

お客様各位

カタログ等資料中の旧社名の扱いについて

2010 年 4 月 1 日を以って NEC エレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願い申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010 年 4 月 1 日
ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。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。
3. 当社製品を改造、改変、複製等しないでください。
4. 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お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、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に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。

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

高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）

特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障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。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、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、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害など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、機器またはシステムとしての出荷保証をお願いいたします。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、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、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。お客様がかかる法令を遵守しないこと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。
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固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

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その総株主の議決権の過半数を直接または間接に保有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。

アプリケーション・ノート

NECエレクトロニクス株式会社
マイクロコンピュータ事業部
製品ソリューション グループ
グループ マネージャー

大場 浩司

(担当：西浦 真平)



78K0R/Kx3-L

サンプル・プログラム

シリアル・アレイ・ユニット UART通信編

この資料は、サンプル・プログラムの「シリアル・アレイ・ユニットを用いた3線シリアルI/O(スレーブ送受信)」の動作概要と、基本的な設定内容を説明したものです。サンプル・プログラムでは、3線シリアルI/O通信でマスタ側からクロック供給，データ0x05，0x50を交互に受信，マスタ側へのデータ送信を行います。

対象デバイス

78K0R/KC3-L(44pin)

マイクロコントローラ

78K0R/KC3-L(48pin)

マイクロコントローラ

78K0R/KD3-Lマイクロコントローラ

78K0R/KE3-Lマイクロコントローラ

78K0R/KF3-Lマイクロコントローラ

78K0R/KG3-Lマイクロコントローラ

目次

第1章	概要	・・・3
第2章	回路イメージ	・・・6
2.1	回路イメージ	・・・6
2.2	周辺ハードウェア	・・・6
第3章	ソフトウェアについて	・・・7
3.1	ファイル構成	・・・7
3.2	使用する内蔵機能	・・・8
3.3	UART通信機能の設定と動作概要	・・・8
3.4	フロー・チャート	・・・8
第4章	設定方法について	・・・13
4.1	シリアル・アレイ・ユニットの設定	・・・13
4.2	割り込みの設定	・・・13
4.3	シリアル・アレイ・ユニットの設定レジスタ	・・・14
4.4	割り込み設定レジスタ	・・・22
4.5	シリアル・アレイ・ユニットの設定概要	・・・28
4.6	割り込みの設定概要	・・・35
第5章	PM+を用いたHEXファイルの生成	・・・36
5.1	ダウンロードファイルの解説	・・・36
5.2	サンプル・プログラムのHEXファイル生成	・・・37
5.3	開発環境のダウンロード，インストール	・・・39
第6章	関連資料	・・・40
付録A	プログラム・リスト	・・・41
	・アセンブリ言語	・・・41
	・C言語	・・・62
付録B	改版履歴	・・・84

・本資料に記載されている内容は2009年1月現在のもので、今後、予告なく変更することがあります。量産設計の際には最新の個別データ・シート等をご参照ください。

・文書による当社の事前の承諾なしに本資料の転載複製を禁じます。当社は、本資料の誤りに関し、一切その責を負いません。

・当社は、本資料に記載された当社製品の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、一切その責を負いません。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。

・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お客様または第三者に生じた損害に関し、当社は、一切その責を負いません。

・当社は、当社製品の品質、信頼性の向上に努めておりますが、当社製品の不具合が完全に発生しないことを保証するものではありません。また、当社製品は耐放射線設計については行っておりません。当社製品をお客様の機器にご使用の際には、当社製品の不具合の結果として、生命、身体および財産に対する損害や社会的損害を生じさせないように、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を行ってください。

・当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定していただく「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。

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

特別水準：輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器

特定水準：航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器、生命維持のための装置またはシステム等

当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。意図されていない用途で当社製品の使用をお客様が希望する場合には、事前に当社販売窓口までお問い合わせください。

（注）

（1）本事項において使用されている「当社」とは、NECエレクトロニクス株式会社およびNECエレクトロニクス株式会社がその総株主の議決権の過半数を直接または間接に保有する会社をいう。

（2）本事項において使用されている「当社製品」とは、（1）において定義された当社の開発、製造製品をいう。

第1章 概 要

このサンプル・プログラムでは、シリアル・アレイ・ユニットのUART通信機能の使用例を示しています。対向機器から送られてくるASCII文字を解析し、応答を返却する処理を行います。

(1)周辺機能設定の主な内容

<オプション・バイトでの設定>

- ウォッチドッグ・タイマのカウンタ動作制御（動作停止）
- 高速内蔵発振回路の周波数（8MHz）
- LVI動作設定
- オンチップ・デバッグ許可

<リセット解除後の初期化処理での設定>

- 入出力ポートの設定※
- CPU／周辺ハードウェア・クロックの設定
- 各周辺ハードウェア・マクロの使用可否の設定
- シリアル・アレイ・ユニットの設定
 - ・入力クロック供給
 - ・動作モードの設定（送信側，受信側）
 - ・通信フォーマットの設定（送信側，受信側）
 - ・データ長8bit
 - ・偶数パリティ
 - ・転送ボー・レートの設定（送信側，受信側）
 - ・9600bps
 - ・送信データのレベル反転の設定
 - ・シリアル・データ出力値の設定
 - ・シリアル出力許可

注：未使用端子のポート設定は、各デバイスによって設定が異なります。以下の各デバイスのポートをLowオープン出力に設定してください。ポートの設定については、サンプル・プログラム初期設定編アプリケーション・ノートの“4.4 ポートの設定レジスタ”を参照して下さい。

- ・ KC3-L(44pin)
 - 未使用端子：P10-13, P20-27, P32, P40-41, P50-52, P70-75, P80-83, P120
P150-151
- ・ KC3-L(48pin)
 - 未使用端子：P10-13, P20-27, P32, P40-41, P50-51, P60-61, P70-75, P80-83
P120, P140, P150-152
- ・ KD3-L
 - 未使用端子：P00-01, P10-13, P20-27, P32, P40-41, P50-52, P60-61, P70-77
P80-83, P120, P140, P150-152

- KE3-L

未使用端子：P00-01, P10-17, P20-27, P32-33, P40-43, P50-53, P60-61, P70-77
P80-83, P120, P140-141, P150-153

- KF3-L

未使用端子：P02-06, P10-17, P20-27, P40-47, P50-55, P60-67, P70-77
P90-91, P110-111, P120, P130, P140, P142-144, P150-153

- KG3-L

未使用端子：P00-06, P10-17, P20-27, P40-47, P50-57, P60-67, P70-77
P80-87, P91, P110-111, P120, P130-131, P140-145, P150-157

(2) 割り込み処理動作の内容

受信完了割り込み発生の場合は、受信データを格納し、受信データに対応した値を送信します。以下に受信データと送信データの対応表を示します。

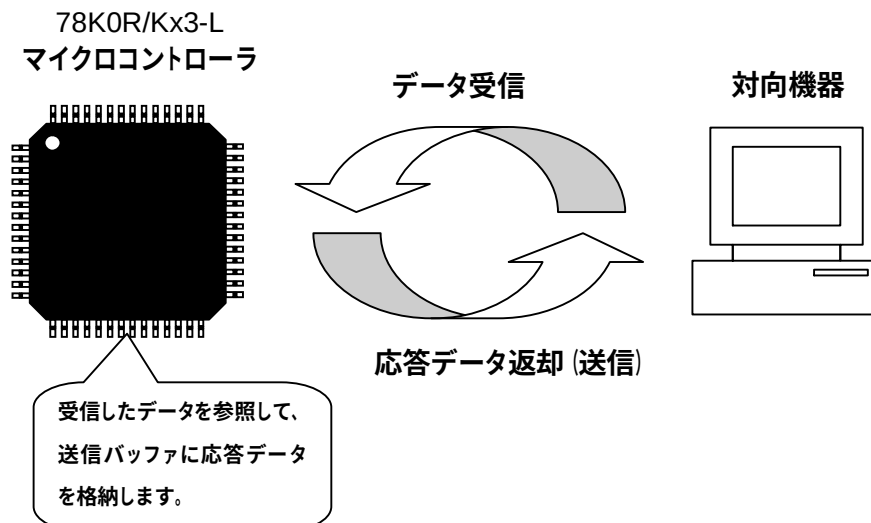
受信したデータ	返却する（送信）データ
T (54H)	O (4FH) , K (4BH) , "CR" (0DH) , "LF" (0AH)
t (74H)	o (6FH) , k (6BH) , "CR" (0DH) , "LF" (0AH)
上記以外	U (55H) , C (43H) , "CR" (0DH) , "LF" (0AH)

エラー割り込み発生の場合は、発生したエラーをチェックし、エラーに対応した値を送信します。以下にエラーと送信データの対応表を示します。

発生したエラー	返却する（送信）データ
パリティ・エラー	P (50H) , E (45H) , "CR" (0DH) , "LF" (0AH)
フレーミング・エラー	F (46H) , E (45H) , "CR" (0DH) , "LF" (0AH)
オーバーラン・エラー	O (4FH) , E (45H) , "CR" (0DH) , "LF" (0AH)

(3) メイン処理動作の内容

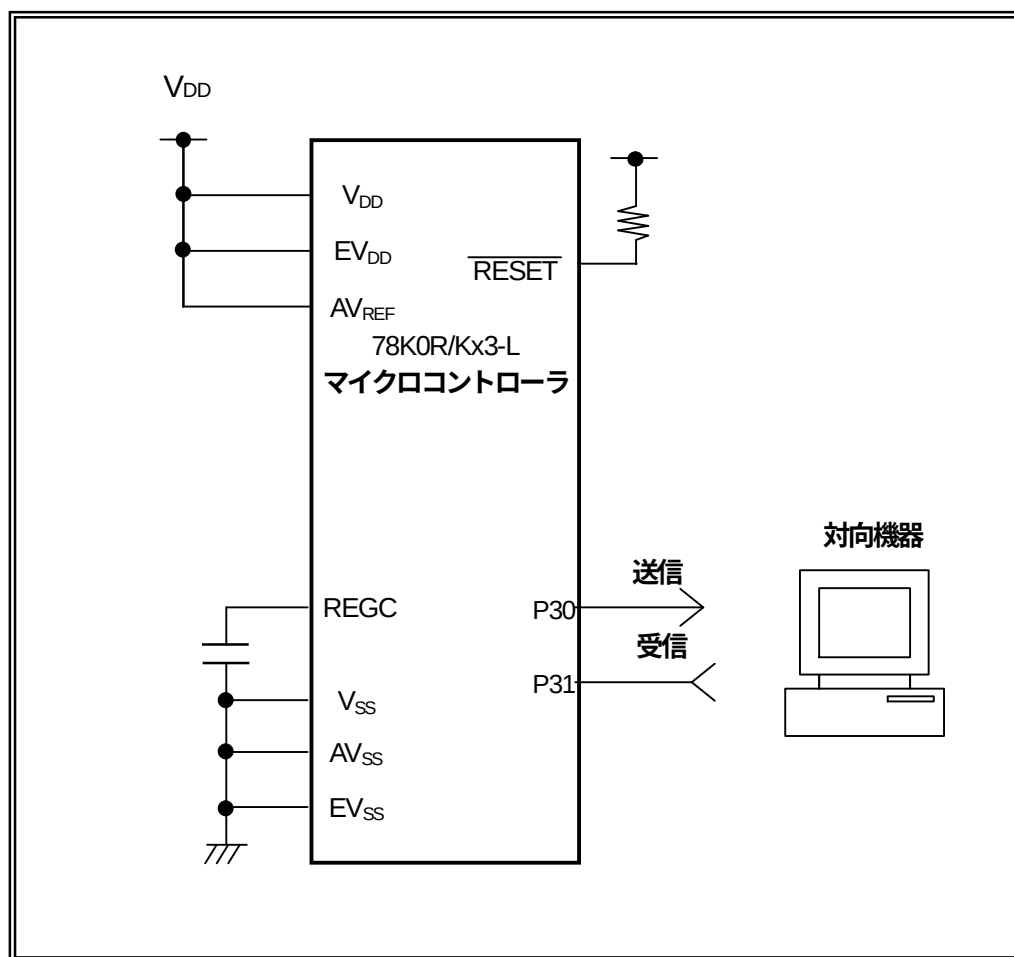
- ループのみ(割り込み発生待機)



第2章 回路イメージ

この章では、このサンプル・プログラムで使用する回路イメージおよび周辺ハードウェアを説明します。

2.1 回路イメージ



注意：この回路イメージは接続の概要を示す為に簡略化しています。実際に回路を作成される場合は、端子処理などを適切に行い、電流など電気的特性を満たすように設計してください。（P121～P124は入力専用ポートになりますから個別に抵抗を介してV_{DD}又はV_{SS}に接続して下さい）。

2.2 周辺ハードウェア

使用する周辺ハードウェアを次に示します。

送信用ポート ： P30
受信用ポート ： P31

第3章 ソフトウェアについて

この章では、ダウンロードする圧縮ファイルのファイル構成、使用するマイコンの内蔵周辺機能、サンプル・プログラムの動作概要、およびフロー・チャートを説明します。

3.1 ファイル構成

ダウンロードする圧縮ファイルのファイル構成は、次のようになっています。

【C言語版】

ファイル名	説 明	同封圧縮 (*.zip) ファイル	
			
Kx3-L_UART.c	マイコンのハードウェア初期化処理とメイン処理のソース・ファイル	●	●
OP.asm	オプション・バイトの指定ファイル	●	●
78K0RKx3-L_sample_program.prw	統合開発環境 PM+用ワーク・スペース・ファイル	-	●
78K0RKx3-L_sample_program.prj	統合開発環境 PM+用設定データ	-	●

備考



：ソース・ファイルのみ同封



：統合開発環境 PM+で使用するファイルを同封

【アセンブリ言語版】

ファイル名	説 明	同封圧縮 (*.zip) ファイル	
			
Kx3-UART.asm	マイコンのハードウェア初期化処理とメイン処理のソース・ファイル	●	●
OP.asm	オプション・バイトの指定ファイル	●	●
78K0RKx3-L_sample_program.prw	統合開発環境 PM+用ワーク・スペース・ファイル	-	●
78K0RKx3-L_sample_program.prj	統合開発環境 PM+用設定データ	-	●

備考



：ソース・ファイルのみ同封



：統合開発環境 PM+で使用するファイルを同封

3.2 使用する内蔵機能

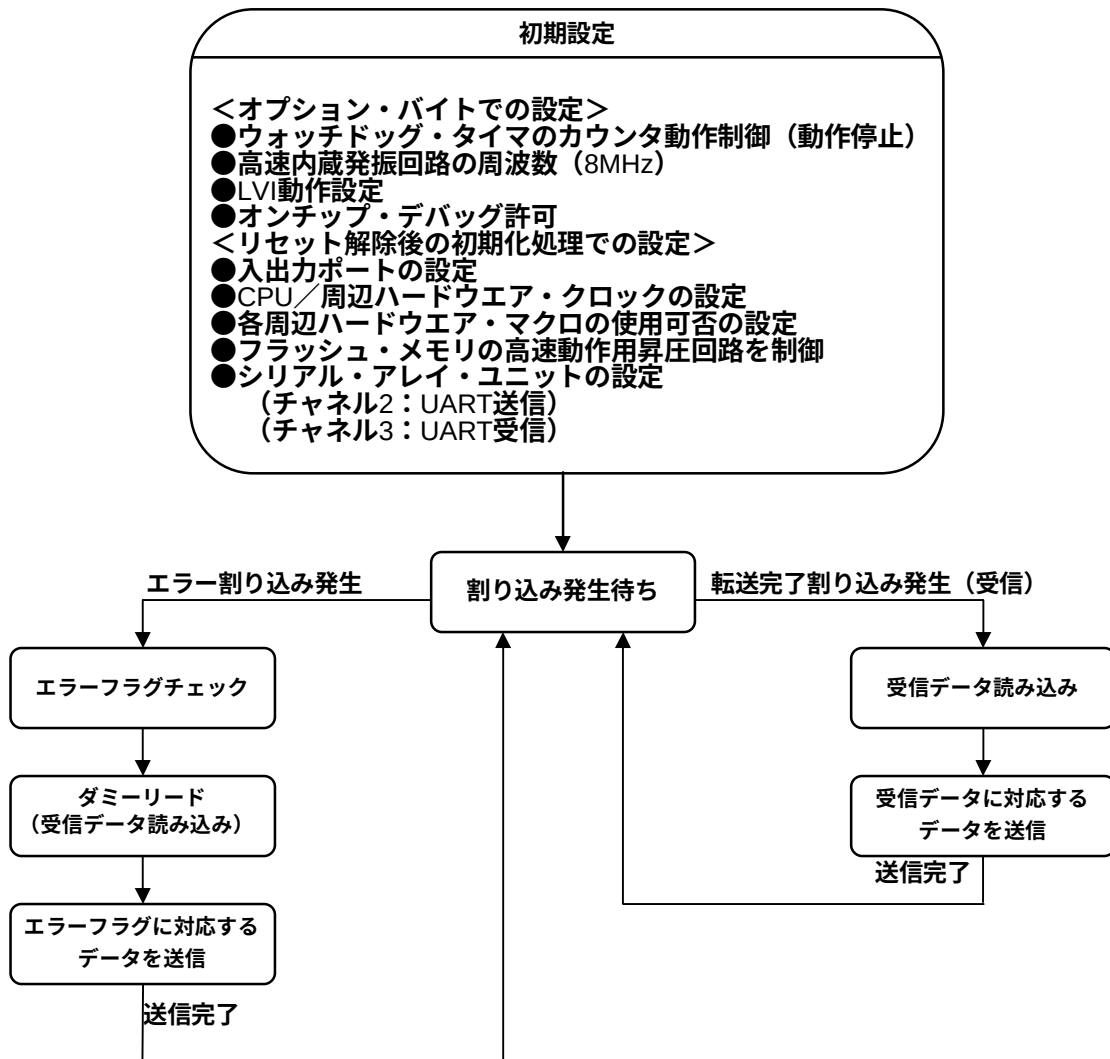
このサンプル・プログラムでは、マイコンに内蔵する次の周辺機能を使用します。

- UART通信機能
- UART1受信完了割り込み : INTSR1
- UART1の通信エラー発生割り込み : INTSRE1

3.3 UART通信機能の設定と動作概要

このサンプル・プログラムでは、UART通信機能の設定を行います。

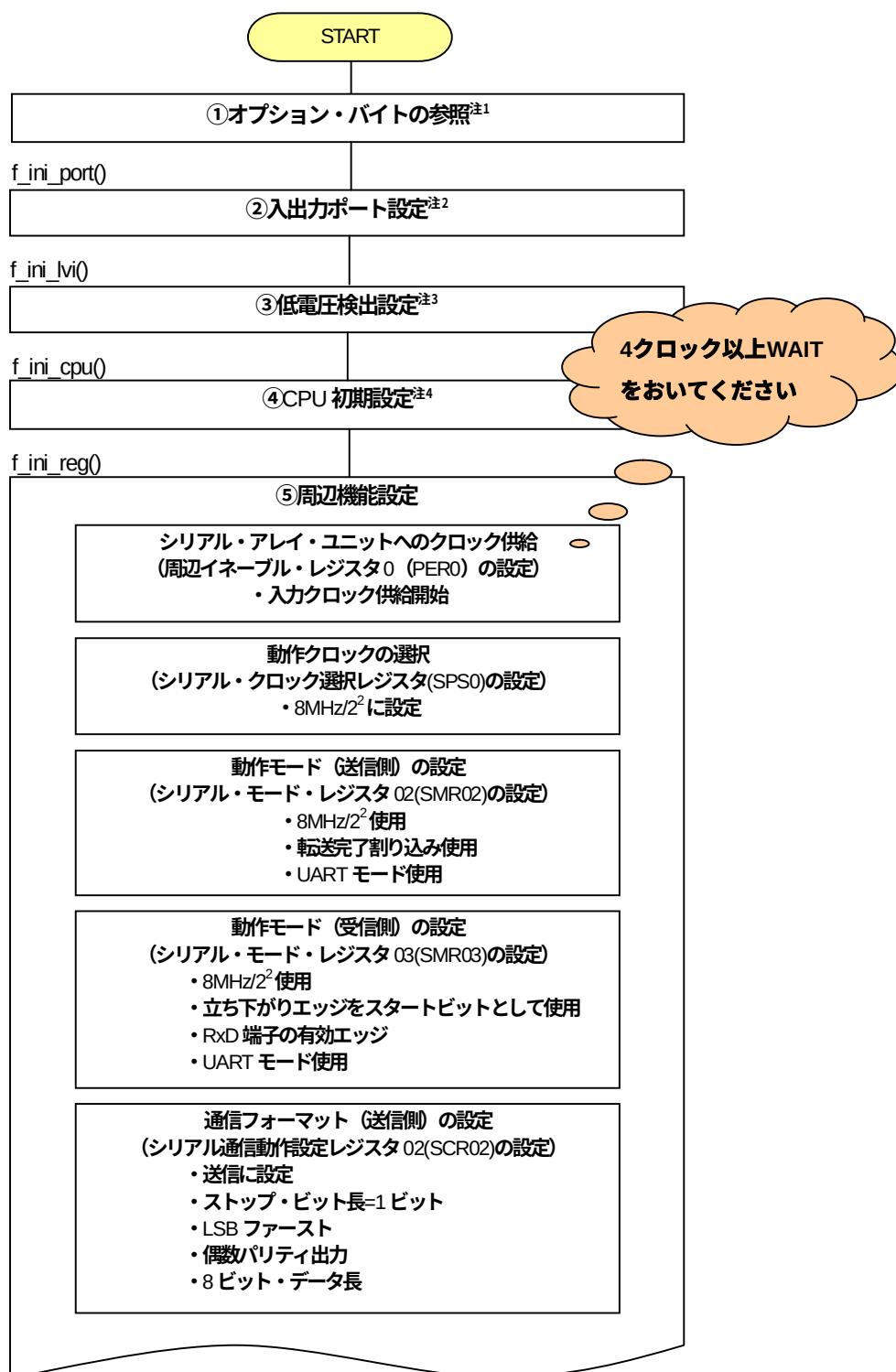
設定完了後は、受信したデータをもとに、対向機器に応答データを送信します。またエラーが発生した場合は、そのエラーの種類をもとに、対向機器に応答データを送信します。応答データの種類は第1章に記載しています。処理詳細については、以下の状態遷移図に示します。



3.4 フロー・チャート

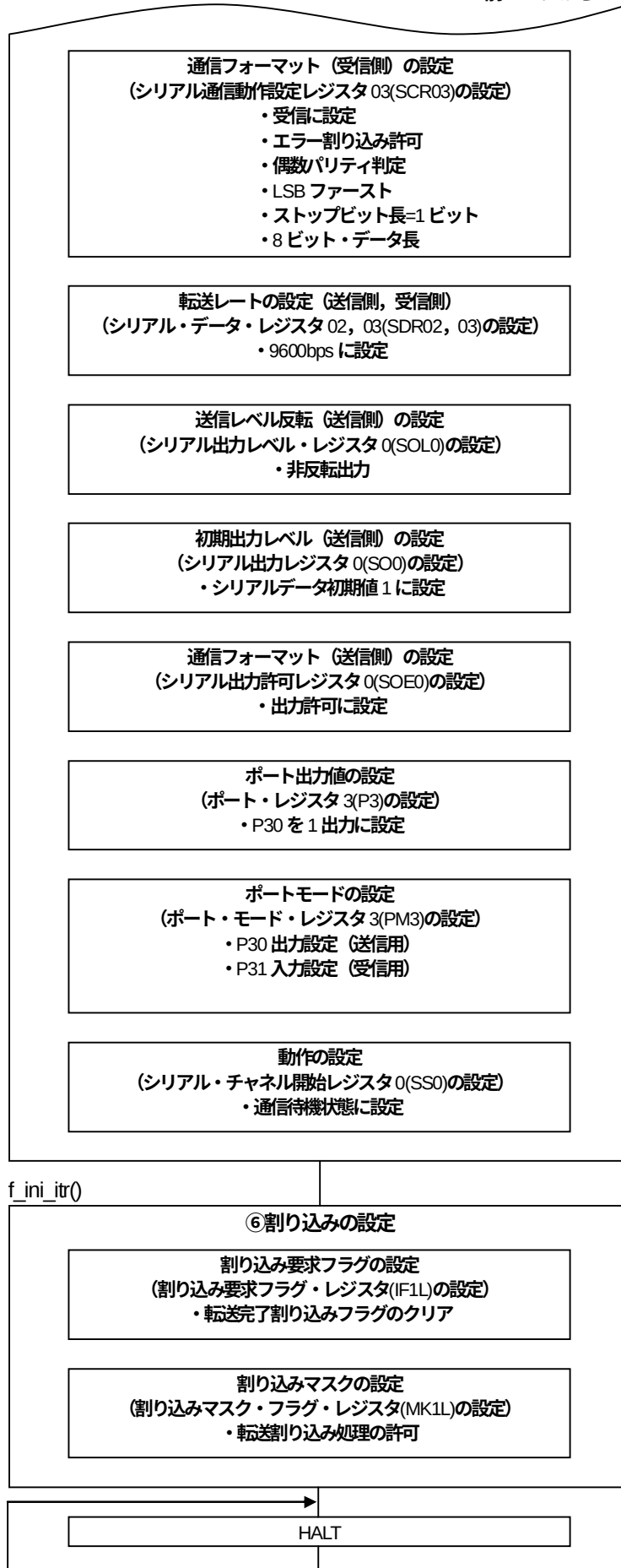
このサンプル・プログラムのフロー・チャートを次に示します。

(1) 全体フロー



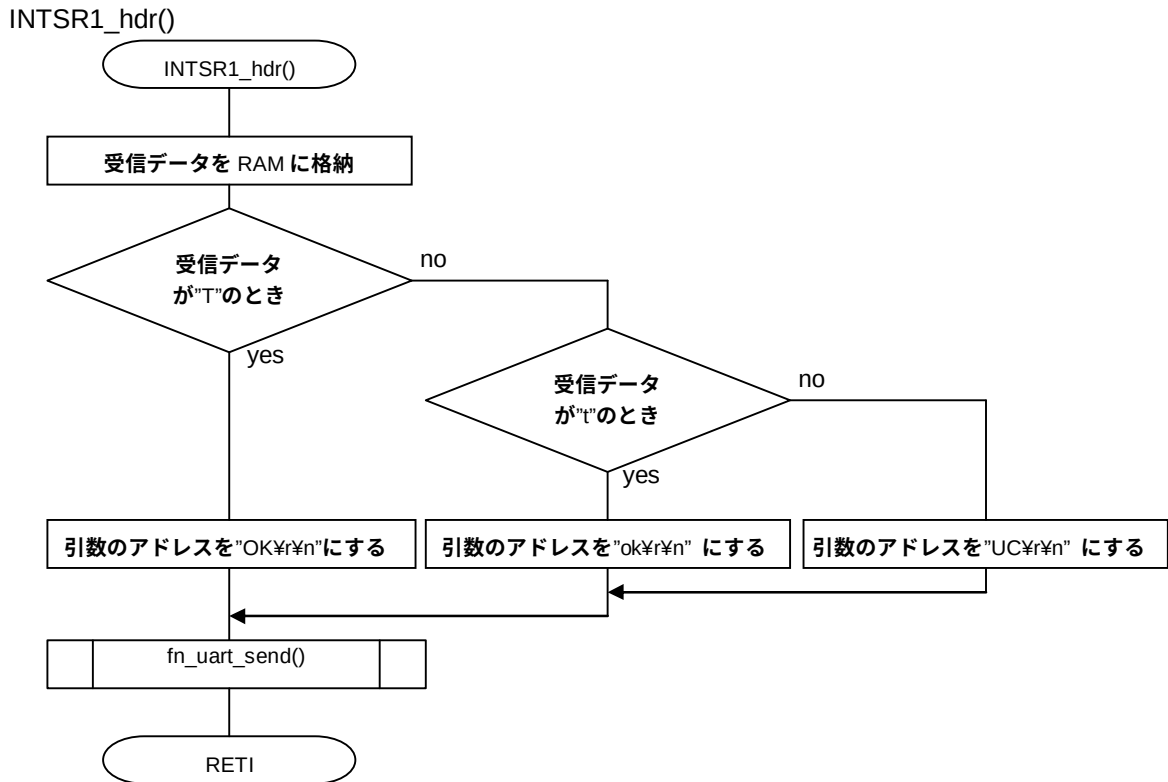
次ページへ

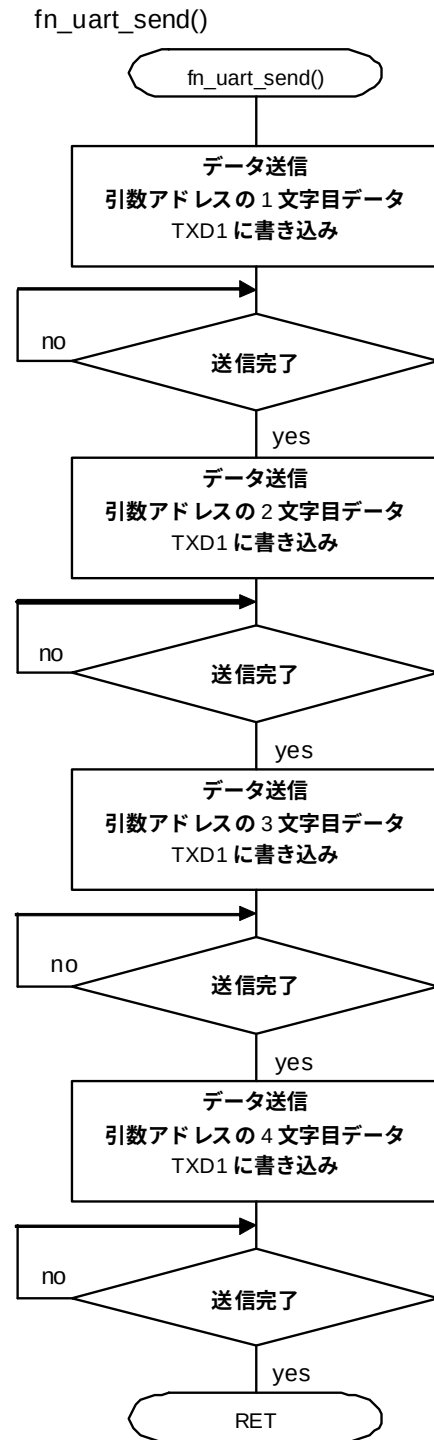
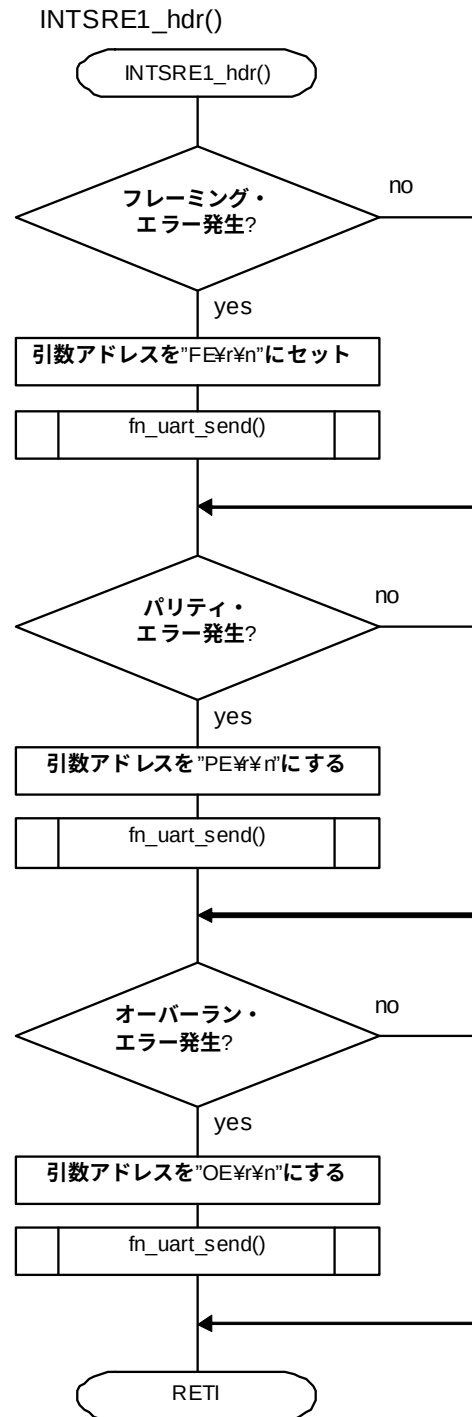
前ページから



- 注1. オプション・バイトについては、サンプル・プログラム初期設定編アプリケーション・ノートの“4.6 オプション・バイトの設定概要”を参照して下さい。
2. 入出力ポートの設定 (f_ini_port()) の設定については、サンプル・プログラム初期設定編アプリケーション・ノートの“3.4 フロー・チャート”を参照して下さい。
3. 低電圧検出設定 (f_ini_lbi()) については、サンプル・プログラム低電圧検出回路編アプリケーション・ノートの“3.4 フロー・チャート”を参照して下さい。
4. CPU初期設定 (f_ini_cpu()) については、サンプル・プログラム初期設定編アプリケーション・ノートの“3.4 フロー・チャート”を参照して下さい。

(2) 割り込み処理





第4章 設定方法について

この章では、シリアル・アレイ・ユニットのチャンネル2, チャンネル3をUART通信機能として使用する際の設定について説明します。

その他の初期設定については、78K0R/Kx3-L サンプル・プログラム (初期設定編) アプリケーション・ノートを参照してください。

レジスタ設定方法の詳細については、各製品のユーザーズ・マニュアル ([78K0R/Kx3-L](#)) を参照してください。

アセンブラ命令については、[78K0Rシリーズ 命令編 ユーザーズ・マニュアル](#)を参照してください。

4.1 シリアル・アレイ・ユニットの設定

UART送受信機能を使用する場合、次のレジスタの設定、操作が必要になります。

初期設定で使用するレジスタ

- ・周辺イネーブル・レジスタ (PER0)
- ・シリアル・クロック選択レジスタ0 (SPS0)
- ・シリアル・モード・レジスタ02, 03 (SMR02, 03)
- ・シリアル通信動作設定レジスタ02, 03 (SCR02, 03)
- ・シリアル・データ・レジスタ02, 03 (SDR02, 03)
- ・シリアル出力レベル・レジスタ0(SOL0)
- ・シリアル出力レジスタ0 (SO0)
- ・シリアル出力許可レジスタ0 (SOE0)
- ・シリアル・チャンネル開始レジスタ0 (SS0)

UART通信中断の際に使用するレジスタ

- ・シリアル・チャンネル停止レジスタ0 (ST0)

4.2 割り込みの設定

割り込みの操作をするには、次のレジスタの設定、操作が必要となります。

割り込み操作で使用するレジスタ

- ・割り込み要求フラグ・レジスタ (IF1L)
- ・割り込みマスク・フラグ・レジスタ (MK1L)

4.3 シリアル・アレイ・ユニットの設定レジスタ

(1) 周辺イネーブル・レジスタ0(PER0)

PER0は、各周辺ハードウェア・マクロの使用可否を設定するレジスタです。使用しないハードウェアへはクロック供給も停止させることで、低消費電力化とノイズ低減をはかります。

シリアル・アレイ・ユニット（SAU）を使用するときは、必ずビット2（SAU0EN）に1を設定してください。

PER0は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

リセット信号の発生により、00Hになります。

略号： PER0

	7	6	5	4	3	2	1	0
RTCEN	0		ADCEN	IICAEN	0	SAU0EN	0	0
	0/1	0	0/1	0/1	0	0/1	0	0

RTCEN	リアルタイム・カウンタ（RTC）の入カクロックの制御
0	入力クロック供給停止 ・リアルタイム・カウンタ（RTC）で使用するSFRへのライト不可（リード可） ・リアルタイム・カウンタ（RTC）の動作は継続可能
1	入力クロック供給許可 ・リアルタイム・カウンタ（RTC）で使用するSFRへのリード／ライト可

ADCEN	A/Dコンバータの入カクロックの制御
0	入力クロック供給停止 ・A/Dコンバータで使用するSFRへのライト不可 ・A/Dコンバータはリセット状態
1	入力クロック供給許可 ・A/Dコンバータで使用するSFRへのリード／ライト可

IICAEN	シリアル・インタフェースIICAの入カクロックの制御
0	入力クロック供給停止 ・シリアル・インタフェースIICAで使用するSFRへのライト不可 ・シリアル・インタフェースIICAはリセット状態
1	入力クロック供給許可 ・シリアル・インタフェースIICAで使用するSFRへのリード／ライト可

SAU0EN	シリアル・アレイ・ユニットの入カクロックの制御
0	入力クロック供給停止 ・シリアル・アレイ・ユニットで使用するSFRへのライト不可 ・シリアル・アレイ・ユニットはリセット状態
1	入力クロック供給許可 ・シリアル・アレイ・ユニットで使用するSFRへのリード／ライト可

(2) シリアル・クロック選択レジスタ0 (SPS0)

SPS0は、各チャンネルに共通して供給される2種類の動作クロック (CK00, CK01) を選択する16ビット・レジスタです。SPS0のビット7-4でCK01を、ビット3-0でCK00を選択します。

SPS0は、動作中 (SE0n = 1のとき) の書き換えは禁止です。

SPS0は16ビット・メモリ操作命令で設定します。またSPS0の下位8ビットは、SPS0Lで8ビット・メモリ操作命令で設定できます。

リセット信号の発生により、SPS0は0000Hになります。

略号： SPS0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

PRS0p 3	PRS0p 2	PRS0p 1	PRS0p 0	動作クロック(CK0p)の選択				
					f _{CLK} =2MHz	f _{CLK} =5MHz	f _{CLK} =10MHz	f _{CLK} =20MHz
0	0	0	0	f _{CLK}	2MHz	5MHz	10MHz	20MHz
0	0	0	1	f _{CLK} /2	1MHz	2.5MHz	5MHz	10MHz
0	0	1	0	f _{CLK} /2 ²	500kHz	1.25MHz	2.5MHz	5MHz
0	0	1	1	f _{CLK} /2 ³	250kHz	625kHz	1.25MHz	2.5MHz
0	1	0	0	f _{CLK} /2 ⁴	125kHz	313kHz	625kHz	1.25MHz
0	1	0	1	f _{CLK} /2 ⁵	62.5kHz	156kHz	313kHz	625kHz
0	1	1	0	f _{CLK} /2 ⁶	31.3kHz	78.1kHz	156kHz	313kHz
0	1	1	1	f _{CLK} /2 ⁷	15.6kHz	39.1kHz	78.1kHz	156kHz
1	0	0	0	f _{CLK} /2 ⁸	7.81kHz	19.5kHz	39.1kHz	78.1kHz
1	0	0	1	f _{CLK} /2 ⁹	3.91kHz	9.77kHz	19.5kHz	39.1kHz
1	0	1	0	f _{CLK} /2 ¹⁰	1.95kHz	4.88kHz	9.77kHz	19.5kHz
1	0	1	1	f _{CLK} /2 ¹¹	977Hz	2.44kHz	4.88kHz	9.77kHz
1	1	1	1	INTTM02				
上記以外				設定禁止				

(3) シリアル・モード・レジスタ0n (SMR0n) n = 2, 3

SMR0nは、チャンネルnの動作モード設定レジスタです。動作クロック (f_{MCK}) の選択、シリアル・クロック (SCK) 入力の使用可否、スタート・トリガ設定、動作モード (CSI, UART, I²C) 設定、割り込み要因の選択を行います。またUARTモード時のみ、受信データのレベル反転の設定を行います。

SMR0nは、動作中 (SE0n = 1のとき) の書き換えは禁止です。ただしMD0n0ビットは、動作中でも書き換えをすることができます。

SMR0nは、16ビット・メモリ操作命令で設定します。

リセット信号の発生により、0020Hになります

略号: SMR0n

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKS0n	CCS0n	0	0	0	0	0	STS0n	0	SISn0	1	0	0	MD0n2	MD0n1	MD0n0
0/1	0/1	0	0	0	0	0	0/1	0	0/1	1	0	0	0/1	0/1	0/1

CKS0n	チャンネルnの動作クロック (f_{MCK}) の選択
0	シリアル・クロック選択レジスタ0 (SPS0) で設定したプリスケアラ出力クロックCK00
1	シリアル・クロック選択レジスタ0 (SPS0) で設定したプリスケアラ出力クロックCK01

CCS0n	チャンネルnの転送クロック (f_{CLK}) の選択
0	CKS0nビットで指定した動作クロック f_{MCK} の分周クロック
1	SCK端子からの入力クロック (CSIモードのスレーブ)

STS0n	スタート・トリガ要因の選択
0	ソフトウェア・トリガのみ有効 (CSI, UART送信, 簡易I ² C時に選択)
1	RxD端子の有効エッジ (UART受信時に選択)

SIS0n0	UARTモードでのチャンネルnの受信データのレベル反転の制御
0	立ち下がりエッジをスタートビットとして検出します。 入力される通信データは、そのまま取り込まれます。
1	立ち上がりエッジをスタートビットとして検出します。 入力される通信データは、反転して取り込まれます。

MD0n2	MD0n1	チャンネルnの動作モードの設定
0	0	CSIモード
0	1	UARTモード
1	0	簡易I ² Cモード
1	1	設定禁止

MD0n0	チャンネルnの割り込み要因の選択
0	転送完了割り込み
1	バッファ空き割り込み

(4) シリアル通信動作設定レジスタ (SCR0n) n = 2, 3

SCR0nは、チャンネルnの通信動作設定レジスタです。データ送受信モード、データとクロックの位相、エラー信号のマスク可否、パリティ・ビット、先頭ビット、ストップ・ビット、データ長などの設定を行います。

SCR0nは、動作中 (SE0n = 1のとき) の書き換えは禁止です。

SCR0nは、16ビット・メモリ操作命令で設定します。

リセット信号の発生により、0087Hになります。

略号： SCR0n

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXE0n	RXE0n	DAP0n	CKP0n	0	EOC0n	PTC0n1	PTC0n0	DIR0n	0	SLC0n1	SLC0n0	0	DLS0n2	DLS0n1	DLS0n0
0/1	0/1	0/1	0/1	0	0/1	0/1	0/1	0/1	0	0/1	0/1	0	0/1	0/1	0/1

TXE0n	RXE0n	チャンネルnの動作モードの設定
0	0	通信禁止
0	1	受信のみを行う
1	0	送信のみを行う
1	1	送受信を行う

EOC0n	エラー割り込み信号 (INTSREx (x = 0, 1)) のマスク可否の選択
0	エラー割り込みINTSRExをマスクする (INTSRxはマスクされない)
1	エラー割り込みINTSRExの発生を許可する (エラー発生時にINTSRxはマスクされる)

PTC0n1	PTC0n1	UARTモードでのパリティ・ビットの設定	
		送信動作	受信動作
0	1	パリティ・ビットを出力しない	パリティなしで受信
0	1	0パリティを出力	パリティ判定を行わない
1	0	偶数パリティを出力	偶数パリティとして判定を行う
1	1	奇数パリティを出力	奇数パリティとして判定を行う

DIR0n	CSI, UARTモードでのデータ転送順序の選択
0	MSBファーストで入出力を行う
1	LSBファーストで入出力を行う

SLC0n1	SLC0n0	UARTモードでのストップ・ビットの設定
0	0	ストップ・ビットなし
0	1	ストップ・ビット長 = 1ビット
1	0	ストップ・ビット長 = 2ビット
1	1	設定禁止

DLS0n2	DLS0n1	DLS0n0	CSIモードでのデータ長の設定
1	0	0	5ビット・データ長
1	1	0	7ビット・データ長
1	1	1	8ビット・データ長
その他			設定禁止

(5) シリアル・データ・レジスタ0n (SDR0n) n = 2, 3

SDR0nは、チャンネルnの送受信データ・レジスタ（16ビット）です。ビット7-0は送受信バッファ・レジスタとして機能し、ビット15-9の部分は動作クロック (f_{MCK}) の分周設定レジスタとして使われます。

シリアル・モード・レジスタ0n (SMR0n) でCCS0nビットを0に設定した場合は、動作クロックをこのSDR0nの上位7ビットで分周設定したクロックが、転送クロックとして使用されます

SDR0nは、16ビット単位でリード／ライト可能です。

SDR0nの上位7ビットへの書き込み、読み出しは動作停止状態 (SE0n=0) のときのみ有効です。

SDR0nの下位8ビットへの書き込み、読み出しは動作停止状態に行ってください。

リセット信号の発生により、0000Hになります。

略号： SDR0n

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
							0								

SDR01[15:9]							動作クロック (f_{MCK}) の分周による転送クロック設定								
0	0	0	0	0	0	0	$f_{MCK}/2$								
0	0	0	0	0	0	1	$f_{MCK}/4$								
0	0	0	0	0	1	0	$f_{MCK}/6$								
0	0	0	0	0	1	1	$f_{MCK}/8$								
.	.	.	.	.	.	.	.								
.	.	.	.	.	.	.	.								
.	.	.	.	.	.	.	.								
1	1	1	1	1	1	0	$f_{MCK}/254$								
1	1	1	1	1	1	1	$f_{MCK}/256$								

(6) シリアル出力レベル・レジスタ0 (SOL0)

SOL0は、各チャンネルのシリアル出力のバッファ・レジスタです。

チャンネル動作停止状態のみ有効で、チャンネル動作開始状態時には、シリアル通信動作の値が優先されます。

SOL0は、16ビット・メモリ操作命令で設定します。また、SOL0の下位8ビットは、SOL0Lで1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定できます。

リセット信号の発生により、0000Hになります。

略号： SOL0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	SOL02	0	SOL00
0	0	0	0	0	0	0	0	0	0	0	0	0	0/1	0	0/1

SOL02	UARTモードでのチャンネル2の送信データのレベル反転の選択
0	通信データは、そのまま出力されます。
1	通信データは、反転して出力されます。

(7) シリアル出力レジスタ0 (SO0)

SO0は、各チャンネルのシリアル通信動作の出力許可／停止を設定するレジスタです。

SO0は16ビット・メモリ操作命令で設定します。

リセット信号の発生により、SO0は0F0FHになります。

略号： SO0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	1	CKO02	CKO01	CKO00	0	0	0	0	1	SO02	SO01	SO00
0	0	0	0	1	0/1	0/1	0/1	0	0	0	0	1	0/1	0/1	0/1

CKO02	チャンネル2のシリアル・クロック出力
0	シリアル・クロック出力値が“0”
1	シリアル・クロック出力値が“1”

SO02	チャンネル2のシリアル・データ出力
0	シリアル・データ出力値が“0”
1	シリアル・データ出力値が“1”

(8) ポート・モード・レジスタ3 (PM3)

このレジスタは使用するポートごとに設定する必要があります。

ポート・モード・レジスタ3 (PM3)

7	6	5	4	3	2	1	0
1	1	1	1	PM33	PM32	PM31	PM30

PM3n	P3n端子の入出力モードの選択 (n = 0-3)
0	出力モード (出力バッファ・オン)
1	入力モード (出力バッファ・オフ)

(9) ポート・レジスタ3 (P3)

このレジスタは使用するポートごとに設定する必要があります。

ポート・レジスタ3 (P3)

7	6	5	4	3	2	1	0
1	1	1	1	P33	P32	P31	P30

P3n	n = 0-3 出力データの制御 (出力モード時)
0	0を出力
1	1を出力

(10) ポート出力モード・レジスタ3 (POM3)

P30-P32の出力モードを1ビット単位で設定するレジスタです。

異電位の外部デバイスとのシリアル通信時および同電位の外部デバイスとの簡易I²C通信時のSDA10端子にN-chオープン・ドレイン出力 (VDD耐圧) モードを選択できます。

POM3は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

リセット信号の発生により、00Hになります。

このレジスタは使用するポートごとに設定する必要があります。

ポート出力モード・レジスタ3 (POM3)

7	6	5	4	3	2	1	0
0	0	0	0	0	POM32	POM31	POM30

POM3n	P3n端子の出力モードの選択 (n = 0-3)
0	通常出力モード
1	N-chオープン・ドレイン出力 (VDD耐圧) モード

(11) シリアル出力許可レジスタ0 (SOE0)

SOE0は、各チャネルのシリアル通信動作の出力許可／停止を設定するレジスタです。

SOE0は16ビット・メモリ操作命令で設定します。また、SOEの下位8ビットは、SOE0Lで1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定できます。

リセット信号の発生により、SOE0は0000Hになります。

略号： SOE0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	SOE02	SOE01	SOE00
0	0	0	0	0	0	0	0	0	0	0	0	0	0/1	0/1	0/1

SOE02	チャネル2のシリアル出力許可／停止
0	シリアル通信動作による出力停止
1	シリアル通信動作による出力許可

(12) シリアル・チャンネル開始レジスタ0 (SS0)

SS0は、通信／カウント**開始の許可のみ**をチャンネル毎に設定するトリガ・レジスタです。

SS0は、16ビット・メモリ操作命令で設定します。また、SS0の下位8ビットは、SS0Lで1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定できます。

リセット信号の発生により、SS0は0000Hになります。

略号： SS0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	SS0 3	SS0 2	SS0 1	SS0 0
0	0	0	0	0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1

SS0n	チャンネルnの動作開始トリガ
0	トリガ動作せず
1	SE0nに1をセットし、通信待機状態に遷移する

(13) シリアル・チャンネル停止レジスタ0 (ST0)

ST0は、通信／カウント**停止の許可のみ**をチャンネル毎に設定するトリガ・レジスタです。

ST0は、16ビット・メモリ操作命令で設定します。また、ST0の下位8ビットは、ST0Lで1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定できます。

リセット信号の発生により、ST0は0000Hになります。

略号： ST0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	ST0 3	ST0 2	ST0 1	ST0 0
0	0	0	0	0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1

ST0n	チャンネルnの動作停止トリガ
0	トリガ動作せず
1	SE0nを0にクリアし、通信動作を停止する

4.4 割り込みの設定レジスタ

(1) 割り込み要求フラグ・レジスタ (IF1L)

割り込み要求フラグは、対応する割り込み要求の発生または命令の実行によりセット (1) され、割り込み要求受け付け時、リセット信号発生時または命令の実行によりクリア (0) されるフラグです。

割り込みが受け付けられた場合、まず割り込み要求フラグが自動的にクリアされてから割り込みルーチンに入ります。

IF1Lは、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

また、IF1LとIF1Hをあわせて16ビット・レジスタIF1として使用するときは、16ビット・メモリ操作命令で設定します。

リセット信号の発生により、00Hになります。

略号： IF1L

7	6	5	4	3	2	1	0
TMIF03	TMIF02	TMIF01	TMIF00	IICAIF	SREIF1	SRIF1	STIF1 CSIIF10 IICIF10

xxIFx	割り込み要求フラグ
0	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

注意1. タイマ、シリアル・インタフェース、A/Dコンバータなどをスタンバイ解除後に動作させる場合、いったん割り込み要求フラグをクリアしてから動作させてください。ノイズなどにより割り込み要求フラグがセットされる場合があります。

2. 割り込み要求フラグ・レジスタのフラグ操作には、1ビット・メモリ操作命令 (CLR1) を使用してください。C言語での記述の場合は、コンパイルされたアセンブラが1ビット・メモリ操作命令 (CLR1) になっている必要があるため、「IF1L.0 = 0;」や「_asm("clr1 IF1L,0");」のようなビット操作命令を使用してください。

なお、C言語で「IF1L &= 0xff;」のように8ビット・メモリ操作命令で記述した場合、コンパイルすると3命令のアセンブラになります。

```
MOV    A        ,    IF1L
AND     A        ,    #0FFH
MOV    IF1L     ,    A
```

この場合、「MOV A, IF1L」後から「MOV IF1L, A」の間のタイミングで、同一の割り込み要求フラグ・レジスタ (IF1L) の他ビットの要求フラグがセット (1) されても、「MOV IF1L, A」でクリア (0) されます。

したがって、C言語で8ビット・メモリ操作命令を使用する場合は注意が必要です。

(2) 割り込み要求フラグ・レジスタ (MK1L)

割り込みマスク・フラグは、対応するマスカブル割り込み処理の許可／禁止を設定するフラグです。

MK1L, 1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。また, MK1LとMK1H, をあわせて16ビット・レジスタMK1として使用するときには, 16ビット・メモリ操作命令で設定します。

リセット信号の発生により, FFHになります。

備考 このレジスタへの書き込み命令を行った場合, 命令実行クロック数が2クロック長くなります。

略号: MK1L

7	6	5	4	3	2	1	0
TMMK03	TMMK02	TMMK01	TMMK00	IICAMK	SREMK1	SRMK1	STMK1 CSIMK10 IICMK10

xxMKx	割り込み処理の制御
0	割り込み処理許可
1	割り込み処理禁止

4.5 シリアル・アレイ・ユニットの設定概要

シリアル・アレイ・ユニット(UART送受信)を用いる際の各レジスタ設定の流れを以下に示します。

① SAU へのクロック供給開始

周辺イネーブル・レジスタ0(PER0)
クロック供給

略号：PER0

7	6	5	4	3	2	1	0
RTCEN	0	ADCEN	IICAEN	0	SAU0EN	0	0
x	0	x	x	0	1	0	0

SAU0EN	シリアル・アレイ・ユニットの入力クロックの制御
0	入力クロック供給停止
1	入力クロック供給

【使用例】

上記、赤字部分と同設定にする場合
(本サンプル・プログラム・ソースと同一内容)

① アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

SET1 SAU0EN

② C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

SAU0EN = 1;

② ウェイト

入力クロックを供給してから f_{CLK} の4クロック以上のウェイト

① アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

NOP
NOP
NOP
NOP

② C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

NOP();
NOP();
NOP();
NOP();

③ シリアル・クロックの選択

シリアル・クロック選択レジスタ0 (SPS0)
動作クロックの設定

略号：SPS0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000
0	0	0	0	0	0	0	0	x	x	x	x	0	0	1	0

PRS 003	PRS 002	PRS 001	PRS 000	動作クロック(CK0p)の選択				
				f _{CLK} =2MHz	f _{CLK} =5MHz	f _{CLK} =10MHz	f _{CLK} =20MHz	
0	0	0	0	f _{CLK}	2MHz	5MHz	10MHz	20MHz
0	0	0	1	f _{CLK} /2	1MHz	2.5MHz	5MHz	10MHz
0	0	1	0	f _{CLK} /2 ²	500kHz	1.25MHz	2.5MHz	5MHz
0	0	1	1	f _{CLK} /2 ³	250kHz	625kHz	1.25MHz	2.5MHz
0	1	0	0	f _{CLK} /2 ⁴	125kHz	313kHz	625kHz	1.25MHz
0	1	0	1	f _{CLK} /2 ⁵	62.5kHz	156kHz	313KHz	625KHz
0	1	1	0	f _{CLK} /2 ⁶	31.3kHz	78.1kHz	156kHz	313kHz
0	1	1	1	f _{CLK} /2 ⁷	15.6kHz	39.1kHz	78.1kHz	156kHz
1	0	0	0	f _{CLK} /2 ⁸	7.81kHz	19.5kHz	39.1kHz	78.1kHz
1	0	0	1	f _{CLK} /2 ⁹	3.91kHz	9.77kHz	19.5kHz	39.1kHz
1	0	1	0	f _{CLK} /2 ¹⁰	1.95kHz	4.88kHz	9.77kHz	19.5kHz
1	0	1	1	f _{CLK} /2 ¹¹	977Hz	2.44kHz	4.88kHz	9.77kHz
1	1	1	1	INTTM02				
上記以外				設定禁止				

【使用例】

上記，赤文字部分と同設定にする場合
(本サンプル・プログラム・ソースと同一内容)

①アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

MOV SPS0L , #02H

②C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

SPS0L = 0b00000010;

④ 送信チャネルの動作モード設定

シリアル・モード・レジスタ02 (SMR02)
 割り込み要因
 動作モード
 転送クロックの選択
 f_{MCK} の選択

略号：SMR02

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CK S02	CC S02	0	0	0	0	0	ST S02	0	SIS 020	1	0	0	MD 022	MD 021	MD 020
0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0

MD 020	チャンネル1の割り込み要因の選択
0	転送完了割り込み
1	バッファ空き割り込み

MD 022	MD 021	チャンネル2の動作モードの設定
0	0	CSIモード
0	1	UARTモード
1	0	簡易I ² Cモード
1	1	設定禁止

SIS 020	UARTモードでのチャンネル2の受信データのレベル反転の制御
0	立ち下りエッジをスタートビットとして検出します
1	立ち上がりエッジをスタートビットとして検出します

STS 02	スタート・トリガ要因の選択
0	ソフトウェア・トリガのみ有効
1	RxD端子の有効エッジ (UART受信時に選択)

CCS 02	チャンネル2の転送クロック (TCLK) の選択
0	CKS02ビットで指定した動作クロック f_{MCK} の分周クロック
1	SCK端子からの入力クロック

CKS 02	チャンネル2の動作クロック (f_{MCK}) の選択
0	SPS0レジスタで設定したプリスケアラ出力クロックCK00
1	SPS0レジスタで設定したプリスケアラ出力クロックCK01

【使用例】

上記，赤字部分と同設定にする場合
 (本サンプル・プログラム・ソースと同内容)

① アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOVW      AX      ,      #0022H
MOVW      SMR02   ,      AX
```

② C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SMR02 = 0b00000000000100010;
```

⑤ 受信チャンネルの動作モード設定

シリアル・モード・レジスタ03 (SMR03)
 割り込み要因
 動作モード
 転送クロックの選択
 f_{MCK} の選択

略号：SMR03

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CK S03	CC S03	0	0	0	0	0	ST S03	0	SIS 030	1	0	0	MD 032	MD 031	MD 030
0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0

MD 030	チャンネル3の割り込み要因の選択
0	転送完了割り込み
1	バッファ空き割り込み

MD 032	MD 031	チャンネル3の動作モードの設定
0	0	CSIモード
0	1	UARTモード
1	0	簡易I ² Cモード
1	1	設定禁止

SIS 030	UARTモードでのチャンネル3の受信データのレベル反転の制御
0	立ち下りエッジをスタートビットとして検出します
1	立ち上がりエッジをスタートビットとして検出します

STS 03	スタート・トリガ要因の選択
0	ソフトウェア・トリガのみ有効
1	RxD端子の有効エッジ (UART受信時に選択)

CCS 03	チャンネル3の転送クロック (TCLK) の選択
0	CKS03ビットで指定した動作クロック f_{MCK} の分周クロック
1	SCK端子からの入力クロック

CKS 03	チャンネル3の動作クロック (f_{MCK}) の選択
0	SPS0レジスタで設定したプリスケアラ出力クロックCK00
1	SPS0レジスタで設定したプリスケアラ出力クロックCK01

【使用例】

上記，赤字部分と同設定にする場合
 (本サンプル・プログラム・ソースと同内容)

① アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOVW      AX      ,      #0122H
MOVW      SMR03   ,      AX
```

② C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SMR03 = 0b00000000100100010;
```

⑥ 送信チャネルの通信動作設定

シリアル通信動作レジスタ02 (SCR02)
データ長の設定
データ転送順序
エラー割り込み信号のマスク可否
動作モード

略号：SCR02

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXE02	RXE02	DAP02	CKP02	0	EOC02	PTC021	PTC020	DIR02	0	SLC021	SLC020	0	DLS022	DLS021	DLS020
1	0	0	0	0	0	1	0	1	0	0	1	0	1	1	1

DL S022	DL S021	DL S020	CSIモードでのデータ長の設定
1	0	0	5ビット・データ長
1	1	0	7ビット・データ長
1	1	1	8ビット・データ長
その他			設定禁止

SLC021	SLC020	UARTモードでのストップ・ビットの設定
0	0	ストップ・ビットなし
0	1	ストップ・ビット長 = 1ビット
1	0	ストップ・ビット長 = 2ビット
1	1	設定禁止

DIR02	CSI, UARTモードでのデータ転送順序の選択
0	MSBファーストで入出力を行う
1	LSBファーストで入出力を行う

PTC 021	PTC 020	UARTモードでのパリティ・ビットの設定	
		送信動作	受信動作
0	0	パリティ・ビットを出力しない	パリティなしで受信
0	1	0パリティを出力	パリティ判定を行わない
1	0	偶数パリティを出力	偶数パリティとして判定を行う
1	1	奇数パリティを出力	奇数パリティとして判定を行う

EOC02	エラー割り込み信号 (INTSREx (x = 0, 1)) のマスク可否の選択
0	エラー割り込みINTSRExをマスクする
1	エラー割り込みINTSRExの発生を許可する

TXE02	RXE02	チャネル1の動作モードの設定
0	0	通信禁止
0	1	受信のみを行う
1	0	送信のみを行う
1	1	送受信を行う

【使用例】

上記、赤字部分と同設定にする場合
(本サンプル・プログラム・ソースと同内容)

①アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOVW    AX, #8297H
MOVW    SCR02, AX
```

②C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SCR02 = 0b1000001010010111;
```

⑦ 受信チャンネルの通信動作設定

シリアル通信動作レジスタ03 (SCR03)
データ長の設定
データ転送順序
エラー割り込み信号のマスク可否
動作モード

略号：SCR03

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXE03	RXE03	DAP03	CKP03	0	EOC03	PTC031	PTC030	DIR03	0	SLC031	SLC030	0	DLS032	DLS031	DLS030
0	1	0	0	0	1	1	0	1	0	0	1	0	1	1	1

DL S03 2	DL S03 1	DL S03 0	CSIモードでのデータ長の設定
1	0	0	5ビット・データ長
1	1	0	7ビット・データ長
1	1	1	8ビット・データ長
その他			設定禁止

SLC 031	SLC 030	UARTモードでのストップ・ビットの設定
0	0	ストップ・ビットなし
0	1	ストップ・ビット長 = 1ビット
1	0	ストップ・ビット長 = 2ビット
1	1	設定禁止

DIR0 3	CSI, UARTモードでのデータ転送順序の選択
0	MSBファーストで入出力を行う
1	LSBファーストで入出力を行う

PTC 031	PTC 030	UARTモードでのパリティ・ビットの設定	
		送信動作	受信動作
0	0	パリティ・ビットを出力しない	パリティなしで受信
0	1	0パリティを出力	パリティ判定を行わない
1	0	偶数パリティを出力	偶数パリティとして判定を行う
1	1	奇数パリティを出力	奇数パリティとして判定を行う

EOC03	エラー割り込み信号 (INTSREx (x = 0, 1)) のマスク可否の選択
0	エラー割り込みINTSRExをマスクする
1	エラー割り込みINTSRExの発生を許可する

TXE03	RXE03	チャンネル1の動作モードの設定
0	0	通信禁止
0	1	受信のみを行う
1	0	送信のみを行う
1	1	送受信を行う

【使用例】

上記、赤文字部分と同設定にする場合
(本サンプル・プログラム・ソースと同内容)

①アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOVW    AX,    #4617H
MOVW    SCR03, AX
```

②C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SCR03 = 0b0100011010010111;
```

⑧ 送信チャネル転送クロックの設定

シリアル・データ・レジスタ02 (SDR02)
転送クロック周波数 : 10kHz

略号 : SDR02

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

1	1	0	0	1	1	1	0								
---	---	---	---	---	---	---	---	--	--	--	--	--	--	--	--

SDR02[15:9]							動作クロック (f_{MCK}) の分周による転送クロック設定
0	0	0	0	0	0	0	$f_{MCK} / 2$
0	0	0	0	0	0	1	$f_{MCK} / 4$
0	0	0	0	0	1	0	$f_{MCK} / 6$
0	0	0	0	0	1	1	$f_{MCK} / 8$
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
1	1	1	1	1	1	0	$f_{MCK} / 254$
1	1	1	1	1	1	1	$f_{MCK} / 256$

【使用例】

上記，赤文字部分と同設定にする場合
(本サンプル・プログラム・ソースと同内容)

①アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOVW      AX      ,      #0CE00H
MOVW      SDR02   ,      AX
```

②C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SDR02 = 0b1100111000000000;
```

⑨ 受信転送クロックの設定

シリアル・データ・レジスタ03 (SDR03)
転送クロック周波数 : 10kHz

略号 : SDR03

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

1	1	0	0	1	1	1	0								
---	---	---	---	---	---	---	---	--	--	--	--	--	--	--	--

SDR03[15:9]							動作クロック (f_{MCK}) の分周による転送クロック設定
0	0	0	0	0	0	0	$f_{MCK} / 2$
0	0	0	0	0	0	1	$f_{MCK} / 4$
0	0	0	0	0	1	0	$f_{MCK} / 6$
0	0	0	0	0	1	1	$f_{MCK} / 8$
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
1	1	1	1	1	1	0	$f_{MCK} / 254$
1	1	1	1	1	1	1	$f_{MCK} / 256$

【使用例】

上記、赤文字部分と同設定にする場合
(本サンプル・プログラム・ソースと同内容)

①アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOVW          AX      ,      #0CE00H
MOVW          SDR03   ,      AX
```

②C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SDR03 = 0b1100111000000000;
```

⑩ 出力レベルの設定

シリアル出力レベル・レジスタ0 (SOL0)
非反転出力

略号 : SOL0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	SOL02	0	SOL00
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x

SOL02	UARTモードでのチャンネル2の送信データのレベル反転選択
0	通信データは、そのまま出力
1	通信データは、反転して出力

【使用例】

上記、赤文字部分と同設定にする場合
(本サンプル・プログラム・ソースと同内容)

①アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOV          SOL0L   ,      #00H
```

②C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SOL0L = 0b00000000;
```


⑪ 初期出力レベルの設定

シリアル出力レジスタ0 (SO0)
初期出力1

略号：SO0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	1	CKO 02	CKO 01	CKO 00	0	0	0	0	1	SO 02	SO 01	SO 00
0	0	0	0	1	0	x	x	0	0	0	0	1	1	x	x

SO02	チャンネル2のシリアル・データ出力
0	シリアル・データ出力値が“0”
1	シリアル・データ出力値が“1”

CKO02	チャンネル2のシリアル・クロック出力
0	シリアル・クロック出力値が“0”
1	シリアル・クロック出力値が“1”

【使用例】

上記、赤字部分と同設定にする場合
(本サンプル・プログラム・ソースと同一内容)

①アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOVW    AX, #080CH
MOVW    SO0, AX
```

②C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SO0 = 0b0000100000001100;
```

⑪ 対象チャンネルのデータ出力許可

シリアル出力許可レジスタ0 (SOE0)
出力許可

略号：SOE0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	SOE 02	SOE 01	SOE 00
0	0	0	0	0	0	0	0	0	0	0	0	0	1	x	x

SOE02	チャンネル2のシリアル出力許可/停止
0	シリアル通信動作による出力停止
1	シリアル通信動作による出力許可

【使用例】

上記、赤字部分と同設定にする場合
(本サンプル・プログラム・ソースと同一内容)

①アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOV     SOE0L, #04H
```

②C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SOE0 = 0b00000100;
```

⑫ ポート設定

ポート・レジスタ3 (P3)
 ポート・モード・レジスタ3 (PM3)
 転送クロック用, 送信データ用, 受信データ用
 にそれぞれポートを設定します。

略号: P3

7	6	5	4	3	2	1	0
0	0	0	0	P33	P32	P31	P30
0	0	0	0	x	x	x	1

P30	出力データの制御 (出力モード時)	入力データの読み出し (入力モード時)
0	0を出力	ロウ・レベルを入力
1	1を出力	ハイ・レベルを入力

略号: PM3

7	6	5	4	3	2	1	0
1	1	1	1	PM33	PM32	PM31	PM30
1	1	1	1	x	x	1	0

PM30	PM70の入出力モードの選択
0	出力モード (出力バッファ・オン)
1	入力モード (出力バッファ・オフ)

PM31	PM71の入出力モードの選択
0	出力モード (出力バッファ・オン)
1	入力モード (出力バッファ・オフ)

【使用例】

上記, 赤字部分と同設定にする場合
 (本サンプル・プログラム・ソースと同一内容)

① アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
MOV      P3      ,      #01H
MOV      PM3     ,      #0F2H
```

② C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
P3 = 0b00000001;
PM3 = 0b11110010;
```

⑬ 通信待機状態に遷移

シリアル・チャネル開始レジスタ0 (SS0)
動作開始

略号：SS0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	SS03	SS02	SS01	SS00
0	0	0	0	0	0	0	0	0	0	0	0	1	1	x	x

SS0n	チャネルnの動作開始トリガ
0	トリガ動作せず
1	SE0nに1をセットし、通信待機状態に遷移する

【使用例】

上記、赤字部分と同設定にする場合
(本サンプル・プログラム・ソースと同一内容)

① アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

MOV SS0L, #0CH

② C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

SS0L = 0b00001100;

4.6 割り込みの設定概要

本サンプル・プログラムで割り込みを使用する際の割り込み初期設定の流れを以下に示します。

① 割り込みの設定

割り込み要求フラグ・レジスタ (IF1L)
割り込みマスク・フラグ・レジスタ (MK1L)

略号：IF1L

7	6	5	4	3	2	1	0
TMIF03	TMIF02	TMIF01	TMIF00	IICAIF	SREIF1	SRIF1	STIF1 CSIIF10 IICIF10
x	x	x	x	x	0	0	x

SRIF1	割り込み要求フラグ
0	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

SREIF1	割り込み要求フラグ
0	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

略号：MK1L

7	6	5	4	3	2	1	0
TMMK03	TMMK02	TMMK01	TMMK00	IICAMK	SREMK1	SRMK1	STMK1 CSIMK10 IICMK10
x	x	x	x	x	0	0	x

CSIMK1	割り込み処理の制御
0	割り込み処理許可
1	割り込み要求信号が発生し、割り込み要求状態

SREMK1	割り込み処理の制御
0	割り込み処理許可
1	割り込み要求信号が発生し、割り込み要求状態

【使用例】

上記，赤字部分と同設定にする場合
(本サンプル・プログラム・ソースと同一内容)

① アセンブリ言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
CLR1      SRIF1
CLR1      SREIF1
CLR1      SRMK1
CLR1      SREMK1
```

② C言語の場合 (78K0R/KE3-Lマイクロコントローラ使用時)

```
SRIF1 = 0;
SREIF1 = 0;
SRMK1 = 0;
SREMK1 = 0;
```

第5章 PM+を用いたHEXファイルの生成

この章では、PM+とダウンロードしたC言語用のファイルを用い、サンプル・プログラムからHEXファイルに生成する方法を説明します。

5.1 ダウンロードファイルの解説

ダウンロードした各種ファイルとの説明をします。

(1) C言語版

	ファイル名	内容
	78K0RKx3-L_sample_program.prw	統合開発環境 PM+用ワーク・スペース・ファイル
	78K0RKx3-L_sample_program.prj	統合開発環境 PM+用設定データ
	Kx3-L_UART.c	UART通信のC言語ソース・ファイル
	OP.asm	オプション・バイトのアセンブリ言語ソース・ファイル

(2) アセンブラ版

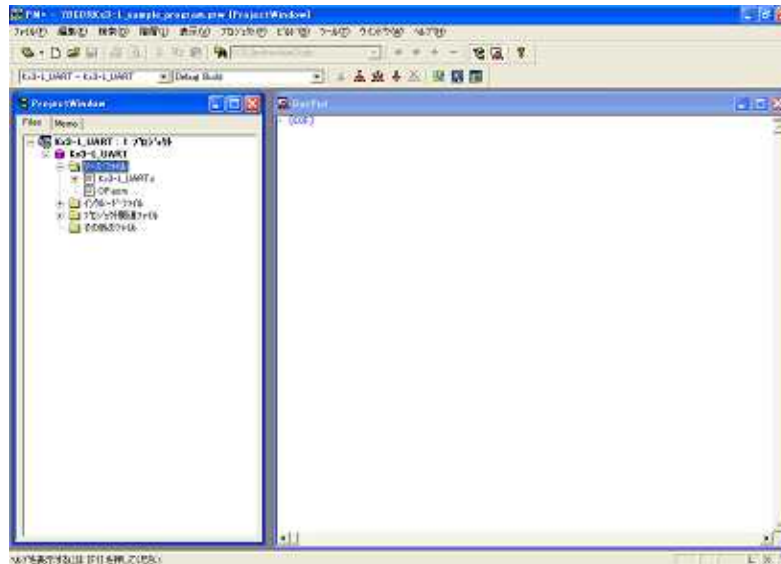
	ファイル名	内容
	78K0RKx3-L_sample_program.prw	統合開発環境 PM+用ワーク・スペース・ファイル
	78K0RKx3-L_sample_program.prj	統合開発環境 PM+用設定データ
	Kx3-L_UART.asm	UART通信のアセンブリ言語ソース・ファイル
	OP.asm	オプション・バイトのアセンブリ言語ソース・ファイル

5.2 サンプル・プログラムのHEXファイル生成（ビルド）

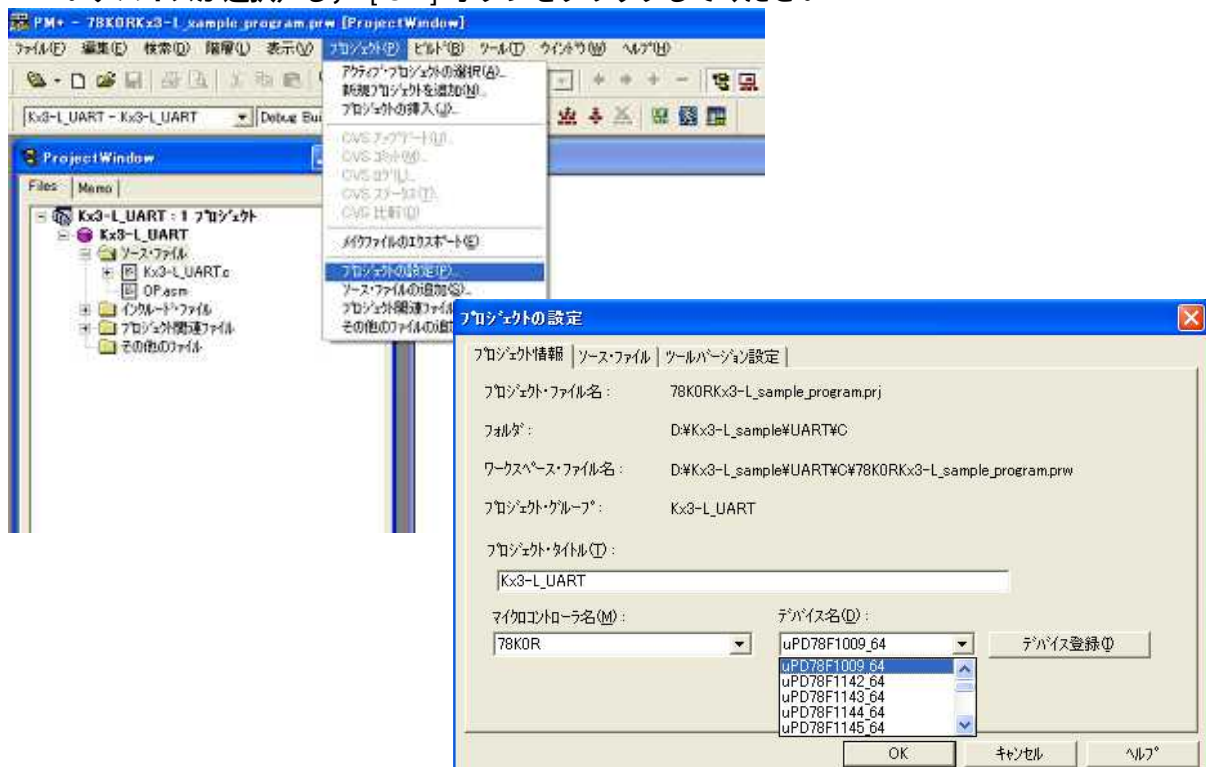
サンプル・プログラムからHEXファイルに生成するために、PM+を用いてサンプル・プログラムをビルドする必要があります。ここでは、 でダウンロードしたC言語版のファイルを用いて、統合開発環境PM+にてビルドしてから、HEXファイルを生成するまでの動作の一例を説明します。

PM+操作方法の詳細については、[PM+ プロジェクト・マネージャ ユーザーズ・マニュアル](#)を参照してください。

- (1) ダウンロードしたファイルを解凍し、「78K0RKx3-L_sample_program.prw」をダブルクリックしてください。ワークスペースが開き、その中にソース・ファイルが自動的に読み込まれます。



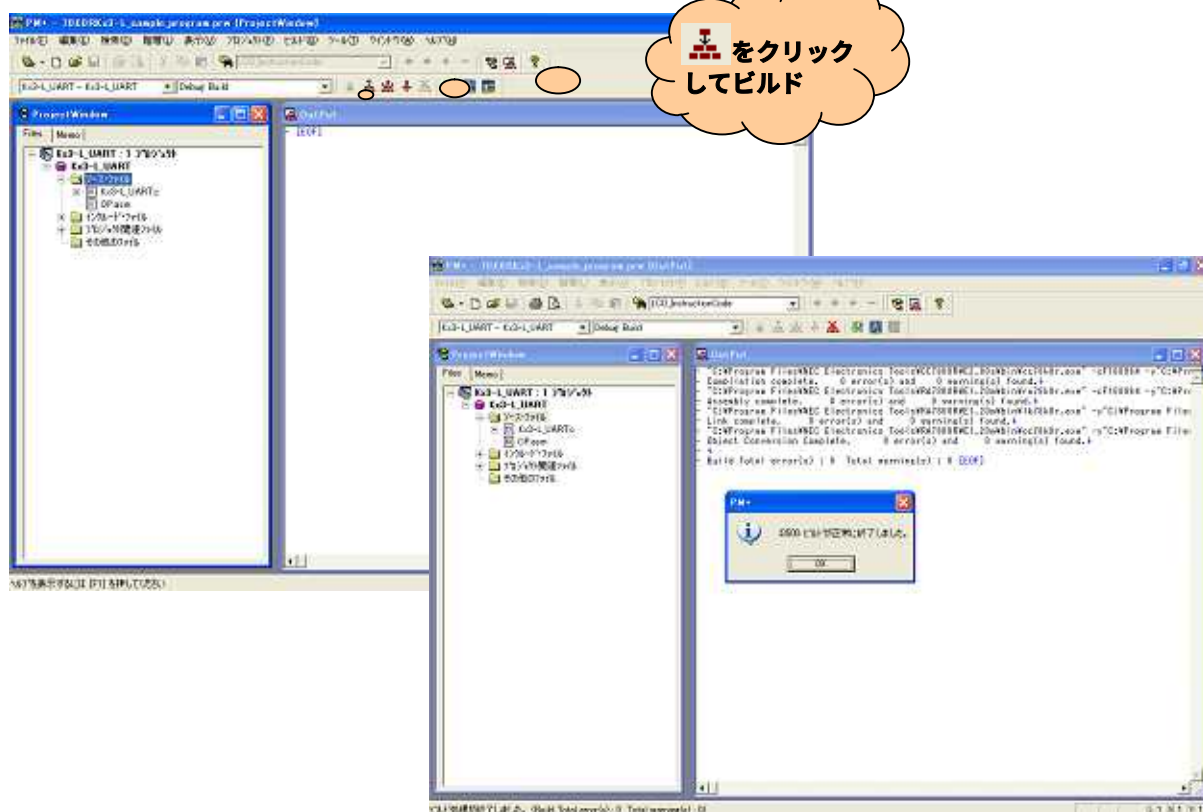
- (2) [プロジェクト] → [プロジェクトの設定] を選択してください。[プロジェクトの設定] 画面が表示されたら、使用するデバイス名を選択（デフォルトでは、ROM/RAMサイズの最も大きいデバイスが選択）し、[OK] ボタンをクリックしてください



- (3) [ツール] → [オブジェクトコンバータオプションの設定] を選択してください。[オブジェクトコンバータオプションの設定] 画面が表示されたら, [出力1] タグページが表示されているのを確認し, その中の出力ファイル名に任意のファイル名+拡張子 (.HEX) を入力し, [OK] をクリックします。



- (4) PM+画面の中央上付近の  をクリックしてください。自動でビルドが実行され, ソース・ファイルの「Kx3-L_UART.c」と「OP.asm」からHEXファイルが生成され, 「I3500:ビルドが正常に終了しました」というメッセージ画面が表示されます。
[OK]をクリックしてビルドを終了します。



5.3 開発環境のダウンロード，インストール

78K0R/Kx3-Lマイクロコントローラの開発ツールのフリーツールは，次のサイトより入手可能です。

→<http://www.necel.com/micro/ja/freesoft/78k0r/index.html>

「RA78K0R」「CC78K0R」「78K0R/Kx3-L用デバイス・ファイル」の3ファイルをダウンロードし，インストールすることで，サンプル・プログラムの動作確認が可能となります。

ダウンロード，インストールは，上記サイトの画面および説明に従って，行ってください。

- 備考1. PM+は，RA78K0Rに同封されています。
2. ダウンロード後，登録したEメール・アドレスに，RA78K0R，CC78K0RのプロダクトIDが送付されます。このプロダクトIDは，各ツールのインストール時に必要となります。

第6章 関連資料

資 料 名		和文／英文
78K0R/Kx3-L ユーザーズ・マニュアル		PDF
78K0Rシリーズ 命令編 ユーザーズ・マニュアル		PDF
RA78K0R アセンブラ・パッケージ ユーザーズ・マニュアル	言語編	PDF
	操作編	PDF
CC78K0R Cコンパイラ ユーザーズ・マニュアル	言語編	PDF
	操作編	PDF
PM+ プロジェクト・マネージャ ユーザーズ・マニュアル		PDF
SM+ システム・シミュレータ 操作編 ユーザーズ・マニュアル		PDF

付録A プログラム・リスト

プログラム・リスト例として、78K0R/KE3-Lマイクロコントローラのソース・プログラムを次に示します。

● Kx3-L_UART.asm (アセンブリ言語版)

```

;*****
;
;      NEC Electronics      78K0R/KE3-Lシリーズ
;
;*****
;      78K0R/KE3-Lシリーズ      サンプル・プログラム (シリアル・アレイ・ユニット)
;*****
;      UART通信
;*****
; 【履歴】
;      2009.01.--      新規作成
;*****
;
; 【概要】
;本サンプル・プログラムでは、シリアル・アレイ・ユニットのUART通信機能の使用例を
;示すとともに、対向機器から送られてくるASCII文字を解析し、応答処理を行います。
;受信処理はバッファを用いず受信データを直接処理し、通信は半二重で行います。
;
;
; <初期設定の主な内容>
;   (オプション・バイトでの設定)
;   ・ウォッチドッグ・タイマの動作禁止
;   ・高速内蔵発振回路に8MHz/20MHzを選択
;   ・LVIデフォルト・スタート機能動作
;   ・オンチップ・デバッグを動作許可に設定
;   (リセット解除後の初期化処理での設定)
;   ・入出力ポートの設定
;   ・低電圧検出回路の機能を使用し、2.7V以上の電源電圧を確保
;   ・CPU/周辺ハードウェア・クロックを高速内蔵発振クロック動作の8MHzに設定
;   ・X1/XT1発振回路の停止
;
; <シリアル・アレイ・ユニットの設定>
;   ・チャンネル2を送信、チャンネル3を受信に設定
;   ・ボー・レート：9600 bps
;   ・データのキャラクタ長：8ビット

```

```

; ・パリティ・ビット指定：偶数パリティ
; ・ストップ・ビット数：1ビット
; ・先頭ビットの指定：LSBファースト
;
;<メイン処理の主な内容>
; ・シリアル・チャネル2,3を通信待機状態に遷移
;
;<シリアル・データ送信処理の主な内容>
; ・シリアル・データの送信
;
;<割り込み処理(INTSR1)の主な内容>
; ・受信データの判定
; ・シリアル・データ送信処理の実行
;
;<割り込み処理(INTSRE1)の主な内容>
; ・エラー内容の判定
; ・シリアル・データ送信処理の実行
;
;
;<受信データと応答する送信データ>
;
; +-----+-----+
; | 受信データ |      応答する送信データ      |
; +-----+-----+
; |  T(54H)   |  O(4FH), K(4BH), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
; |  t(74H)   |  o(6FH), k(6BH), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
; |  上記以外 |  U(55H), C(43H), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
;
;
;<エラー発生時の送信データ>
;
; +-----+-----+
; |   発生したエラー   |      応答する送信データ      |
; +-----+-----+
; | パリティ・エラー   |  P(50H), E(45H), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
; | フレーミング・エラー | F(46H), E(45H), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
; | オーバーラン・エラー | O(4FH), E(45H), "CR"(0DH), "LF"(0AH) |
; +-----+-----+

```

```

;
;
; <入出力ポートの設定>
; 入力ポート：P31(RXD1)
; 出力ポート：P30(TXD1)
; ※未使用のポートで出力に設定できるものは全て出力ポートに設定しておく
;
;
;*****
;

;=====
;
;
;   ベクタ・テーブルの設定
;
;=====
TVECT1      CSEG      AT      00000H
            DW      RESET_START      ;00000H  RESET入力,POC,LVI,WDT,TRAP
TVECT2      CSEG      AT      00004H
            DW      IINIT      ;00004H  INTWDTI
            DW      IINIT      ;00006H  INTLVI
            DW      IINIT      ;00008H  INTP0
            DW      IINIT      ;0000AH  INTP1
            DW      IINIT      ;0000CH  INTP2
            DW      IINIT      ;0000EH  INTP3
            DW      IINIT      ;00010H  INTP4
            DW      IINIT      ;00012H  INTP5
TVECT3      CSEG      AT      00016H
            DW      IINIT      ;00016H  INTSR3
            DW      IINIT      ;00018H  INTSRE3
            DW      IINIT      ;0001AH  INTDMA0
            DW      IINIT      ;0001CH  INTDMA1
            DW      IINIT      ;0001EH  INTST0/INTCSI00
            DW      IINIT      ;00020H  INTSR0/INTCSI01
            DW      IINIT      ;00022H  INTSRE0
            DW      IINIT      ;00024H  INTST1/INTCSI10/INTIIC10
            DW      IINTSR1      ;00026H  INTSR1
            DW      IINTSRE1      ;00028H  INTSRE1
            DW      IINIT      ;0002AH  INTIICA
            DW      IINIT      ;0002CH  INTTM00
            DW      IINIT      ;0002EH  INTTM01
            DW      IINIT      ;00030H  INTTM02
            DW      IINIT      ;00032H  INTTM03

```

```

        DW      IINIT                                ;00034H  INTAD
        DW      IINIT                                ;00036H  INTRTC
        DW      IINIT                                ;00038H  INTRTCI
        DW      IINIT                                ;0003AH  INTKR
TVECT4      CSEG      AT      00040H
        DW      IINIT                                ;00040H  INTSRE2
        DW      IINIT                                ;00042H  INTTM04
        DW      IINIT                                ;00044H  INTTM05
        DW      IINIT                                ;00046H  INTTM06
        DW      IINIT                                ;00048H  INTTM07
        DW      IINIT                                ;0004AH  INTP6
        DW      IINIT                                ;0004CH  INTP7

;=====
;
;      応答メッセージの設定
;
;=====
XMSG      CSEG      AT      1000H
TMSGOK:                                ;OKメッセージ
        DB 'OK'
        DB 0DH,0AH
TMSGOK2:                                ;okメッセージ
        DB 'ok'
        DB 0DH,0AH
TMSGUC:                                ;UCメッセージ
        DB 'UC'
        DB 0DH,0AH
TMSGFE:                                ;フレーミング・エラー・メッセージ
        DB 'FE'
        DB 0DH,0AH
TMSGPE:                                ;パリティ・エラー・メッセージ
        DB 'PE'
        DB 0DH,0AH
TMSGOE:                                ;オーバラン・エラー・メッセージ
        DB 'OE'
        DB 0DH,0AH

;=====
;
;      スタック領域の確保
;

```

```

;=====
DSTK   DSEG   BASEP
STACKEND:
        DS     20H                      ;スタック領域を32バイト確保
STACKTOP:                      ;スタック領域の先頭アドレス

;=====
;
;   RAMの定義
;
;=====
DUART   DSEG   SADDR
RUASTT:      DS     1                      ;UART通信ステータス

XMAIN   CSEG   UNIT
;*****
;
;   不要な割り込み要因による割り込み処理
;
;*****
IINIT:
;   不要な割り込みが発生した場合、ここに分岐します。
;   ここでは何も処理をしないで元の処理に戻ります

        RETI

;*****
;
;   リセット解除後の初期化処理
;
;*****
RESET_START:

;-----
;   割り込み禁止
;-----

        DI

;-----
;   レジスタ・バンク設定
;-----

        SEL     RBO

```

```

;-----
;   スタック・ポインタの設定
;-----
MOVW    SP,    #LOWW STACKTOP    ;スタック・ポインタを設定

;-----
;   入出力ポートの設定
;-----
CALL     !!SINIPOINT    ;出力に設定できるものは全て出力ポート
に設定

;-----
;   低電圧検出
;-----
CALL     !!SINILVI    ;2.7V以上の電源電圧を確保

;-----
;   クロック周波数の設定
;-----
CALL     !!SINICLK    ;高速内蔵発振クロックを8MHzで動作

;-----
;   シリアル・アレイ・ユニットの設定
;-----
;シリアル・アレイ・ユニット初期設定
SET1     SAU0EN    ;シリアル・アレイ・ユニットの入力クロック供給

NOP      ;4クロック分ウエイト
NOP
NOP
NOP

MOV      SPS0L,    #00000010B    ;シリアル・クロック選択レジスタ0
;|||+++------ PRS003-PRS000 ※fCLK/2^2を使用
;++++----- PRS013-PRS010
;
;   [動作クロック(CK00/CK01)の選択]
;
;   0000:fCLK
;   0001:fCLK/2
;   0010:fCLK/2^2
;   0011:fCLK/2^3
;   0100:fCLK/2^4
;   0101:fCLK/2^5
;   0110:fCLK/2^6

```

```

;
; 0111:fCLK/2^7
;
; 1000:fCLK/2^8
;
; 1001:fCLK/2^9
;
; 1010:fCLK/2^10
;
; 1011:fCLK/2^11
;
; 1111:INTTM02
;
; 上記以外:設定禁止

;送信側
MOVW    AX,    #0000000000100010B ;シリアル・モード・レジスタ02
MOVW    SMR02, AX
;|||||||||+ MD020
;||||||||| [チャンネル2の割り込み要因の選択]
;||||||||| 0:転送完了割り込み
;||||||||| 1:バッファ空き割り込み
;|||||||||+- MD022,MD021
;||||||||| [チャンネル2の動作モードの設定]
;||||||||| 00:CSIモード
;||||||||| 01:UARTモード
;||||||||| 10:簡易I2Cモード
;||||||||| 11:設定禁止
;|||||||||+---- 必ず100に設定
;|||||||||+----- SIS020
;||||||||| [UARTモードでのチャンネルnの受信データのレベル反
転の制御]
;||||||||| 0:立ち下がりエッジをスタートビットと
して検出します。
;||||||||| 入力される通信データは,そのまま取り込
まれます。
;||||||||| 1:立ち上がりエッジをスタートビットと
して検出します。
;||||||||| 入力される通信データは,反転して取り込
まれます。
;|||||||||+----- 必ず0に設定
;|||||||||+----- STS02
;||||||||| [スタート・トリガ要因の選択]
;||||||||| 0:ソフトウェア・トリガのみ有効
;||||||||| 1:RxD端子の有効エッジ
;||+----+----- 必ず0に設定
;|+----- CCS02
;| [チャンネル2の転送クロック(TCLK)の選択]
;| 0:CKS02ビットで指定した動作クロック
MCKKの分周クロック

```



```

;| 1:SCK端子からの入力クロック
;+----- CKS02
; [チャンネル2の動作クロック(MCK)の選択]
; 0:PRSレジスタで設定したプリスケアラ出
カクロックCK00
; 1:PRSレジスタで設定したプリスケアラ出
カクロックCK01

;受信側
MOVW AX, #0000000100100010B ;シリアル・モード・レジスタ03
MOVW SMR03, AX
;|||||||||+ MD030
;||||||||| [チャンネル3の割り込み要因の選択]
;||||||||| 0:転送完了割り込み
;||||||||| 1:バッファ空き割り込み
;|||||||||++- MD032,MD031
;||||||||| [チャンネル3の動作モードの設定]
;||||||||| 00:CSIモード
;||||||||| 01:UARTモード
;||||||||| 10:簡易I2Cモード
;||||||||| 11:設定禁止
;|||||||||+++--- 必ず100に設定
;|||||||||+----- SIS030
;||||||||| [UARTモードでのチャンネルnの受信データのレベル反
転の制御]
;||||||||| 0:立ち下がりエッジをスタートビットと
して検出します。
;||||||||| 入力される通信データは,そのまま取り込
まれます。
;||||||||| 1:立ち上がりエッジをスタートビットと
して検出します。
;||||||||| 入力される通信データは,反転して取り込
まれます。
;|||||||||+----- 必ず0に設定
;|||||||||+----- STS03
;||||||||| [スタート・トリガ要因の選択]
;||||||||| 0:ソフトウェア・トリガのみ有効
;||||||||| 1:RxD端子の有効エッジ
;||+++++----- 必ず0に設定
;|+----- CCS03
;| [チャンネル3の転送クロック(TCLK)の選択]
;| 0:CKS03ビットで指定した動作クロック
MCKの分周クロック

```

```

;| 1:SCK端子からの入力クロック
;+----- CKS03
; [チャンネル2の動作クロック(MCK)の選択]
; 0:PRSレジスタで設定したプリスケアラ出
カクロックCK00
; 1:PRSレジスタで設定したプリスケアラ出
カクロックCK01

;送信側
MOVW AX, #1000001010010111B;シリアル通信動作設定レジスタ02
MOVW SCR02, AX
;|||||+++ DLS022-DLS020
;||||| [CSI, UARTモードでのデータ長の設定]
;||||| 100:5ビット・データ長
;||||| 110:7ビット・データ長
;||||| 111:8ビット・データ長
;||||| 上記以外:設定禁止
;|||||+--- 必ず0に設定
;|||||++---- SLC021,SLC020
;||||| [UARTモードでのストップ・ビットの設定]
;||||| 00:ストップ・ビットなし
;||||| 01:ストップ・ビット長 = 1ビット
;||||| 10:ストップ・ビット長 = 2ビット
;||||| 11:設定禁止
;|||||+----- 必ず0に設定
;|||||+----- DIR02
;||||| [CSI, UARTモードでのデータ転送順序の選択]
;||||| 0:MSBファーストで入出力を行う
;||||| 1:LSBファーストで入出力を行う
;|||||++----- PTC021,PTC020
;||||| [UARTモードでのパリティ・ビットの設定]
;||||| 00:パリティ・ビットを出力しない
;||||| 01:0パリティを出力
;||||| 10:偶数パリティを出力
;||||| 11:奇数パリティを出力
;|||||+----- EOC02
;||||| [エラー割り込み信号のマスク可否の選択]
;||||| 0:エラー割り込みをマスクする
;||||| 1:エラー割り込みの発生を許可する
;|||||+----- 必ず0に設定
;||++----- DAP02,CKP02
;|| [CSIモードでのデータとクロックの位相選択]
;|| 00:UARTモード時には、必ず00に設定

```

```

; ++----- TXE02,RXE02
;
; [チャンネル2の動作モードの設定]
;
; 00:通信を開始しない
;
; 01:受信のみ
;
; 10:送信のみ
;
; 11:送受信を行う

;受信側
MOVW AX, #0100011010010111B;シリアル通信動作設定レジスタ03
MOVW SCR03, AX
; ||||| ||||| +++ DLS032-DLS030
; ||||| ||||| [CSI,UARTモードでのデータ長の設定]
; ||||| ||||| 100:5ビット・データ長
; ||||| ||||| 110:7ビット・データ長
; ||||| ||||| 111:8ビット・データ長
; ||||| ||||| 上記以外:設定禁止
; ||||| ||||| +--- 必ず0に設定
; ||||| ||||| ++---- SLC031,SLC030
; ||||| ||||| [UARTモードでのストップ・ビットの設定]
; ||||| ||||| 00:ストップ・ビットなし
; ||||| ||||| 01:ストップ・ビット長 = 1ビット
; ||||| ||||| 10:ストップ・ビット長 = 2ビット
; ||||| ||||| 11:設定禁止
; ||||| ||||| +----- 必ず0に設定
; ||||| ||||| +----- DIR03
; ||||| ||||| [CSI,UARTモードでのデータ転送順序の選択]
; ||||| ||||| 0:MSBファーストで入出力を行う
; ||||| ||||| 1:LSBファーストで入出力を行う
; ||||| ||||| ++----- PTC031,PTC030
; ||||| ||||| [UARTモードでのパリティ・ビットの設定]
; ||||| ||||| 00:パリティなしで受信
; ||||| ||||| 01:パリティ判定を行わない
; ||||| ||||| 10:偶数パリティとして判定を行う
; ||||| ||||| 11:奇数パリティとして判定を行う
; ||||| ||||| +----- EOC03
; ||||| ||||| [エラー割り込み信号のマスク可否の選択]
; ||||| ||||| 0:エラー割り込みをマスクする
; ||||| ||||| 1:エラー割り込みの発生を許可する
; ||||| ||||| +----- 必ず0に設定
; ||||| ||||| ++----- DAP03,CKP03
; ||||| ||||| [CSIモードでのデータとクロックの位相選択]
; ||||| ||||| 00:UARTモード時には、必ず00に設定
; ++----- TXE03,RXE03

```

```

;                                     [チャンネル3の動作モードの設定]
;
;                                     00:通信を開始しない
;                                     01:受信のみ
;                                     10:送信のみ
;                                     11:送受信を行う
;

;送信側
MOVW    AX,      #1100111000000000B;シリアル・データ・レジスタ02
MOVW    SDR02,   AX
;|||||+----- 必ず0に設定
;++++++----- SDR02[15:9]
;                                     [動作クロック(MCK)の分周による転送クロック設
定]

;                                     0000000:MCK/2(UART使用時設定禁止)
;                                     0000001:MCK/4(UART使用時設定禁止)
;                                     0000010:MCK/6
;                                     :      :
;                                     :      :
;                                     1111110:MCK/254
;                                     1111111:MCK/256
; ※1100111:MCK/208 = 2MHz/208 = 9615.38bps
;  目標ボー・レート9600bpsとの誤差 = +0.16%

;受信側
MOVW    AX,      #1100111000000000B ;シリアル・データ・レジスタ03
MOVW    SDR03,   AX
;|||||+----- 必ず0に設定
;++++++----- SDR03[15:9]
;                                     [動作クロック(MCK)の分周による転送クロック設
定]

;                                     0000000:MCK/2(UART使用時設定禁止)
;                                     0000001:MCK/4(UART使用時設定禁止)
;                                     0000010:MCK/6
;                                     :      :
;                                     :      :
;                                     1111110:MCK/254
;                                     1111111:MCK/256
; ※1100111:MCK/208 = 2MHz/208 = 9615.38bps
;  目標ボー・レート9600bpsとの誤差 = +0.16%

;送信側
MOV      SOL0L,  #00000000B          ;シリアル出力レベル・レジスタ0
;                                     +----- SOL02

```

； [UARTモードでのチャンネル2の送信データのレベル反
 転の選択]

； 0: 通信データは、そのまま出力されます
 ； 1: 通信データは、反転して出力されます

```
MOVW    AX,      #0000100000001100B;シリアル出力レジスタ0
MOVW    S00,     AX
；      +--- S002
；      [チャンネル2のシリアル・データ出力]
；      0:シリアル・データ出力値が"0"
；      1:シリアル・データ出力値が"1"
```

```
MOV     SOE0L,   #00000100B          ;シリアル出力許可レジスタ0
；      +----- SOE02
；      [チャンネル2のシリアル出力許可/停止]
；      0: シリアル通信動作による出力停止
；      1: シリアル通信動作による出力許可
```

;シリアル・アレイ・ユニットの割り込み許可

;UART受信割り込み

```
CLR1    SRIF1          ;INTSR1割り込み要求クリア
CLR1    SRMK1          ;INTSR1割り込み処理許可
```

;UARTエラー割り込み

```
CLR1    SREIF1         ;INTSRE1割り込み要求クリア
CLR1    SREMK1         ;INTSRE1割り込み処理許可
```

```
;-----
;      割り込み許可
;      (割り込みを使用する場合はこのタイミングで許可します。)
;-----
```

```
EI          ;割り込み許可
```

```
BR       MAIN_LOOP      ;メイン・ループへ
```

```
;*****
;
;      入出力ポートの設定
;
;*****
```

SINIPOINT:

```

;-----
;   デジタル入出力の設定
;-----
      MOV      ADPC,      #00010000B      ;A/Dポート・コンフィギュレーション・レ
ジスタ
      ;|||+++++----- ADPC4-ADPC0
      ;|||                                [アナログ入力(A)／デジタル入出力(D)の切り替え]
      ;|||                                +++++-----
ANI11-ANI18/P153-P150
      ;|||                                |||+++++-----
ANI7-ANI10/P27-P20
      ;|||                                00000:AAAAAAAAAAAA
      ;|||                                00001:AAAAAAAAAAD
      ;|||                                00010:AAAAAAAAAAD
      ;|||                                00011:AAAAAAAAADD
      ;|||                                00100:AAAAAAAADDDD
      ;|||                                00101:AAAAAAAADDDD
      ;|||                                00110:AAAAAADDDDD
      ;|||                                00111:AAAAAADDDDD
      ;|||                                01000:AAAADDDDDDD
      ;|||                                01001:AAADDDDDDDDD
      ;|||                                01010:AADDDDDDDDD
      ;|||                                01011:ADDDDDDDDDDD
      ;|||                                10000:DDDDDDDDDDDD
      ;+++----- 必ず0に設定

;-----
;   ポート0の設定
;-----
      MOV      P0,      #00000000B      ;P00-P01の出力ラッチLow
      MOV      PM0,     #11111100B      ;P00-P01を出力ポートに設定
                                          ;P00-P01:未使用

;-----
;   ポート1の設定
;-----
      MOV      P1,      #00000000B      ;P10-P17の出力ラッチLow
      MOV      PM1,     #00000000B      ;P10-P17を出力ポートに設定
                                          ;P10-P17:未使用

;-----
;   ポート2の設定
;-----

```

```

MOV    P2,    #00000000B                ;P20-P27の出力ラッチLow
MOV    PM2,    #00000000B                ;P20-P27を出力ポートに設定
                                           ;P20-P27:未使用

;-----
;   ポート3の設定
;-----
MOV    P3,    #00000001B                ;P31-P33の出力ラッチLow,
                                           ;P30の出力ラッチHigh
MOV    PM3,    #11110010B                ;P31を入力, P30,P32,P33を出力ポートに
設定
MOV    POM3,    #00000011B                ;P30,P31をN-chオープン・ドレイン出力
                                           ;(VDD耐圧)モードに設定
                                           ;P30:UART通信の送信用に使用する
                                           ;P31:UART通信の受信用に使用する
                                           ;P32,P33:未使用

;-----
;   ポート4の設定
;-----
MOV    P4,    #00000000B                ;P40-P43の出力ラッチLow
MOV    PM4,    #11110000B                ;P40-P43を出力ポートに設定
                                           ;P40-P43:未使用

;-----
;   ポート5の設定
;-----
MOV    P5,    #00000000B                ;P50-P53の出力ラッチLow
MOV    PM5,    #11110000B                ;P50-P53を出力ポートに設定
                                           ;P50-P53:未使用

;-----
;   ポート6の設定
;-----
MOV    P6,    #00000000B                ;P60-P61の出力ラッチLow
MOV    PM6,    #11111100B                ;P60-P61を出力ポートに設定
                                           ;P60-P61:未使用

;-----
;   ポート7の設定
;-----
MOV    P7,    #00000000B                ;P70-P77の出力ラッチLow
MOV    PM7,    #00000000B                ;P70-P77を出力ポートに設定

```

```

;P70-P77:未使用

;-----
;   ポート8の設定
;-----
MOV     P8,      #00000000B          ;P80-P83の出力ラッチLow
MOV     PM8,     #11110000B         ;P80-P83を出力ポートに設定
                                           ;P80-P83:未使用

;-----
;   ポート12の設定
;-----
MOV     P12,     #00000000B         ;P120の出力ラッチLow
MOV     PM12,    #11111110B         ;P120を出力ポートに設定
                                           ;P120-P124:未使用
                                           ;※P121-P124は入力ポート

;-----
;   ポート14の設定
;-----
MOV     P14,     #00000000B         ;P140-P141の出力ラッチLow
MOV     PM14,    #11111100B         ;P140-P141を出力ポートに設定
                                           ;P140-P141:未使用

;-----
;   ポート15の設定
;-----
MOV     P15,     #00000000B         ;P150-P153の出力ラッチLow
MOV     PM15,    #11110000B         ;P150-P153を出力ポートに設定
                                           ;P150-P153:未使用

RET

;*****
;
;
;   低電圧検出
;
;-----
;   低電圧検出回路の機能を使用し、2.7V以上の電源電圧を確保します。
;*****
SINILVI:
;低電圧検出回路の設定
SET1     LVIMK                      ;INTLVI割り込み禁止
CLR1     LVISEL                     ;検出電圧をVDDに設定

```



```

MOV     LVIS,    #00001001B                ;低電圧検出レベル選択レジスタ
;||||++++----- LVIS3-LVIS0
;||||
;|||| [検出レベル]
;|||| 0000:VLVI0 (4.22±0.1V)
;|||| 0001:VLVI1 (4.07±0.1V)
;|||| 0010:VLVI2 (3.92±0.1V)
;|||| 0011:VLVI3 (3.76±0.1V)
;|||| 0100:VLVI4 (3.61±0.1V)
;|||| 0101:VLVI5 (3.45±0.1V)
;|||| 0110:VLVI6 (3.30±0.1V)
;|||| 0111:VLVI7 (3.15±0.1V)
;|||| 1000:VLVI8 (2.99±0.1V)
;|||| 1001:VLVI9 (2.84±0.1V)
;|||| 1010:VLVI10(2.68±0.1V)
;|||| 1011:VLVI11(2.53±0.1V)
;|||| 1100:VLVI12(2.38±0.1V)
;|||| 1101:VLVI13(2.22±0.1V)
;|||| 1110:VLVI14(2.07±0.1V)
;|||| 1111:VLVI15(1.91±0.1V)
;++++----- 必ず0に設定

CLR1    LVIMD                ;低電圧検出時の動作モードを割り込み信
号発生に設定

SET1    LVION                ;低電圧検出動作許可

;低電圧検出回路の動作安定待ち(約10us)
MOV     B,    #10            ;カウント回数設定
HRES100:
NOP                                           ;(1clk)
DEC     B                                           ;(1clk)
BNZ     $HRES100                ;ウエイト完了? No,(2clk/4clk)

;VLVI≦VDDになるまでのウエイト
HRES300:
NOP
BT      LVIF,    $HRES300                ;VDD<VLVI? Yes,
CLR1    LVION                ;低電圧検出動作停止

RET

;*****
;
; クロック周波数の設定
;

```

```

;-----
;      高速内蔵発振クロックで動作が行えるように設定します。
;*****
SINICK:
      MOV      CMC,      #0000000B      ;クロック動作モード
      ;|||||+----- AMPH
      ;|||||      [高速システム・クロック発振周波数の制御]
      ;|||||      0: 2MHz≤fMX<10MHz
      ;|||||      1: 10MHz<fMX≤20MHz
      ;|||||++----- AMPHS1-AMPHS0
      ;|||||      [XT1発振回路の発振モード選択]
      ;|||||      00:低消費発振（デフォルト）
      ;|||||      01:通常発振
      ;|||||      10:超低消費発振
      ;|||||      11:超低消費発振
      ;|||+----- 必ず0に設定
      ;|||+----- OSCSELS
      ;|||      [サブシステム・クロック端子の動作モード]
      ;|||      0:入力ポート・モード
      ;|||      1:XT1発振モード
      ;||+----- 必ず0に設定
      ;++----- EXCLK/OSCSEL
      ;      [高速システム・クロック端子の動作モード]
      ;      00:入力ポート・モード
      ;      01:X1発振モード
      ;      10:入力ポート・モード
      ;      11:外部クロック入力モード

      MOV      CSC,      #1100000B      ;クロック動作ステータス制御
      ;|||||+----- HI0STOP
      ;|||||      [高速内蔵発振クロックの動作制御]
      ;|||||      0:高速内蔵発振回路動作
      ;|||||      1:高速内蔵発振回路停止
      ;||++++----- 必ず0に設定
      ;|+----- XTSTOP
      ;|      [サブシステム・クロックの動作制御]
      ;|      0:XT1発振回路動作
      ;|      1:XT1発振回路停止
      ;+----- MSTOP
      ;      [高速システム・クロックの動作制御]
      ;      0:X1発振回路動作
      ;      1:X1発振回路停止

```

```

MOV    OSMC,    #10000000B    ;動作スピード・モード
;|||||++----- FSEL/FLPC
;|||||          [fCLKの周波数選択]
;|||||          00:10MHz以下の周波数で動作（デフォルト）
;|||||          01:10MHzを越える周波数で動作
;|||||          10:1MHzの周波数で動作
;|||||          11:設定禁止
;|+++++----- 必ず0に設定
;+----- RTCLPC
;
;          [サブシステム・クロックHALTモード時の設定]
;          0:周辺機能へのサブシステム・クロック供
給許可
;          1:リアルタイム・カウンタ以外の周辺機能
への
;          サブシステム・クロ
ック供給停止

MOV    CKC,    #00001000B    ;クロック選択
;|+|+|+++----- CSS/MCM0/MDIV2-MDIV0
;| | |          [CPU/周辺ハードウェア・クロック(fCLK)の選択]
;| | |          00000: fIH
;| | |          00001: fIH/2（デフォルト）
;| | |          00010: fIH/2^2
;| | |          00011: fIH/2^3
;| | |          00100: fIH/2^4
;| | |          00101: fIH/2^5
;| | |          01000: fMX
;| | |          01001: fMX/2
;| | |          01010: fMX/2^2
;| | |          01011: fMX/2^3
;| | |          01100: fMX/2^4
;| | |          01101: fMX/2^5
;| | |          1xxxx: fSUB/2
;| | +----- 必ず1に設定
;| +----- MCS <Read Only>
;|          [メイン・システム・クロック(fMAIN)のステータス]
;|          0:高速内蔵発振クロック(fIH)
;|          1:高速システム・クロック(fMX)
;+----- CLS <Read Only>
;          [CPU/周辺ハードウェア・クロック(fCLK)のステータ
ス]
;          0:メイン・システム・クロック(fMAIN)
;          1:サブシステム・クロック(fSUB)

```

```

MOV      DSCCTL, #00000000B                ;20MHz高速内蔵発振制御
;|||||+----- DSCON
;||||| [20MHz高速内蔵発振クロック(fIH20)の動作許可/禁止]
;||||| 0:動作禁止
;||||| 1:動作許可
;|||||+----- 必ず0に設定
;||||+----- SELDSC
;|||| [CPU/周辺ハードウェア・クロック(fCLK)への20MHz高速内蔵発
振選択]
;|||| 0:20MHz高速内蔵発振を選択しない
;|||| 1:20MHz高速内蔵発振を選択
;||||+----- DSCS <Read Only>
;|||| [20MHz高速内蔵発振供給状態フラグ]
;|||| 0:供給していない
;|||| 1:供給している
;++++----- 必ず0に設定

RET

;*****
;
;      メイン・ループ
;
;*****
MAIN_LOOP:
MOV      SSOL, #00001100B                ;シリアル・チャネル開始レジスタ0
;||||++++----- SS0n (n=0-3)
;|||| [チャネルnの動作開始トリガ]
;|||| 0:トリガ動作せず
;|||| 1:SE0nに1をセットし、通信待機状態に遷
移する
;++++----- 必ず0に設定

LMAIN500:
HALT
BR       LMAIN500

;*****
;
;      シリアル・データ送信処理
;
;*****
SENDUA:

```

```

MOV      B,      #4                                ;送信回数カウンタ初期化

JSUA200:
MOV      A,      [HL]                              ;送信データ設定
MOV      TXD1,   A

JSUA300:
BF       STIF1,   $JSUA300                          ;送信完了? No,
CLR1     STIF1                                ; Yes,送信完了割り込みフラグクリア

INCW     HL                                ;送信データ格納アドレスの更新
DEC      B                                ;送信回数カウント
BNZ      $JSUA200                          ;全データ送信完了? No,

RET                                             ; Yes,return

;*****
;
;      SR1割り込み処理(INTSR1)
;
;*****
I INTSR1:
SEL      RB1                                ;レジスタ・バンク設定

MOV      A,      RXD1                          ;受信データ取得
MOVW     HL,      #TMSGUC                    ;応答メッセージ('U','C',"CR","LF")設定

CMP      A,      #54H                        ;受信データが't'(54H)?
SKNZ                                           ; No,
MOVW     HL,      #TMSGOK                    ; Yes,応答メッセージ('O','K',"CR","LF")設定

CMP      A,      #74H                        ;受信データが't'(74H)?
SKNZ                                           ; No,
MOVW     HL,      #TMSGOK2                   ; Yes,応答メッセージ('o','k',"CR","LF")設定

CALL     !!SENDUA                            ;応答メッセージ送信処理

RETI

;*****
;
;      SRE1割り込み処理(INTSRE1)
;
;*****

```

I INTSRE1:

```

SEL      RB2                                ;レジスタ・バンク設定

;エラー発生状況の取得
MOV      A,      SSR03L
AND      A,      #00000111B
MOV      RUASTT, A                        ;エラー発生状況をRAMに格納

BF      RUASTT.2,$HSRE1200                ;フレーミング・エラー発生? No,

MOV      SIR03L, #00000100B                ; Yes,フレーミング・エラー検出フラグのクリア
MOVW     HL,      #TMSGFE                  ;応答メッセージ('F','E',"CR","LF")設定

CALL     !!SENDUA                        ;シリアル・データ送信処理

```

HSRE1200:

```

BF      RUASTT.1,$HSRE1400                ;パリティ・エラー発生? No,

MOV      SIR03L, #00000010B                ; Yes,パリティ・エラー検出フラグのクリア
MOVW     HL,      #TMSGPE                  ;応答メッセージ('P','E',"CR","LF")設定

CALL     !!SENDUA                        ;シリアル・データ送信処理

```

HSRE1400:

```

BF      RUASTT.0,$HSRE1600                ;オーバーラン・エラー発生? No,

MOV      SIR03L, #00000001B                ; Yes,オーバーラン・エラー検出フラグのクリア
MOVW     HL,      #TMSGOE                  ;応答メッセージ('O','E',"CR","LF")設定

CALL     !!SENDUA                        ;シリアル・データ送信処理

```

HSRE1600:

```

MOV      A,      RXD1                      ;ダミーリード

RETI

```

end

● Kx3-L_UART.c (C言語版)

/*****

NEC Electronics 78K0R/KE3-Lシリーズ

78K0R/KE3-Lシリーズ サンプル・プログラム

UART通信

【履歴】

2008.11.16 新規作成

2009.1.31 全面変更

【概要】

;本サンプル・プログラムでは、シリアル・アレイ・ユニットのUART通信機能の使用例を

;示すとともに、対向機器から送られてくるASCII文字を解析し、応答処理を行います。

;受信処理はバッファを用いず受信データを直接処理し、通信は半二重で行います。

;

;

;<初期設定の主な内容>

; (オプション・バイトでの設定)

; ・ウォッチドッグ・タイマの動作禁止

; ・高速内蔵発振回路に8MHz/20MHzを選択

; ・LVIデフォルト・スタート機能動作

; ・オンチップ・デバッグを動作許可に設定

; (リセット解除後の初期化処理での設定)

; ・入出力ポートの設定

; ・低電圧検出回路の機能を使用し、2.7V以上の電源電圧を確保

; ・CPU／周辺ハードウェア・クロックを高速内蔵発振クロック動作の8MHzに設定

; ・X1/XT1発振回路の停止

;

;<シリアル・アレイ・ユニットの設定>

; ・チャンネル2を送信、チャンネル3を受信に設定

; ・ボー・レート：9600 bps

; ・データのキャラクタ長：8ビット

; ・パリティ・ビット指定：パリティ無し

; ・ストップ・ビット数：1ビット

; ・先頭ビットの指定：LSBファースト

;

;<メイン処理の主な内容>

; ・シリアル・チャンネル2,3を通信待機状態に遷移

```

;
;<シリアル・データ送信処理の主な内容>
;  ・シリアル・データの送信
;
;<割り込み処理(INTSR1)の主な内容>
;  ・受信データの判定
;  ・シリアル・データ送信処理の実行
;
;<割り込み処理(INTSRE1)の主な内容>
;  ・エラー内容の判定
;  ・シリアル・データ送信処理の実行
;
;
;<受信データと応答する送信データ>
;
; +-----+-----+
; | 受信データ |      応答する送信データ      |
; +-----+-----+
; |  T(54H)   |  O(4FH), K(4BH), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
; |  t(74H)   |  o(6FH), k(6BH), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
; |  上記以外 |  U(55H), C(43H), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
;
;
;<エラー発生時の送信データ>
;
; +-----+-----+
; |   発生したエラー   |      応答する送信データ      |
; +-----+-----+
; | パリティ・エラー   |  P(50H), E(45H), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
; | フレーミング・エラー |  F(46H), E(45H), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
; | オーバーラン・エラー |  O(4FH), E(45H), "CR"(0DH), "LF"(0AH) |
; +-----+-----+
;
;
;<入出力ポートの設定>
;  入力ポート：P31(RXD1)
;  出力ポート：P30(TXD1)
;  ※未使用のポートで出力に設定できるものは全て出力ポートに設定しておく

```



```

*****/

/*=====
    前処理指令（#pragma指令）
=====*/

#pragma SFR                /* 特殊機能レジスタ(SFR)名を記述可能にする */
#pragma NOP                /* NOP()を記述可能にする */
#pragma HALT               /* HALT()を記述可能にする */
#pragma DI                 /* DI()を記述可能にする */
#pragma EI                 /* EI()を記述可能にする */
/*=====

    割り込みハンドラ定義
=====*/

#pragma interrupt INTSR1    INTSR1_hdr rb2          /* 受信完了割り込み */
#pragma interrupt INTSRE1   INTSRE1_hdr rb1          /* 受信エラー割り込み */


#include <string.h>

/*=====

    関数プロトタイプ宣言
=====*/

void fn_uart_send(unsigned char *senddata); /* シリアル・データ送信用関数 */
static void f_ini_cpu(void); /* 動作クロック設定 */
static void f_ini_lvi(void); /* 電源電圧立ち上がり待ち */
static void f_ini_port(void); /* ポート設定 */
static void f_ini_reg(void); /* 使用する内蔵周辺機能の設定 */
static void f_ini_itr(void); /* 使用する割り込み関係の設定 */

/*=====

    グローバル変数の定義
=====*/

char senddata[5];
unsigned char rcvdata;
int i;
static const unsigned char g_ucmsgOK[4] = {"OK\r\n"};
static const unsigned char g_ucmsgok[4] = {"ok\r\n"};
static const unsigned char g_ucmsgUC[4] = {"UC\r\n"};
static const unsigned char g_ucmsgFE[4] = {"FE\r\n"};
static const unsigned char g_ucmsgPE[4] = {"PE\r\n"};
static const unsigned char g_ucmsgOE[4] = {"OE\r\n"};

```

```

/*****
リセット解除後の初期化処理
*****/

void hdwinit( void ){
    DI();
    f_ini_port();          /* ポート初期設定 */
    f_ini_lvi();           /* 2.7V以上の電源電圧を確保 */
    f_ini_cpu();           /* CPU初期設定 */
    f_ini_reg();           /* 周辺レジスタ初期設定 */
    f_ini_itr();           /* 割り込み初期設定 */
}

/*****
*   Title       :   電源電圧検出
*****/

*   Module      :   static void f_ini_lvi(void)
*   Arg         :
*   Ret         :
*   -----
*   Note        :   LVI機能を使い電源電圧が2.7V以上になるのを待ちます。
*****/

static void f_ini_lvi(void)
{
    unsigned char ucCnt10us;          /* LVI起動待ち時間計測用 */
/*-----
;   低電圧検出回路の機能を使用し、2.7V以上の電源電圧を確保します。
;-----*/

    LVIMK = 1;                        /* INTLVI割り込み禁止 */
    LVIM = 0b00000000;               /* 低電圧検出レジスの設定 */
/*
|||||+---LVIFフラグ
|||||
|||||+---LVIMD: 低電圧検出の動作モード選択
|||||                                0 : 割り込みモード
|||||                                1 : リセット・モード
|||||
|||||+----LVISEL: 電圧検出の選択
|||||                                0 : 電源電圧 (VDD) のレベルを検出
|||||                                1 : 外部入力端子からの入力電圧のレベルを検出
|||||
|++++----- 必ず0に設定
|

```

```

+-----LVION: 低電圧検出動作許可
                                0 : 動作禁止
                                1 : 動作許可

*/

    LVIS =  0b00001001;          /* 検出電圧の設定 */
/*
    |||++++--LVIS3-0: [検出レベル]
    |||                                0000 : 4.22±0.1V
    |||                                0001 : 4.07±0.1V
    |||                                0010 : 3.92±0.1V
    |||                                0011 : 3.76±0.1V
    |||                                0100 : 3.61±0.1V
    |||                                0101 : 3.45±0.1V
    |||                                0110 : 3.30±0.1V
    |||                                0111 : 3.15±0.1V
    |||                                1000 : 2.99±0.1V
    |||                                1001 : 2.84±0.1V
    |||                                1010 : 2.68±0.1V
    |||                                1011 : 2.53±0.1V
    |||                                1100 : 2.38±0.1V
    |||                                1101 : 2.22±0.1V
    |||                                1110 : 2.07±0.1V
    |||                                1111 : 1.91±0.1V
    |||
    ++++----- 必ず0に設定

*/

    LVION =  1;                  /* 低電圧検出動作許可 */

/*
    低電圧検出回路の動作安定待ち(約10us)
*/
    for( ucCnt10us = 0; ucCnt10us < 3; ucCnt10us++ ){
        NOP();
        NOP();
    }

/*
    VLVI ≤ VDDになるまでのウェイト
*/
    while(LVIF){
        NOP();
    }

```

```

        LVION = 0;                                /* 低電圧検出動作停止 */
    }

/*****
*   Title       :       CPU初期設定
*****/

*   Module      :       static void f_ini_cpu(void)
*   Arg         :
*   Ret         :
*-----
*   Note       :
*****/

static void f_ini_cpu(void)
{
/*-----
;       動作クロックを内蔵高速発振器(8MHz)に設定します。
;-----*/

        CMC = 0b00000000;                        /* クロック動作モード制御レジスタ
(CMC) */

/*
|||||+---AMPH: 高速システム・クロック発振周波数の制御
|||||
0 : 2MHz<=fMX<=10MHz
|||||
1 : 10MHz<=fMX<=20MHz
|||||
|||||+---AMPHS1,AMPS0: XT1発振回路の発振モード選択
|||||
00 : 低消費発振(デフォルト)
|||||
01 : 通常発振
|||||
1x : 超低消費発振
|||||
||+|+----- 必ず0に設定
|| |
|| +-----OSCELS: サブシステム・クロック端子の動作モード
||
0 : 入力ポート・モード
||
1 : XT1発振モード
++-----EXCLK,OSCESEL: 高速システム・クロック端子の動作モード
00 : 入力ポート・モード
01 : X1発振モード
10 : 入力ポート・モード
11 : 外部クロック入力モード

*/

        CSC = 0b11000000;                        /* クロック動作ステータス制御レジスタ
(CSC) */

```

```

/*          |||||+---HI0STOP: 高速内蔵発振クロックの動作制御
          |||||
          |||||          0 : 高速内蔵発振回路動作
          |||||          1 : 高速内蔵発振回路停止
          |||||
          ||+++++--- 必ず0に設定
          ||
          |+-----XTSTOP: サブシステム・クロックの動作制御
          |
          |          0 : XT1発振回路動作
          |          1 : XT1発振回路停止
          |
          +-----MSTOP: 高速システム・クロックの動作制御
          |
          |          0 : X1発振回路動作
          |          1 : X1発振回路停止

*/

CKC =      0b00001000;          /* システム・クロック制御レジスタ(CKC)

*/

/*          |+|+|+++---CSS,MCM0,MDIV2,MDIV1,MDIV0:
          |||          CPU/周辺ハードウェア・クロックの選択
          |||          00000 : fIH
          |||          00001 : fIH/2
          |||          00010 : fIH/4
          |||          00011 : fIH/8
          |||          00100 : fIH/16
          |||          00101 : fIH/32
          |||          01000 : fMX
          |||          01001 : fMX/2
          |||          01010 : fMX/4
          |||          01011 : fMX/8
          |||          01100 : fMX/16
          |||          01101 : fMX/32
          |||          1xxxx : fSUB/2
          |||
          || +----- 必ず1に設定
          ||
          | +-----MCS: メイン・システム・クロック(fMAIN)のステータス
          |
          |          0 : 高速内蔵発振クロック(fIH)
          |          1 : 高速システム・クロック(fMX)
          |
          +-----CLS: CPU/周辺ハードウェア・クロック(fCLK)のステータス
          |
          |          0 : メイン・システム・クロック(fMAIN)
          |          1 : サブシステム・クロック(fSUB)

*/

```

```

DSCCTL = 0b00000000; /* 6) 20 MHz高速内蔵発振制御レジスタ (DSCCTL) */
/*
|||||+---DSCON: 20 MHz高速内蔵発振クロックの動作許可／禁止
|||||
||||| 0 : 動作禁止
||||| 1 : 動作許可
|||||
++++|+--- 必ず0に設定
||
|+----SELDSC: CPU／周辺ハードウェア・クロック (fCLK) への20 MHz高
速内蔵発振選択
|
| 0 : 20 MHz高速内蔵発振を選択しない
| 1 : 20 MHz高速内蔵発振を選択
|
+-----20 MHz高速内蔵発振供給状態フラグ
*/

/*-----
; 周辺機能へのクロック供給 (イネーブル・レジスタ)
;-----*/

/*
この段階では周辺機器へのクロック供給は禁止しておく

*/
PERO = 0b00000000; /* 周辺イネーブル・レジスタ0(PERO) */
/*
|+|+|+--- 必ず0に設定
| | |
| | +----SAU0EN: シリアル・アレイ・ユニットの入力クロックの制御
| | 0 : 入力クロック供給停止
| | 1 : 入力クロック供給
| |
| | +-----IICAEN: シリアル・インタフェースIICAの入力クロックの制御
| | 0 : 入力クロック供給停止
| | 1 : 入力クロック供給
| |
| +-----ADCEN: A/Dコンバータの入力クロックの制御
| 0 : 入力クロック供給停止
| 1 : 入力クロック供給
|
+-----RTCEN: リアルタイム・カウンタ(RTC)の入力クロックの制御
0 : 入力クロック供給停止
1 : 入力クロック供給

```

```

*/

PER1 = 0b00000000; /* 周辺イネーブル・レジスタ1(PER1) */
/*      ++++|++-- 必ず0に設定
          +-----OACMPEN: オペアンプの入カクロックの制御
                        0 : 入カクロック供給停止
                        1 : 入カクロック供給
*/

PER2 = 0b00000000; /* 周辺イネーブル・レジスタ2(PER2) */
/*      ++++++|-- 必ず0に設定
          |
          +---TAU0EN: タイマ・アレイ・ユニットTAUSの入カクロックの制御
                        0 : 入カクロック供給停止
                        1 : 入カクロック供給
*/

/*
/*-----
;      動作スピード・モード制御レジスタ(OSMC)
;-----*/

/*

      設定はデフォルトのままでいいので設定は省略

*/

OSMC = 0b00000000; /* 動作スピード・モード制御レジスタ(OSMC) */
/*      |+++++|-- 必ず0に設定
          |      |
          |      +---FLPC,FSEL: fCLKの周波数選択
          |
          |      00 : 10MHz以下の周波数で動作(デフォルト)
          |
          |      01 : 10MHzを越える周波数で動作
          |      10 : 1MHz以下の周波数で動作
          |      11 : 設定禁止
          |
          +-----RTCLPC: サブシステム・クロックHLATモード時の設定
                        0 : 周辺機能へのサブシステム・クロック
                        1 : RTC以外の周辺機能へのサブシステム・クロック供給許可
*/

```

}

```

/*****
*      Title      :      ポート初期設定
*****/

*      Module      :      static void f_ini_port(void)
*      Arg          :
*      Ret          :
*-----
*      Note        :
*****/

static void f_ini_port(void)
{
/*****
;   ポート 0 の設定（未使用）
;
;*****/

        P0 =      0b00000000;                /* P01,P00を0に設定
                                           */

        PM0 =     0b11111100;                /* P01,P00を出力に設定
                                           */

/*****
;   ポート 1 の設定（未使用）
;
;*****/

        P1 =      0b00000000;                /* 全ての出力ラッチを0に設定
                                           */

        PM1 =     0b00000000;                /* P17～P10を出力に設定する
                                           */

/*****
;   ポート 2 の設定（未使用）
;
;*****/

        ADPC =    0b00010000;                /* A/Dポート・コンフィギュレーション・
                                           レジスタ（ADPC）：全てデジタルに
                                           */

```



```

P2 =    0b00000000;          /* 全ての出力ラッチを0に設定
*/
PM2 =    0b00000000;          /* 全てのビットを出力に設定
*/

/*****
; ポート3の設定
;
; P30をTxD出力、P31をRxD入力として使用する。そのほかのビットは
;出力に設定する
;
; *****/

P3 =    0b00000001;          /* P30以外の出力ラッチは0に設定
*/
PM3 =    0b11110010;          /* P31(RxD)を入力、他を出力に設定
-      +-PM31-30: 端子の入出力モードの選択
                                0: 出力モード
                                1: 入力モード

*/

POM3 =    0b00000011;          /* 端子の出力モードの選択
-      +-POM31-30: 端子の出力モードの選択
                                0: 通常出力モード
                                1: N-chオープン・ドレイン出力(VDD耐圧)
モード
*/

/*****
; ポート4の設定（未使用）
;
; *****/

P4 =    0b00000000;          /* P43～P40の出力ラッチを0に設定
*/
PM4 =    0b11110000;          /* 全てのビットを出力に設定
*/

/*****
; ポート5の設定（未使用）
;
; *****/

```

```

P5 =    0b00000000;          /* P53～P50の出力ラッチを0に設定
*/
PM5 =    0b11110000;          /* 全てのビットを出力に設定
*/

/*****
; ポート6の設定（未使用）
;
; *****/

P6 =    0b00000000;          /* P61～P60の出力ラッチを0に設定
*/
PM6 =    0b11111100;          /* 全てのビットを出力に設定
*/

/*****
; ポート7の設定（未使用）
;
; *****/
/* PIM7とPOM7は設定しないが参考として示す
PIM7 =    0b00000000;*/          /* P75～4、P72～1は通常入力          */
/* POM7 =    0b00000000;*/          /* P75、P73～2、P70は通常出力モード */
P7 =    0b00000000;          /* P77～P70の出力ラッチを0に設定
*/
PM7 =    0b00000000;          /* 全てのビットを出力に設定
*/

/*****
; ポート8の設定（未使用）
;
; *****/
/* PIM8は設定しないが参考として示す
PIM8 =    0b00001111;*/          /* P83～P80はデジタル入力許可          */
P8 =    0b00000000;          /* P83～P80の出力ラッチを0に設定
*/
PM8 =    0b11110000;          /* 全てのビットを出力に設定
*/

/*****
; ポート12の設定
;
; *****/

```

```

        P12 =    0b00000000;                /* P124～P120の出力ラッチを0に設定
        */
        PM12 =   0b11111110;                /* P120を出力に設定する
        */

/*****
;   ポート14の設定（未使用）
;
; *****/

        P14 =    0b00000000;                /* P61～P60の出力ラッチを0に設定
        */
        PM14 =   0b11111100;                /* 全てのビットを出力に設定
        */

/*****
;   ポート15の設定（未使用）
;
; *****/

        P15 =    0b00000000;                /* P153～P150の出力ラッチを0に設定
        */
        PM15 =   0b11110000;                /* 全てのビットを出力に設定
        */

}

/*****
*   Title       :       周辺レジスタ初期設定
*****
*   Module      :       static void f_ini_reg(void)
*   Arg         :
*   Ret         :
*   -----
*   Note        :
*****/
static void f_ini_reg(void)
{

/*-----
;   シリアル・アレイ・ユニットの設定
;-----*/

```

```

/*-----
-      周辺イネーブル・レジスタ0(PER0)
      SAU0EN: シリアル・アレイ・ユニットの入力クロックの制御
-----*/
SAU0EN = 1;

/* ウェイト */
NOP();
NOP();
NOP();
NOP();

SPS0L = 0b000000010;          /*シリアル・クロック選択レジスタ0
; |||+----- PRS003-PRS000 ※fCLK/2^2を使用
; +++----- PRS013-PRS010
      [動作クロック(CK00/CK01)の選択]
                                0000 : fCLK
                                0001 : fCLK/2
                                ☆ 0010 : fCLK/2^2
                                0011 : fCLK/2^3
                                0100 : fCLK/2^4
                                0101 : fCLK/2^5
                                0110 : fCLK/2^6
                                0111 : fCLK/2^7
                                1000 : fCLK/2^8
                                1001 : fCLK/2^9
                                1010 : fCLK/2^10
                                1011 : fCLK/2^11
                                1111 : INTTM02
                                上記以外 : 設定禁止

*/

/* 送信側 */

SMR02 = 0b00000000000100010; /*
- |||||+ MD020[チャネル2の割り込み要因の選択]
- ||||| 0 : 転送完了割り込み
- ||||| 1 : バッファ空き割り込み
- |||||
- |||||++- MD022,MD021[チャネル2の動作モードの設定]
- ||||| 00 : CSIモード

```

```

- ||||| 01 : UARTモード
- ||||| 10 : 簡易I2Cモード
- ||||| 11 : 設定禁止
- |||||
- |||||+++--- 必ず100に設定
- |||||
- |||||+----- SIS020[UARTモードでのチャンネルnの受信データのレベル反転の制御]
- ||||| 0 : 立ち下がりエッジをスタートビットとして検出
します。
- ||||| 入力される通信データは,そのまま取り込
まれます。
- ||||| 1 : 立ち上がりエッジをスタートビットとして検出
します。
- ||||| 入力される通信データは,反転して取り込
まれます。
- |||||
- |||||+----- 必ず0に設定
- |||||
- |||||+----- STS02[スタート・トリガ要因の選択]
- ||||| 0 : ソフトウェア・トリガのみ有効
- ||||| 1 : RxD端子の有効エッジ
- |||||
- |||++++----- 必ず0に設定
- ||
- |+----- CCS02[チャンネル2の転送クロック(TCLK)の選択]
- | 0 : CKS02ビットで指定した動作クロック
MCKの分周クロック
- | 1 : SCK端子からの入力クロック
- |
- +----- CKS02[チャンネル2の動作クロック(MCK)の選択]
0 : PRSレジスタで設定したプリスケアラ
出力クロックCK00
1 : PRSレジスタで設定したプリスケアラ
出力クロックCK01
*/
/* 受信側 */
SMR03 = 0b0000000100100010; /* シリアル・モード・レジスタ03(SMR03)
- |||||+ MD030[チャンネル3の割り込み要因の選択]
- ||||| 0 : 転送完了割り込み
- ||||| 1 : バッファ空き割り込み
- |||||
- |||||++- MD032,MD031[チャンネル3の動作モードの設定]
- ||||| 00 : CSIモード

```

まま取り込まれます。

して取り込まれます。

MCKの分周クロック

出力クロックCK00

出力クロックCK01

*** /**

```
/* 送信側 */
```

```
SCR02 = 0b1000001010010111;
```

/* シリアル通信動作設定レジスタ02(SCR02)

```
- |||||+++ DLS022-DLS020[CSI,UARTモードでのデータ長の設定]
```

- ||||| 100 : 5ビット・データ長

```

- ||||| 110 : 7ビット・データ長
- ||||| ☆ 111 : 8ビット・データ長
- ||||| 上記以外:設定禁止
- |||||
- |||||+--- 必ず0に設定
- |||||
- |||||+----- SLC021,SLC020[UARTモードでのストップ・ビットの設定]
- ||||| 00 : ストップ・ビットなし
- ||||| ☆ 01 : ストップ・ビット長 = 1ビット
- ||||| 10 : ストップ・ビット長 = 2ビット
- ||||| 11 : 設定禁止
- |||||
- |||||+----- 必ず0に設定
- |||||
- |||||+----- DIR02[CSI, UARTモードでのデータ転送順序の選択]
- ||||| 0 : MSBファーストで入出力を行う
- ||||| ☆ 1 : LSBファーストで入出力を行う
- |||||
- |||||+----- PTC021,PTC020[UARTモードでのパリティ・ビット
- ||||| の設定]
- ||||| 00 : パリティ・ビットを出力しない
- ||||| 01 : 0パリティを出力
- ||||| ☆ 10 : 偶数パリティを出力
- ||||| 11 : 奇数パリティを出力
- |||||
- |||||+----- EOC02[エラー割り込み信号のマスク可否の選択]
- ||||| ☆ 0 : エラー割り込みをマスクする
- ||||| 1 : エラー割り込みの発生を許可する
- |||||
- |||||+----- 必ず0に設定
- |||||
- |||||+----- DAP02,CKP02[CSIモードでのデータとクロックの
- ||||| 位相選択]
- ||||| 00 : UARTモード時には、必ず00に設定
- |||||
- ++----- TXE02,RXE02[チャネル2の動作モードの設定]
- ||||| 00 : 通信を開始しない
- ||||| 01 : 受信のみ
- ||||| ☆ 10 : 送信のみ
- ||||| 11 : 送受信を行う

```

*/

/* 受信側 */

```

SCR03 = 0b0100011010010111;          /* シリアル通信動作設定レジスタ03(SCR03)

- |||||++++ DLS032-DLS030[CSI, UARTモードでのデータ長の設定]
- |||||      100 : 5ビット・データ長
- |||||      110 : 7ビット・データ長
- |||||☆     111 : 8ビット・データ長
- |||||      上記以外:設定禁止
- |||||
- |||||++++ 必ず0に設定
- |||||
- |||||++++ SLC031,SLC030[UARTモードでのストップ・ビットの設定]
- |||||      00 : ストップ・ビットなし
- |||||☆     01 : ストップ・ビット長 = 1ビット
- |||||      10 : ストップ・ビット長 = 2ビット
- |||||      11 : 設定禁止
- |||||
- |||||+----- 必ず0に設定
- |||||
- |||||+----- DIR03[CSI, UARTモードでのデータ転送順序の選択]
- |||||      0 : MSBファーストで入出力を行う
- |||||☆     1 : LSBファーストで入出力を行う
- |||||
- |||||+----- PTC031,PTC030[UARTモードでのパリティ・ビットの設定]
- |||||      00 : パリティ・ビットを出力しない
- |||||      01 : 0パリティを出力
- |||||☆     10 : 偶数パリティを出力
- |||||      11 : 奇数パリティを出力
- |||||+----- EOC03[エラー割り込み信号のマスク可否の選択]
- |||||      0 : エラー割り込みをマスクする
- |||||☆     1 : エラー割り込みの発生を許可する
- |||||
- |||||+----- 必ず0に設定
- |||||
- ||+----- DAP03,CKP03[CSIモードでのデータとクロックの位相選択]
- ||      00 : UARTモード時には、必ず00に設定
- ||
- ++----- TXE03,RXE03[チャネル3の動作モードの設定]
                                00 : 通信を開始しない
                                ☆ 01 : 受信のみ
                                10 : 送信のみ
                                11 : 送受信を行う

/* 送信側 */
/*-----

```


シリアル・データ・レジスタ02

-----*/

SDR02 = 0b1100111000000000; /*シリアル・データ・レジスタ02

- |||||+----- 必ず0に設定

- |||||

- ++++++----- SDR02[15:9][動作クロック(MCK)の分周による転送クロック設

定]

0000000 : MCK/2(UART使用時設定禁止)

0000001 : MCK/4(UART使用時設定禁止)

0000010 : MCK/6

: :

: :

1111110 : MCK/254

1111111 : MCK/256

※1100111:MCK/208 = 2MHz/208 = 9615.38bps

目標ボー・レート9600bpsとの誤差 = +0.16%

*/

/* 受信側 */

/*-----

シリアル・データ・レジスタ03

-----*/

SDR03 = 0b1100111000000000; /*シリアル・データ・レジスタ02

- |||||+----- 必ず0に設定

- |||||

- ++++++----- SDR02[15:9][動作クロック(MCK)の分周による転送クロック設

定]

0000000 : MCK/2(UART使用時設定禁止)

0000001 : MCK/4(UART使用時設定禁止)

0000010 : MCK/6

: :

: :

1111110 : MCK/254

1111111 : MCK/256

※1100111:MCK/208 = 2MHz/208 = 9615.38bps

目標ボー・レート9600bpsとの誤差 = +0.16%

*/

/* 送信側 */

/*-----

- シリアル出力レベル・レジスタ0(SOL0L)

-----*/

```

SOL0L = 0b00000000;          /* シリアル出力レベル・レジスタ0
-      +-----SOL02[UARTモードでのチャンネル2の送信データのレベル反転の選択]
                                0 : 通信データは、そのまま出力されます
                                1 : 通信データは、反転して出力されます

*/

/*-----
-      シリアル出力レジスタ0(S00)
-----*/
S00 = 0b0000100000001100;      /* シリアル出力レジスタ0
-      +-----S002: チャンネル2のシリアル・データ出力
                                0 : シリアル・データ出力値が"0"
                                1 : シリアル・データ出力値が"1"

*/

/*-----
-      シリアル出力許可レジスタ0(S0E0)
-----*/
S0E0L = 0b00000100;          /* シリアル出力許可レジスタ0
-      +-----S0E02: チャンネル2のシリアル出力許可/停止
                                0 : シリアル通信動作による出力停止
                                1 : シリアル通信動作による出力許可

*/

}

/*****
*      Title      :      割込初期設定
*****/
*      Module      :      static void f_ini_itr(void)
*      Arg          :
*      Ret          :
* -----
*      Note       :
*****/
static void f_ini_itr(void)
{
    /* 受信割込み */
    SRIF1= 0;                  /* 受信割り込み要求フラグクリア */

    SRMK1= 0;                  /* 割り込みマスクの解除 */

```

```

SREIF1=0;                                /* 受信エラー割り込み要求フラグクリア */

SREMK1=0;                                /* 割り込みマスクの解除 */

}

/*****
*      Title      :      割込処理
*****/

*      Module     :
*      Arg        :
*      Ret        :
*-----
*      Note       :
*****/

static void INTSR1_hdr(void)              /* 受信時 */
{
    rcvdata = RXD1;                      /* 受信データチェック */
    if(rcvdata == 'T')                   /* 受信データが'T'のとき */
    {
        fn_uart_send(g_ucmsgOK);
    }

    else if(rcvdata == 't')              /* 受信データが't'のとき */
    {
        fn_uart_send(g_ucmsgok);
    }

    else                                 /* 受信データが上記2つ以外のとき */
    {
        fn_uart_send(g_ucmsgUC);
    }
}

static void INTSRE1_hdr(void)            /* エラー割込み */
{
    unsigned short status;

    status = SSR03L & 0b00000111;

    if(status & 0x04)                   /* フレーミング・エラー */

```

```

    {
        SIR03L = 0x04;                /* エラークリア */
        fn_uart_send(g_ucmsgFE);
    }

    if(status & 0x02)                /* パリティ・エラー */
    {
        SIR03L = 0x02;                /* エラークリア */
        fn_uart_send(g_ucmsgPE);
    }

    if(status & 0x01)                /* オーバーラン・エラー */
    {
        SIR03L = 0x01;                /* エラークリア */
        fn_uart_send(g_ucmsgOE);
    }
    status = RXD1;                    /* ダミーリード */
}

/*****
*   Title       :       シリアル・データ送信処理
*****/
*   Module      :       void fn_uart_send(unsigned char *senddata)
*   Arg         :
*   Ret         :
*-----
*   Note        :
*****/
void fn_uart_send(unsigned char *senddata)
{
    unsigned int count;

    for( count = 0 ; count < 4 ; count++ ){
        TXD1 = senddata[count];
        while( !STIF1 ){
            NOP();
        }
        STIF1 = 0;
    }
}

```

```
/******
```

メイン・ループ

```
*****/
```

```
void main(void)
```

```
{
```

```
    EI();
```

```
    SS0L = 0b00001100;          /* シリアル・チャネル開始レジスタ0(SS0)
```

```
        -||||++++----- SS0n (n=0-3)[チャネルnの動作開始トリガ]
```

```
        -||||                                0:トリガ動作せず
```

```
        -||||                                1:SE0nに1をセットし、通信待機状態に遷
```

移する

```
        -||||
```

```
        -++++----- 必ず0に設定
```

```
    */
```

```
    while(1)
```

```
    {
```

```
        HALT();
```

```
    }
```

```
}
```

付録B 改版履歴

版 数	発行年月	改版箇所	改版内容
第1版	-	—	—

【発 行】

NECエレクトロニクス株式会社

〒211-8668 神奈川県川崎市中原区下沼部1753

電話（代表）：044(435)5111

お問い合わせ先

【ホームページ】

NECエレクトロニクスの情報がインターネットでご覧になれます。

URL（アドレス） <http://www.necel.co.jp/>

【営業関係、デバイスの技術関係お問い合わせ先】

半導体ホットライン

（電話：午前 9:00～12:00，午後 1:00～5:00）

【マイコン開発ツールの技術関係お問い合わせ先】

開発ツールサポートセンター

電 話：044-435-9494

E-mail：info@necel.com

E-mail：toolsupport-micom@ml.necel.com

【資料請求先】

NECエレクトロニクスのホームページよりダウンロードいただくか、NECエレクトロニクスの販売特约店へお申し付けください。
