

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

78K0S/Kx1+

サンプル・プログラム (8ビット・タイマH1)

PWM出力編

この資料は、サンプル・プログラムの動作概要や使用方法、および8ビット・タイマH1のPWM出力機能の設定方法や活用方法を説明したものです。サンプル・プログラムでは、8ビット・タイマH1のPWM出力機能を使用して、パルス出力のデューティを制御し、500 msごとにLEDの輝度を変更します。

対象デバイス

78K0S/KA1+マイクロコントローラ
 78K0S/KB1+マイクロコントローラ
 78K0S/KU1+マイクロコントローラ
 78K0S/KY1+マイクロコントローラ

目次

第1章 概要 ...	3
1.1 初期設定の主な内容 ...	3
1.2 メイン・ループ以降の内容 ...	4
第2章 回路図 ...	5
2.1 回路図 ...	5
2.2 周辺ハードウェア ...	5
第3章 ソフトウェアについて ...	6
3.1 ファイル構成 ...	6
3.2 使用する内蔵周辺機能 ...	7
3.3 初期設定と動作概要 ...	7
3.4 フロー・チャート ...	9
第4章 設定方法について ...	10
4.1 8ビット・タイマH1のPWM出力機能の設定 ...	10
第5章 システム・シミュレータ SM+での動作確認 ...	20
5.1 サンプル・プログラムのビルド ...	20
5.2 SM+での動作 ...	22
第6章 関連資料 ...	27
付録A プログラム・リスト ...	28
付録B 改版履歴 ...	40

- 本資料に記載されている内容は2008年7月現在のものです、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。当社製品の不具合により生じた生命、身体および財産に対する損害の危険を最小限度にするために、冗長設計、延焼対策設計、誤動作防止設計等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、8ビット・タイマH1のPWM出力機能の使用例を示しています。パルス出力のデューティを制御し、500 msごとにLEDの輝度を変更します。

1.1 初期設定の主な内容

初期設定の主な内容は、次のとおりです。

システム・クロック・ソースとして、高速内蔵発振器を選択^注

ウォッチドッグ・タイマの動作停止

V_{LVI} (低電圧検出電圧) を4.3 V ± 0.2 Vに設定

V_{DD} (電源電圧) V_{LVI}になったあとに、V_{DD} < V_{LVI}を検出した場合、内部リセット (LVIリセット) 信号を発生

CPUクロック周波数を8 MHzに設定

入出力ポートの設定

8ビット・タイマH1の設定

- ・カウント・クロックを $f_{XP}/2^6$ (125 kHz) に、動作モードをPWM出力モードに、TOH1からのタイマ出力を許可に、出力レベル (デフォルト時) をロウ・レベルに設定

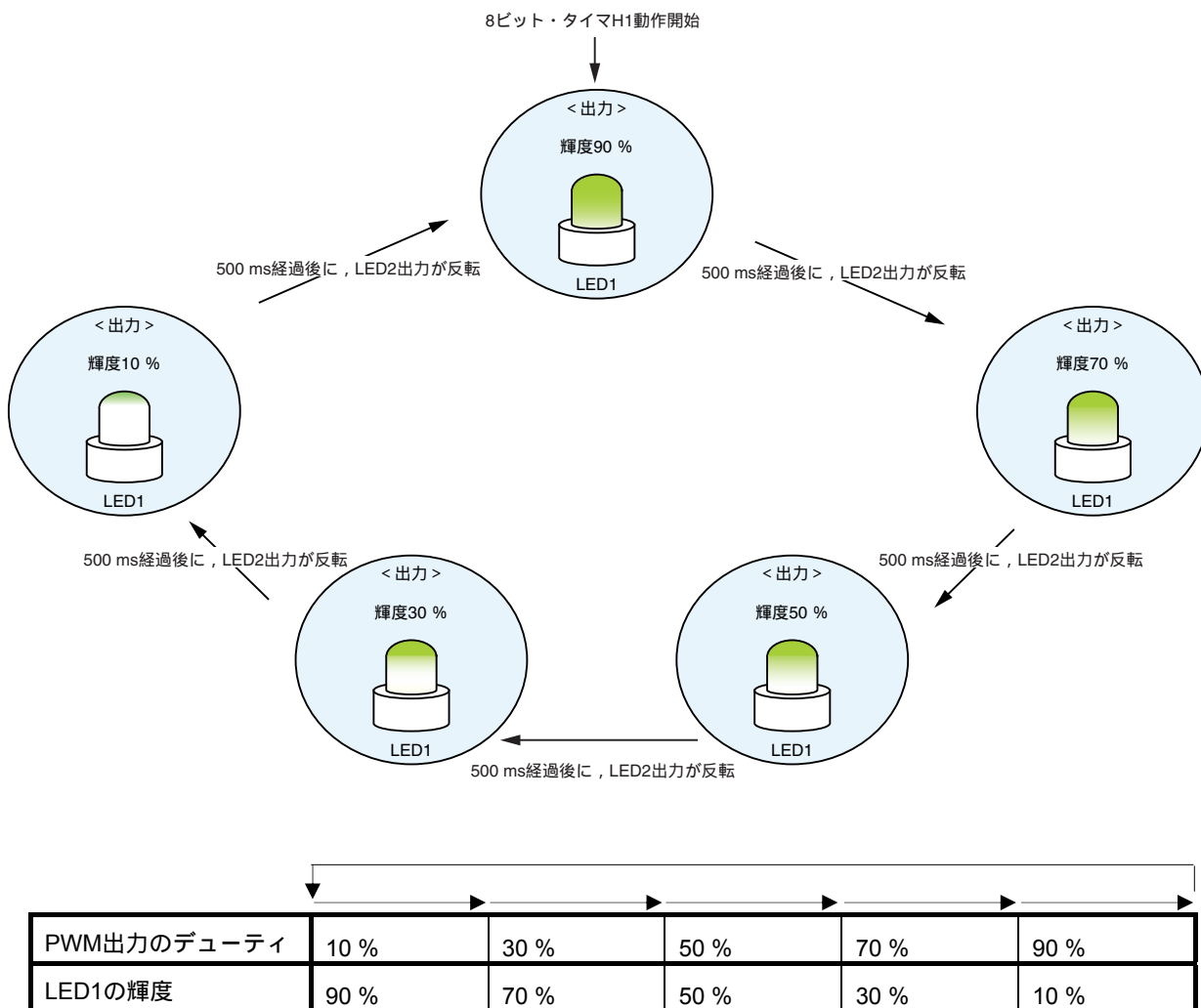
- ・PWMパルス出力周期を2 ms ($8 \mu\text{s} \times 250$) に、デューティを10 %に設定

INTTMH1の割り込みを許可

注 オプション・バイトで設定します。

1.2 メイン・ループ以降の内容

初期設定完了後は、8ビット・タイマH1のPWM出力のデューティを制御して、LED1の輝度を変更します。デューティは、8ビット・タイマH1の割り込み（INTTMH1）を利用して、500 msごとに変更します。変更時には、LED2の出力を反転します。



このサンプル・プログラムでは、PWM出力のアクティブ・レベルがハイ・レベルになるように設定され、LED1がロウ・レベル時に点灯するので、「LED1の輝度 = 100 - デューティ」となります。

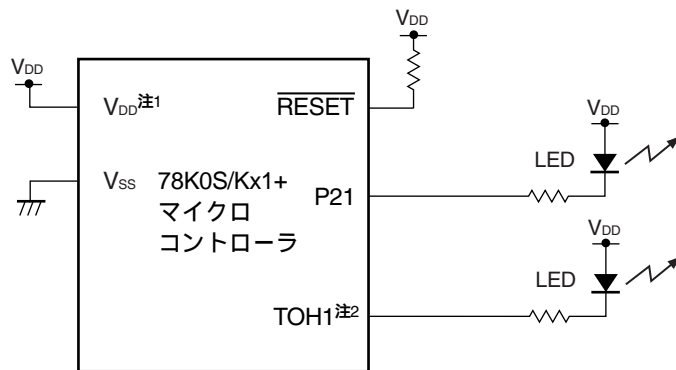
注意 デバイス使用上の注意事項については、各製品のユーザズ・マニュアル ([78K0S/KU1+](#), [78K0S/KY1+](#), [78K0S/KA1+](#), [78K0S/KB1+](#)) を参照してください。

第2章 回路図

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

2.1 回路図

回路図を次に示します。



注1. 4.5 V V_{DD} 5.5 Vの電圧範囲で使用してください。

2. TOH1/P42: 78K0S/KA1+, 78K0S/KB1+マイクロコントローラ
TOH1/ANI0/TI000/P20: 78K0S/KY1+, 78K0S/KU1+マイクロコントローラ

注意1. AVREF端子はVDDに直接接続してください(78K0S/KA1+, 78K0S/KB1+マイクロコントローラのみ)。

2. AVSS端子はGNDに直接接続してください(78K0S/KB1+マイクロコントローラのみ)。

3. 回路図中の端子およびAVREF, AVSS端子以外の未使用端子はすべて出力ポートのため、オープン(未接続)にしてください。

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。




- LED1 : PWM出力
- LED2 : PWM出力のデューティ変更と同時に反転出力(500 msごとに反転)

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1	注1	
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 (システム・クロック・ソースなどを設定)			
tmh1pwm.prw	統合開発環境 PM+用ワーク・スペース・ファイル			
tmh1pwm.prj	統合開発環境 PM+用プロジェクト・ファイル			
tmh1pwm.pri tmh1pwm.prs tmh1pwm.prm	システム・シミュレータ SM+ for 78K0S/Kx1+用プロジェクト・ファイル		注2	
tmh1pwm0.pnl	システム・シミュレータ SM+ for 78K0S/Kx1+用入出力パネル・ファイル (周辺ハードウェア動作を確認するために使用)		注2	
tmh1pwm0.wvo	システム・シミュレータ SM+ for 78K0S/Kx1+用タイミング・チャート・ファイル (波形を確認するために使用)			

注1. 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。

2. 78K0S/KU1+マイクロコントローラには、同封されていません。

備考



: ソース・ファイルのみ同封



: 統合開発環境 PM+とシステム・シミュレータ SM+ for 78K0S/Kx1+で使用するファイルを同封



: システム・シミュレータ SM+ for 78K0S/Kx1+で使用するマイコン動作シミュレーション・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。

- ・ $V_{DD} < V_{LVI}$ 検出 : 低電圧検出 (LVI) 回路
- ・ PWM出力機能 : 8ビット・タイマH1
- ・ PWM出力ポート (LED1) : TOH1^注
- ・ 出力ポート (LED2) : P21

注 TOH1/P42: 78K0S/KA1+, 78K0S/KB1+マイクロコントローラ

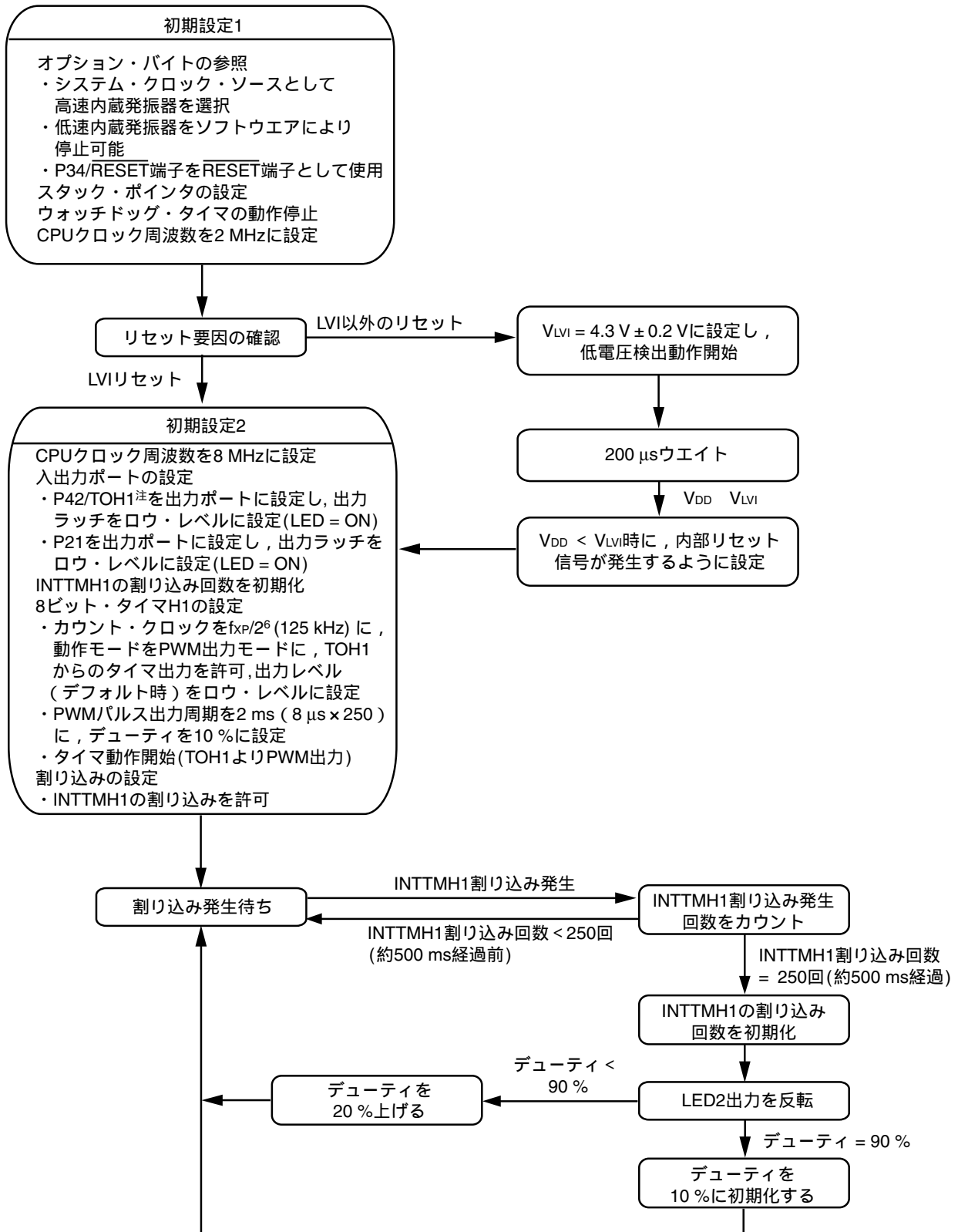
TOH1/ANI0/TI000/P20: 78K0S/KY1+, 78K0S/KU1+マイクロコントローラ

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、低電圧検出機能の設定、クロック周波数の選択、入出力ポートの設定、8ビット・タイマH1 (PWM出力) の設定、割り込みの設定などを行います。

初期設定完了後は、8ビット・タイマH1のPWM出力のデューティを制御して、LED1の輝度を変更します。デューティは、8ビット・タイマH1の割り込み (INTTMH1) を利用して、500 msごとに変更します、変更時には、LED2の出力を反転します。

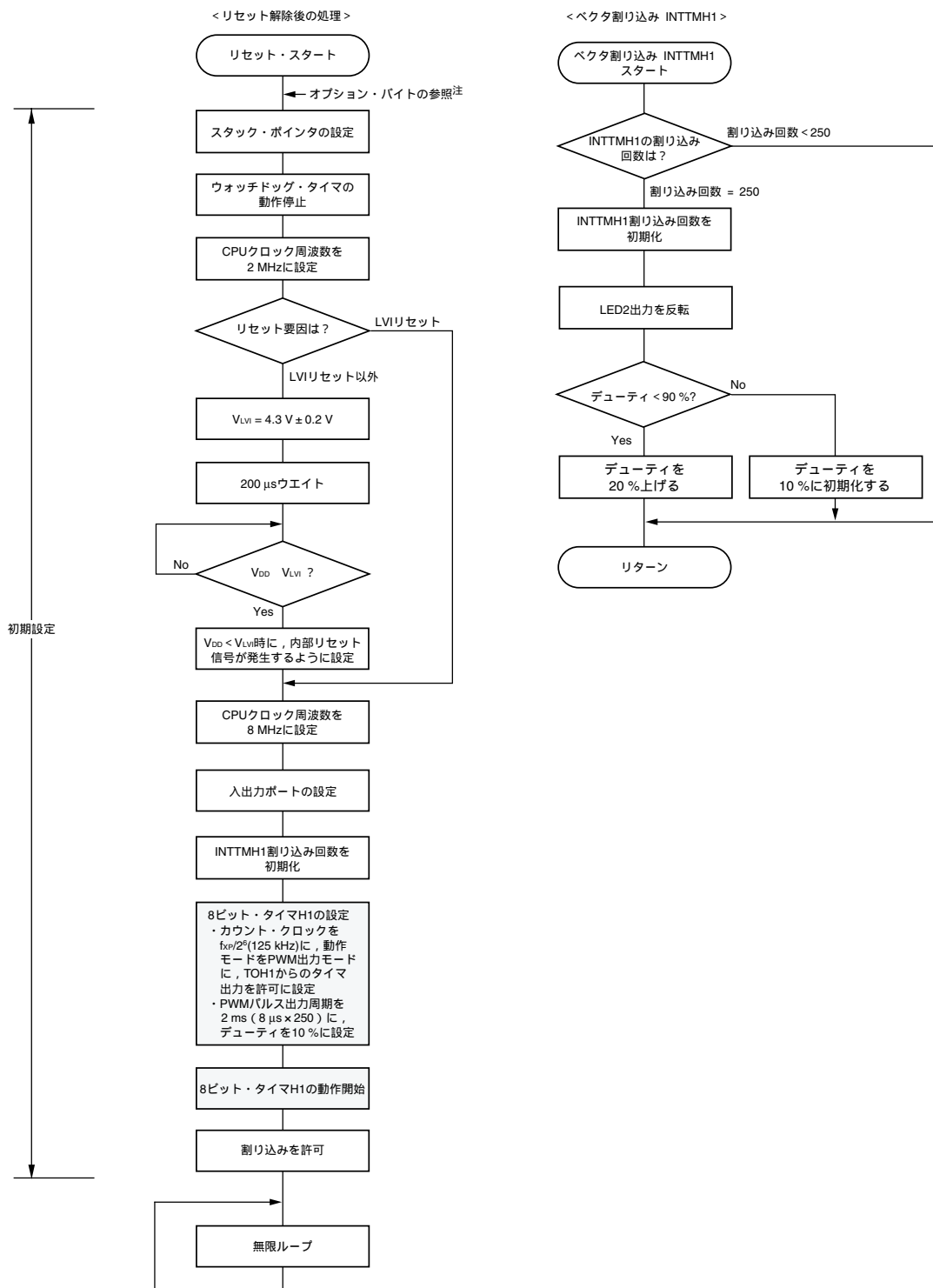
詳細については、次の状態遷移図（ステート・チャート）に示します。



注 TOH1/P42: 78K0S/KA1+, 78K0S/KB1+マイクロコントローラ
TOH1/ANI0/TI000/P20: 78K0S/KY1+, 78K0S/KU1+マイクロコントローラ

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 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・バイトの参照により、次の内容が設定されます。

- ・システム・クロック・ソースとして、高速内蔵発振クロック（8 MHz (TYP.)）を使用
- ・低速内蔵発振器をソフトウェアで停止可
- ・P34/RESET端子をRESET端子として使用

第4章 設定方法について

この章では、8ビット・タイマH1のPWM出力機能について説明します。

その他の初期設定については、[78K0S/Kx1+ サンプル・プログラム \(初期設定\) LED点灯のスイッチ制御編 アプリケーション・ノート](#)を、割り込みについては、[78K0S/Kx1+ サンプル・プログラム \(割り込み\) スイッチ入力による外部割り込み編 アプリケーション・ノート](#)を、低電圧検出 (LVI) については、[78K0S/Kx1+ サンプル・プログラム \(低電圧検出\) 2.7 V未満検出時リセット発生編 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 ([78K0S/KU1+](#), [78K0S/KY1+](#), [78K0S/KA1+](#), [78K0S/KB1+](#)) を参照してください。

アセンブラ命令については、[78K0S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 8ビット・タイマH1のPWM出力機能の設定

8ビット・タイマH1を使用する際に設定するレジスタには、主に次の5種類があります。

- ・8ビット・タイマHモード・レジスタ1 (TMHMD1)
- ・8ビット・タイマHコンペア・レジスタ01 (CMP01)
- ・ポート・モード・レジスタx (PMx) ^注
- ・ポート・レジスタx (Px) ^注
- ・ポート・モード・コントロール・レジスタx (PMCx) ^注

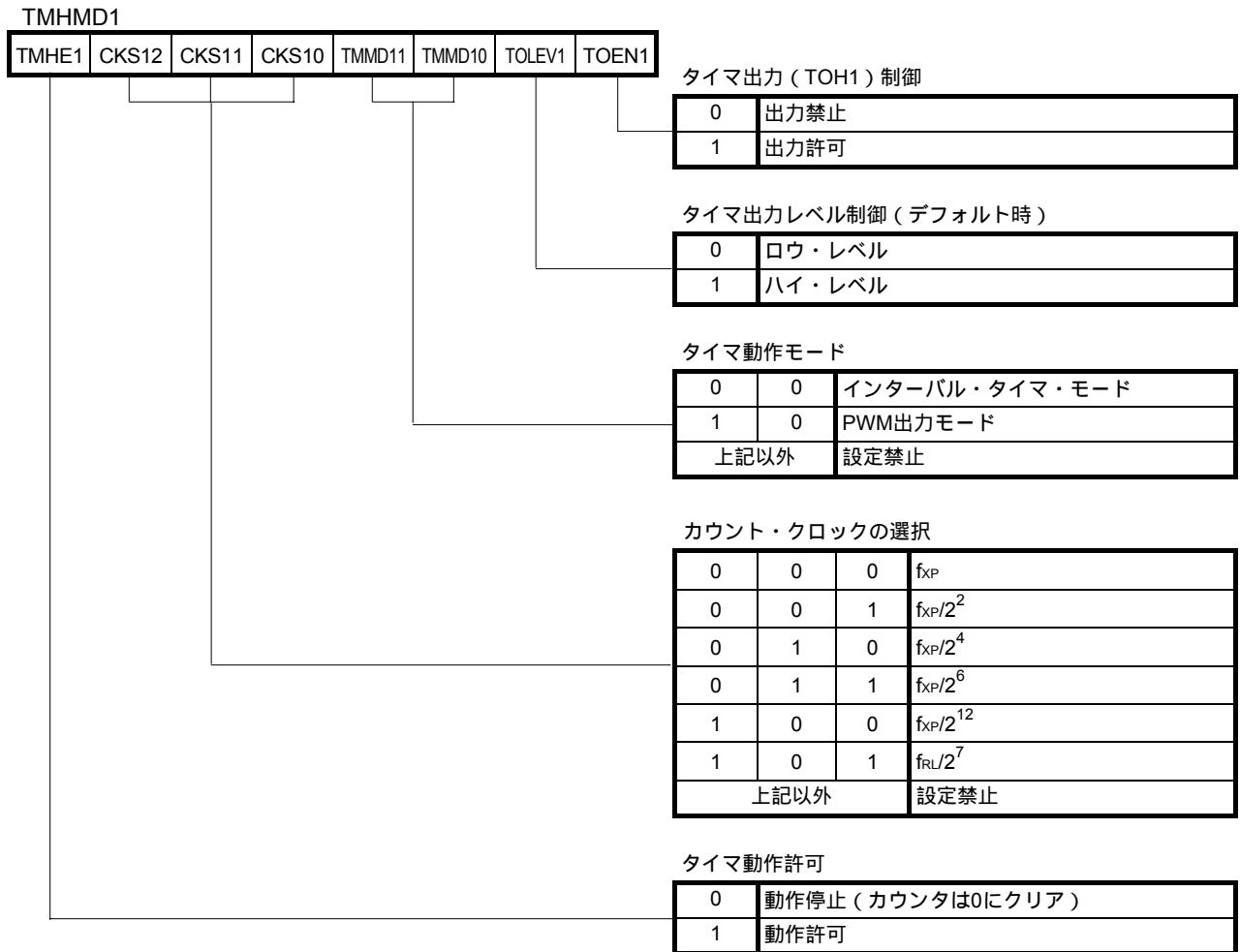
注 8ビット・タイマH1をPWM出力モードとして使用する場合、次のように設定します。

	Pxレジスタ	PMxレジスタ	PMCxレジスタ
78K0S/KA1+, 78K0S/KB1+マイクロコントローラ	P42 = 0	PM42 = 0	設定不要
78K0S/KY1+, 78K0S/KU1+マイクロコントローラ	P20 = 0	PM20 = 0	PMC20 = 0

(1) 8ビット・タイマH1の動作モードについての設定

8ビット・タイマHモード・レジスタ1 (TMHMD1) で、8ビット・タイマH1の動作モードの設定、カウント・クロックの選択、および動作の制御を行います。

図4 - 1 8ビット・タイマHモード・レジスタ1 (TMHMD1) のフォーマット



注意 TMHE1 = 1 のとき, TMHMD1レジスタの他のビットを設定することは禁止です。

備考 f_{XP} : 周辺ハードウェアへのクロックの発振周波数

f_{RL} : 低速内蔵発振クロック周波数

(2) PWMパルス出力周期とデューティの設定

8ビット・タイマHコンペア・レジスタ01 (CMP01) でPWMパルス出力周期を, 8ビット・タイマHコンペア・レジスタ11 (CMP11) でデューティを設定します。

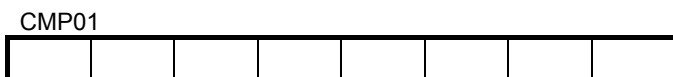
- ・ PWMパルス出力周期 = (N + 1) / f_{CNT}
- ・ デューティ = (M + 1) / (N + 1)

備考 N : CMP01の設定値
 M : CMP11の設定値
 f_{CNT} : 8ビット・タイマH1のカウンタ・クロック周波数

注意 CMP01レジスタの設定値 (N) とCMP11レジスタの設定値 (M) は, 必ず次の範囲内にしてください。

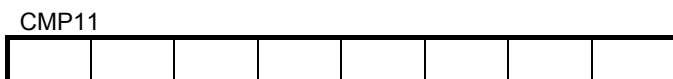
$$00H \leq \text{CMP11 (M)} < \text{CMP01 (N)} \leq FFH$$

図4 - 2 8ビット・タイマHコンペア・レジスタ01 (CMP01) のフォーマット



注意 CMP01レジスタは, タイマ・カウンタ動作中に値を書き換えることは禁止です。

図4 - 3 8ビット・タイマHコンペア・レジスタ11 (CMP11) のフォーマット



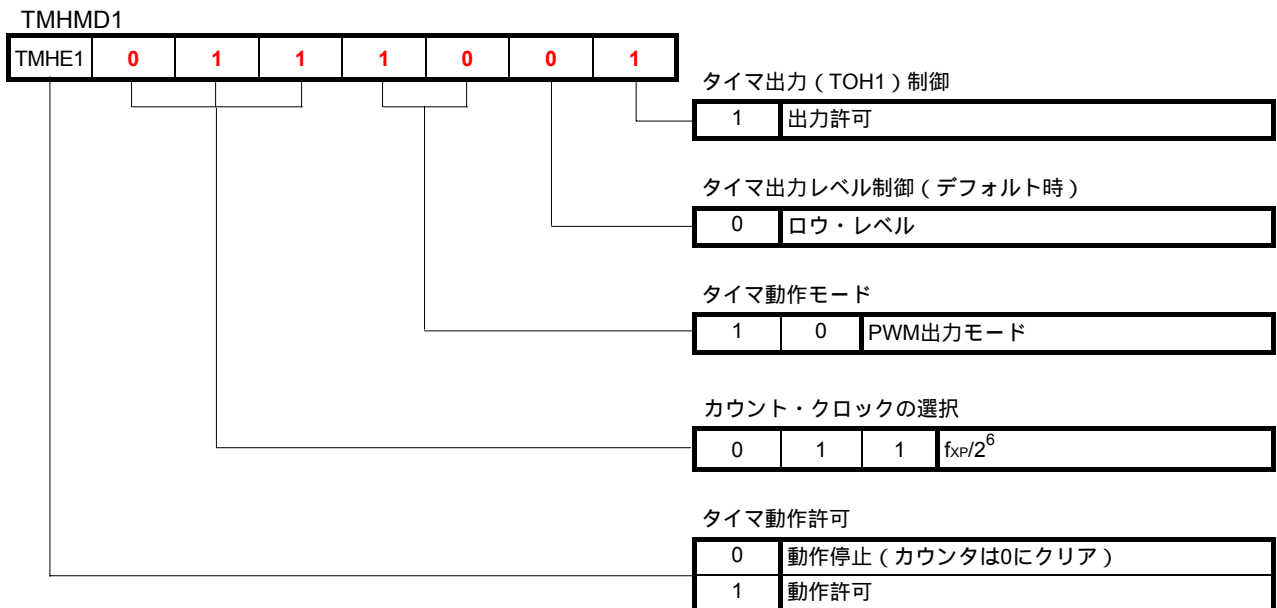
- 注意1.** CMP11レジスタは, タイマ・カウンタ動作中に値の書き換えが可能です。ただしCMP11レジスタの値を変更してからレジスタに値が転送されるまでに, 動作クロック (TMHMD1レジスタのCKS12-CKS10ビットで選択された信号) の3クロック分以上かかります。
2. タイマ・カウンタ動作停止 (TMHE1 = 0) 設定後, タイマ・カウンタ動作を開始する (TMHE1 = 1) 場合, 必ずCMP11レジスタを設定してください (CMP11レジスタへの設定値が同値の場合でも, 必ず再設定してください)。
 3. タイマ動作中にCMP11の値を書き換えた場合, 書き換え後のコンペア値は, カウンタ値と書き換え前のコンペア値が一致したタイミングで有効となります。カウンタ値とコンペア値が一致するタイミングと, CPUからCMP11への書き込みが競合した場合, 書き込み後のコンペア値が有効となるのは, 次のカウンタ値と書き込み前のコンペア値が一致したタイミングとなります。

(3) TOH1端子の設定

8ビット・タイマH1をPWM出力モードとして使用する場合, ポート・レジスタx (Px), ポート・モード・レジスタx (PMx), ポート・モード・コントロール・レジスタx (PMCx) を次のように設定します。

	Pxレジスタ	PMxレジスタ	PMCxレジスタ
78K0S/KA1+, 78K0S/KB1+マイクロコントローラ	P42 = 0	PM42 = 0	設定不要
78K0S/KY1+, 78K0S/KU1+マイクロコントローラ	P20 = 0	PM20 = 0	PMC20 = 0

- 【例 1】・8ビット・タイマH1の動作モードをPWM出力モードに，カウント・クロックを $f_{XP}/2^6$ ($f_{XP} = 8 \text{ MHz}$) に，タイマ出力 (TOH1) を許可に，出力レベル (デフォルト時) をロウ・レベルに設定
 ・PWMパルス出力周期を2 msに，デューティを10 %に設定し，タイマ動作開始 (サンプル・プログラムと同内容)



CMP01の設定値 (N) : 249

- ・カウント・クロック $f_{CNT} = 8 \text{ MHz} / 2^6 = 0.125 \text{ MHz} = 125 \text{ kHz}$
- ・PWMパルス出力周期 $2 \text{ ms} = (N + 1) / 125 \text{ kHz}$
 $N = 2 \text{ ms} \times 125 \text{ kHz} - 1 = 249$

CMP11の設定値 (M) : 24

- ・0.1 (= デューティ 10%) = $(M + 1) / (249 + 1)$
 $M = 0.1 \times 250 - 1 = 24$

TOH1端子の設定

- ・78K0S/KA1+, 78K0S/KB1+マイクロコントローラ : P42 = 0, PM42 = 0
- ・78K0S/KY1+, 78K0S/KU1+マイクロコントローラ : P20 = 0, PM20 = 0, PMC20 = 0

78K0S/KA1+, 78K0S/KB1+マイクロコントローラの場合，P42に「0」，PM42に「0」，TMHMD1に「00111001」，CMP01に「249」，CMP11に「24」を設定したあとに，TMHE1に1を設定して，タイマ動作を開始します。

78K0S/KY1+, 78K0S/KU1+マイクロコントローラの場合，P20に「0」，PM20に「0」，PMC20に「0」，TMHMD1に「00111001」，CMP01に「249」，CMP11に「24」を設定したあとに，TMHE1に1を設定して，タイマ動作を開始します。

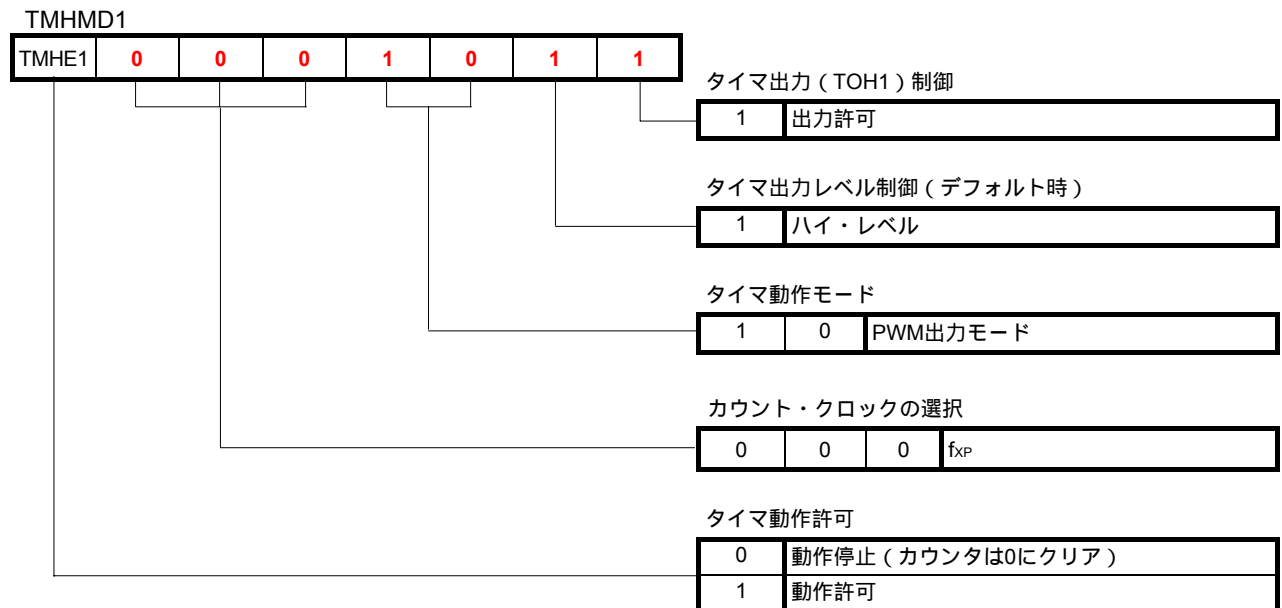
• アセンブリ言語の場合 (78K0S/KA1+, 78K0S/KB1+マイクロコントローラ使用時)

```
CLR1   P4.2
CLR1   PM4.2
MOV    TMHMD1, #00111001B
MOV    CMP01,  #249
MOV    CMP11,  #24
SET1   TMHE1
```

• C言語の場合 (78K0S/KA1+, 78K0S/KB1+マイクロコントローラ使用時)

```
P4.2 = 0;
PM4.2 = 0;
TMHMD1 = 0b00111001;
CMP01 = 249;
CMP11 = 24;
TMHE1 = 1;
```


- 【例 2】・8ビット・タイマH1の動作モードをPWM出力モードに，カウント・クロックを f_{XP} ($f_{XP} = 8 \text{ MHz}$) に設定，タイマ出力 (TOH1) を許可に設定，タイマ出力レベル (デフォルト時) をハイ・レベルに設定
 ・PWMパルス出力周期を $31.25 \mu\text{s}$ に設定し，デューティを50 %に設定し，タイマ動作開始



CMP01の設定値 (N) : 249

- ・カウント・クロック $f_{CNT} = 8 \text{ MHz}$
- ・PWMパルス出力周期 $31.25 \mu\text{s} = (N + 1) / 8 \text{ MHz}$
 $N = 31.25 \mu\text{s} \times 8 \text{ MHz} - 1 = 249$

CMP11の設定値 (M) : 124

- ・0.5 (= デューティ 50%) = $(M + 1) / (249 + 1)$
 $M = 0.5 \times 250 - 1 = 124$

TOH1端子の設定

- ・78K0S/KA1+, 78K0S/KB1+マイクロコントローラ : P42 = 0, PM42 = 0
- ・78K0S/KY1+, 78K0S/KU1+マイクロコントローラ : P20 = 0, PM20 = 0, PMC20 = 0

78K0S/KA1+, 78K0S/KB1+マイクロコントローラの場合，P42に「0」，PM42に「0」，TMHMD1に「00001011」，CMP01に「249」，CMP11に「124」を設定したあとに，TMHE1に1を設定して，タイマ動作を開始します。

78K0S/KY1+, 78K0S/KU1+マイクロコントローラの場合，P20に「0」，PM20に「0」，PMC20に「0」，TMHMD1に「00001011」，CMP01に「249」，CMP11に「124」を設定したあとに，TMHE1に1を設定して，タイマ動作を開始します。

• アセンブリ言語の場合 (78K0S/KA1+, 78K0S/KB1+マイクロコントローラ使用時)

```
CLR1   P4.2
CLR1   PM4.2
MOV    TMHMD1, #00001011B
MOV    CMP01,  #249
MOV    CMP11,  #124
SET1   TMHE1
```

• C言語の場合 (78K0S/KA1+, 78K0S/KB1+マイクロコントローラ使用時)

```
P4.2 = 0;
PM4.2 = 0;
TMHMD1 = 0b00001011;
CMP01 = 249;
CMP11 = 124;
TMHE1 = 1;
```


C言語のプログラム例 (前述の【例 1】とサンプル・プログラムと同内容)

```

void hdwinit(void) {
    unsigned char ucCnt200us; /* 200usウェイト用8ビット変数 */
    .
    .
    .
    P4  = 0b00000000; /* P40-P47の出力ラッチLow (P42:LED1点灯) */
    PM4 = 0b00001000; /* P40-P47を出力ポートに設定 */
    .
    .
    .

    TMHMD1 = 0b00111001; /* カウント・クロック = fxp/26 = 125kHz、PWMモード、*/
    /* TOH1出力許可 */
    CMP01 = 250-1; /* CMP01を初期化(周期:2ms) */
    CMP11 = 25-1; /* CMP11を初期化(デューティ:10%) */
    TMHE1 = 1; /* タイマ動作開始 */

    IF0 = 0x00; /* 無効割り込み要求をクリアしておく */
    TMMKH1 = 0; /* INTTMH1割り込みマスク解除 */

    return;
}

void main(void) {
    EI(); /* ベクタ割り込み許可 */

    while (1) {
        NOP();
        NOP();
    }
}

interrupt void fn inttmH1() {
    .
    .
    .
    if (CMP11 >= 225-1) { /* デューティが90%以上の場合の処理 */
        CMP11 = 25-1; /* デューティを10%に初期化 */
    }
    else {
        CMP11 += 50; /* デューティを20%上げる */
    }
    .
    .
    .
}
    
```

PWMパルス出力
周期を設定

P42の出力ラッチと
PM42を0に設定

8ビット・タイマH1の動作モード、
カウント・クロック、TOH1出力、
出力レベル(デフォルト時)を設定

タイマ動作開始

デューティを設定

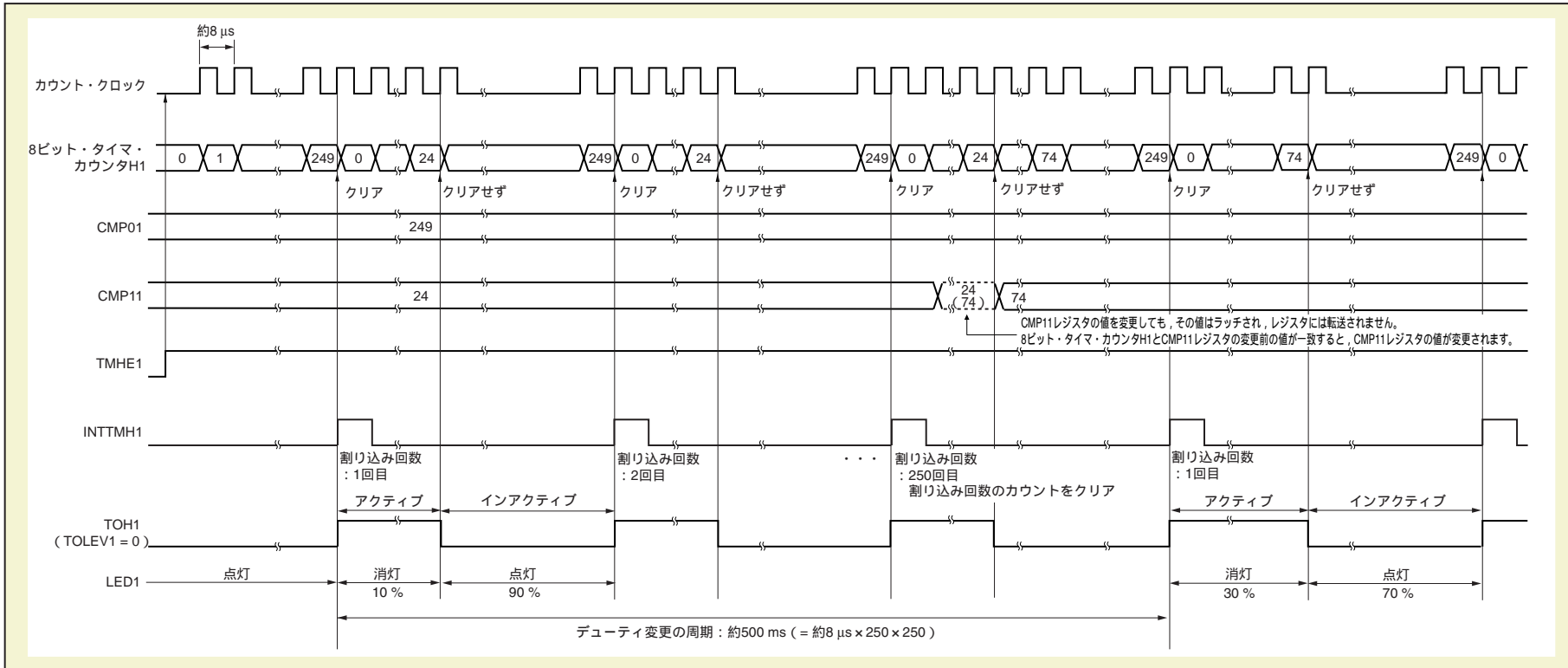
INTTMH1
割り込み要求
フラグをクリア

INTTMH1
割り込み処理
を許可


INTTMH1割り込み発生に
より、割り込み処理開始

8ビット・タイマH1
動作中にデューティを
変更可能

図4 - 4 PWM出力のデューティを10% 30% (LED1の輝度を90% 70%) に変更する場合のタイミング・チャート例




第5章 システム・シミュレータ SM+での動作確認

この章では、のアイコンを選択してダウンロードしたアセンブリ言語用のファイル（ソース・ファイル+プロジェクト・ファイル）を用い、サンプル・プログラムが、システム・シミュレータ SM+ for 78K0S/Kx1+でどのように動作するかを説明します。

注意 SM+ for 78K0S/Kx1+は、78K0S/KU1+マイクロコントローラには対応していません（2008年7月現在）。
したがって、78K0S/KU1+マイクロコントローラはSM+ for 78K0S/Kx1+で動作確認することはできません。

5.1 サンプル・プログラムのビルド

サンプル・プログラムをSM+ for 78K0S/Kx1+（以降、「SM+」と表記します）で動作確認をするために、サンプル・プログラムをビルドしてから、SM+を起動する必要があります。ここでは、サンプル・プログラムのビルド方法について、のアイコンからダウンロードしたアセンブリ言語用サンプル・プログラム（ソース・プログラム+プロジェクト・ファイル）を使用し、説明します。その他のダウンロードしたプログラムのビルド方法については、[78K0S/Kx1+ サンプル・プログラム スタートアップ・ガイド アプリケーション・ノート](#)を参照してください。

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。

【コラム】ビルドのエラー


PM+でビルドしているときに「A006 File not found 'C:¥NECTOOLS32¥LIB78K0S¥s0sl.rel'」または、「*** ERROR F206 Segment '@@DATA' can't allocate to memory - ignored.」というエラー・メッセージが出た場合、次の手順にてコンパイラオプションの設定を変更してください。

[ツール] [コンパイラオプションの設定]を選択してください。

[コンパイラオプションの設定]ダイアログが開いたら、「スタートアップ・ルーチン」タグを選択してください。

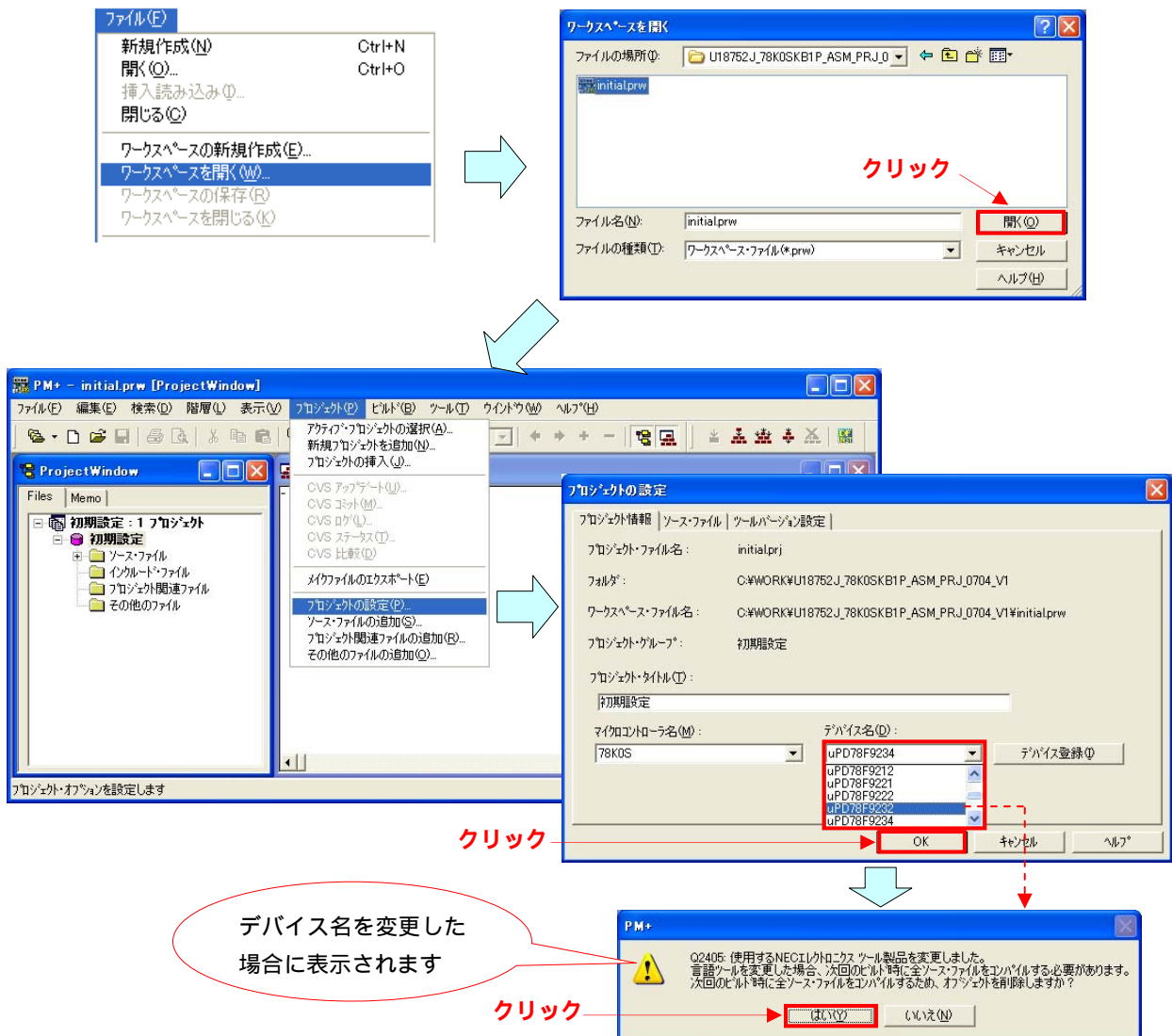
「標準ライブラリ固定領域を使用する」のチェックを外してください（それ以外のチェックは、そのまま）。


「標準ライブラリ固定領域を使用する」のチェックを外すと、標準ライブラリ固定領域として確保されていた118バイトのRAM領域が使用可能になりますが、標準ライブラリ（getchar関数やmalloc関数など）を使用できなくなります。

このサンプル・プログラムでは、のアイコンを選択してダウンロードしたファイルを使用する場合、デフォルトで「標準ライブラリ固定領域を使用する」のチェックが外されています。

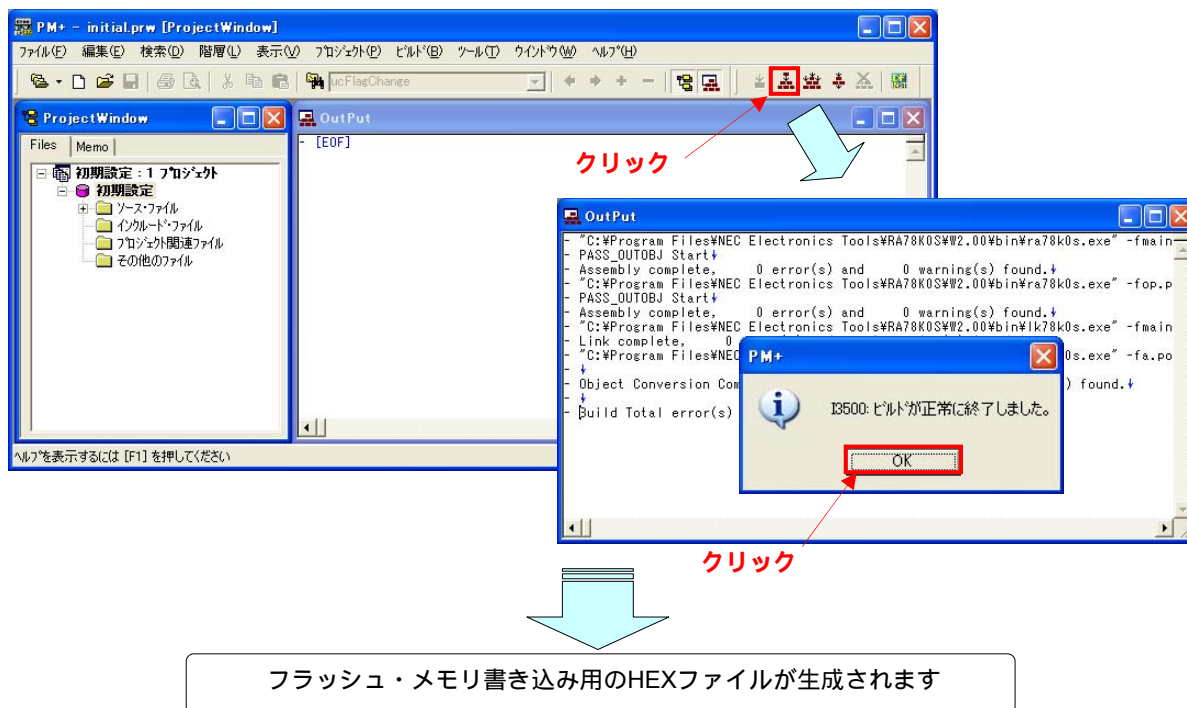
- (1) PM+を起動してください。
- (2) [ファイル] [ワークスペースを開く] から、「tmh1pwm.prw」を選択し、[開く] ボタンをクリックしてください。ワークスペースが作成され、その中にソース・ファイルが自動的に読み込まれます。
- (3) [プロジェクト] [プロジェクトの設定] を選択してください。[プロジェクトの設定] 画面が立ち上がったら、使用するデバイス名を選択(デフォルトでは、ROM/RAMサイズの最も大きいデバイスが選択)し、[OK] ボタンをクリックしてください。

備考 下の図は、「サンプル・プログラム(初期設定) LED点灯のスイッチ制御」の画面例です。



- (4)  (「ビルド」ボタン)をクリックしてください。ソース・ファイルが正常にビルドされると、「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立ち上がります。
- (5) メッセージ画面にある [OK] ボタンをクリックしてください。フラッシュ・メモリ書き込み用のHEXファイルが作成されます。

備考 下の図は、「サンプル・プログラム(初期設定) LED点灯のスイッチ制御」の画面例です。




5.2 SM+での動作

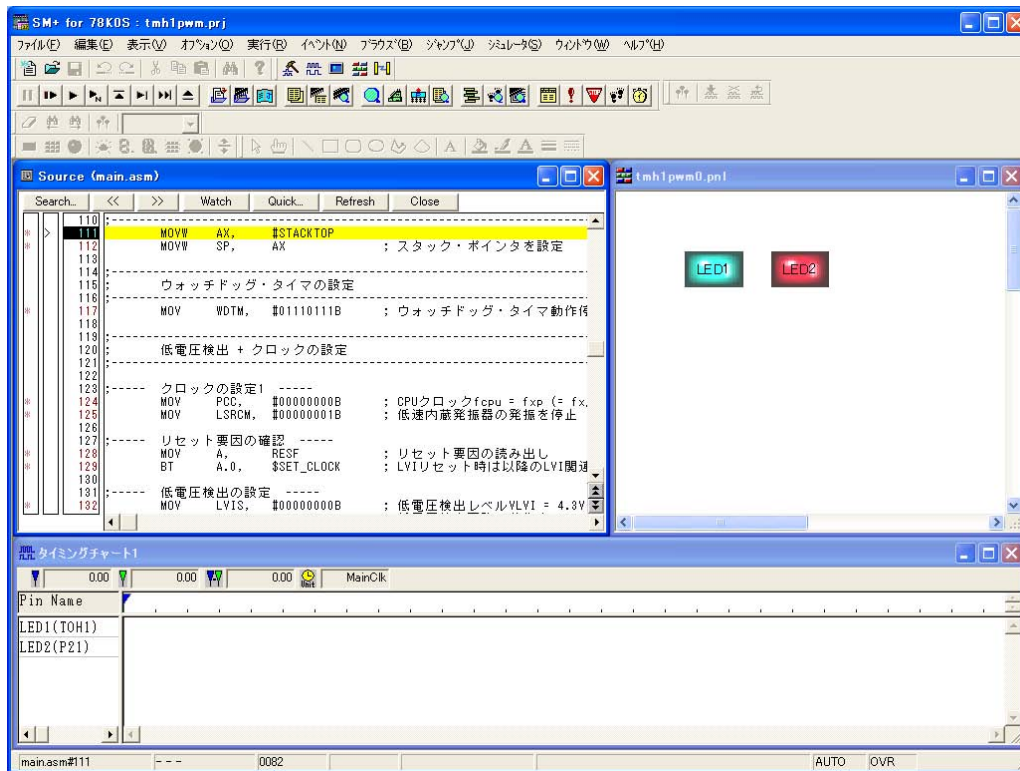
ここでは、SM+の入出力パネル・ウィンドウやタイミング・チャート・ウィンドウ上での動作確認の例を説明します。


SM+操作方法の詳細については、[SM+ システム・シミュレータ 操作編 ユーザーズ・マニュアル](#)を参照してください。

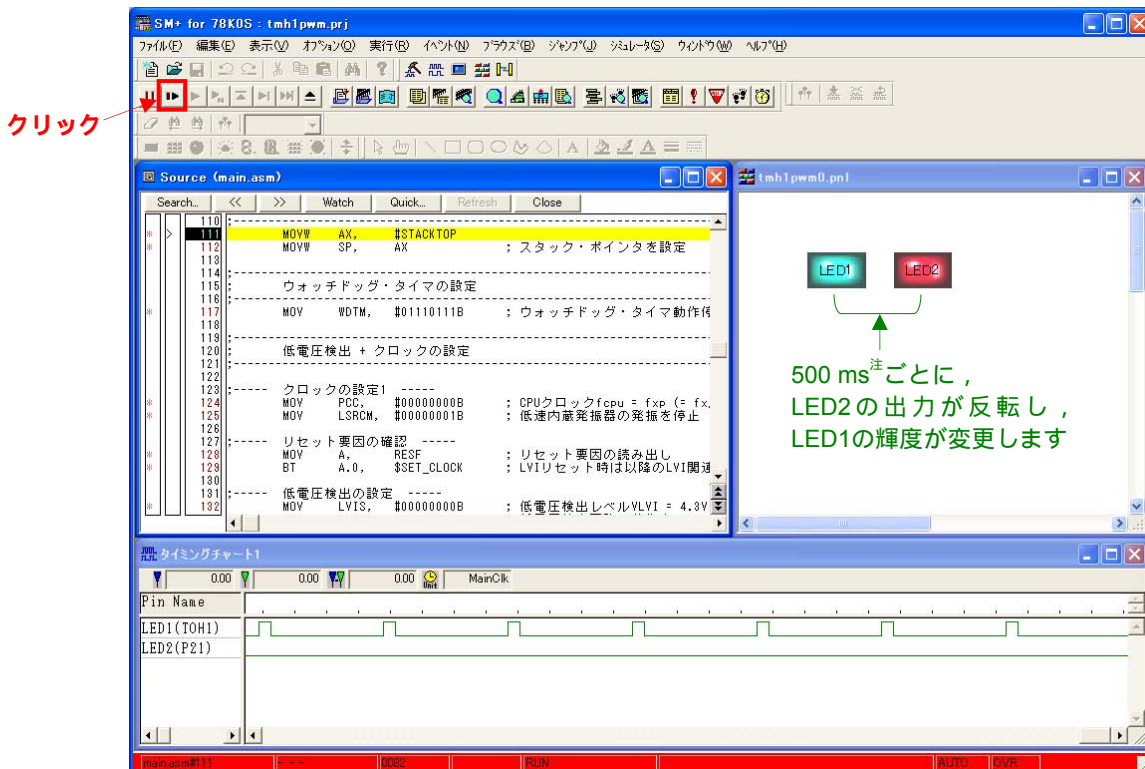
- (1) SM+ for 78K0S/Kx1+ W1.02をPM+ Ver6.30の環境で使用するために、次のサイトにあるPDFファイルを参照して、「外部ツールの登録」を行い、SM+を起動してください。

・ <http://www.necel.com/micro/ja/freesoft/pdf/ZUD-CD-07-0189.pdf>

- (2) SM+を起動すると、次のような画面になります（これは、のアイコンを選択してダウンロードしたアセンブリ言語のソース・ファイルを使用した場合の画面例です）。



- (3) （「リスタート」ボタン）をクリックしてください。CPUリセット後、プログラムが実行され、次のような画面になります。



プログラム実行中は、赤になります。

注 ご使用のPCの動作環境によっては、実際の周期と異なる場合があります。

(4) プログラム実行中に、PWM出力のデューティ（[LED1]の輝度）が変化し、そのたびに[LED2]の出力が反転することを、タイミング・チャート・ウィンドウの波形で確認してください。

PWM出力のデューティ ([LED1] の輝度)	タイミング・チャート・ウィンドウ
デューティ : 10 % (輝度 : 90 %) デューティ : 30 % (輝度 : 70 %)	デューティ10%のPWM出力開始からデューティ30%のPWM出力開始までの時間 デューティ10%の波形 デューティ30%の波形 
デューティ : 30 % (輝度 : 70 %) デューティ : 50 % (輝度 : 50 %)	デューティ30%のPWM出力開始からデューティ50%のPWM出力開始までの時間 デューティ30%の波形 デューティ50%の波形 
・ ・ ・	・ ・ ・
デューティ : 90 % (輝度 : 10 %) デューティ : 10 % (輝度 : 90 %)	デューティ90%のPWM出力開始からデューティ10%のPWM出力開始までの時間 デューティ90%の波形 デューティ10%の波形 

注

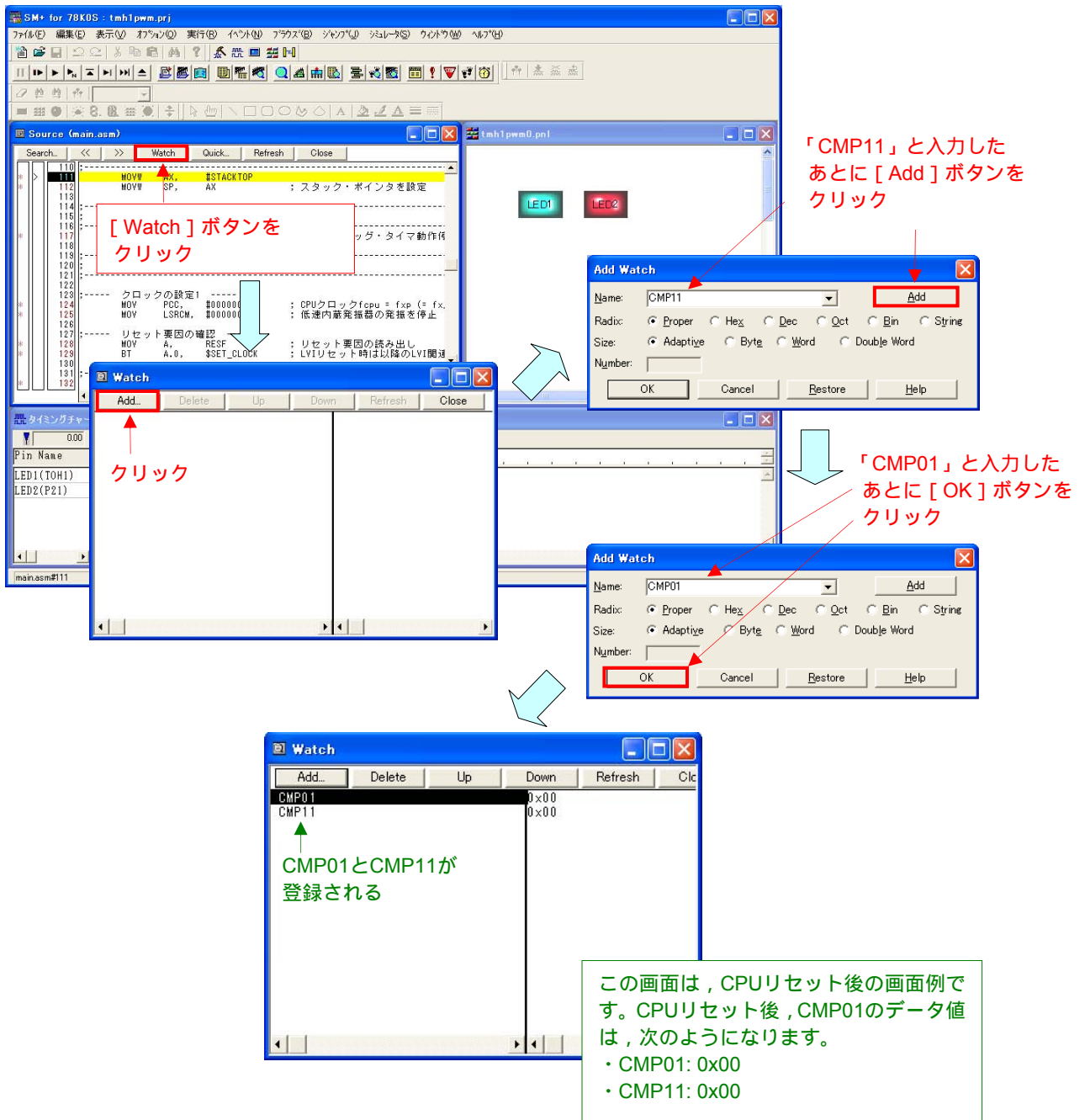
注 PWM出力のデューティ90%以降は、デューティ10%からの繰り返しになります。

【補足】 SM+のウォッチ機能を使用することにより ,CMP01レジスタとCMP11レジスタのデータ値の変化を確認することができます。

ソース・ウインドウの [Watch] ボタンをクリックしてください。 [Watch] ウィンドウが立ち上がります。

[Add]ボタンをクリックすると , [Add Watch] ウィンドウが立ち上がります(このとき , [Watch] ウィンドウは開いたままです)。

Nameに「CMP01」と「CMP11」を入力し , [OK] ボタンをクリックすると , [Watch] ウィンドウに , 「CMP01」と「CMP11」が登録され , [Add Watch] ウィンドウが閉じられます。



プログラムを実行し、[Watch] ウィンドウ上のCMP01とCMP11のデータ値が変化することを確認してください。

PWM出力のデューティ ([LED1] の輝度) ^注	[Watch] ウィンドウのデータ値
デューティ : 10% (輝度 : 90 %)	CMP01: 0xF9 (249) , CMP11: 0x18 (24)
デューティ : 30% (輝度 : 70 %)	CMP01: 0xF9 (249) , CMP11: 0x4A (74)
デューティ : 50% (輝度 : 50 %)	CMP01: 0xF9 (249) , CMP11: 0x7C (124)
デューティ : 70% (輝度 : 30 %)	CMP01: 0xF9 (249) , CMP11: 0xAE (174)
デューティ : 90% (輝度 : 10 %)	CMP01: 0xF9 (249) , CMP11: 0xE0 (224)

注 PWM出力のデューティ90%以降は、デューティ10%からの繰り返しになります。

第6章 関連資料

資料名		和文 / 英文
78K0S/KU1+ ユーザーズ・マニュアル		PDF
78K0S/KY1+ ユーザーズ・マニュアル		PDF
78K0S/KA1+ ユーザーズ・マニュアル		PDF
78K0S/KB1+ ユーザーズ・マニュアル		PDF
78K0Sシリーズ 命令編 ユーザーズ・マニュアル		PDF
RA78K0S 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
CC78K0S 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
フラッシュ書き込み簡単マニュアル (MINICUBE2編) インフォメーション	78K0S/KU1+	PDF
	78K0S/KY1+	PDF
	78K0S/KA1+	PDF
	78K0S/KB1+	PDF
78K0S/Kx1+ アプリケーション・ ノート	サンプル・プログラム スタートアップ・ガイド	PDF
	サンプル・プログラム (初期設定) LED点灯のスイッチ制御編	PDF
	サンプル・プログラム (割り込み) スイッチ入力による外部割り込み編	PDF
	サンプル・プログラム (低電圧検出) 2.7V未満検出時リセット発生編	PDF
	サンプル・プログラム (8ビット・タイマH1) インターバル・タイマ編	PDF

付録A プログラム・リスト

プログラム・リスト例として、78K0S/KB1+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      78K0S/KB1+シリーズ
;
;*****
; 78K0S/KB1+シリーズ      サンプル・プログラム
;*****
; 8ビット・タイマH1 (PWM出力)
;*****
; 【履歴】
; 2007.7.--      新規作成
;*****
;
; 【概要】
;
; 本サンプルプログラムは、8ビット・タイマH1のPWM出力機能の使用例を示すものである。
; 8ビット・タイマH1のタイマ出力(T0H1端子)にLED1を接続し、PWM出力のデューティにより
; LED1の輝度を制御する。デューティの変更はタイマH1の割り込みを利用して500ms周期で
; 行い、同時にLED2の出力を反転する。
;
;
; <主な設定内容>
;
; ・ウォッチドッグ・タイマの動作停止
; ・低電圧検出電圧VLVIを4.3V ± 0.2Vに設定
; ・VDD VLVIとなった後にVDD < VLVIとなった場合、内部リセット信号発生(低電圧検出回路)
; ・CPUクロックを8MHzに設定
; ・周辺ハードウェアへの供給クロックを8MHzに設定
;
;
; <8ビット・タイマH1の設定>
; ・PWMモードに設定
; ・T0H1端子のタイマ出力許可
; ・カウント・クロック = fxp/26 (125kHz)
; ・タイマ周期 = 2ms (8[us/clk] × 250[count] = 2[ms])
```

```

;
;
; < PWM出力のデューティとLED >
;
; ・500ms毎に以下の順でデューティが変わると同時に、LED2出力が反転する。
; +-----+-----+
; | PWM出力のデューティ | 10% | 30% | 50% | 70% | 90% | (以降10%から繰り返し)
; +-----+-----+
; | LED1の輝度          | 90% | 70% | 50% | 30% | 10% |
; +-----+-----+
; # PWM出力がHighアクティブであり、LED1がLowアクティブであることから、
;   LEDの輝度 = 100 - デューティ比となる。
;
;
; 【ポート入出力の設定】
;
; 入力ポート：-
; 出力ポート：P00-P03, P20-P23, P30-P33, P40-P47, P120-P123, P130
; 未使用のポート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
;
;=====
XVCT  CSEG  AT      0000H
      DW   RESET_START      ;(00)  RESET
      DW   RESET_START      ;(02)  --
      DW   RESET_START      ;(04)  --
      DW   RESET_START      ;(06)  INTLV1
      DW   RESET_START      ;(08)  INTP0
      DW   RESET_START      ;(0A)  INTP1
      DW   INTERRUPT_TMH1    ;(0C)  INTTMH1
      DW   RESET_START      ;(0E)  INTTM000
      DW   RESET_START      ;(10)  INTTM010
      DW   RESET_START      ;(12)  INTAD
      DW   RESET_START      ;(14)  --
      DW   RESET_START      ;(16)  INTP2
      DW   RESET_START      ;(18)  INTP3
      DW   RESET_START      ;(1A)  INTTM80

```

```

DW    RESET_START    ;(1C)  INTSRE6
DW    RESET_START    ;(1E)  INTSR6
DW    RESET_START    ;(20)  INTST6

;=====
;
;   RAMの定義
;
;=====
XRAM  DSEG    SADDR
CNT_TMH1:  DS      1          ; INTTMH1割り込みのカウント用

;=====
;
;   スタック領域の確保
;
;=====
XSTK  DSEG    AT      0FEE0H
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 = FF00H

;*****
;
;   リセット解除後の初期化処理
;
;*****
XMAIN CSEG    UNIT
RESET_START:
;-----
;   スタック・ポインタの設定
;-----
      MOVW   AX,    #STACKTOP
      MOVW   SP,    AX          ; スタック・ポインタを設定

;-----
;   ウォッチドッグ・タイマの設定
;-----
      MOV    WDTM,  #01110111B  ; ウォッチドッグ・タイマ動作停止

;-----
;   低電圧検出 + クロックの設定
;-----

```



```

;----- クロックの設定1 -----
MOV    PCC,    #00000000B    ; CPUクロックfcpu = fxp (= fx/4 = 2MHz)
MOV    LSRCM,  #00000001B    ; 低速内蔵発振器の発振を停止

;----- リセット要因の確認 -----
MOV    A,      RESF          ; リセット要因の読み出し
BT     A.0,    $SET_CLOCK    ; LVIリセット時は以降のLVI関連処理を省略し、SET_CLOCKへ

;----- 低電圧検出の設定 -----
MOV    LVIS,   #00000000B    ; 低電圧検出レベルVLVI = 4.3V±0.2Vに設定
SET1   LVION   ; 低電圧検出回路の動作許可

MOV    A,      #40          ; 200usウェイト用のカウント値を代入
;----- 200usウェイト -----
WAIT_200US:
DEC    A
BNZ   $WAIT_200US          ; 0.5[us/clock]×10[clock]×40[count] = 200[us]

;----- VDD VLVI待ち処理 -----
WAIT_LVI:
NOP
BT     LVIF,   $WAIT_LVI    ; VDD < VLVIなら分岐

SET1   LVIMD          ; VDD < VLVI時に内部リセット信号が発生するように設定

;----- クロックの設定2 -----
SET_CLOCK:
MOV    PPCC,   #00000000B    ; 周辺ハードウェアへの供給クロックfxp = fx (= 8MHz)
; -> CPUクロックfcpu = fxp = 8MHz

;-----
;   ポート0の設定
;-----
MOV    P0,     #00000000B    ; P00-P03の出力ラッチLow
MOV    PM0,    #11110000B    ; P00-P03を出力ポートに設定

;-----
;   ポート2の設定
;-----
MOV    P2,     #00000000B    ; P20-P23の出力ラッチLow(P21:LED2点灯)
MOV    PM2,    #11110000B    ; P20-P23を出力ポートに設定

```

```

;-----
;   ポート3の設定
;-----
MOV   P3,    #00000000B    ; P30-P33の出力ラッチLow
MOV   PM3,   #11110000B    ; P30-P33を出力ポートに設定

;-----
;   ポート4の設定
;-----
MOV   P4,    #00000000B    ; P40-P47の出力ラッチLow(P42:LED1点灯)
MOV   PM4,   #00000000B    ; P40-P47を出力ポートに設定

;-----
;   ポート12の設定
;-----
MOV   P12,   #00000000B    ; P120-P123の出力ラッチLow
MOV   PM12,  #11110000B    ; P120-P123を出力ポートに設定

;-----
;   ポート13の設定
;-----
MOV   P13,   #00000001B    ; P130の出力High

;-----
;   RAMの初期化
;-----
MOV   CNT_TMH1, #250        ; INTTMH1割り込み回数を初期化

;-----
;   8ビット・タイマH1の設定
;-----
MOV   TMHMD1, #00111001B   ; カウント・クロック =  $f_{xp}/2^6 = 125\text{kHz}$ 、PMMモード、TOH1出力許可
MOV   CMP01,  #250-1       ; CMP01を初期化(周期:2ms)
MOV   CMP11,  #25-1        ; CMP11を初期化(デューティ:10%)
SET1  TMHE1              ; タイマ動作開始

;-----
;   割り込みの設定
;-----
MOV   IF0,    #00H         ; 無効割り込み要求をクリアしておく
CLR1  TMMKH1              ; INTTMH1割り込みマスク解除

EI                                       ; ベクタ割り込み許可

```

```
.;
.;
.;   メイン・ループ
.;
.;
.;*****
MAIN_LOOP:
    NOP
    BR    $MAIN_LOOP        ; MAIN_LOOPへ

.;*****
.;
.;   割り込みINTTMH1
.;
.;*****
INTERRUPT_TM1:
    DBNZ  CNT_TM1, $END_INTTMH1 ; INTTMH1割り込み回数 < 250なら分岐

    MOV   CNT_TM1, #250        ; 割り込み回数を初期化
    XOR   P2,    #00000010B    ; LED2反転

    CMP   CMP11, #225-1       ; デューティを90%と比較
    BC   $INC_CMP11           ; デューティ < 90%なら分岐

    MOV   CMP11, #25-1        ; デューティを10%に初期化
    BR   $END_INTTMH1        ; END_INTTMH1へ分岐

INC_CMP11:
    ADD   CMP11, #50          ; デューティを20%上げる

END_INTTMH1:
    RETI                      ; 割り込み処理から復帰

end
```

main.c (C言語版)

/******

NEC Electronics 78K0S/KB1+シリーズ

78K0S/KB1+シリーズ サンプル・プログラム

8ビット・タイマH1 (PWM出力)

【履歴】

2007.7.-- 新規作成

【概要】

本サンプルプログラムは、8ビット・タイマH1のPWM出力機能の使用例を示すものである。8ビット・タイマH1のタイマ出力(TOH1端子)にLED1を接続し、PWM出力のデューティによりLED1の輝度を制御する。デューティの変更はタイマH1の割り込みを利用して500ms周期で行い、同時にLED2の出力を反転する。

<主な設定内容>

- ・ウォッチドッグ・タイマの動作停止
- ・低電圧検出電圧VLVIを4.3V±0.2Vに設定
- ・VDD VLVIとなった後にVDD < VLVIとなった場合、内部リセット信号発生(低電圧検出回路)
- ・CPUクロックを8MHzに設定
- ・周辺ハードウェアへの供給クロックを8MHzに設定

<8ビット・タイマH1の設定>

- ・PWMモードに設定
- ・TOH1端子のタイマ出力許可
- ・カウント・クロック = $f_{xp}/2^6$ (125kHz)
- ・タイマ周期 = 2ms ($8[\mu\text{s}/\text{clk}] \times 250[\text{count}] = 2[\text{ms}]$)

< PWM出力のデューティとLED >

- ・ 500ms毎に以下の順でデューティが変わると同時に、LED2出力が反転する。

```
+-----+-----+
| PWM出力のデューティ | 10% | 30% | 50% | 70% | 90% | (以降10%から繰り返し)
+-----+-----+
| LED1の輝度          | 90% | 70% | 50% | 30% | 10% |
+-----+-----+
```

PWM出力がHighアクティブであり、LED1がLowアクティブであることから、
LEDの輝度 = 100 - デューティ比となる。

【ポート入出力の設定】

入力ポート：-

出力ポート：P00-P03, P20-P23, P30-P33, P40-P47, P120-P123, P130

未使用のポート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 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 */
#pragma interrupt INTTMH1 fn_inttmH1 /* 割り込み関数宣言:INTTMH1 */
```

/*=====

グローバル変数の定義

=====*/

```
sreg unsigned char ucTMH1cnt = 250; /* INTTMH1割り込み回数カウンタ用8ビット変数 */
```

リセット解除後の初期化処理

*****/

```

void hdwinit(void){
    unsigned char ucCnt200us;      /* 200usウェイト用8ビット変数 */

    /*-----
    ウォッチドッグ・タイマの設定 + 低電圧検出 + クロックの設定
    -----*/

    /* ウォッチドッグ・タイマの設定 */
    WDTM = 0b01110111;           /* ウォッチドッグ・タイマ動作停止 */

    /* クロックの設定1 */
    PCC = 0b00000000;           /* CPUクロックfcpu = fxp (= fx/4 = 2MHz) */
    LSRCM = 0b00000001;        /* 低速内蔵発振器の発振を停止 */

    /* リセット要因の確認 */
    if (!(RESF & 0b00000001)){   /* LVIリセット時は以降のLVI関連処理を省略 */

        /* 低電圧検出の設定 */
        LVIS = 0b00000000;      /* 低電圧検出レベルVLVI = 4.3V ± 0.2Vに設定 */
        LVION = 1;              /* 低電圧検出回路の動作許可 */

        for (ucCnt200us = 0; ucCnt200us < 9; ucCnt200us++){ /* 約200usウェイト */
            NOP();
        }

        while (LVIF){           /* VDD VLVI待ち */
            NOP();
        }

        LVIMD = 1;             /* VDD < VLVI時に内部リセット信号が発生するように設定 */
    }

    /* クロックの設定2 */
    PPCC = 0b00000000;         /* 周辺ハードウェアへの供給クロックfxp = fx (= 8MHz)
                                -> CPUクロックfcpu = fxp = 8MHz */

    /*-----
    ポート0の設定
    -----*/

    P0 = 0b00000000;          /* P00-P03の出力ラッチLow */
    PM0 = 0b11110000;        /* P00-P03を出力ポートに設定 */

    /*-----
    ポート2の設定

```

```

-----*/
P2   = 0b00000000;      /* P20-P23の出力ラッチLow(P21:LED2点灯) */
PM2  = 0b11110000;      /* P20-P23を出力ポートに設定 */

/*-----
   ポート3の設定
-----*/

P3   = 0b00000000;      /* P30-P33の出力ラッチLow */
PM3  = 0b11110000;      /* P30-P33を出力ポートに設定 */

/*-----
   ポート4の設定
-----*/

P4   = 0b00000000;      /* P40-P47の出力ラッチLow(P42:LED1点灯) */
PM4  = 0b00000000;      /* P40-P47を出力ポートに設定 */

/*-----
   ポート12の設定
-----*/

P12  = 0b00000000;      /* P120-P123の出力ラッチLow */
PM12 = 0b11110000;      /* P120-P123を出力ポートに設定 */

/*-----
   ポート13の設定
-----*/

P13  = 0b00000001;      /* P130の出力High */

/*-----
   8ビット・タイマH1の設定
-----*/

TMHMD1 = 0b00111001;      /* カウント・クロック = fxp/26 = 125kHz、PWMモード、TOH1出力許可 */
CMP01  = 250-1;           /* CMP01を初期化(周期:2ms) */
CMP11  = 25-1;           /* CMP11を初期化(デューティ:10%) */
TMHE1  = 1;              /* タイマ動作開始 */

/*-----
   割り込みの設定
-----*/

IF0   = 0x00;            /* 無効割り込み要求をクリアしておく */
TMMKH1 = 0;             /* INTTMH1割り込みマスク解除 */

return;
}

```

```
/******  
  
    メイン・ループ  
  
*****/  
void main(void){  
  
    EI();                /* ベクタ割り込み許可 */  
  
    while (1){  
        NOP();  
        NOP();  
    }  
}  
  
/******  
  
    割り込みINTTMH1  
  
*****/  
__interrupt void fn_inttmH1(){  
  
    if (--ucTMH1cnt == 0){ /* INTTMH1の割り込み回数が250回の場合の処理 */  
  
        ucTMH1cnt = 250;    /* 割り込み回数を初期化 */  
        P2 ^= 0b00000010; /* LED2を反転 */  
  
        if (CMP11 >= 225-1){ /* デューティが90%以上の場合の処理 */  
            CMP11 = 25-1; /* デューティを10%に初期化 */  
        }  
        else {  
            CMP11 += 50; /* デューティを20%上げる */  
        }  
    }  
  
    return;  
}
```


op.asm (アセンブリ言語版とC言語版共通)

```

;=====
;
; オプション・バイトの設定
;
;=====
OPBT      CSEG  AT      0080H
          DB      10011100B      ; オプション・バイトの設定
;
;          |||
;          |||+----- 低速内蔵発振器はソフトウェアで停止可能
;          |++----- 高速内蔵発振クロック(8MHz)を使用
;          +----- P34/RESET端子をリセット端子として使用

          DB      11111111B      ; プロテクト・バイトの設定(セルフプログラミング用)
;
;          |||
;          +----- 全てのブロックへの書き込み許可

end

```

付録B 改版履歴

本文欄外の 印は、本版で改訂された主な箇所を示しています。この" "をPDF上でコピーして「検索する文字列」に指定することによって、改版箇所を容易に検索できます。

版 数	発行年月	改版箇所	改版内容
第1版	July 2007	-	-
第2版	July 2008	p.20	第5章 システム・シミュレータ SM+での動作確認 ・注意文中の(2007年7月現在)を(2008年7月現在)に変更
		pp.20-22	5.1 サンプル・プログラムのビルドを変更
		p.22	5.2 SM+での動作 ・(1)を追加
		p.27	第6章 関連資料 ・フラッシュ書き込み簡単マニュアル(MINICUBE2編) インフォメーションを追加

【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，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)

電 話 : 044-435-9494

E-mail : info@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約店へお申し付けください。
