RL78コード生成ツール

CS+ RL78コード生成 (CS+ for CC/CA,CX) V2.21.00, e² studio Code Generatorプラグイン V2.18.0, AP4 for RL78 V1.20.00, Applilet3 for RL78 V1.20.00 リリースノート

R20UT4632JJ0105
Rev.1.05
2021.10.15

要旨

この度は、RL78 コード生成ツールをご使用いただきまして、誠にありがとうございます。この添付資料では、本製品をお使いいただく上でのサポート機能および注意事項等を記載しております。ご使用の前に、必ずお読みくださいますようお願い申し上げます。

RENESAS の Web サイトでも最新のリリースノートを確認いただけます。

- RL78コード生成ツール リリースノート

目次

1. はじめに 4
1.1 製品バージョン 4
1.2 動作環境 5
1.2.1 PC 5
1.2.2 開発ツール 5
1.2.2.1 CS+ 5
1.2.2.2 e² studio, AP4 for RL78およびApplilet3 for RL78 5

2. デバイス一覧 6

3. 変更点 11
3.1 変更点一覧 11
3.2 変更点一覧 15
3.2.1 サポートデバイスの追加 15
3.2.2 IICA0 またはIICA1 でシングルマスタ・システムを選択した場合の注意事項の解除 15
3.2.3 CSI設定で連続転送モードを選択した場合の注意事項の解除 15
3.2.4 R_CGC_Set_ClockMode()のクロック・モード選択(SELPLL)の誤りの修正 15
3.2.5 CSI およびUART のコールバック機能設定に関する注意事項 15
3.2.6 UARTの送信データ・レベル設定誤表記の修正 15
3.2.7 オンチップ・デバック設定のトレース機能を使用する場合の注意事項の解除 16
3.2.8 時間キャプチャ入力機能使用時のRTCCRn（n = 0-2）設定のTCSTビットの制限を解除 16
3.2.9 クロック設定の「動作モード」と「PLL出力クロック(fPLL)」の誤ったUI操作を改善 16
3.2.10 R_CGC_Create()にfCLK設定コードを追加 17
3.2.11 R_CGC_Set_LPMode()とR_CGC_Set_LSMode()のコード・プレビューの改善 17
3.2.12 R_ELC_Stop()の改善 17
3.2.13 低速メイン・モード選択時の中速オンチップ・オシレータ・クロックの改善 17
3.2.14 高速メイン・モード選択時のfMX入力範囲の改善 17
3.2.15 r_lvd_vddinterrupt()生成条件の改善 17

3.2.16 EXLVD設定を維持できない問題の改善 18
3.2.17 オプション・バイト設定値の誤りの修正 18
3.2.18 ユーザーズマニュアル(HW)のサポートバージョン更新 18
3.2.19 PIORの特定設定値におけるELCとDTC生成コードの改善 18
3.2.20 fIH選択を維持できない問題の改善 18
3.2.21 IICAマスタのリスタートの改善 19
3.2.22 簡易IICのWait処理の改善 20
3.2.23 LVDオフのオプション・バイト設定値の改善 24

4. RENESAS TOOL NEWSの改修履歴 25

5. 制限事項 31
5.1 制限事項一覧 31
5.2 制限事項詳細 33
5.2.1 タイマ・アレイ・ユニット入力ソースについて 33
5.2.2 24ピンデバイスのTAU0チャネル1設定の制限について 33
5.2.3 リアルタイム・クロックのAPI関数について 33
5.2.4 PGA+ΔΣA/Dコンバータ ゲイン設定の単位の表記ミスについて 33
5.2.5 UARTの送信データ・レベル設定誤表記について 34
5.2.6 CSIの連続転送モード時の制限について 34
5.2.7 データ・フラッシュ・ライブラリの関数説明の誤りについて 37
5.2.8 TAUの入力パルス間隔測定でfSUB、fIL選択時の動作クロックについて 37
5.2.9 e² studio 64ビット環境の制限について 37
5.2.10 R_ELC_Stop()関数の誤りについて 38
5.2.11 トレース・アドレスの誤りについて 39
5.2.12 IAR Embedded WorkbenchのR_ELC_Stop()関数ビルド・エラーについて 39
5.2.13 CS+およびe² studioでのデバイス変更機能に関する制限 40

6. 注意事項 41
6.1 注意事項一覧 41
6.2 注意事項詳細 44
6.2.1 オンラインヘルプについて (Applilet3, AP4) 44
6.2.2 MISRA-Cのコーディング規約対応について 44
6.2.3 高速オンチップ・オシレータ周波数選択レジスタ(HOCODIV)について 44
6.2.4 高速及び低速内蔵発振器精度トリミングレジスタについて 44
6.2.5 シリアル・アレイ・ユニットの制限について 44
6.2.6 フラッシュ・メモリCRC演算機能（高速CRC）について 44
6.2.7 ポート・モード選択レジスタ（PMS）について 44
6.2.8 UARTのLIN-bus機能について 44
6.2.9 IICAの拡張コード、マルチマスタ、ウエイク・アップ機能および簡易IICのマルチマスタ機能について 44
6.2.10 CANコントローラ機能について 45
6.2.11 安全機能について 45
6.2.12 USB機能について 45
6.2.13 RI78V4プロジェクトについて 45
6.2.14 DTC機能を使用する時の注意(CS+ for CA,CX) 45
6.2.15 高速DTCのチェイン転送について 47

6.2.16 IICAのスレーブ時のファスト・モード・プラス設定について 47
6.2.17 高速オンチップ・オシレータの設定について(CS+ for CA,CX) 47
6.2.18 端子配置ツールについて(CS+ for CA,CX) 47
6.2.19 RL78/G13Aの生成ファイルのVersion表記について 47
6.2.20 簡易IIC停止条件の生成について 48
6.2.21 デバイス変更機能について 48

改訂記録 49

1. はじめに

RL78 コード生成ツールは、マイコン周辺機能（タイマ，UART，A/D，etc）を制御するプログラム（デバイス・ドライバ・プログラム）を GUI 設定により自動生成するツールです。各周辺の初期化処理以外にも周辺機能を操作する関数を API (Application Programming Interface) として提供します。

RL78 コード生成ツールとして、以下の製品を提供しています。

- RL78 コード生成プラグイン（統合開発環境 CS+ for CC, CS+ for CA,CX, e² studio）
- AP4 for RL78
- Applilet3 for RL78

1.1 製品バージョン

- CS+ Code Generator for RL78 (CS+ for CC)　2.21.00
- CS+ Code Generator for RL78 (CS+ for CA,CX)　2.21.00
- e² studio Code Generator Plug-in　2.18.00
- Applilet3 for RL78　1.20.00 (4.08.05.01)

グループ	バージョン	グループ	バージョン
RL78/D1A	V2.04.04.01	RL78/G13, G13A	V2.05.05.01
RL78/F12	V2.04.05.01	RL78/G14	V2.05.05.01
RL78/F13	V2.03.06.01	RL78/G1A	V2.04.03.01
RL78/F14	V2.03.06.01	RL78/I1A	V2.04.04.01
RL78/F15	V1.01.07.01	RL78/L12	V2.04.04.01
RL78/G12	V2.04.05.01		

- AP4 for RL78　1.20.00 (2.10.06.01)

グループ	バージョン	グループ	バージョン
RL78/F1E	V1.01.06.01	RL78/H1D	V1.00.02.01
RL78/G10	V1.05.04.01	RL78/I1B	V1.03.03.01
RL78/G11	V1.02.05.01	Rl78/I1C	V1.01.06.01
RL78/G1C	V1.03.03.01	RL78/I1D	V1.01.04.01
RL78/G1D	V1.01.03.01	RL78/I1E	V1.03.04.01
RL78/G1E	V1.04.03.01	RL78/L13	V1.04.04.01
RL78/G1F	V1.01.05.01	RL78/L1A	V1.01.04.01
RL78/G1G	V1.01.02.01	RL78/L1C	V1.03.02.01
RL78/G1H	V1.01.04.01		

1.2　動作環境

1.2.1　PC

- BM PC/AT 互換機（Windows® 10, Windows® 8.1）
- プロセッサ： 1GHz 以上（ハイパースレッディング，マルチコア CPU に対応）
- メモリ容量： 推奨 2GB 以上。最低 1GB 以上（64 ビット版 Windows では 2G バイト以上）
- ハードディスク容量： 空き容量 200MB 以上
- ディスプレイ： 1024×768 以上の解像度，65536 色以上
- Windows OS 以外に必要なソフトウェア環境： .NET Framework 4.5 + 言語パック

1.2.2　開発ツール

1.2.2.1　CS+

- ルネサスエレクトロニクス製 統合開発環境 CS+ V8.05.00 以上
- ルネサスエレクトロニクス製 RL78 用コンパイラ CC-RL V1.10 以上
- ルネサスエレクトロニクス製 78K0R 用コンパイラ CA78K0R V1.72 以上

1.2.2.2　e^2 studio, AP4 for RL78 および Applilet3 for RL78

- ルネサスエレクトロニクス製 統合開発環境 e^2 studio (64-bit) 2021-01 以上
- ルネサスエレクトロニクス製 RL78 用コンパイラ CC-RL V1.10 以上
- Renesas GCC for RL78 V4.9 以上
- ルネサスエレクトロニクス RL78 用 IAR Embedded Workbench V4.20 以上

2. デバイス一覧

RL78 コード生成ツールのサポートデバイス一覧です。

表 2-1　サポートデバイス

○: Support -: Not support

グループ (HW マニュアル番号)	ピン数	デバイス名	CS+	e² studio	AP4	Applilet3
RL78/F12 グループ (R01UH0231JJ0111)	20pin	R5F1096E, R5F1096D, R5F1096C, R5F1096B, R5F1096A, R5F10968	○	○	-	○
	30pin	R5F109AE, R5F109AD, R5F109AC, R5F109AB, R5F109AA	○	○	-	○
	32pin	R5F109BE, R5F109BD, R5F109BC, R5F109BB, R5F109BA	○	○	-	○
	48pin	R5F109GE, R5F109GD, R5F109GC, R5F109GB, R5F109GA	○	○	-	○
	64pin	R5F109LE, R5F109LD, R5F109LC, R5F109LB, R5F109LA	○	○	-	○
RL78/F13 グループ (R01UH0368JJ0210)	20pin	R5F10A6A, R5F10A6C, R5F10A6D, R5F10A6E	○	○	-	○
	30pin	R5F10AAA, R5F10AAC, R5F10AAD, R5F10AAE, R5F10BAC, R5F10BAD, R5F10BAE, R5F10BAF, R5F10BAG	○	○	-	○
	32pin	R5F10ABA, R5F10ABC, R5F10ABD, R5F10ABE, R5F10BBC, R5F10BBD, R5F10BBE, R5F10BBF, R5F10BBG	○	○	-	○
	48pin	R5F10AGA, R5F10AGC, R5F10AGD, R5F10AGE, R5F10AGF, R5F10AGG, R5F10BGC, R5F10BGD, R5F10BGE, R5F10BGF, R5F10BGG	○	○	-	○
	64pin	R5F10BLC, R5F10ALD, R5F10ALE, R5F10ALF, R5F10ALG, R5F10BLC, R5F10BLD, R5F10BLE, R5F10BLF, R5F10BLG	○	○	-	○
	80pin	R5F10AME, R5F10AMF, R5F10AMG, R5F10BME, R5F10BMF, R5F10BMG	○	○	-	○
RL78/F14 グループ (R01UH0368JJ0210)	30pin	R5F10PAD, R5F10PAE	○	○	-	○
	32pin	R5F10PBD, R5F10PBE	○	○	-	○
	48pin	R5F10PGD, R5F10PGE, R5F10PGF, R5F10PGG, R5F10PGH, R5F10PGJ	○	○	-	○
	64pin	R5F10PLE, R5F10PLF, R5F10PLG, R5F10PLH, R5F10PLJ	○	○	-	○
	80pin	R5F10PME, R5F10PMF, R5F10PMG, R5F10PMH, R5F10PMJ	○	○	-	○
	100pin	R5F10PPE, R5F10PPF, R5F10PPG, R5F10PPH, R5F10PPJ	○	○	-	○
RL78/F15 グループ (R01UH0559JJ0100)	48pin	R5F113GL, R5F113GK	○	○	-	○
	64pin	R5F113LL, R5F113LK	○	○	-	○
	80pin	R5F113ML, R5F113MK	○	○	-	○
	100pin	R5F113PL, R5F113PK, R5F113PJ, R5F113PH, R5F113PG	○	○	-	○
	144pin	R5F113TL, R5F113TK, R5F113TJ, R5F113TH, R5F113TG	○	○	-	○
RL78/F1E グループ (R01UH0611JJ0052)	64pin	R5F11KLE, R5F11LLE, R5F11KLF, R5F11LLF, R5F11KLG, R5F11LLG	○	○	○	-

表 2-2　サポートデバイス

○: Support, -: Not support

グループ (HW マニュアル番号)	ピン数	デバイス名	CS+	e² studio	AP4	Applilet3
RL78/G10 グループ (R01UH0384JJ0311)	10pin	R5F10Y14, R5F10Y16, R5F10Y17	○	○	○	-
	16pin	R5F10Y44, R5F10Y46, R5F10Y47	○	○	○	-
RL78/G11 グループ (R01UH0637JJ0110)	10pin	R5F1051A	○	○	○	-
	16pin	R5F1054A	○	○	○	-
	20pin	R5F1056A	○	○	○	-
	24pin	R5F1057A	○	○	○	-
	25pin	R5F1058A	○	○	○	-
RL78/G12 グループ (R01UH0200JJ0210)	20pin	R5F10266, R5F10267, R5F10268, R5F10269, R5F1026A, R5F10366, R5F10367, R5F10368, R5F10369, R5F1036A	○	○	-	○
	24pin	R5F10277, R5F10278, R5F10279, R5F1027A, R5F10377, R5F10378, R5F10379, R5F1037A	○	○	-	○
	30pin	R5F102A7, R5F102A8, R5F102A9, R5F102AA, R5F103A7, R5F103A8, R5F103A9, R5F103AA	○	○	-	○
RL78/G13 グループ (R01UH0146JJ0330)	20pin	R5F1006A, R5F1006C, R5F1006D, R5F1006E, R5F1016A, R5F1016C, R5F1016D, R5F1016E	○	○	-	○
	24pin	R5F1007A, R5F1007C, R5F1007D, R5F1007E, R5F1017A, R5F1017C, R5F1017D, R5F1017E	○	○	-	○
	25pin	R5F1008A, R5F1008C, R5F1008D, R5F1008E, R5F1018A, R5F1018C, R5F1018D, R5F1018E	○	○	-	○
	30pin	R5F100AA, R5F100AC, R5F100AD, R5F100AE, R5F100AF, R5F100AG, R5F101AA, R5F101AC, R5F101AD, R5F101AE, R5F101AF, R5F101AG	○	○	-	○
	32pin	R5F100BA, R5F100BC, R5F100BD, R5F100BE, R5F100BF, R5F100BG, R5F101BA, R5F101BC, R5F101BD, R5F101BE, R5F101BF, R5F101BG	○	○	-	○
	36pin	R5F100CA, R5F100CC, R5F100CD, R5F100CE, R5F100CF, R5F100CG, R5F101CA, R5F101CC, R5F101CD, R5F101CE, R5F101CF, R5F101CG	○	○	-	○
	40pin	R5F100EA, R5F100EC, R5F100ED, R5F100EE, R5F100EF, R5F100EG, R5F100EH, R5F101EA, R5F101EC, R5F101ED, R5F101EE, R5F101EF, R5F101EG, R5F101EH	○	○	-	○
	44pin	R5F100FA, R5F100FC, R5F100FD, R5F100FE, R5F100FF, R5F100FG, R5F100FH, R5F100FJ, R5F100FK, R5F100FL, R5F101FA, R5F101FC, R5F101FD, R5F101FE, R5F101FF, R5F101FG, R5F101FH, R5F101FJ, R5F101FK, R5F101FL	○	○	-	○
	48pin	R5F100GA, R5F100GC, R5F100GD, R5F100GE, R5F100GF, R5F100GG, R5F100GH, R5F100GJ, R5F100GK, R5F100GL, R5F101GA, R5F101GC, R5F101GD, R5F101GE, R5F101GF, R5F101GG, R5F101GH, R5F101GJ, R5F101GK, R5F101GL	○	○	-	○
	52pin	R5F100JC, R5F100JD, R5F100JE, R5F100JF, R5F100JG, R5F100JH, R5F100JJ, R5F100JK, R5F100JL, R5F101JC, R5F101JD, R5F101JE, R5F101JF, R5F101JG, R5F101JH, R5F101JJ, R5F101JK, R5F101JL	○	○	-	○

表 2-3　サポートデバイス

○: Support, -: Not support

グループ (HW マニュアル番号)	ピン数	デバイス名	CS+	e² studio	AP4	Applilet3
RL78/G13 グループ (R01UH0146JJ0330)	64pin	R5F100LC, R5F100LD, R5F100LE, R5F100LF, R5F100LG, R5F100LH, R5F100LJ, R5F100LK, R5F100LL, R5F101LC, R5F101LD, R5F101LE, R5F101LF, R5F101LG, R5F101LH, R5F101LJ, R5F101LK, R5F101LL	○	○	-	○
	80pin	R5F100MF, R5F100MG, R5F100MH, R5F100MJ, R5F100MK, R5F100ML, R5F101MF, R5F101MG, R5F101MH, R5F101MJ, R5F101MK, R5F101ML	○	○	-	○
	100pin	R5F100PF, R5F100PG, R5F100PH, R5F100PJ, R5F100PK, R5F100PL, R5F101PF, R5F101PG, R5F101PH, R5F101PJ, R5F101PK, R5F101PL	○	○	-	○
	128pin	R5F100SH, R5F100SJ, R5F100SK, R5F100SL, R5F101SH, R5F101SJ, R5F101SK, R5F101SL	○	○	-	○
RL78/G13A グループ (R01UH0856JJ0050)	44pin	R5F140FK, R5F140FL	○	○	-	○
	48pin	R5F140GK, R5F140GL	○	○	-	○
	64pin	R5F140LK, R5F140LL	○	○	-	○
	100pin	R5F140PK, R5F140PL	○	○	-	○
RL78/G14 グループ (R01UH0186JJ0330)	30pin	R5F104AA, R5F104AC, R5F104AD, R5F104AE, R5F104AF, R5F104AG	○	○	-	○
	32pin	R5F104BA, R5F104BC, R5F104BD, R5F104BE, R5F104BF, R5F104BG	○	○	-	○
	36pin	R5F104CA, R5F104CC, R5F104CD, R5F104CE, R5F104CF, R5F104CG	○	○	-	○
	40pin	R5F104EA, R5F104EC, R5F104ED, R5F104EE, R5F104EF, R5F104EG, R5F104EH	○	○	-	○
	44pin	R5F104FA, R5F104FC, R5F104FD, R5F104FE, R5F104FF, R5F104FG, R5F104FH, R5F104FJ	○	○	-	○
	48pin	R5F104GA, R5F104GC, R5F104GD, R5F104GE, R5F104GF, R5F104GG, R5F104GH, R5F104GJ, R5F104GK, R5F104GL	○	○	-	○
	52pin	R5F104JC, R5F104JD, R5F104JE, R5F104JF, R5F104JG, R5F104JH, R5F104JJ	○	○	-	○
	64pin	R5F104LC, R5F104LD, R5F104LE, R5F104LF, R5F104LG, R5F104LH, R5F104LJ, R5F104LK, R5F104LL	○	○	-	○
	80pin	R5F104MF, R5F104MG, R5F104MH, R5F104MJ, R5F104MK, R5F104ML	○	○	-	○
	100pin	R5F104PF, R5F104PG, R5F104PH, R5F104PJ, R5F104PK, R5F104PL	○	○	-	○
RL78/G1A グループ (R01UH0305JJ0200)	25pin	R5F10E8A, R5F10E8C, R5F10E8D, R5F10E8E	○	○	-	○
	32pin	R5F10EBA, R5F10EBC, R5F10EBD, R5F10EBE	○	○	-	○
	48pin	R5F10EGA, R5F10EGC, R5F10EGD, R5F10EGE	○	○	-	○
	64pin	R5F10ELC, R5F10ELD, R5F10ELE	○	○	-	○
RL78/G1C グループ (R01UH0348JJ0100)	32pin	R5F10JBC, R5F10KBC	○	○	○	-
	48pin	R5F10JGC, R5F10KGC	○	○	○	-

表 2-4　サポートデバイス

○: Support, -: Not support

グループ (HW マニュアル番号)	ピン数	デバイス名	CS+	e² studio	AP4	Applilet3
RL78/G1D グループ (R01UH0515JJ0100)	48pin	R5F11AGG, R5F11AGH, R5F11AGJ	○	○	○	-
RL78/G1E グループ (R01UH0353JJ0101)	64pin	R5F10FLC, R5F10FLD, R5F10FLE	○	○	○	-
	80pin	R5F10FMC, R5F10FMD, R5F10FME	○	○	○	-
RL78/G1F グループ (R01UH0516JJ0100)	24pin	R5F11B7C, R5F11B7E	○	○	○	-
	32pin	R5F11BBC, R5F11BBE	○	○	○	-
	36pin	R5F11BCC, R5F11BCE	○	○	○	-
	48pin	R5F11BGC, R5F11BGE	○	○	○	-
	64pin	R5F11BLC, R5F11BLE	○	○	○	-
RL78/G1G グループ (R01UH0499JJ0100)	30pin	R5F11EA8, R5F11EAA	○	○	○	-
	32pin	R5F11EB8, R5F11EBA	○	○	○	-
	44pin	R5F11EF8, R5F11EFA	○	○	○	-
RL78/G1H グループ (R01UH0575JJ0100)	64pin	R5F11FLJ, R5F11FLK, R5F11FLL	○	○	○	-
RL78/H1D グループ (R01UH0756JJ0080)	48pin	R5F11NGG, R5F11NGF	○	○	○	-
	64pin	R5F11NLG, R5F11PLG, R5F11NLF, R5F11PLF	○	○	○	-
	80pin	R5F11RMG, R5F11NMG, R5F11NMF, R5F11NME	○	○	○	-
RL78/I1A グループ (R01UH0169JJ0210)	20pin	R5F1076C	○	○	-	○
	30pin	R5F107AC, R5F107AE	○	○	-	○
	38pin	R5F107DE	○	○	-	○
RL78/I1B グループ (R01UH0407JJ0100)	80pin	R5F10MME, R5F10MMG	○	○	○	-
	100pin	R5F10MPE, R5F10MPG	○	○	○	-
RL78/I1C グループ (R01UH0587JJ0210)	64pin	R5F10NLE, R5F10NLG, R5F11TLE, R5F11TLG	○	○	○	-
	80pin	R5F10NME, R5F10NMG, R5F10NMJ	○	○	○	-
	100pin	R5F10NPJ, R5F10NPG	○	○	○	-
RL78/I1C (512KB) グループ (R01UH0889JJ0100)	80pin	R5F10NML, R5F10NML(DUAL)	○	○	○	-
	100pin	R5F10NPL, R5F10NPL(DUAL)	○	○	○	-

表 2-5　サポートデバイス

○: Support, -: Not support

グループ (HW マニュアル番号)	ピン数	デバイス名	CS+	e² studio	AP4	Applilet3
RL78/I1D グループ (R01UH0474JJ0100)	20pin	R5F11768, R5F1176A	○	○	○	-
	24pin	R5F11778, R5F1177A	○	○	○	-
	30pin	R5F117A8, R5F117AA, R5F117AC	○	○	○	-
	32pin	R5F117BA, R5F117BC	○	○	○	-
	48pin	R5F117GA, R5F117GC	○	○	○	-
RL78/I1E グループ (R01UH0524JJ0100)	32pin	R5F11CBC	○	○	○	-
	36pin	R5F11CCC	○	○	○	-
RL78/L12 グループ (R01UH0330JJ0200)	32pin	R5F10RBC, R5F10RBA, R5F10RB8	○	○	-	○
	44pin	R5F10RFC, R5F10RFA, R5F10RF8	○	○	-	○
	48pin	R5F10RGC, R5F10RGA, R5F10RG8	○	○	-	○
	52pin	R5F10RJC, R5F10RJA, R5F10RJ8	○	○	-	○
	64pin	R5F10RLC, R5F10RLA	○	○	-	○
RL78/L13 グループ (R01UH0382JJ0100)	64pin	R5F10WLA, R5F10WLC, R5F10WLD, R5F10WLE, R5F10WLF, R5F10WLG	○	○	○	-
	80pin	R5F10WMA, R5F10WMC, R5F10WMD, R5F10WME, R5F10WMF, R5F10WMG	○	○	○	-
RL78/L1A グループ (R01UH0636JJ0100)	80pin	R5F11MMD, R5F11MME, R5F11MMF	○	○	○	-
	100pin	R5F11MPE, R5F11MPF, R5F11MPG	○	○	○	-
RL78/L1C グループ (R01UH0409JJ0100)	80pin	R5F110MJ, R5F110MH, R5F110MG, R5F110MF, R5F110ME, R5F111MJ, R5F111MH, R5F111MG, R5F111MF, R5F111ME	○	○	○	-
	100pin	R5F110PJ, R5F110PH, R5F110PG, R5F110PF, R5F110PE, R5F111PJ, R5F111PH, R5F111PG, R5F111PF, R5F111PE	○	○	○	-
RL78/D1A グループ (R01UH0317EJ0003)	48pin	R5F10CGB, R5F10CGC, R5F10CGD, R5F10DGC, R5F10DGD, R5F10DGE	-	○	-	○
	64pin	R5F10CLD, R5F10DLD, R5F10DLE	-	○	-	○
	80pin	R5F10CMD, R5F10CME, R5F10DMD, R5F10DME, R5F10DMF, R5F10DMG, R5F10DMJ	-	○	-	○
	100pin	R5F10DPE, R5F10DPF, R5F10DPG, R5F10DPJ, R5F10TPJ	-	○	-	○

3. 変更点

RL78 コード生成ツールの本リリースにおける変更点を説明します。

3.1 変更点一覧

表 3-1 変更点一覧 1/2

○：対象項目，/：非対象項目

No	内容	グループ													
		RL78/D1A	RL78/G10	RL78/G11	RL78/G12	RL78/G13, G13A	RL78/G14	RL78/G1A	RL78/G1C	RL78/G1D	RL78/G1E	RL78/G1F	RL78/G1G	RL78/G1H	RL78/H1D
1	サポートデバイスの追加	/	/	/	/	/	/	/	/	/	/	/	/	/	/
2	IICA0 または IICA1 でシングルマスタ・システムを選択した場合の注意事項の解除 (r20ts0459)	/	/	/	/	/	/	/	/	/	/	/	/	/	○
3	CSI 設定で連続転送モードを選択した場合の注意事項の解除 (r20ts0244)	/	/	/	/	/	/	/	/	/	/	/	/	/	○
4	R_CGC_Set_ClockMode()のクロック・モード選択(SELPLL)の誤りの修正	/	/	/	/	/	/	/	/	/	/	/	/	/	/
5	CSI および UART のコールバック機能設定に関する注意事項 (r20ts0545)	/	/	○	/	/	/	/	/	/	/	/	/	/	/
6	UART の送信データ・レベル設定誤表記の修正	/	/	/	/	/	/	/	/	/	/	○	/	/	/
7	オンチップ・デバック設定のトレース機能を使用する場合の注意事項の解除 (r20ts0544 含む)	/	/	/	/	/	/	/	/	/	/	/	/	/	/
8	時間キャプチャ入力機能使用時の RTCCRn（n = 0-2）設定の TCST ビットの制限を解除	/	/	/	/	/	/	/	/	/	/	/	/	/	/
9	クロック設定の「動作モード」と「PLL 出力クロック(fPLL)」の誤った UI 操作を改善	/	/	/	/	/	/	/	/	/	/	/	/	/	/
10	R_CGC_Create()に fCLK 設定コードを追加	/	/	/	/	/	/	/	/	/	/	/	/	/	/
11	R_CGC_Set_LPMode()と R_CGC_Set_LSMode()のコード・プレビューの改善	/	/	/	/	/	/	/	/	/	/	/	/	/	/
12	R_ELC_Stop()の改善	/	/	/	/	/	/	/	/	/	/	/	/	/	/
13	低速メイン・モード選択時の中速オンチップ・オシレータ・クロックの改善	/	/	/	/	/	/	/	/	/	/	/	/	/	/
14	高速メイン・モード選択時の fMX 入力範囲の改善	/	/	/	/	/	/	/	/	/	/	/	/	/	/
15	r_lvd_vddinterrupt()生成条件の改善	/	/	/	/	/	/	/	/	/	/	/	/	/	/
16	EXLVD 設定を維持できない問題の改善	/	/	/	/	/	/	/	/	/	/	/	/	/	/
17	オプション・バイト設定値の誤りの修正	/	/	/	/	/	/	/	/	/	/	/	/	/	/

【注】 No.2 から 17 は Applilet3 と AP4 のみの変更内容です。 コード生成プラグイン（CS+、e^2 studio）は、前バージョンで改善されています。

表 3-1 変更点一覧 2/2

○：対象項目，/：非対象項目

No	内容	グループ													
		RL78/D1A	RL78/G10	RL78/G11	RL78/G12	RL78/G13, G13A	RL78/G14	RL78/G1A	RL78/G1C	RL78/G1D	RL78/G1E	RL78/G1F	RL78/G1G	RL78/G1H	RL78/H1D
18	ユーザーズマニュアル(HW)のサポートバージョン更新	/	/	/	/	/	/	/	/	/	/	/	/	/	/
19	PIOR の特定設定値における ELC と DTC 生成コードの改善	/	/	/	/	/	/	/	/	/	/	/	/	/	/
20	fIH 選択を維持できない問題の改善	/	/	/	/	/	/	/	/	/	/	/	/	/	/
21	IICA マスタのリスタートの改善	/	○	○	○	○	○	○	○	○	/	○	/	○	○
22	簡易 IIC の Wait 処理の改善	○	○	○	○	○	○	○	○	○	○	○	○	/	○
23	ユーザ・オプション・バイトの LVD オフ設定値の改善 (r20TS0571)	/	/	/	/	○	/	/	/	/	/	/	/	/	/

【注】 No.18 から 20 は Applilet3 と AP4 のみの変更内容です。 コード生成プラグイン（CS+、e^2 studio）は、前バージョンで改善されています。

表 3-2 変更点一覧 1/2

○：対象項目，/：非対象項目

No	内容	グループ													
		RL78/F12	RL78/F13	RL78/F14	RL78/F15	RL78/F1E	RL78/I1A	RL78/I1B	RL78/I1C	RL78/I1D	RL78/I1E	RL78/L12	RL78/L13	RL78/L1A	RL78/L1C
1	サポートデバイスの追加	/	/	/	/	/	/	/	○	/	/	/	/	/	/
2	IICA0 または IICA1 でシングルマスタ・システムを選択した場合の注意事項の解除 (r20ts0459)	/	/	/	/	/	/	/	○	/	/	○	○	/	/
3	CSI 設定で連続転送モードを選択した場合の注意事項の解除 (r20ts0244)	/	/	/	/	/	/	/	○	/	/	○	○	/	/
4	R_CGC_Set_ClockMode()のクロック・モード選択(SELPLL)の誤りの修正	/	/	/	/	○	/	/	/	/	/	/	/	/	/
5	CSI および UART のコールバック機能設定に関する注意事項 (r20ts0545)	/	/	/	/	/	/	/	/	/	/	/	/	/	/
6	UART の送信データ・レベル設定誤表記の修正	/	/	/	/	/	/	/	/	/	/	/	/	/	/
7	オンチップ・デバック設定のトレース機能を使用する場合の注意事項の解除 (r20ts0544 含む)	/	/	/	○	/	/	/	/	/	/	/	/	/	/
8	時間キャプチャ入力機能使用時の RTCCRn（n = 0-2）設定の TCST ビットの制限を解除	/	/	/	/	/	/	/	○	/	/	/	/	/	/
9	クロック設定の「動作モード」と「PLL 出力クロック(fPLL)」の誤った UI 操作を改善	/	/	/	/	/	/	/	○	/	/	/	/	/	/
10	R_CGC_Create()に fCLK 設定コードを追加	/	/	/	/	/	/	/	○	/	/	/	/	/	/

【注】 No.2 から 10 は Applilet3 と AP4 のみの変更内容です。 コード生成プラグイン（CS+、e^2 studio）は、前バージョンで改善されています。

表 3-2 変更点一覧 2/2

○：対象項目，/：非対象項目

No	内容	グループ													
		RL78/F12	RL78/F13	RL78/F14	RL78/F15	RL78/F1E	RL78/I1A	RL78/I1B	RL78/I1C	RL78/I1D	RL78/I1E	RL78/L12	RL78/L13	RL78/L1A	RL78/L1C
11	R_CGC_Set_LPMode()と R_CGC_Set_LSMode()のコード・プレビューの改善	/	/	/	/	/	/	/	○	/	/	/	/	/	/
12	R_ELC_Stop()の改善	/	/	/	/	/	/	/	○	/	/	/	/	/	/
13	低速メイン・モード選択時の中速オンチップ・オシレータ・クロックの改善	/	/	/	/	/	/	/	○	/	/	/	/	/	/
14	高速メイン･モード選択時の fMX 入力範囲の改善	/	/	/	/	/	/	/	○	/	/	/	/	/	/
15	r_lvd_vddinterrupt()生成条件の改善	/	/	/	/	/	/	/	○	/	/	/	/	/	/
16	EXLVD 設定を維持できない問題の改善	/	/	/	/	/	/	/	○	/	/	/	/	/	/
17	オプション・バイト設定値の誤りの修正	/	/	/	/	○	/	/	/	/	/	/	/	/	/
18	ユーザーズマニュアル(HW)のサポートバージョン更新	/	/	/	/	/	/	/	○	/	/	/	/	/	/
19	PIOR の特定設定値における ELC と DTC 生成コードの改善	/	/	/	/	/	/	/	○	/	/	/	/	/	/
20	fIH 選択を維持できない問題の改善	/	/	/	/	/	/	/	○	/	/	/	/	/	/
21	IICA マスタのリスタートの改善	○	○	○	○	/	○	○	○	/	/	○	○	○	○
22	簡易 IIC の Wait 処理の改善	○	○	○	○	○	/	○	○	○	○	/	○	○	○
23	ユーザ・オプション・バイトの LVD オフ設定値の改善 (r20TS0571)	/	/	/	/	/	/	/	/	/	/	/	/	/	/

【注】 No.11 から 20 は Applilet3 と AP4 のみの変更内容です。 コード生成プラグイン（CS+、e^2 studio）は、前バージョンで改善されています。

3.2　変更点一覧

3.2.1　サポートデバイスの追加

サポートデバイスに以下のデバイスを追加しました。
RL78/I1C (128K): R5F10NPG
RL78/I1C (512KB): R5F10NML, R5F10NML(DUAL), R5F10NPL, R5F10NPL(DUAL)

3.2.2　IICA0 または IICA1 でシングルマスタ・システムを選択した場合の注意事項の解除

シリアル・インタフェースIICA0またはIICA1をシングルマスタ・システムとして使用する場合の生成コードの誤りを修正しました。
RENESAS TOOL NEWS 資料番号 R20TS0459 をご参照ください。

3.2.3　CSI 設定で連続転送モードを選択した場合の注意事項の解除

CSIを連続転送モードで使用時、関数の引数に1を指定した場合に2バイト受信しないよう下記赤枠内のコードを修正しました。
RENESAS TOOL NEWS 資料番号 R20TS0244 をご参照ください。

3.2.4　R_CGC_Set_ClockMode()のクロック・モード選択(SELPLL)の誤りの修正

マクロの値がR_CGC_Set_ClockMode()で正しく生成されるようにコードを変更しました。

```
MD_STATUS R_CGC_Set_ClockMode(clock_mode_t mode)
{
    (省略)
    if (mode != old_mode)
    {
        switch (mode)
        {
            case HIOCLK:

                (省略)                                    _08_CGC_CLK_MODE_PLL
                if (_00_CGC_CLK_MODE_FMAIN != (PLLSTS & _00_CGC_CLK_MODE_FMAIN))
                {
                    SELPLL = 0U;    /* clock through mode (fMAIN) */
                }
```

3.2.5　CSI および UART のコールバック機能設定に関する注意事項

受信機能から送信機能に変更すると、CSI11で誤ったGUIを表示し、誤ったコールバック関数をロックする問題を修正しました。
RENESAS TOOL NEWS 資料番号 R20TS0545 をご参照ください。

3.2.6　UART の送信データ・レベル設定誤表記の修正

UARTの送信タブに表示される「受信データ・レベル設定」を「送信データ・レベル設定」に修正しました。

3.2.7　オンチップ・デバック設定のトレース機能を使用する場合の注意事項の解除

RL78/F15のオンチップトレース時にRAM領域を確保する必要がないデバイスを使用した場合、トレース機能、ホットプラグイン機能を選択できないように修正しました。
RENESAS TOOL NEWS 資料番号 R20TS0544 をご参照ください。

また、RL78/I1CのR5F10NLE、R5F10NME、R5F11TLEも、オンチップトレース時にRAM領域を確保する必要がないため、トレース機能を選択できないように修正しました。

3.2.8　時間キャプチャ入力機能使用時の RTCCRn（n = 0-2）設定の TCST ビットの制限を解除

「時間キャプチャ入力設定」の「RTCICn」（n = 0-2）選択時の生成コードの誤りを修正しました。
r_cg_rtc.hのコード修正

```
/* Time Capture Status (TCST) */
#define _00_RTC_EVENT0_NOTDETECTED                    (0x00U) /* no event is detected */
#define _04_RTC_EVENT0_DETECTED                       (0x04U) /* an event is detected */
```

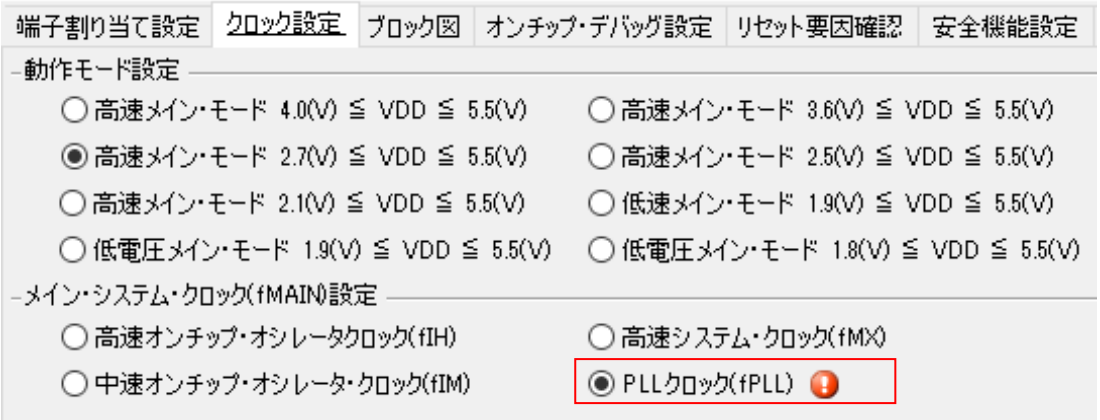
r_cg_rtc.cのR_RTC_Create（）のコード修正

```
RTCCR0 = _80_RTC_RTCICN0_ENABLE | _00_RTC_EVENT0_NOTDETECTED | _02_RTC_CAPTURE0_FALLING;
RTCCR1 = _00_RTC_RTCICN1_DISABLE | _00_RTC_EVENT1_NOTDETECTED;
RTCCR2 = _00_RTC_RTCICN2_DISABLE | _00_RTC_EVENT2_NOTDETECTED;
/* Set RTCIC0 pin */
PM15 |= 0x01U;
```

3.2.9　クロック設定の「動作モード」と「PLL 出力クロック(fPLL)」の誤った UI 操作を改善

「動作モード」と「PLL出力クロック（fPLL）」のUI動作を修正し、「PLL出力クロック（fPLL）」のエラー表示を改善しました。

- 「動作モード」を「高速メイン・モード2.5（V）≦VDD≦5.5（V）」から「高速メイン・モード4.0（V）≦VDD≦5.5（V）」に切り替えた場合、「PLL出力クロック（fPLL）」のエラーを解除し選択できるように改善しました。
- 「高速メイン・モード2.7（V）≤VDD≤5.5（V）」選択時、「PLL出力クロック（fPLL）」にエラーを表示させ、選択できないように改善しました。



3.2.10 R_CGC_Create()に fCLK 設定コードを追加

R_CGC_Create()のfCLK設定に関する生成コードを追加しました。

```
void R_CGC_Create(void)
{
    volatile uint32_t w_count;

    /* Set fMX */
    CMC = _00_CGC_HISYS_PORT | _10_CGC_SYSOSC_PERMITTED | _00_CGC_SYSOSC
    MSTOP = 1U;     /* X1 oscillator/external clock stopped */

    （省略）
    /* Set fSUB */
    SELLOSC = 0U;   /* sub clock (fSX) */
    /* Set fCLK */
    CSS = 0U;       /* main system clock (fMAIN) */
    /* Set fMAIN */
    MCM0 = 0U;      /* selects the main on-chip oscillator clock (fOCO)
```

3.2.11 R_CGC_Set_LPMode()と R_CGC_Set_LSMode()のコード・プレビューの改善

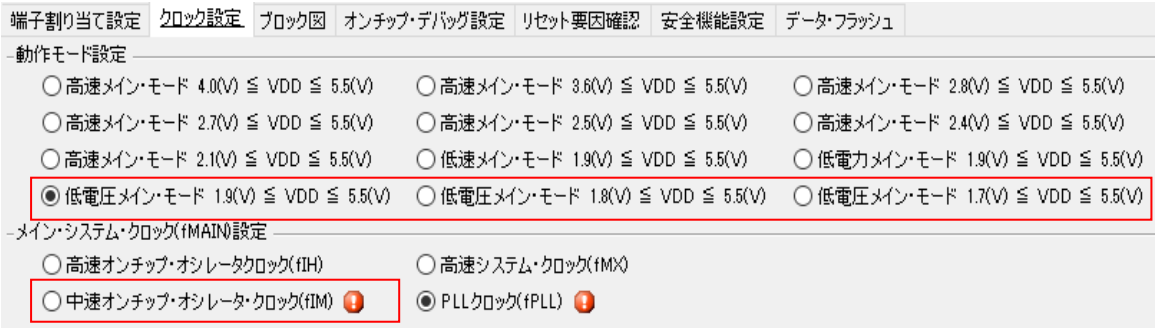
プロジェクト・ツリーのコード・プレビューで R_CGC_Set_LPMode() および R_CGC_Set_LSMode()を参照したときにコードが正しくプレビューできるように改善しました。

3.2.12 R_ELC_Stop()の改善

R_ELC_Stop（）の生成コードを修正しました。
修正前：sfr_addr = _04_ELC_EVENT_LINK_xxx;
修正後：sfr_addr = _00_ELC_EVENT_LINK_OFF;

3.2.13 低速メイン・モード選択時の中速オンチップ・オシレータ・クロックの改善

低電圧モードで中速オンチップ・オシレーターを選択できないようにUI操作を修正しました。



3.2.14 高速メイン・モード選択時の fMX 入力範囲の改善

「高速メイン・モード2.1（V）≦VDD≦5.5（V）」を選択した場合に1～6MHzを入力できるように改善しました。

3.2.15 r_lvd_vddinterrupt()生成条件の改善

電圧検出動作を選択したときに、r_lvd_vddinterrupt()が自動的に生成されないようにコード生成条件を修正しました。

3.2.16 EXLVD 設定を維持できない問題の改善

A/DモジュールがA/Dチャンネル選択として「ANI0」を設定した場合に、EXLVD設定が正しく読み込まれるように修正しました。

3.2.17 オプション・バイト設定値の誤りの修正

クロック・モニター機能を選択し、リセットモードとしてLVD（VLVD = 2.75V）を同時に選択した場合のOptionByte1値を修正しました。

3.2.18 ユーザーズマニュアル(HW)のサポートバージョン更新

RL78/I1C ユーザーズマニュアル(HW) V1.0、V2.0、V2.1に対応しました。 詳細については、ユーザーズマニュアル(HW)の「改訂履歴」を参照ください。

3.2.19 PIOR の特定設定値における ELC と DTC 生成コードの改善

PIORで特定の値を設定したときの生成コードを修正しました。

デバイス	コード修正の詳細
RL78I1C 64pin	PIOR 設定 PIOR04 INTP5 - PIOR04 INTP6 - PIOR04 INTP7 -
	r_cg_elc.c のコード修正： ➢ R_ELC_Create () ELSELR08 = _01_ELC_EVENT_LINK_AD; r_cg_dtc.c のコード修正： ➢ R_DTCD0_Start() DTCEN1 |= _40_DTC_KEY_ACTIVATION_ENABLE; ➢ R_DTCD0_Stop() DTCEN1 &= (uint8_t)~_40_DTC_KEY_ACTIVATION_ENABLE;
RL78I1C 80pin	PIOR 設定 PIOR04 INTP7 -
	r_cg_elc.c のコード修正： ➢ R_ELC_Create () ELSELR08 = _01_ELC_EVENT_LINK_AD;
	PIOR 設定 PIOR04 INTP7 P77
	r_cg_dtc.c のコード修正： ➢ R_DTCD0_Start() DTCEN1 |= _40_DTC_KEY_ACTIVATION_ENABLE; ➢ R_DTCD0_Stop() DTCEN1 &= (uint8_t)~_40_DTC_KEY_ACTIVATION_ENABLE;

3.2.20 fIH 選択を維持できない問題の改善

fIHを「8MHz」に選択し、再度CGCの設定画面に戻ったときに、fIHの選択を維持するようにGUIを修正しました。

3.2.21 IICA マスタのリスタートの改善

ストップ・コンディションが生成されていない場合に、IICAマスタがスタート・コンディションを再度生成する前に通信ステータスを判断するように改善しました。

1. R_IICA0_Master_Send()の場合

[変更前]

```
MD_STATUS R_IICA0_Master_Send(uint8_t adr, uint8_t * const tx_buf, uint16_t tx_num, uint8_t wait)
{
    MD_STATUS status = MD_OK;

    IICAMK0 = 1U;   /* disable INTIICA0 interrupt */

    if (1U == IICBSY0)
    {
        /* Check bus busy */
        IICAMK0 = 0U;   /* enable INTIICA0 interrupt */
        status = MD_ERROR1;
    }
    ......
```

[変更後]

```
MD_STATUS R_IICA0_Master_Send(uint8_t adr, uint8_t * const tx_buf, uint16_t tx_num, uint8_t wait)
{
    MD_STATUS status = MD_OK;

    IICAMK0 = 1U;   /* disable INTIICA0 interrupt */

    if ((1U == IICBSY0) && (0U == MSTS0))
    {
        /* Check bus busy */
        IICAMK0 = 0U;   /* enable INTIICA0 interrupt */
        status = MD_ERROR1;
    }
    ......
```

2. R_IICA0_Master_ Receive()の場合

[変更前]

```
MD_STATUS R_IICA0_Master_Receive(uint8_t adr, uint8_t * const rx_buf, uint16_t rx_num, uint8_t wait)
{
    MD_STATUS status = MD_OK;

    IICAMK0 = 1U;   /* disable INTIICA0 interrupt */

    if (1U == IICBSY0)
    {
        /* Check bus busy */
        IICAMK0 = 0U;   /* enable INTIICA0 interrupt */
        status = MD_ERROR1;
    }
    ......
```

[変更後]

```
MD_STATUS R_IICA0_Master_Receive(uint8_t adr, uint8_t * const rx_buf, uint16_t rx_num, uint8_t wait)
{
    MD_STATUS status = MD_OK;

    IICAMK0 = 1U;   /* disable INTIICA0 interrupt */

    if ((1U == IICBSY0) && (0U == MSTS0))
    {
        /* Check bus busy */
        IICAMK0 = 0U;   /* enable INTIICA0 interrupt */
        status = MD_ERROR1;
    }
    ......
```

3.2.22 簡易 IIC の Wait 処理の改善

スレーブ処理が間に合わない場合に備えて、Wait処理を追加しました。

1. R_IIC00_StartCondition()の場合
   赤枠箇所のコードを追加しました。

```
void R_IIC00_StartCondition(void)
{
    volatile uint8_t w_count;

    SO0 &= (uint16_t)~(_0001_SAUm_CH0_DATA_OUTPUT_1);

    /* Change the waiting time according to the system */
    for (w_count = 0U; w_count <= IIC00_WAITTIME; w_count++)
    {
        NOP();
    }

    SO0 &= (uint16_t)~(_0100_SAUm_CH0_CLOCK_OUTPUT_1);
    SOE0 |= _0001_SAUm_CH0_OUTPUT_ENABLE;
    SS0 |= _0001_SAUm_CH0_START_TRG_ON;

    for (w_count = 0U; w_count <= IIC00_WAITTIME; w_count++)
    {
        NOP();
    }
}
```

2. R_IIC00_StopCondition()の場合
   赤枠箇所のコードを追加しました。

```
void R_IIC00_StopCondition(void)
{
    volatile uint8_t w_count;

    ST0 |= _0001_SAUm_CH0_STOP_TRG_ON;
    SOE0 &= (uint16_t)~_0001_SAUm_CH0_OUTPUT_ENABLE;
    SO0 &= (uint16_t)~(_0001_SAUm_CH0_DATA_OUTPUT_1);

    for (w_count = 0U; w_count <= IIC00_WAITTIME; w_count++)
    {
        NOP();
    }

    SO0 |= _0100_SAUm_CH0_CLOCK_OUTPUT_1;

    /* Change the waiting time according to the system */
    for (w_count = 0U; w_count <= IIC00_WAITTIME; w_count++)
    {
        NOP();
    }

    SO0 |= _0001_SAUm_CH0_DATA_OUTPUT_1;

    for (w_count = 0U; w_count <= IIC00_WAITTIME; w_count++)
    {
        NOP();
    }
}
```

3. r_iic00_interrupt()の場合

[変更前]

```
__interrupt static void r_iic00_interrupt(void)
{
    if (((SSR00 & _0002_SAU_PARITY_ERROR) == 0x0002U) && (g_iic00_tx_count != 0U))
    {
        r_iic00_callback_master_error(MD_NACK);
    }
    else if(((SSR00 & _0001_SAU_OVERRUN_ERROR) == 0x0001U) && (g_iic00_tx_count != 0U))
    {
        r_iic00_callback_master_error(MD_OVERRUN);
    }
    else
    {
        /* Control for master send */
        if ((g_iic00_master_status_flag & _01_SAU_IIC_SEND_FLAG) == 1U)
        {
            if (g_iic00_tx_count > 0U)
            {
                SIO00 = *gp_iic00_tx_address;
                gp_iic00_tx_address++;
                g_iic00_tx_count--;
            }
            else
            {
                R_IIC00_StopCondition();
                r_iic00_callback_master_sendend();
            }
        }
        /* Control for master receive */
        else
        {

          （省略）
          else
          {
              if (g_iic00_rx_count < g_iic00_rx_length)
              {
                  *gp_iic00_rx_address = SIO00;
                  gp_iic00_rx_address++;
                  g_iic00_rx_count++;

                  if (g_iic00_rx_count == (g_iic00_rx_length - 1U))
                  {
                      SOE0 &= (uint16_t)~_0001_SAUm_CH0_OUTPUT_ENABLE;
                      SIO00 = 0xFFU;
                  }
                  else if (g_iic00_rx_count == g_iic00_rx_length)
                  {
                      R_IIC00_StopCondition();
                      r_iic00_callback_master_receiveend();
                  }
                  else
                  {
                      SIO00 = 0xFFU;
                  }
              }
          }
```

[変更後]

```
__interrupt static void r_iic00_interrupt(void)
{
    volatile uint16_t w_count;

    for (w_count = 0U; w_count <= IIC00_WAITTIME_2; w_count++)
    {
        NOP();
    }

    if (((SSR00 & _0002_SAU_PARITY_ERROR) == 0x0002U) && (g_iic00_tx_count != 0U))
    {
        r_iic00_callback_master_error(MD_NACK);
    }
    else if(((SSR00 & _0001_SAU_OVERRUN_ERROR) == 0x0001U) && (g_iic00_tx_count != 0U))
    {
        r_iic00_callback_master_error(MD_OVERRUN);
    }
    else
    {
        /* Control for master send */
        if ((g_iic00_master_status_flag & _01_SAU_IIC_SEND_FLAG) == 1U)
        {
            if (g_iic00_tx_count > 0U)
            {
                SIO00 = *gp_iic00_tx_address;
                gp_iic00_tx_address++;
                g_iic00_tx_count--;
            }
            else
            {
                /* IIC master transmission finishes and a callback function can be called here. */
                r_iic00_callback_master_sendend();
            }
        }
        /* Control for master receive */
        else
        {
```

（省略）

```
else
{
    if (g_iic00_rx_count < g_iic00_rx_length)
    {
        *gp_iic00_rx_address = SIO00;
        gp_iic00_rx_address++;
        g_iic00_rx_count++;

        if (g_iic00_rx_count == (g_iic00_rx_length - 1U))
        {
            SOE0 &= (uint16_t)~_0001_SAUm_CH0_OUTPUT_ENABLE;
            SIO00 = 0xFFU;
        }
        else if (g_iic00_rx_count == g_iic00_rx_length)
        {
            /* IIC master reception finishes and a callback function can be called here. */
            r_iic00_callback_master_receiveend();
        }
        else
        {
            SIO00 = 0xFFU;
        }
    }
}
```

4. In r_cg_serial.h の場合

[Before]

```
#define IIC00_WAITTIME      (13U) /* change the waiting time according to the system */
```

[After]

```
#define IIC00_WAITTIME      (13U) /* change the waiting time according to the system */
#define IIC00_WAITTIME_2    (53U) /* change the waiting time according to the system */
```

3.2.23 LVD オフのオプション・バイト設定値の改善

ユーザ・オプション・バイト(000C1H/010C1H)のLVD オフ設定値を改善しました。
RENESAS TOOL NEWS 資料番号 R20TS0571 をご参照ください。

4. RENESAS TOOL NEWS の改修履歴

RENESAS TOOL NEWS 注意事項の改修状況について記載します。

発行日	資料番号	概容	対象デバイス	改修バージョン
2012/05/21	120521/tn2	RL78/G13グループR5F1007xおよびR5F1017x用のコードを生成する場合の注意	RL78/G13	CS+ V1.00.06
2012/08/01	120801/tn3	RL78/G13およびRL78/G14グループ用コード生成を使用する場合の注意	RL78/G13, RL78/G14	CS+ V1.00.06
2012/09/01	120901/tn1	RL78/G12グループ用コード生成を使用する場合の注意	RL78/G12	CS+ V1.00.06
2013/02/01	130201/tn1	RL78/G14グループのコード生成を使用する場合の注意	RL78/G14	CS+ V2.00.00
2013/07/01	130701/tn1	編集したソース・コードが消える場合の注意事項	RL78/F12, RL78/F13, RL78/F14, RL78/F15, RL78/G10, RL78/G12, RL78/G13, RL78/G14, RL78/G1A, RL78/G1C, RL78/G1D, RL78/G1E, RL78/G1F, RL78/G1G, RL78/G1H, RL78/I1A, RL78/I1B, RL78/I1D, RL78/I1E, RL78/L12, RL78/L13, RL78/L1C	CS+ V2.11.00
		ポートが正しく設定されない場合の注意事項	RL78/G1A	CS+ V2.00.01
2013/08/01	130801/tn1	RL78/G12グループのコード生成を使用する場合の注意事項	RL78/G12	CS+ V2.00.01
2013/10/16	131016/tn1	2. RL78/G13 100pinパッケージ選択時の注意事項	RL78/G13	CS+ V2.03.00
		3. キー入力割り込み設定の注意事項	RL78/L12	CS+ V2.03.00
		4. A/Dコンバータ動作設定の注意事項	RL78/G1A	CS+ V2.03.00
		5. タイマKB20使用時の注意事項	RL78/L13	CS+ V2.03.00
2014/04/16	140416/tn5	1. RL78/F13およびRL78/F14グループで、20、30または32ピンパッケージを選択した時の注意事項	RL78/F13, RL78/F14	CS+ V2.04.00
		2. RL78/L12およびRL78/L13グループでの、リモコン搬送波マスク信号使用時の注意事項	RL78/L12, RL78/L13	CS+ V2.04.00
		3. RL78/G12グループでの、A/Dを設定した時の端子配置反映処理の注意事項	RL78/G12	CS+ V2.04.00
		4. RL78/G14グループで、マイコンに存在しないポートが表示される際の注意事項	RL78/G14	CS+ V2.04.00
		5. RL78/G1Eグループでの、シリアル・アレイ・ユニット1のUART2設定の注意事項	RL78/G1E	CS+ V2.04.00

発行日	資料番号	概容	対象デバイス	改修バージョン
2014/07/01	140701/tn1	Port2 設定時の注意事項	RL78/L13	CS+ V2.07.00
		インターバルタイマ設定時の注意事項	RL78/G10, RL78/G12, RL78/G13, RL78/G14, RL78/G1A, RL78/G1C, RL78/L12, RL78/L13, RL78/L1C, RL78/I1A	CS+ V2.07.00
2014/08/16	140816/tn1	Port2のP20およびP21設定時の注意事項	RL78/L1C	CS+ V2.05.00
		ポート1設定時の注意事項	RL78/G14	CS+ V2.05.00
2014/11/01	141101/tn2	1. CPUスタック・ポインタ・モニタ機能設定時の注意事項	RL78/F13	CS+ V2.07.00
		2. 3線シリアル使用時のSIRレジスタ書き込みの注意事項	RL78/F12	CS+ V2.07.00
2014/12/16	141216/tn3	1. コンパレータ設定時の生成コードの注意事項	RL78/I1A	CS+ V2.07.00
		2. DTC設定時の注意事項	RL78/F13, F14	CS+ V2.07.00
		3. 電圧検出回路で「割り込みモード」設定時の注意事項	RL78/L12, RL78/I1A, RL78/G1A, RL78/F13, RL78/F14	CS+ V2.07.00
		4. A/Dコンバータの設定を行ってプロジェクト保存するときの注意事項	RL78/L1C	CS+ V2.07.00
		5. コード生成で端子配置へ反映させた時の注意事項	RL78/G12, RL78/G13, RL78/G14	CS+ V2.07.00
2015/07/16	150716/tn2	1. クロック発生回路(PLL回路動作時)の注意事項	RL78/D1A, RL78/F13, RL78/F14, RL78/L1C, RL78/G1C	CS+ V2.11.00 AP4 V1.10.00 Applilet3 V1.10.00
		2. ポート4のP40 設定時の注意事項	RL78/D1A, RL78/F12, RL78/F13, RL78/F14, RL78/G12, RL78/G13, RL78/G14, RL78/G1A, RL78/I1A, RL78/L12, RL78/G10, RL78/G1C, RL78/G1E, RL78/G1F, RL78/G1G, RL78/I1B, RL78/I1D, RL78/L1C, RL78/L13	CS+ V2.11.00 AP4 V1.10.00 Applilet3 V1.10.00
		3. UART0 および UARTF 生成コードの注意事項	RL78/F12	CS+ V2.11.00 Applilet3 V1.10.00

発行日	資料番号	概容	対象デバイス	改修バージョン
2015/10/01	151001/tn3	シリアル・アレイ・ユニットの UART 通信時の注意事項	RL78/I1B	CS+ V2.11.00 AP4 V1.10.00
2015/11/16	151116/tn2	1. シリアル・インタフェース IICA のチャネル表示に関する注意事項	RL78/G14	CS+ V2.11.00 Applilet3 V1.10.00
		2. PLL クロックの設定手順に関する注意事項	RL78/F13, RL78/F14, RL78/F15	CS+ V2.11.00 Applilet3 V1.10.00
2016/01/16	160116/tn5	シリアル・アレイ・ユニット CSI および UART で 10 ビット以上のデータ長を送受信する時の注意事項	RL78/F12, RL78/F13, RL78/F14, RL78/F15, RL78/D1A	CS+ V2.11.00 Applilet3 V1.10.00
2016/02/16	160216/tn5	1. シリアル・アレイ・ユニット 4(UART4 または DALI4)のエラー割り込み使用時の 注意事項	RL78/I1A	CS+ V2.11.00 Applilet3 V1.10.00
		2. シリアル・アレイ・ユニット 4(DALI4)の注意事項	RL78/I1A	CS+ V2.11.00 Applilet3 V1.10.00
2016/03/16	160316/tn1	PIOR レジスタ設定によるピン配置変更時のシリアル・インタフェース IICA の端子設定 の注意事項	RL78/G12	CS+ V2.11.00 Applilet3 V1.10.00
2016/06/16	R20TS0038JJ0100	A/D コンバータのスキャン・モードに関する注意事項	RL78/F12, RL78/F13, RL78/F14, RL78/F15, RL78/G1A	CS+ V2.12.00 Applilet3 V1.11.00
2016/08/01	R20TS0045JJ0100	周辺 I/O リダイレクション・レジスタ 0 (PIOR0)に関する注意事項	RL78/G1F	CS+ V2.12.00 AP4 V1.11.00
2017/03/01	R20TS0139JJ0100	1. ポート P10, P11 の入力に関する注意事項	RL78/G13 (20/24/25pin製品)	CS+ V2.14.00 Applilet3 V1.13.00
		2. リセット処理に関するポート設定の注意事項	RL78/F12 (20pin製品)	CS+ V2.14.00 Applilet3 V1.13.00

発行日	資料番号	概容	対象デバイス	改修バージョン
2017/12/16	R20TS0244JJ0100	CSI 設定で連続転送モードを選択した場合の注意事項	RL78/D1A, RL78/F12, RL78/F13, RL78/F14, RL78/F15	CS+ V2.16.00 Applilet3 V1.15.00
			RL78/H1D, RL78/I1C, RL78/L12, RL/78L13	CS+ V2.20.00 AP4 V1.20.00 Applilet3 V1.20.00
2018/03/16	R20TS0290JJ0100	旧バージョンのコード生成を使用して作成した RL78/G11 用プロジェクトを開く場合の注意事項	RL78/G11 (20pin R5F1056A)	CS+ V2.16.00 AP4 V1.15.00
2018/05/16	R20TS0313JJ0100	未使用端子に対するポート関連レジスタの設定に関する注意事項	RL78/I1D	CS+ V2.16.00 AP4 V1.15.00
2018/11/16	R20TS0370JJ0100	シリアル UART4 を設定する時の注意事項	RL78/I1A	CS+ V2.17.00 Applilet3 V1.16.00
2019/06/01	R20TS0432JJ0100	1. クロック発生回路の PLL クロック設定についての注意事項	RL78/F13, RL78/F14, RL78/F15	CS+ V2.18.00 Applilet3 V1.17.00
		2. クロック発生回路の RTC 動作クロック設定についての注意事項	RL78/F13, RL78/F14, RL78/F15	CS+ V2.18.00 Applilet3 V1.17.00
			RL78/D1A	CS+ V2.19.00 Applilet3 V1.18.00
2019/08/01	R20TS0459JJ0100	1. IICA0 または IICA1 でシングルマスタ・システムを選択した場合の注意事項	RL78/G10, RL78/G11, RL78/G12, RL78/G13, RL78/G14, RL78/F12, RL78/F13, RL78/F14, RL78/F15	CS+ V2.19.00 AP4 V1.18.00 Applilet3 V1.18.00
			RL78/H1D, RL78/I1C, RL78/L12, RL78/L13	CS+ V2.20.00 AP4 V1.20.00 Applilet3 V1.20.00

発行日	資料番号	概容	対象デバイス	改修バージョン
2019/08/01	R20TS0459JJ0100	1. IICA0 または IICA1 でシングルマスタ・システムを選択した場合の注意事項	RL78/G1A, RL78/G1C, RL78/G1D, RL78/G1F, RL78/G1H, RL78/I1A, RL78/I1B, RL78/L1A, RL78/L1C	未対応
		2. A/D コンバータで R_ADC_Set_ADChannel() 関数を使用する場合の注意事項	RL78/D1A, RL78/G1A, RL78/G1F, RL78/I1D	CS+ V2.19.00 AP4 V1.18.00 Applilet3 V1.18.00
2019/09/16	R20TS0472JJ0100	1. データ・フラッシュ・ライブラリを使用時の注意事項	RL78/D1A, RL78/G11, RL78/G12, RL78/G13, RL78/G14, RL78/F12, RL78/F13, RL78/F14, RL78/F15	CS+ V2.19.00 AP4 V1.18.00 Applilet3 V1.18.00
2020/02/01	R20TS0545JJ0100	1. CSI および UART のコールバック機能設定に関する注意事項 2. CSI および UART の端子機能割り当てが解除される操作の注意事項	RL78/G11	CS+ V2.20.00 AP4 V1.20.00
			RL78/I1E	未対応
2020/02/01	R20TS0544JJ0100	1. オンチップ・デバック設定のトレース機能を使用する場合の注意事項	RL78/F15	CS+ V2.20.00 Applilet3 V1.20.00
2020/05/16	R20TS0571JJ0100	1. ユーザ・オプション・バイト (000C1H/010C1H)LVD オフ設定値の注意事項	RL78/G13A	CS+ V2.21.00 Applilet3 V1.20.00

5. 制限事項

RL78 コード生成ツールの制限事項について説明します。

5.1 制限事項一覧

表 5-1 制限事項一覧 1/2

○：対象項目，/：非対象項目

No	内容	グループ													
		RL78/D1A	RL78/G10	RL78/G11	RL78/G12	RL78/G13, G13A	RL78/G14	RL78/G1A	RL78/G1C	RL78/G1D	RL78/G1E	RL78/G1F	RL78/G1G	RL78/G1H	RL78/H1D
1	タイマ・アレイ・ユニット入力ソースについて	/	/	/	/	/	/	/	/	/	/	/	/	/	/
2	24 ピンデバイスの TAU0 チャネル 1 設定の制限について	/	/	/	/	/	/	/	/	/	/	○	/	/	/
3	リアルタイム・クロックの API 関数について	/	/	/	/	/	/	/	/	/	/	○	/	○	/
4	PGA+ΔΣA/D コンバータ ゲイン設定の単位の表記ミスについて	/	/	/	/	/	/	/	/	/	/	/	/	/	○
5	UART の送信データ・レベル設定誤表記について	/	/	○	/	/	/	/	/	/	/	/	○	○	○
6	CSI の連続転送モード時の制限について	○	/	/	/	/	/	○	○	○	○	○	○	○	/
7	データ・フラッシュ・ライブラリの関数説明の誤りについて	/	/	○	○	○	○	/	/	/	/	/	/	/	/
8	TAU の入力パルス間隔測定で fSUB、fIL を選択した場合の動作クロックについて	/	/	/	/	○	○	/	/	/	/	/	/	/	/
9	e² studio 64 ビット環境の制限について	○	/	/	/	/	/	/	/	/	/	/	/	/	/
10	R_ELC_Stop()関数の誤りについて	/	/	/	/	/	/	/	/	/	/	○	○	○	/
11	トレース・アドレスの誤りについて	/	/	/	/	/	/	/	/	/	/	○	/	/	/
12	IAR Embedded Workbench の R_ELC_Stop()関数ビルド・エラーについて	/	/	/	/	/	○	/	/	/	/	/	/	/	/
13	CS+および e² studio のデバイス変更機能に関する制限	/	/	/	/	/	/	/	/	/	/	/	/	/	/

表 5-2 制限事項一覧 2/2

○：対象項目，/：非対象項目

No	内容	グループ													
		RL78/F12	RL78/F13	RL78/F14	RL78/F15	RL78/F1E	RL78/I1A	RL78/I1B	RL78/I1C	RL78/I1D	RL78/I1E	RL78/L12	RL78/L13	RL78/L1A	RL78/L1C
1	タイマ・アレイ・ユニット入力ソースについて	○	○	○	/	/	/	/	/	/	/	/	/	/	/
2	24 ピンデバイスの TAU0 チャネル 1 設定の制限について	/	/	/	/	/	/	/	/	/	/	/	/	/	/
3	リアルタイム・クロックの API 関数について	/	/	/	/	/	/	/	/	/	○	/	/	/	/
4	PGA+ΔΣA/D コンバータ ゲイン設定の単位の表記ミスについて	/	/	/	/	/	/	/	/	/	/	/	/	/	/
5	UART の送信データ・レベル設定誤表記について	/	/	/	/	/	/	/	○	○	○	/	/	○	/
6	CSI の連続転送モード時の制限について	/	/	/	/	/	○	○	/	○	○	/	/	○	○
7	データ・フラッシュ・ライブラリの関数説明の誤りについて	○	○	○	○	/	/	/	/	/	○	/	/	/	/
8	TAU の入力パルス間隔測定で fSUB、fIL を選択した場合の動作クロックについて	○	/	/	/	/	/	/	/	/	/	/	/	/	/
9	e² studio 64 ビット環境の制限について	/	/	/	/	/	/	/	/	/	/	/	/	/	/
10	R_ELC_Stop()関数の誤りについて	/	/	/	/	/	/	/	/	/	/	○	○	○	/
11	トレース・アドレスの誤りについて	/	/	/	/	/	/	/	/	/	/	/	/	/	/
12	IAR Embedded Workbench の R_ELC_Stop() 関数ビルド・エラーについて	/	/	/	/	/	/	/	/	/	/	/	/	/	/
13	CS+および e² studio のデバイス変更機能に関する制限	/	/	/	/	/	/	/	/	/	/	/	/	/	○

5.2　制限事項詳細

5.2.1　タイマ・アレイ・ユニット入力ソースについて

タイマ・アレイ・ユニットの設定で、タイマ入力のクロック・ソースをRTC1HZ出力信号に設定すると、リアルタイム・クロックの設定でRTC1HZ端子の出力に関する設定が無効となり、RTC1HZを出力するコードは生成されません。

【回避策】　タイマ・アレイ・ユニットの設定で RTC1HZ 信号に設定した場合は、リアルタイム・クロックを使用する設定を選択しつつ、RTC1HZ を出力するコードを追加してください。

5.2.2　24 ピンデバイスの TAU0 チャネル 1 設定の制限について

24ピンデバイスでTAU0チャネル1の設定がインターバルタイマしか選択できません。

【回避策】　ありません。申し訳ありませんが、32 ピンデバイスでは、TAU0 チャネル 1 の設定が「インターバルタイマ」以外でも選択できます。その設定を参考に修正してください。

5.2.3　リアルタイム・クロックの API 関数について

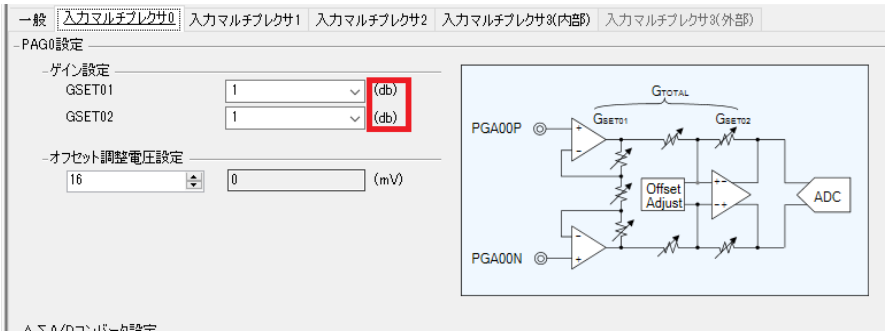
R_RTC_Set_AlarmOn()関数内に不要な待ち時間コードが出力されます。

```
/* Change the waiting time according to the system */
 for (w_count = 0U; w_count < RTC_WAITTIME_2FRTC; w_count++)
 {
      NOP();
 }
```

【回避策】　ありません。コード生成を行った後に R_RTC_Set_AlarmOn()関数にある待ち時間のコードを削除してください。

5.2.4　PGA+ΔΣA/D コンバータ　ゲイン設定の単位の表記ミスについて

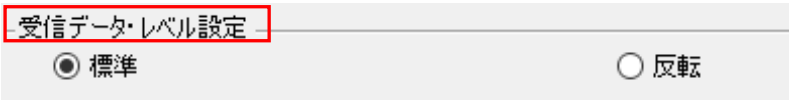
入力マルチプレクサ0/1/2/3(内部)/3(外部)のPAG0設定のゲイン設定の単位が'db'となっておりますが、これは'Gain'（倍）の間違いです。



【回避策】　GSET01 および、GSET02 の指定時に'db'を'Gain'(倍)と読み替えてください。

5.2.5　UART の送信データ・レベル設定誤表記について

UARTの送信タブに表示される「受信データ・レベル設定」は、「送信データ・レベル設定」の誤りです。



【回避策】　　「送信データ・レベル設定」と読み替えてください。

5.2.6　CSI の連続転送モード時の制限について

CSIを連続転送モードで使用時、関数の引数に1を指定しても2バイト受信してしまいます。

【回避策】　　下記赤枠内のコードを変更してください。
コード変更後に再度コード生成を実行すると、コードが上書きされ削除されますので、ご注意ください。

[R_CSIn_Receive 関数]　CSI00 の場合

変更前

```
MD_STATUS R_CSI00_Receive(uint8_t * const rx_buf, uint16_t rx_num)
{
    MD_STATUS status = MD_OK;

    if (rx_num < 1U)
    {
        status = MD_ARGERROR;
    }
    else
    {
        SMR00 |= _0001_SAU_BUFFER_EMPTY;
        g_csi00_rx_length = rx_num;     /* receive data length */
        g_csi00_rx_count = 0U;          /* receive data count */
        gp_csi00_rx_address = rx_buf;   /* receive buffer pointer */
        SIO00 = 0xFFU;     /* start receive by dummy write */
    }

    return (status);
}
```

変更後

```
MD_STATUS R_CSI00_Receive(uint8_t * const rx_buf, uint16_t rx_num)
{
    MD_STATUS status = MD_OK;

    if (rx_num < 1U)
    {
        status = MD_ARGERROR;
    }
    else
    {
        if ( 1U == rx_num )
        {
            SMR00 &= ~_0001_SAU_BUFFER_EMPTY;
        }
        else
        {
            SMR00 |= _0001_SAU_BUFFER_EMPTY;
        }
        g_csi00_rx_length = rx_num;     /* receive data length */
        g_csi00_rx_count = 0U;          /* receive data count */
        gp_csi00_rx_address = rx_buf;   /* receive buffer pointer */
        SIO00 = 0xFFU;     /* start receive by dummy write */
    }

    return (status);
}
```

[R_CSIn_Send_Receive 関数]　CSI00 の場合

変更前

```
MD_STATUS R_CSI00_Send_Receive(uint8_t * const tx_buf, uint16_t tx_num, uint8_t * const rx_buf)
{
    MD_STATUS status = MD_OK;

    if (tx_num < 1U)
    {
        status = MD_ARGERROR;
    }
    else
    {
        g_csi00_send_length = tx_num;     /* send data length */
        g_csi00_tx_count = tx_num;        /* send data count */
        gp_csi00_tx_address = tx_buf;     /* send buffer pointer */
        gp_csi00_rx_address = rx_buf;     /* receive buffer pointer */
        SMR00 |= _0001_SAU_BUFFER_EMPTY;
        CSIMK00 = 1U;                     /* disable INTCSI00 interrupt */

        if (0U != gp_csi00_tx_address)
        {
            SIO00 = *gp_csi00_tx_address;    /* started by writing data to SDR[7:0] */
            gp_csi00_tx_address++;
        }
        else
        {
            SIO00 = 0xFFU;
        }

        g_csi00_tx_count--;
        CSIMK00 = 0U;                     /* enable INTCSI00 interrupt */
    }

    return (status);
}
```

変更後

```
MD_STATUS R_CSI00_Send_Receive(uint8_t * const tx_buf, uint16_t tx_num, uint8_t * const rx_buf)
{
    MD_STATUS status = MD_OK;

    if (tx_num < 1U)
    {
        status = MD_ARGERROR;
    }
    else
    {
        g_csi00_send_length = tx_num;     /* send data length */
        g_csi00_tx_count = tx_num;        /* send data count */
        gp_csi00_tx_address = tx_buf;     /* send buffer pointer */
        gp_csi00_rx_address = rx_buf;     /* receive buffer pointer */
        if ( 1U == tx_num )
        {
            SMR00 &= ~_0001_SAU_BUFFER_EMPTY;
        }
        else
        {
            SMR00 |= _0001_SAU_BUFFER_EMPTY;
        }

        CSIMK00 = 1U;                     /* disable INTCSI00 interrupt */

        if (0U != gp_csi00_tx_address)
        {
            SIO00 = *gp_csi00_tx_address;    /* started by writing data to SDR[7:0] */
            gp_csi00_tx_address++;
        }
        else
        {
            SIO00 = 0xFFU;
        }

        g_csi00_tx_count--;
        CSIMK00 = 0U;                     /* enable INTCSI00 interrupt */
    }

    return (status);
}
```

5.2.7　データ・フラッシュ・ライブラリの関数説明の誤りについて

R_FDL_BlankCheck()関数とR_FDL_Iverify()関数の説明に誤りがあります。
コード生成ツール RL78 APIリファレンス編 (P741) の記載が正しい説明となりますので、そちらを参照ください。

【回避策】　コードには影響ございません。

5.2.8　TAU の入力パルス間隔測定で fSUB、fIL 選択時の動作クロックについて

TAUで入力パルス間隔測定を指定しfSUB、fILを選択した場合、動作クロックがfclk/2^8に固定されます。動作クロックの固定により、安全機能の周波数検出で意図した検出精度にならない場合があります。

【回避策】　コード生成後、タイマ・モード・レジスタの動作クロック(f_{MCK})を CK00 から CK01 に変更してください。コード変更後に再度コード生成を実行すると、コードが上書きされ削除されますので、ご注意ください。

5.2.9　e^2 studio 64 ビット環境の制限について

32ビット環境で保存されたプロジェクト（e^2 studioプロジェクトまたはAP3プロジェクト）を読み込むと、TAU0チャネル1入力選択（以下の図の赤枠を参照）を保持できません。

【回避策】　プロジェクトを読み込んだ後、赤枠内の設定を確認してください。

5.2.10 R_ELC_Stop()関数の誤りについて

API R_ELC_Stop()関数で以下の誤ったコードが生成されます。

【回避策】　以下の赤枠コードを変更してください。 コード変更後に再度コード生成を実行すると、コードが上書きされて削除されますのでご注意ください。

【変更前】

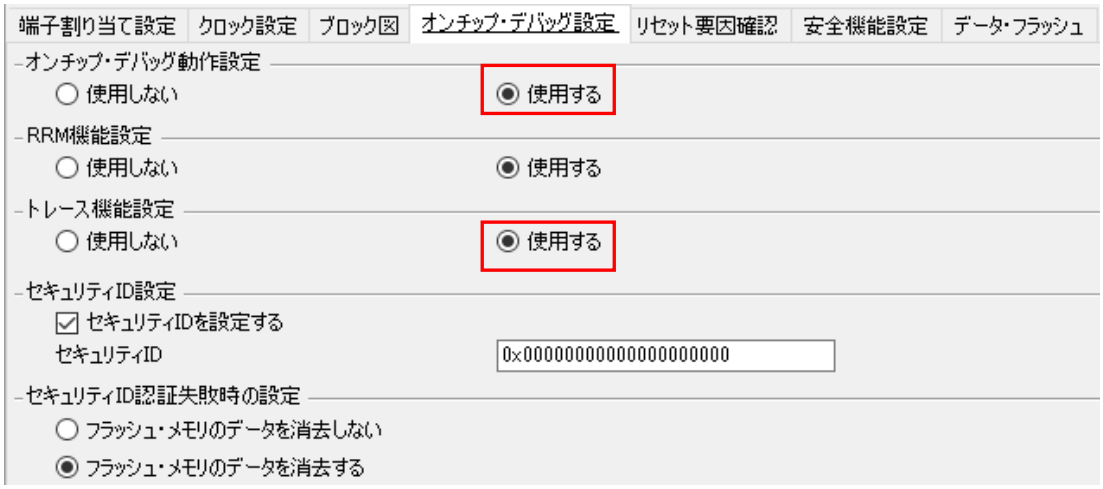
```
void R_ELC_Stop(uint32_t event)
{
    volatile uint32_t                   w_count;
    volatile uint8_t __no_bit_access * sfr_addr;

    sfr_addr = &ELSELR00;

    for (w_count = 0U; w_count < ELC_DESTINATION_COUNT; w_count++)
    {
        if (0x1U == ((event >> w_count) & 0x1U))
        {
            *sfr_addr = _04_ELC_EVENT_LINK_TAU05;
        }

        sfr_addr++;
    }
}
```

【変更後】

```
void R_ELC_Stop(uint32_t event)
{
    volatile uint32_t                   w_count;
    volatile uint8_t __no_bit_access * sfr_addr;

    sfr_addr = &ELSELR00;

    for (w_count = 0U; w_count < ELC_DESTINATION_COUNT; w_count++)
    {
        if (0x1U == ((event >> w_count) & 0x1U))
        {
            *sfr_addr = _00_ELC_EVENT_LINK_OFF;
        }

        sfr_addr++;
    }
}
```

5.2.11 トレース・アドレスの誤りについて

IARコンパイラを選択し、オンチップ・デバッグ設定のトレース機能を使用する場合、誤ったトレース・アドレスが生成されます。



【回避策】　以下の赤枠コードを変更してください。 コード変更後に再度コード生成を実行すると、コードが上書きされて削除されますのでご注意ください。

【変更前】

```
/* Secure trace RAM area */
__no_init __root unsigned char ocdtraceram[1024] @ 0xFE300;
```

【変更後】

```
/* Secure trace RAM area */
__no_init __root unsigned char ocdtraceram[1024] @ 0xFED00;
```

5.2.12 IAR Embedded Workbench の R_ELC_Stop()関数ビルド・エラーについて

RL78/G14のIARプロジェクトで、ELC関数を使用すると、次の図のようにビルド・エラーが発生します。

Error[Pe513]: a value of type "unsigned char volatile __no_bit_access *" cannot be assigned to an entity of type "uint8_t volatile *"

【回避策】　以下の赤枠コードを追加してください。 コード変更後に再度コード生成を実行すると、コードが上書きされて削除されますのでご注意ください。

```
void R_ELC_Stop(uint32_t event)
{
    volatile uint32_t   w_count;
    volatile uint8_t __no_bit_access * sfr_addr;

    sfr_addr = &ELSELR00;

    for (w_count = 0U; w_count < ELC_DESTINATION_COUNT; w_count++)
    {
        if (((event >> w_count) & 0x1U) == 0x1U)
        {
            *sfr_addr = _00_ELC_EVENT_LINK_OFF;
        }

        sfr_addr++;
    }
}
```

5.2.13 CS+および e² studio のデバイス変更機能に関する制限

CS+およびe² studioのデバイス変更機能使用時、以下のデバイスには既知の問題があるため、正常に変更できない可能性があります。

- CS+の [マイクロコントローラーの変更] 機能を使用すると、デバイスの変更に失敗し、エラーが表示されます。
- e² studioの [Change device] 機能を使用すると、変更後 [コード生成] ノードがプロジェクト・ツリーから消えます。

変更前	変更後
R5F10NME(80pin)	R5F10NML(80pin), R5F10NML(Dual) (80pin)
R5F10NMG(80pin)	R5F10NML(80pin) , R5F10NML(Dual) (80pin)
R5F10NMJ(80pin)	R5F10NML(80pin) , R5F10NML(Dual) (80pin)
R5F10NML(80pin)	R5F10NME(80pin), R5F10NMG(80pin), R5F10NMJ(80pin)
R5F10NML(Dual) (80pin)	R5F10NME(80pin), R5F10NMG(80pin), R5F10NMJ(80pin)
R5F10NPG(100pin)	R5F10NPL(100pin), R5F10NPL(Dual) (100pin)
R5F10NPJ(100pin)	R5F10NPL(100pin), R5F10NPL(Dual) (100pin)
R5F10NPL(100pin)	R5F10NPG(100pin), R5F10NPJ(100pin)
R5F10NPL(Dual) (100pin)	R5F10NPG(100pin), R5F10NPJ(100pin)

【回避策】　デバイス変更機能を使用する場合は、上記表のデバイスを選択しないでください。

6. 注意事項

RL78 コード生成の注意事項について説明します。

6.1 注意事項一覧

表 6-1 注意事項一覧

○：対象項目，/：非対象項目

No	内容	グループ													
		RL78/D1A	RL78/G10	RL78/G11	RL78/G12	RL78/G13, G13A	RL78/G14	RL78/G1A	RL78/G1C	RL78/G1D	RL78/G1E	RL78/G1F	RL78/G1G	RL78/G1H	RL78/H1D
1	オンラインヘルプについて (Applilet3, AP4)	○	○	○	○	○	○	○	○	○	○	○	○	○	○
2	MISRA-C のコーディング規約対応について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
3	高速オンチップ･オシレータ周波数選択レジスタについて	○	○	○	○	○	○	○	○	○	○	○	○	○	○
4	高速及び低速内蔵発振器精度トリミングレジスタについて	○	○	○	○	○	○	○	○	○	○	○	○	○	○
5	シリアル・アレイ・ユニットの制限について	/	/	/	/	/	/	/	/	/	/	/	/	/	/
6	フラッシュ・メモリ CRC 演算機能（高速 CRC）について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
7	ポート・モード選択レジスタ（PMS）について	○	○	/	/	○	/	○	○	○	○	○	○	○	○
8	UART の LIN-bus 機能について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
9	IICA の拡張コード、マルチマスタ、ウエイク・アップ機能および簡易 IIC のマルチマスタ機能について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
10	CAN コントローラ機能について	/	/	/	/	/	/	/	/	/	/	/	/	/	/
11	安全機能について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
12	USB 機能について	/	/	/	/	/	/	/	/	/	/	/	/	/	/
13	RI78V4 プロジェクトについて	/	/	/	/	/	/	/	/	/	/	/	/	/	/
14	DTC 機能を使用する時の注意 (CS+ for CA,CX)	○	/	/	/	○	/	/	/	/	/	/	/	/	/
15	高速 DTC のチェイン転送について	/	/	/	/	/	/	/	/	/	/	/	/	/	/
16	IICA のスレーブ時のファスト・モード・プラス設定について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
17	高速オンチップ・オシレータ設定について (CS+ for CA,CX)	/	/	○	○	○	○	/	/	/	/	/	/	/	/
18	端子配置ツールについて(CS+ for CA,CX)	/	/	○	○	○	○	/	/	/	/	/	/	/	/
19	RL78/G13A の生成ファイルの Version 表記について	/	/	/	/	○	/	/	/	/	/	/	/	/	/
20	簡易 IIC ストップ・コンディションの生成について	○	○	○	○	○	○	○	○	○	○	○	○	/	○
21	デバイス変更機能について	○	○	○	○	○	○	○	○	○	○	○	○	○	○

表 6-1 注意事項一覧

○：対象項目，/：非対象項目

No	内容	グループ													
		RL78/F12	RL78/F13	RL78/F14	RL78/F15	RL78/F1E	RL78/I1A	RL78/I1B	RL78/I1C	RL78/I1D	RL78/I1E	RL78/L12	RL78/L13	RL78/L1A	RL78/L1C
1	オンラインヘルプについて (Applilet3, AP4)	○	○	○	○	○	○	○	○	○	○	○	○	○	○
2	MISRA-C のコーディング規約対応について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
3	高速オンチップ･オシレータ周波数選択レジスタについて	○	○	○	○	○	/	○	○	○	○	○	○	○	○
4	高速及び低速内蔵発振器精度トリミングレジスタについて	○	○	○	○	○	○	○	○	○	○	○	○	○	○
5	シリアル・アレイ・ユニットの制限について	/	/	/	/	/	○	/	/	/	/	/	/	/	/
6	フラッシュ・メモリ CRC 演算機能（高速 CRC）について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
7	ポート・モード選択レジスタ（PMS）について	/	○	○	○	○	○	○	○	○	○	/	○	○	○
8	UART の LIN-bus 機能について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
9	IICA の拡張コード、マルチマスタ、ウエイク・アップ機能および簡易 IIC のマルチマスタ機能について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
10	CAN コントローラ機能について	/	○	○	○	○	/	/	/	/	/	/	/	/	/
11	安全機能について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
12	USB 機能について	/	/	/	/	/	/	/	/	/	/	/	/	/	○
13	RI78V4 プロジェクトについて	/	/	/	/	/	/	/	/	/	/	/	/	/	/
14	DTC機能を使用する時の注意 (CS+ for CA,CX)	/	○	○	○	○	/	/	/	/	/	/	/	○	/
15	高速 DTC のチェイン転送について	/	○	○	○	/	/	/	/	/	/	/	/	/	/
16	IICA のスレーブ時のファスト・モード・プラス設定について	○	○	○	○	○	○	○	○	○	○	○	○	○	○
17	高速オンチップ・オシレータ設定について (CS+ for CA,CX)	○	○	○	/	/	○	/	/	/	/	○	/	/	/
18	端子配置ツールについて(CS+ for CA,CX)	○	○	○	○	/	○	/	/	/	/	○	/	/	/
19	RL78/G13A の生成ファイルの Version 表記について	/	/	/	/	/	/	/	/	/	/	/	/	/	/
20	簡易 IIC ストップ・コンディションの生成について	○	○	○	○	○	/	○	○	○	○	/	○	○	○
21	デバイス変更機能について	○	○	○	○	○	○	○	○	○	○	○	○	○	○

6.2　注意事項詳細

6.2.1　オンラインヘルプについて (Applilet3, AP4)

オンラインヘルプ機能に対応していません。

6.2.2　MISRA-C のコーディング規約対応について

コード生成から出力されるソース・コードは，自動車向け組み込みC言語用ガイドラインMISRA-Cのコーディング規約に対応していません。

6.2.3　高速オンチップ・オシレータ周波数選択レジスタ(HOCODIV)について

コード生成は、高速オンチップ・オシレータ周波数選択レジスタの設定に対応していません。

6.2.4　高速及び低速内蔵発振器精度トリミングレジスタについて

コード生成は、高速及び低速内蔵発振器精度トリミングレジスタの設定に対応していません。

6.2.5　シリアル・アレイ・ユニットの制限について

コード生成は、1線UARTモード及びDMX512通信の設定に対応していません。

6.2.6　フラッシュ・メモリ CRC 演算機能（高速 CRC）について

コード生成は、フラッシュ・メモリCRC演算機能（高速CRC）に対応していません。アプリケーションノート r01an0736を参考にしてください。

https://www.renesas.com/document/apn/rl78g13-safety-function-flash-memory-crc-operation-function

6.2.7　ポート・モード選択レジスタ（PMS）について

コード生成は、ポート・モード選択レジスタ（PMS）に対応していません。

6.2.8　UART の LIN-bus 機能について

コード生成は、シリアル・インタフェースUART(0, 2, 3, 6,F)のLIN- bus機能に対応していません。

6.2.9　IICA の拡張コード、マルチマスタ、ウエイク・アップ機能および簡易 IIC のマルチマスタ機能について

コード生成は、シリアル・インタフェースIICAの拡張コード、マルチマスタ、ウエイク・アップ機能および簡易IICのマルチマスタ機能に対応していません。

6.2.10 CAN コントローラ機能について

コード生成は，CAN機能に対応していません。

6.2.11 安全機能について

安全機能のRAMパリティ・エラー検出機能に対応していません。

6.2.12 USB 機能について

USBホスト、ファンクションモジュール機能に対応していません。

6.2.13 RI78V4 プロジェクトについて

RI78V4のプロジェクトで、本来使えないはずのコード生成が表示されます。コード生成を行っても出力コードが未サポートのため、ビルド・エラーとなります。

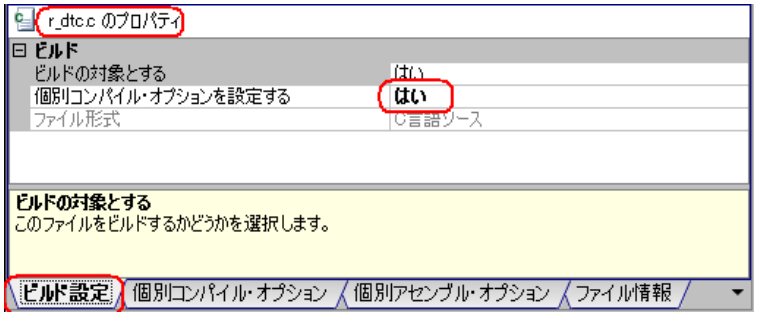
6.2.14 DTC 機能を使用する時の注意(CS+ for CA,CX)

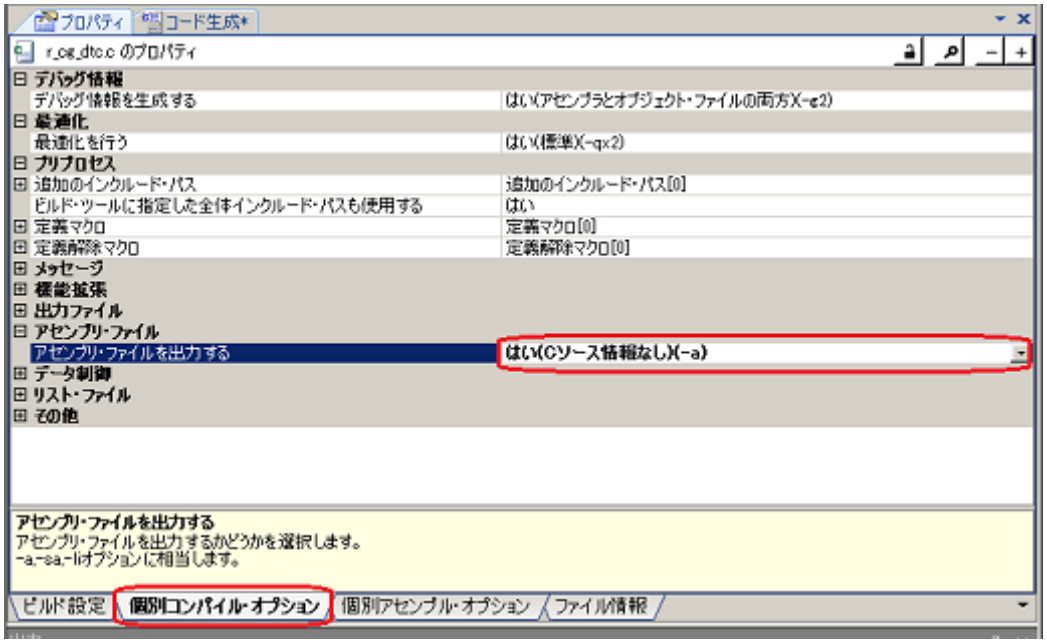
DTCを使用する場合、下記のワーニングが表示されオブジェクト・ファイルが生成されません。

CC78K0R warning W0837: Output assembler source file, not object file

【回避策】

ビルド時に下記の個別オプションを設定してください。





6.2.15 高速 DTC のチェイン転送について

高速DTCのチェイン転送の設定項目がありますが、チェイン転送に対応したコードをサポートしていません。

【回避策】　　チェイン転送に使用しないでください。

6.2.16 IICA のスレーブ時のファスト・モード・プラス設定について

IICAのスレーブ使用時にファスト・モード・プラスを設定した場合、IICAロウ・レベル幅設定レジスタ(IICWLn, nはチャネル番号)、IICAハイ・レベル幅設定レジスタ(IICWHLn)が正しく設定されません。

【回避策】　　ありません。コード生成を行った後に R_IICAn_Create()関数にある IICWLn, IICWHn のレジスタ設定の数値を書き換えてください。数値はシステムに依存します。デバイス UM を参考に変更してください。

6.2.17 高速オンチップ・オシレータの設定について(CS+ for CA,CX)

CubeSuite+ RL78,78K0R,78K0コード生成V2.01.00より前のバージョンで、高速オンチップ・オシレータ・クロックを設定した場合、それをCubeSuite+V2.03.00で読み込むと高速オンチップ・オシレータの動作周波数設定が正しくない時があります。

【回避策】　　正しい周波数を再設定してください。

6.2.18 端子配置ツールについて(CS+ for CA,CX)

コード生成から端子配置への反映を実行しても反映されない端子があります。
また、端子割り当て設定で端子割り当てを変更(PIOR機能)しても、端子配置へ反映されない端子があります。

【回避策】　　端子配置ツールで端子情報を編集してください。

6.2.19 RL78/G13A の生成ファイルの Version 表記について

RL78/G13Aで生成されるファイルのVersionに表記されるデバイス名は、「RL78/G13A」ではなく「RL78/G13」と表記されます。

```
/*****************************************************************************
* File Name     : r_main.c
* Version       : CodeGenerator for RL78/G13 V2.05.04.02 [20 Nov 2019]
* Device(s)     : R5F140PL
* Tool-Chain    : CCRL
* Description   : This file implements main function.
* Creation Date: 2019/11/27
*****************************************************************************/
```

6.2.20 簡易 IIC ストップ・コンディションの生成について

簡易IIC (マスタ) を使用してデータを送信し、すべてのデータがスレーブに送信された後、ストップ・コンディションを生成し、バスを解放する必要があります。そうしないと、後続のデータ送信を正しく開始できません。しかし、スレーブ処理がいつ完了するかを知ることができないため、コード生成で生成されたコードには、ストップ・コンディションはありません。

【回避策】　スレーブ処理の完了を確認し、main() 関数でストップ・コンディションを設定する必要があります。

【参考例】

```
void main(void)
{
    ....
    R_IICA0_Slave_Receive(IICA_rx_buff,5U);
    R_IIC00_Master_Send(16U, IIC0_tx_buff, 5U);
    while(IICA_Receive_flag != 1U);//After IICA0 receive all data, IICA_Receive_flag will become 1.
    R_IIC00_StopCondition();//Trigger stop condition signal.
    ....
}
```

6.2.21 デバイス変更機能について

コード生成は、同じグループおよび同じピン数デバイス内でのみデバイス変更機能をサポートします。
変更前後に選択したデバイスが同じグループにあり、ピン数が同じであれば、現在の設定でコード生成を正常に変更できます。
それ以外の場合、デバイス変更後にコード生成の設定は初期状態に復元されます。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2019.10.08	-	新規作成
1.01	2020.01.20	7	「2. デバイス一覧」を 2020/01/20 リリース向けに更新
		11	「3. 変更内容」を 2020/01/20 リリース向けに更新
		21	「4. RENESAS TOOL NEWS の改修履歴」を 2020/01/20 リリース向けに更新
		26	「5. 制限事項」を 2020/01/20 リリース向けに更新
		36	「6. 注意事項」を 2020/01/20 リリース向けに更新
1.02	2020.04.20	3	「1.1 製品バージョン」を 2020/01/20 リリース向けに更新
		4	「1.2.2 開発ツール」に e² studio 64-bit 環境を追加
		10	「3. 変更内容」を 2020/01/20 リリース向けに更新
		16	「5. 制限事項」を 2020/01/20 リリース向けに更新
1.03	2020.10.06	4	「1.1 製品バージョン」を 2020/10/06 リリース向けに更新
		6	「2. デバイス一覧」を 2020/10/06 リリース向けに更新
		11	「3. 変更内容」を 2020/10/06 リリース向けに更新
		19	「4. RENESAS TOOL NEWS の改修履歴」を 2020/10/06 リリース向けに更新
		24	「5. 制限事項」を 2020/10/06 リリース向けに更新
		34	「6.2.9」にリスタートを追記
1.04	2021.01.20	4	「1.1 製品バージョン」を更新
		5	「1.2.2 開発ツール」を更新
		9	「2. デバイス一覧」を更新
		11	「3. 変更点」を更新
		23	「4. RENESAS TOOL NEWS の改修履歴」を更新
		28	「5. 制限事項」を更新
		38	「6. 注意事項」を更新
1.05	2021.10.15	28	「5.1. 制限事項一覧」を更新
		36	「5.2. 制限事項詳細」を更新
		37	「6.1. 注意事項一覧」を更新
		43	「6.2. 注意事項詳細」を更新

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 静電気対策

   CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

2. 電源投入時の処置

   電源投入時は、製品の状態は不定です。電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. 電源オフ時における入力信号

   当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

   未使用端子は、「未使用端子の処理」に従って処理してください。CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。

5. クロックについて

   リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

6. 入力端子の印加波形

   入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、V_{IL}（Max.）から V_{IH}（Min.）までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、V_{IL}（Max.）から V_{IH}（Min.）までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

   リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

8. 製品間の相違について

   型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違うと、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。以下同じです。）に関し、当社は、一切その責任を負いません。
2. 当社製品、本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。

   標準水準：　コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等

   高品質水準：輸送機器（自動車、電車、船舶等）、交通制御（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等

   当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
6. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
9. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
10. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものといたします。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
12. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。

注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.4.0-1　2017.11)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

© 2021 Renesas Electronics Corporation. All rights reserved.