



V. 1. 00 Release 01

2010 年 2 月 1 日

RI600/4 V.1.00 Release 01

リリースノート

目 次

1. 提供形態.....	1	8. 注意事項.....	8
2. 必要なシステム構成.....	1	8.1 RX610 グループ使用時の注意事項.....	8
3. ツールニュース.....	1	8.2 ID コードプロテクトに関する注意事項.....	8
4. ユーザ登録のお願い.....	2	8.3 0 番地に関する注意事項.....	8
5. インストール・アンインストール.....	2	9. ユーザーズマニュアルの訂正.....	9
6. カーネルソースコードのビルド方法.....	3	9.1 「5.6.5 タスクの強制終了(ter_tsk)」 エラ ーコード説明(62 ページ).....	9
7. スタックサイズについて.....	4	9.2 「表 5.34」(159 ページ).....	9
7.1 システムクロック割込みハンドラのスタック サイズ.....	4	9.3 「11.1 概要」(225 ページ).....	10
7.2 サービスコールで使用するスタックサイ ズ.....	4	10. アップデート履歴.....	11
7.3 カーネルライブラリをビルドした場合.....	7	10.1 V.1.00 Release 00 (2009 年 10 月).....	11
		10.2 V.1.00 Release 01 (2010 年 2 月).....	12
		10.2.1 カーネル.....	12



警告

本製品の動作が原因でご使用機器が人身事故、火災事故、その他の拡大損害を生じないようにフェールセーフ等のシステム上の対策を講じて頂きますようお願い致します。この対策が不可能な場合、本製品を使用しないでください。

Microsoft, Windows, および Windows Vista は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

IBM PC は、米国 International Business Machines Corporation の登録商標です。

その他すべての会社名および製品名は、各社の登録商標または商標です。

1. 提供形態

RI600/4は、下表に示す形態で提供されています。

項番	型名	ホスト環境	媒体	数量
1	R0R5RX00TRW011 *1 (評価契約, 1 台の PC にインストール可能)	Windows(R)	CD-R	1 枚
2	R0R5RX00TRW015 *1 (評価契約, 5 台の PC にインストール可能)	Windows(R)	CD-R	1 枚
3	R0R5RX00TRW01A *1 (評価契約, 10 台の PC にインストール可能)	Windows(R)	CD-R	1 枚
4	R0R5RX00TRW01K *1 (量産契約, 量産数は 1000 台まで)	Windows(R)	CD-R	1 枚
5	R0R5RX00TRW01U *1 (量産契約, 量産数は無制限)	Windows(R)	CD-R	1 枚
6	R0R5RX00TRW01Z (量産契約, 量産数は無制限、カーネルソースコード付き)	Windows(R)	CD-R	1 枚

【注】 *1 これらの提供内容は同一です。

2. 必要なシステム構成

- ホストコンピュータ : Windows(R) 2000, Windows(R) XP, または Windows Vista(R) が動作する IBM PC 互換機
- メモリ容量 : 最低 128MB、256MB 以上を推奨
- I/O 装置 : CD-ROM ドライブ
- その他 : マウス等のポインティングデバイス
- C/C++ Compiler Package for RX family

3. ツールニュース

ツールニュースは、ツール製品を購入いただいたお客様によりよくご使用いただくために、製品の情報や各種案内を提供することを目的として発行しています。

ツールニュースで新製品の情報、バージョンアップの情報、使用上のご注意等、最新の情報を入手いただき、開発にお役立てください。

特に、本バージョンリリース後の情報は本リリースノートには掲載されないため、必ず最新のツールニュースを確認してください。ツールニュースは以下の 2 種類の形で提供しています。

(1) 電子メール版ツールニュース

製品をご購入いただき、なおかつユーザ登録いただいたお客様に配信しています。(月 2 回発行)

(2) Web 版ツールニュース

すべてのお客様に閲覧いただけます。(月 2 回更新、5 日と 20 日頃)

電子メール版ツールニュースを発行した約 3 日～1 週間後に閲覧可能となります。

URL : <http://tool-support.renesas.com/jpn/toolnews/index.htm>

4. ユーザ登録のお願い

バージョンアップ情報や技術サポートなどのサービスを受けるためにユーザ登録を行って下さい。ユーザ登録をされていない場合は、これらのサービスを受けることができません。また、ご購入後 30 日以内に登録して下さいようお願い申し上げます。

- ユーザ登録方法
 1. インストールすると以下のファイルが生成されます。
C:\Program Files\Renesas\ri600-4\v100r01にインストールした場合、C:\Program Files\Renesas\ri600-4\v100r01\support\ri600-4\regsis.txt
 2. regsis.txt のファイル内容にシリアル番号を記入の上、すべてカット& ペーストして以下の電子メールアドレス宛に送付して下さい。

【ユーザ登録窓口】

regist_tool@renesas.com

※(株)ルネサス テクノロジーの個人情報保護方針につきましては、ルネサステクノロジーのホームページ「個人情報保護について」をご覧ください。

ホームページ : <http://japan.renesas.com/fmwk.jsp?cnt=privacy.htm&fp=/privacy/&site=i>

ユーザ登録でご提供頂きました個人情報は、お客様のサポート活動に活用させて頂き、そのために必要な範囲で(株)ルネサス テクノロジー、およびその関係会社、ならびに特約店に、電子データ、書面により提供させて頂きますので、ご了承の程お願い申し上げます。なお、提供を希望されない場合は、提供を停止させて頂きますので、お問い合わせ時にその旨ご連絡ください。その場合、サポート範囲が制約される場合がございます。

5. インストール・アンインストール

インストール・アンインストールには、ご使用のコンピュータの管理者権限が必要です。

提供する CD-R のルートディレクトリにある setup.exe を起動し、画面に表示されるインストールの指示に従ってください。インストール時は、あらかじめ他のアプリケーションを終了させておいてください。

setup.exe により、以下のコンポーネントがインストールされます。

- カーネルライブラリファイル、標準ヘッダファイル、システム定義ファイル
- コマンドラインコンフィギュレータ(cfg600)
- テーブル生成ユーティリティ(mkritbl)
- GUI コンフィギュレータ
- カーネルソースコード(R0R5RX00TRW01Z にのみ付属)

アンインストールするには、コントロールパネルの[プログラムの追加と削除]から、「RI600/4 V.1.00 Release 01」を選択してください。

6. カーネルソースコードのビルド方法¹

カーネルのソースコードは、”<インストールディレクトリ>%src600”に格納されます。カーネルをビルドするためには、ソースコードインストールディレクトリに移動し、”nmake”²を実行します。

カーネルビルド時には、コンパイラが必要とする環境変数の設定が必要です。

ビルド例：

```
C:\RENESAS\src600>nmake
```

カーネルライブラリのビルド後、カーネルライブラリは以下のディレクトリに生成されます。

カーネルライブラリ名	内容
product\big\debug\ri600big.lib	デバッグ情報付きビッグエンディアンライブラリ
product\big\release\ri600big.lib	デバッグ情報なしビッグエンディアンライブラリ
product\little\debug\ri600lit.lib	デバッグ情報付きリトルエンディアンライブラリ
product\little\release\ri600lit.lib	デバッグ情報なしリトルエンディアンライブラリ

【注意】

製品インストールディレクトリに対する書き込み権限がない場合、”src600”ディレクトリを書き込み可能なディレクトリにコピーしてビルドしてください。ビルド後、製品インストールディレクトリに対する書き込み権限が書き込み権限のあるユーザにて、生成されたライブラリをインストールディレクトリ下の lib600 ディレクトリにコピーしてください。

¹ ソースコードが付属するのは、ソース付き量産契約版 (R0R5RX00TRW01Z) のみです。

² “nmake.exe” は、米国 Microsoft Corporation により提供されるプロジェクトをビルドするためのツールです。“nmake.exe” は、Microsoft Visual Studio 2008 等に含まれています。

7. スタックサイズについて

7.1 システムクロック割込みハンドラのスタックサイズ

マニュアル 12.4 節に記載の ϵ_1 , ϵ_2 および ϵ_3 の値は、以下の通りです。

$\epsilon_1 = 104$

$\epsilon_2 = 104$

$\epsilon_3 = 196$

7.2 サービスコールで使用するスタックサイズ

サービスコールでは、以下のようにスタックを使用します。

- (1) タスクコンテキストから呼び出された場合
タスクコンテキスト実行中のスタックはユーザスタックです。サービスコールでは、
 - (a) ユーザスタック (呼び出し元スタック)
 - (b) システムスタックを使用します。
- (2) 非タスクコンテキストから呼び出された場合
非タスクコンテキスト実行中のスタックはシステムスタックです。サービスコールでは、
 - (c) システムスタック (呼び出し元スタック)を使用します。

サービスコールが使用する呼び出し元のスタック ((a), (c)) の使用サイズは、Call Walker によって表示されます。ただし、アセンブリ言語のアプリケーションからサービスコールを呼び出した場合は、(a), (c) の値は、Call Walker に表示されないため、手動で加算する必要があります。

また、(b) および (c) のサイズは、マニュアル 12.4 記載のようにシステムスタックサイズを算出するために必要となります。(12.4 節では α と記載しています) 以下に、各サービスコールの (a) ~ (c) のサイズを示します。

	サービスコール	ユーザスタック 使用サイズ (a)	システムスタック 使用サイズ (b), (c)	備考
タスク管理機能				
1	act_tsk	0	24	
2	iact_tsk	0	24	
3	can_act	0	24	
4	ican_act	0	24	
5	sta_tsk	0	24	
6	ista_tsk	0	24	
7	ext_tsk	0	64	タスク開始関数からのリターン時にも ext_tsk が呼び出されます。
8	ter_tsk	0	132	
9	chg_pri	0	36	
10	ichg_pri	0	52	
11	get_pri	0	24	
12	iget_pri	0	24	
13	ref_tsk	0	28	
14	iref_tsk	0	28	
15	ref_tst	0	24	
16	iref_tst	0	24	
タスク付属同期機能				
17	slp_tsk	0	24	
18	tslp_tsk	0	24	
19	wup_tsk	0	40	
20	iwup_tsk	0	52	
21	can_wup	0	24	
22	ican_wup	0	24	

	サービスコール	ユーザスタック 使用サイズ (a)	システムスタック 使用サイズ (b), (c)	備考
23	rel_wai	0	116	
24	irel_wai	0	132	
25	sus_tsk	0	24	
26	isus_tsk	0	24	
27	rsm_tsk	0	24	
28	irms_tsk	0	24	
29	frsm_tsk	0	24	
30	ifrm_tsk	0	24	
31	dly_tsk	0	24	
セマフォ				
32	sig_sem	0	44	
33	isig_sem	0	60	
34	wai_sem	0	32	
35	pol_sem	0	24	
36	ipol_sem	0	24	
37	twai_sem	0	36	
38	ref_sem	0	24	
39	iref_sem	0	24	
イベントフラグ				
40	set_flg	0	48	
41	iset_flg	0	64	
42	clr_flg	0	24	
43	iclr_flg	0	24	
44	wai_flg	0	44	
45	pol_flg	0	24	
46	ipol_flg	0	24	
47	twai_flg	0	48	
48	ref_flg	0	24	
49	iref_flg	0	24	
データキュー				
50	snd_dtq	0	36	
51	psnd_dtq	0	32	
52	ipsnd_dtq	0	52	
53	tsnd_dtq	0	40	
54	fsnd_dtq	0	32	
55	ifsnd_dtq	0	52	
56	rcv_dtq	0	32	
57	prcv_dtq	0	32	
58	iprcv_dtq	0	52	
59	trcv_dtq	0	32	
60	ref_dtq	0	24	
61	iref_dtq	0	24	
メールボックス				
62	snd_mbx	0	40	
63	isnd_mbx	0	56	
64	rcv_mbx	0	32	
65	prcv_mbx	0	28	
66	iprcv_mbx	0	28	
67	trcv_mbx	0	36	
68	ref_mbx	0	24	
69	iref_mbx	0	24	
ミューテックス				
70	loc_mtx	0	40	
71	ploc_mtx	0	24	
72	tloc_mtx	0	44	
73	unl_mtx	0	56	

	サービスコール	ユーザスタック 使用サイズ (a)	システムスタック 使用サイズ (b), (c)	備考
74	ref_mtx	0	24	
メッセージバッファ				
75	snd_mbf	0	44	
76	psnd_mbf	0	44	
77	ipsnd_mbf	0	60	
78	tsnd_mbf	0	44	
79	rcv_mbf	0	56	
80	prcv_mbf	0	56	
81	trcv_mbf	0	56	
82	ref_mbf	0	24	
83	iref_mbf	0	24	
固定長メモリプール				
84	get_mpf	0	48	
85	pget_mpf	0	36	
86	ipget_mpf	0	36	
87	tget_mpf	0	48	
88	rel_mpf	20	36	
89	irel_mpf	0	52	
90	ref_mpf	0	24	
91	iref_mpf	0	24	
可変長メモリプール				
92	get_mpl	36	96	
93	pget_mpl	0	112	
94	ipget_mpl	0	112	
95	tget_mpl	36	96	
96	rel_mpl	0	108	
97	ref_mpl	0	24	
98	iref_mpl	0	24	
時間管理機能				
99	set_tim	0	24	
100	iset_tim	0	24	
101	get_tim	0	24	
102	iget_tim	0	24	
周期ハンドラ				
103	sta_cyc	0	24	
104	ista_cyc	0	24	
105	stp_cyc	0	24	
106	istp_cyc	0	24	
107	ref_cyc	0	24	
108	iref_cyc	0	24	
アラームハンドラ				
109	sta_alm	0	24	
110	ista_alm	0	24	
111	stp_alm	0	24	
112	istp_alm	0	24	
113	ref_alm	0	24	
114	iref_alm	0	24	
システム状態管理機能				
115	rot_rdq	0	24	
116	irotd_rdq	0	24	
117	get_tid	0	24	
118	iget_tid	0	24	
119	loc_cpu	0	16	
120	iloc_cpu	0	16	
121	unl_cpu	0	24	
122	iunl_cpu	0	24	

	サービスコール	ユーザスタック 使用サイズ (a)	システムスタック 使用サイズ (b), (c)	備考
123	dis_dsp	0	16	
124	ena_dsp	0	24	
125	sns_ctx	0	24	
126	sns_loc	0	24	
127	sns_dsp	0	24	
128	sns_dpn	0	24	
129	vsta_knl	0	52	システムスタックポインタを初期化後に使用します。
130	ivsta_knl	0	52	
131	vsys_dwn	0	16	
132	ivsys_dwn	0	16	
割込み管理機能				
133	chg_ims	0	28	
134	ichg_ims	0	16	
135	get_ims	4	0	アセンブリ言語のプログラムから呼び出した場合も、Call Walker でスタック使用サイズが表示されます。
136	iget_ims	0	4	
137	ret_int	0	32	
システム構成管理機能				
138	ref_ver	0	24	
139	iref_ver	0	24	
オブジェクトリセット機能				
140	vrst_dtq	0	48	
141	vrst_mbx	0	24	
142	vrst_mbf	0	48	
143	vrst_mpf	0	48	
144	vrst_mpl	0	68	

7.3 カーネルライブラリをビルドした場合

コンパイラのバージョンやオプション設定を変更してカーネルライブラリをビルドした場合、スタックサイズが変わる場合があるので、注意してください。

8. 注意事項

8.1 RX610 グループ使用時の注意事項

RX610 グループでは、プロセッサ割り込み優先レベルは 0～7 に限定されます。GUI コンフィギュレータ、cfg600、および chg_ims・ichg_ims サービスコールでは、プロセッサ割り込み優先レベルとして 0～15 まで使用可能ですが、RX610 使用時に 8～15 の値を設定してもエラーとして検出しません。ユーザ側で 8～15 の値を設定していないことを十分に確認するようにしてください。

8.2 IDコードプロテクトに関する注意事項

RI600/4 では、固定ベクタを使用するハンドラが定義されていない場合、デフォルトのハンドラアドレスがセットされるようになっています。ID コード領域も同様にデフォルトのハンドラアドレスがセットされます。ID コード領域をユーザ任意の値に設定するには、以下のように対応してください。

1. 任意のアセンブリ言語のシンボルを固定ベクタに対する割り込みハンドラとして定義します。
2. 定義したシンボルに対して任意の値を設定します。

以下に例を示します。

[例] 0xFFFFFA0 番地に 0xFFFFFFFF を設定する場合。

- コンフィギュレーションファイルの記述
interrupt_fvector[8]{
 entry_address = ID_CODE1;
};
- リンクオプションの指定
-define=ID_CODE1=0xFFFFFFFF

8.3 0 番地に関する注意事項

以下を 0 番地に配置しないでください。

1. 可変長メモリプールのセクション
2. 固定長メモリプールのセクション
3. メールボックスに送信するメッセージのアドレス

9. ユーザーズマニュアルの訂正

以下のとおり、ユーザーズマニュアル(ドキュメント番号：RJJ10J2562-0100)を訂正します。

9.1 「5.6.5 タスクの強制終了(ter_tsk)」 エラーコード説明(62 ページ)

【正】

E_OBJ オブジェクト状態エラー(tskid のタスクが休止状態)

【誤】

E_OBJ オブジェクト状態エラー(tskid のタスクが休止状態でない)

9.2 「表 5.34」 (159 ページ)

【正】

表 5.34 マクロと定数

分類	マクロ	定義内容	定義場所	説明
...
カーネル構成	VTMAX_AREASIZE	0x10000000	kernel.h	各種領域サイズの最大値
...
エラーコード	E_NOSPT	-9	itron.h	未サポート機能
...

【誤】

表 5.34 マクロと定数

分類	マクロ	定義内容	定義場所	説明
...
カーネル構成	VTMAX_AREASIZE	0x20000000	kernel.h	各種領域サイズの最大値
...
エラーコード	E_NOSPT	-11	itron.h	未サポート機能
...

9.3 「11.1 概要」(225 ページ)

【正】

表 11.1 サンプルプログラムの関数一覧

関数名	種類	ID 番号	優先度	機能
task1()	タスク	1	1	“task1 running”を出力します。
task2()	タスク	2	2	“task2 running”を出力します。
cyh1()	周期ハンドラ	1	—	task1 を起床します。

【誤】

表 11.1 サンプルプログラムの関数一覧

関数名	種類	ID 番号	優先度	機能
main()	タスク	1	1	task1,task2 を起動させます。
task1()	タスク	2	2	“task1 running”を出力します。
task2()	タスク	3	3	“task2 running”を出力します。
cyh1()	周期ハンドラ	1	—	task1 を起床します。

【正】

以下に、処理内容を説明します。

- task1 は次の順で動作します。

【誤】

以下に、処理内容を説明します。

- main タスクは task1, task2, cyh1 を起動し、自タスクを終了させます。
- task1 は次の順で動作します。

10. アップデート履歴

10.1 V.1.00 Release 00 (2009 年 10 月)

最初のリリース。

10.2 V.1.00 Release 01 (2010 年 2 月)

10.2.1 カーネル

(1) 時間待ちを伴うサービスコールの不具合の改修

以下の不具合を改修しました。

時間待ちを伴うサービスコール(下記参照)によって待ち状態に遷移したタスクが、時間経過以外の要因で待ち解除された場合、以下のような不具合現象が発生する場合があります。

- (a) 時間待ちを伴うサービスコールを呼び出したタスクが、指定した時間が経過しても待ち状態が解除されない。
- (b) 時間待ちを伴うサービスコールを呼び出したタスクの待ち状態が解除される時刻が遅れる。
- (c) タスクが不正動作となる。

時間待ちを伴うサービスコール：

tslp_tsk, dly_tsk, twai_sem, twai_flg, tsnd_dtq, trcv_dtq, trcv_mbx, tloc_mtx, tsnd_mbf, trcv_mbf, tget_mpf, tget_mpl

(2) 周期ハンドラおよびアラームハンドラの不具合(1)

以下の不具合を改修しました。

周期ハンドラまたはアラームハンドラが R5 レジスタを更新して終了した場合、カーネル内部で不正アドレスへのアクセス(リード・ライト)が発生します。

(3) 周期ハンドラおよびアラームハンドラの不具合(2)

以下の不具合を改修しました。

周期ハンドラまたはアラームハンドラが R8～R13 レジスタを更新して終了した場合、システムクロック割込み発生前に実行していたプログラムの R8～R13 レジスタ値が破壊されます。

(4) システムクロック割込みハンドラの不具合

以下の不具合を改修しました。

アプリケーションで DSP 機能命令を使用している場合、システムクロック割込み発生前に実行していたプログラムの ACC レジスタ値が破壊される場合があります。

(5) irel_mpf の不具合

以下の不具合を改修しました。

irel_mpf 呼び出し前後で、R6 レジスタが保証されない場合があります。

(6) スタックサイズの変更

「7.1 システムクロック割込みハンドラのスタックサイズ」の内容を以下のように変更しました。

【変更前】

マニュアル 12.4 節に記載の $\epsilon 1$, $\epsilon 2$ および $\epsilon 3$ の値は、以下の通りです。

$\epsilon 1 = 72$
 $\epsilon 2 = 72$
 $\epsilon 3 = 144$

【変更後】

マニュアル 12.4 節に記載の $\epsilon 1$, $\epsilon 2$ および $\epsilon 3$ の値は、以下の通りです。

$\epsilon 1 = 104$
 $\epsilon 2 = 104$
 $\epsilon 3 = 196$

また、「7.2 サービスコールで使用するスタックサイズ」の表を変更しました。

変更したサービスコールは、以下の通りです。

- `ter_tsk` のシステムスタック使用サイズ : 112⇒132
- `rel_wai` のシステムスタック使用サイズ : 96⇒116
- `irel_wai` のシステムスタック使用サイズ : 112⇒132 *
- `irel_mpf` のシステムスタック使用サイズ : 48⇒52 *
- `get_mpl` のシステムスタック使用サイズ : 76⇒96
- `pget_mpl` のシステムスタック使用サイズ : 92⇒112
- `ipget_mpl` のシステムスタック使用サイズ : 92⇒112 *
- `tget_mpl` のシステムスタック使用サイズ : 76⇒96
- `rel_mpl` のシステムスタック使用サイズ : 88⇒108

また、*で示したサービスコールについては、スタック算出ユーティリティ“Call Walker”で表示される値も変更になります。

(7) 注意事項の追加

本資料の「8.3 0 番地に関する注意事項」を追加しました。

(8) その他

以下の値が変更になりました。

項目	変更前	変更後
TKERNEL_PRVER マクロ、および <code>ref_ver</code> で返る <code>T_RVER.prver</code>	0x0100	0x0101

RI600/4リリースノート

発行日：2010年2月1日

Copyright (C) 2009(2010) Renesas Technology Corp. and Renesas Solutions Corp. All rights reserved.

<http://www.renesas.com/>