

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ひ申し上げます。

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、誤りが無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と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

78K0R/Lx3

サンプル・プログラム（初期設定）

LED点灯のスイッチ制御編

この資料は、78K0R/Lx3マイクロコントローラのオプション・バイトの設定、クロック周波数の選択、入出力ポートの設定など基本的な初期設定を行うサンプル・プログラムを説明したものです。

対象デバイス

78K0R/LF3マイクロコントローラ
 78K0R/LG3マイクロコントローラ
 78K0R/LH3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 ...	4
2.1 回路イメージ ...	4
2.2 マイコン以外の使用デバイス ...	4
第3章 ソフトウェアについて ...	5
3.1 ファイル構成 ...	5
3.2 使用する内蔵周辺機能 ...	6
3.3 初期設定と動作概要 ...	6
3.4 フロー・チャート ...	7
第4章 設定方法について ...	8
4.1 オプション・バイトの設定 ...	9
4.2 ベクタ・テーブルの設定 ...	14
4.3 スタック・ポインタの設定 ...	15
4.4 ウォッチドッグ・タイマの設定と制御 ...	16
4.5 クロックの設定 ...	17
4.6 ポートの設定 ...	23
4.7 メイン処理 ...	28
第5章 関連資料 ...	30
付録A プログラム・リスト ...	31
付録B 改版履歴 ...	53

- ・本資料に記載されている内容は2009年09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。
- 注2. 本事項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、製造製品をいう。
- (M8E0909J)

第1章 概 要

このサンプル・プログラムでは、78K0R/Lx3マイクロコントローラのオプション・バイトの設定、クロック周波数の選択、入出力ポートの設定など基本的な初期設定を行います。

また、初期設定完了後のメイン処理動作では、2つのスイッチ入力により、3つのLED点灯を制御します。

(1) 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
- ・高速内蔵発振回路の周波数を8MHzに設定
- ・LVIデフォルト・スタート機能停止
- ・オンチップ・デバッグを動作許可に設定

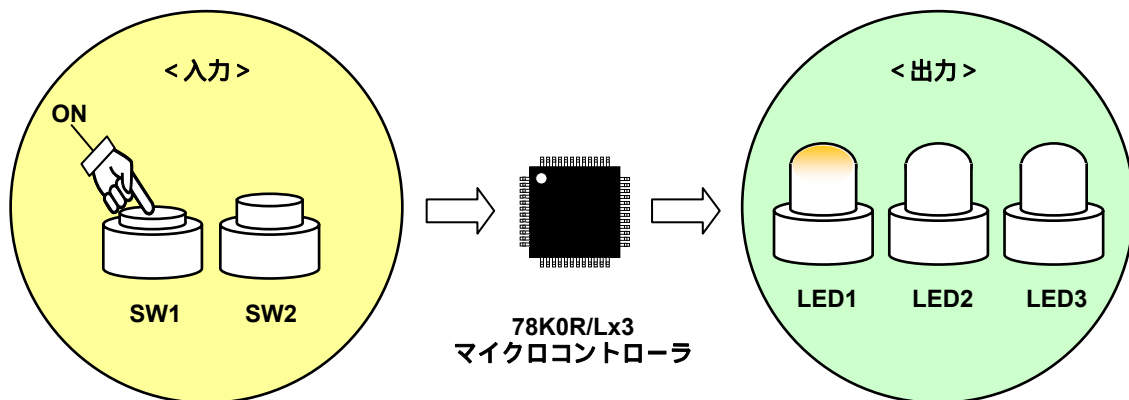
<リセット解除後の初期化処理での設定>

- ・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

注 低電圧検出回路についての詳細は、ユーザズ・マニュアルを参照してください。

(2) メイン処理動作の内容

78K0R/Lx3マイクロコントローラにて、スイッチ入力 (SW1, SW2) を検出し、LED点灯 (LED1, LED2, LED3) を制御します。



スイッチ入力		LED出力		
SW1	SW2	LED1	LED2	LED3
OFF	OFF	OFF	OFF	OFF
ON	OFF	ON	OFF	OFF
OFF	ON	OFF	ON	OFF
ON	ON	OFF	OFF	ON

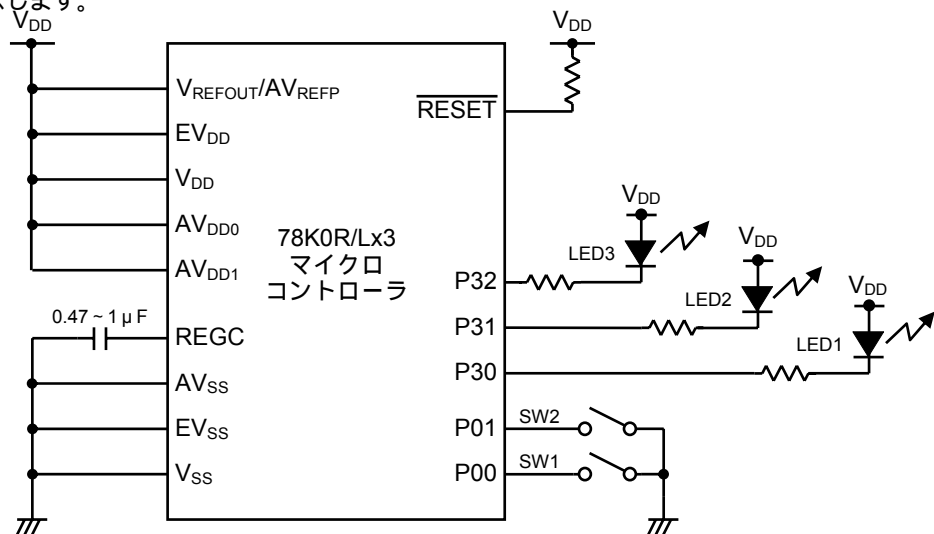
注意 デバイス使用上の注意事項については、ユーザズ・マニュアルを参照してください。

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場合の回路イメージおよびマイコン以外の使用デバイスを説明します。

2.1 回路イメージ

回路イメージを次に示します。



注意1. 2.94 V V_{DD} 5.5 Vの電圧範囲で使用してください(2.84 ± 0.1 V < V_{DD} の設定で低電圧検出を行っているため)。

- EV_{DD}, AV_{DD0}, AV_{DD1}, およびV_{REFOUT}/AV_{REFP}は、V_{DD}と同電位にしてください。
- AV_{SS}はEV_{SS}, V_{SS}と同電位にし、GNDに直接接続してください。
- REGCはコンデンサ(0.47 ~ 1 μF)を介し、V_{SS}に接続してください。
- 回路イメージ中に記載のない未使用端子は以下のように処理してください。
出力ポート : 出力モードに設定し、オープン(未接続)にしてください
入力ポート : 個別に抵抗を介して、V_{DD}またはV_{SS}に接続してください
- このサンプル・プログラムでは、P40/TOOL0端子、およびP41/TOOL1端子をオンチップ・デバッグ用に使用します。

2.2 マイコン以外の使用デバイス

マイコン以外に使用するデバイスを次に示します。

(1) スイッチ (SW1, SW2)

LED点灯制御用の入力として、スイッチを使用します。

(2) LED (LED1, LED2, LED3)



スイッチ入力に対応した出力として、LEDを使用します。

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・プログラムの初期設定と動作概要，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。

ファイル名	説明	同封圧縮 (*.zip) ファイル	
			
main.asm (アセンブリ言語版) ----- main.c (C言語版)	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	注	注
op.asm	オプション・バイト設定用アセンブラ・ソース・ファイル (ウォッチドッグ・タイマの設定，高速内蔵発振回路の周波数選択，LVIデフォルト・スタート機能の設定などを行います)		
78K0RLx3_sample_program.prw	統合開発環境 PM+用ワーク・スペース・ファイル		
78K0RLx3_sample_program.prj	統合開発環境 PM+用プロジェクト・ファイル		

注. アセンブリ言語版には「main.asm」，C言語版には「main.c」が同封されています。

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 使用する内蔵周辺機能

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。

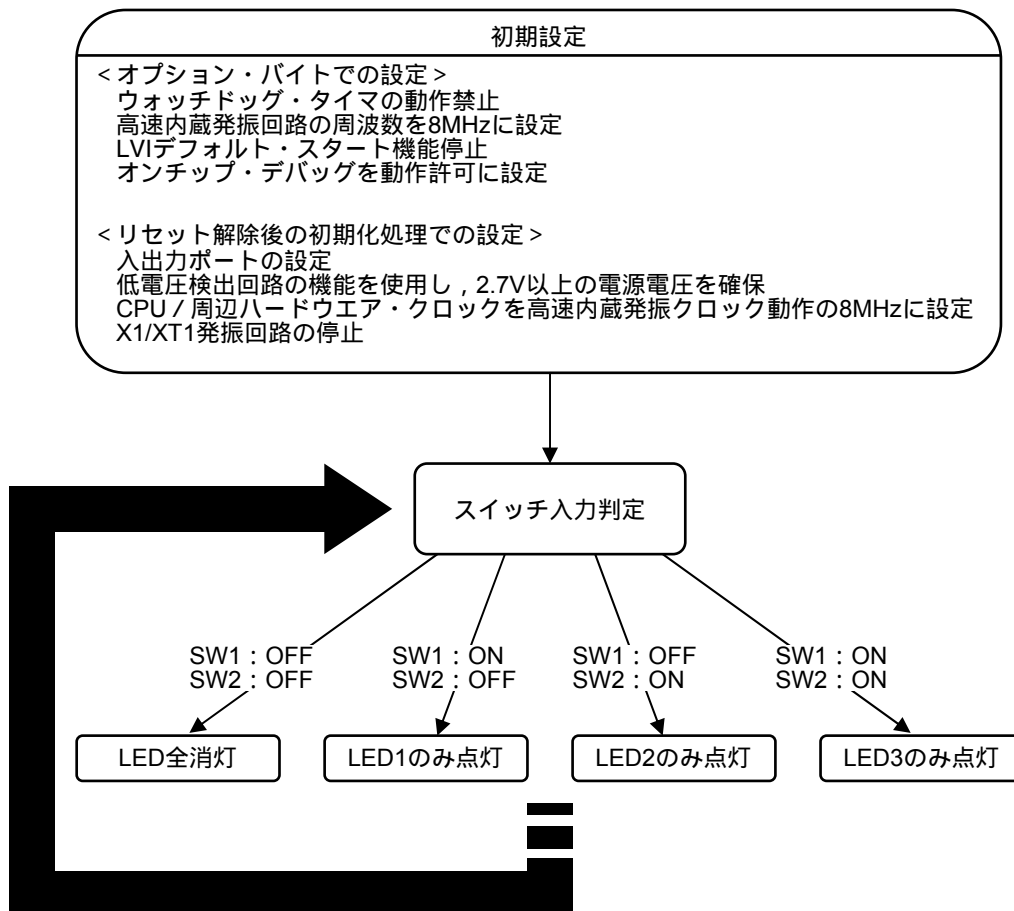
- ・ P00, P01 : スイッチ入力用に使います。
- ・ P30, P31, P32 : LED点灯用に使います。
- ・ 低電圧検出回路 : 2.7V V_{DD} の確認用に使います。

3.3 初期設定と動作概要

このサンプル・プログラムでは、初期設定にて、クロック周波数の選択や、入出力ポートの設定などを行います。

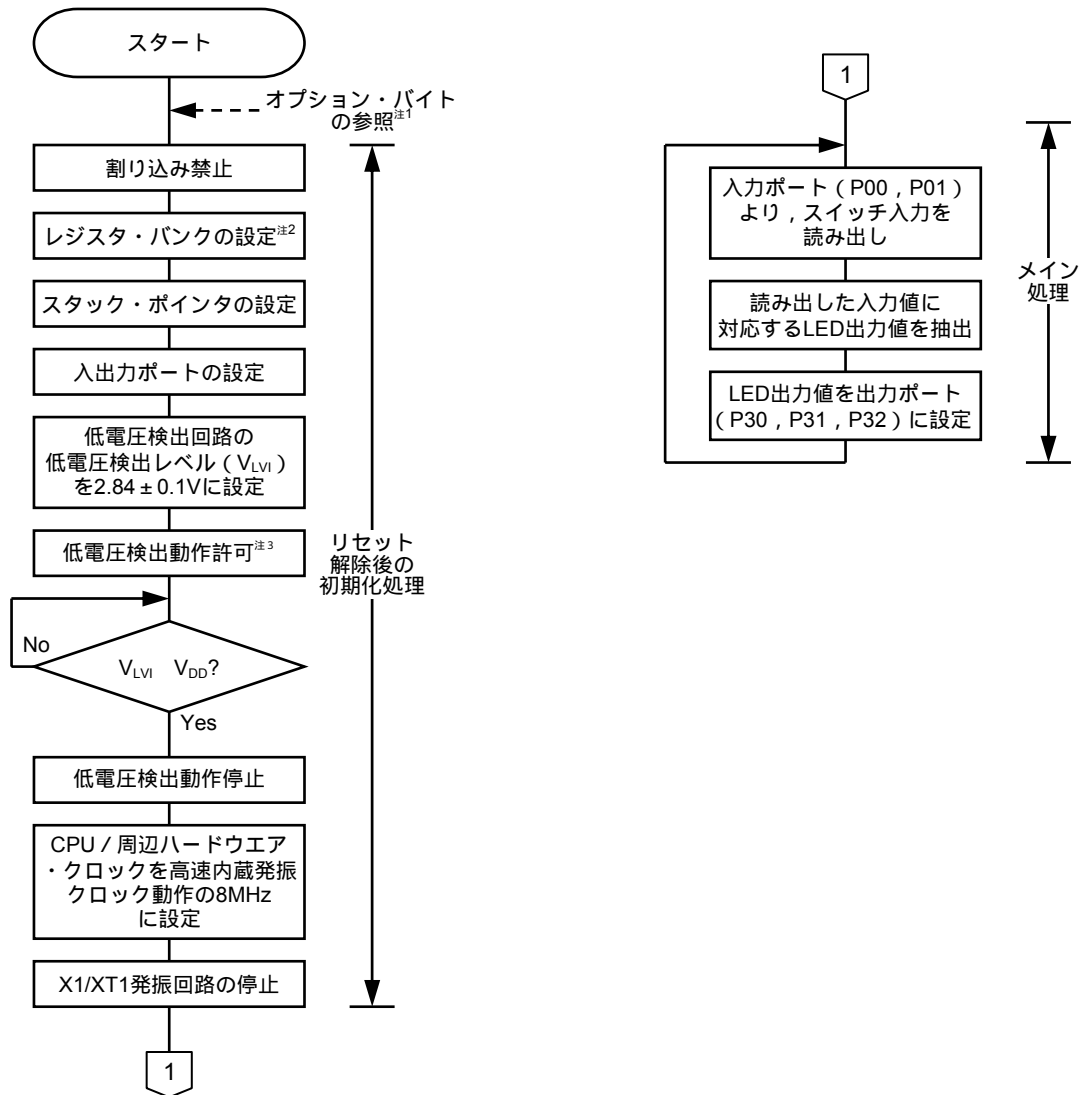
初期設定完了後は、2つのスイッチ入力 (SW1, SW2) の組み合わせに応じて、3つのLED (LED1, LED2, LED3) の点灯を制御します。

詳細については、次の状態遷移図 (ステート・チャート) 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。



注1. オプション・バイトの参照は、リセット解除後にマイコンが自動的に行います。このサンプル・プログラムでは、オプション・バイトで以下の設定を行います。

- ・ウォッチドッグ・タイマの動作禁止
- ・高速内蔵発振回路の周波数を8MHzに設定
- ・LVIデフォルト・スタート機能停止
- ・オンチップ・デバッグを動作許可に設定

2. 78K0R/Lx3シリーズの汎用レジスタは、4レジスタ・バンク構成になっていますので、通常処理で使用するレジスタと割り込み時で使用するレジスタをバンクごとに切り替えることにより、効率のよいプログラムを作成できます。なお、このサンプル・プログラムでは、レジスタ・バンク0のみを使用します。

3. 低電圧検出動作を許可した後、低電圧検出回路の動作安定待ち用に10 μ s以上のウェイト処理を行います。

注意. C言語版のサンプル・プログラムの場合、レジスタ・バンクの設定およびスタック・ポインタの設定をスタートアップ・ルーチンで行いますので、ソース・プログラム (main.c) には記述しません。なお、スタートアップ・ルーチンについての詳細はCC78K0Rの操作編のユーザーズ・マニュアルを参照してください。

第4章 設定方法について

この章では、オプション・バイト、ベクタ・テーブル、スタック・ポインタ、ウォッチドッグ・タイマ、クロック周波数、入出力ポートの設定、およびメイン処理について説明します。

C言語によるプログラムを実行させるには、システムへ組み込むためのROM化処理、ユーザー・プログラム(main関数)の起動などを行うプログラムが必要となります。このプログラムのことをスタート・アップ・ルーチンと呼びます。このスタート・アップ・ルーチンは一般的に、マイコンがリセット(初期化)後、最初に動くプログラムで、CPU、メモリ、I/Oポートなどのハードウェア初期設定や、main関数を動かすための初期設定を行います。プログラムは基本的に、このスタート・アップ・ルーチンから始まり、その次にメイン・ルーチン処理が実行され、その後サブ・ルーチン処理や割り込み処理と続いて実行されます。

このサンプル・プログラムのC言語版では、hdwinit関数によりクロック関連の設定や周辺ハードウェアの初期設定等を行っており、hdwinit関数を実行した後にmain関数を実行するのでメイン処理はmain関数に記述します。アセンブリ言語版ではリセット(初期化)後、ベクタ・テーブルの00000H番地に書かれたRESET_START番地からプログラムが実行され、C言語版のhdwinit関数と同じように、クロック関連の設定や周辺ハードウェアの初期設定等を行い、メイン処理に入ります。

スタート・アップ・ルーチンの詳細については、CC78K0R 操作編 ユーザーズ・マニュアル スタート・アップ・ルーチンの章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、ユーザーズ・マニュアルを参照してください。

アセンブラ命令については、78K0Rマイクロコントローラ 命令編 ユーザーズ・マニュアルを参照してください。



【コラム】hdwinit関数とmain関数

C言語でプログラムを作成する場合、CPUリセット直後に、周辺装置(SFR)の初期化を行うために、hdwinit関数を呼び出します。したがって、クロック周波数選択やポート入出力設定などの初期設定は、基本的にhdwinit関数に記述します。

hdwinit関数を呼び出したあとに、main関数を呼び出しますので、メイン処理はmain関数に記述します。

main関数からは、hdwinit関数を呼び出さないでください。この場合、hdwinit関数は2度実行され、1度のみしか許可されていないウォッチドッグ・タイマの設定が、2度実行されてしまいます。その結果、2度目の実行時に内部リセット信号が発生され、プログラムが初期設定から進まなくなります。

詳細については、CC78K0Rの言語編のユーザーズ・マニュアル、当社HPのFAQの「初期設定」「プログラミング」の『最初に行う内容について』を参照してください。

4.1 オプション・バイトの設定

オプション・バイトの設定は、必須です。オプション・バイトで、次の項目を設定します。

- (1) ウォッチドッグ・タイマのカウンタ動作設定
- (2) ウォッチドッグ・タイマのインターバル時間の設定
- (3) ウォッチドッグ・タイマのウインドウ・オープン期間の設定
- (4) LVIデフォルト・スタートの動作制御
- (5) 高速内蔵発振回路の周波数の選択
- (6) オンチップ・デバッグ動作制御

図4-1-1 オプション・バイトのフォーマット (1/4)

アドレス：000C0H/010C0H^{注1}

WDTINT	WINDOW1	WINDOW0	WDTON	WDCS2	WDCS1	WDCS0	WDSTBYON	
WDSTBYON		ウォッチドッグ・タイマのカウンタ動作制御 (HALT/STOPモード時)						
0		HALT/STOPモード時, カウンタ動作停止 ^{注2}						
1		HALT/STOPモード時, カウンタ動作許可						
WDCS2			WDCS1			WDCS0		ウォッチドッグ・タイマのオーバフロー時間 ^{注3}
0			0			0		$2^7/f_{IL}$ (3.88ms)
0			0			1		$2^8/f_{IL}$ (7.76ms)
0			1			0		$2^9/f_{IL}$ (15.52ms)
0			1			1		$2^{10}/f_{IL}$ (31.03ms)
1			0			0		$2^{12}/f_{IL}$ (124.12ms)
1			0			1		$2^{14}/f_{IL}$ (496.48ms)
1			1			0		$2^{15}/f_{IL}$ (992.97ms)
1			1			1		$2^{17}/f_{IL}$ (3971.88ms)
WDTON		ウォッチドッグ・タイマのカウンタの動作制御						
0		カウンタ動作禁止 (リセット解除後, カウント停止)						
1		カウンタ動作許可 (リセット解除後, カウント開始)						
WINDOW1		WINDOW0		ウォッチドッグ・タイマのウインドウ・オープン期間 ^{注2}				
0		0		25%				
0		1		50%				
1		0		75%				
1		1		100%				
WDTINT		ウォッチドッグ・タイマのインターバル割り込みの使用 / 不使用						
0		インターバル割り込みを使用しない						
1		オーバフロー時間の75%到達時にインターバル割り込みを発生する						

注1. ブート・スワップ使用時は, 000C0Hと010C0Hが切り替わるので, 010C0Hにも000C0Hと同じ値を設定してください。

2. WDSTBYON = 0のときは, WINDOW1, WINDOW0の値に関係なく, ウインドウ・オープン期間100%となります。
3. ()内は $f_{IL} = 33 \text{ kHz (MAX.)}$ の場合。

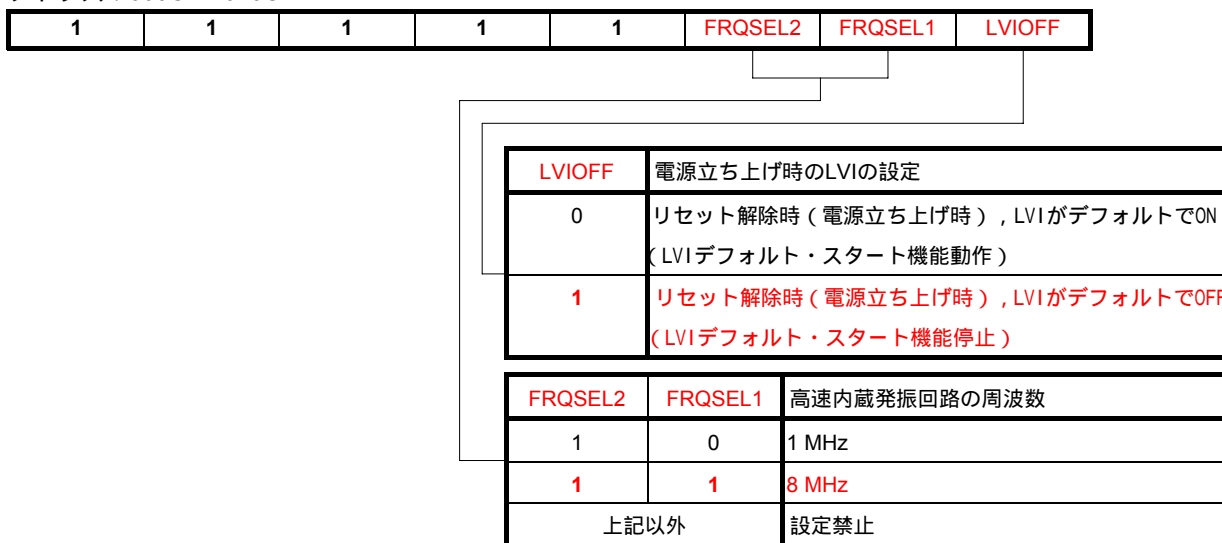
注意. フラッシュ・メモリのセルフ・プログラミング時およびEEPROMエミュレーション時でも, ウォッチドッグ・タイマの動作は継続します。ただし, これらの処置中には割り込みの受け付け時間が遅れるので, 遅延を考慮し, オーバフロー時間およびウインドウ・サイズを設定してください。

備考1. f_{IL} : 低速内蔵発振クロック周波数

2. 図の赤字部分がサンプル・プログラムでの設定値となります

図4-1-2 オプション・バイトのフォーマット (2/4)

アドレス : 000C1H/010C1H^注



注. ブート・スワップ使用時は, 000C1Hと010C1Hが切り替わるので, 010C1Hにも000C1Hと同じ値を設定してください。

注意1. ビット7-3には, 必ず1を書き込んでください。

2. LVIデフォルト・スタート機能使用時でも, ソフトウェアでLVI動作禁止(LVIMレジスタのビット7(LVION) = 0) に設定した場合には, 次の動作となります。

- ・ LVION = 0の期間は低電圧検出しない。
- ・ LVION = 0の期間にリセットが発生した場合, リセット解除後にCPUがスタートするとLVION = 1に再設定されます。しかし, WDTによるリセットと不正命令の実行によるリセットが発生した場合は, 正常に低電圧検出できない期間があります。これはLVIが検出するパルス幅が最小200 μs必要なのに対し, リセット発生によってLVION = 1に設定されて, LVIの動作安定を待たずにCPUが動作を開始するためです。

備考. 図の赤字部分がサンプル・プログラムでの設定値となります

図4-1-3 オプション・バイトのフォーマット (3/4)

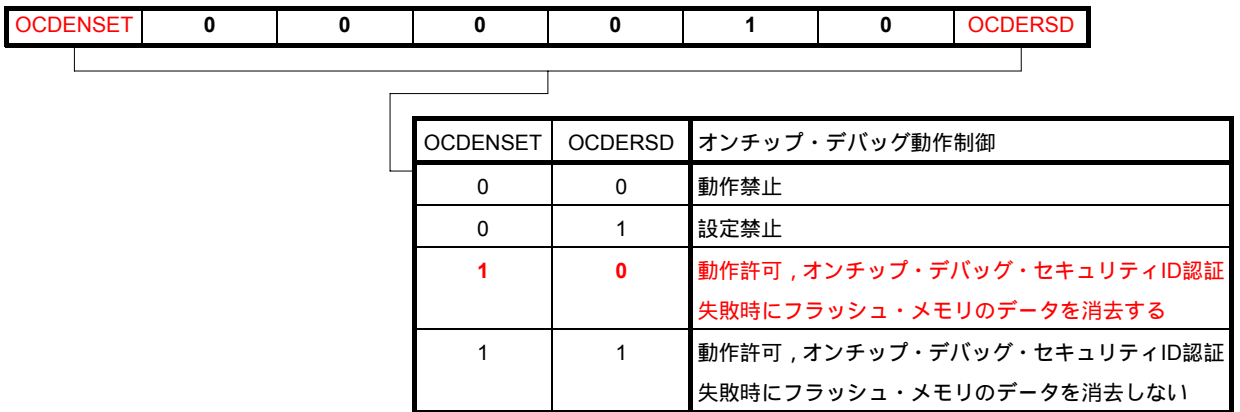
アドレス : 000C2H/010C2H^注

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

注. 000C2Hは予約領域なので, 必ずFFHを設定してください。またブート・スワップ時は, 000C2Hと010C2Hが切り替わるので, 010C2HにもFFHを設定してください。

図4 - 1 - 4 オプション・バイトのフォーマット (4/4)

アドレス : 000C3H/010C3H^注



注. ブート・スワップ時は, 000C3Hと010C3Hが切り替わるので, 010C3Hにも000C3Hと同じ値を設定してください。

注意1. ビット7, 0 (OCDENSET, OCDERSD) のみ, 値を指定できます。ビット6-1には, 必ず000010Bを書き込んでください。

2. ビット3-1は, オンチップ・デバッグ機能使用時に値が書き変わるので, 設定後は不定となります。ただし, 設定時にはビット3-1にも, 必ず初期値 (0, 1, 0) を設定してください。

備考1. 図の赤字部分がサンプル・プログラムでの設定値となります

2. このサンプル・プログラムでは, ソース・ファイル (ファイル名: 「op.asm」) にてオプション・バイトの設定を行いますので, RA78K0Rのリンカ・オプションにてオプション・バイトの設定を行う必要はありません。

前述したオプション・バイトの設定値をソフトウェアで記述すると次のようになります。

```
XOPTB   CSEG   OPT_BYTE
         DB     01101110B
         DB     11111111B
         DB     11111111B
         DB     10000100B
```

C言語を使用する場合は, 次のようなアセンブリ言語のソース・ファイル (ファイル名: 「*.asm (*: 任意)」) を準備し, プロジェクトのソース・ファイルに指定して, 他のソース・ファイル (main.c) と一緒にビルドしてください。

```
XOPTB   CSEG   OPT_BYTE
         DB     01101110B
         DB     11111111B
         DB     11111111B
         DB     10000100B

END
```

**【コラム】 CSEG (Code Segment), DSEG (Data Segment), BSEG (Bit Segment) とは**

CSEG, DSEG, BSEGは、命令やデータなどの生成されたコードをどこに配置するかを示す疑似命令です。これらの疑似命令以降に記述された命令やデータは、CSEG疑似命令ではROM領域に、DSEG疑似命令ではRAM領域に、BSEG疑似命令ではRAMのsaddr領域へ配置されます。

例えば、オプション・バイトの設定内容を内部ROM(フラッシュ・メモリ)の000C0Hから配置する場合は、まず、CSEG疑似命令とOPT_BYTE属性を使用します。そのあとに、DB疑似命令で000C0H番地以降に設定したい値を定義し、それをアセンブリ言語にて記述したプログラム内に記述してください。

CSEG疑似命令で指定したROM領域のみ、DB, DW疑似命令は使用可能です。DSEG, BSEG疑似命令で指定したRAM領域では、DB, DW疑似命令の記述はエラーにはなりませんが、使用しないでください。この場合、オブジェクトは生成され、MINICUBE2(オンチップ・デバッグ・エミュレータ)やSM+(システム・シミュレータ)では記述された命令やデータをRAM領域に展開するため、デバッグ動作を行うことができます。しかし、実際のデバイスでは、それらをRAM領域に展開できないため、動作できなくなります。

CSEG, DSEG, BSEG疑似命令の詳細については、RA78K0Rの言語編のユーザーズ・マニュアルを参照してください。

4.2 ベクタ・テーブルの設定

ベクタ・テーブル領域は、リセットや各割り込み要求発生により分岐するときのプログラム・スタート・アドレスを格納する領域です。このサンプル・プログラムでは、割り込み処理は行わないため、リセット・スタート時に使用するリセット・ベクタのみ設定します。

なお、この設定はアセンブリ言語で記述するときに必要な設定です。C言語で記述する場合、設定は不要です。

【設定例】 リセット・スタート時に使用するリセット・ベクタのみ設定（サンプル・プログラムの設定と同内容）

				アドレス	機能名
XVECT1		CSEG	AT	00000H	
	DW	RESET_START			;00000H RESET入力, POC, LVI, WDT, TRAP
XVECT2		CSEG	AT	00004H	
	DW	IINIT			;00004H INTWDTI
	DW	IINIT			;00006H INTLVI
	DW	IINIT			;00008H INTP0
	DW	IINIT			;0000AH INTP1
・・・(略)・・・					
	DW	IINIT			;0005AH INTTM12
	DW	IINIT			;0005CH INTTM13
	DW	IINIT			;0005EH INTMD
・・・(略)・・・					
;*****					
;					
; 不要な割り込み要因による割り込み処理					
;					
;*****					
XMAIN	CSEG	UNIT			
		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、プログラムは、リセット・ベクタで指定したアドレス（上記では、の「RESET_START」）からスタートします。

このサンプル・プログラムでは、ベクタ・テーブル・アドレスの00000H以外は使用しません。残りのベクタ・テーブル・アドレスには、すべて「IINIT」を設定します（上記）。このように設定することにより、万が一、割り込みが発生した場合でも「IINIT」（上記）に分岐させ、不要な割り込みとして何も処理をさせずに割り込みから復帰させます。



【コラム】#pragma指令とは

#pragma指令は、C言語で使用する前処理命令で、ソース・プログラムの冒頭に記述します。

主な#pragma指令は、次のとおりです。

- ・ #pragma sfr : SFR領域に関する操作をCソース・レベルで記述可能
- ・ #pragma ei : EI命令をCソース・レベルで記述可能
- ・ #pragma di : DI命令をCソース・レベルで記述可能
- ・ #pragma nop : NOP命令をCソース・レベルで記述可能（CPUを動作させずにクロックを進めることが可能）
- ・ #pragma interrupt : 割り込み関数をCソース・レベルで記述可能

#pragma指令の詳細は、CC78K0Rの言語編のユーザーズ・マニュアルの拡張機能の章を参照してください。

4.3 スタック・ポインタの設定

スタック領域は、プログラム・カウンタ、レジスタ値、PSW（プログラム・ステータス・ワード）などのデータを、一時的に格納するメモリ領域です。スタック領域は、内部RAMにのみ指定できます。このスタック領域の先頭アドレスはスタック・ポインタで設定し、スタック領域を確保します。

スタック領域は、次の命令実行および割り込み時に使用します。

- ・PUSH, CALL, CALLT, CALLF 割り込み : スタック領域へデータ退避
- ・POP, RET, RETI : スタック領域からデータ復帰

スタック領域の確保は、アセンブリ言語で記述するときに必要な設定です。C言語で記述する場合は、スタートアップ・ルーチンにてスタック領域が自動的に確保されるため、設定は不要です。

【例】 内部RAMの先頭の32バイトを、スタック領域として使用する場合（サンプル・プログラムの設定と同内容）

DSTK	DSEG	BASEP	}	内部高速RAMの先頭に スタック領域確保
STACKEND:	DS	20H		
STACKTOP:				;スタック領域を32バイト確保 ;スタック領域の先頭アドレス
...	(略)	...		
RESET_START:				リセット解除後に スタック・ポインタ設定
...	(略)	...		
	MOVW	SP,	#LOWW STACKTOP	;スタック・ポインタを設定

上記の記述により、内部RAMの先頭の32バイトをスタック領域として確保することが可能となります。
なお、内部RAMの先頭のアドレスはデバイスによって異なりますので、スタックとして確保される領域のアドレスは次のようになります。

- μ PD78F1500, μ PD78F1503, μ PD78F1506の場合 : FEF00H - FEF1FH
- μ PD78F1501, μ PD78F1504, μ PD78F1507の場合 : FE700H - FE71FH
- μ PD78F1502, μ PD78F1505, μ PD78F1508の場合 : FE300H - FE31FH

このサンプル・プログラムではDSEG疑似命令BASEP[※]を使用することで、絶対的なアドレスを記述することなく内部RAMの先頭を指定しています。

注. 詳細はRA78K0Rの言語編のユーザーズ・マニュアルを参照してください。

4.4 ウォッチドッグ・タイマの設定と制御

ウォッチドッグ・タイマの設定は、オプション・バイトで行います。詳細は 4.1 オプション・バイトの設定 を参照してください。

なお、ウォッチドッグ・タイマを使用する場合（WDTON=1の場合）、ウォッチドッグ・タイマの制御は、ウォッチドッグ・タイマ・イネーブル・レジスタ（WDTE）で行います。WDTEにACHを書き込むことにより、ウォッチドッグ・タイマのカウンタをクリアし、再びカウントを開始します。リセット信号により、WDTEは9AH^{*}になります。

注. オプション・バイト（000C0H）のWDTONの設定値によって、WDTEのリセット値が異なります。

WDTONの設定	WDTEのリセット値
0（ウォッチドッグ・タイマのカウント動作禁止）	1AH
1（ウォッチドッグ・タイマのカウント動作許可）	9AH

- 注意1. WDTEにACH以外の値を書き込んだ場合、内部リセット信号を発生します。
- WDTEに1ビット・メモリ操作命令を実行した場合、内部リセット信号を発生します。
 - WDTEのリード値は、“9AH / 1AH”（書き込んだ値（“ACH”）とは異なる値）になります。



【コラム】2進数値の表記

2進数値を表記する場合、アセンブリ言語では2進数値の後ろに「B」または「Y」を、C言語では2進数値の前に「0b」または「0B」を付加してください。

4.5 クロックの設定

CPUおよび周辺ハードウェアへの供給クロック (f_{CLK}) は、メイン・システム・クロック (f_{MAIN}) を分周して、生成されます。

(1) クロック動作モードの選択

クロック動作モード制御レジスタ (CMC) でクロック動作モードを選択します。

図4 - 2 - 1 クロック動作モード制御レジスタ (CMC)

CMC

EXCLK	OSCSEL	0	OSCSELS	0	AMPHS1	AMPHS0	AMPH
-------	--------	---	---------	---	--------	--------	------

AMPH	高速システム・クロック発振周波数の制御	
0	2 MHz	f_{MX} 10 MHz
1	$10 \text{ MHz} < f_{MX}$ 20 MHz	

AMPHS1	AMPHS0	XT1発振回路の発振モード選択
0	0	低消費発振 (デフォルト)
0	1	通常発振
1	0	超低消費発振
1	1	

OSCSELS	サブシステム・クロック端子の動作モード	XT1/P123端子	XT2/P124端子
0	入力ポート・モード	入力ポート	
1	XT1発振モード	水晶発振子接続	

EXCLK	OSCSEL	高速システム・クロック端子の動作モード	X1/P121端子	X2/EXCLK/P122端子
0	0	入力ポート・モード	入力ポート	
0	1	X1発振モード	水晶/セラミック発振子接続	
1	0	入力ポート・モード	入力ポート	
1	1	外部クロック入力モード	入力ポート	外部クロック入力

- 注意1. CMCは、リセット解除後、8ビット・メモリ操作命令で1回のみ書き込み可能です。
- リセット解除後、クロック動作ステータス制御レジスタ（CSC）の設定でX1発振またはXT1発振を開始する前に、CMCを設定してください。
 - X1クロック発振周波数が10MHzを越える場合は、必ずAMPHに1を設定してください。
 - CMCを初期値(00H)のまま使用する場合、暴走時の誤動作を防止するためにリセット解除後は必ず00Hに設定してください。
 - XT1発振回路は低消費電力を実現するために、増幅度が低い回路になっています。設計の際は、次の点に注意してください。
 - 端子や回路基板には寄生容量が含まれています。したがって実際に使用する回路基板にて発振評価を行い、問題がないことを確認してください。
 - 低消費発振および超低消費発振を選択する場合、通常発振を選択する場合に比べ、低消費電力を実現することができます。しかし、この場合、XT1発振余裕度が低下しますので、XT1発振に使用する発振子は、発振評価を十分に行った上で、ご使用ください。
 - XT1端子、XT2端子と発振子との配線は極力短くし、寄生容量、配線抵抗を小さくしてください。特に超低消費発振（AMPHS1 = 1）を選択している場合はご注意ください。
 - 回路基板は寄生容量、配線抵抗の少ない材質で回路を構成してください。
 - XT1発振回路の周辺には、できるかぎり V_{SS} と同電位のグランド・パターンを配置してください。
 - XT1端子、XT2端子と発振子の信号線は他の信号と交差させないでください。また、変化する大電流が流れる線と接近させないでください。
 - 高湿度環境における回路基板の吸湿や、基板上での結露によってXT1端子とXT2端子間のインピーダンスが低下し発振に障害が発生する場合があります。このような環境でご使用される場合は、回路基板をコーティングするなどの防湿対策を行ってください。
 - 回路基板上をコーティングする場合は、XT1端子、XT2端子間に容量やリークが生じない材料をご使用ください。
 - ビット5, 3には、必ず0を書き込んでください。

- 備考1. 図の赤字部分がサンプル・プログラムでの設定値となります
- f_{MX} : 高速システム・クロック周波数

(2) 高速システム・クロック，高速内蔵発振クロック，サブシステム・クロックの動作の制御

クロック動作ステータス制御レジスタ (CSC) で高速システム・クロック，高速内蔵発振クロック，サブシステム・クロックの動作を制御します。

図4 - 2 - 2 クロック動作ステータス制御レジスタ (CSC)

CSC

MSTOP	XTSTOP	0	0	0	0	0	HIOSTOP
-------	--------	---	---	---	---	---	---------

HIOSTOP	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		

XTSTOP	サブシステム・クロックの動作制御		
	XT1発振モード時		入力ポート・モード時
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		
	X1発振モード時	外部クロック入力モード時	入力ポート・モード時
0	X1発振回路動作	EXCLK端子からの外部クロック有効	-
1	X1発振回路停止	EXCLK端子からの外部クロック無効	-

- 注意1. リセット解除後，MSTOPの設定でX1発振またはXTSTOPの設定でXT1発振を開始する前に，クロック動作モード制御レジスタ (CMC) を設定してください。
- MSTOPの設定でX1発振を開始する場合，X1クロックの発振安定時間を発振安定時間カウンタ状態レジスタ (OSTC) で確認してください。
 - CPU / 周辺ハードウェア・クロック (f_{CLK}) に選択しているクロックは，CSCレジスタで停止させないでください。
 - クロック発振停止 (外部クロック入力無効) するためのレジスタのフラグ設定と停止前の条件は，次のようになります。

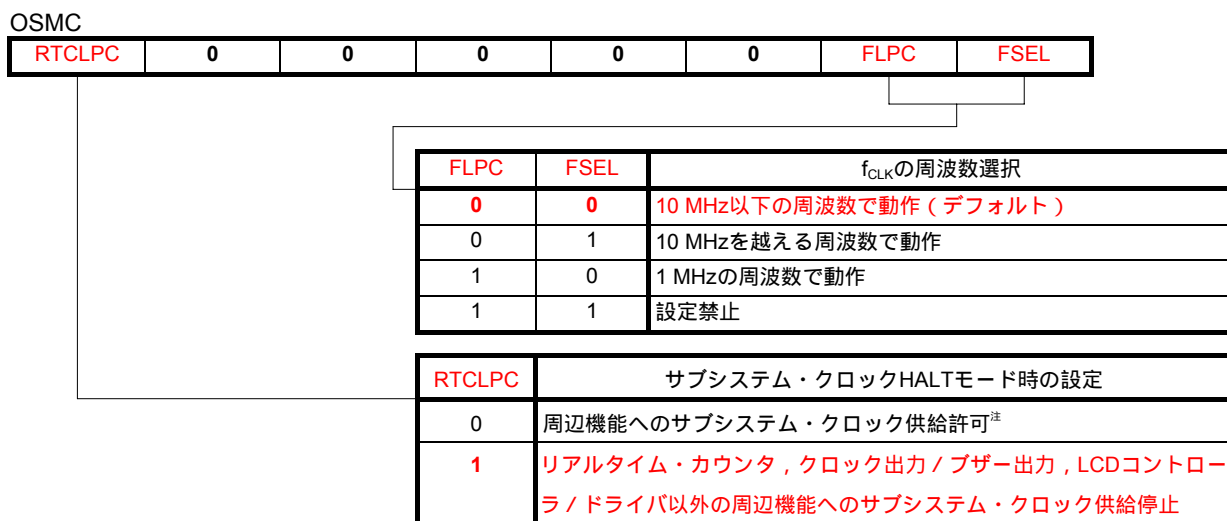
クロック	クロック停止 (外部クロック入力無効) 前条件	CSCレジスタのフラグ設定
X1クロック	CPU / 周辺ハードウェア・クロックが高速システム・クロック以外で動作 ($\cdot CLS = 0$ かつ $MCS = 0$) ($\cdot CLS = 1$)	MSTOP = 1
外部メイン・システム・クロック		
サブシステム・クロック	CPU / 周辺ハードウェア・クロックがサブシステム・クロック以外で動作 ($CLS = 0$)	XTSTOP = 1
高速内蔵発振クロック	CPU / 周辺ハードウェア・クロックが高速内蔵発振クロック以外で動作 ($\cdot CLS = 0$ かつ $MCS = 1$) ($\cdot CLS = 1$)	HIOSTOP = 1

- ビット5-1には，必ず0を書き込んでください。

(3) フラッシュ・メモリの高速動作用昇圧回路の制御

動作スピード・モード制御レジスタ (OSMC) でフラッシュ・メモリの高速動作用昇圧回路を制御します。

図4 - 2 - 3 動作スピード・モード制御レジスタ (OSMC)



注. 動作許可となる周辺機能については、ユーザズ・マニュアルのスタンバイ機能の章を参照してください。

注意1. FSELに“1”を書き込む場合は、必ず以下の2つの動作よりも前に行なってください。

- ・ f_{CLK}をf_{IH}以外に切り替える
 - ・ DMAコントローラを動作させる
2. FSELに“1”を書き込むと、CPUはウエイトします。ウエイト時間は、f_{CLK} = f_{IH}のとき15 μs ~ 20 μs (ターゲット)、f_{CLK} = f_{IH}/2のとき30 μs ~ 40 μs (ターゲット)です。ただし、CPUウエイト中も、f_xの発振安定時間のカウントは継続可能です。
 3. f_{CLK}を10 MHz以上にする場合には、FSELを“1”にセットしてから、2クロック以上経過後に切り替えてください。
 4. FSEL = 0に設定する場合は、10 MHz以下で動作していることを確認してから設定してください。
 5. V_{DD} 2.7 VでSTOPモードに移行する場合は、f_{CLK}を10 MHz以下に設定したあとに、FSEL = 0にしてください。
 6. RTCLPCを1に設定することでサブシステム・クロック時のHALTモード電流を低減できます。ただし、サブシステム・クロックHALTモード中はリアルタイム・カウンタ以外の周辺機能へクロックを供給できなくなります。サブシステム・クロックHALTモードに設定する前に、PER0のビット7 (RTCEN) は1に、それ以外のPER0の6-0ビットは0にしてください。
 7. FLPCは一度0 1に設定すると、リセット以外には、1 0に戻すことは禁止です。
 8. FSELに“1”を書き込む場合、RMC = 00Hの状態で行ってください。また、FLPCに“1”を書き込む場合、RMC = 5AHの状態で行ってください。
 9. ビット6-2には、必ず0を書き込んでください。

(4) CPU / 周辺ハードウェア・クロックの設定

システム・クロック制御レジスタ (CKC) でCPU / 周辺ハードウェア・クロックの選択, および分周比を設定します。

図4 - 2 - 4 システム・クロック制御レジスタ (CKC)

CKC

CLS	CSS	MCS	MCM0	SDIV	MDIV2	MDIV1	MDIV0																																																																																										
<table border="1"> <thead> <tr> <th>CSS</th> <th>MCM0</th> <th>SDIV</th> <th>MDIV2</th> <th>MDIV1</th> <th>MDIV0</th> <th>CPU / 周辺ハードウェア・クロック (f_{CLK}) の選択</th> </tr> </thead> <tbody> <tr> <td rowspan="6">0</td> <td rowspan="6">0</td> <td>x</td> <td>0</td> <td>0</td> <td>0</td> <td>f_{IH}</td> </tr> <tr> <td>x</td> <td>0</td> <td>0</td> <td>1</td> <td>f_{IH}/2</td> </tr> <tr> <td>x</td> <td>0</td> <td>1</td> <td>0</td> <td>f_{IH}/2²</td> </tr> <tr> <td>x</td> <td>0</td> <td>1</td> <td>1</td> <td>f_{IH}/2³ 注1</td> </tr> <tr> <td>x</td> <td>1</td> <td>0</td> <td>0</td> <td>f_{IH}/2⁴ 注1</td> </tr> <tr> <td>x</td> <td>1</td> <td>0</td> <td>1</td> <td>f_{IH}/2⁵ 注1</td> </tr> <tr> <td rowspan="6">0</td> <td rowspan="6">1</td> <td>x</td> <td>0</td> <td>0</td> <td>0</td> <td>f_{MX}</td> </tr> <tr> <td>x</td> <td>0</td> <td>0</td> <td>1</td> <td>f_{MX}/2</td> </tr> <tr> <td>x</td> <td>0</td> <td>1</td> <td>0</td> <td>f_{MX}/2²</td> </tr> <tr> <td>x</td> <td>0</td> <td>1</td> <td>1</td> <td>f_{MX}/2³</td> </tr> <tr> <td>x</td> <td>1</td> <td>0</td> <td>0</td> <td>f_{MX}/2⁴</td> </tr> <tr> <td>x</td> <td>1</td> <td>0</td> <td>1</td> <td>f_{MX}/2⁵ 注2</td> </tr> <tr> <td rowspan="2">1注3</td> <td rowspan="2">x注3</td> <td>0</td> <td>x</td> <td>x</td> <td>x</td> <td>f_{SUB}</td> </tr> <tr> <td>1</td> <td>x</td> <td>x</td> <td>x</td> <td>f_{SUB}/2</td> </tr> <tr> <td colspan="6">上記以外</td> <td>設定禁止</td> </tr> </tbody> </table>								CSS	MCM0	SDIV	MDIV2	MDIV1	MDIV0	CPU / 周辺ハードウェア・クロック (f _{CLK}) の選択	0	0	x	0	0	0	f _{IH}	x	0	0	1	f _{IH} /2	x	0	1	0	f _{IH} /2 ²	x	0	1	1	f _{IH} /2 ³ 注1	x	1	0	0	f _{IH} /2 ⁴ 注1	x	1	0	1	f _{IH} /2 ⁵ 注1	0	1	x	0	0	0	f _{MX}	x	0	0	1	f _{MX} /2	x	0	1	0	f _{MX} /2 ²	x	0	1	1	f _{MX} /2 ³	x	1	0	0	f _{MX} /2 ⁴	x	1	0	1	f _{MX} /2 ⁵ 注2	1注3	x注3	0	x	x	x	f _{SUB}	1	x	x	x	f _{SUB} /2	上記以外						設定禁止
CSS	MCM0	SDIV	MDIV2	MDIV1	MDIV0	CPU / 周辺ハードウェア・クロック (f _{CLK}) の選択																																																																																											
0	0	x	0	0	0	f _{IH}																																																																																											
		x	0	0	1	f _{IH} /2																																																																																											
		x	0	1	0	f _{IH} /2 ²																																																																																											
		x	0	1	1	f _{IH} /2 ³ 注1																																																																																											
		x	1	0	0	f _{IH} /2 ⁴ 注1																																																																																											
		x	1	0	1	f _{IH} /2 ⁵ 注1																																																																																											
0	1	x	0	0	0	f _{MX}																																																																																											
		x	0	0	1	f _{MX} /2																																																																																											
		x	0	1	0	f _{MX} /2 ²																																																																																											
		x	0	1	1	f _{MX} /2 ³																																																																																											
		x	1	0	0	f _{MX} /2 ⁴																																																																																											
		x	1	0	1	f _{MX} /2 ⁵ 注2																																																																																											
1注3	x注3	0	x	x	x	f _{SUB}																																																																																											
		1	x	x	x	f _{SUB} /2																																																																																											
上記以外						設定禁止																																																																																											
<table border="1"> <thead> <tr> <th>MCS注4</th> <th>メイン・システム・クロック (f_{MAIN}) のステータス</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>高速内蔵発振クロック (f_{IH})</td> </tr> <tr> <td>1</td> <td>高速システム・クロック (f_{MX})</td> </tr> </tbody> </table>								MCS注4	メイン・システム・クロック (f _{MAIN}) のステータス	0	高速内蔵発振クロック (f _{IH})	1	高速システム・クロック (f _{MX})																																																																																				
MCS注4	メイン・システム・クロック (f _{MAIN}) のステータス																																																																																																
0	高速内蔵発振クロック (f _{IH})																																																																																																
1	高速システム・クロック (f _{MX})																																																																																																
<table border="1"> <thead> <tr> <th>CLS注4</th> <th>CPU / 周辺ハードウェア・クロック (f_{CLK}) のステータス</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>メイン・システム・クロック (f_{MAIN})</td> </tr> <tr> <td>1</td> <td>サブシステム・クロック (f_{SUB})</td> </tr> </tbody> </table>								CLS注4	CPU / 周辺ハードウェア・クロック (f _{CLK}) のステータス	0	メイン・システム・クロック (f _{MAIN})	1	サブシステム・クロック (f _{SUB})																																																																																				
CLS注4	CPU / 周辺ハードウェア・クロック (f _{CLK}) のステータス																																																																																																
0	メイン・システム・クロック (f _{MAIN})																																																																																																
1	サブシステム・クロック (f _{SUB})																																																																																																

- 注1. f_{IH} = 1 MHzのとき, 設定禁止です。
 2. f_{MX} < 4 MHzのとき, 設定禁止です。
 3. CSS = 1を設定した状態で, MCM0ビットの値を変更することは禁止です。
 4. ビット7, 5は, Read Onlyです。

- 備考1. f_{IH} : 高速内蔵発振クロック周波数
 f_{MX} : 高速システム・クロック周波数
 f_{SUB} : サブシステム・クロック周波数
 2. x : don't care
 3. 図の赤字部分がサンプル・プログラムでの設定値となります

(1) から (4) の設定値をソフトウェアで記述すると、次のようになります。

・アセンブリ言語の場合

```
MOV    CMC,    #00000000B
MOV    CSC,    #11000000B
MOV    OSMC,   #10000000B
MOV    CKC,    #00001000B
```

・C言語の場合

```
CMC    = 0b00000000;
CSC    = 0b11000000;
OSMC   = 0b10000000;
CKC    = 0b00001000;
```


4.6 ポートの設定

注意 各製品により、内蔵するポートは異なるため、設定するポートも異なります。

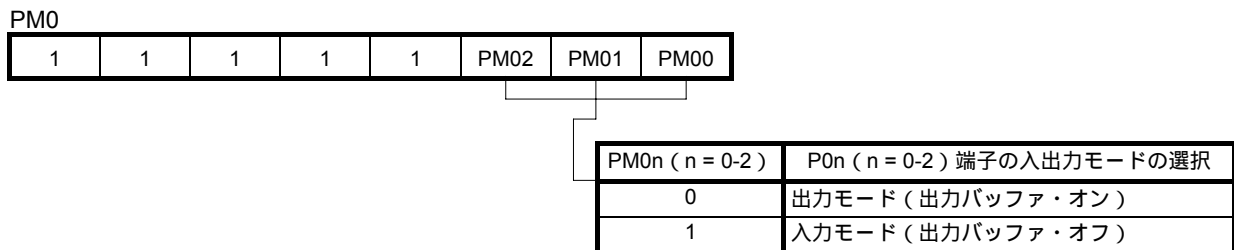
	78K0R/LG3	78K0R/LF3	78K0R/LH3
ポート0	P00-P02	P00-P02	P00-P02
ポート1	P10-P15	P10-P16	P10-P17
ポート2	P20-P26	P20-P27	P20-P27
ポート3	P30-P33	P30-P34	P30-P34
ポート4	P40-P41	P40-P41	P40-P41
ポート5	P50-P57	P50-P57	P50-P57
ポート6	-	P60-P61	P60-P61
ポート7	-	-	P70-P77
ポート8	-	P80-P82	P80-P87
ポート9	P90-P92	P90-P97	P90-P97
ポート10	P100	P100	P100-P102
ポート11	P110-P111	P110-P111	P110-P111
ポート12	P120-P124	P120-P124	P120-P124
ポート13	P130	P130	P130
ポート14	P140-P147	P140-P147	P140-P147
ポート15	P157	P150-P152, P157	P150-P152, P157

(1) ポートの入力/出力の設定

PMxxで、ポートを入力ポートまたは出力ポートとして使用するかを設定します。リセット解除後は、入力ポートに設定されます。

PMxxのフォーマットは、PM0を例にして、説明します。

図4 - 3 - 1 ポート・モード・レジスタ0 (PM0) のフォーマット



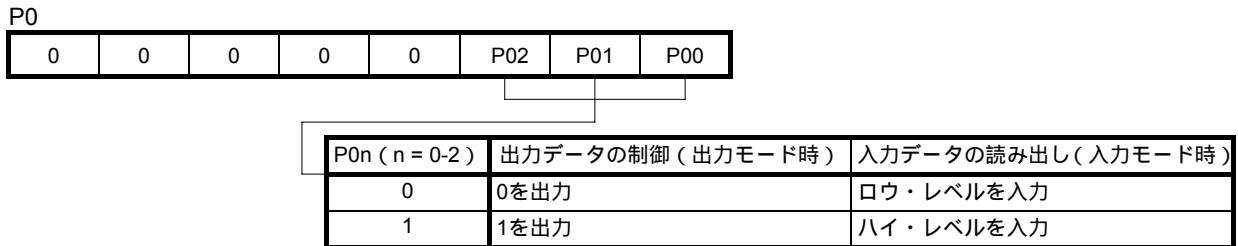
注意. 次のビット7-3には必ず1を設定してください

(2) 出力ポートの出力ラッチの設定

Pxxで、出力ポートの出力ラッチをハイ・レベルまたはロウ・レベルにするかを設定します。リセット解除後は、ロウ・レベル出力に設定されます。

Pxxのフォーマットは、P0を例にして、説明します。

図4 - 3 - 2 ポート・レジスタ0 (P0) のフォーマット



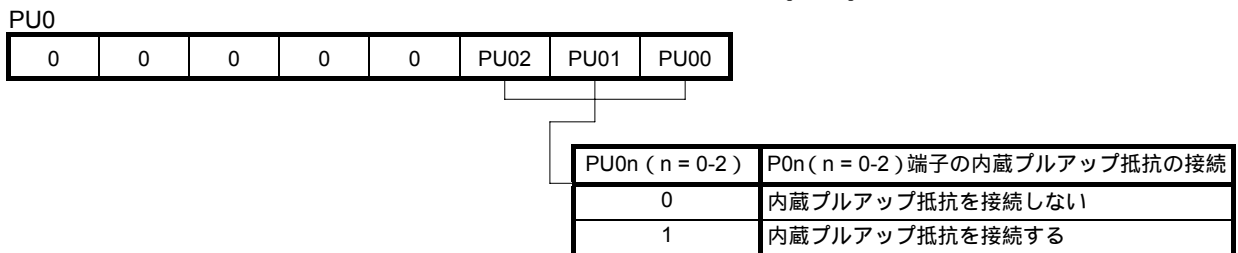
注意. ビット7-3には必ず0を設定してください

(3) 入力ポートの内蔵プルアップ抵抗接続の設定

PUxxで、入力ポートに内蔵プルアップ抵抗を接続するか否かを設定します。リセット解除後は、内蔵プルアップ抵抗に接続しません。

PUxxのフォーマットは、PU0を例にして、説明します。

図4 - 3 - 3 プルアップ抵抗オプション・レジスタ0 (PU0) のフォーマット



注意. ビット7-3には必ず0を設定してください

【例 1】 スイッチ入力用に使用するため，P0を次のように設定する。

- ・ P00，P01を入力ポートに設定
- ・ P00，P01に内蔵プルアップ抵抗を接続する
(サンプル・プログラムの設定と同内容)

PM0

1	1	1	1	1	0 ^注	1	1
---	---	---	---	---	----------------	---	---

P00，P01端子の入出力モードの選択

1	入力モード
---	-------

注. 未使用端子は出力ポートとします

PU0

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

P00，P01端子の内蔵プルアップ抵抗の接続

1	内蔵プルアップ抵抗を接続する
---	----------------

なお，このサンプル・プログラムではスイッチ入力 (SW1，SW2) をロウ・アクティブで使用しますので，スイッチ入力用ポート (P00, P01) にはスイッチがオンのときにロウ・レベル (0)，スイッチがオフのときにハイ・レベル (1) が入力されます。

スイッチ入力 (SW1，SW2) とスイッチ入力用ポート (P00, P01) の関係は次のとおりです。

スイッチ入力		スイッチ入力用ポート	
SW1	SW2	P00	P01
オン	オン	0	0
オフ	オン	1	0
オン	オフ	0	1
オフ	オフ	1	1

ソフトウェアを記述すると，次のようになります。

【アセンブリ言語】

```
MOV    P0,      #00000000B
MOV    PM0,     #11111011B
MOV    PU0,     #00000011B
```

【C言語】

```
P0      = 0b00000000;
PM0     = 0b11111011;
PU0     = 0b00000011;
```

【例 2】 LED出力用に使用するため、P3を次のように設定する。

- ・ P30, P31, P32を出力ポートに設定
- ・ P30, P31, P32の出力ラッチをハイ・レベル出力に設定する
(サンプル・プログラムの設定と同内容)

PM3

・ 78K0R/LF3の場合

1	1	1	1	0 ^注	0	0	0
---	---	---	---	----------------	---	---	---

P30, P31, P32端子の入出力モードの選択

0	出力モード
---	-------

・ 78K0R/LG3, 78K0R/LH3の場合

1	1	1	0 ^注	0 ^注	0	0	0
---	---	---	----------------	----------------	---	---	---

P30, P31, P32端子の入出力モードの選択

0	出力モード
---	-------

注. 未使用端子は出力ポートとします

P3

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

P30, P31, P32端子の出力ラッチのレベル選択

1	ハイ・レベル出力
---	----------

なお、このサンプル・プログラムではLED点灯 (LED1, LED2, LED3) をロウ・アクティブで使用しますので、LED出力用ポート (P30, P31, P32) の出力値が0のときに点灯、1のときに消灯となります。

LED出力用ポート (P30, P31, P32) とLED点灯 (LED1, LED2, LED3) の関係は次のとおりです。

LED出力用ポート			LED点灯		
P30	P31	P32	LED1	LED2	LED3
0	1	1	点灯	消灯	消灯
1	0	1	消灯	点灯	消灯
1	1	0	消灯	消灯	点灯
1	1	1	消灯	消灯	消灯

ソフトウェアを記述すると、次のようになります。

・78K0R/LF3の場合

[アセンブリ言語]

```
MOV    P3,      #00000111B
MOV    PM3,     #11110000B
```

[C言語]

```
P3     = 0b00000111;
PM3    = 0b11110000;
```

・78K0R/LG3 , 78K0R/LH3の場合

[アセンブリ言語]

```
MOV    P3,      #00000111B
MOV    PM3,     #11100000B
```

[C言語]

```
P3     = 0b00000111;
PM3    = 0b11100000;
```

4.7 メイン処理

アセンブリ言語のメイン処理では、次の動作を行います。

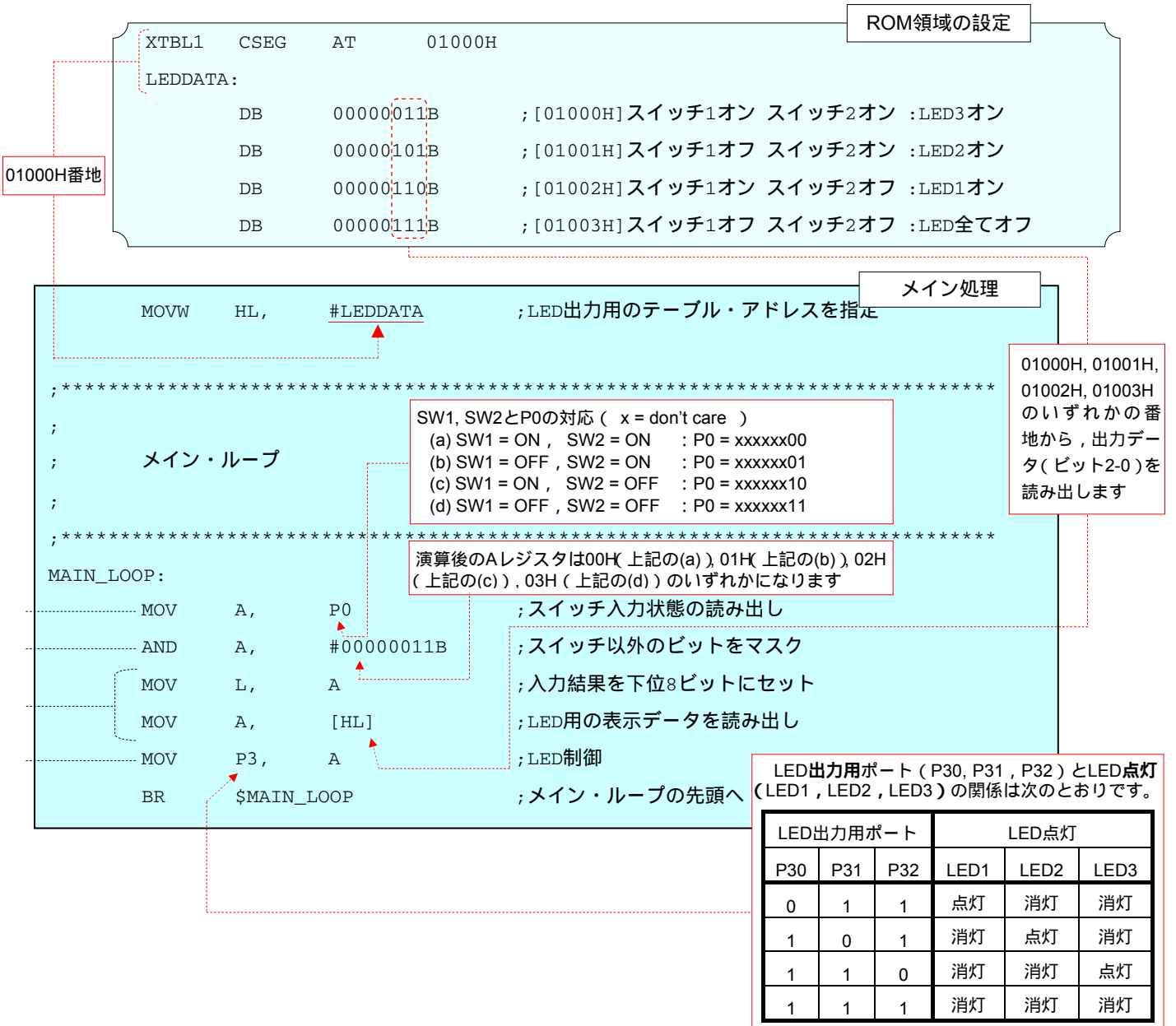
P0のデータを読み出します

読み出した8ビット・データのうち、スイッチ入力用ポート（P00，P01）以外のビットを0にします

01000H-01003H番地（「LEDDATA」テーブル）から、P00，P01の入力レベルの組み合わせに対応する出力データを読み出します。

読み出した出力データをP3に出力します

この動作により、P00，P01に接続しているスイッチ（SW1，SW2）の入力の組み合わせのみを判断することが可能になります。なお、このサンプル・プログラムでは、スイッチをロウ・アクティブで使用しますので、P00，P01にはスイッチがオンのときにロウ・レベル（0），スイッチがオフのときにハイ・レベル（1）が入力されます。



C言語のメイン処理も、アセンブリ言語と同様な動作を行います。

C言語の場合は、入力データと出力データの対応を、配列で設定します。

```

/*****
*****

メイン・ループ

*****
*****/
void main(void)
{
    const unsigned char aLedOut[4]
    = {0b00000011,0b00000101,0b00000110,0b00000111}; /* LED出力用のテーブル */
    unsigned char ucSwitchBuffer; /* スイッチ入力データ保存領域 */

    while(1){
        /* 有効スイッチ情報取得 */
        ucSwitchBuffer = ( P0 & 0b00000011 );

        /* テーブルから表示データを読み出して表示 */
        P3 = ( aLedOut[ucSwitchBuffer] & 0b00000111 );
    }
}

```

{ } 内に4個のデータがあると定義し、その中に出力データを設定しています。

入力データと出力データの対応は次のようになります。

スイッチ入力	P00, P01	ucSwitchBuffer	aLedOut	LED点灯
SW1 = ON, SW2 = ON	P00 = 0, P01 = 0	0b00000000	0b00000011	LED3のみ点灯
SW1 = OFF, SW2 = ON	P00 = 1, P01 = 0	0b00000001	0b00000101	LED2のみ点灯
SW1 = ON, SW2 = OFF	P00 = 0, P01 = 1	0b00000010	0b00000110	LED1のみ点灯
SW1 = OFF, SW2 = OFF	P00 = 1, P01 = 1	0b00000011	0b00000111	全LED消灯

第5章 関連資料

資料名	和文 / 英文
78K0R/Lx3 ユーザーズ・マニュアル	PDF
78K0Rマイクロコントローラ 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	言語編 PDF
ユーザーズ・マニュアル	操作編 PDF
CC78K0R Cコンパイラ	言語編 PDF
ユーザーズ・マニュアル	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

付録A プログラム・リスト

プログラム・リスト例として、78K0R/LH3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。

```
main.asm (アセンブリ言語版)
;*****
;
;
;   NEC Electronics    78K0R/LH3シリーズ
;
;*****
;
;   78K0R/LH3シリーズ サンプル・プログラム (初期設定)
;*****
;
;   LED点灯のスイッチ制御編
;*****
; 【履歴】
;
;   2009.3.-- 新規作成
;*****
;
; 【概要】
;本サンプル・プログラムでは、クロック周波数の選択，入出力ポートの設定などを行う
;ことで、マイコンの基本的な初期設定を行う。初期設定完了後のメイン・ループでは、
;2つのスイッチ入力を検出して3つのLED点灯を制御する。
;
; <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
;   ・高速内蔵発振回路の周波数を8MHzに設定
;   ・LVIデフォルト・スタート機能停止
;   ・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
;
;
;
; <スイッチ入力とLED点灯>
;
;
;   +                               +
;   | スイッチ1 | スイッチ2 | LED1 | LED2 | LED3 |
```

```

; | (P00) | (P01) | (P30) | (P31) | (P32) |
; |
; | OFF | OFF | OFF | OFF | OFF |
; | ON  | OFF | ON  | OFF | OFF |
; | OFF | ON  | OFF | ON  | OFF |
; | ON  | ON  | OFF | OFF | ON  |
; +
;   スイッチがONのときポートの入力値は0, OFFのとき1となります。
;   LEDはポートの出力値が1のときOFF(消灯), 0のときON(点灯)となります。
;
;
; <入出力ポートの設定>
; 入力ポート：P00-P01
; 出力ポート：P30-P32
;   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
            DW     IINIT             ;00014H  INTST3
            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      IINIT      ;00026H  INTSR1
DW      IINIT      ;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
DW      IINIT      ;0003CH  INTST2/INTCSI20/INTIIC20
DW      IINIT      ;0003EH  INTSR2
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
DW      IINIT      ;0004EH  INTP8
DW      IINIT      ;00050H  INTP9
DW      IINIT      ;00052H  INTP10
DW      IINIT      ;00054H  INTP11
DW      IINIT      ;00056H  INTTM10
DW      IINIT      ;00058H  INTTM11
DW      IINIT      ;0005AH  INTTM12
DW      IINIT      ;0005CH  INTTM13
DW      IINIT      ;0005EH  INTMD

```

=====

;

; ROMの定義

;

=====

XTBL1 CSEG AT 01000H

LEDDATA:

```

DB      0000011B    ;[01000H]スイッチ1オン スイッチ2オン :LED3オン
DB      00000101B    ;[01001H]スイッチ1オフ スイッチ2オン :LED2オン
DB      00000110B    ;[01002H]スイッチ1オン スイッチ2オフ :LED1オン
DB      00000111B    ;[01003H]スイッチ1オフ スイッチ2オフ :LED全てオフ

```

```

;=====
;
;   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

;*****
;
;   不要な割り込み要因による割り込み処理
;
;*****
XMAIN  CSEG   UNIT
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           ;割り込み禁止

;-----
;   レジスタ・バンクの設定
;-----
        SEL     RBO           ;レジスタ・バンク設定

;-----
;   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

;-----
;    レジスタ値の初期化
;-----
MOVW    HL,    #LEDDATA    ;LED出力用のテーブル・アドレスを指定

;-----
;    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    ;割り込みを許可する場合は
;          ;EIの前の";"を削除します。

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]

```

```

;|||          +----- ANI15/P157
;|||          |+++----- ANI10-ANI8/P152-P150
;|||          |||++++++--- ANI7-ANI0/P27-P20
;|||          0000:AAAAAAAAAA
;|||          00001:AAAAAAAAAAD
;|||          00010:AAAAAAAAAAD
;|||          00011:AAAAAAAAADD
;|||          00100:AAAAAAAADDD
;|||          00101:AAAAAAAADDDD
;|||          00110:AAAAAADDDDD
;|||          00111:AAAADDDDDDD
;|||          01000:AAAADDDDDDD
;|||          01001:AAADDDDDDDDD
;|||          01010:AADDDDDDDDD
;|||          01011:ADDDDDDDDD
;|||          10000:DDDDDDDDDD
;+++----- 必ず0に設定

```

; ポート0の設定

```

MOV    P0,    #0000000B    ;P00-P02の出力ラッチLow
MOV    PM0,   #11111011B   ;P00-P01を入力ポートに設定
                                ;P02を出力ポートに設定
MOV    PU0,   #00000011B   ;P00-P01に内蔵プルアップ抵抗を使用
                                ;P02に内蔵プルアップ抵抗を使用
                                ;P00:スイッチ1入力用に使用
                                ;P01:スイッチ2入力用に使用
                                ;P02:未使用

```

; ポート1の設定

```

MOV    P1,    #0000000B    ;P10-P17の出力ラッチLow
MOV    PM1,   #0000000B    ;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,    #0000000B    ;P20-P27の出力ラッチLow
MOV    PM2,   #0000000B    ;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,      #00000111B      ;P30-P32の出力ラッチHigh
                                           ;P33-P34の出力ラッチLow
MOV     PM3,     #11100000B      ;P30-P34を出力ポートに設定
                                           ;P30:LED1出力用に使用する
                                           ;P31:LED2出力用に使用する
                                           ;P32:LED3出力用に使用する
                                           ;P33-P34:未使用

;-----
;   ポート4の設定
;-----
MOV     P4,      #00000000B      ;P40-P41の出力ラッチLow
MOV     PM4,     #11111100B      ;P40-P41を出力ポートに設定
                                           ;P40-P41:未使用

;-----
;   ポート5の設定
;-----
MOV     P5,      #00000000B      ;P50-P57の出力ラッチLow
MOV     PM5,     #00000000B      ;P50-P57を出力ポートに設定
                                           ;P50-P57: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-P87の出力ラッチLow
MOV      PM8,     #00000000B      ;P80-P87を出力ポートに設定
                                           ;P80-P87:未使用

;-----
;      ポート9の設定
;-----

MOV      P9,      #00000000B      ;P90-P97の出力ラッチLow
MOV      PM9,     #00000000B      ;P90-P97を出力ポートに設定
                                           ;P90-P97:未使用

;-----
;      ポート10の設定
;-----

MOV      P10,     #00000000B      ;P100-P102の出力ラッチLow
MOV      PM10,    #11111000B      ;P100-P102を出力ポートに設定
                                           ;P100-P102:未使用

;-----
;      ポート11の設定
;-----

MOV      P11,     #00000000B      ;P110-P111の出力ラッチLow
MOV      PM11,    #11111100B      ;P110-P111を出力ポートに設定
                                           ;P110-P111: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は入力専用ポート

;-----
;      ポート14の設定
;-----

MOV      P14,     #00000000B      ;P140-P147の出力ラッチLow
MOV      PM14,    #00000000B      ;P140-P147を出力ポートに設定
                                           ;P140-P147:未使用

;-----
;      ポート15の設定
;-----

```



```

MOV     P15,    #0000000B      ;P150-P152,P157の出力ラッチLow
MOV     PM15,   #01111000B     ;P150-P152,P157を出力ポートに設定
                                           ;P150-P152,P157: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(10us以上)
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。
;*****
SINICKL:
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-AMPHSO
;||||| [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,   #1000000B    ;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,    #0000100B    ;クロック選択
;| +| +++++----- CSS/MCMO/MDIV2-MDIV0
;| | | | | | | | | | [CPU/周辺ハードウェア・クロック(fCLK)の選択]
;| | | | | | | | | | 00x000:fIH
;| | | | | | | | | | 00x001:fIH/2(デフォルト)
;| | | | | | | | | | 00x010:fIH/2^2
;| | | | | | | | | | 00x011:fIH/2^3
;| | | | | | | | | | 00x100:fIH/2^4
;| | | | | | | | | | 00x101:fIH/2^5
;| | | | | | | | | | 01x000:FMX
;| | | | | | | | | | 01x001:FMX/2
;| | | | | | | | | | 01x010:FMX/2^2
;| | | | | | | | | | 01x011:FMX/2^3
;| | | | | | | | | | 01x100:FMX/2^4

```

```

;| | 01x101: fMX/2^5
;| | 1x0xxx: fSUB
;| | 1x1xxx: fSUB/2
;| | ( x : don't care )
;| +----- 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)

```

RET

```

;*****
;
;   メイン・ループ
;
;*****

```

MAIN_LOOP:

```

MOV    A,    P0           ;スイッチ入力状態の読み出し
AND    A,    #00000011B  ;スイッチ以外のビットをマスク
MOV    L,    A           ;入力結果を下位8ビットにセット
MOV    A,    [HL]        ;LED用の表示データを読み出し
MOV    P3,   A           ;LED制御
BR     $MAIN_LOOP       ;メイン・ループの先頭へ

```

end

main.c (C言語版)

/******

NEC Electronics 78K0R/LH3シリーズ

78K0R/LH3シリーズ サンプル・プログラム (初期設定)

LED点灯のスイッチ制御編

【履歴】

2009.1.-- 新規作成

【概要】

本サンプル・プログラムでは、クロック周波数の選択，入出力ポートの設定などを行うことで、マイコンの基本的な初期設定を行う。初期設定完了後のメイン・ループでは、2つのスイッチ入力を検出して3つのLED点灯を制御する。

<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
- ・高速内蔵発振回路の周波数を8MHzに設定
- ・LVIデフォルト・スタート機能停止

(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

<スイッチ入力とLED点灯>

+						+				
	スイッチ1		スイッチ2		LED1		LED2		LED3	
	(P00)		(P01)		(P30)		(P31)		(P32)	
	OFF		OFF		OFF		OFF		OFF	
	ON		OFF		ON		OFF		OFF	
	OFF		ON		OFF		ON		OFF	
	ON		ON		OFF		OFF		ON	
+										+

スイッチがONのときポートの入力値は0, OFFのとき1となります。

LEDはポートの出力値が1のときOFF(消灯), 0のときON(点灯)となります。

< 入出力ポートの設定 >

入力ポート : P00-P01

出力ポート : P30-P32

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      DI                /* DI命令を記述可能にする */
#pragma      EI                /* EI命令を記述可能にする */
#pragma      NOP               /* NOP命令を記述可能にする */
```

/*=====

関数プロトタイプ宣言

=====*/

```
void fn_InitPort( void );    /* 入出力ポートの設定 */
void fn_InitLvi( void );    /* 低電圧検出 */
void fn_InitClock( void );  /* クロック周波数の設定 */
```

*****/

リセット解除後の初期化処理

*****/

```
void hdwinit( void )
```

```
{
```

```
/*-----
```

割り込み禁止

```
-----*/
```

```
DI();
```

```
/* 割り込み禁止 */
```

```

/*-----
    入出力ポートの設定
-----*/
fn_InitPort();          /* 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
-----*/
fn_InitLvi();          /* 2.7V以上の電源電圧を確保 */

/*-----
    クロック周波数の設定
-----*/
fn_InitClock();        /* 高速内蔵発振クロックを8MHzで動作 */

/*-----
    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();          */          /* 割り込みを許可する場合は          */
                                   /* コメントアウトを外します。*/

}

/*****

    入出力ポートの設定

*****/

void fn_InitPort( void )
{
/*-----
    デジタル入出力の設定
-----*/
    ADPC    = 0b00010000;    /* 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]
*/
                                   /* |||          +----- ANI15/P157 */
                                   /* |||          |++++-----
ANI10-ANI8/P152-P150 */
                                   /* |||          |||+++++++---

```

ANI7-ANI0/P27-P20 */

```

/* ||| 00000:AAAAAAAAAAAA */
/* ||| 00001:AAAAAAAAAAAAAD */
/* ||| 00010:AAAAAAAAAAAAADD */
/* ||| 00011:AAAAAAAAAADDD */
/* ||| 00100:AAAAAAAAADDDD */
/* ||| 00101:AAAAAADDDDD */
/* ||| 00110:AAAAAADDDDDDD */
/* ||| 00111:AAAAAADDDDDDD */
/* ||| 01000:AAAADDDDDDDDD */
/* ||| 01001:AAADDDDDDDDDDD */
/* ||| 01010:AADDDDDDDDDDDDD */
/* ||| 01011:ADDDDDDDDDDDDD */
/* ||| 10000:DDDDDDDDDDDDDD */
/* +++----- 必ず0に設定 */

```

/*-----*/

ポート0の設定

-----*/

```

P0          = 0b00000000; /* P00-P02の出力ラッチLow */
PM0         = 0b11111011; /* P00-P01を入力ポートに設定 */
                                   /* P02を出力ポートに設定 */
PU0         = 0b00000011; /* P00-P01に内蔵プルアップ抵抗を使用 */
                                   /* P02に内蔵プルアップ抵抗を使用 */
                                   /* P00:スイッチ1入力用に使用 */
                                   /* P01:スイッチ2入力用に使用 */
                                   /* P02:未使用 */

```

/*-----*/

ポート1の設定

-----*/

```

P1          = 0b00000000; /* P10-P17の出力ラッチLow */
PM1         = 0b00000000; /* P10-P17を出力ポートに設定 */
                                   /* P10-P17:未使用 */

```

/*-----*/

ポート2の設定

-----*/

```

P2          = 0b00000000; /* P20-P27の出力ラッチLow */
PM2         = 0b00000000; /* P20-P27を出力ポートに設定 */
                                   /* P20-P27:未使用 */

```

/*-----*/

ポート3の設定

```

-----*/
P3          = 0b00000111;    /* P30-P32の出力ラッチHigh */
                                   /* P33-P34の出力ラッチLow */
PM3         = 0b11100000;    /* P30-P34を出力ポートに設定 */
                                   /* P30:LED1出力用に使用する */
                                   /* P31:LED2出力用に使用する */
                                   /* P32:LED3出力用に使用する */
                                   /* P33-P34:未使用 */

```

/*-----*/

ポート4の設定

```

-----*/
P4          = 0b00000000;    /* P40-P41の出力ラッチLow */
PM4         = 0b11111100;    /* P40-P41を出力ポートに設定 */
                                   /* P40-P41:未使用 */

```

/*-----*/

ポート5の設定

```

-----*/
P5          = 0b00000000;    /* P50-P57の出力ラッチLow */
PM5         = 0b00000000;    /* P50-P57を出力ポートに設定 */
                                   /* P50-P57:未使用 */

```

/*-----*/

ポート6の設定

```

-----*/
P6          = 0b00000000;    /* P60-P61の出力ラッチLow */
PM6         = 0b11111100;    /* P60-P61を出力ポートに設定 */
                                   /* P60-P61:未使用 */

```

/*-----*/

ポート7の設定

```

-----*/
P7          = 0b00000000;    /* P70-P77の出力ラッチLow */
PM7         = 0b00000000;    /* P70-P77を出力ポートに設定 */
                                   /* P70-P77:未使用 */

```

/*-----*/

ポート8の設定

```

-----*/
P8          = 0b00000000;    /* P80-P87の出力ラッチLow */
PM8         = 0b00000000;    /* P80-P87を出力ポートに設定 */

```

/* P80-P87:未使用 */

/*-----*/

ポート9の設定

-----*/

P9 = 0b00000000; /* P90-P97の出力ラッチLow */

PM9 = 0b00000000; /* P90-P97を出力ポートに設定 */

/* P90-P97:未使用 */

/*-----*/

ポート10の設定

-----*/

P10 = 0b00000000; /* P100-P102の出力ラッチLow */

PM10 = 0b11111000; /* P100-P102を出力ポートに設定 */

/* P100-P102:未使用 */

/*-----*/

ポート11の設定

-----*/

P11 = 0b00000000; /* P110-P111の出力ラッチLow */

PM11 = 0b11111000; /* P110-P111を出力ポートに設定 */

/* P110-P111:未使用 */

/*-----*/

ポート12の設定

-----*/

P12 = 0b00000000; /* P120の出力ラッチLow */

PM12 = 0b11111110; /* P120を出力ポートに設定 */

/* P120-P124:未使用 */

/* P121-P124は入力専用ポート */

/*-----*/

ポート14の設定

-----*/

P14 = 0b00000000; /* P140-P147の出力ラッチLow */

PM14 = 0b00000000; /* P140-P147を出力ポートに設定 */

/* P140-P147:未使用 */

/*-----*/

ポート15の設定

-----*/

P15 = 0b00000000; /* P150-P152,P157の出力ラッチLow */

PM15 = 0b01111000; /* P150-P152,P157を出力ポートに設定 */

/* P150-P152,P157:未使用 */

}

/*****

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。

*****/

void fn_InitLvi(void)

{

unsigned char ucCounter; /* カウント用変数 */

/* 低電圧検出回路の設定 */

LVIMK = 1; /* INTLVI割り込み禁止 */

LVISEL = 0; /* 検出電圧をVDDに設定 */

LVIS = 0b00001001; /* 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 */

LVIMD = 0; /* 低電圧検出時の動作モードを割り込み信号発生に設定 */

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(10us以上) */

for(ucCounter = 0; ucCounter < 4; ucCounter++){

NOP();

```

}

/* VLVI VDDになるまでのウエイト */
while( LVIF ){
    NOP();
}
LVION = 0; /* 低電圧検出動作停止 */
}

```

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。
 *****/

```

void fn_InitClock( void )
{
    CMC          = 0b00000000; /* 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
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
モード] */
    /*          00: 入力ポート・モード */
    /*          01: XT1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 */
/*          |||||+--- HIOSTOP */
/*          |||||                [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 */

OSMC        = 0b10000000;    /* 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 */

CKC          = 0b00001000;    /* 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] */
/*          ||                    00x000:fIH */
/*          ||                    00x001:fIH/2 (デフォルト) */
/*          ||                    00x010:fIH/2^2 */
/*          ||                    00x011:fIH/2^3 */
/*          ||                    00x100:fIH/2^4 */
/*          ||                    00x101:fIH/2^5 */
/*          ||                    01x000:fMX */

```

```

/*      ||      01x001:fMX/2 */
/*      ||      01x010:fMX/2^2 */
/*      ||      01x011:fMX/2^3 */
/*      ||      01x100:fMX/2^4 */
/*      ||      01x101:fMX/2^5 */
/*      ||      1x0xxx:fSUB */
/*      ||      1x1xxx:fSUB/2 */
/*      ||      ( x : don't care ) */
/*      | +----- 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) */
}

/*****

メイン・ループ

*****/
void main(void)
{
    const unsigned char aLedOut[4]
        = {0b00000011,0b00000101,0b00000110,0b00000111}; /* LED出力用のテーブル */
    unsigned char ucSwitchBuffer; /* スイッチ入力デ
ータ保存領域 */

    while(1){
        /* 有効スイッチ情報取得 */
        ucSwitchBuffer = ( P0 & 0b00000011 );

        /* テーブルから表示データを読み出して表示 */
        P3 = ( aLedOut[ucSwitchBuffer] & 0b00000111 );
    }
}
→

```

付録B 改版履歴

版 数	発行年月	改版箇所	改版内容
第1版	September 2009	-	-

【発行】NECエレクトロニクス株式会社 (<http://www.necel.co.jp/>)

【問い合わせ先】 <http://www.necel.com/contact/ja/>