# ROM レス品にて、外部 FlashROM 品種を追加する場合の説明

Rev1.01 DEFバージョン10.10A仕様より

# 【対象CPU】

1. ROMレス品種が対象になります。(H8SX/1651、SH7264等)

#### 【機能】

- 1. H8SX/1651は、サンプルで「TC58FVM5T」の対応ソフトを用意してあります。(ルネサスCのみ対応)
- 2. SH7264は、サンプルで「AT49BV320D」の対応ソフトを用意してあります。(ルネサスCのみ対応)
- 3. HewにてFlashROMの品種追加が出来るよう対応する。
- 4. FlashROM仕様にあわせたセクターイレーズとバイト書き込みをプログラミングするだけで追加が可能になります。
- 5. FlashROM対応ソフトをターゲット側に流し込み、Hーデバッガと通信させるためのポートがCPU品種ごとに相違があるため、品種別に対応しています。

#### 【品種追加前の準備】

[1-1]

1. AH7000コントロールソフト (Ver10. 10A) のインストールDIRにあるワークスペースを開きます。(SH7264での追加例)

ワークスペースを聞	K		? 🛛
ファイルの場所中:	C SH7264		* 📰•
AT49BV320D			
ファイル名(11):	SH7264.hws		選択
ファイルの種類(工):	HEW Workspaces (*.hws)	•	キャンセル

<デフォルトディレクトリ>

"c:¥Program Files¥Aone¥DEF¥rom-custom¥SH7264" になります。

2. プロジェクト名「NewFROM」をアクティブプロジェクトに設定します。



<NewFROM>をマウスクリックし、右クリックのポップアップ メニューから選択します。 3. 「プロジェクトタイプの作成」をします。



4. 新規プロジェクトタイプで使用する名前を指定します。



[1-5] プロジェクトタイプウィザード - ステップ 2 ? 🗙 どのようなプロジェクトタイプを生成しますか? ● ダイアログなし(Q): ○ 情報ダイアログとデフォルトビットマップ(D: ○ 情報ダイアログとビットマップ(M): 参照(R)\_\_ Γ 使用するアイコンは? ○ デフォルトアイコン(D) ○ ファイルから選択(A): 参照(W)... 1メージ: 🐻 生成サイズの概算: 1.75MB < 戻る(B) 法人(N) > 完了(E) キャンセル

Hewメニュー <プロジェクト>-<プロジェクトタイプの作成>をクリックします。

名前は、重複しないようにして下さい。他のMPU品種でも追加が必要な場合は、例 として「NewFROM\_7264」等が良いかもしれません。 ここの説明では「NewFROM」としておきます。

<- 「次へ」をクリックします。

この操作で「品種追加前の準備」は完了です。

\_\_\_\_

<一完了をクリックします。

# 【FlashROM品種の追加】

1. FlashROM品種を追加するため、「プロジェクトの挿入」をします。



Hewメニュー <プロジェクト>-<プロジェクトの挿入>をクリックします。

ロジェクトの挿入	?
挿入	ОК
○ 新規プロジェクト(N)	キャンセル
( 就任フロジェクト生):	
	参照(B)_
1	

「新規プロジェクト」を指定して、「OK」をクリックします。

[2-3]

新規プロジェクトの挿入		? 🔀
プロジェクトタイプ Application Demonstration Empty Application Import Makefile University NewFROM プロパティー	ワークスペース名働: SH7264 プロジェクト名(P): AT498V320D ディレクトリ①: CWProgram Files¥Aone¥DEF¥rom-custom¥SH7264 CPU種別②: SuperH RISC engine ッールチェイン①: Renesas SuperH Standard ▼	参照(2)
	ОК	キャンセル

- 1) プロジェクトタイプを作成した「NewFROM」に指定し ます。
- 2) プロジェクト名を指定します。推奨としては FlashROM 名 が良いかと思います。

<- 「OK」をクリックします。



2. 新規プロジェクトに追加するFlashROM品種の定義およびプログラムを作成します。

1)「FromTemp.c」に追加フラッシュROMのセクター情報を定義します。



2)「FromTemp.c」に追加フラッシュROMの「オープン」「クローズ」関数を作成します。 [2-6]



3)「FromTemp.c」に追加フラッシュROMの「セクターイレーズ」関数を作成します。 [2-7]







5)「FromTemp.c」に追加フラッシュROMの「フルステータスチェック」関数を作成します。 [2-9]



以上で、関数の作成は終了です。

6)新規プロジェクトのFlashROMプログラムをビルドします。 [2-10]



<ビルド>-<すべてをビルド> で、「0 Errors 0 Warnings」になったことを確認します。

7)新規プロジェクトのFlashROMプログラム作成上のルール [2-11]



# $< 1 \nu - 1 \nu 1 >$ セクター情報定義のセクション名「SectTbl」の、ロケートは必ず、実行プ ログラムアドレス(TopAdr)の「+0x800」に割付て下さい。

#### < / $\nu$ -- / $\nu$ 2 >

セクション「SectTbl」のデータ配置にルールがあります。 long SectorMax; // セクター数 const long SectorTopAdr[SectorMax+1];// セクター毎 Top アドレス const この順番に配置されることが条件になります。

# < / $\nu$ -- / $\nu$ 3 > 最大セクター(ブロック)数は、「1024」です。

< 12 - 12 - 124 >スタックまで含めたオブジェクトサイズは「0x4000」16KBまでです。

#### < 12 - 125 >0 #define OK #define NG

0xee 数値変更しないで下さい。

// 変更不可 // 変更不可

#### < / $\nu$ -- / $\nu$ 6 > 下記4関数は、名称・引数・帰り値等の仕様変更は不可です。 FromOpen(void); 1) int

- 2) int FromClose(void);
- 3) int FromSectorEraseProc(char \*cmd);
- FromWriteProc(char \*cmd); 4) int

以上のルールは厳守して下さい。

### 【作成したFlashROMプログラムのデバッグ方法】

- 1. Hewにてデバッグの準備をします。
  - [3-1]



1)モジュール「main.c」の 「#define FROM\_TEST」のコメン トを外します。

- Hewメニューの <ビルド>-<ビルド>を指示 します。
- 3)「0 Errors 0 Warning」を確認し ます。

4) デバッガ使用の通信ポートと、このFlashROMプログラム使用の通信ポートと兼用させているため、通常のデバッグ操作でのデバッグは出来ません。 よって、上記「#define FROM\_TEST」のコメントを外すことにより、通信ポートを使用せず、内部メモリ操作のみで各関数のテストできるようにテスト用プログラ ムを用意しました。 2. デバッガ用コントロールソフト「DEF」にてデバッグする為の設定をします。



#### [3-2-1]



DEF ヘキサ/アブソ	リュートファイル指言	定でのダウンロード			? 🛛
ファイルの場所の:	debug			💌 🗢 🖻 🖻	<b></b>
Bit da har / II	名前 🔺	5	サイズ 43 KB	種類 ABS ファイル	更新日時 2011/03/03 14:40
取り立いそうに ノアイル ごう デスクトップ					
21 F#1X24					
<b>्रा</b> २१ २)/१३-१					
- <b>S</b>					
マイ ネットワーク	<	10			>
	ファイル名(1):	at49bv320d.abs		-	<b>開</b> (@)
	ファイルの種類(①:	ABS File (*.abs,*.x)		<u>.</u>	キャンセル

DEFメニュー <ファイル>ー<ダウンロード> で、ダウンロードします。

インストールディレクトリ

"c:¥Program Files¥Aone¥DEF¥rom-custom¥SH7264" 下の"ProjectName¥Debug" に作成したアブソリュートファイルがありますので指定し ます。 (例 ProjectName:AT49BV320D)

# [3-4]

A	//////////////////////////////////////	<ダウンロードが成功した初期画面>
A-one H-Debugger & Fischwriter for Verlu, IUA A-one AH/DUU Verlu, IU 75(1(0) - 5-500) 105500 - 11-500 - 1053300 - 475(5)(0) - 51700	C2011-02-15) for SH2ACHODD	
	2 0000.42 @3100800000 TODO OFFORDER DOTO OFFORDER DAMAGE	
1912 Spo 10 PC OPPROVICE   John   They OP SHUEL AND IN COMPANY	2.0000 X12 SADAGOUDU IUPOSPFF80000 BDI032FF813FF PCMemory • •	
FEERIN24 IN and Power(N Reset PC) and		
103 {	🔟 DEF レジスタ SH-2/A 🛛 🔀 着	(PC レジスタ値が)
<pre>FF80924: 104 set_vbr((void *)((_UBYTE *)bINT_Vectors - INT_OFFSET));</pre>	SR 00F0 \$210 FPU	「PowerON Reset PCO」 関数の
FFF8192C: 106 set fpscr(FP3CR Init & FP3CR RM);	PC OFFF80924	上面マドレフになっていてこしが
107	R0 00000000 R8 005010401	一 元頭ノトレスになっていることが
FFF80334: 108 sp_init(): 109	R1 0FFFC08D0 R9 01490F32F	―――― 確認出来ます。 )
110	R2 0000000FA R10 00000000	
111 // HardwareSetup(); // Use Hardware Setup	R3 00000000 R11 0160512C1	
112 FFF81944: 118 INITSCT():	R4 0FFFE2004 R12 005010401	
114	R5 0FFFE2008 R13 01490F32F	
115 // _CALL_INIT(); // Remove the comment a	hen you use plobe RS 080000050 R14 00000000	SP レジスタ値が内蔵 RAM のホト
115 // INIT IOLIB(): // Enable I/0 in the ap	elication(both Silwary losossesses ward losessesses	ムアドレスになっていることが確
118		認できます。
113 // ermoi0; // Remove the comment w	hen you use erring TBR 0303388882 BeristerBank	
121 // _s1ptr=NULL; // Remove the comment w	hen you use strtd	
122		>
FFF80358: 128 set_cr(3R_Init): 124		<b>VBR</b> レジスタ値が内蔵 RAM のト
• FFF80368: 125 main():		いプアドレフになっていることが確
126		ッファトレスになっていることが唯
127 // _CLUSEALL(); // Close I/U in the app 128	lication(both SIM I/U andhardware I/U)	し認できます。
128 // _CALL_END(); // Remove the comment w	hen you use global class object	
130		
131 // white(1);	<u>*</u>	
CLR •	<ul> <li>OLR ・ ブレーク算手編 ソフトフリーク</li> </ul>	
- (なし) - char	16개 💌 🖬 周期 東行回動	
	1	
	÷	
	*	
Go Break Rstillon Reset Vin Reg Vatch Syn Trace	Step CTrac CStep DI EI IntFig PUTCH	
Language		
Start Info Log	Esc Stop	
	$\langle \rangle$	
	(  <操作>	
		$p_{a} > y_{a} - b \mathbf{P} \mathbf{P} \mathbf{E} \mathbf{E} \mathbf{p} $
		$ \begin{array}{c} \mathbf{u} \\ \mathbf{v} \\ \mathbf{u} \\ \mathbf{v} \\ \mathbf$
	(	unv」よい進めより。

A-one H-DEbugger								
the one of prooffeet	& Flashwriter for Ver	10.10A A-one AH70	00 Ver10.10(2011-0	3-07) fo	r SH2A(HUDI	D		
イル(E) データ( <u>D</u> ) 実行	テ(G) ブレーク(B) 割り込み	ゆ オブション(の) ヘルプ(	B					
手上割込 15 PC 0F	FF80CAA T 周期 T View	CPU SH7264(RAN-1M C	LK-Advanced 12.0000 ×12	書込回数	00000 TOP0x000	00800 BOT	0xFFF815FF PcM	
FF80CAA TestMain	-	Src C Nix C Asm	main.c	▼□実行	テステート数 高速	-		
52 /	//						-	a ±
53 /	// TestMain()	テスト用		関数「Te	stMain0+ ≀3	進んで		
54 / EEE80CAA: 55 v	/			いるのを	確認します。		FPU	
56 {	ord research()						FFF80CA6	
57	int stat;		~				000000830	
58 EEE800AE+ 59	NewTrit():		// More2TIXE(F	R1	0000000F0	R9	0A957F886	
FFF80CB2: 60	stat = FromOpen();		// 開始処理	R2	000040001	R10	000000002	
FFF80CCA: 61	while(stat == OK) {			R3	0FFF80CA0	R11	0FFFF4000	
FFF80D38: (61)	while(stat == OK) {	-(D, ( ) +	// コマン お注土	R4	0000000F0	R12	OFFE7FFFF	
FFF80CE6: 63	if (Cnd == 'D')	idbur /;	// SectorErase(Delet	e) R5	OFFFE2008	R13	000000200	
FFF80CEC: 64	stat = FromSe	ectorEraseProc(CmdBuf);	;	N6	000000010	- 1814	UFFE/FFF	
65 EEE80002+ 88	}	(II)		57	000030000	or	077763776	
FFF80D0C: 67	stat = From/In	iteProc(CmdBuf);	// 1/1 <操作1	>			010	
68	}		このテス	・ トプログ	ラムが動作中	「にエラー	一発生	
FFF80D22: 69	else if (Cnd == '	E') {	// Exi // # か終了し	たことを	確認するため	、このう	ライン	
71	break:	1036(7,	にブレー	クポイン	トを張ってお	きます。		_
72	}		[#74:mai	n.cl			)	
73	}			-				
FFF80D50: 75 1	AULIS(1)();		77 critor完主		/坦/~~~			
76 /	//					/CTm-	Na-L DD	た / 11
11 /	/ MemInit()	テスト用			「UStep テ 月月米4	$r \cup 1rac $	ンコート PB WoitTOL まべ	センリツ 准み士士
BP1 0FFF80D3C #74:a	main.c 💌 MO 💌	✓ BP2 00	0000000	- CL			waitio z ci	進めまり。
Sectorila	x 🗸 🚺	なし) 💌 char	💌 16進		期実行			
								±
								Ŧ
Go Break RstMo	on Reset 🛛 🖬 in 🛛 F	Reg Watch Sym	Trace Step	CTrac 📶	OStep DI	EI	IntFlg P	UTCH
tart		Info	102				- Fee S	Ston
		Inte	200					
2]			and an and a second s					
-6]								
-6] <b>  A-one H-DEbugge</b> r	& Flashwriter for Ve	r10.10A A-one AH7	000 Ver10.10(2011	-03-07)	for SH2A(HI	מסר		
6] <mark>A-one H-DEbugger</mark> ァイル(E) データ(D) 実	r & Flashwriter for Ve 行優) ブレーク®) 割り込	<b>r10.10A A-one AH7</b> みの オプション(の) ヘル	7000 Ver10.10亿011 ブ(1)	-03-07)	for SH2A(HI	udu)		
6] <mark>- A-one H-DEbugger</mark> マイル(E) データ( <u>D</u> ) 実 停止 割込 15 PC 0	r & Flashwriter for Ve 行(g) ブレーク(g) 書的込 FFF80D98 「 周期 「 Yin	<mark>r10.10A A-one AH7</mark> みゆ オプション(の) ヘル ッ CPU SH7264(RAN-1M	1000 Ver10.10(2011 ブ(円) 1 CLK-Advanced 12.0000	-03-07) ×12 書込回	for SH2A(HI )鼓00000 TOP0x	0000000	BOT0×FFF815FF	
6] A-one H-DEbugger ァイル(E) データ(D) 実行 存止 割込 15 PC 0 FF80D98 - CndWait1	<mark>7 &amp; Flashwriter for Ve</mark> 行(G) ブレーク( <u>B</u> ) 割り込 FFF80D98 「周期 II Vie	<mark>r10.10A A-one AH7</mark> みゆ オプション(の) ヘル ew CPU SH7264(RAM-1M ● Src C Mix C Asm	7000 Ver10,10(2011 ブ(円) 1 CLK-Advanced 12,0000	-03-07) ×12 書込回	for SH2A(HI 酸00000TOP0x 実行ステー教 7	00000800 5速 -	BOTO×FFF815FF	
6] A-one H-DEbugger アイル(E) データ(D) 実行 専止 割込 15 PC 0 FF80D98 菜 CndWaitT FFF80D98: 96		<mark>r10.10A A-one AH7</mark> みゆ オブション(の) ヘル <sup>ew</sup> CPU SH7264(RAM-1M ● Src C Nix C Asm md)	7000 Ver10.10亿011 ブ(日) 1 CLK-Advanced 12.0000	-03-07) ×12 書込回 	for SH2A(HI 酸00000TOP0x 実行ステート数 7	JDD 00000800 新速 👤	BOT0×FFF815FF	
6] A-one H-DEbugger アイル(E) データ(D) 実行 存止 割込 15 PC 0 FF80D98 一 CndWaitT FFF80D98: 96 97	r & Flashwriter for Ve 行(G) ブレーク(B) 割り込 FFF80D98 「周期「 Yin Char CadWaitT(char *c {	r10.10A A-one AH7 みゆ オプション(の) ヘル ** CPU SH7264(RAM-1M ・ Src へ Nix へ Asm md)	7000 Ver10.10亿2011 プ(日) I CLK-Advanced 12.0000 Main.c	-03-07) ×12 書込回 ▼ □	for SH2A(HI 酸00000 TOP0xi 実行ステト数 R F レジスタ SI	JDD 00000800 罰速 <u>-</u> 1-2/A	BOTO×FFF815FF	
5] A-one H-DEbuggen アイル(E) データ(D) 実i 事止 割込 15 PC 0 FF80D98 〇〇 CmdWait1 FFF80D98: 96 97 FFF80D9A: 98	7 & Flashwriter for Ve 行(G) ブレーク(B) 書内)込 FFF80D98 「周期 「Yra Char CudVaitT(char *c { cnd[3] = 0;	r10.10A A-one AH7 みゆ オブション(の) ヘル ッ CPU SH7264(RAM-1M ・ Src C Mix C Asm md)	1000 Ver10.10(2011 7(H) 1 CLK-Advanced 12.0000	-03-07) ×12 書込回 ▼ □	for SH2A(HI 激00000 TOPOxi 実行ステト数 F レジスタ SH 00F13	JDD 00000800 新速 <u>・</u> I-2/A 210T	BOT0xFFF815FF	
6] A-one H-DEbugger アイル(E) データ(D) 実行 事止 割込 15 PC 0 FF80D98 こ CmdWait1 FFF80D98: 98 FFF80D9A: 98 FFF80D42: 99 FFF80D42: 99	7 & Flashwriter for Ve 行(G) ブレーク(E) 割り込 FFF80D98 「周期 「 Yin char CadWaitT(char *c { cnd[3] = 0; while(cnd[3] == 0){ while(cnd[3] == 0){	r10.10A A-one AH7 みゆ オプション(の) ヘル ew CPU SH7264(RAM-1M ● Src ① Mix ① Asm md)	1 CLK-Advanced 12.0000 1 CLK-Advanced 12.0000 1 main.c	-03-07) ×12 書込回 マロロ ロロ こ進ん	for SH2A(HI 酸00000TOP0x 実行ステート数 7 F レジスタ SI 00F13 0FFF80D38	JDD 00000800 筋速 <u>)</u> H-2/A 210T PR 「		
6] A-one H-DEbugger アイル(E) データ(D) 実行 夢止 割込 15 PC 0 FF80D98 第7 FFF80D98: 98 FFF80D9A: 98 FFF80D9A FFF80D9A FFF80D9A FFF80D9A FFF80	* & Flashwriter for Ve 行(G) ブレーク(E) 割り込 FFF80D98 「周期」 Via char CadVaitT(char *c { cnd[3] = 0; while(cad[3] == 0){ while(cad[3] == 0){ switch(CadStep)	r10.10A A-one AH7 みゆ オプション(の) ヘル cru SH7264(RAM-1M ・ Src C Mix C Asm md)	1000 Ver10,10(2011 ブ(L) 1 CLK-Advanced 12.0000 1 main.c 目数「CmdWaitT0」( こいるのを確認します。	-03-07) ×12 書込回 マロロ こ進ん	for SH2A(HI 酸00000T0P0× 実行ステート数 F レジスタ SI 00F13 0FFF80D98 000000000	JDD) 00000800 気速 _ H-2/A 210T PR R8	E0T0xFFF815FF E0T0xFFF815FF FPU 0FFF80CD6 000000830	
5] A-one H-DEbugger アイル(E) データ(D) 実行 夢止 割込 15 PC 0 FF80D98 〇〇 CndWaitT FFF80D98: 98 97 FFF80D94: 98 FFF80D42: 99 FFF80D42: 99 FFF80D4E: 100 101	7 & Flashwriter for Ve 行(G) ブレーク(E) 割り込 FFF80D98 「周期」「Via Char CadWaitT(char %c { cnd[3] = 0; while(cnd[3] == 0){ while(cnd[3] == 0){ switch(CadStep) case 0; break;	r10.10A A-one AH7 みゆ オプション(の) ヘル w CPU SH7264(RAM-1M © Src ① Nix ① Asm md)	2000 Ver10.10(2011 ブ(H) 1 CLK-Advanced 12.0000 1 main.c 目数「CmdWaitT()」に ざいるのを確認します。	-03-07) ×12 書込回 マロロ こ進ん	for SH2A(HI 酸00000TOP0x 実行ステト数 7 F レジスタ SI 00F13 0FFF80D38 00000000 0FFE7FFFF	JDD) 000008000 高速 _ H-2/A 210T R8 R9	BOT0×FFF815FF F P U 0FFF80CD6 000000830 0A957F886	
5] A-one H-DEbugger アイル(E) データ(D) 実行 予止 割込 15 PC 0 FF80D98 〇〇 CmdWaitT FFF80D98: 98 FFF80D98: 98 FFF80D94: 98 FFF80D94: 98 FFF80D42: 99 FFF80D42: 99 FFF80D42: 100 101 102	* & Flashwriter for Ve 行(3) ブレーク(B) 割り込 FFF80D98 「周期」「Yn Char CudWaitT(char *C { cnd[3] = 0; while(cnd[3] == 0){ while(cnd[3] == 0){ switch(CudStep) case 0: break; case 1;	r10.10A A-one AH7 みゆ オブション(②) ヘル ッ CPU SH7264(RAM-1M ・ Src へ Nix へ Asm md)	2000 Ver10.10(2011 プロ) 1 CLK-Advanced 12.0000 1 main.c 国数「CmdWaitTO」」 ごいるのを確認します。	-03-07) ×12 書込回 マロロ こ進ん	for SH2A(HI 酸00000TOP0x 実行ステト数 7 F レジスタ SI 00F1 [3 0FFF80D38 000000000 0FFE7FFFF 000000000	JDD) 000008800 新速 _ H-2/A 210T PR R8 R9 R10	BOT0×FFF815FF F P U 0FFF80CD6 000000830 0A957F886 00000002	
5] A-one H-DEbugger Pイル(E) データ(D) 実行 多止 割込 15 PC 0 FF80D98 CmdWaitT FFF80D98: 98 FFF80D98: 98 FFF80D94: 98 FFF80D42: 99 FFF80D42: 99 FFF80D42: 99 FFF80D42: 100 101 102 FFF80DCC: 103 FFF80D00- 104	7 & Flashwriter for Ve 行(3) ブレーク(B) 割内込 FFF80D98 周期 「 Yin char CadWaitT(char *c { cnd[3] = 0; while(cnd[3] == 0){ while(cnd[3] == 0){ switch(CadStep) case 0: break; case 1: case 1: cad[0] car(1)	r10.10A A-one AH7 みゆ オブション(の) ヘル マ CPU SH7264(RAM-1M 「Src C Mix C Asm md)	1000 Ver10.10(2011 プ(H) 1 CLK-Advanced 12.0000 1 main.c 国数「CmdWaitT()」( いるのを確認します。	-03-07) ×12 書込回 マロロ ご進ん 。	for SH2A(HI 酸00000 TOPOxi 実行ステート数 F F レジスタ SI 00F13 0FFF80D98 000000000 0FFE7FFFF 000000000	JDD) 00000800 新速 ・ 1-2/A 210T - PR - R8 - R9 - R9 - R10 - R11	BOT0xFFF815FF E P U 0FFF80CD6 000000830 0A957F886 00000002 0FFFF4000	
5] A-one H-DEbugger Pイル(E) データ(D) 実行 多止 割込 15 PC 0 FF80D98 CmdWaitT FFF80D98: 96 97 FFF80D98: 98 FFF80D94: 98 FFF80D94: 98 FFF80D94: 98 FFF80D94: 98 FFF80D94: 100 101 102 FFF80D05: 103 FFF80D05: 105	r & Flashwriter for Ve 行(3) ブレーク(B) 割り込 FFF80D98 周期 (1) Yr char CadWaitT(char *c { cnd[3] = 0; while(cnd[3] == 0){ while(cnd[3] == 0){ switch(CadStep) case 0: break; case 1: case 1: cand[0] cmd[2]	r10.10A A-one AH7 みゆ オブション(の) ヘル マ CPU SH7264(RAM-1M 「Src C Mix C Asm md) 【 【 【 【 【 【 【 】 【 】	1000 Ver10.10(2011 プ(日) 1 CLK-Advanced 12.0000 1 main.c 引数「CmdWaitT()」( ざいるのを確認します。 // Head // Size	-03-07) ×12 書込回 マロロ ご進ん こ進ん R3 R4	for SH2A(HI 激00000 TOPOx 実行ステート数 ア F レジスタ SH 00F13 0FFF80D98 000000000 0FFE7FFFF 000000000 00FFE7FFFF 000000000 0FFF815C0	JDD) 00000800 新速 ・ 1-2/A 210T - R8 R9 R10 R11 R12	BOT0×FFF815FF F P U OFFF80CD6 000000830 0A357F886 000000002 0FFFF4000 0FFFF4000 0FFFFFFF	
5] A=one H=DEbugger →イル(E) データ(Q) 実 多止 割込 15 PC 0 FF80D98: 96 97 FFF80D98: 98 FFF80D94: 98 FFF80D42: 99 FFF80D42: 99 FFF80D42: 99 FFF80D42: 99 FFF80D42: 100 101 102 FFF80D45: 103 FFF80D05: 104 FFF80D05: 105 FFF80D05: 106	7 & Flashwriter for Ve 行(G) ブレーク(E) 割り込 FFF80D98 周期 「Via char CadVaitT(char *c { cmd[3] = 0; while(cmd[3] == 0){ while(cmd[3] == 0){ switch(CadStep) case 0: break; case 1: case 1: cad[0] cmd[1] cmd[2] cmd[3]	r10.10A A-one AH7 みの オプション(の) ヘル ew CPU SH7264(RAM-1M @ Src ① Mix ① Asm md) 【 【 【 【 【 【 【 】 【 】 【 】 【 】 【 】 】 【 】	1000 Ver10.10(2011 つけい 1 CLK-Advanced 12.0000 1 main.c 同数「CmdWaitT0」に だいるのを確認します。 // Head // Size // Command(Delete)	-03-07) ×12 書込回 マロロ ご進ん 。 ・ ・ ・ ・	for SH2A(HI 酸00000 TOPOx 実行ステート数 7 F レジスタ SH 00F13 0FFF80D38 000000000 0FFF7FFFF 000000000 00FFF815C0 000000000	JDD) 00000800 筋速 👤 H-2/A 210T R8 R9 R10 R11 R11 R12 R13	B0T0×FFF815FF F P U 0FFF80CD6 000000830 0A957F886 00000002 0FFF4000 0FFF7FFF 000000200	
5] A-one H-DEbugger アイル(E) データ(D) 実 争止 割込 15 PC 0 FF80D98 ● CndWaitT FFF80D98: 98 FFF80D94: 98 FFF80D42: 99 FFF80D42: 99 FFF80D4E: 100 101 102 FFF80D05: 103 FFF80D05: 105 FFF80D05: 106 FFF80D05: 106 FFF80D05: 107	7 & Flashwriter for Ve 行(G) ブレーク(E) 割り込 FFF80D98 周期 「Via char CadVaitT(char *c { cnd[3] = 0; while(cnd[3] == 0){ while(cnd[3] == 0){ switch(CadStep) case 0: break; case 1: case 1: cad[0] cad[1] cad[2] cad[3] cad[3]	r10.10A A-one AH7 みの オプション(の) ヘル マ CPU SH7264(RAM-1M Src C Mix C Asm md) ( こ ( ( ( ) ( ) ( ) ( ) ( ) ( ) ( ) (	1000 Ver10.10(2011) つくし) 1 CLK-Advanced 12.0000 1 main.c 引数「CmdWaitT()」に ごいるのを確認します。 // Head // Size // Command(Delete) // Sector(H)	-03-07) ×12 書込回 マロロ ご進ん へ R3 R4 R5 R6	for SH2A(H) 激00000 TOPOx 実行ステート数 7 F レジスタ SI 00F13 0FFF80D38 000000000 0FFF7FFF 000000000 0FFF815C0 00000000 00000000 00000000	JDD) 00000800 筋速 H-2/A 210T R8 R9 R10 R11 R12 R13 R14	BOT0×FFF815FF BOT0×FFF815FF F P U OFFF80CD6 000000830 0A957F886 000000002 0FFFF4000 0FFF7FFFF 000000200 000000000	
5] A-one H-DEbuccer Pイル(E) データ(D) 実i 多止 割込 15 PC 0 FF80D98: 96 97 FFF80D98: 98 FFF80D94: 98 FFF80D94: 98 FFF80D42: 99 FFF80D42: 99 FFF80D42: 100 101 102 FFF80D00: 104 FFF80D00: 104 FFF80D00: 106 FFF80DF0: 108 FFF80DF0: 108		r10.10A A-one AH7 みゆ オプション(の) ヘル マ CPU SH7264(RAM-1M ・ Src へ Nix へ Asm md) ( こ : *X'; = 0; = 4; = 'D'; = Sector ≫ 8; = Sector ※ 8; = Sector ※ 0xff; = 0;	1000 Ver10,10(2011) つびし 1 CLK-Advanced 12.0000 1 main.c 国数「CmdWaitT(0」) こいるのを確認します。 // Head // Size // Command(Delete) // Sector(H) // Sector(L) // Sector(L)	-03-07) ×12 書込回 マロロ こ進ん 。 R4 R5 R6 R7	for SH2A(HI 酸00000TOP0x 実行ステト数 7 のFF80D38 000000000 0FFE7FFFF 000000000 0FFF815C0 000000000 0FFF815C0 000000000 000000000	JDD) 000000800 新速 H-2/A 210T R8 R9 R10 R11 R12 R13 R14 SP	BOT0×FFF815FF EDT0×FFF815FF F P U 0FFF80CD6 000000830 0A957F886 000000002 0FFFF4000 0FFF7FFFF 000000200 0FFF83FF0	
A-one H-DEbugger →(ル(E) データ(D) 実i AL 割込 15 PC 0 F80D98 CndWaitT FF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 100 101 102 FFF80D00: 103 FFF80D00: 108 FFF80D00: 108 FFF80E00: 109 FFF80E04: 110		r10.10A A-one AH7 ∂ Q オブション(Q) ∧JU W CPU SH7264(RAM-1M ( Src ∩ Nix ∩ Asm md) (	1000 Ver10,10(2011) ブ(日) 1 CLK-Advanced 12.0000 1 CLK-Advanced 12.0000 1 main.c 国数「CmdWaitT(0」) 二、このを確認します。 // Head // Size // Size // Size // Sector(L) // Sector(L) // Sun(dumy) // 全セクター級7	-03-07) ×12 書込回 マロロ こ進ん 。 ・ ・ ・ ・ ・ ・ ・	for SH2A(HI 酸00000 TOP0x 実行ステート数 7 F レジスタ SI 00F13 0FFF80D38 000000000 0FFE7FFFF 000000000 0FFF815C0 000000000 00FFF815C0 000000000 00000000 00000000 00000000	JDD) 000000800 新速 ▼ 1-2/A 210T PR R8 R9 R10 R11 R12 R13 R14 R13 R14 SP MACL	BOT0×FFF815FF BOT0×FFF815FF × F P U 0FFF80CD6 000000830 0A957F886 000000002 0FFFF4000 0FFF7FFF 000000200 0FFF783FF0 012345678	
S A-one H-DEbugger →イル(E) データ(D) 実i 多止 割込 15 PC 0 F80D98 CndWaitT FF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 100 101 102 FFF80D00: 103 FFF80D00: 108 FFF80D00: 108 FFF80E00: 109 FFF80E00: 109 FFF80E04: 110	* Flashwriter for Ve     (〒(3) ブレーク(B) 割り込     (〒)(3) ブレーク(B) 割り込     (〒)(3) = 0)     (md[3] = 0;     while(cmd[3] == 0){         while(cmd[3] == 0){         while(cmd[3] == 0){             while(cmd[3] == 0){                  while(cmd[3] == 0){	r10.10A A-one AH7 みの オブション(の) ヘル マ CPU SH7264(RAM-1M ・ Src へ Mix へ Asm nd) ( ・ X'; = 0; = 4; = 'D'; = Sector & 0xff; = 0; ector >= SectorMax) { Step = 0;	(000 Ver10.10(2011) ブ(h) I CLK-Advanced 12.0000 main.c 国数「CmdWaitT()」( いるのを確認します。 // Head // Size // Command(Delete) // Sector(H) // Sector(L) // Sun(dumny) // 全セクター終了	-03-07) ×12 書込回 マロロ ご進ん に進ん R3 R4 R5 R6 R7 MACH GBR	for SH2A(HI 酸00000 TOP0x 実行ステート数 ア F レジスタ SI 00F13: 0FFF80D98 000000000 0FFE7FFFF 000000000 0FFF815C0 000000000 000000000 00000000 00000000	JDD) 000008800 5 速 ▼ 1-2/A 210T PR R3 R10 R11 R12 R13 R14 SP MACL VBR	BOT0×FFF815FF BOT0×FFF815FF × F P U 0FFF80CD6 0000000830 0A957F886 000000002 0FFFF4000 0FFF7FFF 000000200 000000000 0FFF83FF0 012345678 0FFF80000	
A-one H-DEbugger →(ル(E) データ(D) 実 み上 割込 15 PC 0 F80D98 CadWaitT FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 100 FFF80D00: 104 FFF80D00: 104 FFF80D00: 105 FFF80D00: 106 FFF80D00: 108 FFF80D00: 109 FFF80D00: 109 FFF80D00: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 101 FFF80E01: 101 FFF	* Flashwriter for Ve     (〒(④) ブレーク(B) 割内込     (FFF80D98 「周期日 「 Yii     (char CadWaitT(char *c     (and[3] = 0;     while(cnd[3] == 0){         while(cnd[3] == 0){             while(cnd[3] == 0){                  while(cnd[3] == 0){	r10.10A A-one AH7 みの オブション(の) ヘル マ CPU SH7264(RAM-1M (* Src C Mix C Asm md) (* Src C Mix C Asm md) (* * **; = 0; = 4; = *0; = Sector >> 8; = Sector >> 8;	(000 Ver10.10(2011) プ(日) I CLK-Advanced 12.0000 main.c 調数「CmdWaitT()」( さいるのを確認します。 // Head // Size // Command(Delete) // Sector(H) // Sector(L) // Sun(dumny) // 全セクター終了	-03-07) ×12 書込回 マロロロ ご進ん R3 R4 R5 R6 R7 MACH GBR TBR	for SH2A(HI 激00000 TOPOxi 実行ステート数 ア F レジスタ SI 00F13: 0FFF80D98 000000000 0FFF2FFFF 000000000 0FFF815C0 000000000 000000000 000000000 000000	JDD) 00000800 5j速 ▼ 1-2/A 210T PR R3 R10 R11 R12 R13 R14 SP MACL VBR Res	BOT0×FFF815FF BOT0×FFF815FF F P U OFFF80CD6 000000830 0A357F886 000000002 0FFFF4000 0FFF7FFF 000000200 0FFF83FF0 012345678 0FFF80000 isterBank	
A-one H-DEbugger →イル(E) データ(D) 実行 多止 割込 15 PC 0 FF80D98 CmdWait1 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 100 101 102 FFF80D00: 103 FFF80D00: 104 FFF80D00: 105 FFF80D00: 108 FFF80D00: 108 FFF80D00: 109 FFF80E01: 109 FFF80E	* Flashwriter for Ve     (〒(④) ブレーク(B) 割り込     FFF80D98 「周期日 「 Yin     (char CadWaitT(char *c     {         cmd[3] = 0;         while(cmd[3] == 0){         while(cmd[3] == 0){         while(cmd[3] == 0){             switch(CmdStep)             case 0: break;             case 1:                  cad[0]                  cand[1]                 cand[2]                 cand[3]                 cand[4]                 cand[5]                 cand[6]                 if (++S	r10.10A A-one AH7 みの オブション(の) ヘル マ CPU SH7264(RAM-1M ・ Src C Mix C Asm md) { { { { { { { { { { { { { { { { { {	(000 Ver10,10(2011) プ(日) 1 CLK-Advanced 12,0000 1 main.c 割数「CmdWaitT()」( ないるのを確認します。 // Head // Size // Command(Delete) // Sector(L) // Sector(L	-03-07) ×12 書込回 マロロ ご進ん こ進ん R3 R4 R5 R6 R7 MACH GBR TBR	for SH2A(HI 激00000 TOPOx 実行スケート数 ア F レジスタ SH 00F13 0FFF80D98 000000000 0FFE7FFFF 000000000 0FFF815C0 000000000 000000000 000000000 000000	JDD) 00000800 新速 ▼ 1-2/A 210T PR R10 R11 R12 R13 R14 SP MACL VBR Reg	BOT0×FFF815FF BOT0×FFF815FF  FPU OFFF80CD6 000000830 0A957F886 000000002 0FFF4000 0FFF7FFF 000000200 0FFF83FF0 012345678 0FFF80000 isterBank	
5] A-one H-DEbuccer →イル(E) データ(D) 実i 多止 割込 15 PC 0 FF80D98: 96 97 FFF80D98: 98 FFF80D94: 98 FFF80D94: 98 FFF80D42: 99 FFF80D42: 99 FFF80D42: 100 101 102 FFF80D00: 104 FFF80D00: 104 FFF80D00: 105 FFF80D00: 106 FFF80D00: 108 FFF80D00: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 101 112 113 114 FFF80E28: 115	* Flashwriter for Ve     ********************************	r10.10A A-one AH7 みの オブション(の) ヘル マ CPU SH7264(RAM-1M ・ Src C Mix C Asm md) ( Src C Mix C Asm md) ( ま '%'; = 0; = 4; = '2'; = 5ector シ 8; = Sector & 0xff; = 0; ector >= SectorMax) [ Step = 0; = '%';	(000 Ver10,10(2011) (CLK-Advanced 12,0000) 1 main.c 国数「CmdWaitT()」) だいるのを確認します。 // Head // Size // Command(Delete) // Sector(L) // Sector(L)	-03-07) ×12 書込回 マロロロロロロロロロロロロロロロロロロロロロロロロロロロロロロロロロロロロ	for SH2A(HI 激00000 TOPOx 実行スケー接 ア F レジスタ SI 00F13 0FFF80D38 000000000 0FFF7FFF 000000000 00FFF815C0 000000000 00000000 00000000 000000	JDD) 00000800 訪速 -2/A 210T R8 R9 R10 R11 R11 R12 R13 R14 SP MACL VBR Reg	BOT0×FFF815FF  BOT0×FFF815FF  F P U  OFFF80CD6  000000830  0A957F886  000000002  OFFF4000  OFFF7FFF  000000200  OFFF83FF0  012345678  OFFF80000  isterBank	
5] A-one H-DEbugger 7-(ル(E) データ(D) 実i 夢止 割込 15 PC 0 FF80D98: 98 FFF80D98: 98 FFF80D94: 98 FFF80D94: 98 FFF80D42: 99 FFF80D42: 99 FFF80D4E: 100 101 102 FFF80D00: 104 FFF80D00: 104 FFF80D00: 104 FFF80D00: 108 FFF80D00: 108 FFF80D00: 108 FFF80D00: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 109 FFF80E01: 110 FFF80E11: 111 112 113 114 FFF80E28: 115 FFF80E20: 116	x & Flashwriter for Ve         (T(G)       ブレーク(E)       割り込         FFF80D98       周期       Yin         (m)       (m)       (m)         (char       CudWaitT(char *color       (m)         (char       CudWaitT(char *color       (m)         (m)       (m)       (m)       (m)         (char       CudWaitT(char *color       (m)         (char       CudWaitT(char *color       (m)         (m)       (m)       (m)       (m)         (char       CudWaitT(char *color       (m)       (m)         (char       (m)       (m)       (m)       (m)       (m)         (char       (m)       (m)       (m)       (m) <td< td=""><td>r10.10A A-one AH7 みの オブション(の) ヘル マ CPU SH7264(RAM-1M ・ Src へ Nix へ Asm nd) ( Src へ Nix へ Asm nd) ( = '%'; = 0; = 4; = '0; = Sector ※ 8; = Sector ※ 8; = Sector ※ 0xff; = 0; = ctor &gt;= SectorNax) { Step = 0; = '%'; = 0; = '%'; = 0; = '%'; = 0; = ( ); = ( ); =</td><td>1000 Ver10,10(2011) つけいのでは、このでは、このののでは、、、、、、、、、、、、、、、、、、、、、、、、、</td><td>-03-07) ×12 書込回 マロロ ご進ん こ進ん R3 R4 R5 R6 R7 MACH GBR TBR</td><td>for SH2A(HI 激00000 TOPOx 実行ステー接 7 F レジスタ SI 00F13 0FFF80D38 000000000 0FFF7FFF 000000000 0FFF815C0 000000000 000000000 000000000 000000</td><td>JDD) 00000800 筋速 H-2/A 210T R8 R9 R10 R11 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R14 R15 R15 R15 R15 R15 R15 R15 R15 R15 R15</td><td>BOT0×FFF815FF BOT0×FFF815FF F P U OFFF80CD6 000000830 0A957F886 000000002 0FFF74000 0FFF7FFF 000000200 0FFF83FF0 012345678 0FFF80000 isterBank</td><td></td></td<>	r10.10A A-one AH7 みの オブション(の) ヘル マ CPU SH7264(RAM-1M ・ Src へ Nix へ Asm nd) ( Src へ Nix へ Asm nd) ( = '%'; = 0; = 4; = '0; = Sector ※ 8; = Sector ※ 8; = Sector ※ 0xff; = 0; = ctor >= SectorNax) { Step = 0; = '%'; = 0; = '%'; = 0; = '%'; = 0; = ( ); =	1000 Ver10,10(2011) つけいのでは、このでは、このののでは、、、、、、、、、、、、、、、、、、、、、、、、、	-03-07) ×12 書込回 マロロ ご進ん こ進ん R3 R4 R5 R6 R7 MACH GBR TBR	for SH2A(HI 激00000 TOPOx 実行ステー接 7 F レジスタ SI 00F13 0FFF80D38 000000000 0FFF7FFF 000000000 0FFF815C0 000000000 000000000 000000000 000000	JDD) 00000800 筋速 H-2/A 210T R8 R9 R10 R11 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R12 R13 R14 R14 R15 R15 R15 R15 R15 R15 R15 R15 R15 R15	BOT0×FFF815FF BOT0×FFF815FF F P U OFFF80CD6 000000830 0A957F886 000000002 0FFF74000 0FFF7FFF 000000200 0FFF83FF0 012345678 0FFF80000 isterBank	
6] A-one H-DEbugger アイル(E) データ(D) 実i 亭止 割込 15 PC 0 FF80D98: 96 97 FFF80D98: 98 FFF80D94: 98 FFF80D94: 98 FFF80D94: 98 FFF80D94: 98 FFF80D94: 98 FFF80D94: 100 101 102 FFF80D00: 104 FFF80D00: 104 FFF80D00: 108 FFF80D00: 108 FFF80D00: 108 FFF80D00: 108 FFF80D00: 108 FFF80E04: 110 FFF80E04: 110 FFF80E04: 110 FFF80E1E: 111 112 113 114 FFF80E28: 115 FFF80E22: 117	x        Flashwriter for Ve         (T(G)       ブレーク(E)       割り込         FFF80D98       「割川」「 Yiii         (m)       (m)         (char       CudWaitT(char *Coll         (char       CudWaitT(char *Coll         (m)       (m)         (char       CudWaitT(char *Coll         (m)       (m)         (char       CudWaitT(char *Coll         (m)       (m)         (m)       (m)         (char       (m)         (m)       (m)         (case 0: break;       case 1:         (m)       (m)         (m)       (m)         (cm)       (m)         (m)       (m)         (case 1:       (m)         (m)       (m)         (m)       (m)         (cm)       (m)         (m)       (m)         (case 2:       (m)         (m)       (m)	r10.10A A-one AH7 ∂ Q オブション(Q) ∧J/ ∞ CPU SH7264(RAM-1M ( Src ∩ Nix ∩ Asm md) ( Src ∩ Nix ∩ Asm md) ( = '%'; = 0; = 4; = '%'; = 0; = 5ector ≫ 8; = Sector ≫ 8; = Sector ≫ 8; = Sector ≫ 8; = Sector ≫ 8; = 0; =	1000 Ver10,10(2011) ブ(日) 1 CLK-Advanced 12.0000 1 main.c 国数「CmdWaitT(0」) ごいるのを確認します。 // Head // Size // Command(Delete) // Sector(H) // Sector(H	-03-07) ×12 書込回 こ進ん こ進ん R3 R4 R5 R6 R7 MACH GBR TBR	for SH2A(HI 激00000 TOPOx 実行ステート数 7 F レジスタ SI 00F13 0FFF80D38 000000000 0FFF7FFFF 000000000 0FFF815C0 000000000 000000000 000000000 000000	JDD) 000000800 新速 - 2/A 210T R8 R9 R10 R11 R12 R13 R14 SP R14 SP WACL VBR Reg	BOT0×FFF815FF F P U OFFF80CD6 000000830 0A957F886 000000002 0FFF74000 0FFF7FFF 000000200 0FFF83FF0 012345678 0FFF80000 isterBank	
6] A-one H-DEbugger アイル(E) データ(D) 実i 亭止 割込 15 PC 0 FF80D98 ○ CndWait1 FFF80D98: 98 97 FFF80D98: 98 FFF80D98: 98 97 FFF80D98: 98 97 97 97 97 97 97 97 97 97 97	x Elashwriter for Ve         (T(G) ブレーク(E) 割り)込         FFF80D98 「周川月」「Yn         char CudWaitT(char *Coll         char CudWaitT(char *Coll         cmd[3] = 0;         while(cmd[3] == 0){         while(cmd[3] == 0){         while(cmd[3] == 0){         switch(CmdStep)         case 0: break;         case 1:         cmd[0]         cmd[1]         cmd[4]         cmd[5]         cmd[6]         if (++S)         Cade         pbreak;         case 2:         cmd[0]         cmd[1]         case 2:         cmd[0]         cmd[1]         cmd[2]         case 2:         cmd[0]         cmd[1]         case 2:         cmd[1]         cmd[2]         cmd[2]         cmd[2]         cmd[2]         cmd[2]         case 2:         cmd[2]         cmd[2]         cmd[2]	r10.10A A-one AH7 ar Q オブション(Q) ヘル W CPU SH7264(RAM-1M CPU SH7264(RAM-1M C Src C Mix C Asm md) ( Src C Mix C Asm md) ( ( - 'X'; = 0; = sector & 0xff; = 0; = sector >> 8; = 0; = 0;	1000 Ver10,10(2011) つびいつ 10 main.c 同数「CmdWaitT(0」)( こいるのを確認します。 パ Size パ Command(Delete) パ Sector(L) パ Sector(L) Sector(L) パ Sector(L) パ Sector(L) Sector(L) Sector(L) Sector(L) Sector(L) Sector(L) Sector(L) Sector(L) Sector(L) Sector(L) Secto	-03-07) ×12 書込回 マロロ こ進ん R3 R4 R5 R6 R7 MACH GBR TBR	for SH2A(HI 酸00000 TOP0x 実行ステード数 ア F レジスタ SI 00F13: 0FFF80D98 000000000 0FFE7FFFF 000000000 000000000 000000000 000000	JDD) 000008800 5 速 ▼ 1-2/A 210T PR R3 R10 R11 R12 R13 R14 SP NACL VBR Reg	BOT0%FFF815FF	
6] A-one H-DEbugger アイル(E) データ(D) 実 う止 割込 15 PC 0 FF80D98 ○ CndWait1 FF80D98: 98 FFF80D94: 98 FFF80D94: 98 FFF80D94: 98 FFF80D94: 99 FFF80D94: 99 FFF80D94: 99 FFF80D94: 99 FFF80D94: 99 FFF80D94: 99 FFF80D94: 99 FFF80D94: 99 FFF80D94: 99 FFF80D94: 100 FFF80D05: 108 FFF80D95: 108 FFF80E25: 117 FFF80E28: 115 FFF80E26: 118 FFF80E26: 118 FFF80E40: 119 FFF80E40: 119 FFF80E40: 119 FFF80E40: 119	x & Flashwriter for Ve         (T(G) ブレーク(E) まり)込         FFF80D98 □ BUR □ Ye         char CndWaitT(char *C         cmd[3] = 0;         while(cnd[3] == 0){         while(cnd[3] == 0){         while(cnd[3] == 0){         switch(CndStep)         case 0: break:         case 1:         cnd[0]         cmd[1]         cnd[4]         cnd[5]         cnd[6]         if (++S)         Cnd         preak;         case 2:         cnd[0]         cnd[1]         cnd[2]         cnd[3]	r10.10A A-one AH7 ar Q オブション(Q) ヘル W CPU SH7264(RAM-1M (* Src (* Mix (* Asm md)) (* Src (* Src (* Mix (* Asm md)) (* Src (* Src	1000 Ver10,10(2011) ブ(H) 1 CLK-Advanced 12.0000 1 main.c 調数「CmdWaitT()」( いるのを確認します。 // Head // Size // Connand(Delete) // Sector(L) // Sector(L) // Sector(L) // Sun(dumy) // 全セクター終了 (Weiteつフンよどの く操作> 「Watch」ショート P リックします。	-03-07) ×12 書込回 マロロ ご進ん こ進ん R3 R4 R5 R6 R7 MACH GBR TBR B をク	for SH2A(HI 酸00000 TOP0x 実行ステート数 ア F レジスタ SI 00F13: 0FFF80D98 000000000 0FFE7FFFF 000000000 000000000 000000000 000000	JDD) 00000800 5j速 ▼ 1-2/A 210T PR R3 R10 R11 R12 R13 R14 SP MACL VBR Res	BOT0xFFF815FF BOT0xFFF815FF  F P U OFFF80CD6 000000830 0A357F886 00000002 0FFFF4000 0FFF7FFF 000000200 0FFF83FF0 012345678 0FFF80000 isterBank	
6] A-one H-DEbugger アイル(E) データ(D) 実 等止 割込 15 PC 0 FF80D98 ○ CmdNait1 FF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D0: 100 FFF80D00: 104 FFF80D00: 104 FFF80D00: 105 FFF80D00: 108 FFF80D00: 108 FFF80D000: 108 FFF80D00: 108	* * Flashwriter for Ve         (7(G) ブレーク(E) *))2         FFF80D98         (a) = 0;         (a) = 0; <td>r10.10A A-one AH7 ar Q オブション(Q) ヘル W CPU SH7264(RAM-1M (* Src C Mix C Asm nd) (* Src C Mix</td> <td>1000 Ver10.10(2011 プ(E) 1 CLK-Advanced 12.0000 1 main.c 割数「CmdWaitT()」( いるのを確認します。 // Head // Size // Connand(Delete) // Sector(L) // Sector(L)</td> <td>-03-07) ×12 書込回 マロロロ ご進ん の R3 R4 R5 R6 R7 MACH GBR TBR 武</td> <td>for SH2A(HI 識00000 TOPOxi 実行スケート数 ア F レジスタ SI 00F13: 0FFF80D98 000000000 0FFF2FFFF 000000000 0FFF815C0 000000000 000000000 000000000 000000</td> <td>JDD) 00000800 5j速 ▼ 1-2/A 210T PR R9 R10 R11 R12 R13 R14 SP MACL VBR Reg</td> <td>BOT0×FFF815FF BOT0×FFF815FF  F P U OFFF80CD6 000000830 0A357F886 000000002 0FFF7886 00000000 0FFF83FF0 012345678 0FFF80000 isterBank 7'L~73¥48 1</td> <td></td>	r10.10A A-one AH7 ar Q オブション(Q) ヘル W CPU SH7264(RAM-1M (* Src C Mix C Asm nd) (* Src C Mix	1000 Ver10.10(2011 プ(E) 1 CLK-Advanced 12.0000 1 main.c 割数「CmdWaitT()」( いるのを確認します。 // Head // Size // Connand(Delete) // Sector(L) // Sector(L)	-03-07) ×12 書込回 マロロロ ご進ん の R3 R4 R5 R6 R7 MACH GBR TBR 武	for SH2A(HI 識00000 TOPOxi 実行スケート数 ア F レジスタ SI 00F13: 0FFF80D98 000000000 0FFF2FFFF 000000000 0FFF815C0 000000000 000000000 000000000 000000	JDD) 00000800 5j速 ▼ 1-2/A 210T PR R9 R10 R11 R12 R13 R14 SP MACL VBR Reg	BOT0×FFF815FF BOT0×FFF815FF  F P U OFFF80CD6 000000830 0A357F886 000000002 0FFF7886 00000000 0FFF83FF0 012345678 0FFF80000 isterBank 7'L~73¥48 1	
6] A=one H=DEbugger (7) (E) 7 - 9(D) ₹ (7) (F) (E) (E) (E) (E) (E) (E) (E) (E) (E) (E	* * Flashwriter for Ve         (T(G) ブレーク(B) *))(2)         FFF80D98         ************************************	r10.10A A-one AH7 ∂×Q オブション(Q) ∧J/ W CPU SH7264(RAM-1M (* Src C Mix C Asm md) (* Src C Mix	1 CLK-Advanced 12.0000 1 CLK-Advanced 12.0000 1 main.c	-03-07) ×12 書込回 マロロ ご進ん こ進ん R3 R4 R5 R6 R7 MACH GBR TBR Bをク	for SH2A(HI 激00000 TOPOx 実行ステー接 ア F レジスタ SI 00F13: 0FFF80D38 000000000 0FFF7FFFF 000000000 000000000 000000000 000000	JDD) 00000800 新速 - 2/A 210T PR R8 R9 R10 R11 R12 R13 R14 SP MACL VBR Reg	EDT0×FFF815FF EDT0×FFF815FF EDT0×FFF815FF EDT000000000 EFFF80CD6 EDT00000000 EFFF80000 EFFF80000 EFFF80000 EFFF80000 EI12345678 EFFF800	
6] A=one H=DEbuccer アイル(E) データ(D) 実行 等止 割込 15 PC 0 FF80D98 ■ CndWait1 FF80D98 97 FFF80D98: 98 97 FFF80D98: 98 97 FFF80D00: 100 FFF80D00: 100 FFF80D00: 100 FFF80D00: 100 FFF80D00: 100 FFF80D90: 100 FFF80D98: 10	x & Flashwriter for Ve         (T(G) ブレーク(B) 割り込         FFF80D98         (main (G) = 0;         while(cmd[3] = 0;         while(cmd[3] == 0){         while(cmd[3] == 0){         while(cmd[3] == 0){         switch(CmdStep)         case 0: break;         case 1:         cmd[0]         cmd[1]         cmd[2]         cmd[3]         cmd[4]         cmd[5]         case 2:         case 2:         cmd[1]         cmd[2]         cmd[3]         cmd[4]         case 2:         cmd[1]         cmd[2]         cmd[1]         cmd[2]         cmd[3]         cmd[4]         case 2:         cmd[3]         cmd[4]         cmd[3]         cmd[4]         cmd[4]         case 1:         case 2:         cmd[3]         cmd[4]         cmd[4]         cmd[4]         cmd[4]         cmd[4]         cmd[4]         cmd[4]         cmd[4]     <	r10.10A A-one AH7 ar Q	1 CLK-Advanced 12.0000 1 CLK-Advanced 12.0000 1 main.c	-03-07) ×12 書込回 マロロ ご進ん こ進ん R3 R4 R5 R6 R7 MACH GBR TBR Bをク	for SH2A(H) 激00000 TOPOx 実行ステー技 7 F レジスタ SH 00F13 0FFF80D38 000000000 0FFF7FFF 000000000 000000000 000000000 000000	JDD) 00000800 訪速 -2/A 210T R8 R9 R10 R11 R12 R13 R14 R12 R13 R14 VBR R10 R11 R12 R13 R14 VBR R10 R10 R11 R12 R13 R14 VBR	BOT0×FFF815FF BOT0×FFF815FF  F P U OFFF80CD6 000000830 0A957F886 000000002 0FFF74000 0FFF27FFFF 000000000 0FFF83FF0 012345678 0FFF80000 isterBank 7'L~?%¥#8	
6] A-one H-DEbugger アイル(E) データ(D) 実i 亭止 割込 15 PC 0 FF80D98 ○ CndWait1 FFF80D98: 98 FFF80D98: 98 FFF80D00: 103 FFF80D00: 104 FFF80D00: 104 FFF80D00: 104 FFF80D00: 108 FFF80D00: 108 FFF80D00: 108 FFF80D00: 109 FFF80D00: 109 FFF80E00: 109 FFF80E00: 109 FFF80E00: 109 FFF80E00: 109 FFF80E00: 109 FFF80E00: 108 FFF80E2: 117 FFF80E28: 115 FFF80E28: 115 FFF80E28: 115 FFF80E28: 117 FFF80E28: 118 FFF80E34: 118 FFF80E34: 119 FFF80E34: 119	x Flashwriter for Ve         (T(G) ブレーク(E) 割り)込         FFF80D98 「周川月」「Yn         char CudWaitT(char *C         char CudWaitT(char *C         cmd[3] = 0;         while(cmd[3] == 0){         while(cmd[3] == 0){         while(cmd[3] == 0){         switch(CmdStep)         case 0: break;         case 1:         cmd[0]         cmd[1]         cmd[4]         cmd[5]         cmd[6]         if (++S)         Cmd[1]         cmd[2]         cmd[4]         cmd[2]         cmd[1]         cmd[2]         cmd[4]         cmd[2]         cmd[2]         cmd[1]         cmd[2]         cmd[1]         cmd[2]         cmd[3]         cmd[4]         switfn1         main.c         H0	r10.10A A-one AH7 ar Q	1000 Ver10,10(2011) つけいの 10 Main.c 回数「CmdWaitT(0」)( こいるのを確認します。 // Head // Size // Command(Delete) // Sector(L) // Se	-03-07) ×12 書込回 マロロ ご進ん こ進ん R3 R4 R5 R6 R7 MACH GBR TBR Bをク	for SH2A(H) 激00000 TOPOx 実行ステー接 F レジスタ SI 00F13 00FF780D38 000000000 00FF787FFF 000000000 000000000 000000000 000000	JDD) 00000800 新速  - 2/A 210T R8 R9 R10 R11 R12 R13 R14 SP MACL VBR Res	BOT0×FFF815FF F P U OFFF80CD6 000000830 0A957F886 000000020 0FFF74000 0FFF7FFF 000000200 0FFF83FF0 012345678 0FFF80000 isterBank 7'L-75¥48	
A-one H-DEbugger -(ル(E) データ(D) 実i AL 割込 15 PC 0 F80D98 CmdWait1 FF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 98 FFF80D98: 100 101 102 FFF80D00: 103 FFF80D00: 104 FFF80D00: 108 FFF80D00: 108 FFF80E2: 117 FFF80E28: 115 FFF80E28: 115 FFF80E28: 117 FFF80E28: 118 FFF80E28: 118 FFF80E40: 119 FFF80E40:	x Flashwriter for Ve         (T(G) ブレーク(E) 割り)込         FFF80D98 □ 周川月 □ Ye         char CudWaitT(char *C         char CudWaitT(char *C         cmd[3] = 0;         while(cmd[3] == 0){         while(cmd[3] == 0){         while(cmd[3] == 0){         switch(CmdStep)         case 0: break:         case 1:         cmd[0]         cmd[1]         cmd[3]         cmd[4]         cmd[6]         if (++S)         Cmd         pbreak;         case 2:         cmd[0]         cmd[1]         cmd[2]         cmd[3]         cmd[4]         cmd[4]         cmd[5]         cmd[6]         if (++S)         Cmd[1]         cmd[2]         cmd[3]         cmd[3]         cmd[4]         switch(CmdS)         ax         on Reset       ¥in	r10.10A A-one AH7 ar Q	1 CLK-Advanced 12.0000 1 CLK-Advanced 12.0000 1 main.c	-03-07) ×12 書込回 マロロ ご進ん こ進ん R3 R4 R5 R6 R7 MACH GBR TBR Bをク	for SH2A(HI 酸00000 TOP0xi 実行ステード数 ア F レジスタ SI 00F13: 0FFF80D98 000000000 0FFF815C0 000000000 000000000 000000000 000000	JDD) 00000800 新速  - 2/A 210T R8 R9 R10 R11 R12 R13 R14 SP MACL VBR Reg	EI IntFla	
A=one       H= DEbugger         (1)(()       ブータ(D)       実i         A= one       H= DEbugger       (1)(()       ブータ(D)       実i         A= below       第月込       15 PC 0       (1)(1)       (1)(1)       (1)(1)         FF80D38:       80       97       FF80D34:       98       (1)(1) <td>* * Flashwriter for Ve         (7(G) ブレーク(E) *))3         FFF80D98         (a) = 0;         (a) = 0;<td>r10.10A A-one AH7 ar Q オブション(Q) ヘル W CPU SH7264(RAM-1M Src C Mix C Asm md) (* Src C Mix C A</td><td>1 CLK-Advanced 12.0000 1 CLK-Advanced 12.0000 1 main.c 調数「CmdWaitT()」( いるのを確認します、 // Head // Size // Connand(Delete) // Sector(L) // Sector(L) // Sector(L) // Sun(dumy) // 全セクター終了 (UniteDD2') ECC (UniteDD2') ECC</td><td>-03-07) ×12 書込回 マロロロ ご進ん の 不 HZ R3 R4 R5 R6 R7 MACH GBR TBR Bをク</td><td>for SH2A(HI 識00000 TOP0x 実行ステー接 ア F レジスタ SI 00F13: 0FFF80D38 000000000 0FFF7FFF 000000000 000000000 000000000 000000</td><td>JDD) 00000800 新速 1-2/A 210T PR   R8 R9 R10 R11 R12 R13 R14 SP   MACL   VBR   Reg</td><td>EI IntFla</td><td></td></td>	* * Flashwriter for Ve         (7(G) ブレーク(E) *))3         FFF80D98         (a) = 0;         (a) = 0; <td>r10.10A A-one AH7 ar Q オブション(Q) ヘル W CPU SH7264(RAM-1M Src C Mix C Asm md) (* Src C Mix C A</td> <td>1 CLK-Advanced 12.0000 1 CLK-Advanced 12.0000 1 main.c 調数「CmdWaitT()」( いるのを確認します、 // Head // Size // Connand(Delete) // Sector(L) // Sector(L) // Sector(L) // Sun(dumy) // 全セクター終了 (UniteDD2') ECC (UniteDD2') ECC</td> <td>-03-07) ×12 書込回 マロロロ ご進ん の 不 HZ R3 R4 R5 R6 R7 MACH GBR TBR Bをク</td> <td>for SH2A(HI 識00000 TOP0x 実行ステー接 ア F レジスタ SI 00F13: 0FFF80D38 000000000 0FFF7FFF 000000000 000000000 000000000 000000</td> <td>JDD) 00000800 新速 1-2/A 210T PR   R8 R9 R10 R11 R12 R13 R14 SP   MACL   VBR   Reg</td> <td>EI IntFla</td> <td></td>	r10.10A A-one AH7 ar Q オブション(Q) ヘル W CPU SH7264(RAM-1M Src C Mix C Asm md) (* Src C Mix C A	1 CLK-Advanced 12.0000 1 CLK-Advanced 12.0000 1 main.c 調数「CmdWaitT()」( いるのを確認します、 // Head // Size // Connand(Delete) // Sector(L) // Sector(L) // Sector(L) // Sun(dumy) // 全セクター終了 (UniteDD2') ECC (UniteDD2') ECC	-03-07) ×12 書込回 マロロロ ご進ん の 不 HZ R3 R4 R5 R6 R7 MACH GBR TBR Bをク	for SH2A(HI 識00000 TOP0x 実行ステー接 ア F レジスタ SI 00F13: 0FFF80D38 000000000 0FFF7FFF 000000000 000000000 000000000 000000	JDD) 00000800 新速 1-2/A 210T PR   R8 R9 R10 R11 R12 R13 R14 SP   MACL   VBR   Reg	EI IntFla	
A=one H=DEbugger (JL(E)	* * Flashwriter for Ve         (7(G) ブレーク(E) *))2         FFF80D98         ************************************	r10.10A A-one AH7 a Q	1 CLK-Advanced 12.0000 1 CLK-Advanced 12.0000 1 main.c	-03-07) ×12 書込回 マロロ ご進ん こ進ん R3 R4 R5 R6 R7 MACH GBR TBR Bをク	for SH2A(H) 激00000 TOP0x 実行ステー技 ア F レジスタ SH 00F13: 0FFF80D98 000000000 0FFE7FFFF 000000000 000000000 000000000 000000	JDD) 00000800 新速 1-2/A 210T PR   R8   R10   R11   R12   R13   R14   SP   MACL   VBR   Reg	EI IntFls	

# [3-6-1]

	DEF Watch							×
	Addres 💌 実行	Ŧ	Туре	更新	Data	□ 周期サ	ンプル	
0	CmdStep	•	long	FFF815	AO: He:	<=00000000	Dec=0000000000	
0	Sector	•	long	FFF815/	A4: He:	<=00000000	Dec=0000000000	
0	WriteAdr	•	long	FFF815	A8: He:	<=00000000	Dec=0000000000	
0	FromError	•	long	FFF815	B0: He:	<=00000000	Dec=0000000000	
œ.		•	char					
C		•	char					

デバッグを進めるため、Watch 画面に変数を登録します。

1) CmdStep
2) Sector
3) WriteAdr
4)FromError

関数「CmdWaitT」のコントロールステップ変数 セクターイレーズの進行カウンター FROM 書き込み中アドレス エラー発生時情報

以上、4変数を登録します。





#### [3-8]

🧱 DEF Watch			×
Addres ▼ 実行	Туре	更新 Data 🔽 周期サンブル	
🔿 CmdStep 🔄 💌	long	FFF815A0: Hex=00000000 Dec=0000000000	
🔿 Sector 📃 💌	long	FFF815A4: Hex=00000047 Dec=0000000071 🧹	_
♥riteAdr	long	FFF815A8: Hex=00000000 Dec=0000000000	
○ FromError	long	FFF815B0: Hex=00000000 Dec=0000000000	
•	char		
○	char		

セクターイレーズが正常に動作していますと、変数 「Sector」が、ゼロ「0」から登録セクター数までイ ンクリメントしていきます。 最終セクターまで正常終了しますと、変数 「CmdStep」がゼロ「0」になります。

異常終了しますと、[3·5]で設定したブレークポイン トで停止します。

[3-9]

	,																								
00300	000	-	1×30_0000					•	X	(Ŧ!	ノタ	シ	ヺ	•	cł	18.7				•	16逋	•	原期	実行	回戰
00	30000	0:		-F	FF	FF	FF	FF	FF	FF						1									
00	30001	0:		FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						-
00	30002	0:		FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						120
00	30003	0:		FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						-
00	30004	0:		FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						
00	30005	0:		FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						
00	30006	0:		FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						
00	30007	0:		FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						
00	30008	0:		FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						-
00	30009	0:		FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						Ŧ
Go	b E	Ireak	RstMon	Rese	t		lin		R	eg		¥a	atc	h	8	Sym	100	T	race		Step	CTra	CSt	ep	DI

<ダンプ画面>

イレーズされているかダンプ画面で確認します。

5. 書き込み関数「FromWriteProc(char \*cmd)」をデバッグします。

## [3-10]

🔤 DEF Watch 🔀	<操作1> プログラム進行を確認するため、「周期サンプル」
Addres ▼ 実行 Type 更新 Data ▼ 周期サンプル	をチェックします。
🔿 CmdStep 🔽 Iong 2	
C Sector 🔄 Iong FFF815A4: Hex=00000047 Dec=0000000	
♥ WriteAdr    Iong FFF815A8: Hex=00000000 Dec=0000000000	$\langle 梁 h 2 \rangle$ 亦粉「CmdSton」な粉値「9」にします
C FromError 🔄 Iong FFF815B0: Hex=00000000 Dec=000000000	✓ 変数「Ondotep」を数値「2」にします。 [Enter][2][Enter]で変更できます。
• char	
C char	<備考> 関数「CmdWaitT」のソースを見てもらえれば理解で きるかと思いますが、「CmdSton」を「2」にします
	と、メモリー上でコマンドデータを作成して関数
	「FromWriteProc(char *cmd)」に渡す仕組みに
[3-7-1]	なっています。
	全て正常終了しますと、全エリア書き込み後、変数 「CmdStep」が数値「0」になります。

<操作3> 「Go」ショート PB をクリックします。

FEODER	Cedfeitt	*	@ Sro C Wix	C As	a sain,	6		▲ □ 第日スラートは 進速	Terring !!	attentis .
	85 /	/								
177 (10)(10)	- <b>88</b> d	har – GedfeitTúcher Au	will little							
	87 4				Life a	10010	Turks	WHE Date IX HURST'S	÷u.	1.00
TTROOM	50	ced[1] = 0;		-	and it am			TTTELLAR Have ARRANGED	Concert Balances and 1	
FFFEEDAZ:	89	while(cwd[3] == 0)(		14	and the b		TONE	FFFETSMC, PRO-OPPORT	000-0000000000	
FF88F36;	(89)	whiteCowdE81 ** 030		IC E	Sector		Tone	FTYS1544: Mex+00000000	Dec-0000000000	
TTUTAL	100	seitch(DedStep)		0	hiteAdr		lone	FYF015A0: Hex+00000000 1	Dec-0000000000	
	101	case 01 break1		0	rostroom		tone	FTF01500: Nex+00000000 1	Dec+0000000000	_
	10.2	case 1:		- 01		-	aba c			
FF BBCCC	183	CM0(0)	+ (K)	- 14			- and			
TTORNEY .	104	CNOLTS				-	forer.			_
TT OUTLOS	195	CMOLIJ			11 -	120	a la la			
TTANK		cito(s)	A Destar Valle		110	and and M	eridt.0			
TTOOLCO.	150	cacity (m/0.1	· Sector 1 Buff-			ect cell				
TTOOLOG-	100	10000	- secon e 4001;		11 *	al day				
FFEDFOX.	110	if feet	and an 24 Sectors	feel 1	11.4	40.75.12	-8.7			
TRANS IF-	111		Care I In			1974	10.			anna (SS)
	112									
	112	breakt								
	114	case 21			1/ 1	rile 37	0 M	241		
TFREETS:	115	cmd001	< 1816		// H	ned				
TTUELC:	116	cad[1]	× 4:							(2))
TFEOEDJ:	117	cad()]	+ \$+128;		// \$	29				-
TFEBERA:	118	emdEX3	< 1411 C		// 0	(March 1				
TTTOIT II.	114	الالبيد	E BUTH ALLOW MY BE	6		_	al l	and the second second second second	Contraction of the local division of the loc	
1P1 STTT	1000C \$745e	ein.o - 90 -	*		PROPERTY	10		• Q.R • ×	77-38	M 9717
	- Sectorile	×	(ML) 3	1 0	ur.	*	16:0		1	
						agent.		teres and the second se		
-	_									- <u>-</u>
ao Br	eak RstWo	n Reset Tin	Reg Tatch	5	yn Tr	ace	Step	Cirac CStep DI	El Ini	(FLa PE
	and the second second									10
171				Info	Los	1			Esc	Stop
				A DESCRIPTION OF						and the second se

#### [3-11]

🔤 DEF Watch		
Addres ▼ 実行	Type 更新 Data 🔽 周期サンブル	
🔿 CmdStep 📃 💌	long FFF815A0: Hex=00000000 Dec=000000000	
🔿 Sector 📃 💌	Iong FFF815A4: Hex=00000047 Dec=0000000071	
🔿 WriteAdr 📃 💌	long FFF815A8: Hex=00400000 Dec=0004194304	
🔿 FromError 🛛 💌	long FFF815B0: Hex=00000000 Dec=000000000	
• •	char	
C 🔹	char	

書き込みが正常に動作していますと、変数
「WriteAdr」が、ゼロ「0」から最終アドレスまで
「0x80」ごとに加算していきます。
最終アドレスまで正常終了しますと、変数
「CmdStep」がゼロ「0」になります。
異常終了しますと、[3-5]で設定したブレークポイン

トで停止します。

# [3-12]

00100000 🚆 0×10_0000		-	メモリ	ダンブ 👱	char	💌 16道 💌 🗖 周期 実行回
00100000:	30 30 3	31 30 30	30 30 30	00 00 00	00 00 00	0 00 00 00100000
00100010:	00 00 0	00 00 00	00 00 00	00 00 00	00 00 00	0 00 00
00100020:	00 00 0	00 00 00	00 00 00	00 00 00	00 00 00	0 00 00
00100030:	00 00 0	00 00 00	00 00 00	00 00 00	00 00 00	0 00 00 [
00100040:	00 00 0	00 00 00	00 00 00	00 00 00	00 00 00	0 00 00 [
00100050:	00 00 0	00 00 00	00 00 00	00 00 00	00 00 00	0 00 00 [
00100060:	00 00 0	00 00 00	00 00 00	00 00 00	00 00 00	00 00
00100070:	00 00 0	00 00 00	00 00 00	00 00 00	00 00 00	0 00 00
00100080:	30 30 3	31 30 30	30 38 30	01 01 01	01 01 01	01 01 00100080
00100090:	01 01 0	01 01 01	01 01 01	01 01 01	01 01 01	01 01 1
Go Break RstMon R	eset	¶in	Reg	Tatch	Sym	Trace Step CTrac CStep

#### <ダンプ画面>

正しくデータが書き込まれたかダンプ画面で確認します。

ゼロ番地以外は、0x80 番地ごとに先頭のアドレス位置をASCII 文 字データを書き、残りは、ゼロから始まるパターンを+1 ごとに埋め ています。なお、ゼロ番地から 8 バイト分は、PC/SP レジスタの初 期データになりますので、誤動作防止のため[0xff]にしています。

- 5. その他関数「FromOpen(void)/FromClose(void)」が必要な場合は、必要に応じてデバッグして下さい。
  - 1) 関数「FromOpen(void)」
    - テスト用メイン関数「TestMain()」の60行でコール(Call)しています。
  - 2) 関数 「FromClose(void)」

テスト用関数「CmdWaitT(char\*cmd)」で、コントロールステップ変数「CmdStep」を「3」以上の数値をセットしますとコール(Call)されます。

6. 異常終了時のエラー情報-「long FromError[4]」の参照

# [3-12]

== DEF (1)評価/変)	E	
評価変数 更新 <mark>↑</mark> FromError	·[0]	-
Data Address[FFF8	13C4]	
[0] == 0×00000057	Dec=0000000087	
[1] == 0x00320000	Dec=0003276800	
[2] == 0x00000082	Dec=0000000130	
[3] == 0×00000000	Dec=0000000000	
		•

#### <評価・変更窓>

エラー発生にて異常終了した場合は、変数「FromError[4]」に情報が残りますので、プログラム修正 の参考にして下さい。なお、この変数の仕様および名称等に規約はありませんので、自由に変更しても 構いません。

#### 【作成したFlashROMプログラムを正規リリース登録する】

1. FlashROMプログラムのデバッグが終了しましたら、Hewにて再コンパイルします。



1) モジュール「main.c」の 「// #define FROM\_TEST」をコメ ントします。

- Hewメニューの <ビルド>-<ビルド>を指示 します。
- 3)「0 Errors 0 Warning」を確認し ます。

2. デバッガ用コントロールソフト「DEF」にFlashROMプログラムの登録をします。



# 【作成したFlashROMプログラムの最終確認をします】

1. デバッガ用コントロールソフト「DEF」を「Start」後、FlashROMに、ダウンロードします。



[5-2] DEF ヘキサプアブソリュートファイル 指定でのダウンロード ファイルの場所中. 🗀 FROM - + 🗈 🗗 🗖 #sh7264\_ROM m 泉近使ったファイル 5201-97 TH FALSCH 71 I.K1-9 -C16\$ 17 77-11-名位 ab7254 ROM mot • THE CO Hex File (\*hex,\*.mot) ファイルの種類の • キャンセル

[5-3] A-one H-DEbugger & Flashwriter for Ver10.10A A-one AH7000 Ver10.10(2011-02-15) for SH2A(HUDD ファイルモ データロ 東行日 ブレータロ 割り込み中 オブルンロ ヘルブゼ ロード中 割込 15 PC 0000000 「原用 「アッティ DPU SPC544048-18 Q.X 00000 BOTO-003FFFFF TargetM OPU (3)(7)64 書込回時100000 TOP0.00 00 ・ 戸 実行ステート数 高速 ・ ro C Mix @ Asm ri,ri ri,ri ri,ri ri,ri 04: 3038 0042 0000 0062 0000 0082 1010 0042 1010 0022 1010 0022 1010 0022 1010 0122 1010 0122 1010 セクターイレーズ r1,8(bd,r1) r1,8(bd,r1) r1,8(bd,r1) r1,8(bd,r1) r1,8(bd,r1) r1,8(bd,r1) r1,8(bd,r1) r1,8(bd,r1) r1,8(bd,r1) 
 0818:
 1018

 0818:
 1018

 0818:
 1018

 0812:
 1018

 0812:
 1018

 0820:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0822:
 1018

 0830:
 1018

 0830:
 1018
 1.60.4. (1,003,01) (1,003,01) (1,003,01) (1,003,01) (1,003,01) (1,003,01) (1,003,01) (1,003,01) · 8P1 000000000 • CLR • フレークロレーター - BP2 00000000 • Q.R • × FronError (なし) • cha ▼ 1.6進 × Ì Go Break RstWon Reset Win Reg Watch Syn Trace Step CTrac CStep DI EI IntFig PUT Info Log Esc Stop

FlashROM内にロケートされたプログラムを選択します。

<動作状況>

- 1) 【書き込みプログラムの転送】
- 2) 【セクターイレーズ】
- 3) 【ユーザーモードでダウンロード】
- と、順次メッセージBOX表示され、処理が進行します。

00800	_Start	llo 💌	@ Src C Mix (	C Asm start7284h.c	• <b>[</b> ] 実行ステート	敵 高速 💌		
00000800:	24	VERTEL():				EF レジスタ SH	2/A	×
100000000:	26	858(	mov.1 8(vbrtbl,pc	:),r0 *); // VERTEL	SR	00F032	10 FPU	215
00000802:	27	asa("	ldo r0,vbr	0:	PC	000000800	PR OFFF80DE0	and (
00000804:	28	asa("	add \$4,r0	-): // SP	RO	000000000	R8 005010401	-3
10101808*	23	858(	nov. i Sru, sp	r) r8 "): // Baillan():	R1	0FFFC08D0	R8 01490F22F	-
:40800000	31	458("	isr 8r8	"):	P2	0000000EA	R10 000000000	-3
0000080C:	\$2	asa("	nop		0.0	8000000FA	P11 010000000	-1
0000080E:	33	asa("	sov.1 8(ssin.pc).	r8 "); // _main();	No D.4	000000000	P10 0001201	-3
00000810:	34	ase("	jap 8r8	*):	K4	OFFFEZUU4	K12 005010401	_
0000812:	35	858("	nop	0:	R5	OFFFE2008	K18 01490F82F	_8
10101814:	25	asa(	Jone Builles		R6	080000050	R14 000000000	_8
10000824:	38	asa("aa int	long main		R7	000000000	SP 0FFF90000	_8
00000828:	39	ase("vbrtb1;	. Ione VBRTBL	- Di	MACH	0303388B2	MACL 0AE4548D1	
0000082C:	40	1			GBR	000000000	VBR 000000000	- 8
	41	//*********	*****	****	TBR	0303388B2	RegisterBank	
	42	// Waitlms(	) デフォルトPLL 1Φ×	CLK=12.0×4		a construction of the		_
	43	//	キャウンユ悪功	******				
	45	void Waitlas(	)	// las 971-342- (	48,000MHz)			
00000830:	46	(						
10000830:	47	asa("	nov.1 8(tins.pc).	rt ();				
0000882:	48	asm(" wait:		);				
00000882:	49	asa("	dt r1	"); // 1 clock				
00000834:	50	858(	bra send	7; // 3 clock				
001 000001	1 000	- 10		C 000 00100000			are arrest unit at	Col.
Del papero	1 400	<u> </u>		Est horgonara			710-70#### 77F7	1-9
30300	- 0		メモリダンプ	char 🔹 16;	# 💌 🗆 周期 美	行回数	adr{0x0}	
00000000:		80 00 08	00 FF F9 00 00 00 00 0	8 50 00 00 08 50	PP			
0000010:		80 00 08	60 00 00 08 50 00 00 0	8 70 00 00 08 50F	·PP			
0000020:		00 00 08	50 00 00 08 80 00 00 00	8 30 00 00 08 50  P 8 50 00 00 08 50   P. 5	D D			
00000040:		00 00 08	50 00 00 08 50 00 00 0	8 50 00 00 08 50 Jun Prove	PP			
0000050-		00 00 08	50 00 00 08 50 00 00 0	8 50 00 00 08 50  PF	PP			
			and a second					

#### <最終確認画面>

全て正常終了しますと、オブジェクトが外部Flash ROMに登録され、このような初期画面になります。

# ⚠

\_\_\_\_ 2. 御願い

本説の方法で、フラッシュROM品種を追加した場合、必ず、プロジェクトのバックアップすることを御願いします。 今回は「c:¥Program Files¥Aone¥DEF¥rom\_custom」で作成する例で記述しましたが「¥rom\_custom」をホルダごと別のディレクトリに貼り付けても作成できます。 つまり、ユーザーアプリのプロジェクトごとに管理するのも一案かもしれません。追加作成したプロジェクトは、ユーザー様の責任のもとで管理願います。

以上で、外付けFlashROMの品種追加作業が終了です。