

RZ/T2M, RZ/T2ME, RZ/N2L, RZ/T2L グループ

クイックスタートガイド: lwIP プロトコルスタックサンプルプログラム

要旨

本書は、ルネサス RZ/T2M, RZ/T2ME, RZ/N2L, RZ/T2L プラットフォームに lwIP(lightweight IP)プロトコルスタックを実装したサンプルプログラムの設定手順について説明します。

動作確認デバイス

RZ/T2M, RZ/T2ME, RZ/N2L, RZ/T2L

目次

1. 概説.....	3
1.1. 略語／定義.....	3
1.2. 参照.....	3
1.3. 制限事項／既知の問題.....	3
2. 特徴.....	4
2.1. プロジェクトのフォルダー.....	4
3. 要件(Software&Hardware).....	5
3.1. RZ/T2M.....	5
3.2. RZ/N2L.....	6
3.3. RZ/T2L.....	7
4. ハードウェア設定.....	8
4.1. RZ/T2M RSK Board.....	8
4.2. RZ/N2L RSK Board.....	9
4.3. RZ/T2L RSK Board.....	10
4.4. デバッガの接続.....	11
5. ホスト PC の設定.....	12
5.1. インターフェースのプロパティ設定.....	12
5.2. Ethernet 通信のモード.....	14
6. サンプルアプリケーションの実行.....	16
6.1. e ² studio のサンプルプロジェクトの設定.....	16
6.1.1. e ² studio の起動.....	16
6.1.2. コードの生成とビルド方法.....	18
6.1.3. アプリケーションのダウンロードとデバッガの実行.....	21
6.2. EWARM のサンプルプロジェクトの設定.....	23
6.2.1. EWARM の起動.....	23
6.2.2. コードの生成とビルド方法.....	24
6.2.3. アプリケーションのダウンロードとデバッガの実行（シングルコア版）.....	27

6.2.4. アプリケーションのダウンロードとデバッグの実行（デュアルコア版）	29
7. サンプルアプリケーションのデモンストレーション	31
7.1. PC と接続	31
7.2. SNTP クライアント	33
7.3. iperf クライアント／サーバー	34
7.3.1. 通信速度の測定手順	34
7.4. エコーサーバー	36
8. ソフトウェア仕様	37
8.1. ソフトウェア構造	37
8.1.1. コンポーネントの概要	38
8.1.2. 使用ドライバ	38
8.2. タスク	39
8.2.1. リソース	39
8.2.2. タスク操作	40
8.3. 受信パケットのミラーリングと送信機能	41
8.4. マルチインターフェース	43
8.4.1. インターフェースのプロパティテーブル	43
8.4.2. ルーティング	45
8.4.3. Management port specific tag + 非ルーティングモード	46
8.5. メモリ配置	47
8.5.1. e2 studio のプロジェクト	47
8.5.2. EWARM のプロジェクト	54
8.6. CPU 使用率	62
Revision History	63

1. 概説

本書は、ルネサス RZ/T2M, RZ/T2ME, RZ/N2L, RZ/T2L プラットフォームに lwIP プロトコルスタックを実装したサンプルプログラムの設定手順について説明します。

なお、RZ/T2ME の場合、RZ/T2M のプロジェクトファイルで動作可能な為、RZ/T2ME 個別の記載は省略いたします。

1.1. 略語／定義

Table 1.1. 略語／定義

Index	Abbreviations /Definitions	Description
1	IP	Internet Protocol
2	TCP	Transmission Control Protocol
3	USB	Universal Serial Bus
4	PC	Personal Computer
5	SW	Switch
6	EWARM	Embedded Workbench® for ARM
7	lwIP	lightweight IP

1.2. 参照

RZ/T2M, RZ/N2L, RZ/T2L に関する技術情報は、ルネサスエレクトロニクスから入手することができます。

Table 1.2. Technical Inputs for RZ/T2M

種類	目的	ドキュメント名	ドキュメント No.
User's Manual	RSK+RZ/T2M,RZ/T2ME の仕様説明	Renesas Starter Kit+ for RZ/T2M, RZ/T2ME ユーザーズマニュアル	r20ut4939jj****
User's Manual	RZ/T2M の仕様説明	RZ/T2M グループユーザーマニュアル ハードウェア編	r01uh0916jj****

Table 1.3. Technical Inputs for RZ/N2L

種類	目的	ドキュメント名	ドキュメント No.
User's Manual	RSK+RZ/N2L の仕様説明	Renesas Starter Kit+ for RZ/N2L ユーザーズマニュアル	r20ut4984jj****
User's Manual	RZ/N2L の仕様説明	RZ/N2L グループユーザーマニュアル ハードウェア編	r01uh0955jj****

Table 1.4. Technical Inputs for RZ/T2L

種類	目的	ドキュメント名	ドキュメント No.
User's Manual	RSK+RZ/T2L の仕様説明	Renesas Starter Kit+ for RZ/T2L ユーザーズマニュアル	r20ut5164jj****
User's Manual	RZ/T2L の仕様説明	RZ/T2L グループユーザーマニュアル ハードウェア編	r01uh0985jj****

1.3. 制限事項／既知の問題

特にありません。

2. 特徴

lwIP(lightweight IP)はオープンソースの組み込み向け TCP/IP プロトコルスタックです。lwIP は少ないリソースでフルスケールの TCP 機能を実装でき、組み込みシステムでの使用に適しています。元々スウェーデンコンピュータサイエンス研究所(SICS)の Adam Dunkels 氏によって開発され、現在は世界中の開発者のネットワークによって開発および維持されています。lwIP のライセンスについてはサンプルプログラムの \common\oss\lwip\COPYING を参照ください。

2.1. プロジェクトのフォルダー

主要ファイルのフォルダー概要を示します。

なお、フォルダーツリーは、リリースノートで確認してください。

フォルダー名	概要
common/oss	lwip+freeRTOS のパッケージファイル
common/renesas/application	アプリケーション
common/renesas/module/ether_netif	Ethernet の API
common/renesas/module/lwip_port	lwip の API
common/renesas/module/serial_io	LOG 出力用のシリアル API
SRC	エントリー関数など

3. 要件(Software&Hardware)

このプロジェクトは、以下のボードとツールを使用してこれらの環境で開発およびテストされました。

3.1. RZ/T2M

Table 3.1. RZ/T2M environment

Category	Name	Version	Description
Board	Renesas Starter Kit+ for RZ/T2M, RZ/T2ME	ユーザーズマニュアル 2.00	Renesas RZ/T2M-RSK - Renesas Starter Kit Plus for RZ/T2M Renesas
IDE	EWARM	9.60.3	IAR Systems IAR Embedded Workbench for Arm IAR
	e ² studio	2025-12	RZ FSP
Configurator	FSP Smart Configurator	2025-12	https://github.com/renesas/rz-fsp/releases/tag/v4.0.0
Flexible Software Package	FSP	4.0.0	
GCC Compiler	GNU ARM Embedded Toolchain	13.3.Rel1	
Emulator	J-Link TM	8.60	SEGGER SEGGER - The Embedded Experts - Downloads - J-Link / J-Trace
	I-jet		IAR Systems IAR debug probes IAR
	Tera Term		Tera Term Tera Term Open Source Project
Software	Wireshark		WIRESHARK Wireshark • Go Deep Download

3.2. RZ/N2L

Table 3.2. RZ/N2L environment

Category	Name	Version	Description
Board	Renesas Starter Kit+ for RZ/N2L	ユーザーズマニュアル 1.04	Renesas RZ/N2L-RSK - Renesas Starter Kit+ for RZ/N2L Renesas
IDE	EWARM	9.60.3	IAR Systems IAR Embedded Workbench for Arm IAR
	e ² studio	2025-12	RZ FSP https://github.com/renesas/rz-fsp/releases/tag/v4.0.0
Configurator	FSP Smart Configurator	2025-12	
Flexible Software Package	FSP	4.0.0	
GCC Compiler	GNU ARM Embedded Toolchain	13.3.Rel1	
Emulator	J-Link TM	8.60	SEGGER SEGGER - The Embedded Experts - Downloads - J-Link / J-Trace
	I-jet		IAR Systems IAR debug probes IAR
	Tera Term		Tera Term Tera Term Open Source Project
Software	Wireshark		WIRESHARK Wireshark • Go Deep Download

3.3. RZ/T2L

Table 3.3. RZ/T2L environment

Category	Name	Version	Description
Board	Renesas Starter Kit+ for RZ/T2L	ユーザーズマニュアル 1.21	Renesas RZ/T2L-RSK - Renesas Starter Kit+ for RZ/T2L Renesas
IDE	EWARM	9.60.3	IAR Systems IAR Embedded Workbench for Arm IAR
	e ² studio	2025-12	RZ FSP
Configurator	FSP Smart Configurator	2025-12	https://github.com/renesas/rz-fsp/releases/tag/v4.0.0
Flexible Software Package	FSP	4.0.0	
GCC Compiler	GNU ARM Embedded Toolchain	13.3.Rel1	
Emulator	J-Link TM	8.60	SEGGER SEGGER - The Embedded Experts - Downloads - J-Link / J-Trace
	I-jet		IAR Systems IAR debug probes IAR
	Tera Term		Tera Term Tera Term Open Source Project
Software	Wireshark		WIRESHARK Wireshark • Go Deep Download

4. ハードウェア設定

主要なハードウェアについて説明します。使用するボードの詳細については、[ユーザーズマニュアル](#)を参照してください。

4.1. RZ/T2M RSK Board

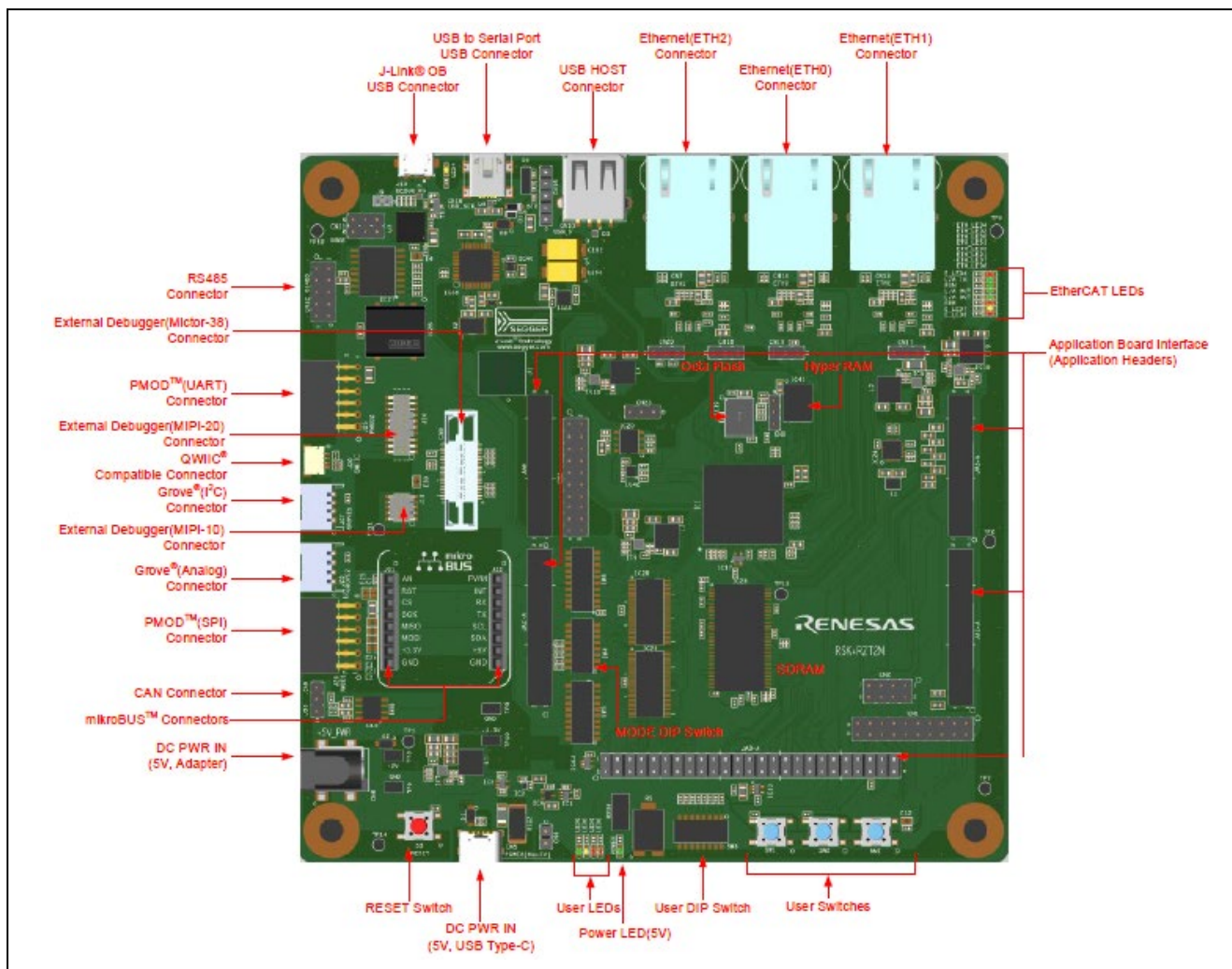


Figure 4.1 RZ/T2M RSK board layout

1. DIP SW と Jumper SW の設定

Renesas Starter Kit+ for RZ/T2M ユーザーズマニュアル「[6. コンフィグレーション](#)」の設定です。

2. J-Link は micro USB へ、I-jet は J20 に接続し、J9 で切り替えます。
3. 電源の供給には、USB ケーブル (Type-C) または、AC/DC アダプターを使用してください。
 - ・ USB ケーブルを使用する際は CN5 に接続をしてください。
 - ・ AC/DC アダプターを使用する際は CN6 に接続してください。
4. イーサネットケーブルは ETH0、ETH1、ETH2 の、同時または、いずれか、自由に接続してください。

4.2. RZ/N2L RSK Board

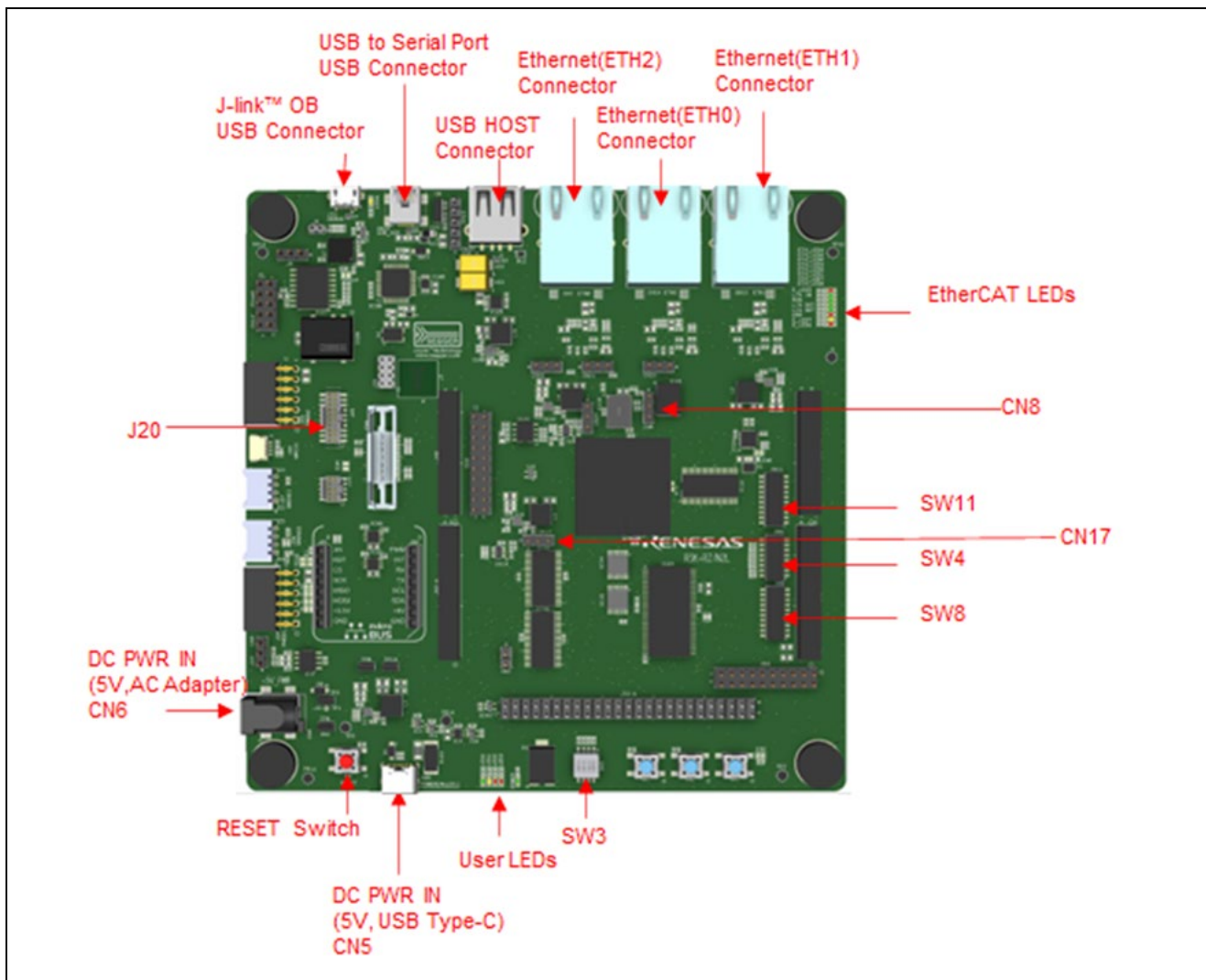


Figure 4.2 RZ/N2L RSK board layout

1. DIP SW と Jumper SW の設定

Renesas Starter Kit+ for RZ/N2L ユーザーズマニュアル「[6. コンフィグレーション](#)」の設定です。

2. J-Link は micro USB へ、I-jet は J20 に接続し、J9 で切り替えます。
3. 電源の供給には、USB ケーブル (Type-C) または、AC/DC アダプターを使用してください。
 - ・ USB ケーブルを使用する際は CN5 に接続をしてください。
 - ・ AC/DC アダプターを使用する際は CN6 に接続してください。
4. イーサネットケーブルは ETH0、ETH1、ETH2 の、同時または、いずれか、自由に接続してください。

4.3. RZ/T2L RSK Board

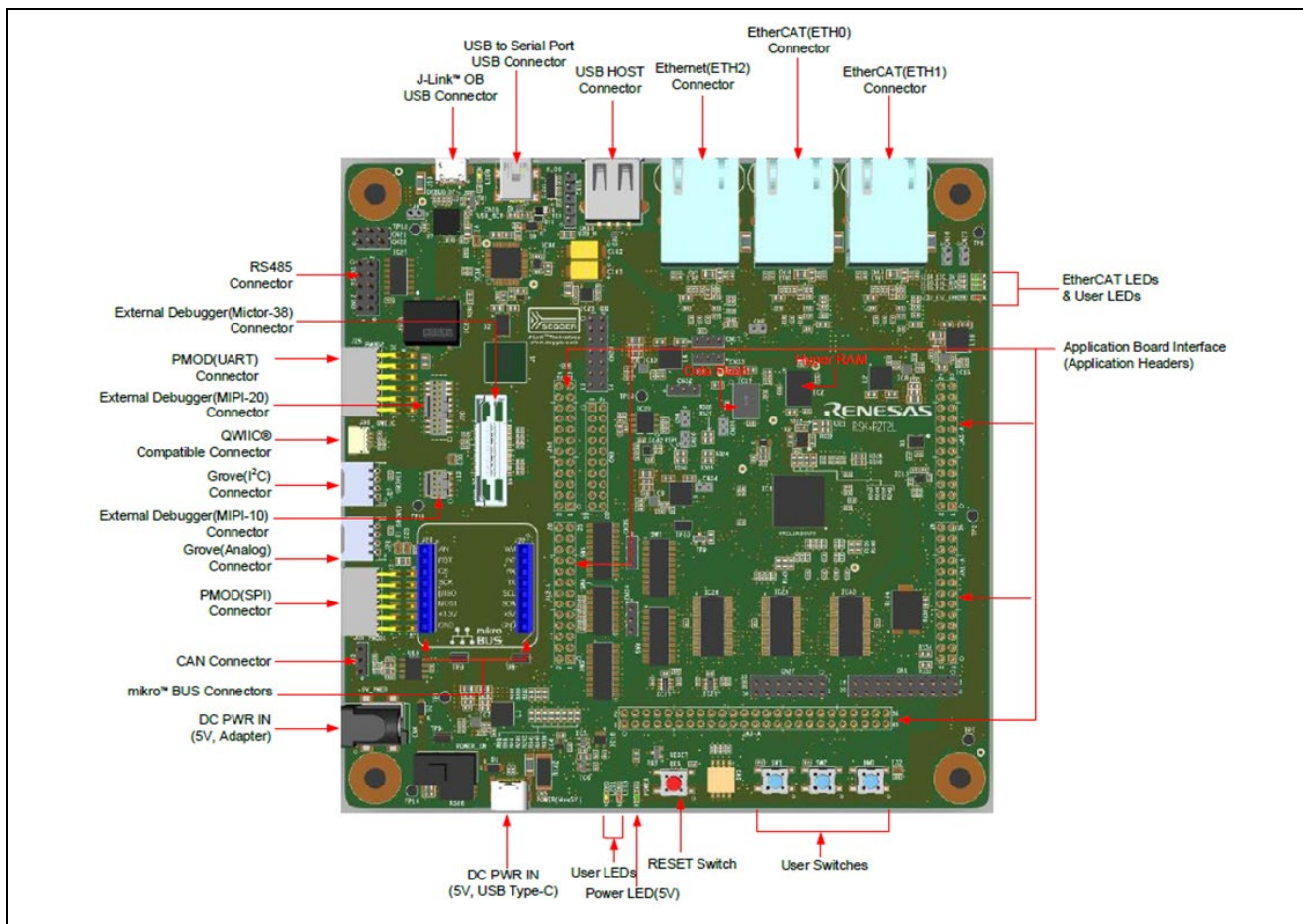


Figure 4.3 RZ/T2L RSK board layout

1. DIP SW と Jumper SW の設定

Renesas Starter Kit+ for RZ/T2L ユーザーズマニュアル「[6. コンフィグレーション](#)」の設定です。

- J-Link は micro USB へ、I-jet は J20 に接続し、J9 で切り替えます。
- 電源の供給には、USB ケーブル (Type-C) または、AC/DC アダプターを使用してください。
 - USB ケーブルを使用する際は CN5 に接続をしてください。
 - AC/DC アダプターを使用する際は CN6 に接続してください。
- イーサネットケーブルは ETH2 に接続してください。
- 本ドキュメントに記載の有る「3つのインターフェース」、「3つのポート」などは、RZ/T2L 以外の評価ボードで有効になります。

4.4. デバッガの接続

評価ボードとデバッガを、使用ツールに合わせて接続してください。



Figure 4.4 Debugger Connection

5. ホスト PC の設定

5.1. インターフェースのプロパティ設定

次の手順で、3つのインターフェースのプロパティを、IP アドレスと、MAC アドレスを重複なく実行環境に合わせて設定します。

1. プロジェクトパッケージ内のソースを参考に、IP アドレス(ip_address)、サブネットマスク(subnet_mask)、デフォルトゲートウェイ(gateway_address)、MAC アドレス(hwaddr)、接続 LAN ポート番号(eth_port)、を確認してください。
2. エコーサーバーのホスト側ポート番号
プロジェクトパッケージ内のソースを参考に、エコーサーバーのホスト側ポート番号を確認してください。

「g_lwip_port_netif_cfg」の配列データは、各評価ボードで LAN ポートに接続予定のデータを設定します。設定したデータの LAN ポートが未接続でも問題ありません。

なお、RZ/T2L は、LAN 接続可能なポート数は 1 です。

詳細については、「[7. サンプルアプリケーションのデモンストレーション](#)」を参照ください。

下記ソースファイルの内容に、e² studio 及び EWARM の違いはありません。

- ▶ src/renesas/application/lwip_port_instance.c
- ▶ src/renesas/application/lwip_port_main.c

```

94  /* Initial settings data such as IP address and MAC address */
95  static lwip_port_netif_ip_cfg_t g_lwip_port_netif_cfg[LWIP_NIC_MAX] =
96  {
97      {
98          #if LWIP_NIC_MAX > 1
99              .p_next      = &g_lwip_port_netif_cfg[1],
100          #else
101              .p_next      = NULL,
102          #endif
103              .dhcp        = LWIP_PORT_DHCP_DISABLE,
104              .ip_address   = PP_HTONL(LWIP_MAKEU32(192,168, 10,170)),
105              .subnet_mask  = PP_HTONL(LWIP_MAKEU32(255,255,255, 0)),
106              .gateway_address = PP_HTONL(LWIP_MAKEU32(192,168, 10, 1)),
107              .hwaddr       = {0x00,0x11,0x22,0x33,0x44,0x55},
108              .p_host_name  = gp_lwip_port0_hostname_0,
109              .eth_port     = ETHER_PORT_0,
110          },
111          #if LWIP_NIC_MAX > 1
112          {
113              .p_next      = &g_lwip_port_netif_cfg[2],
114              .dhcp        = LWIP_PORT_DHCP_DISABLE,
115              .ip_address   = PP_HTONL(LWIP_MAKEU32(192,168, 10,171)),
116              .subnet_mask  = PP_HTONL(LWIP_MAKEU32(255,255,255, 0)),
117              .gateway_address = PP_HTONL(LWIP_MAKEU32(192,168, 10, 2)),
118              .hwaddr       = {0x00,0x11,0x22,0x33,0x44,0x77},
119              .p_host_name  = gp_lwip_port0_hostname_1,
120              .eth_port     = ETHER_PORT_1,
121          },
122          #endif
123          #if LWIP_NIC_MAX > 2
124          {
125              .p_next      = NULL,
126              .dhcp        = LWIP_PORT_DHCP_DISABLE,
127              .ip_address   = PP_HTONL(LWIP_MAKEU32(192,168, 10,172)),
128              .subnet_mask  = PP_HTONL(LWIP_MAKEU32(255,255,255, 0)),
129              .gateway_address = PP_HTONL(LWIP_MAKEU32(192,168, 10, 3)),
130              .hwaddr       = {0x00,0x11,0x22,0x33,0x44,0x88},
131              .p_host_name  = gp_lwip_port0_hostname_2,
132              .eth_port     = ETHER_PORT_2,
133          },
134          #endif
135      };
136

```

Figure 5.1 Static IP address

```

41  * Macro definitions
42
43  #define TCP_SERVER_PORT          (65000)
44  #define TCP_SERVER_TASK_PRIORITY (3 + LWIP_TASK_PRIORITY_BASE)
45  #define TCP_SERVER_TASK_STACK_SIZE (1024)
46  #define TCP_SERVER_TASK_NAME     "TCP Server task"
47  #define TCP_SERVER_CLIENT_MAX    10
48  #define TCP_SNTP_TASK_PRIORITY   (2 + LWIP_TASK_PRIORITY_BASE)

```

Figure 5.2 Port number

5.2. Ethernet 通信のモード

PC と評価ボード間の通信には、以下のモードが有り、組み合わせなどを含め、用途に合わせてご使用ください。

1. ルーティングモード

本モードは、PC と通信で、評価ボード内の Ethernet ドライバからの受信フレームの MAC アドレスを NETIF と紐づけて lwIP に受信通知します。

通常、強制的に、このモードが有効になりますが、下記「Management port specific tag」が「Enable」の場合に、設定内容が有効になります。

2. Management port specific tag

FSP_SC の「Stacks」タグで、「g_ether0 Ethernet(r_gmac)」の「g_ether0 Ethernet(r_ethsw)」内で、「Management port specific tag」の設定は以下のようになります。

➤ Enable

EtherSW の設定や設定値の参照が可能になりますが、送受信時に毎フレーム、データコピーが発生しますので、オーバーヘッドが大きくなります。

また、この設定で、上記ルーティングモードを無効にすることが可能になり、変更は以下のようになります。

ソースファイル : src/renesas/module/lwip_port/um_lwip_port_cfg.h

マクロ名 : UM_LWIP_PORT_ROUTING_SET

設定値 : 1 (有効) 0 (無効)

ソースファイル : src/renesas/oss_deps/lwip/lwipopts.h

マクロ名 : ETH_PAD_SIZE 8

無効にした場合、lwIP の NETIF と LAN ポートが、P2P の接続になります。

詳細は、[8.ソフトウェア仕様](#) を参照ください。

➤ Disabled

EtherSW で強制的なポート間転送など、設定の変更が必要な場合以外、この設定がデフォルトです。

なお、Disabled の場合自動的に、ルーティングモードは有効になります。

また、以下のマクロ定義の数を確認してください。

ソースファイル : src/renesas/oss_deps/lwip/lwipopts.h

マクロ名 : ETH_PAD_SIZE 0

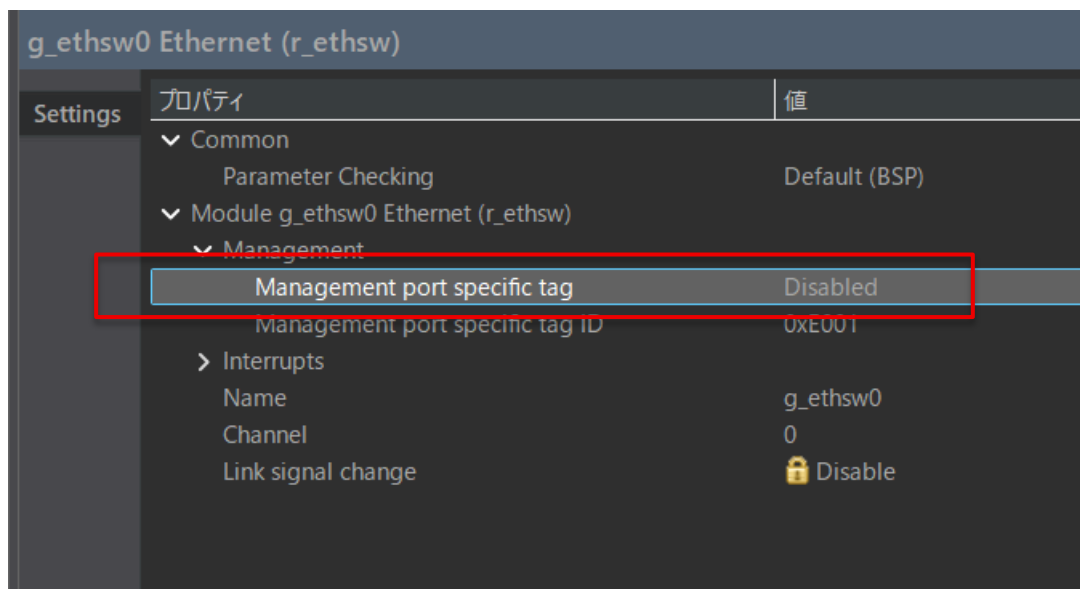


Figure 5.3 Management port specific tag

```
34  /** Routing packets to lwIP based on MAC address. 1 = Enable 0 = Disable */  
35  #define UM_LWIP_PORT_ROUTING_SET (1)  
36
```

Figure 5.4 Routing mode

```
47  /* If the MAC frame mode containing EtherSW management data is ETHSW_SPECIFIC_TAG_ENABLE, this data will be 8. */  
48  #define ETH_PAD_SIZE 0  
49
```

Figure 5.5 ETH_PAD_SIZE

6. サンプルアプリケーションの実行

EWARM のプロジェクトを実行する場合は、[6.2.EWARM のサンプルプロジェクトの設定](#)へ進んでください。

6.1. e² studio のサンプルプロジェクトの設定

6.1.1. e² studio の起動

1. e² studio を開き、ワークスペースのディレクトリを選択します。
2. File タブの”Open Projects from File System...”をクリックします。

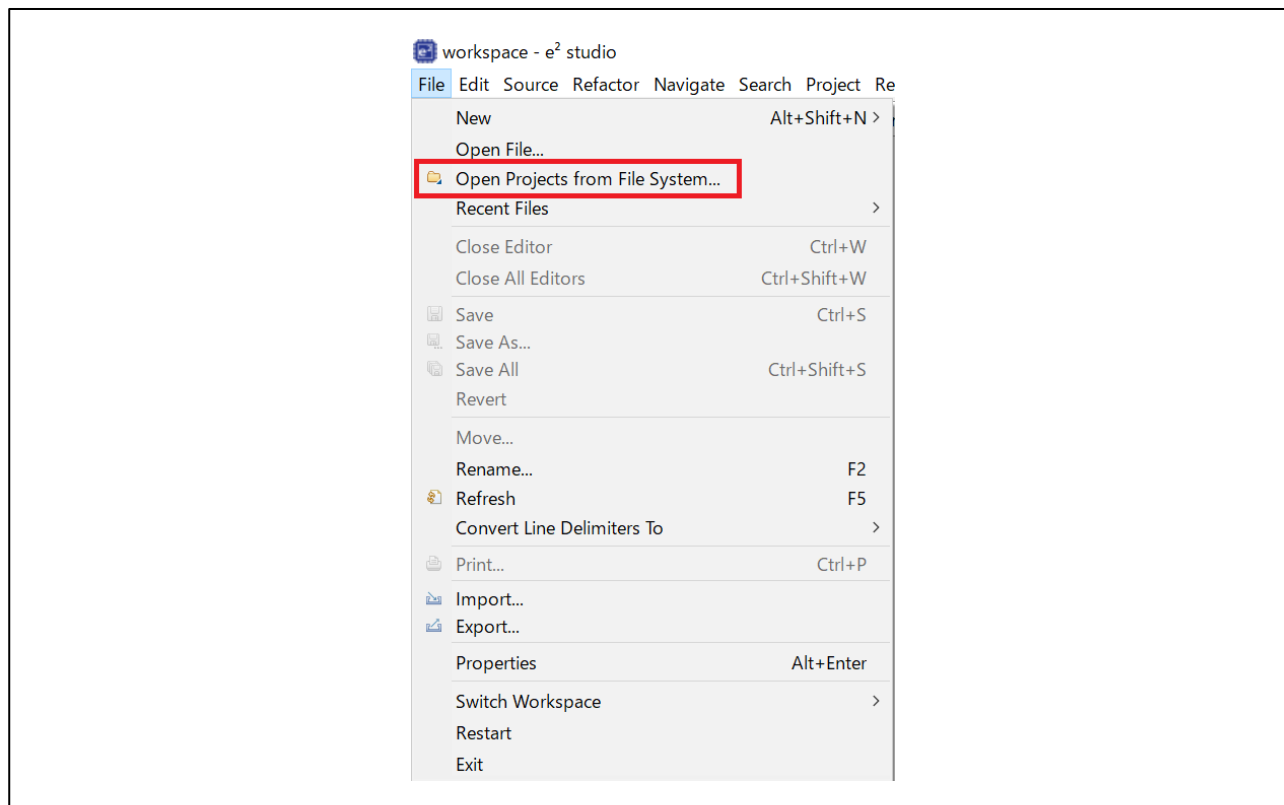


Figure 6.1 e²studio File tab

3. プロジェクトフォルダをインポートします。

▶ RZ/T2M

以下 2 つのプロジェクトをインポートします。

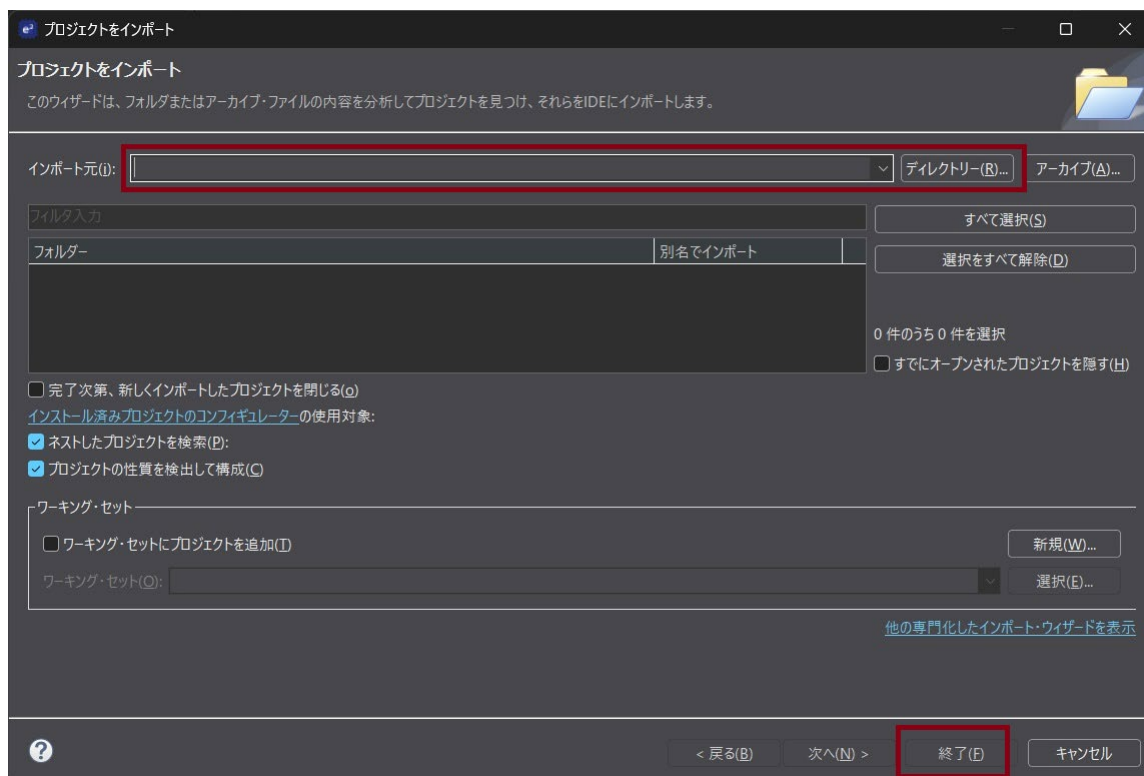
- ・ 下記プロジェクトでは、LED の点滅以外の機能はありません。
プロジェクト名 : RZT2M_RSK_lwIP_cpu0 (以降「CPU0」と記述)
フォルダー : project\rzt2m_rsk\e2studio\RZT2M_RSK_lwIP_cpu0
- ・ 下記プロジェクトでは、「[7. サンプルアプリケーションのデモンストレーション](#)」に関連する機能を含みます。
プロジェクト名 : RZT2M_RSK_lwIP_cpu1 (以降「CPU1」と記述)
フォルダー : project\rzt2m_rsk\e2studio\RZT2M_RSK_lwIP_cpu1

▶ RZ/N2L

プロジェクト名 : RZN2L_RSK_lwIP_single (以降「CPU0」と記述)
フォルダー : project\rzn2l_rsk\e2studio をインポートします。

▶ RZ/T2L

プロジェクト名 : RZT2L_RSK_lwIP_single (以降「CPU0」と記述)
フォルダー : project\rzt2l_rsk\e2studio をインポートします。

Figure 6.2 Import project on e²studio

6.1.2. コードの生成とビルド方法

コード生成前、Project Explorer ウィンドウ内に script フォルダがある場合、以下の注意が必要です。

- ・ script フォルダ内に ld ファイルがある場合、フォルダ内のファイルと異なるファイルを使用すると、ビルドエラーなど、想定外の結果になる可能性が有ります。
- ・ ブートモードや、FSP バージョンを変更する場合、コード生成時に、ld ファイルを書き換えますので、事前に保存し、ビルドを行う前に、事前に保存した ld ファイルへ戻してください。変更が無い場合は、書き換えは有りません。

1. Configuration.xml をクリックします。

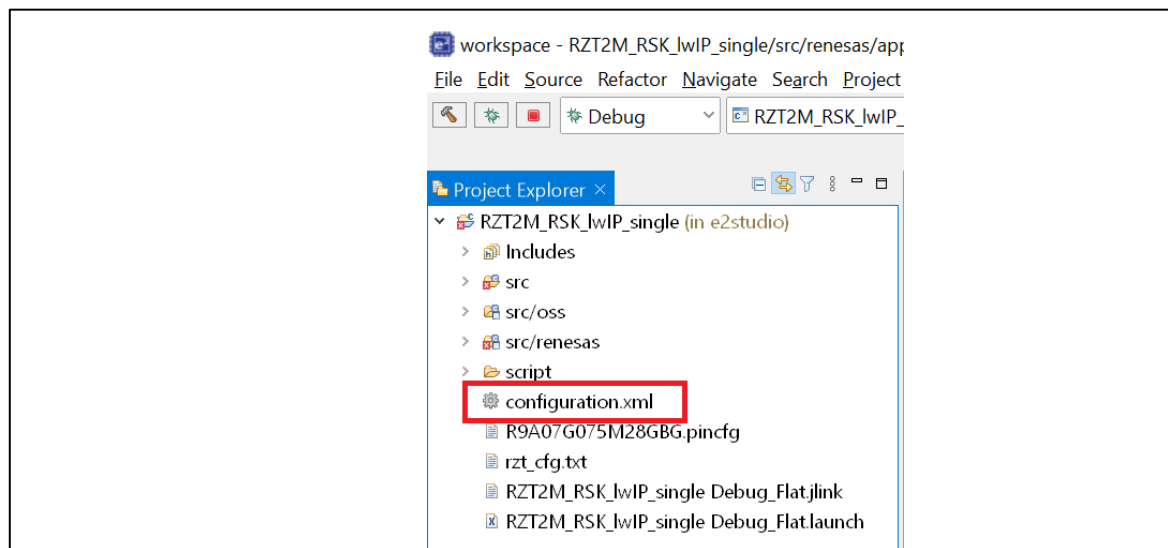


Figure 6.3 Configuration

2. 表示された「FSP Smart Configurator」画面内の「BSP」タブから、以下の設定を確認し「Generate Project Content」をクリックしてください。ワークスペースに、生成されたフォルダが追加されます。
RZ/T2M の場合、必ず CPU0 から行ってください。

- RZ/T2M
CPU0 及び CPU1 は同じ設定です。
Board : RSK+RZT2M (RAM execution without flash memory)
- RZ/N2L
Board : RSK+RZN2L (RAM execution without flash memory)
- RZ/T2L
Board : RSK+RZT2L (RAM execution without flash memory)

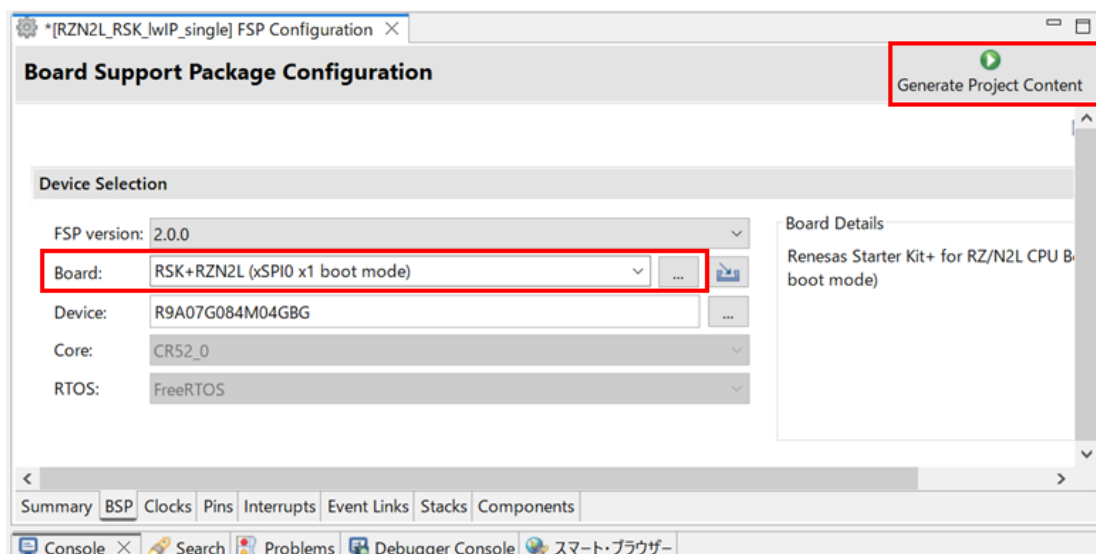


Figure 6.4 Generate Project Content

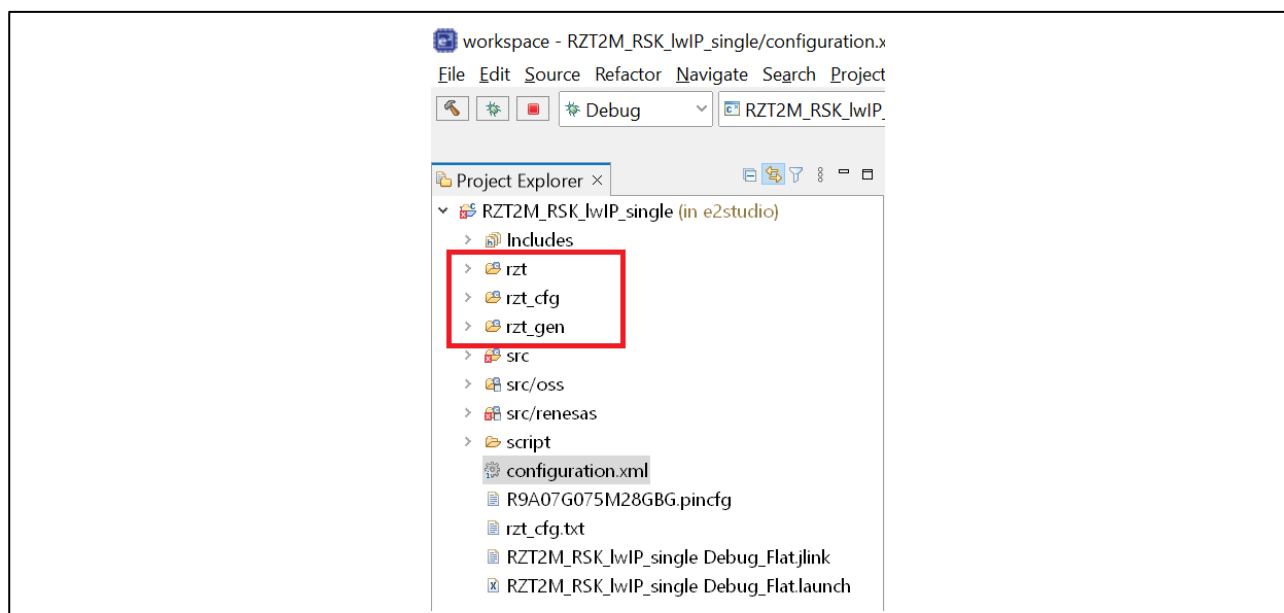


Figure 6.5 Generate project folder

3. ツールバーにある Build ボタンをクリックしてプロジェクトをビルドします。ビルドメッセージのログにエラーがないことを確認してください。
RZ/T2M の場合、必ず CPU0 から行ってください。

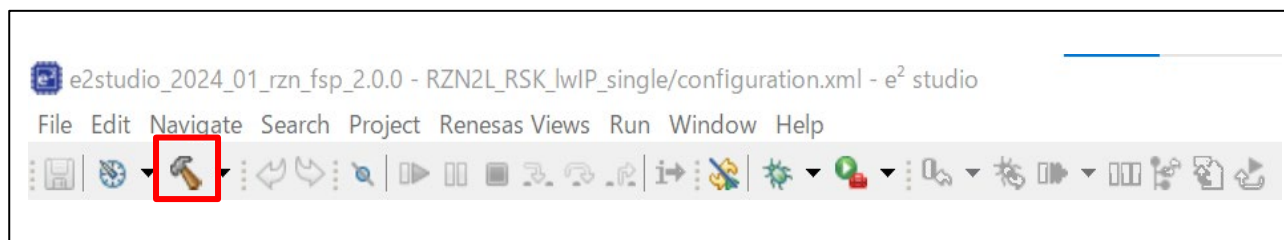


Figure 6.6 Build button

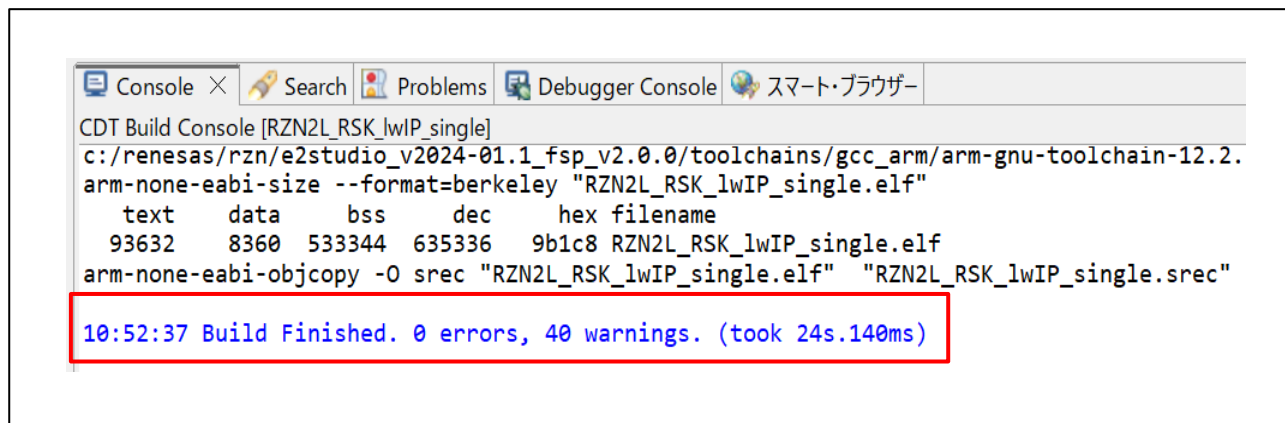


Figure 6.7 Build message

6.1.3. アプリケーションのダウンロードとデバッガの実行

各プロジェクトを以下の手順で実行してください。

1. CPU0 のプロジェクトにフォーカスを当て、ツールバーのデバッグボタンをクリックして、CPU0 のプログラムのダウンロードと、デバッカーを起動してください。

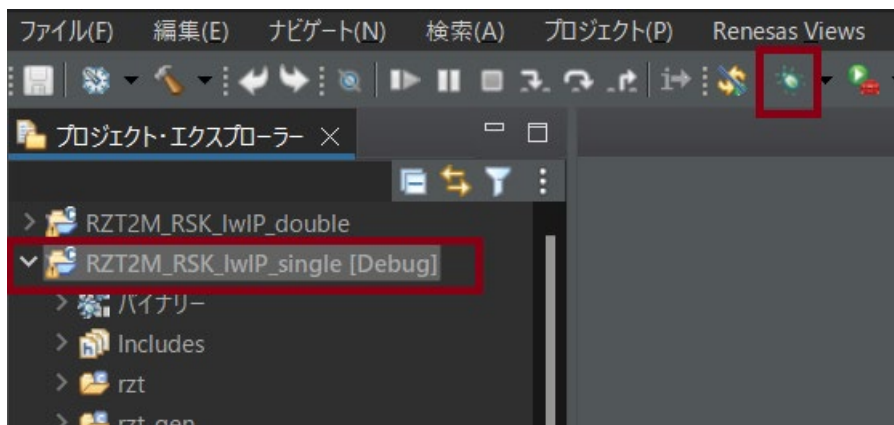


Figure 6.8 CPU0 Debug butto

2. ダイアログの対応
デバッガ起動時、使用環境等により、エラー表示以外のダイアログが表示された場合、「切り替え」「はい」「OK」などを選択して閉じてください。
3. プログラムは起動時に設定されているブレイクポイントで停止しますので、そのままツールバーにある実行ボタンをクリックしてください。
T2M の場合、続いて CPU1 を実施してください。
実行前に、「[7. サンプルアプリケーションのデモンストレーション](#)」を参考に PC を接続してください。

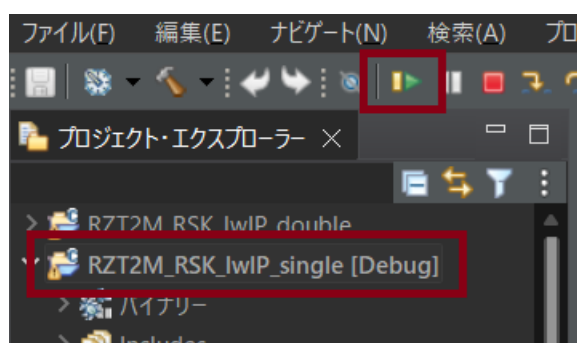


Figure 6.9 CPU0 Run button

4. 続いて、CPU1 のプロジェクトにフォーカスを当て、ツールバーのデバッグボタンをクリックして、CPU1 のプログラムのダウンロードと、デバッカーを起動してください。
その際、以下のダイアログが表示されますので「いいえ」をクリックしてください。

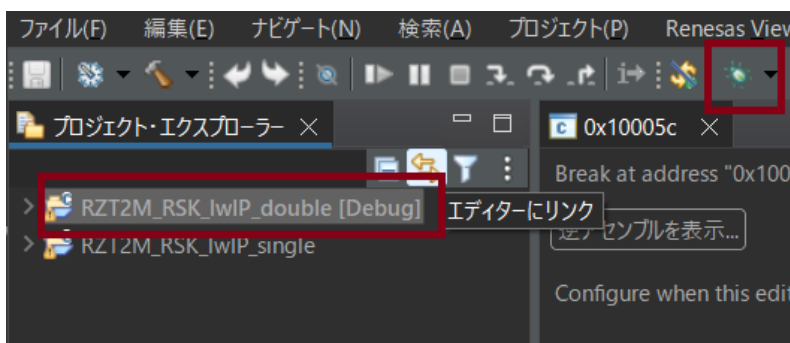


Figure 6.10 CPU1 Debug button

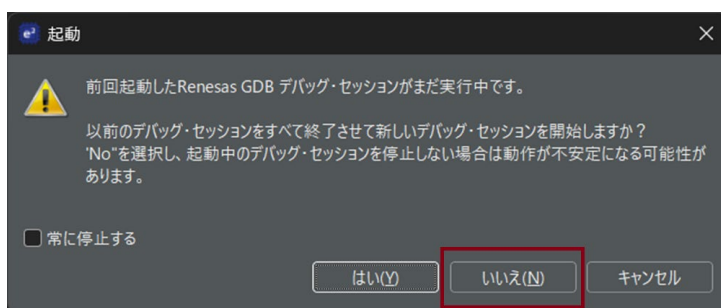


Figure 6.11 CPU1 Duplicate confirmation

5. その他のダイアログ
上記以外で、使用環境等により、エラー表示以外のダイアログが表示された場合、「切り替え」
「はい」「OK」などを選択して閉じてください。
6. プログラムは起動時に設定されているブレイクポイントで停止しますので、そのままツールバーにある実行ボタンをクリックしてください

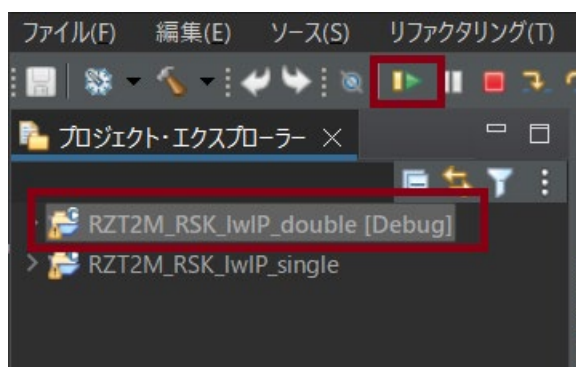


Figure 6.12 CPU1 Run button

6.2. EWARM のサンプルプロジェクトの設定

6.2.1. EWARM の起動

下記プロジェクトファイル毎に、以下の 1 項から 3 項を実施してください。

1. EWARM を開きます。
2. File タブの“Open Workspace...”をクリックします。

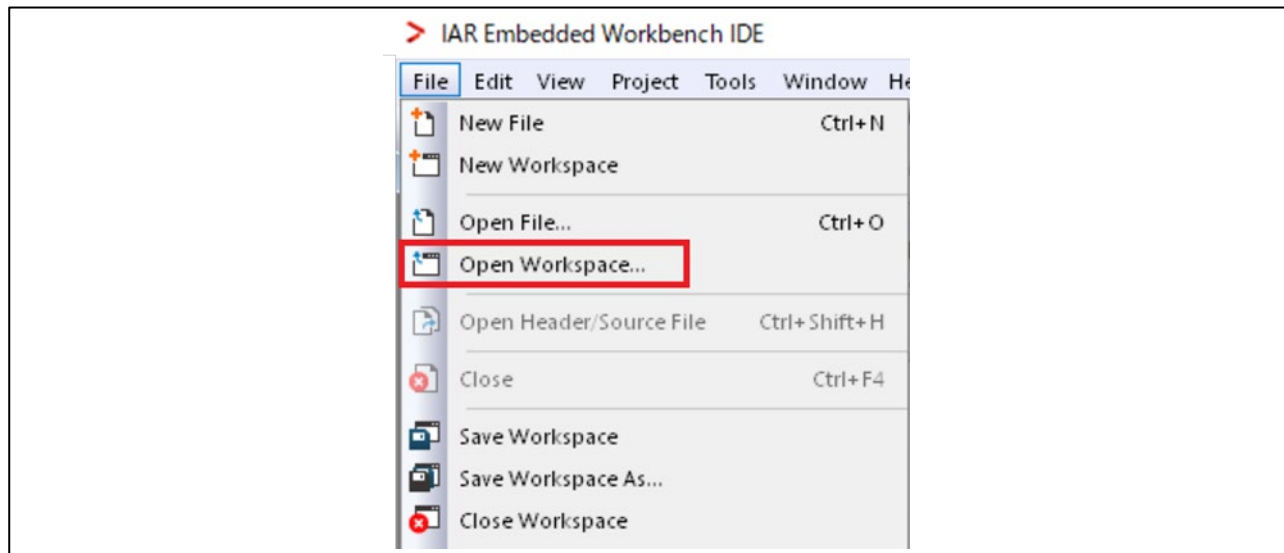


Figure 6.13 EWARM file tab

3. ワークスペースファイル(.eww)を選択し、Open ボタンをクリックします。

➤ RZ/T2M

以下 2 つのプロジェクトを別々の EWARM で開きます。

- ・下記プロジェクトでは、LED の点滅以外の機能はありません。

プロジェクトファイル : RZT2M_RSK_lwIP_cpu0.eww (以降「CPU0」と記述)

フォルダー : project\rzt2m_rsk\ewarm\RZT2M_RSK_lwIP_cpu0

- ・下記プロジェクトでは、「[7. サンプルアプリケーションのデモンストレーション](#)」に関連する機能を含みます。

プロジェクトファイル : RZT2M_RSK_lwIP_cpu1.eww (以降「CPU1」と記述)

フォルダー : project\rzt2m_rsk\ewarm\RZT2M_RSK_lwIP_cpu1

➤ RZ/N2L

プロジェクトファイル : RZN2L_RSK_lwIP_single.eww (以降「CPU0」と記述)

フォルダー : project\rzn2l_rsk\ewarm をインポートします。

➤ RZ/T2L

プロジェクトファイル : RZT2L_RSK_lwIP_single.eww (以降「CPU0」と記述)

フォルダー : project\rzt2l_rsk\ewarm をインポートします。

6.2.2. コードの生成とビルド方法

1. コード生成時の注意事項。

- ・ script フォルダ内にある、icf ファイル以外を使用した場合、ビルドエラーなど、想定外の結果になる可能性が有ります。
- ・ ブートモードや、FSP バージョンを変更する場合、コード生成時に、icf ファイルを書き換えますので、事前に保存し、ビルドを行う前に、事前に保存した icf ファイルへ戻してください。変更が無い場合は、書き換えは有りません。
- ・ RZ/T2M の場合、CPU0 と CPU 1 のプロジェクトでビルドしてください。
- ・ RZ/T2M の場合、「FSP Smart Configurator」を 2 プロジェクト同時に開かないでください。
- ・ 「FSP Smart Configurator」をデフォルトのインストールパス以外にインストールしている場合は、プロジェクトフォルダ内にある buildinfo.ipcf ファイルを開き、RASC_EXE_PATH に設定されているパスを、実際にインストールした rasc.exe の場所に合わせて編集してください。
注意： buildinfo.ipcf を編集した後は、変更を反映させるために EWARM (IAR Embedded Workbench) を再起動してください。
- ・ Tools タブの「FSP Smart Configurator」をクリックします。EWARM で FSP Smart Configurator をまだ設定していない場合は、RZ/T2M, RZ/N2L, RZ/T2L Getting Started with Flexible Software Package (R01AN6434EJ)を参照してください。セクション 5.4 で設定方法が説明されています。

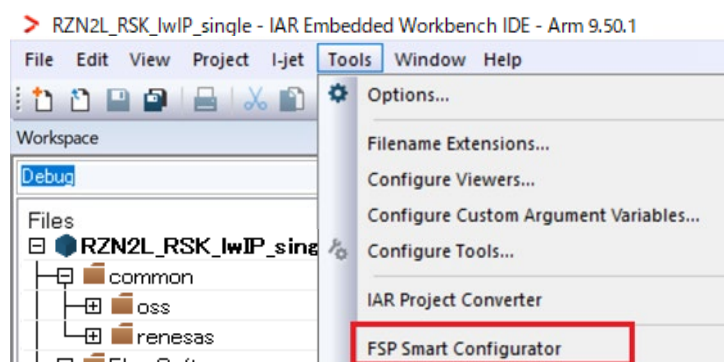


Figure 6.14 Tools tab

2. コード生成手順

「FSP Smart Configurator」をクリック後、表示される画面内の「BSP」タブから、以下の設定を確認し「Generate Project Content」をクリックしてください。生成されるフォルダーが追加されます。

- RZ/T2M
CPU0 及び CPU1 は同じ設定です。
Board : RSK+RZT2M (RAM execution without flash memory)
- RZ/N2L
Board : RSK+RZN2L (RAM execution without flash memory)
- RZ/T2L
Board : RSK+RZT2L (RAM execution without flash memory)

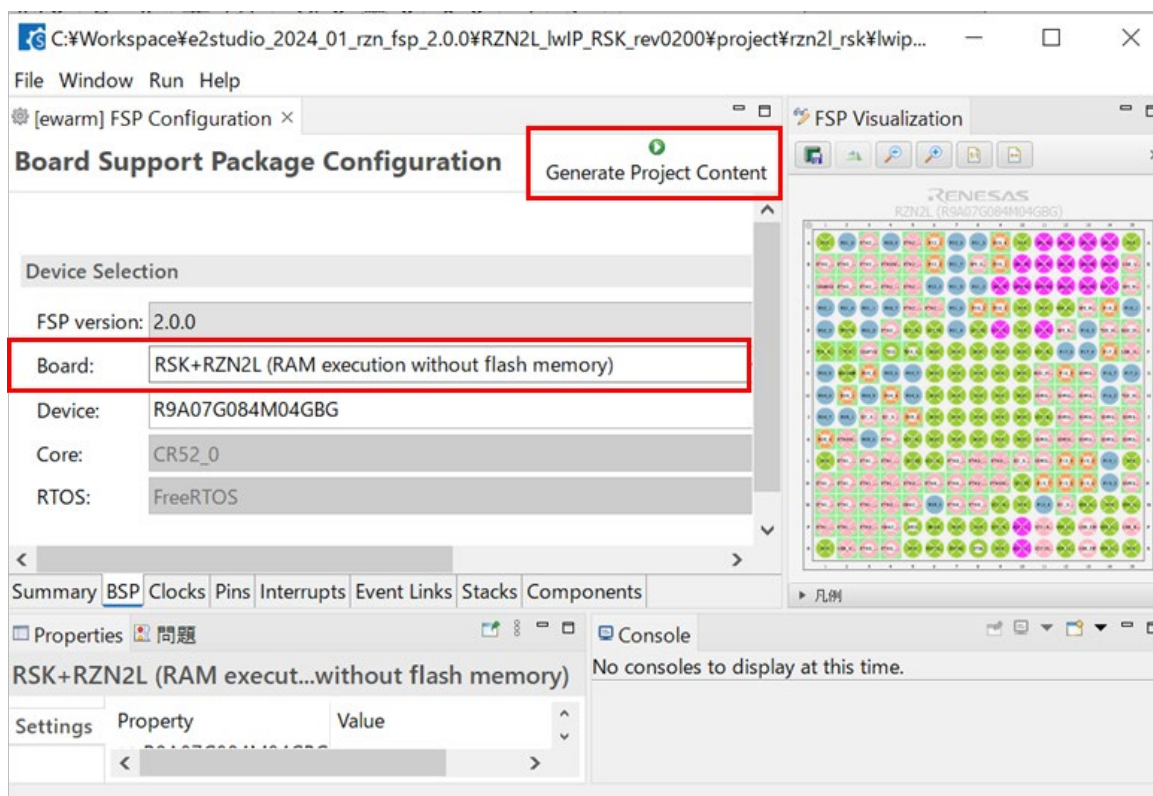


Figure 6.15 FSP SC Smart Configurator

3. ビルド手順

Project タブの“Make”をクリックするか、ボタンをクリックしてビルドします。ビルドが完了すると、ビルドメッセージがビルドコンソールウィンドウに表示され、コンパイルターゲットファイルとエラー/警告の数が表示されます。

なお、RZ/T2M の場合、CPU0 から実行してください。

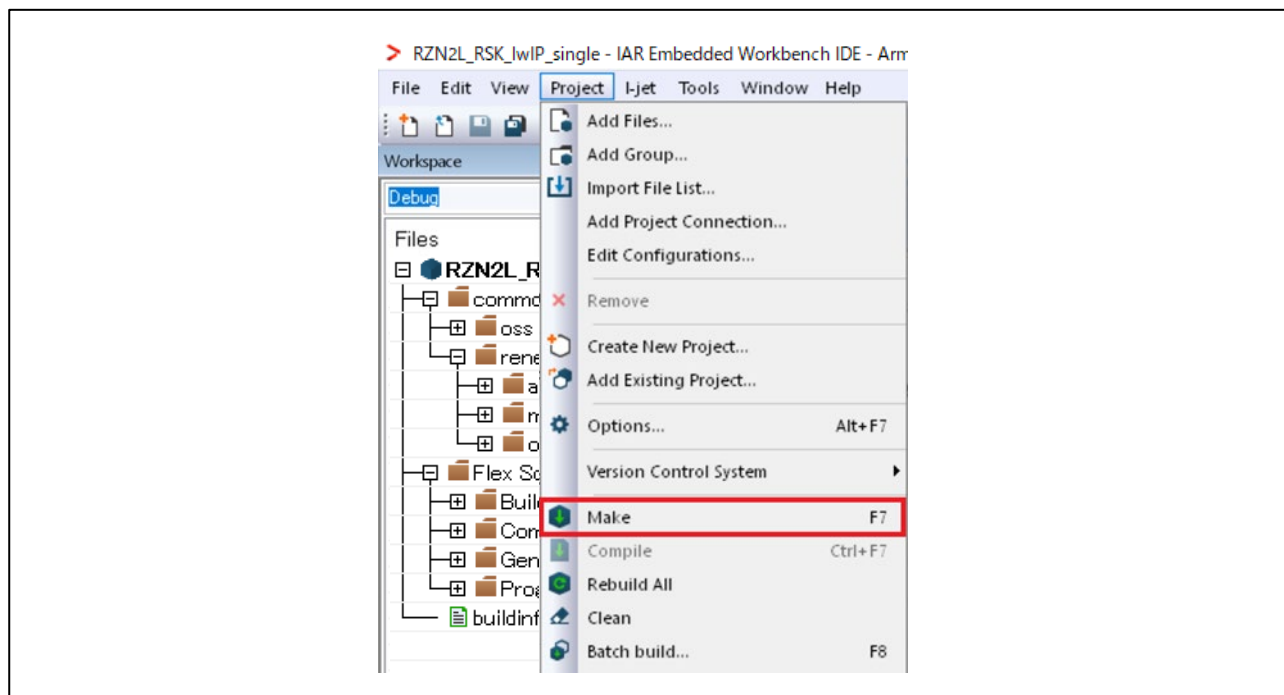


Figure 6.16 Make button 1

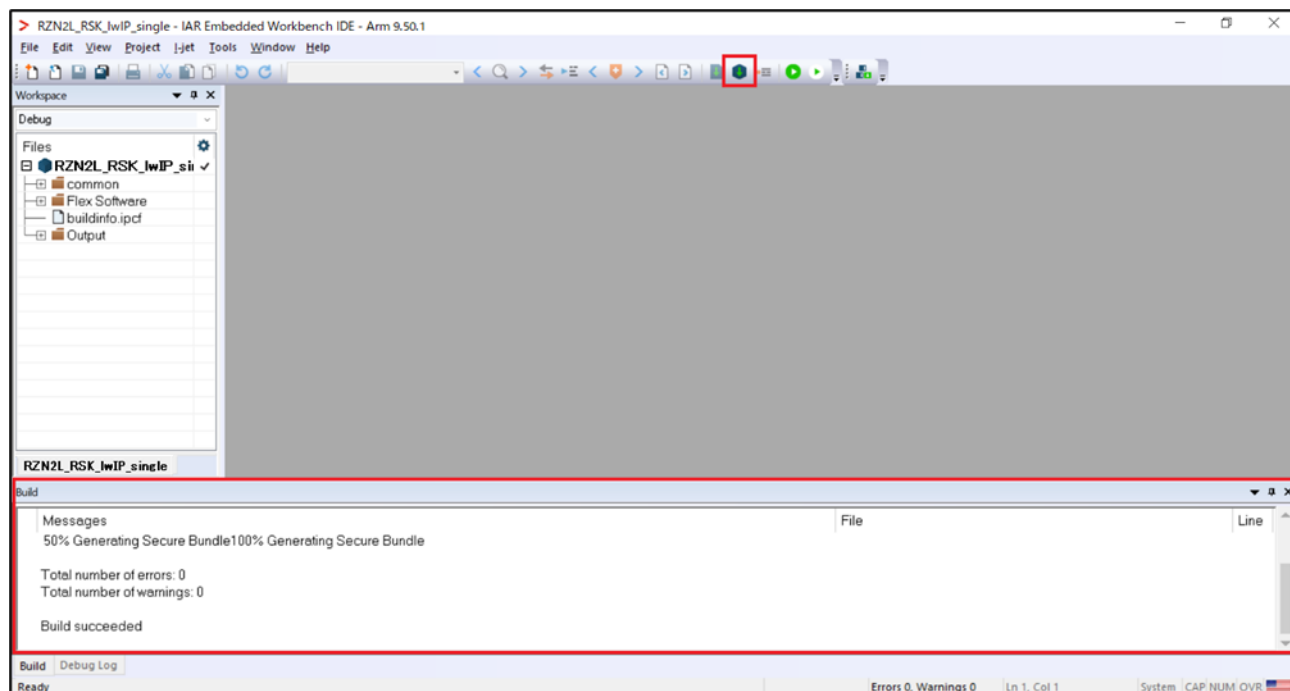


Figure 6.17 Make button 2 and Build console

6.2.3. アプリケーションのダウンロードとデバッグの実行（シングルコア版）

1. アプリケーションのダウンロードとデバッグの実行（シングルコア版）

メニューバー「プロジェクト」の「ダウンロードしてデバッグ」をクリックするか、Debug ボタンをクリックして、アプリケーションプログラムをダウンロードし、デバッグが起動後、プログラムは設定済みのブレークポイントで停止します。

RZ/T2M の場合、この項を飛ばして [6.2.4. アプリケーションのダウンロードとデバッグの実行（デュアルコア版）](#) へ進んでください。

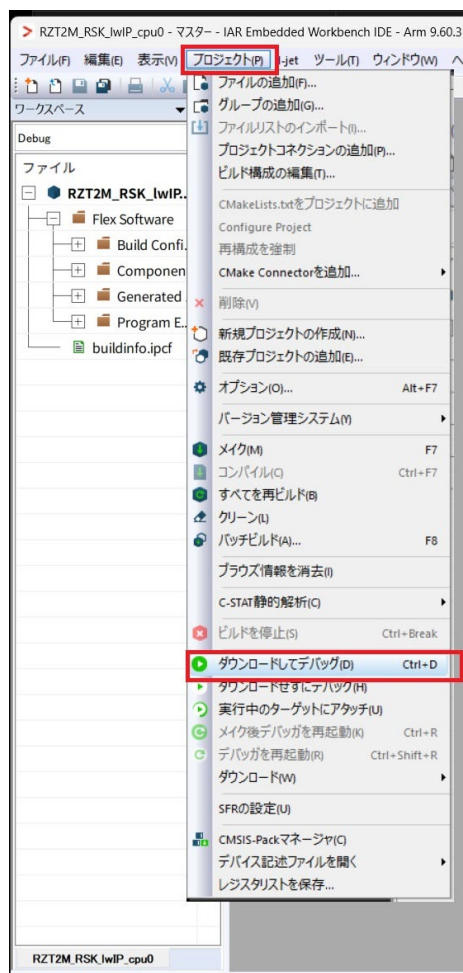


Figure 6.18 Download and Debug button

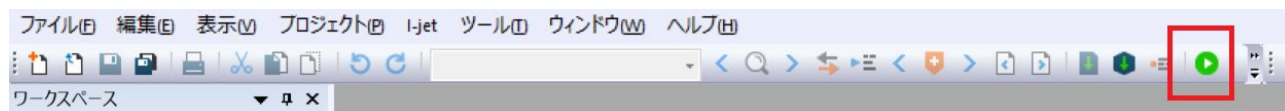


Figure 6.19 Debug button

2. Go ボタンをクリックします。プログラムが正常に動作している場合は、TCP/IP 接続要求で待機します。

実行前に、[7. サンプルアプリケーションのデモンストレーション](#)参考に PC を接続してください。

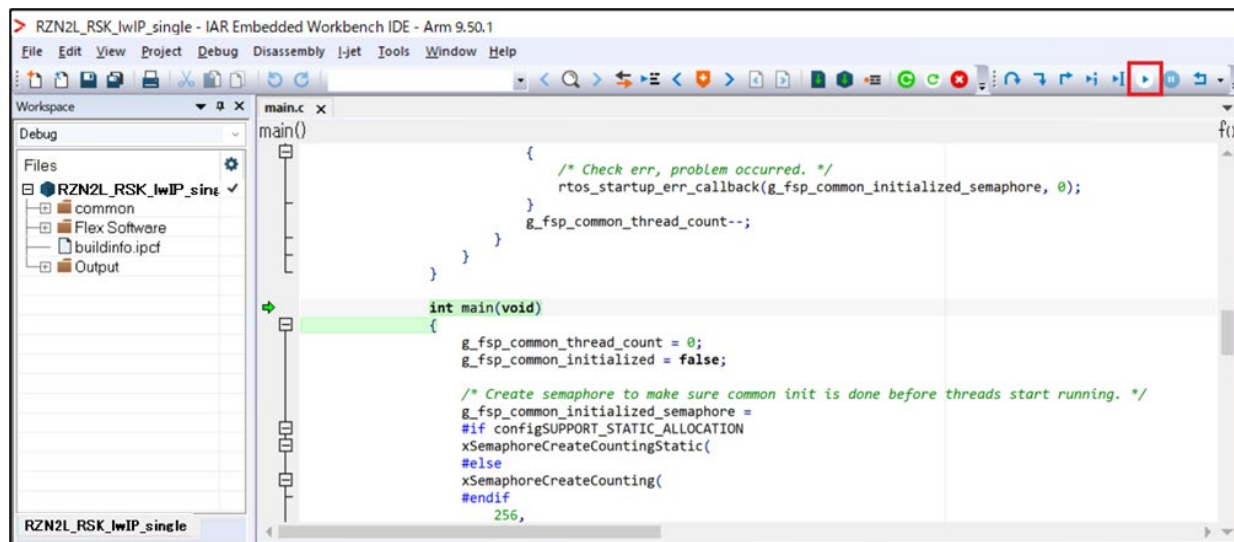


Figure 6.20 Break point and Go button

6.2.4. アプリケーションのダウンロードとデバッグの実行（デュアルコア版）

以下の手順を実施してください。

1. CPU1 用プロジェクトの EWARM を閉じてください。
2. メニューバー「プロジェクト」の「ダウンロードしてデバッグ」をクリックするか、Debug ボタンをクリックして、アプリケーションプログラムをダウンロードし、デバッグが起動後、プログラムは設定済みのブレークポイントで停止します。

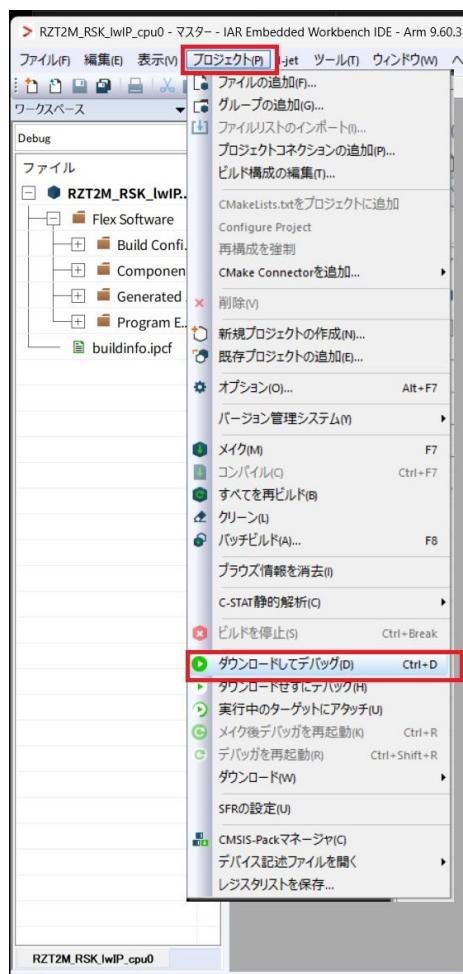


Figure 6.21 Download and Debug button

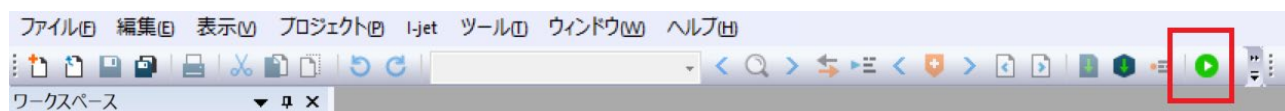


Figure 6.22 CPU0 Debug button

3. CPU0 のダウンロードと共に、CPU0 がマスターとなり、CPU1 のプロジェクトがスレーブとして自動起動し、双方のプロジェクトで、設定済みのブレイクポイントで停止します。
4. CPU0 のプロジェクトから実行ボタンをクリックし、続いて CPU1 のプロジェクトを同様に実行してください。

実行前に、UART にてターミナルソフトを起動した PC に接続してください。

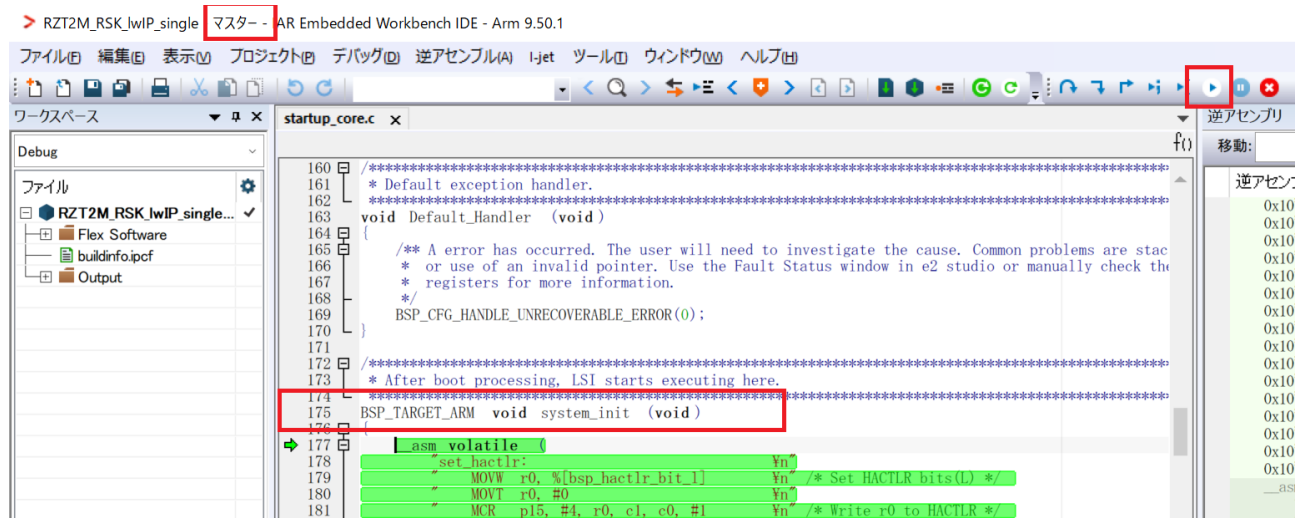


Figure 6.23 CPU0 Go button

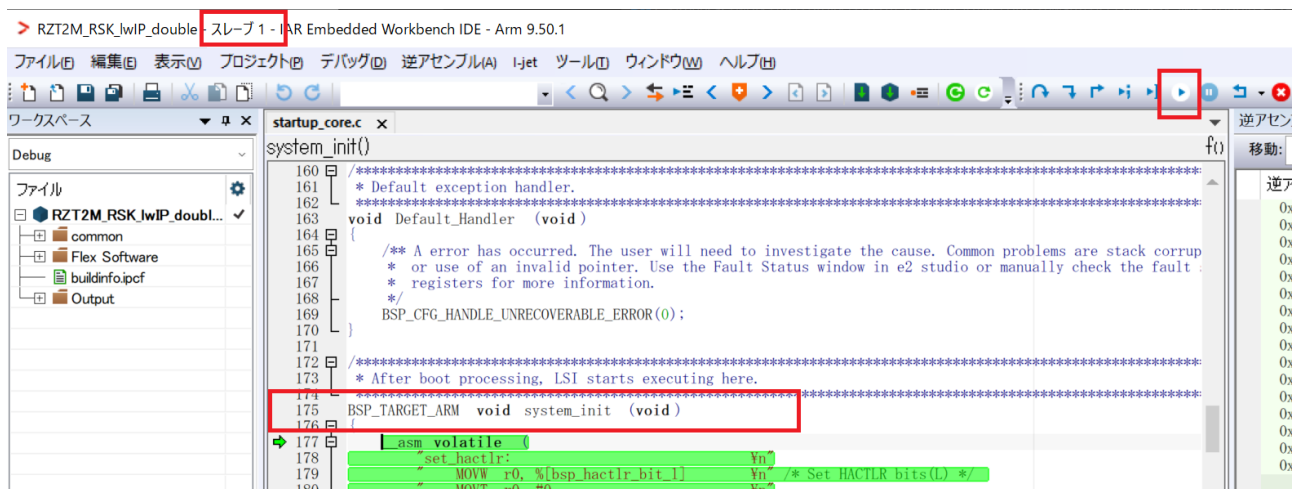


Figure 6.24 CPU1 Go button

7. サンプルアプリケーションのデモンストレーション

サンプルプログラムではデモンストレーションとして、エコーサーバー機能や lwip のアドオンとして「apps」フォルダーの「iperf」と「SNTP」が実行できます。

7.1. PC と接続

以下の手順で、評価ボードと通信用 PC と接続してください。

1. [ハードウェア設定](#)で示しました「usb to serial port」と PC を USB ケーブルでつないでください。
2. 評価ボードからの LOG データなどを表示するため、「Tera Term」などのターミナルソフトを以下のプロトコルで起動してください。

ここでは、「Tera Term」を使用します。

- ・ポート： 評価ボードと接続して追加されたポート
- ・スピード： 115200
- ・その他： DATA8 / NONE / 1bit / ローカルエコーOFF

3. [ハードウェア設定](#)で示しました LAN ポートと PC の NIC へ LAN ケーブルでつないでください。
RZ/T2L は、1 ポートのみ有効です。

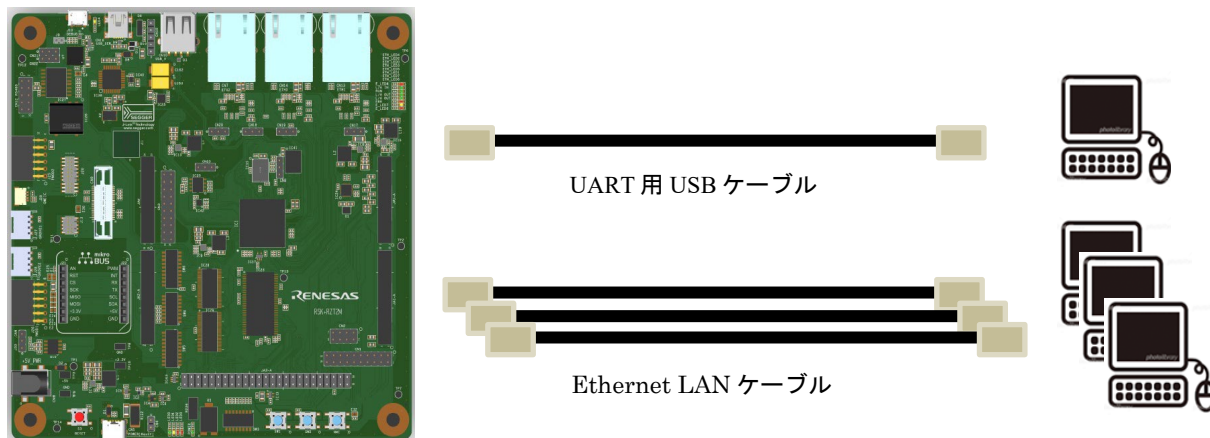


Figure 7.1 Communication terminal connection

4. 起動画面

上記の図で、UART と PC を接続し評価ボードを起動すると、以下の画面が PC のターミナルソフトに表示されます。

```
>>Info : Started Serial I/O interface.
>>Info : lwIP initialization successful
>>Info : Initialized Ethernet network interface.
>>Info : Successfully registered lwIP as a callback to Ethernet.
>>Info : Started Ethernet network interface.
>>Info : <<< Started sample application (v2.0) for lwIP Port. >>>
>>Info : =====
>>Info : lwIP API routing mode ENABLED
>>Info : lwIP ETH_PAD_SIZE = 0
>>Info : ETHSW_SPECIFIC_TAG_DISABLE
>>Info : =====
>>Info : Please connect the IPERF LAN cable to the port.
>>Info : link-up port( NIC_0 )
>>Info : link-up port( NIC_1 )
>>Info : link-up port( NIC_2 )
>>Info : << Start the SNTP client for 15 seconds. >>
>>Info : << SNTP client application completed [ 25.07.2026 07:08:18 ] >>
>>Info : << IPERF2 server application start >>
>>Info : << The echo server will be start. >>
```

通信オプション

LINK ステータス

SNTP

Figure 7.2 Startup screen

7.2. SNTP クライアント

本サンプルプログラムが起動した際に、NTP サーバーと接続可能になった場合 15 秒間時刻データ取得して表示します。15 秒以内にデータを取得出来ない場合、その旨の LOG を表示して、次の処理に進みます。

1. SNTP の設定

▶ NTP サーバーの IP アドレスの確認

NTP サーバーの IP アドレスを、下記ソースファイルの設定を参考に、使用環境に合わせて変更してください。

- ・ソースファイル : src/renesas/application/lwip_add_on_main.c
なお、e² studio 及び EWARM の違いはありません。

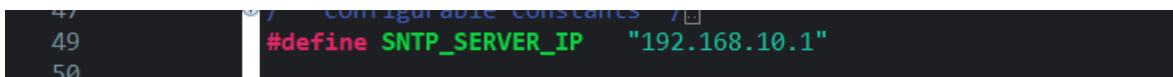


Figure 7.3 NTP server IP address

2. NTP サーバーの設定

本サンプルプログラムでは、Windows の NTP サーバー機能を使用して、動作確認をしました。通常は無効になっている NTP サーバーは、レジストリの変更で使用可能になります。

なお、NTP サーバーについては、サポート外の為、設定方法などは、Web 等で確認してください。

7.3. iperf クライアント／サーバー

本サンプルプログラムでは、常時 iperf を使用して、Ethernet のボーレートの計測を行えます。

- ・ PC 側アプリ : iperf2-1-9win(フリーソフト)
- ・ 評価ボード : lwIP の apps フォルダ内に有るアドオン

7.3.1. 通信速度の測定手順

以下の手順で計測します。

- ・ lwIP サンプルプログラムの起動

起動画面で以下のメッセージが表示されている。

「<< IPERF2 server application start >>」

- ・ PC のターミナルソフトから以下のコマンドで、サーバーを起動
「iperf-2-1-9-win.exe -s」
- ・ PC から以下のコマンドで、送信先の IP アドレスを確認後に 10 秒間送信
「iperf-2-1-9-win.exe -c 192.168.10.170」
- ・ PC からの通信が完了すると、評価ボード内で自動的に送信元(PC)のアドレスへ、10 秒間送信

送信及び受信の通信速度は、評価ボードのターミナルには表示しませんので、PC のターミナルで確認してください。

なお、iperf のパラメータは、以下のようになります。

▶ PC

- ・ クライアント : iperf-2-1-9-win.exe -c 192.168.10.170
- ・ サーバー : iperf-2-1-9-win.exe -s

▶ 評価ボード

- ・ クライアント : 受信時の送信元アドレスを使用し、その他はデフォルト
- ・ サーバー : IP アドレスを ANY としてし、その他はデフォルト

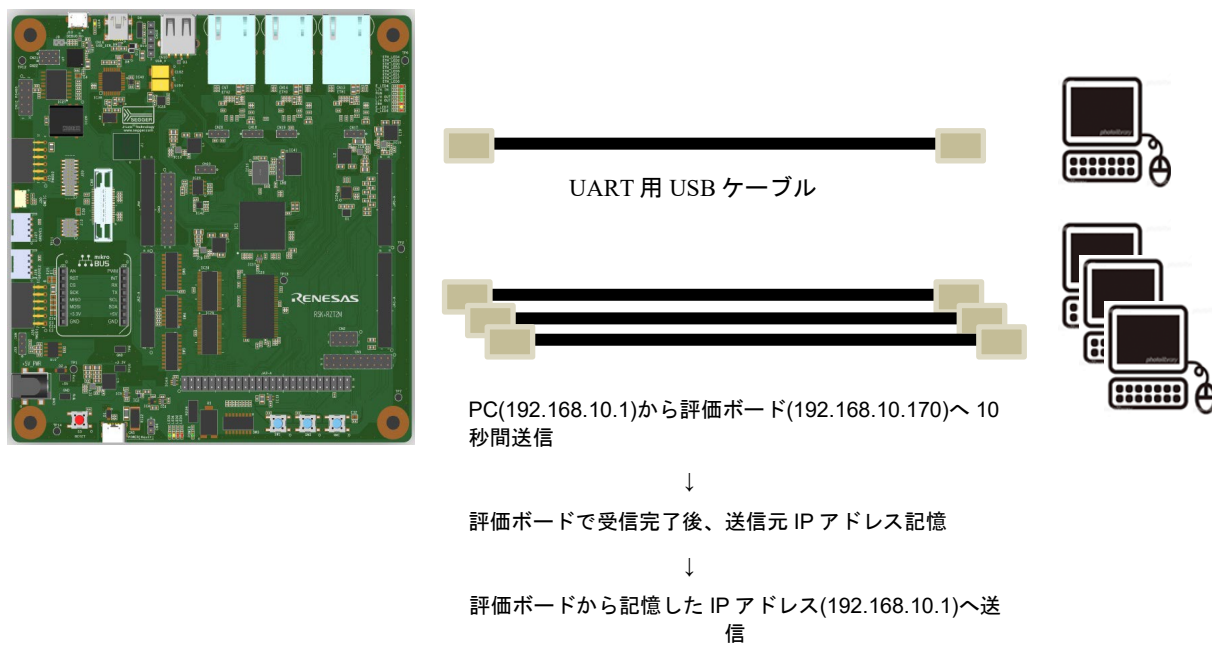


Figure 7.4 Communication sequence

7.4. エコーサーバー

本サンプルプログラムでは、常時エコーサーバーが起動していますので、起動後は PC のターミナルソフトから TELNET 通信が行えます。

- ・ TELNET ポート番号 : [5.1.Ethernet のプロパティ設定](#)
- ・ IP アドレス : [5.1.Ethernet のプロパティ設定](#) で設定した、いずれかの IP アドレス。
- ・ 接続最大数 : 10



Figure 7.5 Tera Term Configuration

8. ソフトウェア仕様

8.1. ソフトウェア構造

以下に、ソフトウェアの概要図を示します。

図内の凡例： () 対応ソースファイルの保存先

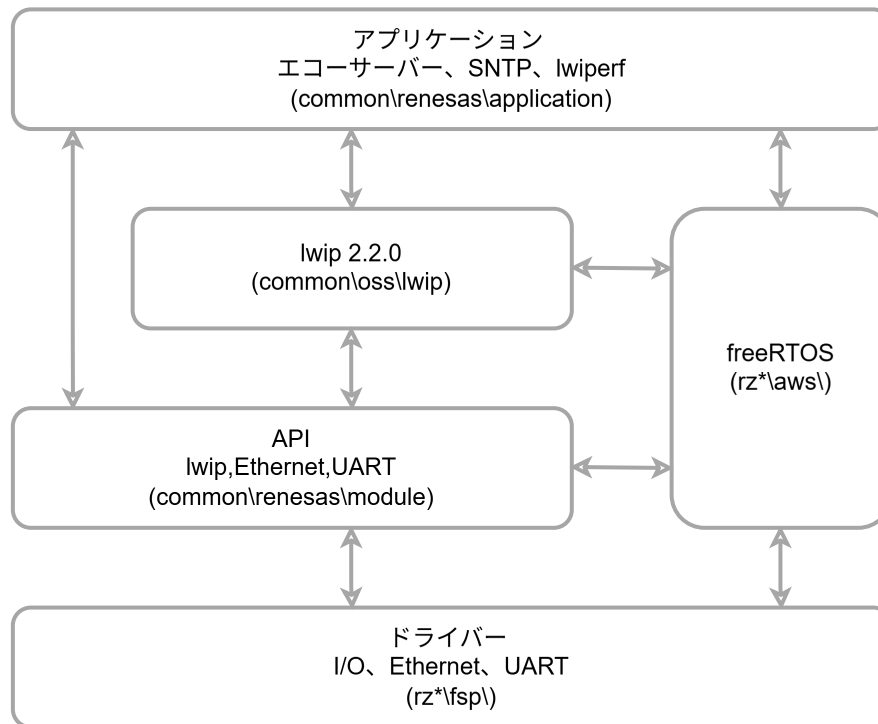


Figure 8.1. Software Architecture Diagram

8.1.1. コンポーネントの概要

概要図内の機能概要

Table 8.1. Overview of Each Component

グループ	フォルダー名	概要
アプリケーション	common/renesas/application	デモンストレーションアプリ
	SRC	エントリー関数など
API	common/renesas/module/ether_netif	Ethernet API
	common/renesas/module/lwip_port	lwip API
	common/renesas/module/serial_io	UART API
OS	Rz*/aws/	freeRTOS
ドライバ	rz*/fsp/	Ethernet ドライバ
	rz*/fsp/	UART ドライバ

8.1.2. 使用ドライバ

FSP で使用する STACKS を以下に示します。

Table 8.2. Sample Software Support Status

Stack 名	機能
g_ioport I/O Port Driver	Pin Settings, Control
g_ether0 Ethernet	Ethernet communication
g_uart0 UART	Serial communication
Heap 4	Allocating heap for FreeRTOS

8.2. タスク

サンプルプログラム内で使用するタスクプロパティを示します。

Table 8.3. Task List

モジュール	タスク名	優先度	スタックサイズ[bytes]	同期モード	概要
Main	Main thread	5	1024	初期化処理後休止	Main task
Ethernet Port	Ethernet Monitor Task	3	1024	1 秒間隔で起床	Ethernet のリンクステータス監視
	Ethernet Reader Task	4	2048	受信イベント	Ethernet の受信データ待ち
lwIP Port	iperf_client_task	4	1024	lperf 送信イベント待ち	lperf 送信イベント
lwIP	lwIP	4	2048	メッセージイベントキュー待ち	lwIP 内のメッセージ処理
application	TCP Server task	3	1024	エコーサーバー接続要求待ち	クライアントソケット作成
	TCP Server task(No.)※	3	1024	受信待ち	受信と送信

※クライアントの接続用タスク、最大 10 回線のまで接続可能で、記載の(No.)は、1 から 10 までの数字が入ります。なお、使用するタスクの仕様は、1 から 10 まで同じです。

Table 8.4. RTOS Max Priority

Item	Value
Max Priority	10

8.2.1. リソース

使用リソースを以下に示します。

Table 8.5. Used Resource

使用リソース	形式	モジュール	用途
Heap4	Heap メモリ	全モジュール	Ethernet の送受信データ
queue	Queue	Ethernet port	Link ステータス通知

8.2.2. タスク操作

以下に各タスクの関連図を示します。

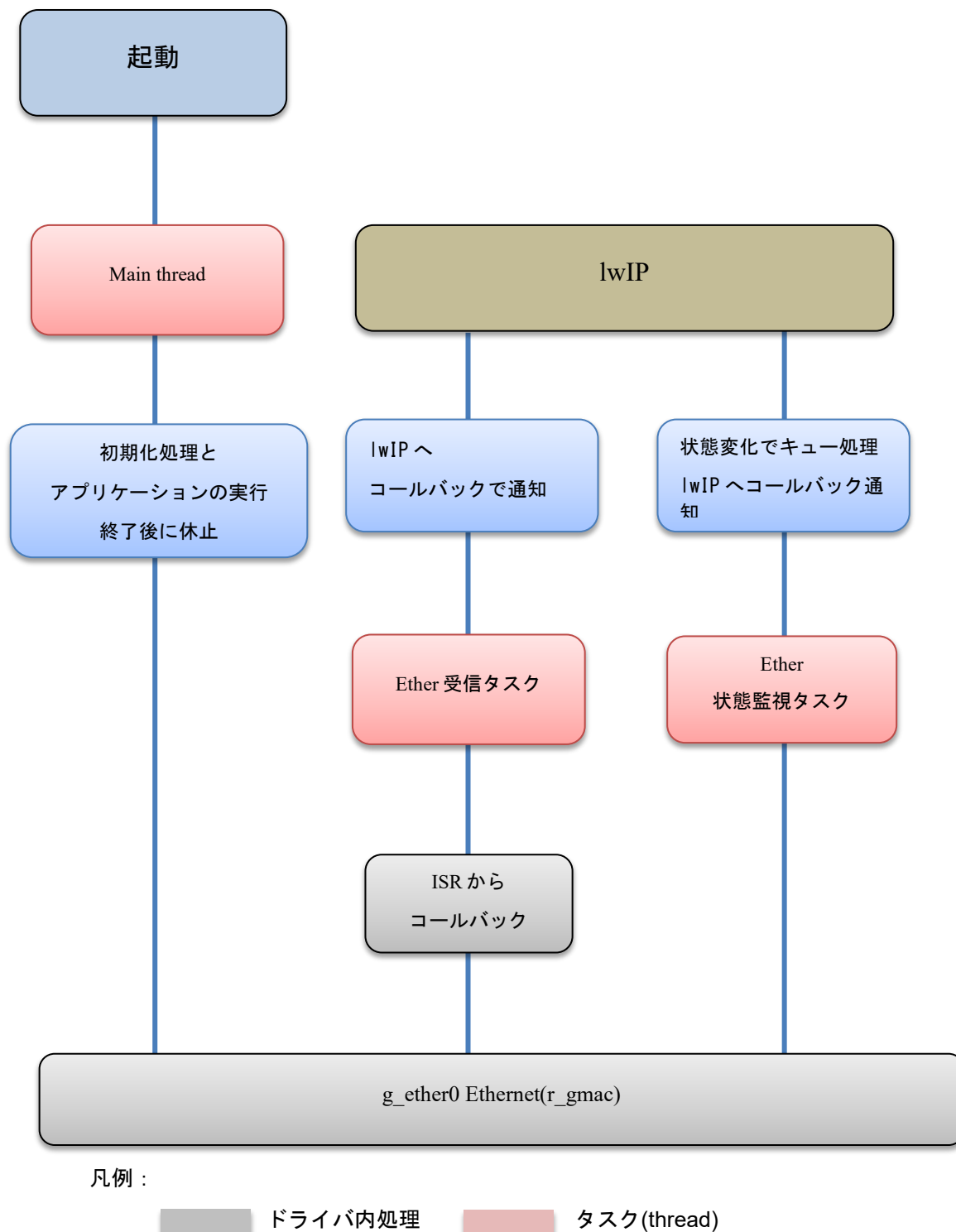


Figure 8.2. sequence

8.3. 受信パケットのミラーリングと送信機能

ドライバからの受信データを直接取得する、lwIP を通さずに直接送信する場合、以下の資料を参考にしてください。

1. 概要図

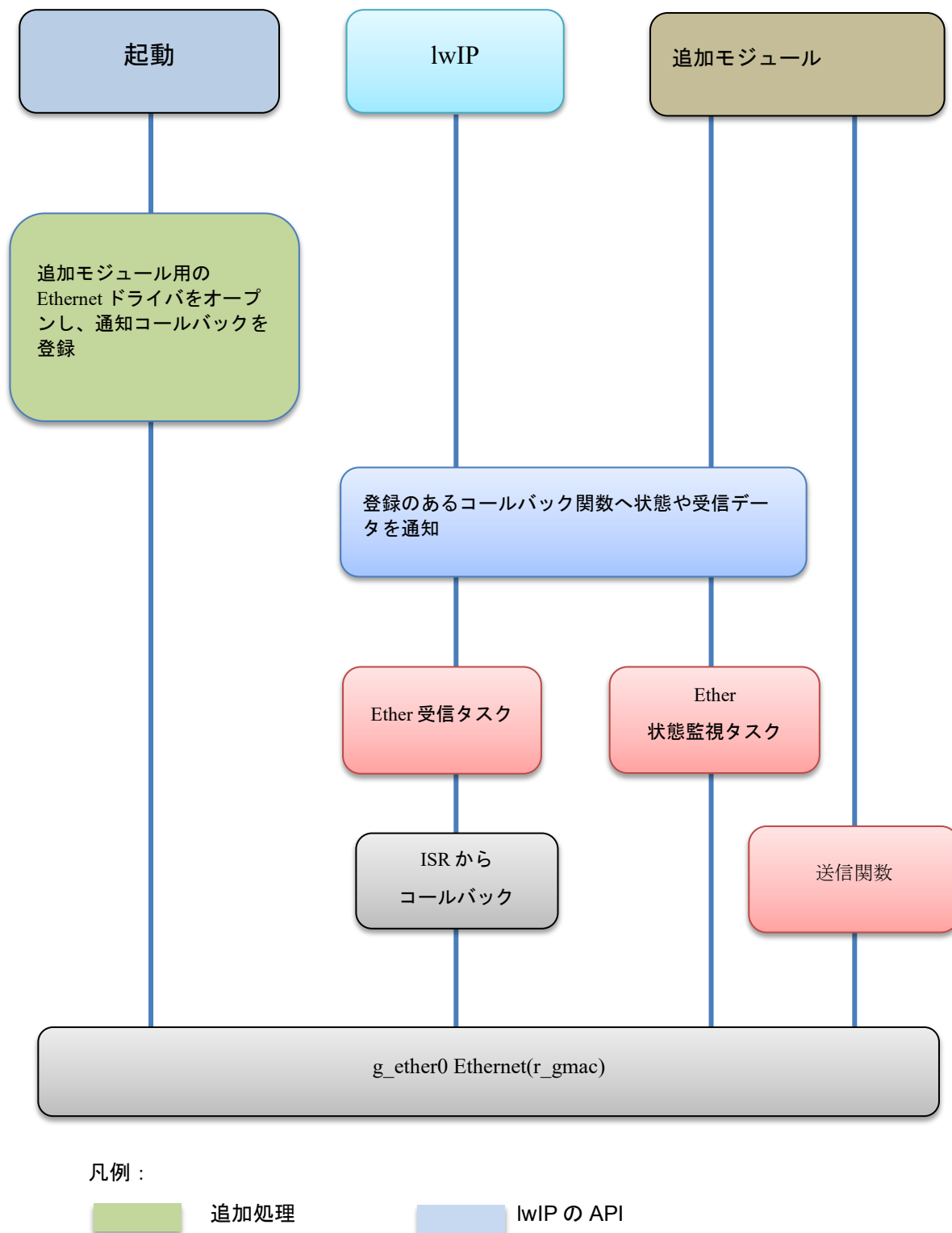


Figure 8.3. sequence

2. ミラーリング

ミラーリング機能を有効にするには、Ethernet の API でコールバック関数を登録し、LINK 状態や受信データ (MAC フレーム) を受け取ります。

- ▶ コールバック関数登録時のパラメータ
既存の処理を参考に、以下の様に作成してください。

Table 8.6. Callback function parameters

データの形	概要
struct st_ether_netif_callback_link_node * p_next;	初期値:NULL 初期値以外は書き込み禁止
void (* p_func) (ether_netif_callback_args_t *);	コールバック関数。通知を受ける関数
ether_netif_callback_args_t * p_memory;	コールバック通知時のセットされる、データ (ether_netif_callback_args_t)
request_port_t Request_port;	コールバック通知の対象ポートを指定します。複数指定する 場合は、各ポートの値をビット論理和 (OR) で組み合わせ ます。 1:ETH0,2:ETH1,4:ETH2
void const * p_context;	コールバック時のパラメータに追加したいデータ

- ▶ コールバック登録関数
usr_err = gp_ether_netif0->p_api->callbackAddstart(gp_ether_netif0->p_ctrl,
ether_netif_callback_link_node_t *);
- ▶ コールバックを受ける関数のパラメータ

Table 8.7. Callback parameters

ether_netif_callback_args_t	コールバック時にデータ設定される。
ether_netif_callback_event_t event	コールバックの種類
ether_netif_frame_t * p_frame_packet	受信データ
ether_netif_monitor_queue_t * p_port_status	イベント時のポートステータス
void const * p_context;	コールバック登録時に設定したデータ

3. 送信関数

lwIP を通さずに、MAC フレームを送信する場合、以下の様に使用してください。

- ▶ 送信関数パラメータ
static ether_netif_frame_t 新規作成で、0 クリアしてください。
. buffer[] 送信データをコピー
. length 送信レングス
. port 送信先ポート番号(1:ETH0,2:ETH1,4:ETH2)
- ▶ 送信関数
usr_err = gp_ether_netif0->p_api->send(gp_ether_netif0->p_ctrl, ether_netif_frame_t *);

8.4. マルチインターフェース

8.4.1. インターフェースのプロパティテーブル

以下に、インターフェースのプロパティテーブル(配列)の詳細を示します。

```

94  /** Initial settings data such as IP address and MAC address */
95  static lwip_port_netif_ip_cfg_t g_lwip_port_netif_cfg[LWIP_NIC_MAX] =
96  {
97      {
98  #if LWIP_NIC_MAX > 1
99          .p_next      = &g_lwip_port_netif_cfg[1],
100  #else
101          .p_next      = NULL,
102  #endif
103          .dhcp        = LWIP_PORT_DHCP_DISABLE,
104          .ip_address   = PP_HTONL(LWIP_MAKEU32(192,168, 10,170)),
105          .subnet_mask  = PP_HTONL(LWIP_MAKEU32(255,255,255, 0)),
106          .gateway_address = PP_HTONL(LWIP_MAKEU32(192,168, 10, 1)),
107          .hwaddr       = {0x00,0x11,0x22,0x33,0x44,0x55},
108          .p_host_name  = gp_lwip_port0_hostname_0,
109          .eth_port     = ETHER_PORT_0,
110      },
111  #if LWIP_NIC_MAX > 1
112      {
113          .p_next      = &g_lwip_port_netif_cfg[2],
114          .dhcp        = LWIP_PORT_DHCP_DISABLE,
115          .ip_address   = PP_HTONL(LWIP_MAKEU32(192,168, 10,171)),
116          .subnet_mask  = PP_HTONL(LWIP_MAKEU32(255,255,255, 0)),
117          .gateway_address = PP_HTONL(LWIP_MAKEU32(192,168, 10, 2)),
118          .hwaddr       = {0x00,0x11,0x22,0x33,0x44,0x77},
119          .p_host_name  = gp_lwip_port0_hostname_1,
120          .eth_port     = ETHER_PORT_1,
121      },
122  #endif
123  #if LWIP_NIC_MAX > 2
124      {
125          .p_next      = NULL,
126          .dhcp        = LWIP_PORT_DHCP_DISABLE,
127          .ip_address   = PP_HTONL(LWIP_MAKEU32(192,168, 10,172)),
128          .subnet_mask  = PP_HTONL(LWIP_MAKEU32(255,255,255, 0)),
129          .gateway_address = PP_HTONL(LWIP_MAKEU32(192,168, 10, 3)),
130          .hwaddr       = {0x00,0x11,0x22,0x33,0x44,0x88},
131          .p_host_name  = gp_lwip_port0_hostname_2,
132          .eth_port     = ETHER_PORT_2,
133      }
134  #endif
135  };
136

```

Figure 8.4. sequence

1. プロパティの詳細

以下にプロパティ項目の詳細を示します。

Table 8.8. Interface properties

変数	概要
.p_next	次のインターフェースの配列ポインタ プロパティデータの数制限する場合、このデータを「NULL」に設定してください。以降のデータは無効になります。
.dhcp	DHCP 機能の有無
.ip_address	IP アドレス
.subnet_mask	サブネットマスク
.gateway_address	デフォルトゲートウェイ (接続先ホスト PC)
.hwaddr	MAC アドレス
.p_host_name	ネットワーク名
.eth_port	使用 LAN ポート 1:ETH0, 2:ETH1, 4:ETH2 複数の場合は論理和 ここで指定した、ビットデータの LAN ポートの受信データおよび LINK 状態の コールバックが可能になります。

2. プロパティの注意事項

- ・ IP アドレスと、MAC アドレスは、他のインターフェースプロパティと重複できません。
- ・ 使用 LAN ポートの設定で、挿抜を前提に複数の LAN ポートを指定した場合、指定したポートの何れか 1 ポートのみに、LAN ケーブルを接続してください。

IP アドレスと MAC アドレスが違ふ場合は、LAN ケーブルの差し替えではなく、2つのインターフェースを設定して、ルーティングにて 1IP アドレス向けのアクセスをしてください。

8.4.2. ルーティング

マルチインターフェースに伴い、Ethernet ドライバと lwIP の間で、MAC アドレスを基に、受信フレームを対応する NETIF に紐づけ、lwIP へ渡します。

1. 概要図

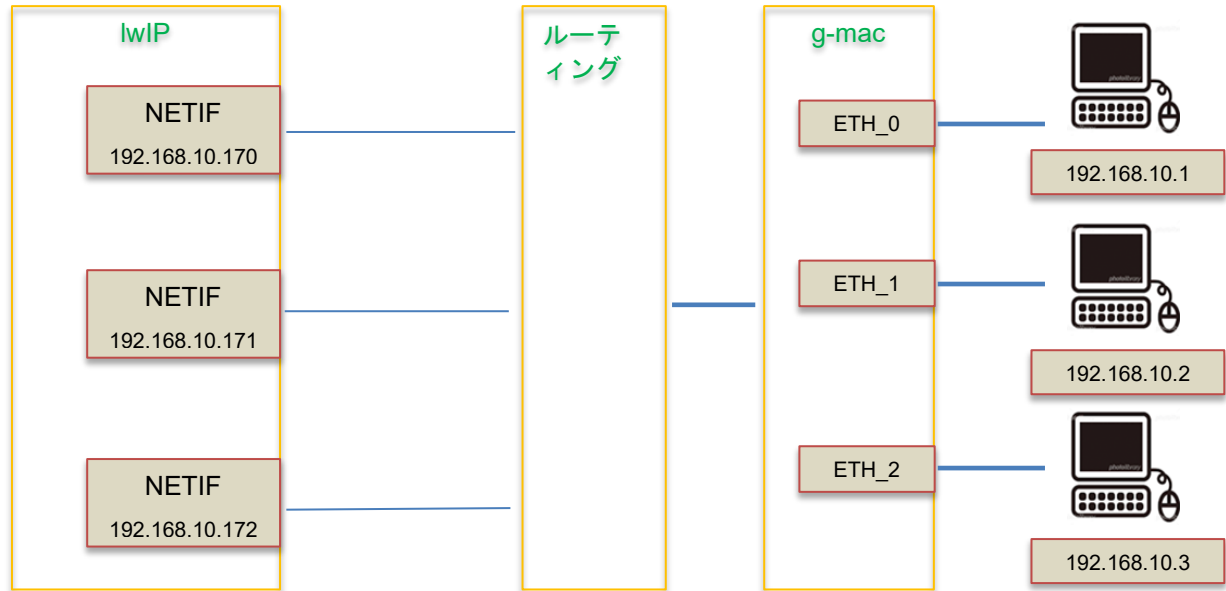


Figure 8.5. Routing Overview Diagram

2. ルーティングで通信可能な IP アドレスの組み合わせ例

Table 8.9. IP address combinations

PC 192.168.10.1 → 192.168.10.170 → 192.168.10.171 → 192.168.10.172	lwIP 192.168.10.170 → 192.168.10.1 → 192.168.10.2 → 192.168.10.3
PC 192.168.10.2 → 192.168.10.170 → 192.168.10.171 → 192.168.10.172	lwIP 192.168.10.171 → 192.168.10.1 → 192.168.10.2 → 192.168.10.3
PC 192.168.10.3 → 192.168.10.170 → 192.168.10.171 → 192.168.10.172	lwIP 192.168.10.172 → 192.168.10.1 → 192.168.10.2 → 192.168.10.3

8.4.3. Management port specific tag + 非ルーティングモード

「Management port specific tag」が「Enable」で、ルーティングが無効の場合、以下の様な通信になります。

1. 概要図

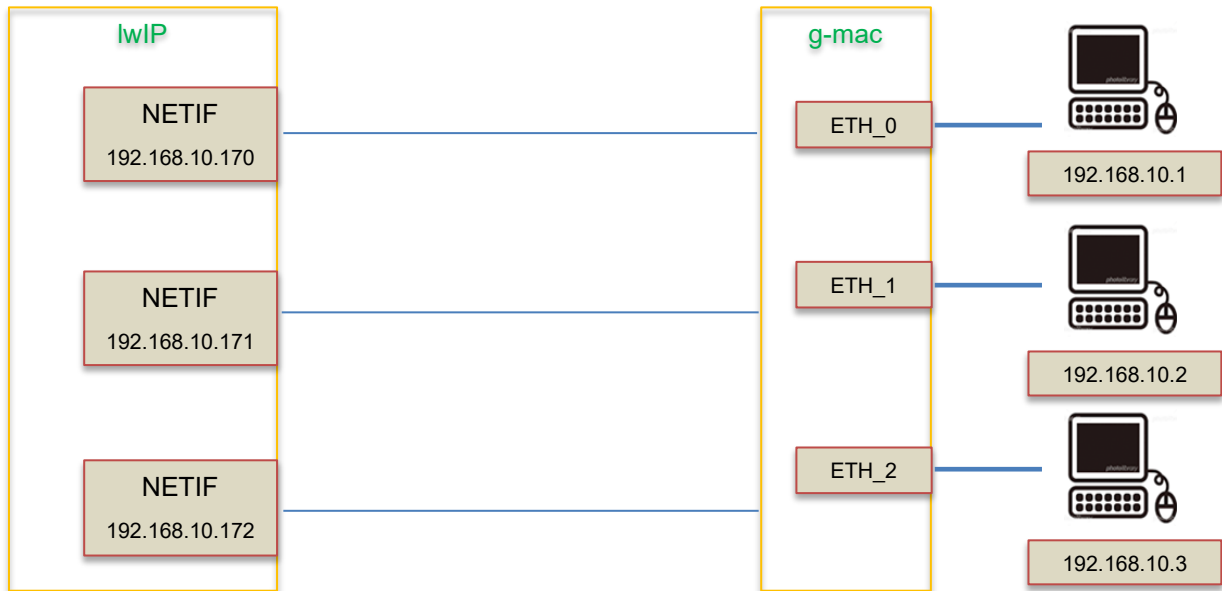


Figure 8.6. Non-routing overview diagram

2. 通信可能な IP アドレスの組み合わせ例

Table 8.10. IP address combinations

PC 192.168.10.1 → 192.168.10.170	lwIP 192.168.10.170 → 192.168.10.1
PC 192.168.10.2 → 192.168.10.171	lwIP 192.168.10.171 → 192.168.10.2
PC 192.168.10.3 → 192.168.10.172	lwIP 192.168.10.172 → 192.168.10.3

8.5. メモリ配置

プロジェクトでは、下記 3 つのグループが配置されます。

- ・ローダープログラム
起動処理や、クロック、割り込み、IO、などの初期化などの処理が、BTCM へ配置されます。
- ・ユーザープログラム
RTOS を含め、割り込みベクタ、ドライバ、アプリケーションなど、ローダープログラム以外の処理が、ATCM または、SYSTM_RAM に配置されます。
- ・共用ノンキャッシュ
Ethernet 送受信データなどの保存場所として使用され、SYSTM_RAM のミラー領域に配置されます。

8.5.1. e2 studio のプロジェクト

下記、「script」フォルダー内の定義ファイルで配置されます。

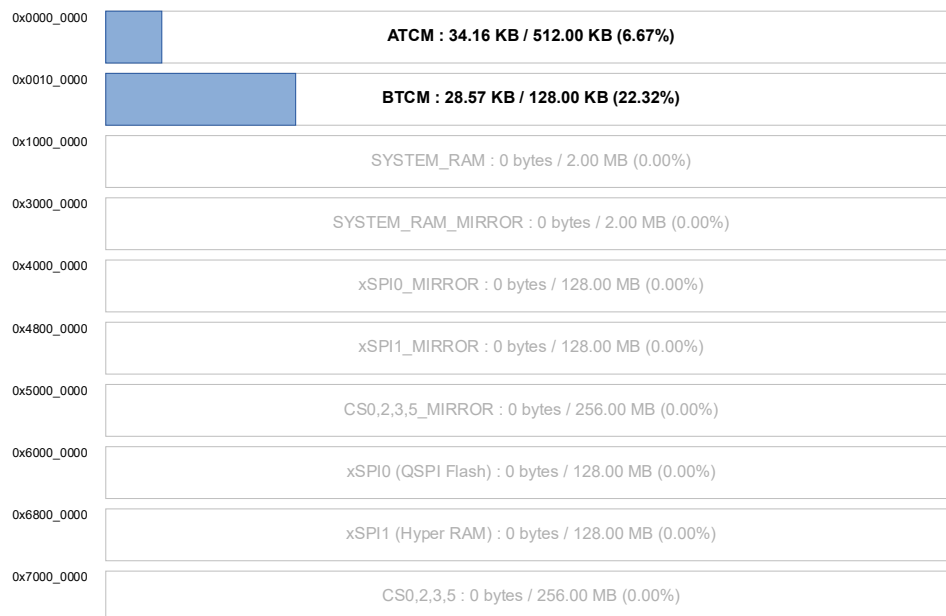
- ▶ xxxxxxxx.ld
リンクスクリプトファイルで、ブートモードなどでファイル名などが変わります。
- ▶ xxxxxxxx.h
物理メモリの定義ファイルで、実装デバイスのアドレスなどが定義されています。

1. RZ/T2M

以下にグループ別のメモリ使用量を示します。

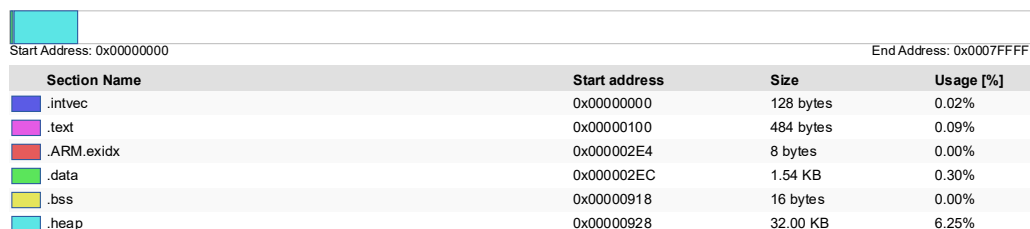
- ▶ CPU0(CR52_0)
CPU0 で動作するアプリケーションは、LED 点滅機能のみになります。

RZT2M Memory Map

Figure 8.7. Memory footprint of the CPU0 project (e² studio)

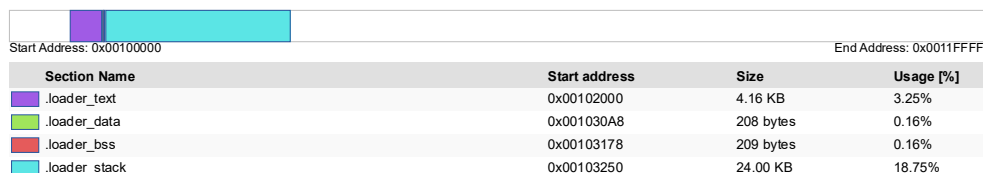
ATCM

Total Usage : 34.16 KB

Figure 8.8. ATCM footprint of the CPU0 project (e² studio)

BTCM

Total Usage : 28.57 KB

Figure 8.9. BTCM footprint of the CPU0 project (e² studio)

- CPU1(CR52_1)
CPU1 で動作するアプリケーションは、LED 点滅以外の機能になります。

RZT2M Memory Map

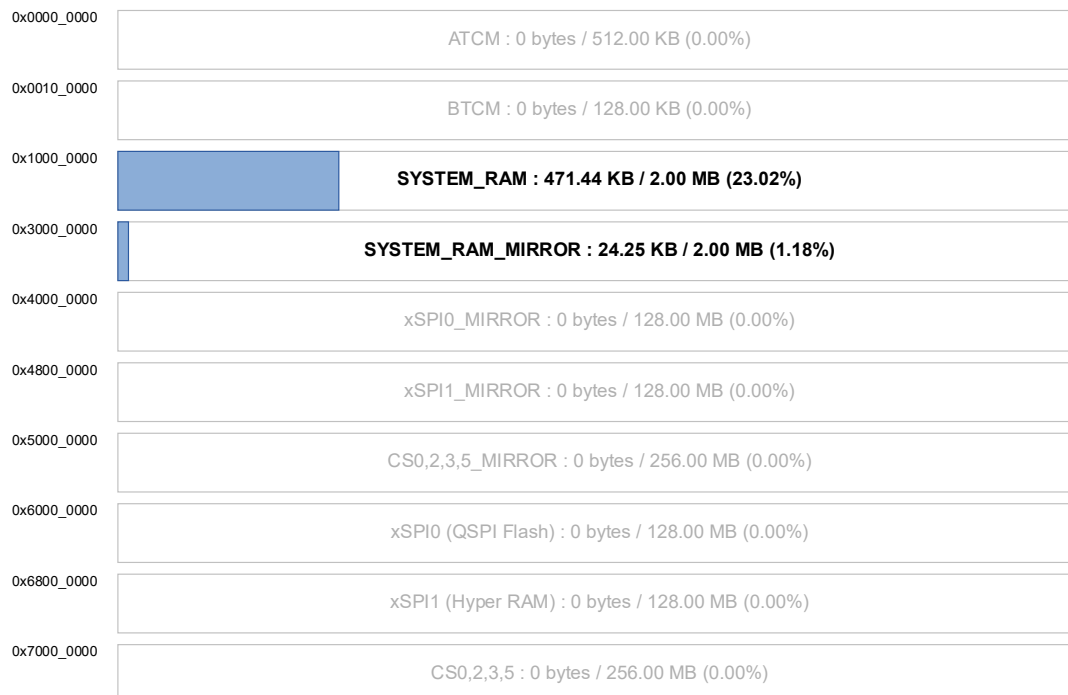


Figure 8.10. Memory footprint of the CPU1 project (e² studio)

SYSTEM_RAM

Total Usage : 471.44 KB

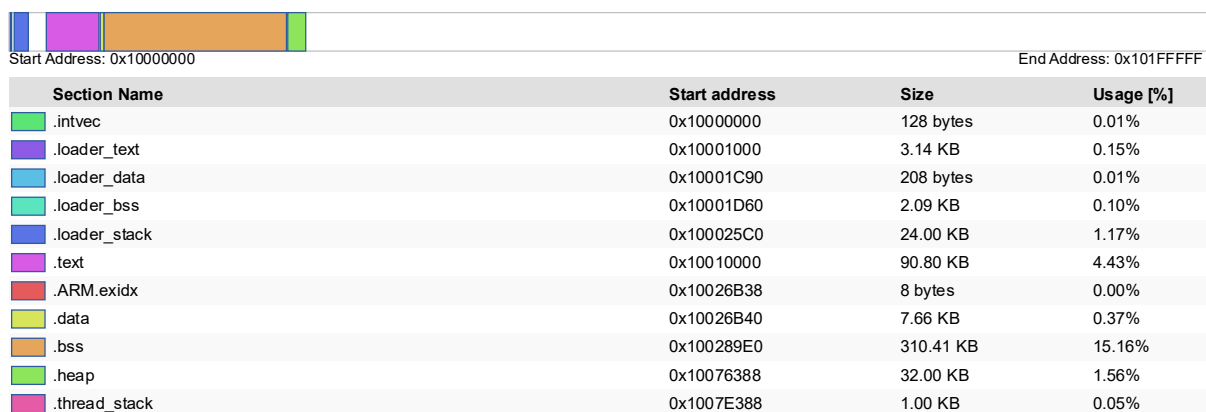


Figure 8.11. SYSTM_RAM footprint of the CPU1 project (e² studio)

SYSTEM_RAM_MIRROR

Total Usage : 24.25 KB



Figure 8.12. SYSTM_RAM (MIRROR) footprint of the CPU1 project (e² studio)

2. RZ/N2L

RZ/N2L の場合、ATCM のサイズが小さいため、ユーザープログラムは SYSTEM_RAM へ配置します。

RZN2L Memory Map

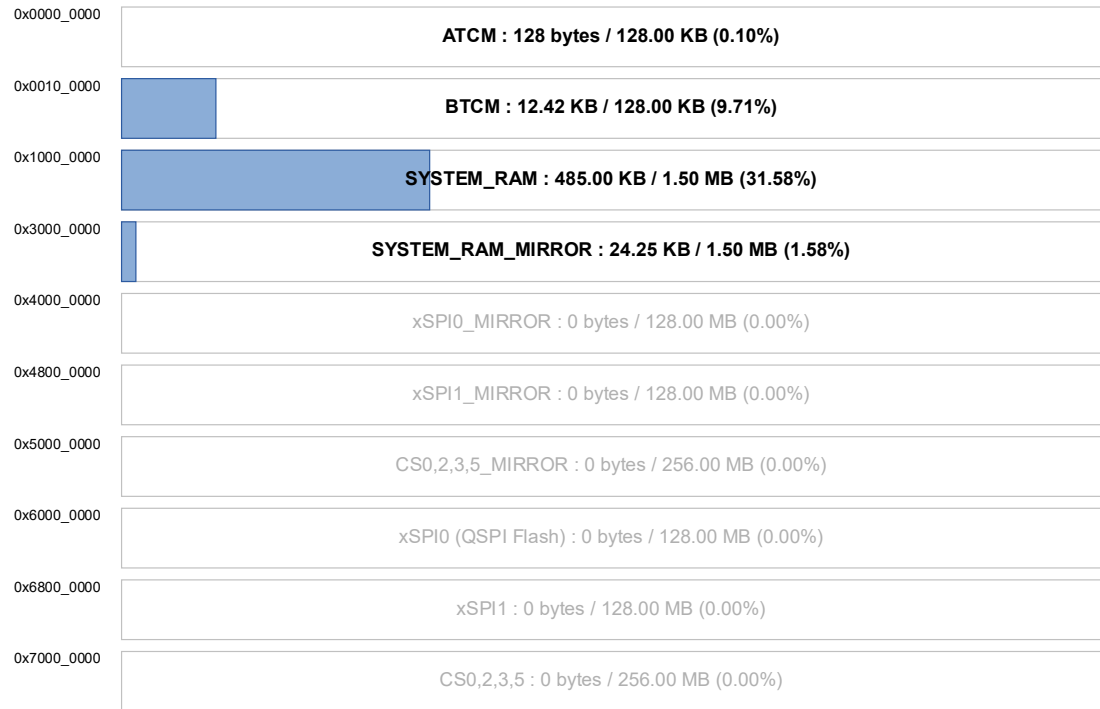


Figure 8.13. Memory footprint of the CPU0 project (e² studio)

ATCM

Total Usage : 128 bytes

Start Address: 0x00000000				End Address: 0x0001FFFF			
Section Name		Start address	Size	Usage [%]			
.intvec		0x00000000	128 bytes	0.10%			

Figure 8.14. ATCM footprint of the CPU0 project (e² studio)

BTCM

Total Usage : 12.42 KB

Start Address: 0x00100000				End Address: 0x0011FFFF			
Section Name		Start address	Size	Usage [%]			
.loader_text		0x00102000	4.13 KB	3.23%			
.loader_data		0x00103088	208 bytes	0.16%			
.loader_bss		0x00103158	2.09 KB	1.63%			
.loader_stack		0x001039B8	6.00 KB	4.69%			

Figure 8.15. BTCM footprint of the CPU0 project (e² studio)

SYSTEM_RAM

Total Usage : 485.00 KB

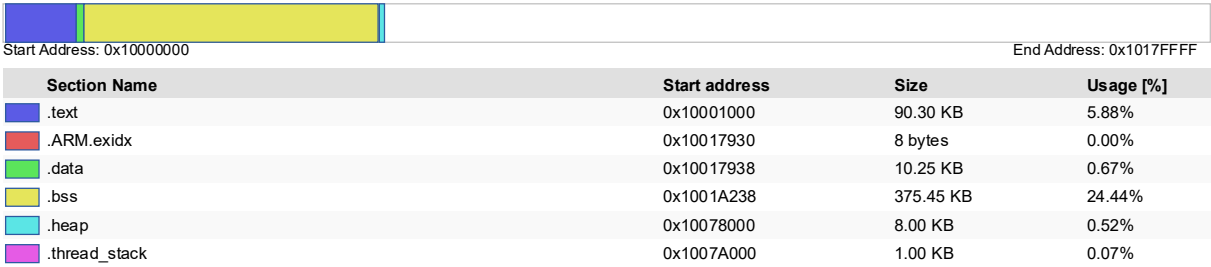


Figure 8.16. SYSTEM_RAM footprint of the CPU0 project (e² studio)

SYSTEM_RAM_MIRROR

Total Usage : 24.25 KB

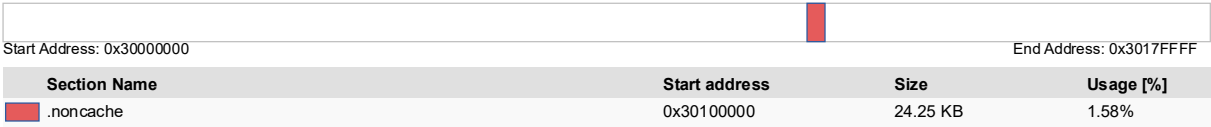


Figure 8.17. SYSTEM_RAM (MIRROR) footprint of the CPU0 project (e² studio)

3. RZ/T2L

以下にグループ別のメモリ使用量を示します。

RZT2L Memory Map

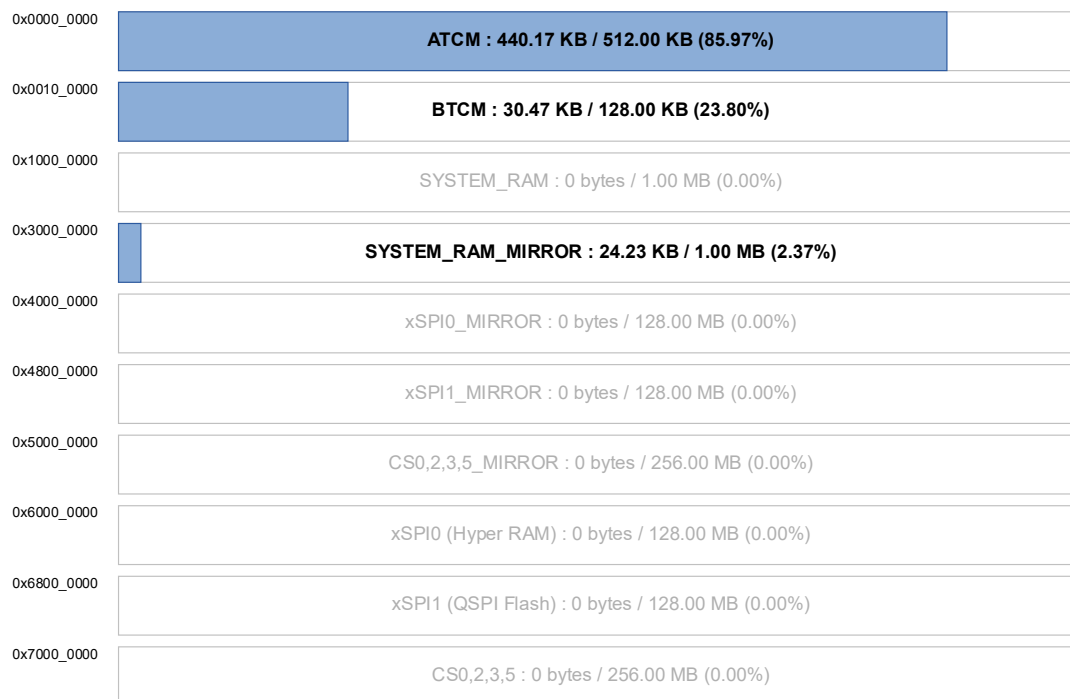


Figure 8.18. Memory footprint of the CPU0 project (e² studio)

ATCM

Total Usage : 440.17 KB

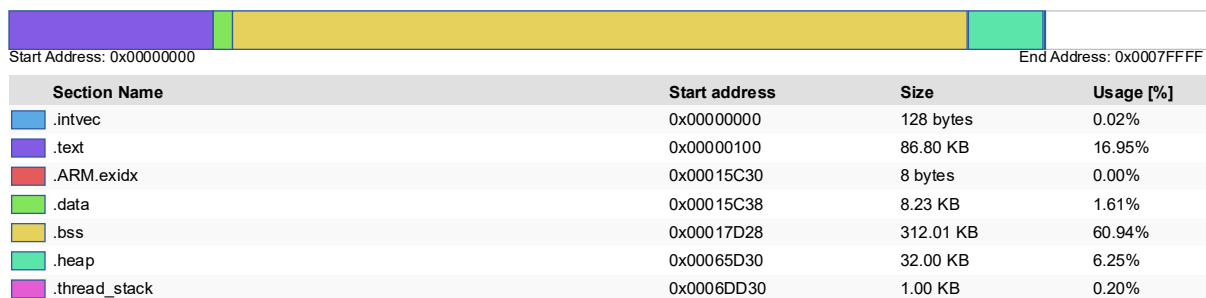


Figure 8.19. ATCM footprint of the CPU0 project (e² studio)

BTCM

Total Usage : 30.47 KB



Figure 8.20. BTCM footprint of the CPU0 project (e² studio)

SYSTEM_RAM_MIRROR

Total Usage : 24.23 KB

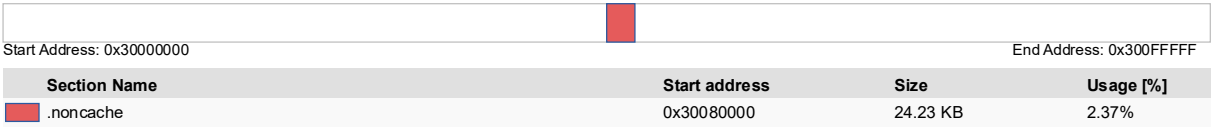


Figure 8.21. SYSTEM_RAM (MIRROR) footprint of the CPU0 project (e² studio)

8.5.2. EWARM のプロジェクト

「script」フォルダー内の下記、定義ファイルで配置されます。

- ▶ xxxxxxxx.icf
リンクスクリプトファイルで、ブートモードなどでファイル名などが変わります。
- ▶ xxxxxxxx.h
物理メモリの定義ファイルで、実装デバイスのアドレスなどが定義されています。

1. RZ/T2M

以下にグループ別のメモリ使用量を示します。

- ▶ CPU0(CR52_0)
CPU0 で動作するアプリケーションは、LED 点滅機能のみになります。

RZT2M Memory Map

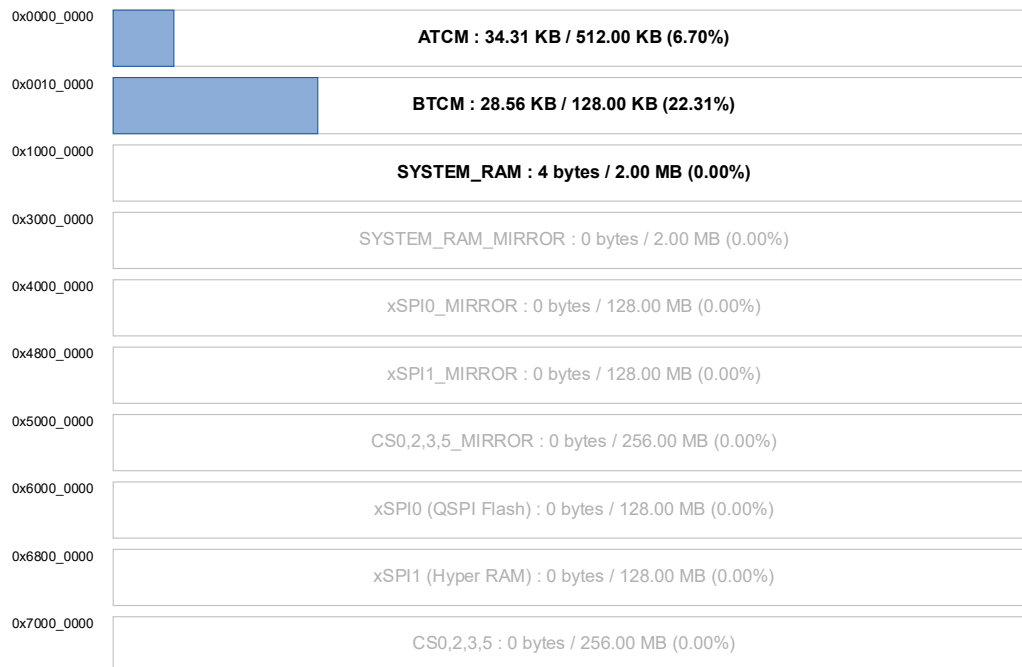


Figure 8.22. Memory footprint of the CPU0 project (EWARM)

ATCM

Total Usage : 34.31 KB

Start Address: 0x00000000				End Address: 0x0007FFFF			
Section Name	Start address	Size	Usage [%]				
VECTOR_RBLOCK	0x00000000	136 bytes	0.03%				
USER_PRG_RBLOCK	0x00000100	562 bytes	0.11%				
.rodata	0x00000334	1.47 KB	0.29%				
.version	0x00000914	56 bytes	0.01%				
.rodata	0x0000094C	76 bytes	0.01%				
.version	0x00000998	12 bytes	0.00%				
HEAP_BLOCK	0x000009A8	32.00 KB	6.25%				
USER_DATA_ZBLOCK	0x000009A8	12 bytes	0.00%				
ATCM_END_block	0x000009B4	4 bytes	0.00%				

Figure 8.23. ATCM footprint of the CPU0 project (EWARM)

SYSTEM_RAM

Total Usage : 4 bytes

Start Address: 0x10000000				End Address: 0x101FFFFF			
Section Name	Start address	Size	Usage [%]				
NONCACHE_END_block	0x10180000	4 bytes	0.00%				

Figure 8.24. BTCM footprint of the CPU0 project (EWARM)

SYSTEM_RAM

Total Usage : 4 bytes

Start Address: 0x10000000				End Address: 0x101FFFFF			
Section Name		Start address	Size	Usage [%]			
 NONCACHE_END_block		0x10180000	4 bytes	0.00%			

Figure 8.25. SYSTM_RAM footprint of the CPU0 project (EWARM)

➤ CPU1(CR52 1)

CPU1 で動作するアプリケーションは、LED 点滅以外の機能になります。

RZT2M Memory Map

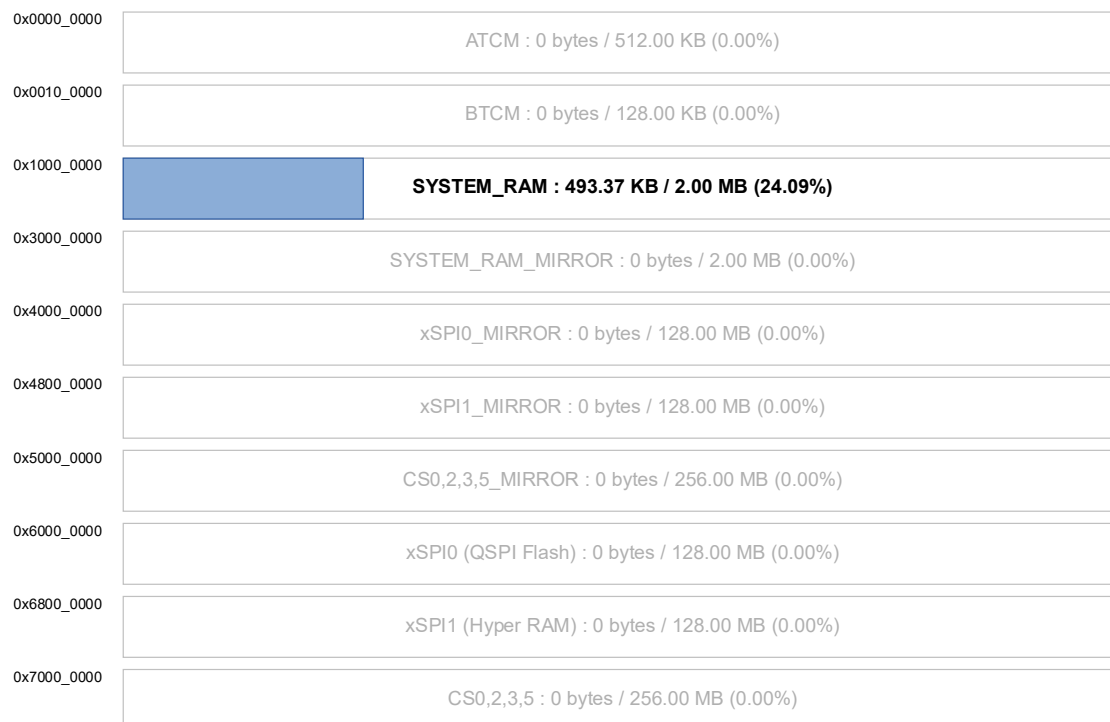


Figure 8.26. Memory footprint of the CPU1 project (EWARM)

SYSTEM RAM

Total Usage : 493.37 KB

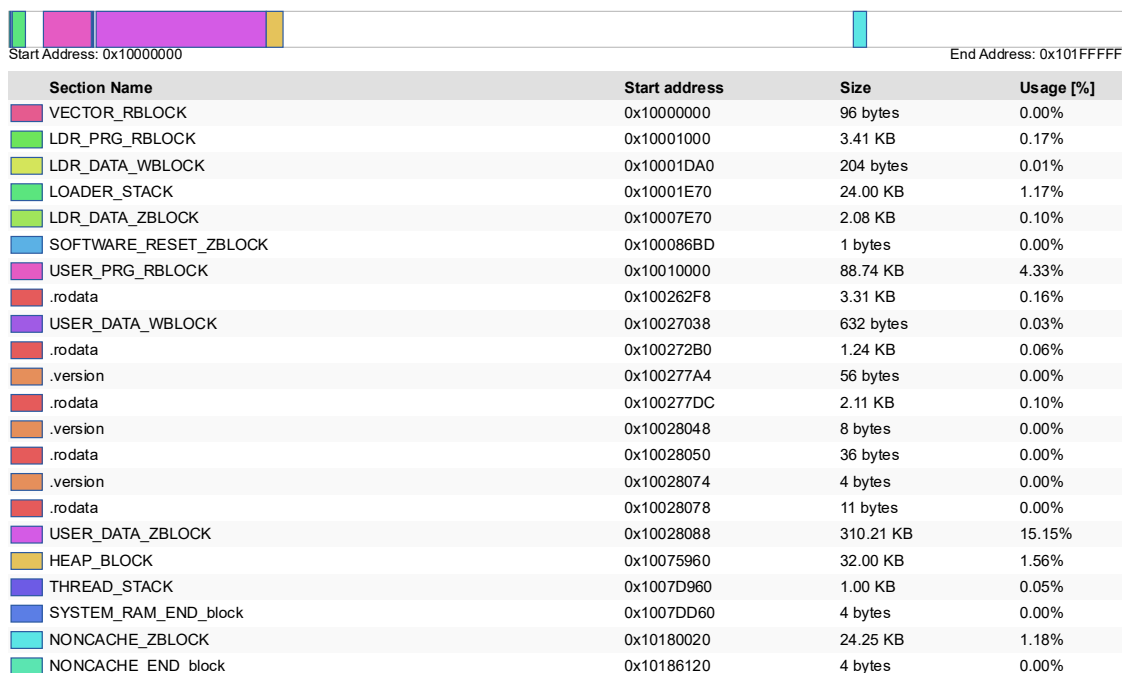


Figure 8.27. SYSTM_RAM footprint of the CPU1 project (EWARM)

2. RZ/N2L

RZ/N2L の場合、ATCM のサイズが小さいため、ユーザープログラムは SYSTM_RAM へ配置します。

RZN2L Memory Map

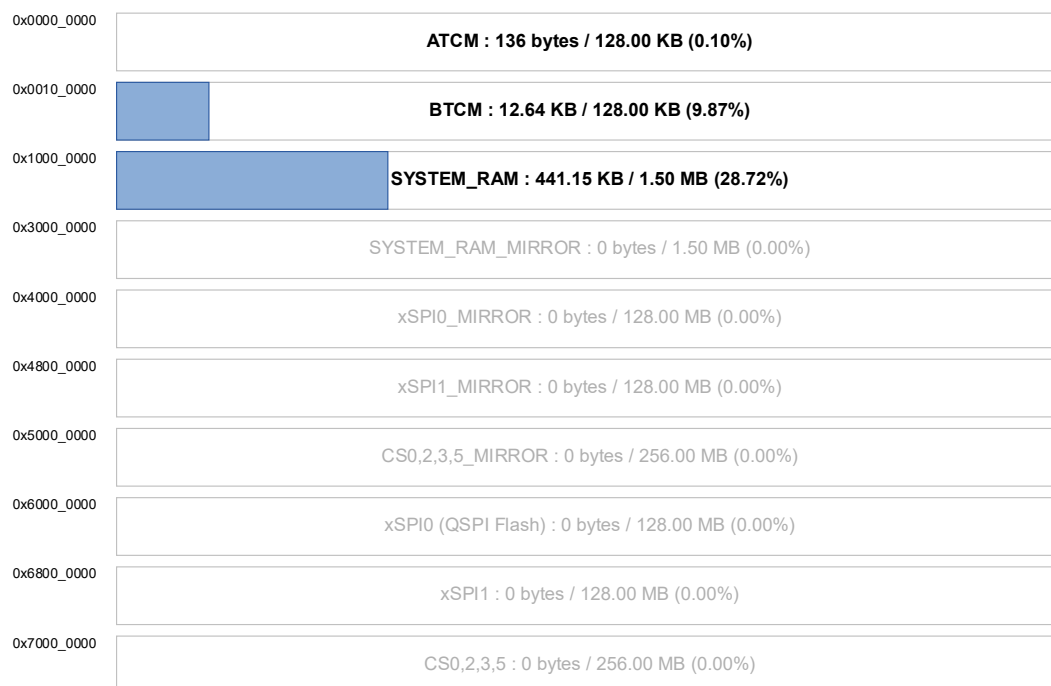


Figure 8.28. Memory footprint of the CPU0 project (EWARM)

ATCM

Total Usage : 136 bytes

Start Address: 0x00000000				End Address: 0x0001FFFF			
Section Name		Start address	Size	Usage [%]			
VECTOR_RBLOCK		0x00000000	136 bytes	0.10%			

Figure 8.29. ATCM footprint of the CPU0 project (EWARM)

BTCM

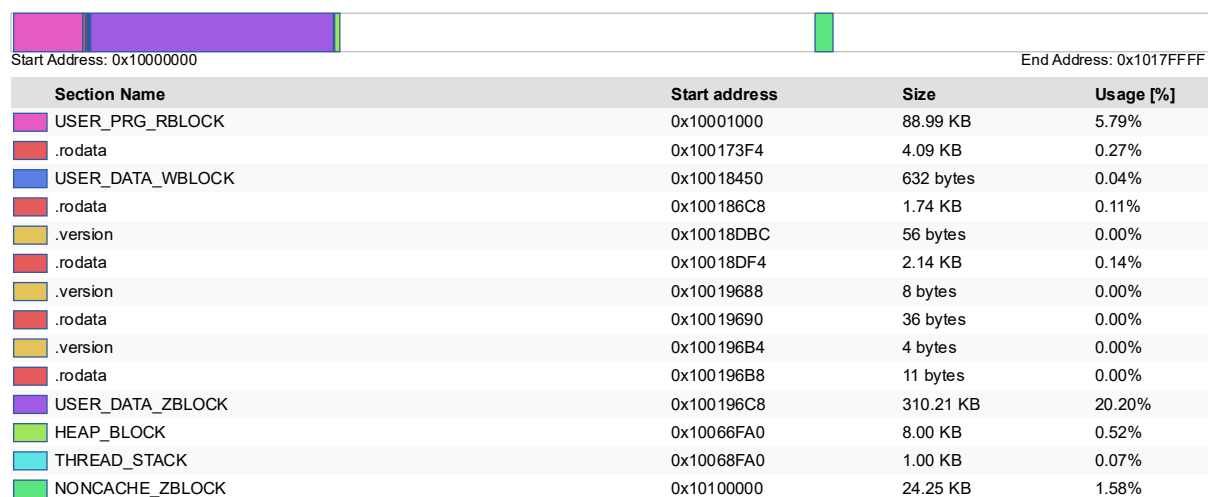
Total Usage : 12.64 KB

Start Address: 0x00100000				End Address: 0x0011FFFF			
Section Name		Start address	Size	Usage [%]			
LDR_PRG_RBLOCK		0x00102000	4.36 KB	3.41%			
LDR_DATA_WBLOCK		0x00103178	204 bytes	0.16%			
LOADER_STACK		0x00103248	6.00 KB	4.69%			
LDR_DATA_ZBLOCK		0x00104A48	2.08 KB	1.62%			
SOFTWARE_RESET_ZBLOCK		0x00105295	1 bytes	0.00%			

Figure 8.30. BTCM footprint of the CPU0 project (EWARM)

SYSTEM_RAM

Total Usage : 441.15 KB

**Figure 8.31. SYSTM_RAM footprint of the CPU0 project (EWARM)**

3. RZ/T2L

以下にグループ別のメモリ使用量を示します。

RZT2L Memory Map

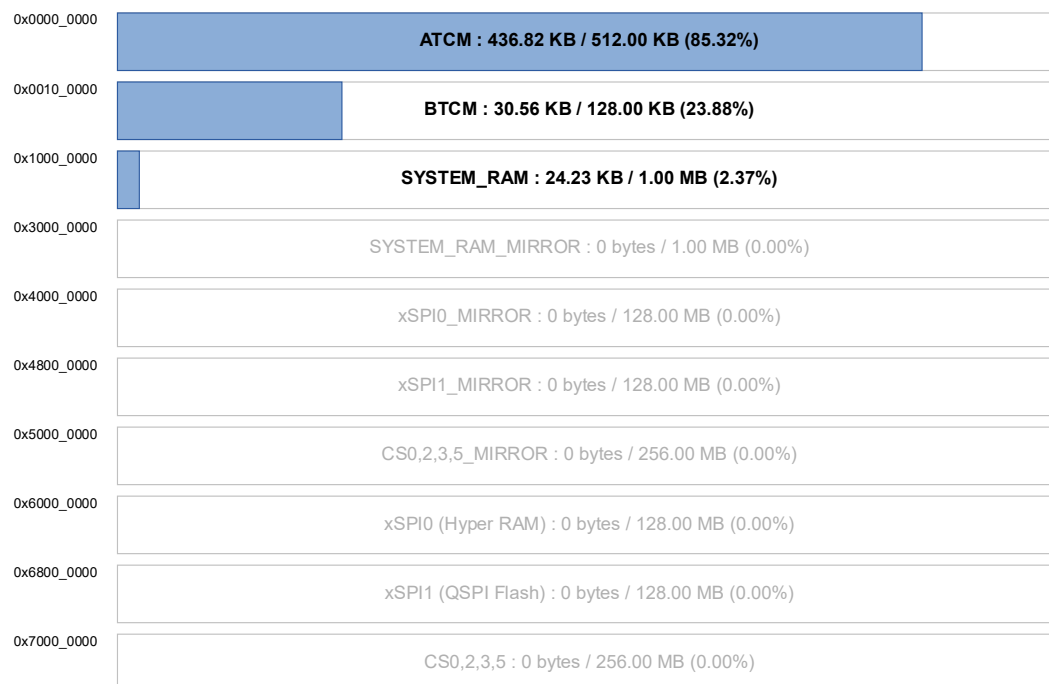


Figure 8.32. Memory footprint of the CPU0 project (EWARM)

ATCM

Total Usage : 436.82 KB

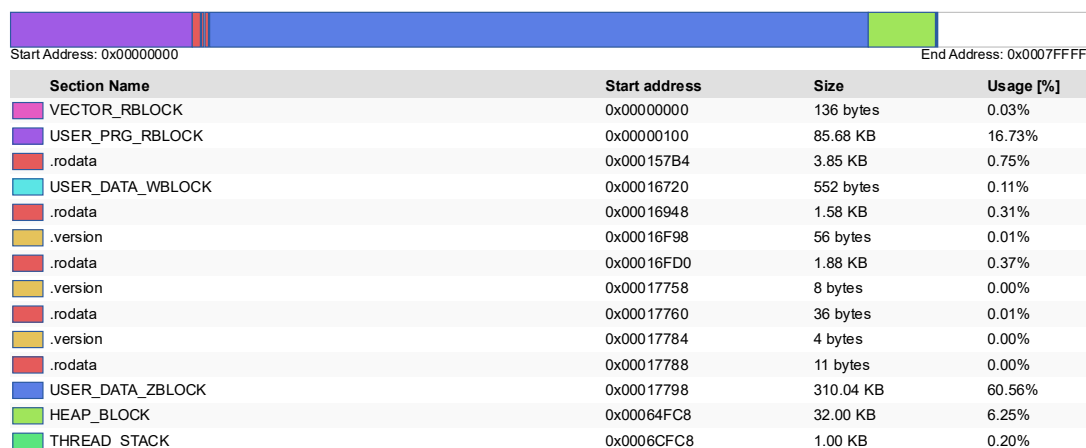


Figure 8.33. ATCM footprint of the CPU0 project (EWARM)

BTCM

Total Usage : 30.56 KB

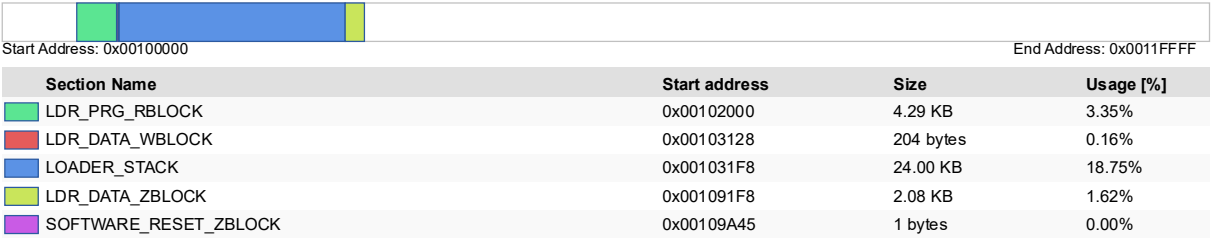


Figure 8.34. BTCM footprint of the CPU0 project (EWARM)

SYSTEM_RAM

Total Usage : 24.23 KB

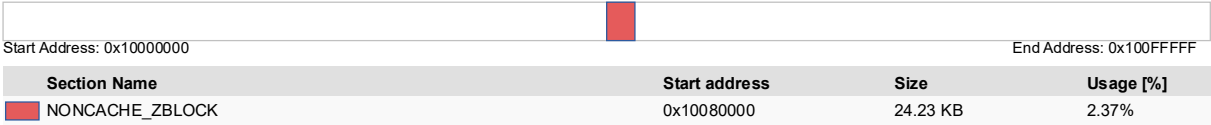


Figure 8.35. SYSTM_RAM footprint of the CPU0 project (EWARM)

8.6. CPU 使用率

評価ボードの CPU 使用率(タスク稼働率)を示します。

なお、計測は、一つの LAN ポートを e² studio で、FreeRTOS の API を使用した簡易的な算出の為、参考値になります。

また、通信モードは、「Management port specific tag Disabled」で、計測中の通信状態による待ち時間なども含まれます。

1. RZ/T2M
 - ・ iperf 通信中
CPU 使用率 69%
 - ・ エコーサーバーのクライアント待ち
CPU 使用率 18%
2. RZ/N2L
 - ・ iperf 通信中
CPU 使用率 71%
 - ・ エコーサーバーのクライアント待ち
CPU 使用率 35%
3. RZ/T2L
 - ・ iperf 通信中
CPU 使用率 62%
 - ・ エコーサーバーのクライアント待ち
CPU 使用率 26%

Revision History

Rev.	Date	Description	
		Page	Summary
1.60	Jul 31, 2026	-	フォントの統一などの改善
		-	表のフォーマット統一などの改善
		-	その他誤記などの改善
		P8,P9,P10, P12,P13,P14 P15, P31-,P36 P39, P40-P45	マルチインターフェース対応
1.50	Mar 20, 2026	-	改訂新版

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity.

Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.

3. Input of signal during power-off state

Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.

4. Handling of unused pins

Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

5. Clock signals

After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

6. Voltage application waveform at input pin

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).

7. Prohibition of access to reserved addresses

Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

8. Differences between products

Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

- Arm® and Cortex® are registered trademarks of Arm Limited (or its subsidiaries) in the EU and/or elsewhere. All rights reserved.
- Ethernet is a registered trademark of Fuji Xerox Co. Ltd.
- IEEE is a registered trademark of the Institute of Electrical and Electronics Engineers Inc
- Additionally all product names and service names in this document are a trademark or a registered trademark which belongs to the respective owners. a trademark or a registered trademark which belongs to the respective owners.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.

"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.

"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.

7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION ("Vulnerability Issues"). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:

www.renesas.com/contact/