ [☆] (<u>D</u>): c:¥spi_writer	ОК
C¥ ▲	キャンセル
SPI_Writer	ネットワーク
	フォルダ(<u>E</u>)
[^] ドライフ* (⊻) : ■ c: OS	

Figure 9

C:¥SPI_Writerを選択して、 フォルダ ボタンをクリックします。

[フォルダの作成]ダイアログが開きます。

フォルターの作成		×
現在のフォルダ(<u>F</u>):	C:¥SPI_WRITER	
作成するフォルダ(⊆):	TEST	
0	K キャンセル	

Figure 10

<作成するフォルダ>に"TEST"を入力して OK ボタンをクリックします。



フォルダ名に"スペース"、"/"文字は使用できません。



[フォルダの選択]ダイアログに戻ります。

フォルダの選択	x
フォルダ (<u>D</u>) : c:¥spi_writer¥test	ОК
🗁 c¥ 🔺	キャンセル
SPI_WRITER	ネットワーク
ドライブ(⊻): ■ c: OS	



②フォルダを選択する

作成した TEST フォルダを選択して、 OK ボタンをクリックします。

[プロジェクトの新規作成]ダイアログに戻ります。

③プロジェクト名を指定する

<プロジェクト名>に、プロジェクト名(test)を入力します。 このプロジェクト名がプロジェクトファイル名になり、ダイアログ下部にある<位置>ボックスに表示されます。

プロジェクトの新規作成 フ <u>゚ロジ</u> ェクト名 (<u>N</u>):	ок	
[test]	キャンセル	
- プローブ プローブ種類: 選択されていません	<u>プローブ(Ⴒ)</u>	
ターケット: 選択されていません		
ライセンス: 選択されていません		
位置 (L):]	
C¥SPI_WRITER¥TEST¥testspiwpj	参照(<u>R</u>)	$ \rightarrow $
Figure 12		



フォルダ名に"スペース"、"/"文字は使用できません。



2.3.2. プロジェクトファイルを開く

保存してあるプロジェクトファイルを開く場合は、以下の操作を行ってプロジェクトファイル(.spiwpj)を選択してください。





エクスプローラでプロジェクトファイルをダブルクリックし、プロジェクトファイルを開くこともできます。 また、デスクトップにある本ソフトウェアのアイコンヘドラッグ&ドロップしてプロジェクトファイルを開くこともできます。



2.4. プローブの選択

本ソフトウェアで使用するプローブの選択を行います。 [プロジェクトの新規作成]ダイアログの
フローブ
ボタンをクリックします。

[プローブ選択]ダイアログに、現在接続されているプローブの一覧が表示されます。

プローブ選択		— ×
シリアル番号	ステータス	
HM540000123 HM540000124	接続可能 使用中	
HM540000125	う化以未登録	更新(<u>R</u>)
		<u>ライセンス登録(L)</u>
		閉じる(0)

Figure 13

使用するプローブをクリックし、 接続 ボタンをクリックします。

シリアル番号	プローブに登録されているシリアル番号を表示します。			
ステータス	プローブの状態を表示します。			
	接続可能	:	本ソフトウェアに接続し、使用することができます。	
	使用中	:	他のアプリケーションで既に使用されています。	
			本ソフトウェアで使用することはできません。	
	ライセンス未登録	:	ライセンスが登録されていません。	
			ライセンス登録 ボタンを押し、ライセンスを登録することで、使	
			用可能になります。	
<u> </u>	選択されているプローブに接続します。			
1女小儿	ステータスが「接続可能」のプローブを選択時のみ有効です。			
	接続後はダイアログが閉じ、 [プロジェクトの新規作成]ダイアログに戻ります。			
更新	現在接続されているプローブを再検索し、プローブの一覧を更新します。			
ニノわい.7座谷	ライセンスを登録する為のダイアログが表示されます。			
リービンス豆琢	ステータスが【接続可能」もしくは「ライセンス未登録」のプローブを選択時に有効です。			
	→ 詳細は『2.5. License Code の登録』の章を参照ください。			
閉じる	このダイアログを閉じます。			

Table 2



2.5. License Code の登録

本ソフトウェアはライセンスシステムを採用しています。

License Codeの取得には、別売りの Software Code が必要です。

Software Code とプローブの Serial No.を弊社にご連絡いただくと、License Code を発行します。

①License Code の入力について

本ソフトウェア起動時、[プローブ選択]ダイアログに「ライセンス未登録」または、「接続可能」と表示されているものを選択し、 ^{ライセンス登録} ボタンをクリックすると、License Code の入力ウィンドウが表示されます。 弊社からお送りした License Code を入力していただくと、対象のプローブで本ソフトウェアを使用することができるようになります。

ライセンスの登録
ライセンスの登録が必要です。 License Code を入力してください。
Serial No: HM540000035
License Code:
※ 半角英数字 ※ 区切りのハイフン(-)の有無は問いません
OKキャンセル
F : 44

Figure 14

②License Code 入力時の注意事項

License Codeの入力は、お送りした情報通りに入力してください。



License Code を送付したメールの内容と一致しているにもかかわらず受け付けられない場合は、当社までお問い合わせください。



③エラー表示一覧

Table 3

表示	意味	対処方法
License Code が不正です。	License Code の書式が不正など、 License Code の解析ができない場合 に表示されます。	License Code が送られてきたメールを ご確認の上、再度 License Code を入 力してください。
License Code と Serial No が一致 しません。 License Code と 本体の Serial No の組み合わせをご確認ください。	ご使用のプローブの Serial Noと、 License Code が対象としているプロー ブの Serial No が一致していない場合に 表示されます。	License Code が送られてきたメールを ご確認の上、再度 License Code を入 力してください。



2.6. SPI フラッシュメモリ設定

本ソフトウェアで使用する SPI デバイスの選択または、SPI フラッシュメモリの設定ファイル(*.fsh)を読み込みます。 このダイアログでは、SPI フラッシュメモリへアクセスを行うために必要な設定を行います。

SPIフラッシュメモリ設定	23
┌ デンドイス!情幸履	
 (● デバイス選択(D) メーカー(M): 	
デバイス(<u>E</u>): ▼	
○ ファイル指定(E) ファイル名(N):	
- 1/O リファレンス電源	
● 外部供給(2) ○ 内部生成(1) 電圧(1): 1.2V ▼ ○ あり(2) ◎ なし(0)	
СОК	

Figure 15

Table 4				
デバイス選択	本ソフトウェアに登録されている SPI フラッシュメモリを操作の対象にする場合に選択します。			
メーカー	本ソフトウェアに登録されている SPI フラッシュメモリのメーカーをプルダウンリストから選択します。			
サイズ	本ソフトウェアに登録されている SPI フラッシュメモリのサイズをプルダウンリストから選択します。			
	既にメーカーが選択されている場合、プルダウンリストはそのメーカーのものにフィルタリングされています。			
	選択を解除する場合はプルダウンリスト先頭の空白を選択します。			
デバイス	本ソフトウェアに登録されている SPI フラッシュメモリのデバイスをプルダウンリストから選択します。			
	メーカー、サイズにより本ソフトウェアに登録されている SPI フラッシュメモリのデバイスが一意に決まる場			
	合は、自動で選択されます。選択を解除する場合はプルダウンリスト先頭の空白を選択します。			
ファイル指定	フラッシュメモリの設定ファイル(*.fsh)を指定する場合に選択します。			
	フラッシュメモリの設定ファイルは、メモリコマンドビルダーで作成するファイルです。			
ファイル名	読み込むフラッシュメモリの設定ファイル(*.fsh)を指定します。			

①デバイス情報



②I/Oリファレンス電源

Table 5			
外部供給	プローブの I/O 電源が Vtref 経由でターゲットから供給される場合に選択します。		
	デフォルトは ON です。		
内部生成	プローブの I/O 電源が Vtref 経由でターゲットから供給されない場合に選択します。		
	このとき、I/O 電源はプローブ内部で生成した電源を使用します。		
電圧	プローブ内部で生成する電源電圧をプルダウンリストの中から選択します。		
	このプルダウンリストは内部生成が選択されている場合に有効になります。デフォルトは 1.2V です。		
	ターゲット側のインターフェース電圧にあわせて設定してください。		
Vtref 電源出力 あり	本体から Vtref 経由でターゲットへ電源を供給します。		
	ターゲット側に電源が無い場合に選択します。		
Vtref 電源出力 なし	ターゲットに電源はありますが、プローブの I/O 電源が Vtref 経由でターゲットから供給されない場合に		
	選択します。		

→ 供給できる電圧値については、『ハードウェア・ユーザーズ・マニュアル』を参照してください。

③SPI フラッシュメモリの設定

OK ボタンをクリックすると、SPI フラッシュメモリへアクセスするためのリード/ライト用プログラムをプローブに登録します。

SPI フラッシュメモリのリード/ライト用プログラムを登録する際にエラーが発生した場合は、メッセージが表示されます。 本ソフトウェアの起動処理をを中止するか、そのまま起動処理を継続するか選択してください。 起動処理を継続しても SPI フラッシュメモリ設定は、プロジェクトファイル作成後に設定を変更できます。

エラーメッセージは、フラッシュメモリの設定ファイル(*・fsh)を手動で編集したり、フラッシュメモリの設定ファイルと本ソフトウェアのバージョンが異なる場合などに発生します。



2.7. プローブ環境の設定

[プローブ環境の設定]ダイアログでは、SPI フラッシュメモリに合わせてクロック周波数の設定をします。

プローブ環境の設定			- 23
- 知ック周波数の指定			
加かを選択してください		-	
低速加ックの周波数 (1 ~ 1000KHz)	500	KHz	
			_
		OK	

Figure 16

クロック周波数を指定して OK ボタンをクリックしてください。

Table 6

クロック周波数の指定	SPI フラッシュメモリの最大クロック周波数を選択します。	
	低速クロック指定を選択すると KHz 単位で指定できます。(最大 1000KHz)	



2.8. SPI フラッシュメモリのクリア

SPI フラッシュメモリにデータが書き込まれている場合、書き替え前に SPI フラッシュメモリのクリアが必要です。 以下のメニューを選択し、[SPI フラッシュメモリ設定]ダイアログを表示します。 既に[SPI フラッシュメモリ設定]ダイアログを表示している場合は、「クリア」タブを選択してください。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース $ ightarrow$ SPI フラッシュメモリ $ ightarrow$ クリア	

SPIフラッシュメモリ設定	
デバイス クリア フィルアッフ°	
	開始(<u>S</u>)
(● Ŧッフ*4レース(<u>C</u>)	
C ブロックイレース(<u>B</u>)	
ፖኑኄレス(<u>D</u>) :	
	<u>ОК</u> ++уъл

Figure 17

①クリア

Table 7

チップイレース	SPIフラッシュメモリ全てをクリアします。	
ブロックイレース	<アドレス>に入力したアドレスを含む SPI フラッシュメモリの 1 ブロックをクリアします。	

②開始ボタン

クリア方法を指定して 開始 ボタンを押すと、クリア処理を開始します。



クリア処理中は進行状況が表示されます。

チップ消去中です	
アドレス	0×0000000
経過時間	5秒
	\$**)171L
	142 6/



クリア処理が終了すると、操作結果が表示されます。

// コンント実行をします // 終了しました 	// フラッシュメモリクリアを実行します // Chip Erase : FMCLEAR CHIP ERASE : // コマンド実行をします // 終了しました 	
--------------------------------	---	--

Figure 19

指定範囲が正常クリアされたかは、ダンプウィンドウで確認します。

<u> </u>	1 0x0 -	0x0	0000	00ff	(0x1	00)														
7ドレス(<u>S</u>):	0×0				•)終	771	°レス()	E) [)×000)000f	f			-				
テ° <u>-</u> タ(<u>D</u>):	0×ff					6	範	囲長	(<u>L</u>)	0)×100)				•	28	□ 更發	新しな	ι\(<u>U</u>)
Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ASC	Π		* *
0×000000 0×000000 0×000000 0×000000 0×000000	00 ff 10 ff 20 ff 30 ff 30 ff 50 ff 50 ff 70 ff 80 ff 90 ff	ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff ff ff				
0×000000 0×000000 0×000000 0×000000	1a0 ff 160 ff 1c0 ff 1d0 ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff	ff ff ff ff				

Figure 20



2.9. ユーザーデータの書き込み

設定したフラッシュメモリにオブジェクトデータをダウンロードします。 以下のメニューを選択し、[ダウンロード設定]ダイアログを表示します。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース $ ightarrow$ ダウンロード	

[ダウンロード]ダイアログに書き込むモジュールの情報を設定し、ダウンロード」ボタンでをクリックします。 ダウンロード処理中は進行状況のダイアログを表示します。

書き込み中です	
ፖኮՆス	0×00011000
経過時間	0秒
予想残り時間	0秒
	<u>(キャンセル)</u>

Figure 21

<u>ダンプウィンドウ</u>を表示して、SPI フラッシュメモリにオブジェクトデータがダウンロードできているかを確認します。 ダウンロードできていない場合は、SPI フラッシュメモリの設定状況を再度確認してください。

(INFO)	うまく書き込めない場合は、 <mark>フラッシュメモリのクリア</mark> を実施しているかご確認ください。
	容量制限版でダウンロードできるオブジェクトデータのサイズは <u>1KByte</u> までです。



2.10. SPI フラッシュメモリのフィルアップ

設定した SPI フラッシュメモリに対してフィルアップ処理を行います。 以下のメニューを選択し、 [SPI フラッシュメモリ設定]ダイアログを表示します。 既に [SPI フラッシュメモリ設定]ダイアログを表示している場合は、「フィルアップ」タブを選択してください。

 ツールボタン
 メニューバーの操作
 ショートカットキー

 -- リソース → SPI フラッシュメモリ → フィルアップ
 --

iPIフラッシュメモリ設定 デバイス クリア フィルアップ	— ×
- フィルアッフ [*] 「範囲 開始アドレス(S): ● 終了アドレス(E): ● 範囲長(L):	開始(S) データ(1パイト)(<u>D</u>):
▼ フィルアップ前にチップイレースを自動実行する(A)	
	OK キャンセル

Figure 22

①フィルアップ

	Table 8
範囲	開始アドレス :SPI フラッシュメモリへのフィルアップを開始するアドレスを指定します。
	終了アドレス :SPI フラッシュメモリへのフィルアップを終了するアドレスを指定します。
	範囲長 :開始アドレスからの範囲を指定します。
フィルアップデータ	SPIフラッシュメモリに書き込むデータを指定します。
	指定できるデータサイズはバイトサイズのみです。
フィルアップ前にチップイレースを自	SPI フラッシュメモリヘフィルアップを行う前にチップイレースを自動で実行します。
動実行する	デフォルトは ON です。



②開始ボタン

フィルアップの方法を指定して 開始 ボタンを押すと、フィルアップ処理を開始します。 「フィルアップ前にチップイレースを自動実行する」にチェックが付いている場合は、フィルアップする前にチップイレースが実行されます。

フィルアップ中は進行状況を示すダイアログを表示します。



Figure 23

フィルアップ処理が終了すると、ダイアログ右下に処理結果が表示されます。



Figure 24

指定範囲が正常にフィルアップされたか、ダンプウィンドウで確認します。

ቃ*`ንプウィンドウ	2 0x400	0 - 0	×000	0004	ff (0	x100))										×	
アドレス(<u>S</u>):	0×400				•)終	<u> 7</u> 7ト	ั _้ มิิ่ม(<u>)</u>	e) [)×000	1004f	f			-		
データ(<u>D</u>):	0×99				X	/ @	節範	囲長	(L)	0)×100	1				- (\	I
Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	-	3
0×000004	00 99	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99		-
0×000004	10 99	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99		
0×000004	20 99	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99		
0×000004	30 99	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99	_	-
0×000004	40 99	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99		
0×000004	50 99	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99		
0×000004	60 <mark>99</mark>	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99	-	·
•	\mathbf{F}																	

Figure 25



2.11. プロジェクトファイルの保存

プロジェクトの設定状態をプロジェクトファイルへ保存することができます。 「プロジェクトの保存」を参照してください。



2.12. 本ソフトウェアの終了

本ソフトウェアを終了するには以下のメニューを選択するか、ウィンドウ右上隅の X ボタンをクリックします。

ツールボタン	メニューバーの操作	ショートカットキー
	ファイル → アプリケーションの終了	

[アプリケーションの終了]ダイアログが表示され、終了の前にプロジェクトを保存するかどうかを選択します。

アプリケーションの終了	×
終了前の操作 ・ アロジェクトに保存(<u>S</u>)	
○ プロジェクトに名前を付けて保存(<u>A</u>)	
○ プロジェクトlこ保存しない(<u>D</u>)	
OK キャンセル	

Figure 26

①プロジェクトに保存

作成したプロジェクトファイルを上書き保存して終了します。

②プロジェクトに名前をつけて保存

別のプロジェクトファイルとして保存して終了します。

③プロジェクトに保存しない

現在のプロジェクト設定状況を保存せずに終了します。



3. 本ソフトウェアでのデータ表現

本ソフトウェアでのデータ表現方法について説明します。

3.1. 数值表現

本ソフトウェアでは数値表現として2進数,10進数,16進数を扱う事ができます。

		Table 9
書式	基数	例
0x<数值>	16進数	0x12345678
H'<数值>	16進数	H'12345678
@<数值>	2 進数	@01011101
<数値>	10 進数	12346578

3.2. アドレス表現

本ソフトウェアでアドレスを表現する場合は、Table 9 で示す書式と、演算子を組み合わせたものになります。



3.3. データ表現

本ソフトウェアでデータを表現する場合は、Table 9 で示す書式と、演算子を組み合わせたものになります。



3.4. SPI フラッシュメモリの読み込み

コマンドウィンドウの ASSIGN またはドット(.)コマンドや、バッチプログラム内の条件式で SPI フラッシュメモリの内容を読み込むには、 以下のように指定します。

Table 10
意味
指定されたアドレスのバイトデータ
指定されたアドレスのワード(2 バイト)データ
指定されたアドレスのロングワード(4 バイト)データ

●例

通常のメモリ空間はアドレ	レスを入力するだけです。
.[0x4000].B	//0x4000 番地のバイトデータで参照する。
.[0x4000].W	//0x4000 番地のワードデータを参照する。
.\$A=[0x4000].L	//ワーク変数\$A へ 0x4000 番地の 4 バイトデータを代入する。
if([0x4000].W==0x1234)	//0x4000 番地のワードデータが 0x1234 の時、真。



4. メニュー

本ソフトウェアのメニューについて説明します。

4.1. ファイル

「ファイル」メニューではプロジェクトに関するファイル操作を行います。

4.1.1. 閉じる

現在フォーカスしている子ウィンドウを閉じます。 子ウィンドウが無い場合は選択できません。

4.1.2. プロジェクトの新規作成

プロジェクトの新規作成を行います。 初めて使用する場合は、下記ツールボタンもしくはメニューを選択し、プロジェクトを作成してください。 プロジェクトはプロジェクトファイル(.spiwpj)に保存されます。

ツールボタン	メニューバーの操作		ショートカットキー
₽ <u></u>	ファイル → プロジェクトの新規作成		Ctrl + P
	「プロジェクトの新規作成		
	フロジェクト名(<u>N</u>): test	OK	

-プローブ		
プローブ種類: 選択されていません	<u>[</u>]	
ターゲット: 選択されていません		
ライセンス: 選択されていません		
立置 (L):		
12直(L): C¥SPI WRTTER¥test sniwni		

Figure 27


-			~	-
	h		- 1	
d	()		- I.	
5	\sim	· C	-	·

プロジェクト名	プロジェクト名を指定します。
プローブ	プローブ種類 : プローブの種類
	ターゲット : SPI Writer と表示されます。
	ライセンス : ライセンスによる使用制限が表示されます。
	Size-limited \rightarrow 1KByte の書き込み容量制限版のライセンスです。
	Available \rightarrow 1KByte の書き込み容量制限の無いライセンスです。
	[プローブの選択]ダイアログが表示され、
→□ [→] →(⁺)	現在接続されているプローブ一覧が表示されます。
位置	フルパスでセーブされるプロジェクトファイル名が表示されます。
参照	[フォルダの選択]ダイアログが表示され、プロジェクトファイルをセーブするフォルダを選択します。
ОК	プロジェクトファイルを作成します。
キャンセル	プロジェクトファイルの作成を中止します。



4.1.3. プロジェクトを開く

保存してあるプロジェクトファイルを開きます。前回の作業終了時の設定が復元されます。



4.1.4. プロジェクトの保存

現在設定されている各種情報をプロジェクトファイルへ上書き保存します。



4.1.5. プロジェクトを名前をつけて保存

現在設定されている各種情報を別のプロジェクトファイルへ保存します。

ツールボタン	メニューバーの操作	ショートカットキー
	ファイル → プロジェクトを名前をつけて保存	Ctrl + A



4.1.6. プロジェクトを閉じる

現在開いているプロジェクトを閉じます。 閉じる前に[アプリケーションの終了]ダイアログが表示されます。

ツールボタン	メニューバーの操作	ショートカットキー
	ファイル → プロジェクトを閉じる	

4.1.7. アプリケーションの終了

本ソフトウェアを終了します。 詳細は「**本ソフトウェアの終了**」を参照してください。



4.2. リソース

「リソース」メニューでは SPI フラッシュメモリへのリード、ライトおよび、各種設定を行います。

4.2.1. ダウンロード

[ダウンロード設定]ダイアログでは SPI フラッシュメモリに書き込むモジュールを設定し、書き込み指示を行います。 書き込むモジュールは、複数指定する事ができます。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース → ダウンロード	



Figure 28



L-	h	10 1	12
Ы	()	ie –	

モジュールリスト	ダウンロードするモジュールのリストです。
	Ctrl や Shift キーを押しながら選択すると、複数のファイルを選択できます。
フォーマット	ダウンロードするモジュールのオブジェクトフォーマットを選択します。
	デフォルトでは自動選択が選択されます。
	複数のモジュールを選択している場合、この項目はそれぞれのモジュールごとに指定する必
	要があります。
オブジェクトデータをダウンロード	オブジェクトデータをダウンロードする場合にチェックします。この項目はそれぞれのモジュールご
	とに指定する必要があります。
シンボル情報をダウンロード	本ソフトウェアでは使用できません。
次回起動時に再ダウンロード	プロジェクトファイルを開いた時に、自動的にモジュールをダウンロードします。
モジュール情報	モジュールのパス、ファイルサイズ、タイムスタンプが表示されます。
治	ダウンロードするモジュールを追加します。
	[ファイルを開く]ダイアログが表示され、その中で追加するモジュールを選択します。
	追加されたモジュールはモジュールリストに表示されます。
削除	エジュールリフトで選択されているエジュールをリフトから削除します
	してエールリストで医いてれているとフェールをリストルクトリ际しより。
なウンロード	選択されたモジュールをダウンロードします。
	<オブジェクトデータをダウンロード>の項目にチェックを入れておく必要があります。
	モジュールリストで複数のモジュールが選択されている場合は、選択されたモジュール全てをダ
	ウンロードします。
ダウンロード前にチップイレースを自	このチェックボックスが ON の場合は、ダウンロード前に自動でチップイレースを行います。
動実行する	チップイレースはダウンロードの操作1回につき1度だけ実行されます。
	複数のモジュールを選択してダウンロードする場合でもチップイレースは1度だけしか実行さ
	れません。



<ベリファイを有効にする>を設定することにより、本ソフトウェアからのメモリライトが正しく書き込まれたかチェックを行うことが出来ます。標準設定では、ベリファイは無効になっています。



容量制限版でダウンロードできるオブジェクトデータのサイズは <u>1KByte</u>までです。



4.2.2. アップロード

アップロードは、SPI フラッシュメモリ上のデータを読み出し、各種フォーマットに変換してファイルに保存します。 以下のメニューを選択すると[アップロード]のダイアログが開きます。 アップロードをするには[アップロード]ダイアログを表示して、ファイルに保存したいメモリ範囲とフォーマットを指定します。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース $ ightarrow$ アップロード	

797° 0-1×*		—
フォーマット (M): ファイル名 (E): アップロード範囲の設定 開始アドレス (A): ○ 終了アドレス (E): ☞ レングス (L):	<mark>ハイナリテシータ</mark> ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	アッブロード(<u>U)</u> 閉じる

Figure 29



		Table 13			
フォーマット	ファイルフォーマットは、以下の中から選択します。				
	フォーマット	メモリ範囲	PC アドレスの 指定	自動判別	備考
	インテル標準 HEX	0~64K バイト	可能	可能	
	インテル拡張 HEX	0~1M バイト	可能	可能	セグメント値が付 加されます。
	インテル 32 ビット HEX	0~4G バイト	可能	可能	
	モトローラ S 型 HEX(S1-S9)	0~64 <i>K</i> バイト	可能	可能	
	モトローラS型 HEX(S2-S8)	0~16M バイト	可能	可能	
	モトローラ S 型 HEX(S3 – S 7)	0~4G バイト	可能	可能	
	高速ダウンロード (SHF)	0~4G バイト	不可能	可能	当社独自のフ ォ ーマットです。
	バイナリ	0~4G /バイト	不可能	不可能	バイナリデータとし てアップロードしま す。
	(j) 自動判別の 動判別でき	D欄の「可能」とは、本 きるかどうかを指します。	シノフトウェアでダウン	シロードを行う	祭、フォーマットを自
ファイル名	保存するファイル名を指定	定します。			
アップロード範囲の指定	開始アドレス	:開始アドレスを指定	言します。		
	終了アドレス	:終了アドレスを指定	言します。		
	範囲長	:終了アドレス可範[囲長のどちらかを選	観沢して指定し	ます。
アップロード	SPI フラッシュメモリのデー	ータを読み出し、指定	したファイルに保存	します。	



4.2.3. プローブ環境の設定

プローブの設定を行います。

以下のメニューを選択すると[プローブ環境の設定]のダイアログが開きます。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース → プローブ環境の設定	

プローブ環境の設定	×
クロック スイッチ ホペーリング	
「加ック周波数の指定」	- 1
1.25MHz	
低速加ックの周波数 (1 ~ 1000KHz) 500 🗧 KHz	
OK キャンセル	

Figure 30



①クロック

SPI フラッシュメモリの最大クロック周波数を設定します。

プローブ環境の設定		×
クロック スイッチ ホーリング		1
- ケロック周波数の指定		
<u>1.25MHz</u> 低速加ックの周波数(1 ~ 1000KHz)	500	KHz
	OK	キャンセル

Figure 31

Table 14

クロック周波数の指定

SPI フラッシュメモリの最大クロック周波数を選択します。 低速クロック指定を選択すると KHz 単位で指定できます。(最大 1000KHz)



②スイッチ

エンディアンを設定します。

プローブ環境の設定		×
クロック スイッチ ホーリング		
- エンディアンの選択		
C ビッグエンディアン(<u>B</u>)	 ジトルエンディアン(L) 	
		1154
	UK 47	ゼル

Figure 32

Table 15		
エンディアンの選択	ビッグエンディアン:[ダンプウィンドウ]の表示などをビッグエンディアンで行います。	
	リトルエンディアン:[ダンプウィンドウ]の表示などをリトルエンディアンで行います。	



デフォルトはリトルエンディアンです。



① ポーリング

SPI フラッシュメモリの状態を監視するためのポーリング間隔を設定します。

プローブ環境の設定	X
クロック スイッチ ホペーリング	1
-ホ℃-リンケ間隔の設定	
間隔(<u>E</u>):	
□ 既定値とする(<u>S</u>)	
	デフォルト値(こ戻す(<u>R</u>)
	既定値に戻す(工)
	OK キャンセル

Figure 33

т.	I - 1	-	1	\sim
ıа	n	e		h
	~ .	-	-	~

間隔 ポーリング間隔を 100ms ごとに設定にします。	
	100msから4,294,967,200msまでが有効範囲です。
既定値とする	指定した値を既定値とし、以後、新規作成したプロジェクトはこの値になります。
デフォルト値に戻す	出荷時の値(500ms)に戻します。
既定値に戻す	既定値として設定した値に戻します。



4.2.4. バッチファイルの自動実行設定

バッチプログラムを自動実行するタイミングを指定します。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース → バッチファイルの自動実行設定	



Figure 34

Ta	b	e	1	7
iu		C	-	. /

起動時	本ソフトウェア起動時(プロジェクトの内容を読み込む前)にバッチファイルを実行します。	
(プロジェクトの読み込み前)	これにより、各種設定及びウィンドウの情報などを読み込む前に、必要な処理をバッチで実	
	行でき、不正なメモリなどのアクセスを回避できます。	
	バッチファイル名 : {プロジェクト名}.sta	
起動時	本ソフトウェア起動時(プロジェクトの内容を読み込んだ後)にバッチファイルを実行します。	
(プロジェクトの読み込み後)	バッチファイル名 : {プロジェクト名}.fst	

ツールバーのボタンを押すことにより以下の名前のバッチファイルを実行します。 バッチファイル名 : {プロジェクト名}.wrt

ツールボタン	メニューバーの操作	ショートカットキー
The		

バッチファイルはプロジェクトファイルと同じフォルダに保存してください。

 $\langle ! \rangle$



4.2.5. メモリサーチ

指定範囲のメモリ内を検索し、指定データと一致する、または一致しないデータのアドレスを表示します。 検索するデータは、数値もしくはアスキー文字列が指定できます。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース → メモリサーチ	

メモリサーチ	×
┌アドレス範囲の設定――	サーチ結果
開始アドレス(S) 0×8000 💌	開始しました 0×00008021
○ 終了アドレス(E) 📃	0×00008022 0×00008035
● 範囲長(<u>R</u>) 0×100	0×00008036 0×00008038
	0x0000809e 0x000080dd
サーチデ ~ タの設定	0x000080de 終了しました
サーチテ゚ータ(<u>D</u>) ⁰ ×ff	
┌データサイズの設定―――	
ເ€ バイト ○ ロンヴワード	
O ワード O クワッドワード(<u>Q</u>)	
開始 閉じる	

Figure 35

①アドレス範囲の設定

メモリチェックするアドレス範囲を設定します。

Table 18

開始アドレス	開始アドレスを指定します。
終了アドレス	終了アドレスを設定します。
範囲長	範囲長を指定します。

②サーチデータの設定

Table 19

サーチデータ	サーチするデータを指定します。
	不一致データを指定する場合は、データの先頭へエクスクラメーションマーク(!)を付けます。
	文字列の不一致条件は指定できません。
	数値での指定 : 0x34, 128, !0x56 など
	アスキー文字列 : "abcdef", "ghijk" など
データサイズの設定	サーチするデータのサイズを指定します。

Universal Probe ソフトウェア・ユーザーズ・マニュアル - SPI Writer



③サーチ結果

メモリサーチで検索されたアドレスを表示します。 実行結果は、コマンドウィンドウにも表示されます。

ባረንት. እሳላንት እሳ	
<pre>// メモリ検索を実行します // (0×00008000 - 0×000080ff 0×ff) SEARCH 0×00008000 TO 0×000080ff ASIZE // 0×00008021 // 0×00008035 // 0×00008035 // 0×00008036 // 0×00008098 // 0×0000809d // 0×0000809d // 0×000080dd // 0×000080de // 終了しました</pre>	MODE MATCH BYTE 0×ff
LOG-OFF シンボ [*] ルに値を代入	

Figure 36



4.2.6. SPI フラッシュメモリ





4.3. 実行

「実行」メニューには、以下のコマンドがあります。

4.3.1. キャンセル

ホストPCとプローブまたは、ターゲット間の通信が不能になった場合、本コマンドをを実行する事で通信不能状態から復帰できる場合があります。しかし、その後は正しく制御できないことが多いため、復帰できた場合は、一旦プロジェクトファイルを保存し、すべてを 再起動してください。

ツールボタン	メニューバーの操作	ショートカットキー
	実行 → キャンセル	

4.3.2. ベリファイ(ライト時に比較)

このチェックをして書き込みを実行すると、書き込み後にベリファイを行います。

ツールボタン	メニューバーの操作	ショートカットキー
Vfy	実行 → ベリファイ(ライト時に比較)	

4.3.3. ベリファイオンリー(ライトせずに比較)

このチェックをして書き込みを実行すると、書き込みは行わずにベリファイのみを行います。

ツールボタン	メニューバーの操作	ショートカットキー
Vfy only	実行 → ベリファイオンリー(ライトせずに比較)	



4.3.4. チェックサム値の計算

SPI フラッシュメモリに正しくモジュールが書き込めたかを、書き込みモジュールのチェックサムと SPI フラッシュメモリのデータのチェックサムを計算して比較することができます。

チェックサムの計算		—
-短押		範囲
書さえるみ ~ ッのテェックりなし	(E):	
	,	, 計算(<u>A</u>)
L		
		(MCSIC)

Figure 37

Table 20

書き込みデータのチェックサム	SPI フラッシュメモリに書き込むデータのチェックサムを表示します。
	書き込み手順が記録されていない場合は、何も表示されません。
ターゲット上のデータのチェックサム	SPI フラッシュメモリに書き込まれているデータのチェックサムを表示します。
	書き込み手順が記録されていない場合は、何も表示されません。
範囲	SPI フラッシュメモリに書き込むデータのアドレス範囲を表示します。
	書き込み手順が記録されていない場合、"None"を表示します。
	SPI フラッシュメモリに書き込むデータと、SPI フラッシュメモリに書き込まれているデータのチェッ
<u> </u>	クサムを計算します。書き込み手順が記録されていない場合、"None"を表示します。



本機能を使用するには予めプローブに書き込み手順を記録しておく必要があります。 書き込み手順の記録については「9.スタンドアロン機能」を参照してください。



4.4. SPI フラッシュメモリ

4.4.1. SPI フラッシュメモリ設定ダイアログ

SPI フラッシュメモリ関連の操作を行うダイアログです。 ダイアログ左上部のタブを切り替えることで、以下の3つの操作を行うことができます。

ツールボタン メニューバーの操作 ショートカットキー -- リソース → SPI フラッシュメモリ → デバイス設定 --

ダイアログ上部のタブを切り替えることで、以下の3つの操作を行うことができます。

- 1) SPI フラッシュデバイスの変更
- 2) SPI フラッシュメモリのクリア
- 3) SPI フラッシュメモリのフィルアップ

4.4.2. デバイスタブ

PIフラッシュメモリ設定		×
デバイス クリア フィルアップ		
_ デバイス情報		
 デバイス選択(<u>D</u>) 		変更(<u>C</u>) 適用(<u>A</u>)
メーカー(<u>M</u>):	•	サイス(S): 🗨
デバイス(<u>E</u>) :		•
○ ファイル指定(E)		
7ァイル名(N):		
		- Vtref 電源出力
(● 外部供給(X) (○ 内部主成())	`€)±(⊑): 1.2V 💌	C 551(Y) C 52L(Q)
		OK ++>セル
	51 20	
	Figure 38	



①デバイス情報

本ソフトウェアの起動時に設定した内容が表示されています。 このダイアログの起動直後はデバイス情報を変更することはできません。 デバイス情報を変更する場合は、 変更 ボタンを押してからデバイス情報の変更を行う必要があります。

	Table 21
変更	SPI デバイス情報を変更する場合にこのボタンをクリックして変更を行います。
	変更中は、キャンセルボタンになります。
適用	SPI デバイス情報の変更中に有効になります。
	このボタンをクリックするとプローブのリード/ライト用プログラムを再登録します。
デバイス選択	本ソフトウェアに登録されている SPI フラッシュメモリを使用する場合に選択します。
メーカー	本ソフトウェアに登録されている SPI フラッシュメモリのメーカーをプルダウンリストから選択します。
サイズ	本ソフトウェアに登録されている SPI フラッシュメモリのサイズをプルダウンリストから選択します。
	既にメーカーが選択されている場合、プルダウンリストはそのメーカーのものにフィルタリングされています。
	選択を解除する場合はプルダウンリスト先頭の空白を選択します。
デバイス	本ソフトウェアに登録されている SPI フラッシュメモリのデバイスをプルダウンリストから選択します。
	メーカー、サイズにより本ソフトウェアに登録されている SPI フラッシュメモリのデバイスが一意に決まる場
	合は、自動で選択されます。選択を解除する場合はプルダウンリスト先頭の空白を選択します。
ファイル指定	フラッシュメモリの設定ファイル(*.fsh)を指定する場合に選択します。
	フラッシュメモリの設定ファイルは、メモリコマンドビルダーで作成するファイルです。
ファイル名	読み込むフラッシュメモリの設定ファイル(*.fsh)を指定します。

②I/O リファレンス電源

Table 22

外部供給	プローブの I/O 電源が Vtref 経由でターゲットから供給される場合に選択します。
	デフォルトは ON です。
内部生成	プローブの I/O 電源 Vtref 経由でターゲットから供給されない場合に選択します。
	このとき、I/O 電源はプローブ内部で生成した電源を使用します。
電圧	プローブ内部で生成する電源電圧をプルダウンリストの中から選択します。
	このプルダウンリストは内部生成が選択されている場合に有効になります。デフォルトは 1.2V です。
	ターゲット側のインターフェース電圧にあわせて設定してください。
Vtref 電源出力 あり	本体から Vtref 経由でターゲットへ電源を供給します。
	ターゲット側に電源が無い場合に選択します。
Vtref 電源出力 なし	ターゲットに電源はありますが、プローブの I/O 電源が Vtref 経由でターゲットから供給されない場合に
	選択します。

→ 供給できる電圧値については、『ハードウェア・ユーザーズ・マニュアル』を参照してください。



I/O リファレンス電源の設定はダイアログの OK ボタンをクリックしたときに本体に設定されます。



4.4.3. クリアタブ

→ 詳細は「2.8. SPI フラッシュメモリのクリア」を参照してください。

4.4.4. フィルアップタブ

→ 詳細は「2.10. SPI フラッシュメモリのフィルアップ」を参照してください。



5. ウィンドウ

本ソフトウェアのウィンドウについて説明します。

5.1. ウィンドウのドッキングについて

本ソフトウェアの子ウィンドウは、メインウィンドウの上下左右のいずれかの辺に張り付けて位置を固定したり、メインウィンドウの外側に出して配置したりすることができます。

0 Un 771⊮(niversal I E) 表示	Probe - ₹(<u>V</u>) IJ	SPI \ Y-7(B	Write ₹) j	er [S 美行(ize-l <u>G</u>)	imit १४४	ed] ((M25 <u>N)</u>	5P32 1117°	V)- (<u>日</u>)	test	t.spi	wpj												-		×
B	* *		IEM	CM		<u>I</u>		E		fy	Vfy nly																	
] ダ ^ッ ンプ ウィ	(ንኑ* 91 ()x0 -	0x00	0000	fff (0	0×10	00)															-		×			
7	ドレス(<u>S</u>):	0×0				-		終	<u> </u>	レス(<u>E</u>		×000	100fff				-											
Ŧ	፦ <u>\</u> אַיַיי	0×01					0	範	囲長(D	0:	×100	0				- Q	<mark>8</mark> L	更新	Utal	۱ <u>(U)</u>							
Ac	dress	+() +1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F				ASCII				×			
	<000000 <000000	000 0	20 3 30	21 33	21 33 20	20 33	20 32	20 33	20 35	21 33	20 31 40	21 33	20 33	21 33	21 34	21	20 33				. !! 30333 22202	3235	! ! 3133 9597	111 3433 2000	-			
0×	<000000	020 33	5 33 5 46	33	30 34	33	40	36 0d	43 0a	30 53	46 33	30	37	30	40 38	30	44 30			Ē	33302 3F743	3D	5315	2E6D				
0× 0×	<0000000 <0000000	040-30 050-30) 30) 30	30 30	30 38	37 33	38 35	30 30	45 37	30 30	30 30	32 30	30 38	32 33	31 39	30 30	31 37				00007 00083	'80E 3507	0020 0008	2101 3907				
0× 0×	<000000 <000000	060 30 070 30) 30) 30	30 31	38 30	38 33	36 44	0d 30	0a 37	53 30	33 30	31 30	35 38	30 34	38 31	30 30	30 37				00088 00103	36 2007	S315 0008	0800 4107				
0×	0000000 200000	080 30 nan 20) 30) 20	30 30	38 20	34 45	35 22	08 07	37 05	30 52	30 30	30 21	38 25	30 30	30 20	30 30	30 20				000 00/0⊩	., 1	0008 ९२१४	0000	J			
]	_						_	_		_					_			4	_	_	•				
י ז״אלקב	ንብንት" ን		-	-	-	-	_	_	_	-	_	_	_	_	-	_				-		-	-		-		-	
				-																								_
LOG-	OFF 🦻	ンホ"ル(こ値	重を代	<u>, Х</u>																								_
<	< >	»	A	ISSI	GN			BAT	TCH				CD			(CLOSE			CLS	3		D	IR				
νī*1																											ΓΓ	

Figure 39



5.1.1. ウィンドウの状態

①MDI 子ウィンドウ (通常のウィンドウ)

本ソフトウェアのメインウィンドウの内で自由に移動・伸縮が可能なウィンドウです。 各ウィンドウはこの状態で開きます。

O Univers 771⊮(E)	sal Probe - : 表示(⊻) リン 【記】	SPI Wri V-7(<u>R</u>) EM	ter [実行 MD	[Size 7(<u>G</u>)	-lim 94 RTE	ited] ントック) (м: (<u>W</u>) 	25P3 ^// Vfy	32V) 7°(<u>F</u>	- te <u>1</u>)	st.s	piwp	j							
	園 ダンプウイ アドレス(<u>S</u>):	>>**71 0: 0×0	×0 -	0x00	0000	lfff (0 _▼)×10)00))終	7		D [0)×000	IOOffi	f			<u>-</u>			
	⁷ ~4(D): Address 0×000000 0×000000 0×000000 0×000000 0×000000	0×01 +0 00 01 10 33 30 30 30 30 30 30 30 30 3	+1 20 30 33 46 30 30	+2 21 33 37 30 30	+3 21 33 30 34 30 38	+4 20 33 32 33 37 33	+5 20 32 45 44 38 35	* 範 +6 20 33 36 0d 30 30	囲長 20 35 43 0a 45 37	(L) +8 21 33 36 53 30 30 30	+9 20 31 46 33 30 30	+A 21 33 36 31 32 30	+B 20 33 37 35 30 38	+C 21 33 32 30 32 33	+D 21 34 45 38 31 39	+E 21 33 36 30 30 30	✓ Q +F 20 33 44 30 31 37 ▲	● 更新 ASCII • • !! 3033323 33302E6 6F743D. 0000780 0008350	L 1 111 531333433 C6F672E6D .S3150800 E00202101 700083907	
V7*1																				

Figure 40

MDI 子ウィンドウは、メインウィンドウの外に出ることができません。





②ドッキングウィンドウ

メインウィンドウの上下左右のいずれかの辺に張り付いた状態のウィンドウです。(下図) ドッキング状態でも、ウィンドウの伸縮は可能です。また、ドッキングした状態から他の辺に付け替えることもできます。

S Universal Probe - SPI Writer [Size-limited] (M25P32V) - test.spiwpj ファイル(E)表示(Y) リソ-λ(B) 実行(G) タインドウ(W) ヘルプ(出)							
ዎ"ኦፓ° ንብኦት ን1 0x0 - 0x00000fff ((0×1000)						
7ドレス(<u>S</u>): 0×0	● 終了アドレス(E) 0×00000fff ●						
データ(<u>D</u>): 0×01	○ 範囲長(L) 0×1000 □ 更新しない(U)						
Address +0 +1 +	+4 +5 +6 +7 +8 +9 +A +B +C +D +E +F ASCII						
0×0000000 01 20 21 21	20 20 20 20 21 20 21 20 21 21 21 20 . !! !!!!	-					

Figure 42

③フローティングウィンドウ

フローティング状態の時、子ウィンドウはメインウィンドウの外側に出ることができます。 ただし、本ソフトウェアを終了、またはプロジェクトを閉じたときは、フローティング状態のウィンドウも自動的に閉じます。

フローティングウィンドウをメインウ 出す。	ィンドウの外に	ŧ	
<u>ፃ"ን7" ንለንኑ" ን1 0x0 - 0x00000fff</u> 7Ի℃ג(<u>S</u>): 0x0 テ [×] - <u>\$(D</u>): 0x01	ox1000) ▼ ○ 終了アド ○ 範囲長(!		
Address +0 +1 +2 +3 0x00000000 01 20 21 21 0x00000010 33 30 33 33 •	+4 +5 +6 + 20 20 20 2 + 33 32 33 +	Ĭ	

Figure 43



5.1.2. 子ウィンドウ状態の変更方法

子ウィンドウの状態を変更するには、変更したい子ウィンドウのタイトルバーを右クリックします。 すると下図のようなコンテキストメニューが表示されますので、変更したい状態やドッキングしたい位置を指定します。

9">7" 94>1" 9	75					×
ፖԻՆス(<u>S</u>): テ ^ĸ -妳(<u>D</u>):	0×20000000		○ 終了)● 範囲	✓	Docked Floating MDI Child	ĺ
Address	+0 +1	+2 +3 +4	+5 +6		Docked to MDI Child as	•
4	▶ ∢					v

Figure 44

以下に、メニューの図と各項目の説明をします。

Table 23





メニュー項目		- 説明
Docked		対象の子ウィンドウをドッキング状態にします。
		ドッキングする辺は、下辺→左辺→右辺→下辺→…の順に切り替わります。
Floating		対象の子ウィンドウをフローティング状態にします。
		対象の子ウィンドウを MDI 子ウィンドウ状態にします。
MDI Child		子ウィンドウがフローティング状態で、メインウィンドウの外側にあったときは、自動的にメイン
		ウィンドウの内側に移動します。
	Тор	対象の子ウィンドウをドッキング状態にし、メインウィンドウの上辺に張り付けます。
Deckedte	Left	対象の子ウィンドウをドッキング状態にし、メインウィンドウの左辺に張り付けます。
Docked to	Bottom	対象の子ウィンドウをドッキング状態にし、メインウィンドウの下辺に張り付けます。
	Right	対象の子ウィンドウをドッキング状態にし、メインウィンドウの右辺に張り付けます。
	Minimized	対象の子ウィンドウを MDI 子ウィンドウ状態にし、最小化(アイコンの状態に)します。
MDI Child as	Maximized	対象の子ウィンドウを MDI 子ウィンドウ状態にし、最大化します。
	Restored	対象の子ウィンドウを MDI 子ウィンドウ状態にし、普通のサイズで表示します。

Table 24



5.1.3. ドラッグ&ドロップによるドッキング状態の変更方法

①ドッキングしている辺を変更する

子ウィンドウがドッキング状態、またはフローティング状態のときは、「5.1.2.子ウィンドウ状態の変更方法」の他に、その子ウィンドウのタイトルバーをドラッグ&ドロップすることによって、ドッキングさせる辺を変更することができます。 子ウィンドウのタイトルバーをドラッグし、ドッキングしたいメインウィンドウの辺へ移動させると枠の形状が変化しますので、ドロップする と位置が変わります。

ダンプ 9424 75 アトシス(S) 0×20000000

Figure 45

🕓 Universal Probe - SPI Writer [Size-limited] (M25P32V) - test.spiwp	pj 🗕 🗖 💌
ファイル(<u>E)</u> 表示(<u>V</u>) リソース(<u>R</u>) 実行(<u>G</u>) ウィンドウ(<u>W</u>) ヘルプ(<u>H</u>)	
ダンプウインドウ1 0x0 - 0x00000fff (0x1000)	
7ドレス(S): 0x0 🗸 C 終了7ドレス(E) 0x0000	
テ ^ッ -妳(<u>D</u>): 0×01 (* 範囲長(<u>L</u>) 00	
Address +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 + 芸	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	
0x000000f0 30 30 30 30 30 31 30 37 30 30 3 0x000000f0 30 30 30 38 39 36 0d 0a 53 33 3	
ντ [*] 1	

Figure 46



②ドッキング状態とフローティング状態の変更

ドッキング状態の子ウィンドウをメインウィンドウの外側または中央付近にドラッグすると、フローティング状態に変化します。 フローティング状態の子ウィンドウをメインウィンドウの辺の近くヘドラッグ&ドロップすると、ドッキング状態に変化します。



Figure 47

③MDI 子ウィンドウへの変更

ドラッグ&ドロップ操作によって、ドッキング状態またはフローティング状態の子ウィンドウを MDI 子ウィンドウに変更することはできません。MDI 子ウィンドウ状態に変更したい場合は、子ウィンドウのタイトルバーを右クリックして表示されるコンテキストメニューから 「MDI Child」を選択してください。



5.1.4. 同じ辺に複数の子ウィンドウをドッキングさせた場合

メインウィンドウの同じ辺に複数の子ウィンドウをドッキングさせた場合、表示方法として以下の2種類の状態があります。

- 1) 既にドッキングしている子ウィンドウよりさらに内側にドッキングした状態
- 2) メインウィンドウの一辺(ドッキングしている辺)に複数の子ウィンドウがドッキングした状態

以下の図は、(1)の例です。

Ouriversal Probe - SPI Writer [Size-limited] (M25P32V) - test.spiwpj 271//(F) 表示(V) リリース(R) 実行(G) フィンドウ(W) ヘルプ(H)	
ר״אלאני	8
」 LOG-OFF「シンボルに値を代入	
ASSIGN BATCH CD CLOSE CLS DIR I	
タッンプ・フィンド・ウ1 0×0 - 0×00000fff (0×1000)	X
7トゼス(S): 0×0 ・ C 終了アドゼス(E) 0×00000fff ・	
データ(<u>D</u>): 0x01 (・範囲長(<u>L</u>) 0x1000 ▼ Q C 更新しない(<u>U</u>)	
Address +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F ASCII	* *
	_
0x00000000 33 33 33 33 32 33 32 33 35 33 31 33 33 33 34 33 33 34 33 33 30 303332351333433	
0x00000030 36 46 37 34 33 44 0d 0a 53 33 31 35 30 38 30 30 6F743D. S3150800	_
	▼
ν	

Figure 48



以下の図は、(2)の例です。

O Universal Probe - SPI Writer [Size-limited] (M25P32V) - test.spiwpj 27ℓW(F) 表示(V) リソース(R) 実行(G) 242ト***********************************	- • •
ም እር በ በ በ በ በ በ በ በ በ በ በ በ በ በ በ በ በ በ	
7トジス(S): 0×0 ・	
^{〒[×]-∽(D): 0×01}	
Address +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +	
0x00000000 33 33 33 30 32 45 36 43 36 46 36 37 32 4!	
0x00000030 36 46 37 34 33 44 0d 0a 53 33 31 35 30 3	
ASSIGN BATCH	CD
μ̄ [*] ، ۲	

Figure 49

5.1.5. ドッキングウィンドウのサイズ変更

ドッキング状態の子ウィンドウのサイズ変更は、マウスカーソルで子ウィンドウ周辺の枠をドラッグすることで行います。

 終了アドルス(E) 0×200000ff 範囲長(L) 0×100 +5 +6 +7 +8 +9 +A +{ 	図 JTンド 9インド 9 // フラッシュメモリクリアを実行しま BLANK CHECK NOT SUPPORT // 終了しました // フラッシュメモリフィルを実行しま // (R;0×08000000 - 0× BREAK LOG-OFF > Figure 50
	Figure 50



5.2. ダンプウィンドウ

メモリ内容をダンプして表示するウィンドウです。ダンプウィンドウは複数開くことができ、以下の特徴があります。

- 指定された範囲のメモリデータを表示します。
- 各種フォーマットでの表示が可能です。
- 符号付き/符号なし整数での1バイト、2バイト、4バイト表示が可能です。
- 16 進数での1バイト、2バイト、4バイト表示が可能です。
- 実数での4バイト、8バイト、10バイト、12バイト表示が可能です。
- 読みだした際に変更のあったデータは赤色に変わるため、データの変化が容易に判別できます。
- 表示された内容を CSV 形式やテキスト形式で出力することができます。



アドレスビュー、データビュー、文字列ビューの各表示領域は、境界線をマウスでドラッグすることによりサイズを調整することができます。

■ ダンプワィ アドレス(S): データ(<u>D</u>):	0x0 0xff	- 0x00000	Offf (0×10	000) 0 終了アト • 範囲長	*レス(<u>E</u>) (∐	0×00 0×10	000fff 000			- - -	9 = -	更新し	ta (U)		
Address 0x000000 0x000000 0x000000 0x000000 0x000000	+0 100 110 110 110 110 110 110 111 <td>+2 +3 f ff ff f ff ff</td> <td>+4 +5 ff</td> <td>+6 +7 ff ff ff ff</td> <td>+8 + ff f ff ff f ff ff ff</td> <td>9 +A f ff f ff f ff f ff f ff f ff f ff f</td> <td>+B ff ff <td>C +D f ff f ff f ff f ff f ff f ff f ff f</td><td>+E ff ff ff ff ff ff ff ff ff ff</td><td>#f ff ff</td><td></td><td></td><td></td><td></td><td></td></td>	+2 +3 f ff ff f ff ff	+4 +5 ff	+6 +7 ff ff ff ff	+8 + ff f ff ff f ff ff	9 +A f ff f ff f ff f ff f ff f ff f ff f	+B ff ff <td>C +D f ff f ff f ff f ff f ff f ff f ff f</td> <td>+E ff ff ff ff ff ff ff ff ff ff</td> <td>#f ff ff</td> <td></td> <td></td> <td></td> <td></td> <td></td>	C +D f ff f ff f ff f ff f ff f ff f ff f	+E ff ff ff ff ff ff ff ff ff ff	#f ff ff					
	Ţ	'ドレスビュ	_		F	Figure	データt e 51	<u></u>					文字列	<u> </u>	



Table 25

アドレス	表示開始アドレスを指定します。
終了アドレス, 範囲長	終了アドレス、または、表示させる範囲を指定します。
	数値を入力後、リターンキーを押すと指定範囲のメモリ内容をを表示します。
0	<アドレス>-<終了アドレス>or<範囲長>で指定した範囲を、再読み込みします。再読
(表示ボタン)	み込みした際に、前回の表示値から値が変化した場合は、データビューの数値が赤文字で
	表示されます。
Q	[アドレスフォーマット]、[データフォーマット]、[アクセスサイズ]を変更するためのコンテキストメ
🧕 (コンテキストメニュー表示)	ニューが表示されます。
□ 更新しない(U)	このチェックをしておくと、表示ボタンで再読み込みしても表示内容を更新しません。
データ	カーソルがある位置のデータを表示します。



5.2.1. ダンプウィンドウのコンテキストメニュー

データが表示されている領域で、右クリックをするとコンテキストメニューが表示され、表示形式の選択やデータのコピーができます。

アトッレスフォーマット ► デ゛ータフォーマット ► ファイルへ出力する データをコピー(C) 文字列をコピ-(S) アドレスとデータをコピ-(D) アトッレスとデータと文字列をコピ-(A) データと文字列をコピー(F)

Figure 52

Ta	bl	е	26
10	\sim	~	20

アドレスフォーマット	表示アドレスのフォーマットを変更します。
	物理アドレス表示
データフォーマット	メモリデータの表示フォーマットを変更します。
	符号付き1バイト整数
	符号付き2バイト整数
	符号付き4バイト整数
	符号付き8バイト整数
	符号なし1バイト整数
	符号なし2バイト整数
	符号なし4バイト整数
	符号なし8バイト整数
	1 バイト 16 進数
	2 バイト 16 進数
	4 バイト 16 進数
	8 バイト 16 進数
	4 バイト実数
	8 バイト実数
	10 バイト実数
	12 バイト実数
ファイルへ出力する	[ファイルへ出力]ダイアログが表示されます。出力形式は CSV 形式とテキスト形式があ
	り、CSV 形式の場合はオプションの指定が可能です。
データをコピー	データビューで、範囲選択領域にある文字列をクリップボードにコピーします。
	※キーボードから Ctrl + C を入力してもコピーできます。



文字列をコピー	文字列ビューの範囲選択領域にある文字列をクリップボードにコピーします。
	文字列ビューが表示されている時だけ有効になります。
アドレスとデータをコピー	アドレスビューとデータビューの範囲選択領域にある文字列をクリップボードにコピーします。
アドレスとデータと文字列をコピー	アドレスビューとデータビューと文字列ビューの範囲選択領域にある文字列をクリップボード
	にコピーします。
	文字列ビューが表示されている時だけ有効になります。
データと文字列をコピー	データビューと文字列ビューの範囲選択領域にある文字列をクリップボードにコピーします。
	文字列ビューが表示されている時だけ有効になります。

①[ファイルへ出力]ダイアログ

77114へ出力		×
ファイル名(E):	参照	OK
- 出力形式		キャンセル
- オプジョン ▼ 16進数値の前に"0×"を付加する		

Figure 53

Table 27

ファイル名	指定したファイル名で保存します。(CSV 形式を選択された場合、拡張子の".CSV"を付けてください)。
出力形式	CSV 形式、テキスト形式のいずれかの選択が出来ます。
オプション	数値書式を選択します。



②領域選択の方法

マウスもしくはキーボードで表示された値を選択することができます。

<u></u> 9~>7°94>1	*°91 0:	×0 -	0x00	0000	offf (0x10	00)													• 💌
アドレス(S):	0×0				•)終	ፖፖኑ	*レス()	E) [0)×000	100fff	f			-				
テ°ータ(<u>D</u>): [0×ff						範	囲長	Û	0	0×100	10				-	2 🥑 🗆 更新	新しない(U)	
Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	ASCII			A V
0×0000000	0 ff	ff	ff	ff	ff ff	ff	ff	ff	ff ff	ff	ff	ff	ff	ff	ff	ff				^
0x00000002	0 ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff				
0×0000003	0 ff	ff ff	ff ff	ff ff	ff ff	ff ff	ff ff	ff ff	ff ff	ff ff	ff ff	ff ff	ff ff	ff ff	ff ff	ff ff				
0×0000005	0 ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff		ff	ff	ff	ff				
0×0000006 0×0000007	0 ff 0 ff	ff ff	ff ff	ff ff	ff ff	†† f f	ff ff	ff ff	ff ff	ff ff	ff ff	†† f f	†† f f	†† f f	ff ff	ff ff				
0×0000008	0 ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff				.
		++	++	++	++	++	++	++	++	++	++	++	++	++	++	++				►

Figure 54

マウスから領域を選択する	データビューでドラッグ&ドロップすると、その範囲を選択します。
キーボードから領域を選択する	データビューで Shift キーを押しながらカーソルを移動させると、カーソルを移動した範囲を選
	択します。
領域選択を解除する	データビューでマウスを左クリックするか、キーボード操作でカーソルを移動すると選択状態を
	解除します。



③クリップボードへのコピー

範囲選択し、コンテキストメニューから「〜コピー」を選択すると、選択した範囲をクリップボードにコピーします。 キーボードから「Ctrl + C」を入力したときは、「データをコピー」の動作になります。

	Table 29
データをコピー	6b f7 1b d6 71 cc 77 4c b1 35 71 46 c3 e2 31 28 b2 54 63 5c 12 ca d3 f3 03 f0 9f ad 36 9e 43 3f d0 2d 47 ef b1 1d f4 80 43 0e 83 64 47 c4 3c 0a b1 1e 11 4e
文字列をコピー	k · ∃ q フ wL ア 5qF テ · (亻Tc¥. ハモ · ▪ ュ 6 曚?ミ−G · . ▪ C. ヅ G ト<. 7 N
アドレスとデータをコピー	R;0x00a205e0 6b f7 1b d6 71 cc 77 4c b1 35 71 46 R;0x00a205f0 c3 e2 31 28 b2 54 63 5c 12 ca d3 f3 03 f0 9f ad R;0x00a20600 36 9e 43 3f d0 2d 47 ef b1 1d f4 80 43 0e 83 64 R;0x00a20610 47 c4 3c 0a b1 1e 11 4e
アドレスとデータと文字列をコピー	R;0x00a205e0 6b f7 1b d6 71 cc 77 4c b1 35 71 46 k・ヨ q 7 wL 7 5qF R;0x00a205f0 c3 e2 31 28 b2 54 63 5c 12 ca d3 f3 03 f0 9f ad テ・(イ Tc¥. ^モ・・ı R;0x00a20600 36 9e 43 3f d0 2d 47 ef b1 1d f4 80 43 0e 83 64 6 曚?ミ-G・. ・ C. ヅ R;0x00a20610 47 c4 3c 0a b1 1e 11 4e G ト<. 7 N ※右端で折り返しています
データと文字列をコピー	6b f7 1b d6 71 cc 77 4c b1 35 71 46 k ⋅ ∃ q 7 wL 7 5qF c3 e2 31 28 b2 54 63 5c 12 ca d3 f3 03 f0 9f ad $\bar{\tau} \cdot (4 \text{ Tc}¥. ^{+} \bullet 1)$ 36 9e 43 3f d0 2d 47 ef b1 1d f4 80 43 0e 83 64 6 曚?ミーG · . • C. ヅ 47 c4 3c 0a b1 1e 11 4e G ト<. 7 N



5.2.2. ダンプウィンドウの複数表示

下記ツールボタンをクリックする度に、新規のダンプウィンドウを開くことができます。

ツールボタン	メニューバーの操作	ショートカットキー
MEM	表示 → ダンプウィンドウ	



Figure 55



同じアドレス範囲のウィンドウを 2 つ表示しておき、片方のダンプウィンドウに<更新しない>のチェックマークを入れておく と、メモリの変更内容を見比べることができます。


5.3. コマンドウィンドウ

コマンドライン入力を行うウィンドウです。以下の特徴があります。

- ・ キーボードやマウスでの連想選択方式によって、素早いコマンド操作が可能です。
- コマンド実行結果の表示とコマンドのロギング機能があります。
- バッチ機能による自動実行が可能です。
- バッチファイル作成機能があります。

下記ツールボタンをクリックすることで、コマンドウィンドウを開くことができます。

ツールボタン	メニューバーの操作	ショートカットキー
CMD	表示 → コマンドウィンドウ	

JRント*94ント*9					
>cd c.yspi WRITER					
				جح 🤇	テータスバー
LUG-UFF T 10/10/22					
<pre></pre>	CLOSE	CLS	DIR	DUMP	

Figure 56

→ コマンドの入力方法は『7. コマンドラインインターフェース』を参照ください。

ステータスバーにはコマンドラインインターフェースのガイドが表示されます。

クリップボードにコピーした文字列をバッチファイルとして実行することができます。コマンド入力部で右クリックし、コンテキストメニューを開いて「バッチを貼り付け」を選択するか、コマンド入力部にカーソルを合わせた状態で「Ctrl + B」を入力します。



6. キーボード

一般的な Windows キーボード操作以外の、キーボード操作について説明します。

6.1. ダンプウィンドウ

ダンプウィンドウでは以下の様なキーボード操作ができます。

□ ↑ □ ↓ □ ↓ □ ↓ □ → □ … データビュー、文字列ビュー内のカーソルを移動します。

6.2. コマンドウィンドウ

コマンドウィンドウでは以下の様なキーボード操作ができます。

□ 1 □ ↓ □ ↓ □ ↓ □ ↓ □ □ マンド名やパラメータ名の表示を切り替えたり、過去に入力したパラメータを参照します。

スペース … コマンドやパラメータを確定します。

ESC … コマンド入力中の場合は、コマンドやパラメータを1つキャンセルします。 コマンドが入力されていない場合は強制ブレークします。

リターン … コマンドを確定して実行します。省略されたパラメータは前回入力されたものが設定されます。



7. コマンドラインインターフェース

コマンドウィンドウのコマンド入力・操作方法について説明します。 コマンド入力は Figure 57 のコマンドラインで行います。

ָרָאָלאָר װיאָלאָר		
>cd C:¥SPI WRITER		
┃ ┃OG-OFF 「ディレクトリの変更	CT6464	
<< > >> CD	CLOSE CLS DIR	DUMP

Figure 57

7.1. コマンドラインでの操作方法

コマンド名やパラメータ名を覚えていなくても、最初の数文字を入力するだけで該当するコマンド名やパラメータ名が表示され、絞り 込んでいく事でコマンド入力を簡単に行うことができます。

7.1.1. コマンドの入力方法

コマンドラインをクリックし、フォーカスを移動します。 コマンドを入力して、リターンキーを押すことでコマンドを実行します。 パラメータは省略することができるものがあり、省略した場合は前回入力した内容を適用します。

7.2.2. 連想選択方式によるコマンド入力



7.2.3. 履歴から入力する

コマンドラインで 1011 キーを押すと、コマンドヒストリを表示することができます。 ヒストリから実行したいコマンドを選択し、リターンキーを押すことで過去のコマンドを再実行することができます。



コマンドラインへの全角カナの入力はできません。 また、パスを指定のする際、スペースの入力は受け付けないため、スペースを含むフォルダ名は使用できません。



7.2. コマンド解説

7.2.1. ASSIGN とドット(.)(式の評価)

パラメータで指定された式を評価します。 このコマンドでは、データの参照ができます。

●書式



Figure 58

●<式>

評価する式を指定します。

●例

Table 30

.[0x4000].B	0x4000 番地の1バイトデータを参照する。
.[0x4000].W	0x4000 番地の 2 バイトデータを参照する。
.[0x4000].L	0x4000 番地の 4 バイトデータを参照する

→ 詳細は「3.2. アドレス表現」、「3.4. SPI フラッシュメモリの」を参照ください。



ドットを使った場合は、コマンドウィンドウのヒストリには残りません。



7.2.2. BATCH(バッチプログラムの実行)

指定されたバッチプログラムを実行します。 バッチプログラムの詳細は、[バッチ機能]を参照してください。

●書式



Figure 59

● <バッチファイル名>

バッチプログラムファイル名を指定します。

●<引数>

バッチプログラムへ渡す引数を指定します。 各引数はスペースで区切ります。 また各引数は、バッチプログラム内の\$1~\$9 に設定されます。 \$0 はコマンド全体を示します。 10 個目以降の引数は無視します。



7.2.3. CD(フォルダの変更、ドライブの変更、カレントパス表示)

指定されたパスへ作業フォルダを移動します。 フォルダ名を省略した場合は、現在の作業フォルダが表示されます。

●書式



Figure 60

● <パス名>

変更するパス名を指定します。

●例

 $\langle ! \rangle$

[フォルダ名を指定した時の表示] 移動した後の作業フォルダを表示します。

JRンド [®] りィント [®] り	
>CD C:¥SPI_Writer	
C:#SFI_Writer	

Figure 61

[フォルダ名を省略した時の表示] 現在の作業フォルダを表示します。





7.2.4. CLOSE(プロジェクトファイルを閉じる)

現在開いているプロジェクトファイルを閉じます。 コマンドを実行すると、確認のダイアログが表示されます。 確認のダイアログを表示せずに終了する場合は <u>EXIT コマンド</u>を使用します。

●書式



Figure 63



7.2.5. CLS(コマンドウィンドウのクリア)

現在開いているコマンドウィンドウをクリアします。

●書式



Figure 64



7.2.6. DIR(フォルダ内容の参照)

指定されたパスの内容を参照します。 パス名を省略した場合は、現在の作業フォルダの全ファイルの内容が表示されます。 このコマンドは、DOSのDIRコマンドと同じです。

●書式



Figure 65

● <パス名>

参照するパス名を指定します。

●例

<u>วสวหะอัสวหะอั</u>							X
>dir test TEST~1	bak SPI	<dir> <dir></dir></dir>	4770 4770	2014-09-26 2014-09-26 2014-09-29 2014-09-29	21:15 21:15 09:44 09:49	test.bak test.spiwpj	

Figure 66



7.2.7. DUMP(メモリダンプ)

指定されたアドレス範囲のメモリデータを指定された表示フォーマットで表示します。 このコマンドは、ダンプウィンドウでの操作と同じです。

●書式



Figure 67

● <開始アドレスアドレス範囲値>

Table 31

開始アドレス	メモリダンプを行う開始アドレスを指定します。
範囲長	開始アドレスからの長さをバイト数で指定します。
終了アドレス	終了アドレスを指定します。



●<モード>

Table 32

● <表示フォーマット>

	Table 33
HEX	16 進数で表示します。
DEC	10 進数で表示します。
UDEC	符号なし 10 進数で表示します。
REAL	実数で表示します。

● <表示サイズ>

Table 34

BYTE	バイトサイズで表示します。	
WORD	ワードサイズで表示します。	
LWORD	ロングワードサイズで表示します。	
4BYTE	4 バイト実数形式で表示します。	例) -1.073726E+008
8BYTE	8 バイト実数形式で表示します。	例) -9.255963134931E+061
10BYTE	10 バイト実数形式で表示します。	例) -4.7798665708109333850E+0986
12BYTE	12 バイト実数形式で表示します。	例) -4.761268151994454601E+0986



7.2.8. ENV(プローブ環境設定)

各種のプローブ環境の設定を行います。 コマンド実行後は、現在の設定状態が表示されます。 このコマンドは、以下のメニューと同じ機能です。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース → <u>プローブ環境設定</u>	

●書式



Figure 68

JTAGCLOCK

SPIフラッシュメモリの最大クロックの周波数を指定します。

Table 35

AUTO	自動設定します。
SLOWCLOCK	次のパラメータで具体的な周波数を設定します。入力する数値の単位は KHz で数値のみ入力します。
周波数選択	1~1000の数値を指定します。

ENDIAN

エンディアンを指定します。

Tab	le	36
100		20

BIG	ビッグエンディアンに設定します		
LITTLE	リトルエンディアンに設定します		



7.2.9. ERROR_ECHO(エラーメッセージ表示の設定)

エラーメッセージの表示方法を指定します。

●書式



Figure 69

●<設定>

Table 37

OFF	エラーメッセージをメッセージボックスで表示します。
ON	エラーメッセージをコマンドウィンドウに表示します。



7.2.10. EXIT(本ソフトウェアの終了)

本ソフトウェアを終了します。 このコマンドは以下の操作と同じ機能です。

ツールボタン	メニューバーの操作	ショートカットキー
	ファイル → アプリケーションの終了	

●書式



Figure 70

●<モード>

	Table 38
SAVE	現在のプロジェクトファイルに上書き保存して終了します。
SAVEAS	別名のプロジェクトファイルに保存し、終了します。
NOTSAVE	プロジェクトファイルを保存せずに終了します。

●<ファイル名>

保存するファイル名を入力します。ファイル名のみの場合は現在の作業ディレクトリに保存します。



7.2.11. FM(SPI フラッシュメモリの設定)

SPI フラッシュメモリを設定します。 このコマンドは、以下のメニューと同じ機能です。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース → SPI フラッシュメモリ → デバイス設定	

●書式



Figure 71



●<種類>

Table 39

DEVICE	本ソフトウェアに登録されている SPI フラッシュメモリのデバイス名を指定します。
FILE	フラッシュ設定ファイルをフルパスで指定します。

● <デバイス名>

本ソフトウェアに登録されている SPI フラッシュメモリのデバイス名を指定します。

●<ファイル名>

フラッシュ設定ファイルをフルパスで指定します。 パスにスペースが含まれる場合は、ダブルクォーテーション(")で囲って入力してください。

●<I/O リファレンス電源>

Table 40

EXTERNAL	プローブの I/O 電源が Vtref 経由でターゲットから供給される場合に選択します。
INTERNAL	プローブの I/O 電源が Vtref 経由でターゲットから供給されない場合に選択します。

● <電圧>

プローブ内部で生成する電源電圧を設定します。 1.2V~5.0Vまでが選択可能です。

● <Vtref 電源出力>

Table 41

ENABLE	本体から Vtref 経由でターゲットへ電源を供給します。ターゲット側に電源が無い場合に指定します。
DISABLE	ターゲットに電源はありますが、プローブの I/O 電源が Vtref 経由でターゲットから供給されない場合に選択
	します。



7.2.12. FMCLEAR(SPI フラッシュメモリのクリア)

フラッシュメモリの内容をクリアします。 このコマンドは、以下のメニューと同じ機能です。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース $ ightarrow$ SPI フラッシュメモリ $ ightarrow$ クリア	

●書式



Figure 72

●<モード>

クリアモードを指定します。

Table 42

ALL	チップイレースを行います。
BLOCK	<アドレス>に入力したアドレスを含む SPI フラッシュメモリの 1 ブロックをクリアします。



7.2.13. FMFILL(SPI フラッシュメモリのフィルアップ)

指定したアドレス範囲の SPI フラッシュメモリに、指定したデータをフィルアップします。 コマンド実行後は、実行結果が表示されます。 このコマンドは、以下のメニューと同じ機能です。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース $ ightarrow$ SPI フラッシュメモリ $ ightarrow$ フィルアップ	

●書式





● <アドレス範囲>

Tab	e	43
-----	---	----

開始アドレス	メモリフィルを行う開始アドレスを指定します。
範囲長	開始アドレスからの長さをバイト数で指定します。
終了アドレス	終了アドレスを指定します。

● <フィルデータ>

ここで指定したデータでフラッシュメモリをフィルアップします。

● <チップイレース>

	Table 44
ENABLE	フィルアップする前にチップイレースを実行します。
DISABLE	フィルアップする前にチップイレースを実行しません。。



7.2.14. LOAD(オブジェクトデータのロード)

指定されたファイルのオブジェクトデータをダウンロードします。 リロードの場合は、指定されたダウンロードリストのファイルを再ダウンロードします。 削除の場合は、指定されたリスト番号のファイルをダウンロードリストから削除します。 このコマンドは、以下のメニューと同じ機能です。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース → <u>ダウンロード</u>	



●書式

Figure 74

●<モード>

ダウンロードのモードを指定します。

Table 45

OBJECT	指定されたオブジェクトデータファイルをダウンロードします。	
RELOAD	指定された番号のファイルから再ダウンロードします。	
DELETE	指定された番号のファイルをダウンロードリストから削除します。	

●<ファイル名>

ダウンロードするオブジェクトデータファイル名を指定します。



●<番号>

モジュールリストに登録されている1から始まる番号を指定します。 QUERY コマンドの LOAD パラメータにて、番号を確認することができます。

● <フォーマット>

ダウンロードするファイルのフォーマット名を指定します。 通常は AUTO を指定してください。 指定できるフォーマット名は以下のとおりです。

フォーマット名	概要
AUTO	ファイルフォーマットを自動認識します。
COFF	COFF 形式を指定します。
ELF	ELF 形式を指定します。
IHEX	インテルヘキサ形式を指定します。
MHEX	モトローラへキサ形式を指定します。
SHF	オリジナル高速ダウンロード形式を指定します。
SAUF	SAUF 形式を指定します。
BINARY <開始アドレス>	バイナリデータとして、<開始アドレス>からダウンロードします。

Table 46

● <チップイレース>

	Table 47
ENABLE	ダウンロードする前にチップイレースを実行します。
DISABLE	ダウンロードする前にチップイレースを実行しません。。



本ソフトウェアでは、OBJECT に対応しているものが使用可能です。



容量制限版でダウンロードできるオブジェクトデータのサイズは <u>1KByte</u>までです。



7.2.15. LOADPARAM(LOAD コマンドのパラメータ補足)

LOAD コマンドのパラメータを補足します。

●書式



Figure 75

●<モード>

CLEAR	LOAD コマンドのパラメータ補足データを初期化します。
INQUIRY	本ソフトウェアでは使用できません。
LOADLOCAL	本ソフトウェアでは使用できません。
OFFSET	オブジェクトデータのオフセット値を設定します。ロードするモジュールがバイナリの場合は、LOAD コマンドのオ
	フセットが使用されます。<オフセット値>のデフォルトは0です。



7.2.16. LOG(コマンドウィンドウのロギング制御)

コマンドウィンドウの表示内容をロギングする/しない(ファイルに保存する/しない)を設定します。 ロギング機能は、ON または ADD を入力した後からの表示内容がファイルに保存されます。

●書式



Figure 76

●<モード>

記録モードを指定します。

Table 49

ON	新しくファイルを作成してロギングを開始します。
ADD	既存のファイルに追加してロギングを開始します。
OFF	ロギングを終了します。

●<ファイル名>

ログを保存するファイル名を指定します。



NEWBATCH コマンドと併用する場合、同一のファイルを指定しないでください。



7.2.17. MKDIR(フォルダの作成)

指定されたパスのフォルダを作成します。 このコマンドは、DOSのMKDIRコマンドと同じです。

●書式



Figure 77

● <パス名>

作成するフォルダのパスを指定します。



7.2.18. NEWBATCH(バッチファイルの作成)

バッチファイルを新規作成または既存のファイルに追加します。 このコマンドでバッチファイルの作成が開始された後のコマンドは、パラメータのチェックはされますが、実際のコマンドは実行されず、コ マンドラインの文字列がファイルに保存されます。 OFF は、バッチファイルの作成を終了します。

●書式



Figure 78

●<モード>

記録モードを指定します。

Table 50

ON	新規にバッチファイルを作成し、コマンドの記録を開始します。
ADD	既存のバッチファイルに追加してコマンドの記録を開始します。
OFF	バッチファイルの作成を終了します。

●<ファイル名>

作成または追加するバッチファイル名を指定します。



LOG コマンドまたは SAVEWIN コマンドと併用する場合、同一のファイルを指定しないでください。



7.2.19. OPTION(コマンドウィンドウのオプション設定)

コマンドウィンドウでのオプションを設定します。 MORE を ON した場合は、各コマンドの実行結果が1 画面を越える時に一時停止します。 MORE を OFF した場合は、停止せずにコマンド結果を全て表示します。

●書式



Table 51

●<モード>

コマンド実行時の MORE 機能を指定します。

Table 52

ENABLE	表示結果が1画面を越える時、一時停止します。
DISABLE	表示結果が1画面を越える時、停止せずにスクロールします。

●例

>OPTION MORE DISABLE [DISABLE] MORE CONTROL



7.2.20. QUERY(各種設定状態の参照)

各種の設定状態を参照します。

●書式



Table 53

●<モード>

参照する種類を指定します。

Table 54		
各種モード	説明	
ENV	現在のプローブ環境設定状態を参照します。	
FM	SPI フラッシュメモリのデバイス情報を参照します。	
LOAD	現在登録されているダウンロードリストを参照します。	
LOADPARAM	LOADPARAM コマンドの設定を参照します。	
RADIX	入力基数の設定を参照します。	
OPTION	現在のコマンドウィンドウのオプションの設定状態を参照します。	



7.2.21. RADIX(入力基数の設定)

入力基数の設定を行います。 出力の基数には影響を与えません。

●書式



Figure 79

●<基数>

入力基数の10進数/16進数を指定します。

Tahl	۵	55
Iau	E	\mathcal{S}

DECIMAL	入力の基数を 10 進数にします。
HEX	入力の基数を 16 進数にします。
	この基数のとき、10 進数を入力することはできません。 10 進数を扱いたい場合は、RADIX コマンドで 10
	進数に切り替えてください。また、その時、16 進数の数値の先頭には"0x"を付けるようにしてください。



7.2.22. SAVEWIN(コマンドウィンドウのファイル出力)

現在開いているコマンドウィンドウ履歴をファイルに出力します。 LOGコマンドと異なり、既に実行された表示内容が保存対象です。

●書式



Figure 80

●<モード>

NEW	新規にファイルを作成し保存します。
ADD	既存のファイルに追記保存します。

●<ファイル名>

保存するファイル名を指定します。

●例

>SAVEWIN NEW C:¥ARM_Writer¥test.log Log New <C:¥ARM_Writer¥test.log>

>SAVEWIN ADD C:¥ARM_Writer¥test.log Log Add <C:¥ARM_Writer¥test.log>



NEWBATCH コマンドと併用する場合、同一のファイルを指定しないでください。



7.2.23. SEARCH(メモリサーチ)

指定されたメモリ範囲から、指定されたデータを検索します。 コマンド実行後は、実行結果が表示されます。 このコマンドは、以下のメニューと同じ機能です。

ツールボタン	メニューバーの操作	ショートカットキー
	リソース → <mark>メモリサーチ</mark>	

●書式



Figure 81

● <アドレス範囲>

Table 57

開始アドレス	メモリサーチを行う開始アドレスを指定します
範囲長	開始アドレスからの長さをバイト数で指定します
終了アドレス	終了アドレスを指定します



● <サーチデータ>

メモリサーチするデータを指定します。 文字を検索する場合は、シングルクォーテーション(')で囲みます。 文字列を検索する場合は、ダブルクォーテーション(")で囲みます。 なお、文字列の中にスペースやタブを直接入力する事はできませんが、(¥x20)や(¥x09)と言った形式で指定できます。

● <表示サイズ>

メモリサーチのサイズを指定します。 検索データに文字列を指定した場合は、指定したサイズが無効となり文字列のサイズで検索を行います。

Table 58		
BYTE	バイトデータでメモリサーチします。	
WORD	ワードデータでメモリサーチします。	
LWORD	ロングワードデータでメモリサーチします。	

● <サーチ条件>

検索条件を指定します。

MATCHの検索は1バイト分アドレスを進めながら検索します。 NOMATCHの検索は<開始アドレス>からサーチデータサイズごとに検索を行います。

Table 59			
MATCH	サーチデータと一致するデータを検索します。		
NOMATCH	サーチデータと一致しないデータを検索します。		
	検索データに文字列を指定できません。指定を行った場合は、データ文字列の先頭1文字データを有効と		
	して検索を行います。		



7.2.24. SHELLEXE(ファイルの実行)

指定されたファイルを実行します。

●書式



Figure 82

● <パス名>

実行するファイルパス名を指定します。

●例

>SHELLEXE C:¥ARM_Writer¥test.exe



7.2.25. UPLOAD(オブジェクトデータのアップロード)

指定されたオブジェクトデータをアップロードします。 このコマンドは、以下のメニューと同じ機能です。



●書式



Figure 83

●<ファイル名>

アップロードするファイル名を指定します。

● <アドレス範囲>

Table 60		
開始アドレス	アップロードする開始アドレスを指定します。	
長さ	開始アドレスからの長さをバイト数で指定します。	
終了アドレス	終了アドレスを指定します。	

● <出力フォーマット>

出力フォーマットを指定します。

Table 61

IHEX64K	インテル HEX 64K で出力します。
IHEX1M	インテル HEX 1M で出力します。
IHEX4G	インテル HEX 4G で出力します。
MHEX64K	モトローラ HEX 64K で出力します。
MHEX16M	モトローラ HEX 1M で出力します。
MHEX4G	モトローラ HEX 4G で出力します。
SHF	ソフィア高速ダウンロード形式で出力します。
BINARY	バイナリ形式で出力します。



7.2.26. VERIFY(ベリファイの設定)

ベリファイの設定/解除を行います。

このコマンドは、以下のメニューと同じ機能です。

ツールボタン	メニューバーの操作	ショートカットキー
Vfy	実行 → <u>ベリファイ(ライト時に比較)</u>	
Vfy only	実行 → <u>ベリファイオンリー(ライトせずに比較)</u>	

●書式



Figure 84

●<設定>

ベリファイの設定/解除を指定します。

Table 62

ON	ベリファイの設定を「ベリファイ(ライト後に比較)」か「ベリファイオンリー(ライトせずに比較)」にします。
OFF	ベリファイの設定を「無効」にします。

● <メモリライト>

「ベリファイ」か「ベリファイオンリー」かを指定します。

Table 63

MEMORY_WRITE ENABLE	「ベリファイ(ライト後に比較)」にします。
MEMORY_WRITE DISABLE	「ベリファイオンリー(ライトせずに比較)」にします。



>VERIFY ON	MEMORY_WRITE DISABLE
[ON] VERIFY
[DISABLE] VERIFY WRITE
>VERIFY ON	MEMORY_WRITE ENABLE
[ON] VERIFY
[ENABLE] VERIFY WRITE
>VERIFY OF	F
[OFF] VERIFY

●例



8. バッチ機能

本ソフトウェアは、コマンドラインで実行可能な全てのコマンドをバッチプログラムによってバッチ処理させることができます。 また、ワーク変数やシステム変数、実行制御機能を使用してバッチプログラムの実行制御ができます。

バッチ機能は、バッチファイルに記述されている命令(コマンドや変数の定義など)を一行ずつ実行していきます。 命令を実行するには、行の終端に改行文字が含まれていなければなりません。

本ソフトウェアのバッチプログラムは、実行したコマンドの結果を待たずに次のコマンドを実行していきます。

コマンドウィンドウには、バッチファイルを容易に作成するための NEWBATCH コマンドがあり、コマンドウィンドウに入力したコマンドを バッチファイルへ保存する機能があります。

バッチプログラムからバッチプログラムの呼び出しも可能となっており、ネストの上限は規定していませんが、Windows リソースの範囲 に限ります。



コマンドラインのようにパラメータを省略して記述できますが、省略の仕方がコマンドラインとは多少異なります。



コマンドは大文字、小文字を区別しません。


8.1. ワーク変数

バッチプログラム内で使用するワーク変数は、バッチプログラムの起動時にパラメータとして渡される「**バッチ引数**」、1つのバッチプログ ラム内だけで有効な「**ローカル変数**」、全てのバッチプログラムで有効な「**グローバル変数**」が用意されています。 各変数のスコープのイメージを Figure 85 に示します。 バッチ引数は参照のみ可能なグローバル変数です。

[Batch1]
\$aa=5
\$BB=10
[Batch2]
\$aa=30
\$BB=20
 print \$aa ←5 を表示。
print \$BB ←20 を表示。
\$0=100 ←エラー(引数は参照のみ)

Figure 85

●使用可能なワーク変数

Table 64

変数の種類	変数名と内容		例
バッチ引数	\$0 \$1~\$9	:コマンドライン全体の文字列 :バッチ引数の1番目~9番目	\$0
グローバル変数	\$(先頭が英	長大文字から始まる文字列)	\$GLOBAL
ローカル変数	\$(先頭が英大文字以外から始まる文字列)		\$local

変数名に使える文字として有効なものは、以下の通りです。 A~Z、a~z、0~9、_(アンダーバー)

●例		
.\$a=0x10 .\$abc=0x50 .\$B0=[0x40000].W if(\$001==0x1234)	//ローカル変数\$a へ 0x10 を代入する。 //ローカル変数\$abc へ 0x50 を代入する。 //グローバル変数\$B0 へ 0x4000 番地のワードデータを代入する。 //ローカル変数\$001 の値が 0x1234 の時、真。	



8.2. ラベル

バッチプログラム内で分岐先などに使用するラベルを定義します。 ラベルは、行頭からコロン(:)で始まります。 ラベル行に、コマンドは記述できません。

●例

: COME_HERE



8.3. コメント

バッチプログラム内でコメント行を記述する時に指定します。 コメントは、行の先頭または、途中からスラッシュを2つ(//)続けて指定します。 //以降の文字列はコメントとして扱われます。 コメント行はバッチプログラムの実行には影響しません。

●例

// This is Comment Line
if (\$a==0x1234) // if \$a equal to 0x1234



以下のコマンドでは、コマンドと同一行(コマンドの後ろ)にコメントを記述することができません。 batch, dump, exit, mkdir, newbatch, option, search, upload

以下のように記述すると、エラーになります。 batch test.bat // comment



8.4. 数値演算で使用可能な演算子

8.4.1. 演算子

アドレス式などの数値演算で使用可能な演算子は Table 65 の通りです。

		Table 65
種類	記号	意味
代数演算子	+	加算
	_	減算
	*	乗算
	/	除算
	%	剰余
比較演算子	==	等しい
	!=	等しくない
	<	より小さい
	>	より大きい
	<=	以下
	>=	以上
シフト演算子	<<	左シフト
	>>	右シフト
論理演算子	&&	AND
	П	OR
	&	ビットごとの AND
	I	ビットごとの OR
	^	ビットごとの XOR
	~	NOT
代入演算子	_	右辺を左辺に代入 (※以下の代入演算も使用可能です)
	=	+=, -=, *=, /=, %=, <<=, >>=, &=, =
その他	()	שיש ביא איז איז איז איז איז איז איז איז איז א



8.4.2. 優先度と評価順序

C 言語の優先度とほぼ同じですが、代入演算子が独特です。 ※演算子の項目に記載しているカンマ(,)は区切りの記号です。

	Table 66
演算子	結合規則
+(符号), -(符号), ~, !(~と同じ処理)	左から右
*, *=, /, /=, %, %=	左から右
+, +=, -, -=	左から右
<<, <<=, >>, >>=	左から右
<, <=, >, >=	左から右
==, !=	左から右
&, &=	左から右
^, ^=	左から右
, =	左から右
&&	左から右
	左から右

カッコを使って式をグループ化できます。

例えば、ビットテストをする場合は、if((x&mask)==0)のように、カッコでくくることができます。



演算子と変数(または、値)の間には、スペースを挿入できません。



8.5. SPI フラッシュメモリの読み込み

→ 詳細は「3.4. SPI フラッシュメモリの読み込み」を参照ください。



8.6. 実行制御

8.6.1. FOR, FBREAK, NEXT(カウンタ付き繰り返し実行)

FOR 行から NEXT 行までの一連のコマンドを<条件式>が成立している間繰り返します。 FOR-NEXT 中に FBREAK が実行された場合は、ただちに FOR-NEXT のループを抜け出します。

●書式

FOR <ワーク変数>=<初期値> TO <条件式> [STEP <ステップ値>] コマンド.... [FBREAK] コマンド.... NEXT <ワーク変数>

	Table 67
パラメータ	説明
<ワーク変数>	繰り返し処理をするためのカウンタ変数を指定します。
	ワーク変数は、\$A~\$Z,\$a~\$z の 52 個のワーク変数から指定します。
	NEXT で指定するワーク変数は、FOR で指定されたワーク変数と同じものでなければなりません。
<初期值>	ワーク変数へ設定する初期値を符号付き整数で指定します。
<条件式>	繰り返し処理を制御する条件式を指定します。条件式には、以下を使用することができます。
	• 演算子
	 ワーク変数
	 メモリデータ
	• 数值
<ステップ値>	1回の繰り返し処理が終わった時にワーク変数へ加算する値を符号付き整数で指定します。
	STEP を省略した場合は、1 になります。

●例

FOR \$A=10 TO \$A<100 STEP 10 IF \$A==50 FBREAK ENDIF DUMP 0 LENGTH \$A NEXT \$A



8.6.2. WHILE, WBREAK, WEND(繰り返し実行)

<条件式>が真(0 以外)の間、WHILE-WEND 間の一連のコマンドを繰り返し実行します。 <条件式>が偽(0)であれば、ループを終了します。 WHILE-WEND 中に WBREAK が実行された場合は、ただちに WHILE-WEND のループを抜け出します。

●書式

WHILE <条件式> コマンド.... [WBREAK] コマンド.... WEND

	Table 68
パラメータ	
<条件式>	繰り返し処理を制御する条件式を指定します。
<条件式>	繰り返し処理を制御する条件式を指定します。

. \$A=0			
WHILE \$A<100			
IF \$A==50			
WBREAK			
ENDIF			
DUMP O LENG	TH \$A		
. \$A+=10			
WEND			



8.6.3. GOTO(無条件分岐)

バッチプログラムを<ラベル>の行へ分岐します。

●書式

GOTO <ラベル>

	Table 69
パラメータ	- 説明
<ラベル>	バッチプログラム中の分岐先のラベル名を指定します。

:L00P			
コマンド			
GOTO LOOP			



8.6.4. IF, ELSEIF, ELSE, ENDIF(条件判断)

<条件式>が真(0以外)の時、ELSEIF、ELSE または ENDIF の行までコマンドを実行します。ELSEIF はいくつでも指定できます。

●書式

```
IF<条件式>
コマンド....
[ELSEIF <条件式>]
[コマンド....]
[ELSE]
[コマンド....]
ENDIF
```

Tal	bl	le	7	0
TU I				v

パラメータ	説明
<条件式>	実行を制御する条件式を指定します。

	IF \$A>S	₿B		
	DUMP	0 LENGTH \$A		
E	ELSEIF	\$A==\$B		
	DUMP	0x10 LENGTH	\$A	
E	ELSEIF	\$A<\$B		
	DUMP	0x20 LENGTH	\$B	
E	ELSE			
	DUMP	0x30 LENGTH	\$B	
E	ENDIF			



8.6.5. END(全てのバッチプログラムの終了)

現在実行中のネストしているバッチプログラムを含む、全てのバッチプログラムを終了します。

●書式

END

IF \$A>\$B		
END		
ENDIF		



8.6.6. QUIT(現在のバッチプログラムの終了)

現在実行中のバッチプログラムを終了します。 ネストしている場合は現在のバッチプログラムだけを終了し、呼び出し元のバッチプログラムに戻ります。

●書式

QUIT

ENDIF

例			
IF \$A>\$B			
QUIT			



8.7. ECHO(バッチコマンドの表示/非表示の切り替え)

バッチプログラム中のコマンドの表示/非表示を切り替えます。

●書式

ECHO {ON|OFF}

●例		
IF \$A>\$B		
ECHO ON		
ELSE		
ECH0 OFF		
ENDIF		



8.8. KEYIN(キーボードからの入力)

<文字列>が指定されている場合には、コマンドウィンドウのステータスバーにガイド文字列として<文字列>を表示し、キーボードからの入力を待ちます。ワーク変数が指定されている場合には、キーボードから入力された文字列をワーク変数へ代入します。

キーボードからの文字列は、リターンキー(Enter)が入力されると入力終了となります。

<文字列>と<ワーク変数>が共に指定されなかった場合は、入力された文字列を式として評価し、結果を表示します。 この時、代入式などが指定されると、左辺の式へ右辺の式の結果が代入されます。

●書式

KEYIN [<文字列> [<ワーク変数>]]

パラメータ	説明		
<文字列>	コマンドウィンドウのステータスバーに表示するガイド文字列を指定します。		
<ワーク変数>	キー入力された値を設定するためのワーク変数を指定します。		

Talala 71

●例

KEYIN "A=" \$A //コマンドウィンドウに A=と表示してキー入力待ちになる。



8.9. PRINT(文字列の表示)

指定された<数値式>を評価し、<フォーマット>で指定された書式でコマンドウへ表示します。 <文字列>が指定されている場合には、<数値式>が表示される前にコマンドウへ表示されます。 <文字列>、<数値式>は空白で区切っていくつでも指定する事ができます。

●書式

PRINT {[<文字列>] [<数值式>][<フォ-マット>]} +

パラメータ	説明
<文字列>	コマンドウィンドウへ表示するガイド文字列を指定します。
<数值式>	キー入力された値を設定するためのワーク変数を指定します。
<フォーマット>	数値式を表示するときのフォーマットを指定します。

Table 72

● <フォーマット>

Table 73

書式	説明
なし	デフォルトの表示形式。 4 バイト 16 進数値とカッコで囲んで符号つき 10 進数値を表示する。
.#B	2 バイトの2 進数値で表示する。
.#LB	4 バイトの2 進数値で表示する。
.#D	2 バイトの符号付き 10 進数値で表示する。
.#LD	4 バイトの符号付き 10 進数値で表示する。
.#U	2 バイトの符号なし 10 進数値で表示する。
.#LU	4 バイトの符号なし 10 進数値で表示する。
.#H	2 バイトの 16 進数値で表示する。
.#LH	4 バイトの 16 進数値で表示する。

DDINT "chodofg"	
rnini abudutig	
	// 美仃結朱
PRINI \$A= I+2+3 \$B= I*2*3	
\$A=0x00000006 (6) \$B=0x00000006 (6)	// 実行結果
.\$A=0xffffffff	
PRINT "\$A=" \$A	
\$A=0xffffffff (-1)	// 実行結果
PRINT "\$A=" \$A, #B	
\$A=1111 1111 1111	// 実行結果
PRINT "\$A=" \$A #IB	
\$A=IIII IIII IIII IIII IIII IIII IIII II	// 実行結果



8.10. BEEP(BEEP 音を鳴らす)

ビープ音を鳴らします。

●書式

BEEP

●例

IF \$A > \$B			
BEEP			
ENDIF			



BEEP 音を鳴らす時は、Windows のサウンド設定で「一般の警告音」を設定する必要があります。



8.11. WAIT(バッチプログラムの一時停止)

指定された秒数だけバッチプログラムを停止します。

●書式

WAIT <秒数>

	Table 74	
パラメータ	説明	
<秒数>	バッチプログラムを停止する秒数を指定します。	

IF \$A > \$B	
WAIT 10	//10 秒待つ。
ENDIF	



9. スタンドアロン機能

スタンドアロン動作とは、書き込みデータや書き込み手順をプローブに記録させておき、ホスト PC と接続しなくても電源を供給した 状態でプローブの RUN ボタンを押すことによって、記録させておいた処理を単体で実行するものです。

1) 書き込みデータと書き込み手順の記録

ホスト PC に接続し、通常の手順でターゲットにデータを書き込みます。 このとき、書き込みデータと書き込み手順をプローブに記録します。





2) ホスト PC から切り離す

ホスト PC から切り離し、ターゲットと接続します。



Figure 87

3) 電源を接続し、RUN ボタンを押す

例えば充電池で電源を供給し、プローブの POWER LED が点灯したことを確認して RUN ボタンを押します。





4) 書き込み処理を行います

書き込み中は、STATUS LED が点滅します。





5) 書き込みが完了すると、結果を LED で通知します

正常終了時は、STATUS LED が緑色に点灯し、エラー終了時は ERROR LED が赤色に点灯します。



Figure 90



9.1. 手順の記録

本ソフトウェアを起動すると、以下の画面が表示されます。 赤丸のボタンをクリックすると操作手順を記録するモードになります。 この状態でプロジェクトファイルを開き(または新規作成して)、**SPI フラッシュメモリへの書き込みを一度実行します**。



Figure 91

記録中は、ウィンドウタイトル部に「短押記録中」と表示されます。





9.2. 記録の終了

記録を終了するには、本ソフトウェアを閉じるか、プロジェクトを閉じます。



9.3. 記録した手順のバックアップとリストア

記録させた手順は、ホストPCにバックアップすることができます。

また、バックアップした手順をリストアすることもできます。

リストアは別のプローブにも行えますが、リストア先のプローブに本ソフトウェアの License Code が登録されていないと RUN ボタンを 押してもエラーになります。



Figure 93

Table 75

-בבא	説明
短押記録開始	このメニューを選択すると、手順記録モードになります。
本体メモリアップロード	プローブに記録した手順をホスト PC にアップロード(=バックアップ)します。
本体メモリダウンロード	ホスト PC にアップロード(=バックアップ)した手順をプローブにダウンロード(=リストア)します。



スタンドアロン機能と、手順のバックアップ、リストアは以下の様な場合に便利です。

- 工場で書き込みを並行で行う場合
- 保守・修理でメモリ内容を頻繁にリストアする場合



改訂履歴

版数	改訂日	改訂内容
01	2014/09/30	初版。
02	2014/11/17	誤記修正。



製造者情報



株式会社 Sohwa & Sophia Technologies

	〒215-8588
[本社]	神奈川県川崎市麻生区南黒川 6-2
	ホームページ: <u>http://www.ss-technologies.co.jp</u>

子会社



Unit 5-2, Level 5, Tower 6, Avenue 5, The Horizon, Bangsar South No.8, Jalan Kerinchi 59200, Kuala Lumpur, Malaysia

HomePage : http://www.sohwa-m.com.my/