🎾 エーワン株式会社

Rev 1.00.00

Renesas RZ 用リモートデバッガ・ナノ(DEFnano)の初期導入と

操作手順の概略説明 (RZAT)

(ルネサス RZ/A1H、RZ/T1「Cortex-R4F」)

【追記】DEFnano Ver3.00より、MP-RZT1-01(RZ/T1)対応を追加する。

1. 「DEFnano(RZAT)」をインストールします。

ご購入CDに保存してある「DEFnano_Install」の「setup.exe」をクリックします。



後の操作は、操作案内に従い進めて下さい。

)) DEFnano - InstallShield	७४म-४	1 A	OHE 1	デスクトップトにこのアイコ
	InstallShield ウィザードを完了しました	I	1370 +	ンが登録される。
	InstallShield ウィザードは、DEFnano を正第にインストールしました。「完了」をクリックして、ウィザードを終了してください。			
	□ ブログラムを起動する			
R	☑ READMEを表示する			
	< 戻る(B) 売了(F) キャンセル	イン	マトー	ル終了画面



- 2. ハード環境を用意する。
 - 2-1. MP-RZA1H/**-01 基板の用意(電源 OFF の状態)



2-2. MP-RZT1-01 基板の用意(電源 OFF の状態)



2-3. USBケーブルのTypeA側をPCのUSB口に接続する。(USB2.0/USB3.0)



- 2-4. MCU基板のDC5V電源をONにする。 エニュメレーション処理を開始する。
- 2-5. PCにHIDドライバーが登録されるのを待ちます。



3. DEFnanoを起動し、デバッグ操作をする。

3-1. デスクトップ上のアイコンをクリックする。

または、「C:¥Program Files¥Aone¥DEFnano¥DEFnano.exe」を実行させます。



80000	• C Sr	C C Hix G Asm	- 1
20080000: 00220000	EOREO	10.12.10	
20080004: 00110040	SUBED	r0,r1,r0, ASR \$32	
20080008: 22080005	ANDCS	r0, r8, #0x5	~
2008000C: 01014000	TSTED	rl,rl (MCII 其板のデバッガバ	
20080010: EA33D471	R	=0x2007510C	
20080014: 7F9AEA31	SVCTO	0x3AEA31 ージョンが追加で表示さ	
20080018: 9070C14C	REPLICE	r12, sp, r12, ASR 112 ノコンパー 但パル C 3X 小 C	
2008001C: 21564922	CMPCS	18,19 LSR #18 hZ	
20080020: 08880002	STMEDIA	(8, {r]}	/
20080024: 00810A80	ADDEQ	10 x1, r0, L3L 12	
20080028: 20000100	ANDCS	r0, r0, r0, L3L 12	
2008002C: 0A000010	BEU	=0x20080074	_
20080030: 68811FB/	SIMVSIA	- [P7/A1H] 失雨アドレス [0x2008000]	
20080034: 370AAF33	SMLADXUG		
20080038: 52241121	LOWLOTA		
20000030. 30702073	LUNLOIA		
20080044- 00816003	ADDED	からの道アセンブラを表示する	
20080048: 48080010	STMIDA		
2008004C: 19000822	STMNEDB	(r1,r5,r11)	
20080050: 8FDEFCFA	SVCHI	0x06FDFA	_
20080054: 551F8DB1	LDRPL	r8,=0x2007F24B	
20080058: 58329636	LDWPLDA	r2!, {r1, r2, r4, r5, r9, r10, r12, pc}	
2008005C: 1C00A3E0	STCNE	p3,c10,[r0], 1 -0x380	
SB1 00000000	CLR 💌 🤆 SB	82 000000000 CLR - 実行回数 1 ソフトプレーク S	VC c
	▼ (なし	-) - char - 16進 - 「周期	
	1 1		-
Go Break RstMon	¥in Reg	Vatch Sym Trace Step CTrac CStep IntFlg ProF	

3-2. ダウンロード方式を設定する。 【ファイル】-【アブソリュートファイル設定】

ダウンロード(D)	View CPURZ/A1H(R7S721000) Advance13.3333 b:30 TOP 0x20080000 SIZ 0x2097FFFF TargetMemory	
アップロード(U)	C Src C Mix € Asm ▼ 高速 ▼	
ベリファイ(V)	30 r0,r2,r0	
シンボル聴いみ(Y)	0 r0.r1.r0, ASR #32	
>>/////	2S r0, r8, 1 0x5	
アブソリュートファイル設定(A)	E0 r1,r0	
	=0x2007510C	
CPU設定読込み(S)	VC 0X34E431	
CPU設定登録(R)		
	05 10,12, Lon allo	
スクリプト実行(L)		
and a summer		
オフライン作業(0)	=0×20080074	
オフライン環境設定(M)	VSIA r1. {r0-r2.r4.r5.r7-r12}	
1	ADXCC r10,r3,pc,r10	
終了(E)	L p1.10x2.c1.c4.c1.11	-
2008003C: 98F5ECF3	LDHLSIA r5!. {r0.r1.r4-r7.r10.r11.sp-pc}*	
20080040: 20040400	ANDCS r0, r4, r0, LSL #8	
20080044: 00816003	ADDE0 r6,r1,r3	
20080048: 48080010	STMMIDA r8, [r4]	
2008004C: 19000822	STMNEDB r0.[r1.r5,r11]	
20080050: 8FDBFCFA	SYCHI 0xDBFCFA	
20080054: 551F8DB1	LDRPL r8.=0×2007F2AB	
20080058: 58329535	LDMPLDA r21, (r1, r2, r4, r5, r9, r10, r12, pc)	- 18
2008005C: ICOUASED	STUNE p3,c10,[r0], 1-0x380	
SB1 000000000	· CLR · C SB2 000000000 · CLR · 実行回数 · ソフトフィレーク SWC	3 code
	▼ ((なし) ▼ [char ▼ [16]# ▼ □ EM	
		- 2
		. 53
		1
Go Brook Rot Non	Win Pag Watch Sun Traca Stan CTrac CStan IntEl Prof.	100
do break inscholl	The net fatch of the oter offat offer the term	-



- ①「[-r]ディレクトリ情報を作成しない」
 ☑するとCソースデバッグが不可になる。
- ②「[-i]重複モジュール情報をを削除する」 時として不整合な情報が生成された場合に
- ③「[-s]ラインシンボル情報をソートしない」旧ELFバージョン時に追加、現状不要
- ④「[-f]展開可能なインクルードファイルをCViewに登録」*.h等の取得可能なファイルを登録

```
3-3.サンプルプロジェクトをダウンロードする。
3-3-1.【ファイル】-【ダウンロード】
```

ダウンロード(D)	View CP	URZ/A1H(R7S721000) Advance13.3333 1x30 TOP 0x20080000 SIZ 0x2097FFFF TargetMem	ory .
アップロード(U)	C Src	C IIIX @ Asa ▼ 高速 ▼	
ベリファイ(M)	0	r0,r2,r0	
5 5 - Fu 1913 3 (V)	50	r0.r1.r0, ASR \$32	-
シンパンレジロムの(1)	23	r0,r8, 1 0x5	112
アブソリュートファイル設定(A)	EQ	r1,r0	
	10	=0x20D751DC	
CPU設定読込み(S)	C	0x9AEA31	
CPU設定登録(R)	-22	r12,sp,r12, ASK #2	
	FOTA	resres con are	
スクリプト実行(L)	0	rfl.rfl. [S] [2]	
stance (s. frances)	S	r0, r0, r0, LSL 12	
オフライン作業(0)		=0x20080074	
オフライン環境設定(M)	VSIA	r1, {r0-r2, r4, r5, r7-r12}	
44-74-14	ADXCC	r10,r3,pc,r10	
終了(E)	PL	p1,#0x2,c1,c4,c1,#1	
2008003C: 98F5ECF3	LDWLSIA	r5!, {r0, r1, r4-r7, r10, r11, sp-pc}*	
20080040: 20040400	ANDCS	r0,r4,r0, LSL 18	
20080044: 00816003	ADDEO	r6,r1,r3	
20080048: 48080010	STMMIDA	r8, (r4)	
20080046; 13000822	STMINEUB	r0, {r1, r0, r11}	
20080050; SPUDPUPA	SYUNI	0X00F0FA	
20080034. 33176001	LONPL	21 [e1 e2 ed e5 e9 e10 e12 pe]	
20080050: 10004350	STONE	p3.c10.[r0]. #-0x380	
SB1 00000000 -		0000000000 ▼ CLR ▼ 実行回数 1 ソフトフジレ-ク	SVC code
	▼ (なし)	▼ char ▼ 16進 ▼ 「周期	
			*
			8
			-
Go Break RstMon	Vin Reg	Vatch Sym Trace Step CTrac CStep IntFlg ProF	
inter transitional and the	and the second sec		and an owner of the second



3-3-2. シンボルコンバータの起動

・指定拡張子により、起動するコンバータを自動判別します。

1) ARMC 「*.axf」 ARM社

ACsymconv	
GetDebugLocAll() GetDebugInfoAll()	<u> </u>
	2

【起動】 ACsymconv.exe

2) ICC 「*.out」 IAR社

IARsymconv	
GetSection() GetDebugLocAll() GetDebugInfoAll()	*

【起動】 IARsymconv.exe

GCsymo	onv			X
¥HardwareDe GetSection()	bug¥Demo	_NORTI_R4F.x -	adef -i -g	*
GetDebugLoc/	AllO			-
GetDebuaInfa	AllO			8

【起動】 GCCsymconv.exe

4) 共通の注意事項

各コンパイラによって、生成される ELF/Dwarf2 コードに相違があります。 特に、IAR・GCC に関しては、独自の非公開 Dwarf コードを生成しているため、複雑なシ ンボル構造の場合、コンバート出来ない場合があります。特に多重構造体の場合は、分析 不可となり、シンボルがカットされます。

3-3-3. プログラムコードのダウンロード



▲
 再ダウンロード時の注意事項

USB-CH1(ITF_LIB)を使用したアプリ開発時に再ダウンロードする場合、出来るだけ USB-CH1 側のケーブルを抜いた状態で実施して下さい。理由は、ダウンロード完了時にターゲットをリセ ットする仕組みになっていますので、ケーブルを刺した状態だと Windows が USB-CH1のドライ バー切り離し処理とデバッガ側の USB(HID)のエニュメレーション処理を開始しますので完了ま で通常より時間が必要になります。



3-3-4. プログラム実行とブレーク設定と強制停止方法





🌽 エーワン株式会社



・ステップ/トレース機能を使用する場合は、<オプション>-<環境設定>の「Page1」
 「トレース/ステップ実行の動作」を「許可」にチェックして下さい。

▲ RTOS 下での Trace(トレース)/CTrac(Cトレース)使用時の注意事項

RTOS のカーネルは、SWI(SVC)を使用して実現しています。本リモートデバッガは、当命令 への対応はしておりません。実行ポイントがカーネルへの関数になった場合は、Step(ステッ プ)/CStep(Cステップ)に切り替えて操作して下さい。







····································	me@vin.c		構造体や多次元配列の場合は、「評価/変更」を使用す
Data Address[20	007020]		る。
frame_memory	== 0×60800000	Dec=1619001344	
frame_width	== 0x00000320	Dec=0000000800	
frame_height	== 0×000001E0	Dec=0000000480	
frame_line_offset	== 0x00000640	Dec=0000001600	
frame_offset	== 0x000BB800	Dec=0000768000	
frame_num	== 0x00000000	Dec=0000000000	
swap	== 0x00000000	Dec=0000000000	
rotate	== 0×00000000	Dec=0000000000	
format	== 0x00000001	Dec=0000000001	
変更値			
		*	

▲周期サンプリング使用時の注意事項

周期サンプリングの仕組みは、ユーザープログラムの実行中に USB0 の IRQ 割り込みを利用し てデータを取得しています。USB 割り込みの特性上、1 バイト~64 バイトまでのデータ情報の 取得に「250usec~500usec」の割り込み処理が実行されます。 この周期サンプリングがアプリ側処理の妨げになる場合は使用しないで下さい。

3-3-6. その他のデバッグ操作

その他、詳細なデバッグ操作は【ヘルプ】--【ヘルプ】をご覧下さい。

19止 書队	L EI P	C 0×20080000	「周期「	View CP	URZ/A1H(R7S72	1000) A	CPU別デバッ	ガ仕様(C)	0	IOC7CE3 TargetMemor	у
0000000	\$35:	109 ***	- December 1		⊂ Mix C	Åsn	_vect	日本/英語切	替え(J)		Search{LDR}	
20080004:	36	LOR pc	. =Undefin	ed handler		:	Start+	ASCII-	(A)			-
20080008:	37	LDR po	, =Svc_har	dler		1	Start+	COTHER P	SX(A)			
2008000C:	38	LDR pc	, =Prefeta	h_handler		:	Start+	SCINGENDE	BKKT#(5)			
20080010:	39	LDR pc	, =Abort_!	andler		;	Start+	エーワンのオ	- ムページ(A)			
20080014:	40	LDR pc	, =Reserve	d_handler		;	Start+					
20080018:	41	LOR po	, =Irq_har	dler		:	Start+	ヘルプ(H)	\rightarrow			
20080010:	42	LUR pc	, =Fiq_har	dler		1	Start+L					
K0080020:	43	,	- Tadi da	7. 42. 1=36								
	44	; SFHOM	「宣称した」	コーターに渡	9 1首早位							
	40	Info table										
	40	Into_table	I Tanan deus		4Dacal		Ci	20 + rt+誌QAM書	お美生の問題でも	1.7		
	47	000	I Inage \$\$D	TASSI init I	Moasel	- :	Start+0v00	24 · 内部RANA	ジェルの線ファト	1.7 (+1)		
	49	000	vector tal	le		-:	Start+0x00	28 : 27期月中に信	a)2500547171	PAGO		
-	50	Info end	receior_ca				01011110200					
	51	;********										
	52	Literals										
	53	LTORG										
	54											
	55	END										
-												
SB1 000000	000		▼ CLR		000000000			▼ CLR ▼	実行回数 1	_	ソフトプレーク B	RK cor
	-		_	1/(411.)		-		lla civi			ana baanaanaana	
1				■(7a(C))		char		16週	三日期			
	nend Second	1	1	1 - 1		12	1.2	1		1	In the second	
				D		Corre	T at a s	C4	T			

4. デバッグが完了しシリアルフラッシュ ROM にプログラムを書き込む方法 【オプション】- 【フラッシュ ROM ライタ】

00: 8			• arc	CMix	CPU 設定(C)				
04: 3	IS LDR	pc, =Reset_h	andler ed handler	-	フラッシュ	、 LROMライタ(F)	>			-
08: 3 0C: 3	37 LDR 38 LDR	pc, =Svc_han pc, =Prefetc	dler h handler		ファーム	(ージョンアップ(V))	みト		
10: 3	39 LDR	pc, =Abort_h	andler	-	; Start+	H0x0010 : データア	ボート			
14: 4	40 LDR	pc, =Reserve	d_handler		; Starts	H0x0014 : Reserved				
18: 4	41 LDR	pc, =lrq_han	dler		; Starts	H0x0018 : IRQ				
10: 4	42 LDR	pc, =Fig_han	dler		; Starts	HOX001C : FIQ(NWI)				
20: 4	43 ;======							13.		
4	44 ; SFRC	加に登録したロ	コーダーに測	見す情報						
4	45 ;======									
4	46 Info_tab	ole								
4	47 DCD	Inage\$\$VE	CTOR_TABLE	\$\$Base	; Starti	H0x0020:内蔵RAM動	記送先の間	始アドレス		
4	48 DCD	Inage\$\$DA	fA\$\$Limit		; Start+	H0x0024:内蔽RAM	記送先の終	了アドレス(+1)	
4	49 DCD	vector_tab	le		; Start+	HOx0028:初期PC值				-
ŧ	50 Info_end	ł								
	51 ;======									
1	52 Literals	8								
	53 LTOP	łG								
1	54									
	DD ENU									
0000000	1	• CLR	- C \$82	000000000		▼ CLR ▼	実行回義	y 1	<u> ソフトフィレーク</u>	BRK code
1			 (なし) 	•	char	▼ 16進	- 「唐	INI		
1.000				Bartet .		Ibees	illes division			3
										2
	1000 1 1000 1 1 1 1 1 1 1 1 1 1 1 1 1 1	1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.	1 244	The state of a state of the state of the		the second se	24 million 1 million		A designed to the second	
		00: 38 LDR 10: 39 LDR 10: 49 LDR 11: 40 LDR 12: 41 LDR 10: 42 LDR 20: 43 :====== 44 : SFRC 45 :====== 46 Info_tat 47 DCD 48 DCD 50 Info_enc 51 :===== 52 Literals 53 LTDF 54 55 END 10000000	00: 38 LDR pc, =Prefetci 10: 39 LDR pc, =Rosert_b 14: 40 LDR pc, =Rosert_b 14: 40 LDR pc, =Fre_han 10: 42 LDR pc, =Fre_han 10: 42 LDR pc, =Fre_han 10: 42 LDR pc, =Fre_han 44 ; SSROWE 200 LC. E 45 ;	00: 38 LDR pc, =Prefetch_handler 10: 39 LDR pc, =Abort_handler 11: 40 LDR pc, =Reserved_handler 12: 41 LDR pc, =Freq_handler 13: 41 LDR pc, =Freq_handler 14: 40 LDR pc, =Freq_handler 14: 41 LDR pc, =Freq_handler 14: 55 FROMES2834.CTOR_TABLE 43 DCD Insce353VECTOR_TABLE 44 DCD Insce353VECTOR_TABLE 45 DCD vector_table 50 Info_end 51 :====== 52 Literals 53 LTORG 54 55 END 00000000 ▼CLR ▼ C S82	00: 38 LOR pc, =Prefetch_handler 10: 38 LOR pc, =Abort_handler 14: 40 LOR pc, =Reserved_handler 18: 41 LOR pc, =Fre_handler 10: 42 LOR pc, =Fre_handler 44 : SFROWE 28&\LCLC - 5'	00: 38 LDR pc. =Prefetch_handler () / LN 10: 38 LDR pc. =Abort_handler : Start4 14: 40 LDR pc. =Reserved_handler : Start4 18: 41 LDR pc. =Reserved_handler : Start4 10: 42 LDR pc. =Frig_handler : Start4 44 : SFROMIで登録したローダーに注意す情報 45 :====================================	00: 38 LOR pc, =Prefetch_handler	00: 38 LCR pc, =Prefetch_handler () シアーム(->= ア・ワッ()) 10: 38 LCR pc, =Abort_handler ; Start+0x0010 : データアネート 14: 40 LCR pc, =Beserved_handler ; Start+0x0018 : IR0 110: 41 LCR pc, =Freyhandler ; Start+0x0018 : IR0 110: 42 LDR pc, =Freyhandler ; Start+0x0010 : F10(NMI) 111: 44 : SFROMEZ@&U,CL-ローグーニ2渡す情報 45 :====================================	00: 38 LDR pc, =Prefetch_handler アンテム(モンヨンアタン(ア) ト 10: 38 LDR pc, =Abort_handler ; Start40:00101 : データアネート 14: 40 LDR pc, =Reserved_handler ; Start40:0018 : Reserved 18: 41 LDR pc, =Freyhandler ; Start40:0018 : IR0 10: 42 LDR pc, =Freyhandler ; Start40:0010 : FIO(NMI) 20: 43 : 44 : SFROME-28&/LCLダー-1:渡す4846 45 : 46 Info_table 47 DCD [Image\$\$VECTOR_TABLE\$\$Base] : Start40:0020 : Pi酸RAMBE送先の開始にアドレス 48 DCD Vector_table ; Start40:0020 : Pi酸RAMBE送先の開始にアドレス 49 DCD Vector_table ; Start40:0028 : PIMBRAMBE送先のポアドレス 50 Info_end 51 : 52 Literals 53 LTURG 54 54 55 END 50 DOCUMENT (12) Char (16) / IGM	00: 38 LDR pc. :Prefetch_handler アアムパーンランアクパート 10: 38 LDR pc. :Rbort_handler : Start+0x0010 : データアボート 11: 40 LDR pc. :Pregrammander : Start+0x0011 : Reserved 110: 41 LDR pc. :Fiq_handler : Start+0x0011 : FIQ(NHT) 110: 42 LDR pc. :Fiq_handler : Start+0x0011 : FIQ(NHT) 111: 44 : SFROMT空發化たローダーに渡ず情報 45 :====================================

4-1. RZ/A1Hの場合





😋 🔵 🗸 📕 « USER_Debug 🕨 Debug	► ▼ 4 ₂ 1	Debugの検索	2	ライターエード
整理 ▼ 新しいフォルダー		88 • 🗖 🌘		$\frac{1}{1}$
USER_Debug	🌲 src_app			
🍶 Debug	src_sys			Hex ノアイルを打
🎉 src_app	USER_Debug.	mot		定する。
🔒 src_sys				
Ja log				
scatter_file				
🎉 script				
🍶 src_app				
🔒 inc				「盟く」をク
🎉 src_sys	-			
ファイル名(N): USE	R_Debug.mot 👻 H	ex File (*.hex,*.mot)		<u> </u>
	(問く(0) キャンセル	1	









シリアルフラッシュ RO	Mマップ(MP-RZA1H/**-01)
ローダ・デバッグ用ファーム	0~1セクター
	$0x0 \sim 0x1$ _FFFF
アプリケーションエリア	2~153セクター
	0x2_0000~0x99_FFFF

4-2. RZ/T1の場合









RZ/A1H RZ/ Cortex-M3(* m	T1 not/*hex)	
		参照
i Start	end Size	
Cortex-R4F(*.	mot/*hex)	
✓ N:¥Usri	Ap¥C_H28_AICHI¥RZT1¥Sample_IAR¥RZT1_Demo_	NORTi¥R4F¥Debug¥Exe¥De 参照
Start 0.v0	and Dy42dbd Siza Dy42da0	
indi i foxo	enu jox43000 Size jox430c0	
All the second		
込み個数 1	きには しょううしか ちに そうしました。	書込み開始
込み個数 1 消去個数 0	きには しんそうない ろしてき しんそうなし きょう うちょう しんそう しょう しんそう しょう しょう しょう しょう しょう しょう しょう しょう しょう しょ	書込み開始 全消去開始
込み個数 1 消去個数 0	きには しんそうしい うしてき しんそうはこ きょう しんそう しんそう しんそう しんそう しんそう しんそう しんそう しんそ	書込み開始 全消去開始
込み個数 1 消去個数 0	きには 10249000 3128 10249000	書込み開始 全消去開始
込み個数 1 消去個数 0	enu px+ouud Size px+oucd	書込み開始 全消去開始
込み個数 1 消去個数 0 正常終	eriu jox43000 書込みは正常に終了しました。 了しますと、「書込みは正常に約	書込み開始 全消去開始 終了しました。」と表
込み個数 1 消去個数 0 正常終 示され、	erru jox+3000 書込みは正常に終了しました。 了しますと、「書込みは正常に終 、個数がカウントアップします	 書込み開始 全消去開始 注消去開始 終了しました。」と表 「。 個数は、この窓を
込み個数 1 満去個数 0 正常終 示され、 閉じま	erru jox+3000 Size jox43000 書込みは正常に終了しました。 了しますと、「書込みは正常に新 、個数がカウントアップします すとゼロになります。	 書込み開始 全消去開始 除了しました。」と表 で。個数は、この窓を



シリアルフラッシュ I	ROM マップ(MP-RZT1-01)
ローダ・デバッグ用ファーム	0~1セクター
	0x0~0x1_FFFF
アプリケーションエリア	2~15セクター
Cortex-M3	0x2_0000~0x9_FFFF
	0xA_000~0xF_FFFF(予備)
アプリケーションエリア	16~31セクター
Cortex-R4F	0x10_0000~0x17_FFFF
	0x18_0000~0x1F_FFFF(予備)

 ローダーとデバッガファームのバージョンアップ方法 【オプション】-【ファームバージョンアップ】

停止 割込 EI PC 0x2008	0000 명배 View C	CPU RZ// 環境設定(E) 0080000 SIZ 0x2097FFFF TargetMemory .
0080000	· C Sri	rc C Wix CPU設定(C)
20080000: E59FF024	LDR	PC,=0 フラッシュROMライタ(F)
20080004: E53FF024	LDR	pc,-u
2008000C: E53FF024	LDR	PC.=0 ファームバージョンアップ(V)
20080010: E59FF024	LDR	pc,=0x2008003C
20080014: E59FF024	LDR	pc.=0x20080040
20080018: E59FF024	LDR	pc.=0x20080044
2008001C: E59FF024	LDR	pc,=0x20080048
20080020: 20080000	ANDCS	r0,r8,r0
20080024: 200828F8	STRDCS	r2,r3,[r8], -r8
20080028: 20080000	ANDCS	r0, r8, r0
2008002C: 2008014C	ANDCS	r0.r8.r12, ASR \$2
20080030: 200805E8	ANDCS	r0,r8,r8, ROR #11
20080034: 2008047C	ANDCS	r0,r8,r12, ROR r4
20080038: 20080490	MULCS	r8, r0, r4
2008003C: 200805EC	ANDCS	r0,r8,r12, ROR \$11
20080040: 200805F0	STRDCS	r0.r1.[r8]r0
20080044: 200804F8	STRDCS	r0,r1,[r8], -r8
20080048: 200805E4	ANDCS	r0,r8,r4, ROR \$11
2008004C: 00000000	ANDEO	r0, r0, r0
20080050: 00000000	ANDEO	r0,r0,r0
20080054: 00000000	ANDEO	r0, r0, r0
20080058: 00000000	ANDEO	r0, r0, r0
	Auren la an	
. 281 hononon 1		22 000000000 「ICLR I 実行回転 <u>ソフトフィルク</u> BRK co
and a	▼ (なし)	-) - char - 16道 - 日間
1000		
C. Durch Decker	Bin Dan	I work on I take of a lotar logar liter to be the
GO Dreak Astion	win Keg	; match oym frace step Girac Gotep [Intrig ProP

	- neo y bernano y	An_vor	• • • • • • •	CV0P0/1878	7	
整理▼新しい	\フォルダー)III •	0 0	
名前	更新日時	種類	サイズ			
AHmoni.bin	2015/03/05 9:26	BIN ファイル	53	КВ		
[RZ/	T1] AHmoni_]	R4.bin				
のバ	1+11-7-1	ルな世				

【RZ/A1H】

🕒 🔵 💌 📕 🖌 新しいフ:	オルダー + AH_VUP	▼ 4 ⁴	AH_VUP \$
ファイル(F) 編集(E) 表示	テ(V) ツール(T) へル	プ(H)	
整理 • ライブラリに追	跏 ▼ 共有 ▼ 🕴	書き込む » ※	• 🗊 0
名前	更新日時	種類	サイズ
AHmoni.bin	2015/03/05 9:26	BIN ファイル	53 KB
CODE_SPIBSC_INIT	2015/04/01 10:23	コアイル	3 KB
VECTOR_TABLE	2015/04/01 10:23	コアイル	1 KB

<AH_VUP>のファイル名 ①AHmoni.bin ②CODE_SPIBSC_INIT.bin ③VECTOR_TABLE.bin の3ファイルになります。

[RZ/T1]

	ID T1]	
		• •	An_vor >
ファイル(F) 編集(E) :	表示(V) ツール(T)	ヘルプ(H)	
整理 ▼ ライブラリ(こ追加 ▼ 共有 ▼	»	• 🔟 🔞
名前	更新日時	種類	サイズ
Loader_R4.bin	2016/12/16 18:10	BIN ファ	5 KB
AHmoni_R4.bin	2017/02/17 13:35	BIN ファ	30 KB

<AH_VUP_T1>のファイル名 ①AHmoni_R4.bin ②Loader_R4.bin の2ファイルになります。







6. その他の注意事項

- 1) ファームのバージョンアップ中に USB ケーブル等が抜けて失敗した場合は、内蔵 RAM に ファームが常駐していますので、リセット及び電源の再立上げ等を実施しないでバージョ ンアップ操作を再実施して下さい。
- DEFnano デバッガは、USB エニュメレーションを実施するタイミングはターゲットの電源 ON 時のみに処理する仕組みになっていますのでデバッガを有効にしたい場合は電源 ON 時 にターゲットの USB ロにケーブルを刺した状態でいて下さい。
- デバッガを切り離してシリアルフラッシュ ROM ブートで動作させたい場合は、ターゲット 側の USB 口からケーブルを抜いた状態で電源 ON しますとローダーが判断して ROM 側の プログラムを内蔵 RAM に展開して RUN させる仕組みになっています。
 【RZ/T1】特記 MP-RZT1-01の場合は、SW2-3を OFF にして下さい。(USB デバッグモニタ未使用)
- 4) プログラムのデバッグ中は、完成されていないプログラムのため、プログラムミスにより閉 ループに入り抜けてこなくなる場合もあります。その時に割り込み禁止状態に陥り、リモート デバッガの性質上、強制 Break さえ効かなくなる場合もあります。このような状態に陥った 場合は、MCU 基板上にある「ResetSW」を押して、初期状態からデバッグを開始して下さい。

6. DEFnanoを使用する場合のアプリケーション側ソフトの手続きについて

1) デバッグモニタは、割り込みは[Prefetch]と[IRQ]を利用しています。

▲注意事項

・IRQ_SPとABT_SPは、別空間になるようにスタックポインタを設定して下さい。

2) Prefetch・IRQハンドラーには、デバックモニタ用マクロ定義が必要になります。

Prefetch_handler BRK 割り込みハンドラー	Start+0x000C: プリフェッチアボート
ch_Handler	
<mark>SUB Ir, Ir, #4</mark>	;;; [-4]する必要あり BRK コードの PC 位置を知るため
GET_REGSTER #ABT_MODE, Abt_e	 <mark>and</mark> ;;; モニタ用にレジスタ数値をメモリにストアする。
SRSDB sp!, #ABT_MODE	
PUSH {r0-r12}	;;; レジスタを退避
<mark>BL SwiMoniter</mark>	;;; デバックモニタ処理
CPS #ABT_MODE	;;; ABT Mode
POP {r0-r12}	::: レジスタの復帰
KFEIA SP!	割り込みの復帰処理
IRQ Handler	Start+0x0018 : IRQ
ndler	
<mark>sub Ir, Ir, #4</mark>	
<mark>GET_REGSTER #IRQ_MODE, Irq_</mark> e	and ;;; モニタ用にレジスタ数値をメモリにストアする。
↑ 先頭に定義	
(† っ	

【RZ/A1H】「src_app¥_init_handlers_s..s」を参照

【RZ/T1】「R4F¥src¥monitor¥inter_handler.asm」を参照

3) デバックモニタプログラムの初期化する関数をメインの先頭で呼ぶ手続きが必要です

/ ①RTOS(NORTi) 田の記述例	Nmoniter clのリンクが必要!!
/*************************************	
include " <mark>Nmoniter.h</mark> "	// RTOS(NORTi)用のヘッダーファイル
/	
/ main関数	
/	
sysini();	// システム初期化
-	
MoniterInit();	// <mark>モニタの初期化(必ず sysint()の後で!!)</mark>
/	
/	
/ / ******************************	^{wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww}
/ /***********************************	^{はたののリンクが必要!!}
/ /******************************	⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰⁰
/ /******************************	^{bybyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy}
/ / ②ベアメタル用の記述例 /************************************	^{INFORMATION CONTRACTOR OF CONTRACTOR CONT}
/ / ②ベアメタル用の記述例 / ********************************** include <mark>~Umoniter.h</mark> ~ / / main 関数	www.www.www.www.www.www.www.www.www. [Umoniter.c]のリンクが必要!! ^{www.www.www.www.www.www.www.www.www.ww}
/ / ②ベアメタル用の記述例 / ************************************	^{kththththththththththththththththththth}
/ /***********************************	Communitien. c]のリンクが必要!! Chroniter. c]のリンクが必要!! // ベアメタル用のヘッダーファイル
/ / ②ベアメタル用の記述例 /************************************	<pre> // モニタの初期化(必ず main()の先頭で!!) // マリンクが必要!! // マリンクが必要!! // ベアメタル用のヘッダーファイル // マリンクが必要!! // モニタの初期化(必ず main()の先頭で!!)) // マリンクが // マリンクが必要 // モニタの初期化(必ず main()の先頭で!!)) // マリンクが // マリンクが // モニタの初期化(必ず main()の先頭で!!) // マリンクが // マリンク // マリンクが // マリンク // マリンクが // マリンク // マリンクが // マリンク // マリンクが // マリンクが // マリンク // マリンク // マリンク // マリンクが // マリンク // マリンクが // マリンク // マリ</pre>
/ /***********************************	Comparison Comp
/ / ②ベアメタル用の記述例 /************************************	<pre> cv>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>
/ /*******************************	<pre> c>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></pre>

1) Nmoniter.h/Umoniter.h は、サンプルで用意してあります。そのまま利用する。

2) コメントに記述してある注意事項に従ってポーティングする。

以上です。

- 7. 注意事項
 - ・本文書の著作権は、エーワン(株)が保有します。
 - ・本文書を無断での転載は一切禁止します。
 - ・本文書に記載されている内容についての質問やサポートはお受けすることが出来ません。
 - ・本文章に関して、ARM 社およびルネサス エレクトロニクス社への問い合わせは御遠慮願います。
 - ・本文書の内容に従い、使用した結果、損害が発生しても、弊社では一切の責任は負わないもの とします。
 - ・本文書の内容に関して、万全を期して作成しましたが、ご不審な点、誤りなどの点がありましたら弊社までご連絡くだされば幸いです。
 - ・本文書の内容は、予告なしに変更されることがあります。

8. 商標

- ・ARM DS-5は、ARM 社の登録商標、または商品名称です。
- ・EWARMは、IAR社の登録商標、または商品名称です。
- ARM Cortex, Thumb および ARM Cortex-M3/R4F は ARM Limited の EU およびその他の国に おける商標および登録商標です
- ・RZ および RZ/A1H は、ルネサス エレクトロニクス株式会社の登録商標、または商品名です。
- ・その他の会社名、製品名は、各社の登録商標または商標です。

9. 参考文献

- ・「RZ/A1H グループ ユーザーズマニュアル ハードウェア編」 ルネサス エレクトロニクス株式会社
- ・「RZT1 グループ ユーザーズマニュアル ハードウェア編」 ルネサス エレクトロニクス株式会社
- ・ルネサス エレクトロニクス株式会社提供のサンプル集
- ・「armcc ユーザガイド DUI 0472JJ」 ARM 社
- ・「アセンブラの使用 DUI 0473GJ」 ARM 社
- ・「リンカの使用 DUI 0474GJ」 ARM 社
- ・「コンパイラリファレンスガイド DUI 0328BJ」 ARM社
- ・「アセンブラリファレンス DUI 0489GJ」 ARM社
- ・「armkink リファレンスガイド DUI 0804AJ」 ARM 社
- ・「IDE プロジェクト管理およびビルドガイド UIDEARM-9j」 IAR 社
- ・「IAR C/C++開発ガイドコンパイラおよびリンク DARM-14-j」 IAR 社
- ・「IAR アセンブラリファレンスガイドAARM-9j」 IAR社
- ・「IAR デバッグプローブガイド IAR probes-2j」 IAR 社
- ・その他

 $\mp 486-0852$

愛知県春日井市下市場町6-9-20 エーワン株式会社 http://www.robin-w.com