Rev 1.10.00

IAR(EWARM)ツールチェインの設定と必要事項の説明

(ルネサス RZ/A1H用)

IAR(EWARM)ツールチェインの設定方法とサンプルプロジェクト「IAR_debug_rz」に必要な 設定を説明します。

1. IAR Embedded Workbench IDE を起動する。



2) ワークスペースファイル「IAR_debug_rz.eww」を選択する。

10			where a state of the state of t			Land and the second
整理 -	新しいフォルダー					· · · ·
	📕 IAR Workspace	+	名前	更新日	89	種類
	🗼 IAR_debug_rz		Le config	2016/	01/29 13:51	ファイルフォル
	🔔 config		L Debug	2016/	01/28 17:01	ファイルフォル
	📕 Debug	(3)	L settings	2016/9	1/28 18:37	ファイルフォル
	i settings		src_app	2016/	02/04 11:33	ファイル フォル
	arc_app		🗼 src_sys	2016/	02/04 10:59	ファイルフォル
	arc_sys		IAR_debug_rz.eww	2016/01/38 17:15		IAR IDE Worksp.
	PC_Test					
	RX Program		•			
×	IAR_debug_rz.eww IAR IDE Workspace	状 更新日	兄: 24 共有 時: 2016/01/28 17:15	サイズ 作成日時	: 917 /(귀ト : 2016/01/28	17:15
	ファイル名(N)	IAR.	_debug_rz.eww		ワークスペー	·スファイル(*.ew・

2. 各ツールの設定内容を確認する。



- 2-1. 一般オプションの確認
 - 1) ターゲット

カテゴリ: - 10オブション 静的5解析							
ランタイム解析 C/C++コンパイラ		ターゲット	出力	ライブラリ設定	ライブラリオプション	MISRA-C:2004	MISRA-C: 1998
アセンフラ 出力コンバータ カスタムビルド ビルドアクション リンカ	101	ブロセ: © コフ () デ	ッサ選択 P(O) バイス(D)	Cortex-A9 Renesas F	- R75721000		
デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet		1ンデー ③ リト ● E: ●	ィアンモー -ル(L) -グ(B) BE32(3) BE8(8)	ド :浮動 () () () () () () () () () () () () ()	小数点演算の設定 I(F) VFPv ジスタ(R) 32 Itvanced SIMD (NEC	3	

2) 出力

カテゴリ:							
 - 長オブション 		ターゲット 出力2 東ララ 出力 実 フラ 日 方 行 行 し 世 リスト Det	出力 アイル(0) 行可能 イフラリ(1 可能ファ マレクトリ 可能ファ マレクトリ 可能ファ マレクトリ マレクトリ マフラリ(1 マークトリ マンクトフ ファイル いな¥Exe ファイル いな¥Exe ファイル マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトノ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトノ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクトリ マンクト マンクト マンクト マンクトリ マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マンクト マン マント マン マン マン マン マン マン マン マン マン マン	ライブラリ設定 ファイル(E)) マイル/ライブラリ() ァイル(B): (S):	ライブラリオブション 0:	MISRA-C:2004	MISRA-C: 1998
Macraigor PE micro	-				1	ОК	キャンヤル

3) ライブラリ設定

ノード"IAR_debug_rz"のオプショ	عک ا
ノート IAR_debug_rz のオノショ カテゴリ: <u>一般オプション</u> 静的解析 ランタイム解析 C/C++コンパイラ アセンブラ 出力コンバータ カスタムビルド ・	ターゲット 出力 ライブラリ設定 ライブラリオブション MISRA-C:2004 MISRA-C:1998 ライブラリ(1) シーマル C/C++ランタイムライブラリの)通常の設定を使用します。ロ ケールインタフェースなし、Cロケール、ファイル記述子サポー トなし、printf/scentでのマルチバイト文字なし、strtodでの
ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace TI Stellaris	16道鉄字動小数点数なし。 設定ファイル(C): \$TOOLKIT_DIR\$¥INC¥c¥DLib_Config_Normalh ライブラリのスレッドサポートを有効にする(E) 低レベルインタフェースのライブラリ実装(B) ● なし(N) ● セミホスティング(S) ● セミホスティング(S) ● ロミホスティング経由 ● SWO経由
Macraigor PE micro	 OK キャンセル

4) ライブラリオプション

ノード"IAR_debug_rz"のオプシ:	a> 💽
カテゴリ:	ターゲット 出力 ライブラリ設定 ライブラリオブション MISRA-C:2004 MISRA-C:1998 Printfフォーマッタ ・ ・ フォーマッタの自動選択。 Scanfフォーマッタ ・ ・ ・ フォーマッタの自動選択。 ・ ・ ・ フォーマッタの自動選択。 ・ ・ ・ フォーマッタの自動選択。 ・ ・ ・ フォーマッタの自動選択。 ・ ・ ・
PE micro 🔻	OK キャンセル

5) MISRA-C:2004 (設定なし)

テコリ: - 股オプション 浄的解析						
ンタイム解析		ターゲット出力	ライブラリ設定	ライブラリオプション	MISRA-C:2004	MISRA-C: 1998
アセンブラ 出力コンバータ カスタムビルド		MISRA-Cを相 MISRA-C設定 MISRA-C:200	i対にする(E) 定のログ(L) は規則をアクティブ	@ 	MISRA-C:2004 MISRA-C:1998	
レルドアクション デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet	III .	VI.1: [required VI.2: [required VI.3: [required VI.4: [required VI.4: [required VI.4: [required VI.4: [require	All code shall All code shall No reliance s Multiple comp The compiler, Floating-poir Assembly lan Source code	v 9 ACCA l conform to ISO 98 hall be placed on u pilers and/or langue /linker shall be che t implementations guage shall be enc shall only use ISO9	Migs 1990 "Program indefined or unsp ages shall only b icked to ensure t should comply w apsulated and is 1899 1990 '0' styl	iming langu . ecified beh e used if th hat 31 char ith defined plated. e comment

6) MISRA-C:1998 (設定なし)

コテゴリ: 一般オプション #44527945						
シタイム解析	ターゲット	出力	ライブラリ設定	ライブラリオプション	MISRA-C:2004	MISRA-C:199
カスタムビルド ビルドアクション リンカ デバシガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace TI Shellarie	MISRA 73 73 73 73 73 73 73 74 74 74 75 75 77 75 74 75 74 75 75 75 75 75 75 75 75 75 75 75 75 75	-C: 199 duired] dvisory] dvisory] dvisory] dvisory] dvisory] muired] muired] muired]	8規則をアクティブ 必須(F All code shall o Code written in Assembly lang Provisions sho Only those char Values of char Trigranhs shall	に設定 の すべて(A conform to ISO 989 a languages other to uage functions that uid be made for ap racters and escape acter types shall be not be used	9 standard C, wit han C should onl t are called from propriate run-tim e sequences whic e restricted to a	th no exten by be used C should t the checking th are defin defined and t



2-2. 静的解析の確認

1) C-STAT 静的解析(設定なし)

カテゴリ:			
一般オブション	4		
ランタイム解析 C/C++コンパイラ		C-STAT書給り解释析 注意カルオプション	
アセンブラ 出力コンバータ カスタムビルド ビルドアクション	124	C-STATチェックを選択(S)	
リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ		設定をエクスポート(E) モジュールのタイムアウ 600 秒 (0はタイムアウトなし) パラレル解析を有効化 プロセス:2 この製品パージョンで使用可能なC-STATのライセンスがありません。	
I-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraigor			

2) 追加オプション(設定なし)

カテゴリ:				
一般オブション	-			
シンタイム解析 C/C++コンパイラ アセンブラ 出力コンバータ カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ	10	C-STATă¥Ġ为為なれ折 〕 3 □ コマンドラインオブジ コマンドラインオブジ	倉加オブション /ョンの使用(U) /ョン(C)(I行に1コマンド)	
I-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraigor				*

- 2-3. ランタイム解析の確認
 - 1) C-RUN ランタイム解析(設定なし)



- 2-4. C/C++コンパイラの確認
 - 1) 言語1

bテゴリ: 一般オブション 静的解析	-		, 皮棄			工場出	荷時設定
ランタイム解析 C/C++コンパイラ		言語 1 言語 2 コード	最適化出力	リスト	ブリブロセッサ	診断	MISF •
出力コンバータ カスタムビルド ビルドアクション リンカ	в	言語(L) ● C ● C++ ● 自動(拡張子ペー	2)(A)	言語の道 () 標準 () 標準 () 様常	<u>1合(N)</u> 1(IAR拉張赤/)) 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	>	
デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ		Cの派生言語(C) C89 C99 VLAの許可((C++ インライン プロトタイプの強制	の シ動作 KR)	C++の第 ② Emb ③ 拡張 ③ C++	生言語(+) edded C++ Embedded C+- 例外あり RTTIあり	÷	
I-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraigor PE micro				<u>[√]</u> 青争8*	オブジェクトを破	₩ 	le s les

2) 言語2

33 J 3							-	工場出	荷時設定
一般オブション 静的解析	Â	■ 複数ファイルのコ □ 未使用パブ	レパイル リックを破	糜					
209412647 Pセンブラ 出力コンバータ カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace アンジラ	H	 	 コード 動が取ります (第二) (11) (11)	局通化 高速化) −トを有効	出力 にする(E	<u>у</u> дь	7970セッサ	11分世所	MISF

3) コード

iテゴリ: 一般オブション 静的解析	■ 複数ファイルのコンパイル □ 未使用パブリックを破棄	[工場出	简時設定
シンタイム解析 C/C++コンバイラ アセンブラ 出力コンバータ カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace TI Stellaris	 言語 1 言語 2 □-ド 最適化 出力 リスト ブリブロセッ ブロセッサモード クロセッサモード Arm(A) Thumb(T) 位置独立コード/データ コードおよびリードオンリのデータ (ropi)(C) リード/ライトデータ (rwpi)(R) 動きカなリード/ライト狩刀明化なし(N) コードメモリ内のデータRead不生成(N) 	サロジャ	MISF

4) 最適化



5) 出力

ー 股オブション 156056246	-	■複数ファ □ ±4	าเเส	レバイル	t uffer				工場出	荷時設定
5ンタイム解析 C/C++ヨンパイラ アセンブラ 出力コンバータ カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace	10	言語 1 了/Sp Codet? text	 書語2 ヴ情報 ジョン名 	コード <u> ①</u> 生成(((C): .text		出力	UZF	ブリブロセッサ	8会世所	MISF * *

6) リスト



7) プリプロセッサ



<mark>注*1</mark>

「_USED_DEFnano_=0」と使用しない側に定義しても内蔵 RAM へのダウンロードとシリアルフラッシュ ROM への書き込み操作は可能です。ただし、再操作する場合はターゲット側のリセット操作が必要になります。

8)診断(設定なし)

カテゴリ: 一般オブション ###52715	-	複数ファイルのコン/	91JL				工場出	荷時設定
ゆい神和 ランタイム解析		ま語 1 言語 2 コ	クを破棄 ード 最適化	出力	リスト	ブリプロセッサ	珍野	MISF + +
アセンブラ 出力コンバータ カスタムビルド ビルドアかっつ	B)	リマークを有効化し 診断を無効化(S): リマーカトレスがまた(S):	(N)					
リンカ デバッガ シミュレータ		ワーニングとして処理	, (W):					
Angel CMSIS DAP GDBサーバ IAR ROMモニタ		エラーとして処理(E):	ジをエラーンし、てめ	138(11)				
I-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraioor				P1(7				
PE micro	-					OK		ter and the second

9) MISRA-C:2004 (設定なし)

カテゴリ				s.c.			工場出	山荷時設定
ー 股オブション 静的解析	Â	被数774.	ルのコンパイ. 目パブリックを	ル 破棄				
ランタイム解析		コード 最近	邹 化 出力	リスト	プリプロセッサ	診断	MISRA-C:2004	MISR +
カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraigor	E	1.1: 1.2: 1.2: 1.3: 1.3: 1.4: 1.5: 1.5: 2.1: 2.2: 2.2: 2.2: 2.2: 2.2: 2.2: 2.2: 2.2: 2.2: 2.3: 1.5:	(N) [quired] All quired] No quired] Mul quired] Mul quired] The quired] Sou quired] Sou quired] Sou	必須 code sha reliance s tiple com compiler ating-poi sembly la rice code characte	R) 37/ Il conform to IS shall be placed pliers and/or k /linker shall be nt implementat nguage shall be shall only use rr sequence /*	CC(A) O 9899 on unde anguage e checks ions sho e encaps ISO989 shall no	1登元(E) 1990 "Programmin timed or unspecif s shall only be us ed to ensure that wild comply with sulated and isolat 3:1990 °C' style co it he used within	ied beh ied beh ied if th 31 chai defined ed. omment a comn

10) MISRA-C:1998 (設定なし)



11) 追加オプション(設定なし)



2-5. アセンブラの確認

1) 言語

ノード"IAR_debug_rz"のオプショ	シージー 🔀
 ノード"IAR_debug_rz"のオプショ カテゴリ: 一般オプション 静的解析 ランタイム解析 C/C++コンパイラ アセンブラ 出力コンバータ カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace TI Stellaris 	下 まました。 下 まました。 下 まました。 下 まました。 「 まました。
PE micro	OK キャンセル

2) 出力

カテゴリ								工場出荷時設定
ー 股オブション 静的解析 5ンタイム解析 C/C++コンパイラ	ŕ	言語	出力	ሀスト	ブリプロセッサ	非全世斤	「追加オプション」	T-orthological
出力コンパータ 出力コンパータ カスタムビルド ビルドアクション リンカ デパッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ	E	7	バッグ情	報の生成	KOD			
1-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraigor PE micro							ΟΚ	

3) リスト



4) プリプロセッサ



<mark>注*1</mark>

「_USED_DEFnano_=0」と使用しない側に定義しても内蔵 RAM へのダウンロードとシリアルフラッシュ ROM への書き込み操作は可能です。ただし、再操作する場合はターゲット側のリセット操作が必要になります。

5)診断

カテゴリ:	1	丁爆火箭時設定
ー 股オブション 静的解析 ランタイム解析 C/C++コンパイラ アセンブラ	•	
出力コンバータ カスタムビルド ビルドアウション リンカ デバッガ	111	 ● 有効(E) ● すべてのワーニング(A) ● 無効(D) ● 特定ワーング(J): ● ワーニング Plone(F) To(T):
シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ		■最大Iラー数(M): 100 必要に応じて設定
I-jet/JTAGjet J-Link/J-Trace		
TI Stellaris	4	1
Macraigor	-	

6)追加オプション(設定なし)

カテゴリ:		6	工場出葡時設定
 ・ 股オブション ・ 腺はガジョン ・ 腺はの単体 ・ シンタイム解析 c/c++コンパイラ アセンブラ 出力コンバータ ・ コンパータ 			all multiple specie
カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモータ	н		*
I-jet/JTAGjet J-Link/J-Trace TI Stellaris			*
PE micro	-		

2-6. 出力コンバータの確認

1)出力

0710	工場出商時設定
 一般オブション 静的解析 ランタイム解析 C/C++コンパイラ アセンブラ 	出力
カスタムビルド ガスタムビルド ビルドアウション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet	出力フォーマット(F): Motorola 出力ファイル(O) ダブラカルトのオーバライド(V) IAR_debug_rz_mot
TI Stellaris Macraigor PE micro	OK ++>セル

2-7. カスタムビルドの確認(設定なし)

1) カスタムツール設定

カテゴリ			-
ー 股オブション 静的解析 ランタイム解析 C/C++コンパイラ アセンブラ 出力コンバータ		カスタムツール設定 ファイル名拡張子(F)	1
カスタムビルド		コマンドライン(0)	-
ビルドアクション リンカ デバッガ	н	出力ファイル(1行に1ファイル)(0):	 51
シミュレータ Angel			E.
CMSIS DAP		這加入力ファイル(1行に1ファイル)(A):	
GDBサーバ IAR ROMモニタ			
I-jet/JTAGjet J-Link/J-Trace		- 他のすべてのツールより先にこのツールを実行する(
TI Stellaris Macraioor			
PE micro	-	OK Start	

2-8. ビルドアクションの確認(設定なし)

カテゴリ:			
 一般オブション 静的解析 ランタイム解析 C/C++コンパイラ アセンブラ 出力コンバータ カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-Jet/JTAGjet J-Link/J-Trace 	* III	ビルドアクション設定 プリビルドコマンドライン(P): ポストビルドコマンドライン(0):	
Macraigor PE micro	-		OK \$**`#71.

2-9. リンカの確認

1) 設定



1-1) ベクタテーブル



1-2)メモリ領域

リンカ設定ファ	イルエディタ	> -= /-=]	
ROM <	開始: 0x20080200 0x20088000	終了: 0x20087FFF 0x209FFFFF	評価版のため ROM サイズ MAX(0x8000)ま での設定とす る。 残りエリア全て を RAM 側に割 り付ける。
1		保存 キャンセル	

1-3) スタック/ヒープサイズ

ベクタテーブル	メモリ領域 スタック	フ/ヒーブ サイズ	
CSTACK	0×8000		
SVC_STACK	0×1000		(アプリケーミ
IRQ_STACK	0×1000		ョンの規模に
FIQ_STACK	0×1000		応して設定 る。
UND_STACK	0×100		
ABT_STACK	0×1000		
HEAP	0×8000		

2) ライブラリ



3)入力(設定なし)

	1						6	工場出行	前時設定	
Â	-		1+		and a collection of the	Luna	lunt	1	Loopel	
	197.E	21729	~	载速比	PENDAE	出刀	971	#define	19380	100
	シンオ	「ルをキープ	(K)(1行	(21個)						
									^	
=										
									+	
	77	・バイナリイメ イル(F):	-ジ(R)		シンボル(S):	セクショ)(E):	アラインメン	2H(A):	
1				-						
										11
										- 1
		・ 設定 シンオ	設定 ライブラリ シンボルをキーブ ローバイナリイメ ファイル(F):	 設定 ライブラリ 入力 シンボルをキーブ (K)(1行 ローバイナリイメージ(R) ファイル(F): 	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	設定 ライブラリ 入力 最適化 アドバンスト シンボルをキーブ (K)(1行に1個) ローバイナリイメージ(R) ファイル(F): シンボル(S):	■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	設定 ライブラリ 入力 最適化 アドバンスト 出力 リスト シンボルをキーブ (K)(1行に1個) ローバイナリイメージ(R) ファイル(F): シンボル(S): セクション(E): シンボル(S): セクション(E):	工場出行 設定 ライブラリ 入力 最適化 アドバンスト 出力 リスト #define シンボルをキーブ (K)(1行に1個) □-バイナリイメージ(R) ファイル(F): シンボル(S): セクション(E): アラインズ)	工場出荷時設定 設定 ライブラリ 入力 最適化 アドバンスト 出力 リスト #define 診断・ シンボルをキーブ (K)(1行に1個)

4) 最適化(設定なし)

ー 股オブション 556558246	-	工場出荷時設定
ランタイム解析 C/C++コンパイラ アセンブラ 出力コンバータ カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace	## 	設定 ライブラリ 入力 最適化 アドバンスト 出力 リスト #define 診断・・ □ 小さいルーチンのインライン化 □ 重複セクションのマージ(M) □ C++仮想閲覧数除去を実行(P) □ VFE情報を持たないモジュールがある場合(V)

5) アドバンスト (設定なし)

 一般オブション 静的解析 ランタイム解析 C/C++コンパイラ アセンブラ 出カコンバータ カスタムビルド ビルドアウション デバッガ シミュレータ 	\$ \$ \$
ビルドアクション リンカ デバッガ シミュレータ	
Angel CMSIS DAP GDBサーバ IAR ROMモニタ	

6) 出力



7) リスト



8) #define (設定なし)

- 股オブション ^	1	工場出行	前時設定
伸り解析 5ンタイム解析 C/C++コンパイラ アセンブラ 出力コンパータ	設定 ライブラリ 入力 最適化 アドバンスト 出力 リスト シンボル定義(D)(1行に1個) 	#define	18W • • •
カスタムビルド ビルドアウション リンカ デバッガ シミュレータ Angel			
CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace			Ŧ

9)診断(設定なし)

カテゴリ:									工場出	荷時設定
一般オブション 静的解析 ランタイム解析	•	51750	ኢታ	最適化	アドバンスト	出力	U21	#define	ISUA]∓rs+[}
C/C++コンパイラ アセンブラ 出カコンバータ カスタムビルド ビルドアクション		□ リマー ■2世所を無	クを有文 展文が化([カíヒ(N))):			1.000			
リンカ		リマークと	して処理	聖(R):						
デバッガ シミュレータ Appel		ייב-בי	びとしてタ	£3里(₩):						_
CMSIS DAP GDBサーバ		エラーとし	て処理	(E):						
IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace		🗖 বন্দ	:07-:	こングをエラ	ーとして処理(T)				

10) チェックサム (設定なし)

カテゴリ					工場出荷8	糖定
ー般オブション 🔺						
ランタイム解析	アドバンスト出力リ	スト #define I診断	i F17	クサム 注	訪ロオブション	4. 4
アセンブラ	- 未使用コードメモリを	Eフィルする(F)				
出力コンバータ	フィルパターン	0xFF	1			
カスタムビルド	開始アドレス(T)	0×0	1 総了	75220); 0x0	î
ビルドアクション	「チェックサム生成	t(G)	10 A			
リンカ	サイズ(Z):	2/51 + +	75-	CXC-HU	0 1	
アパッカ	アルゴリズム	CRC16	-	0x110	21	-
Angel	□ フ ルサ	イズでの結果(U)		211度日(南)	m-	
CMSIS DAP	2前長女(○):	そのまま使用		0x0	4/1	
GDBサーバ	どット加(B):	MSBが先頭		177.27	1として使用(N)	-91
IAR ROMES	□語句内で	バイトオーダを逆順にす	3(R)			
1-jet/JTAGjet	チェックサム単	位サイズ(H) 8-bi	it	*		
TI Stellaris						
Macraioon						

11)追加オプション(設定なし)



2-10. デバッガの確認

1) 設定

		工場出荷時設定
 一般オブション 時的解析 ランタイム解析 C/C++コンパイラ アセンブラ 出力コンバータ カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ I-jet/JTAGjet J-Link/J-Trace TI Stellaris 	E	設定 ダウンロード イメージ 追加オウション マルチコア フラヴイン ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) ドライノベロ) (E) マクロファイルの使用(U) 使用するデノベッガを指定 する。 デブイン デフォルトのオーパライド(O) 第700LKIT_DIF\$#CONFIG#debugger#Renesas#R75721000.ddf

2) ダウンロード

カテゴリ:			工場出荷時設定
ー 股オブション 師的解析 ランタイム解析 C/C++コンパイラ	•	設定 ダウンロード イメージ 追加オブション マルチコア ブラヴイン	
アセンブラ 出力コンバータ カスタムビルド ビルドアウション リンカ デバッカ	E	□ フロクラムにアタッチする(A) マペリファイする(V) □ ダウンロードしな(V(S) □ フラッシュローダを使用する(U) □ デフォルトのboardファイルのオーバライド(O)	1
シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ		\$TOOLKIT_DIR\$#config#flashloader# 編集	
I-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraigor PE micro		ОК	年初ンセル

3) イメージ (設定なし)

1739:		工場出荷時設定
ー設オブション 神的解析 シンタイム解析	設定 ダウンロード イメージ 追加オブション マルチコア	P 75772
C/C++コンパイラ アセンブラ 出力コンパータ	 ご 追加イメージのダウンロード パス: 	
カスタムビルド ビルドアクション リンカ	オフセット 一 追加イメージのダウンロード	ヴィ香幸宿のみ
シミュレータ Angel CMSIS DAP	パス: オフセット ロデバッ 目道加イメージのダウンロード	ヴ情報記のみ
IAR ROME_S I-jet/JTAGjet J-Link/J-Trace	オンセット ロデバッ	ヴ情報のみ

4) 追加オプション (設定なし)

			工場出荷時設定
ニカスオノション 師的解剖析	Â.		
うンタイム解析		「時安」 おか、カード ノノーの 「良加オブション」 フルエフス グラガノン	
C/C++コンパイラ			
アセンブラ		コマンドラインオブションの使用(U)	
出力コンハーダ		コマンドラインオブション(C)(17行に1コマンド)	
ドルドアケション			~
リンカ			
デバッガ			
シミュレータ			
Angel			
CONSIS DAP			
IAR ROM王二久			
I-jet/JTAGjet			-
J-Link/J-Trace			
TI Stellaris			
Macraigor	2		
PE micro	120	OK	キャンセル

5) マルチコア (設定なし)

カテゴリ:					工場	出荷時設定
 ・ 股オブション ・ 股オブション ・ 時的解析 らンタイム解析 c/c++コンパイラ アセンブラ 出力コンパータ カスタムビルド ビルドアクション リンカ 	H	設定 ダウンロード イメージ 対称型マルチコア(Y) コアの数(N): 非対称型マルチコア(A)	追加オプション 1 有効(L(E)	₹IJ¥⊐₽ 25	772	
デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ		ポード(P) スレーブワークスペース(W) スレーブプロジェクト(S) スレーブ構成(C)	53461			
I-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraigor PE micro				ОК) (+v	201

6) プラグイン (設定なし)

3-19: 				工場出荷時設定
- 綾オブション 軸5解析 シタイム解析 C/C++コンパイラ アセンブラ 出力コンパータ カスタムビルド ビルドアクション リンカ デパッガ ジミュレータ	* III	設定 ダウンロー トドするブラヴ・ Percepio Tr CMX CMX TINY+ SEGGER er	-ド イメージ 減加オブション マルチコア ブラヴイン インの選択: race Exporter rbOS	
Angel CMSTS DAP		1.1月:	HCC Middleware plugin - has detailed view of HCC TCPIP stack and eTaskSync sched	/s on internals Juler.
GDBサーバ		場所:	C#Program Files#IAR Systems#Embedded	Workbench 7.3¥arm¥;
IAR ROMモニタ		作成者	HCC-Embedded	
I-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraigor		バージョン・	1.0.0.9	
J-Link/J-Trace TI Stellaris Macraigor PE micro	÷	1-732	OK	+ #



2-11. I-jet/JTAG jet の確認

1) 設定(デフォルト)

カテゴリ		工場出荷時設定
 一般オブション 静的解析 ランタイム解析 C/C++コンパイラ アセンブラ 出力コンバータ カスタムビルド ビルドアクション リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ 	設定 JTAG/SWD Trace ブレークポイント リセット(R) ハードウェア (デフォルト) 期間(D): 300 ms 遅延(fg)(E): ターゲット電源 ジブローブから供給(F) ● デバッグ後もオンにする(L) ● デバッグ後にオフにする(S)	200 ms
1-jet/JTAGjet	An ere of the state of the stat	
TI Stellaris Macraigor PE micro	(L	OK \$w'd7ll

2) JTAG/SWD (デフォルト)

一般オプション ヘ			上場出何時設定
静的海军析			
シンタイム解析	設定 JTAG/SWD	Trace ブレークポイント	
C/C++コンパイラ アセンブラ 出力コンパータ カスタムドルド	プローブ設定 ③ 自動 ③ ファイルから	ブローブ設定ファイル 「デフォルトのオーバライド(O)	
ビルドアクション	() 明示的	CPU 選択	1
リンカ デバッガ シミュレータ Angel CMSIS DAP	インタフェース ③ JTAG(J) ⑦ SWD(S)	明示的なプローブ設定	
IAR ROMモニタ	JTAG/SWD速度(W) 自動検出 👻	□ JTAGスキャンチェーンに非ARM デバイスを1 先行ビット(P): 0	30



3) Trace (デフォルト)

יעבידו				工場出荷時調	命定
- 股オブション 伸的解析 シンタイム解析 C/C++コンパイラ	•	設定 JTAG/SWD Tr	ace ブレークボイント		
アセンブラ 出力コンバータ カスタムビルド ビルドアクション		トレーステータの40集 モード 自 バッファ制限: 81	けん ・] ナンブル ・]	III >ETBの単キ可	
リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ		プロトコル ● 自動(A) つ マンチェスタ(M) 〇 UART(U)	クロック設定 CPUクロック(C) SWOプリスケーラ	MHz 5(S) 自動 +	
IAR ROME_9 I-jet/JTAGjet J-Link/J-Trace TI Stellaris		TraceD082xØSV	/O(W)		_

4) ブレークポイント(デフォルト)

カテゴリ: 一般オブション ・ #865874C			工場出荷時設定	
ayorymywy ランタイム解析 パイラ		設定 JTAG/SWD Trace ブレー	クポイント	
アセンブラ 出力コンバータ カスタムビルド ビルドアクション	ш	デフォルトのブレークポイントタイプ ③ 自動(A) ③ ハードウェア(H) ③ ソフトウェア(W)	עדלילע 🖾 🛄 🛄	レークボイント復元位置(R)
リンカ デバッガ シミュレータ Angel CMSIS DAP GDBサーバ IAR ROMモニタ		例外をキャッチ □リセット(T) ▽ ブリフェッチ(P) ▽ 未定義(U) □ IRO(D □ SWI(S) □ FIO(F) ⊽ データ(D)	CORERESET MMERR NOCPERR CHKERR	STATERR BUSERR INTERR HARDERR
1-jet/JTAGjet				
J-Link/J-Trace TI Stellaris Macraigor PE micro				ОК 0

- 3. サンプルプロジェクト「IAR_debug_rz」をビルドする。
 - 1) すべてをビルド



2) ビルド結果



- 4. 「I-jet/JTAG」でサンプルプロジェクト「IAR_debug_rz」をダウンロードする。
 - DC5V入力 JTAG(CN7) 20pin 2.54 ピッチ 1.+5V 2. GND LED1 LED2 LED3 LED4 TPWR SW1(DEBUG) USB 💿 O DBG 1番 OFF un -2番 OFF SW2(MODE) **MCU**基板 3番 OFF 1番 **ON** MP-RZA1H-xx
 - 1)動作構成例(MP-RZA1H-01)

2) ダウンロード

7ァイル(F) 編集(E) 表示(V)	ロジェクト(P) I-jet/JTAGjet(I) ツール(T)	ウィンドウ(W)	ヘルプ(H)
D 📽 🖬 🕼 🕼 🕷 🖷	ファイルの追加(F)		1 # # @ @ @ 12 M
-52-2	グループの追加(G)		f0 -
Debug	ファイルリストのインボート(I)		
77-116 2:	プロジェクトコネクションの追加(P)		
🗆 🚮 IAR debug rz 🗸	ビルド構成の編集(T)		c#28 Bc#10 P1c#5 P0c#5/2
-G _ src_app	肥胨(V)		77.
-B board_d.c	新規プロジェクトの作成(N)		
	既存プロジェクトの追加(E)		
-B Stram.c	オプション(0)	Alt+F7	*****
-Q_src_sys	バージョン管理システム(Y)		***********************
-B inc	X ((M)	57	
-E sthandler ss	77.07.07	Ch1+57	r\$main] GNU[_rst_handler.s]
- e .vector_table		Cultry	
- Into_table.c	9/(C&HEDDP(D)		// System#700Ht (resets
E interrupt c	00-0(L)		// 11- KP - 77037880
- B Umonitor c	バッチヒルト(A)	F8	11 11-1-5 1 1004384
Le Output	C-STAT静的触机(C)		// FRANT-3 20100548
IAR debug rz	ビルドを停止(S)	Ctrl+Break	// 28ms
	ダウンロードしてデバッグ(D)	Ctrl+D	
メリセーソ 課題 IAD data are Data	ダウンロードせずにデバッグ(H)		
ビルドシリーを更新中	メイク後デバッガを再起動(K)	Ctrl+R	
41 (別のファイルを削除す) ました	デバッガを再起動(R)	Ctrl+Shift+R	
ビルドンリーを更新中 jnit handler.s.s	ダウンロード(W)	+	
*	SFRの設定(U)		,
FRODO PAF	デバイス記述ファイルを聞く		
プリケーションをダウンロード	the second se		-0. ワーニング0

3) デバッガ起動画面

ファイル(F) 編集(E) ヘルプ(H) D 😂 🔛 🗿 🚳	長示(V) プロジェクト(P) デバッグ(D) 逆アセンブバ	L(A) I-jet/JTAGjet(I) ツー イトトン 注意 P P チ	ul(T) ウィンドウ(W)
5 • • 8 2 3 m m	888 X			
3-52X-2 ×	main.crst_hwider_s.s	f0 • x	建アセンプリ	×
Debug 👻	34 35 :***********************************	· *	1940	12
	97 1012 - 9 + 102 - 8 33 PUBLIC _ isr program_start 41 Reschandler: :::::PUBCTION () 42 Isr program_start: 43 : Disable cache and MMU in case it () 44 : This does not need to be done from 45 : PIC ris, ro, stort () 46 IMC piS, 0, r0, cl, c0, 0 47 BIC ris, ro, stort () 48 BIC ris, ro, stort () 49 BIC ris, ro, stort () 40 BIC ris, ro, stort () 41 BIC ris, ro, stort () 42 BIC ris, ro, stort () 43 BIC ris, ro, stort () 44 BIC ris, ro, stort () 45 BIC ris, ro, stort () 46 BIC ris, ro, stort () 47 BIC ris, ro, stort () 48 BIC ris, ro, stort () 49 BIC ris, ro, stort () 40 BIC ris, ro, stort () 41 BIC ris, ro, stort () 42 BIC ris, ro, stort () 43 BIC ris, ro, stort () 44 BIC r	mas left enabled from an a a cold reset III (SCILENDEwsd システム III disable[]] 金布キッ disable[]] マータキャ disable[]] アータキャ III (SCILENTrite システ) III (SCILENTrite システ)	WRC p15, Reset_handler iar_program 0x200802er 0x200802er 0x200802f4 B1C r0 0x200802f5 0x200802f6 0x200802f6 0x200802f6 0x200802f6 0x200802f6 0x200802f6 0x200802f6 0x200802f6 0x200802f6 0x20080304 CPS BSV0 0x20080304 CPS BSV0 0x20080308	
IAR_debug_rz	·	÷ =	* III. *	
D7 Wed Feb 17, 2016 Wed Feb 17, 2016 Wed Feb 17, 2016 Wed Feb 17, 2016 Wed Feb 17, 2016	7 09 09 37: 11048 パイトがダウンロードあよび研修定済み(171 09 09 37: ダウンロードが完了し、林田に成功しました。 09 09 37: しのM.eve ResetSoftware, deby 200) 09 09 37: サーゲットリセット 09 09 37: サーゲットリセット 09 09 37: サービン Control trace using "Auto.size_Jim 19 09 37: サービン Control tracesure of using "Auto.size_Jim	25 KB/P3) R=6M' setting		-
<	HI H	ner poner na une tergot.		6
デパックロク ビルド				×

5. 「DEFnano」でサンプルプロジェクト「IAR_debug_rz」をダウンロードする。

1)動作構成例(MP-RZA1H-01)



2) 起動とダウンロード

Apr-8/0)	CDUD	/A14/D1ST01000 Adum		un eri
7970-F(U)	Ele Str. CI	G Aca	- 2:0 -	160 314
SU77500		and and an and an		_
()))))(V)				
シンパンレビス249(1)		ファイルー	ダウンロード	
アブソリュートファイル設定(A)	を選	沢する。		
CPU設定読込み(S)				
CPU設定管理(R)	PL8 PC	tra, ere, must tra		
1 Harrison Contraction	D 014	16208420		
スクリプト実行(L)	HI 18.	r0.10x20000000		
オフライン作業(の)	LS rll	, r0, r0		
	CSDA r8	.[r].r6.r8.ir]		
2 2 2 4 2 36 8 US 2 (M)	W Ph	,r12,10x274		
終了(E)	IS els	+12 10-FC00000		
2008103C: D7FDC76F	LORLEB r17	.[sp, pc, ROR 114]		
20080040: 4000C401	ANDME r13	.r0.r1. LSL 18		
20080044: 85570608	LDRBH1 r0.	[r7, 1-8x608]		
20060049: 8200A008	NUVINI PIL	1, BOXE		
1) Start ク	レック	[pc, r7, LSR \$21]		
00800	r9,	[r10, -r9, ROR \$19]!		
20083058: 00	LULLE DO.	.c10,[1/], 10x1E8		
2006805C: D	LORLEB P12	[[Ir, #8, LSR #27]]		
SB1 (600307	CLR 🛨 🦳 SB2 100	100000	■ CLR ● 実行回数 1	
		* char	- 16進 - 二川明	
	5506			- 3
	nu l nu la	al e le	0.000	1
uo Break Kstillon	Tin Reg Ta	tch Sym Frace	Step Cirac CStep	·

3) ダウンロードファイルを選択する。

	ダウンロード
OO 👍 « IAR_debug_rz + Debug + Exe	- 4 Exeの検索 の
整理・ 新しいフォルダー	H • 🖬 🛛
DS-5 Workspace	名前 注 IAR_debug_rz.out
LAK_debug_rz	「IAR_debug_rz.out」を 選択する。
🕌 List 🚹 Obj	- x [
IAR_debug_rz.out 状況: 3 PSpice Simulation Output 更新日時: 20	,共有 116/02/16 18:19
ファイル名(N): IAR_debug_ra	2.out ・ ABSFile(*.x,*.axf,*.out) ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・

4) デバッガ起動画面

080000	\$33:	• Grc C Mix	C Asm _vector_table_s. ▼ 高速 ▼
200800000:	31	LDR pc, =Reset_handler	: Start+0x0000 : Ut2 7 F
20080004:	34	LDR pc. =Undefined_handler	; Start+0x0004 : 未定路命令
20080008:	35	LDR pc, #Svc_hendler	: Start+0x0000 : ソフトウェア割り込み
2008000C:	36	LDR pc, =Prefetch_handler	: Start+0x000C : プリフェッチアポート
20080010:	37	LDR pc. #Abort_handler	: Start+0x0010 : データアポート
20080014:	38	LDR pc, =Reserved_handler	; Start+0x0014 : Reserved
20080018:	39	LDR pc. =lrg_handler	: Start+0x0018 : IR0
2008001C:	40	LDR pc, #Fig_hendler	: Start+0x001C : FIO(NMI)
	41	;======================================	
	42	; SFROMに登録したローダーに渡す情報	
	43	;	
	44	Info_table:	
	45	DCD vector_table	: Start+0x0020: 内蔵HAM転送先の開始アドレス
-	46	DCD Region\$\$Table\$\$Limit	: Start+0x0024: 内蔵RAWE12先の終了アドレス(+ -
	47	DCD vector_table	: Start+0x0028 : 75MMPCH
-	48	bab u	: Start+UxUUZU: T/1y/Jt=FJJ/
-	48		: DEFnanoを未使用こして、UGBDを
-	58	1.1	; UXDEFUDEFU空定義す _{争。}
-	01	Info_end:	
	20	1 How lat	
	0.0	Literals.	
20010050-	55	E I GENER	
F00000000			1
\$81 0000000	00	■ CLR C SB2 000000	2010 CLR • 其行回数 1
-			▼ cher × 16道 ▼ ■ 1000
-	-		

<mark>注*1</mark>

「__USED_DEFnano__=0」と使用しない側に定義してあっても内蔵 RAM へのダウンロード 操作は可能です。ただし、再操作する場合はターゲット側のリセット操作が必要になります。 6. 「DEFnano」でサンプルプロジェクト「IAR_debug_rz.mot」をシリアルフラッシュ ROM に書き込む。

<mark>注*1</mark>

「_USED_DEFnano_=0」と使用しない側に定義してあってもシリアルフラッシュ ROM への書き込み操作は可能です。ただし、再操作する場合はターゲット側のリセット操作が必要になります。

1) シリアルフラッシュ ROM ライタの起動

80000	CPORZA	CPUIRE(C)	protoursk .
31 32	vector_table:	フラッシュROMライタ(F)	-
20080600: 33	LDR pc. =Reset_handler	ファームバージョンアップ(V)	
20080004: 34	LDR pc. =Undef ined_handler	1 Start all 0000 1 1 1 2 1 4 2 7 10 10	
200800000: 35	LIR oc. Prefetch	: Start+0x0000 : フリフィッチアボ	-1-
20080810: 37	LDB pc, "Abort	: Start+0x0010 : 7-97#-1	
20080014: 38	LIR pr. IP	StarteBu0014 ! Reserved	
20030018: 38		JE NY DOMEZA	+
20080810: 40	【オノンヨン】-【	ノフツンユ KOM フイダ】	X
41	選択する		
42	「歴れりる。		
43	1		
44	Info_table:		
45	DCD vector_table	: Start+0x0020 : 内蔵R4M転送先の構	陥アドレス
46	DCD Region\$\$Table\$\$Limit	: Start+0x0024 : 内蔵RAM転送先の新	77FL2(+
47	DCD vector_table	; Start+0x0028 : 初期9C值	
48	DCD 0	: Start+0x002C : デバッグモードフ	50
48		: DEFnanoを未使用に	して、 USBO き
50	12 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C - 2 C	: 0xDEF0DEF0を定義:	70.
51	Info_end:		
52			
2.0	Litereis:		
58	- CLR - C 582 00000	1000 · CLR · 実行回	数「
58 SB1 003000000			THE .
53 581 003000000	w (121)	- LAND	
53 SB1 (0030000000) [★ (なし)	Tana Interest Plane Plane	
58 \$81 (003000000)	• (tau)	The first state	
58 581 (003000000)			

2) Hexファイルの選択と書き込み開始

¥UsrAp¥C_H	125_IAR¥IAR Work	space¥IAR_debug_rz¥Debug¥Exe¥IAR_debug_rz.mot [®] ノファイル違抗
書込み個数	0	書込み開始
全消去個數	0	全演进開始
		①「ファイル選択」 をクリックして Hex フ ァイルを選択する。
		②「書込み開始」 をクリックする。

3) 書き込み中



4)書き込み終了

N¥UsrAp¥C_H25¥DS-5 Workspace¥CD_Sample_ARMC¥USER	ファイル選択
書込み個数 1 書込みは正常に終了しました。	書込み開始
全消去個数 0	全消去開始
正常終了しますと、	
「青込みは止吊に於」しました。」	します。
	/ 0

5) シリアルフラッシュ ROM の消去



7. リンカ設定ファイル (*.icf) について

リンカ設定ファイルは、セグメントごとにアブソリュートアドレスを定義するためのファ イルです。リンカファイル独自の予約語がありますので IAR 社が提供しているドキュメント 「C/C++開発ガイド」の「バート2:リンカ設定ファイル」を参照して下さい。

/*###ICF### Section handled by ICF editor; don't touch! ****/ /*-Editor annotation file-*/ /*IcfEditorFile='\$TOOLKIT_DIR\$EconfigEditorEditora_v1_0.xml''*/ /*-Specials-*/ define symbol __ICFEDIT_intvec_start __ = 0x20080000; /*-MemoryRegions-*/ define symbol __ICFEDIT_region_ROM_start __= 0x20080200; define symbol __ICFEDIT_region_ROM_end __ =0x20087FFF; define symbol __ICFEDIT_region_RAM_start __ = 0x20088000; define symbol __ICFEDIT_region_RAM_end __ = 0x209FFFFF; /*-Sizes-*/ define symbol __ICFEDIT_size_cstack___ =0x8000; define symbol __ICFEDIT_size_svcstack __ = 0x1000; define symbol __ICFEDIT_size_irqstack __= 0x1000; define symbol __ICFEDIT_size_fiqstack __ = 0x1000; define symbol __ICFEDIT_size_undstack __ = 0x100; define symbol __ICFEDIT_size_abtstack __= 0x1000; define symbol __ICFEDIT_size_heap___ =0x8000; /***** End of ICF editor section. ####ICF###*/ define symbol __ICFEDIT_region_RetRAM_start __ =0x2000000; define symbol __ICFEDIT_region_RetRAM_end ____ =0x2001FFFF; define symbol __ICFEDIT_region_MinorRetRAM_start __= 0x6000000; define symbol ICFEDIT region MinorRetRAM end =0x6001FFFF; define symbol __ICFEDIT_region_MirrorRAM_start__ =0x60090000; define symbol __ICFEDIT_region_MinrorRAM_end __ =0x604FFFFF; define symbol __ICFEDIT_region_UncachedRAM_start __ = 0x60500000; define symbol __ICFEDIT_region_UncachedRAM_end __ = 0x609FFFFF; define memory mem with size=4G; define region RetRAM_region = mem: [from _ICFEDIT_region_RetRAM_start_ to _ICFEDIT_region_RetRAM_end_]; define region ROM_region = mem:[from __ICFEDIT_region_ROM_start__ to _ICFEDIT_region_ROM_end__]; define region RAM_region = mem: [from __ICFEDIT_region_RAM_start_ to __ICFEDIT_region_RAM_end_]; define region MinrorRAM_region=mem:[from _ICFEDIT_region_MinrorRAM_start_to_ICFEDIT_region_MinrorRAM_end_]; define region MinorRetRAM_region=mem:[from_ICFEDIT_region_MinorRetRAM_start_to_ICFEDIT_region_MinorRetRAM_end_];

define region UncachedRAM_region = mem:[from __ICFEDIT_region_UncachedRAM_start_to __ICFEDIT_region_UncachedRAM_end__;

/* stacks */

define symbol__ICFEDIT_size_TTB__ =0x4000; define block TTB with alignment=0x4000,size=__ICFEDIT_size_TTB__ { }; define block CSTACK with alignment=16, size=__ICFEDIT_size_cstack__ { }; define block SVC_STACK with alignment=16, size=__ICFEDIT_size_svestack__ { };

```
define block IRQ_STACK with alignment = 16, size = _ICFEDIT_size_irqstack__{};
define block FIQ_STACK with alignment=16, size=_ICFEDIT_size_fiqstack__{};
define block ABT_STACK with a lignment = 16, size = __ICFEDIT_size_abtstack__ { };
define block UND_STACK with alignment = 16, size = __ICFEDIT_size_undstack__ { };
define block HEAP with alignment = 16, size = __ICFEDIT_size_heap__ { };
initialize by copy with packing=none {
 readwrite,
};
donot initialize {
 section .noinit,
};
place at address mem:__ICFEDIT_intvec_start__ {
 section VECTOR_TABLE
                             /*asm*/
};
place at start of ROM_region {
 section RESET_INIT_HANDLER, /*asm*/
 ro code,
 rodata,
};
place in RAM_region {
 block TTB,
 block IRQ_STACK,
 block FIQ_STACK,
 block SVC_STACK,
 blockABT_STACK,
 block UND_STACK,
 block CSTACK,
 block HEAP,
 readwrite.
 zi,
};
place at start of UncachedRAM_region {
};
```

リンカ設定ファイルを編集する場合の注意点

1) 1行 (/*###CF###) から19行(###CF###*)は、エディタ等で直接変更しないで下さい。変更が 必要な場合は、EWARM の【オプション】- 【リンカ】- 「設定」- 「編集」で変更す る。 8. ベクタテーブルとローダーの関係について

MP-RZA1H 基板は、シリアルフラッシュ ROM にローダーとアプリケーションプログラム を記憶させ、電源 ON 時にローダーが RZA1H の内蔵 RAM にアプリケーションプログラムを ロードして実行させる仕組みになっています。ローダーは内蔵 RAM にロードする時にロード 先の先頭アドレスと最終アドレスと実行開始アドレスを知る必要があります。この情報を得 るため独自の定義が必要なため下記に説明します。

1) ローダーが必要な情報をベクタテーブルに登録する。「_vector_table_ss」

Start	
; Entry point for the Reset handler	
<pre>;====================================</pre>	; Start+0x0000 : リセット ; Start+0x0004 : 未定義命令 ; Start+0x0008 : ソフトウェア割り込み ; Start+0x000C : プリフェッチアボート ; Start+0x0010 : データアボート ; Start+0x0014 : Reserved ; Start+0x0018 : IRQ
LDR pc, =Fiq_handler	; Start+0x001C : FIQ(NMI)
; SFROMに登録してあるローダーに渡す情報 ;====================================	
DCD vector_table	; Start+0x0020:①内蔵RAM転送先の開始アドレス
DCD Region\$\$Table\$\$Limit	; Start+0x0024:②内蔵RAM転送先の終了アドレス(+1)
DCD vector_table	; Start+0x0028 :③初期PC值
DCD 0	; Start+0x002C :④デバッグモードフラグ
	; DEFnanoを未使用にして、 USBOを開放する場合は、
Info_end ;====================================	; 0xDEF0DEF0を定義する。
	【注意事項】
①②③④の情報は、ROM 化するためには必要な情報 テーブルです。必ず、定義 して下さい。	 ④で「DEFnano 未使用」 コード「0xDEF0DEF0」をセットし、シリアル フラッシュ ROM に登録した場合、二度と DEFnano を使用することが出来なくなります。 復帰したい場合は、JTAG デバッガ等でシリアル フラッシュ ROM アドレス「0x2_002C」を未使 用コード「0xDEF0DEF0」以外の数値を直接書 き込んでください。

以上です。

- 9. 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。

10. 商標

- ・EWARMは、IAR社の登録商標、または商品名称です。
- ・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。
- 11. 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
 - ・ルネサス エレクトロニクス株式会社提供のサンプル集
 - ・「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-9-j」 IAR社
 - ・「IAR デバッグプローブガイド IAR probes-2-j」 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