

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

## 78K0/Kx2

### サンプル・プログラム (A/Dコンバータ)

この資料は、サンプル・プログラムの「A/Dコンバータ」の動作概要と、A/Dコンバータを使用する際の設定を説明したものです。サンプル・プログラムでは、A/Dコンバータ設定を行ったあとに、4つのアナログ入力電圧に応じて4つのLEDの点灯の制御を行います。

#### 対象デバイス

78K0/KB2マイクロコントローラ  
78K0/KC2マイクロコントローラ  
78K0/KD2マイクロコントローラ  
78K0/KE2マイクロコントローラ  
78K0/KF2マイクロコントローラ

#### 目次

第1章 概要 ...	3
第2章 回路図 ...	4
2.1 回路図 ...	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 ...	7
3.3 初期設定と動作概要 ...	7
3.4 フロー・チャート ...	8
第4章 設定方法について ...	10
4.1 前処理指令 ...	10
4.2 インターバル・タイマの設定 ...	10
4.3 A/Dコンバータの初期設定 ...	11
4.4 A/D変換値の取得 ...	17
第5章 新統合開発環境Cube Suiteでの動作確認 ...	19
5.1 サンプル・プログラムのビルド ...	19
5.2 シミュレータでの動作 ...	21
5.3 オン・チップ・デバッグ時の注意 ...	23
第6章 関連資料 ...	26
付録A 改版履歴 ...	27

・本資料に記載されている内容は2010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、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、マイクロコントローラの初期設定の後、ADコンバータの初期設定を行います。

ADコンバータ初期設定完了後のメイン処理動作では、1msのインターバル・タイマ割り込みとAD変換完了割り込みを待ちます。インターバル・タイマ割り込み処理内では、AD変換を開始します。AD変換完了割り込み内では、変換結果をチャンネル毎に蓄積し、AD変換を停止します。また、32msを1周期とし、4ch×8回分のサンプリングを行った後、平均値を各変数に保存します。平均化された値を元にアナログ入力ポートに対応するLEDの点灯と消灯を行います。

## (1) ADコンバータ初期設定の主な内容

- ・使用するアナログチャンネルポートの設定
- ・使用するアナログチャンネルポートのポートモード設定
- ・AD変換時間の設定

## (2) メイン処理動作の内容

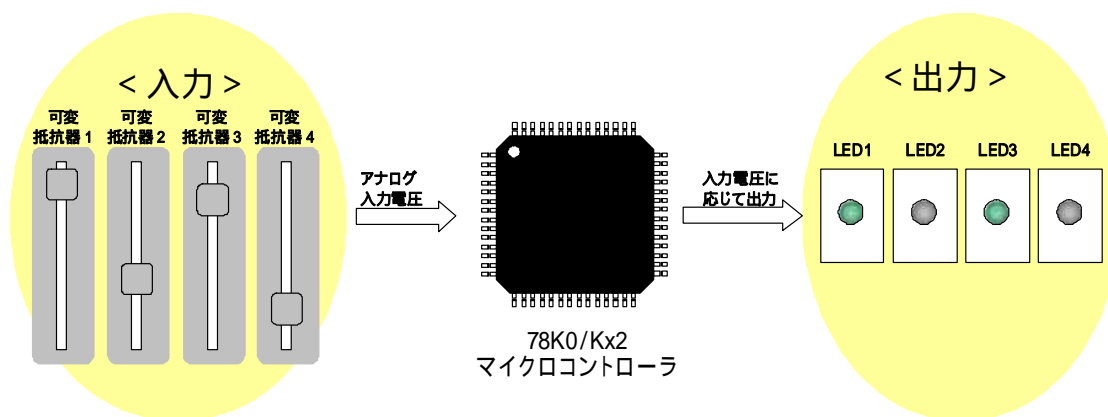
- インターバル・タイマ割り込み待ち
- AD変換完了割り込み待ち

## (3) インターバル・タイマ割り込み処理動作の内容

- A/D変換動作を開始

## (4) A/D変換完了割り込み処理動作の内容

- 取得した10bit変換結果をチャンネル毎に蓄積し、A/D変換を停止。
- 32msを1周期とし、4ch×8回分のサンプリング後、平均値を各変数に保存。
- 平均化された値を元にアナログ入力ポートに対応するLEDの点灯と消灯を行います。

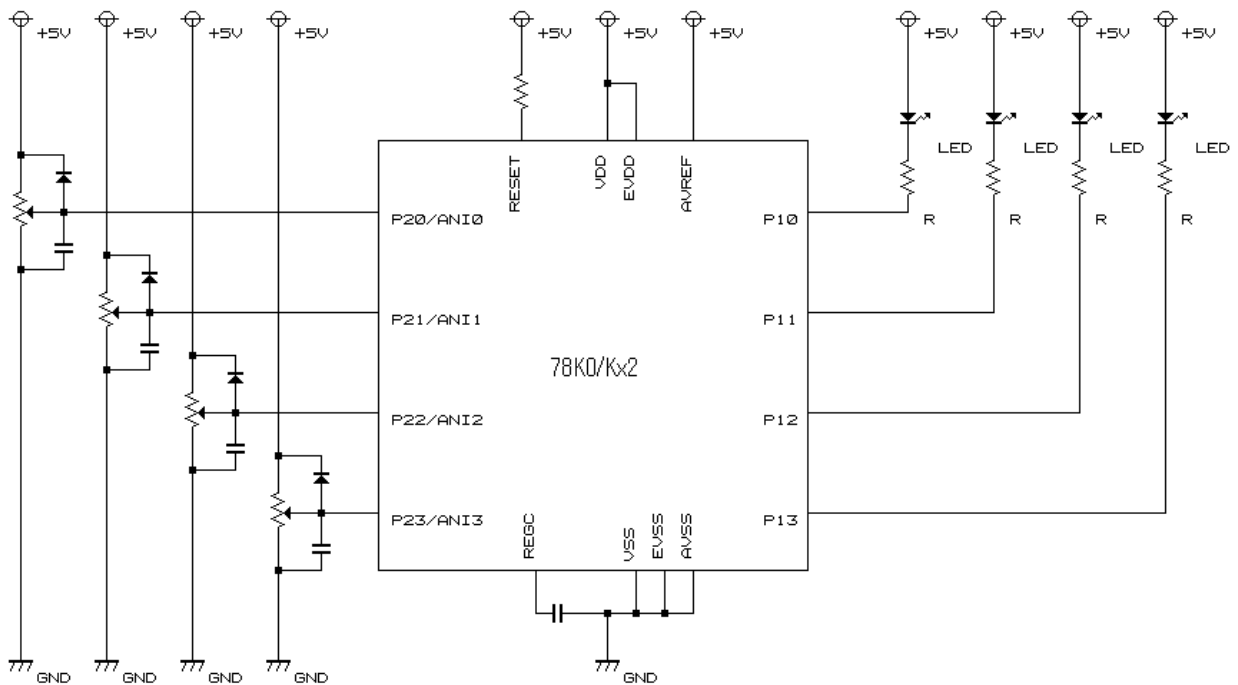


## 第2章 回路イメージ

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周辺ハードウェアを説明します。

### 2.1 回路イメージ

回路イメージを次に示します。



- 注意1. AVREF端子には電圧5Vを供給してください。
2. AVSS端子はGNDに直接接続してください。
3. REGC端子はコンデンサ(0.47 ~ 1  $\mu$ F)を介し、Vssに接続してください。
4. EVDD端子はVDDに直接接続してください。(78K0/KE2、78K0/KF2のみ)
5. EVSS端子はGNDに直接接続してください。(78K0/KE2、78K0/KF2のみ)
6. 使用電源電圧と動作周波数等の詳細についてはユーザズマニュアルを参照ください。

### 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。

#### ・可変抵抗器

アナログ入力チャンネルANI0-ANI3の入力値を変更するため、可変抵抗器を使用します。

#### ・LED

アナログ入力チャンネルANI0-ANI3の入力値の状態を表すため、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。

【C言語版】

ファイル名 <sup>注1</sup>	説明	同封圧縮(*.zip)ファイル		
				
Kx2_AD.c	使用周辺ハードウェアの設定とメイン処理およびベクタ処理			
Kx2_subr.c	マイコンのハードウェア初期化処理			
Kx2_op.asm	オプション・バイトの設定			
78K0サンプルプログラム.cspj	新統合開発環境 CubeSuite用プロジェクト・ファイル			
78K0サンプルプログラム.cssp	新統合開発環境 CubeSuite用サブプロジェクト・ファイル			
ADC.pnl	シミュレータGUI用パネル・ファイル			
78K0サンプルプログラム.prj	統合開発環境 PM+用プロジェクト・ファイル			
78K0サンプルプログラム.prw	統合開発環境 PM+用ワーク・スペース・ファイル			
プロジェクト変換情報_ 78K0サンプルプログラム.txt	PM+からCubeSuiteへの変換情報ファイル			

注1. 各ファイル名の"Kx2"部分は、それぞれのデバイスの名前になります。

ex) 78K0/KB2の場合 "KB2\_AD.c"

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 78K0/Kx2で使用するファイルを同封



: 新統合開発環境 CubeSuite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 <sup>注1</sup>	説明	同封圧縮(*.zip)ファイル		
				
Kx2_AD.asm	使用周辺ハードウェアの設定とメイン処理およびベクタ処理			
Kx2_subr.asm	マイコンのハードウェア初期化処理			
Kx2_op.asm	オプション・バイトの設定			
78K0サンプルプログラム.cspj	新統合開発環境 CubeSuite用プロジェクト・ファイル			
78K0サンプルプログラム.cssp	新統合開発環境 CubeSuite用サブプロジェクト・ファイル			
ADC.pnl	シミュレータGUI用パネル・ファイル			
78K0サンプルプログラム.prj	統合開発環境 PM+用プロジェクト・ファイル			
78K0サンプルプログラム.prw	統合開発環境 PM+用ワーク・スペース・ファイル			
プロジェクト変換情報_ 78K0サンプルプログラム.txt	PM+からCubeSuiteへの変換情報ファイル			

注1. 各ファイル名の“Kx2”部分は、それぞれのデバイスの名前になります。

ex) 78K0/KB2の場合 “KB2\_AD.asm”

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 78K0/Kx2で使用するファイルを同封



: 新統合開発環境 CubeSuiteで使用するファイルを同封



### 3.2 使用するマイコン内蔵周辺機能

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。

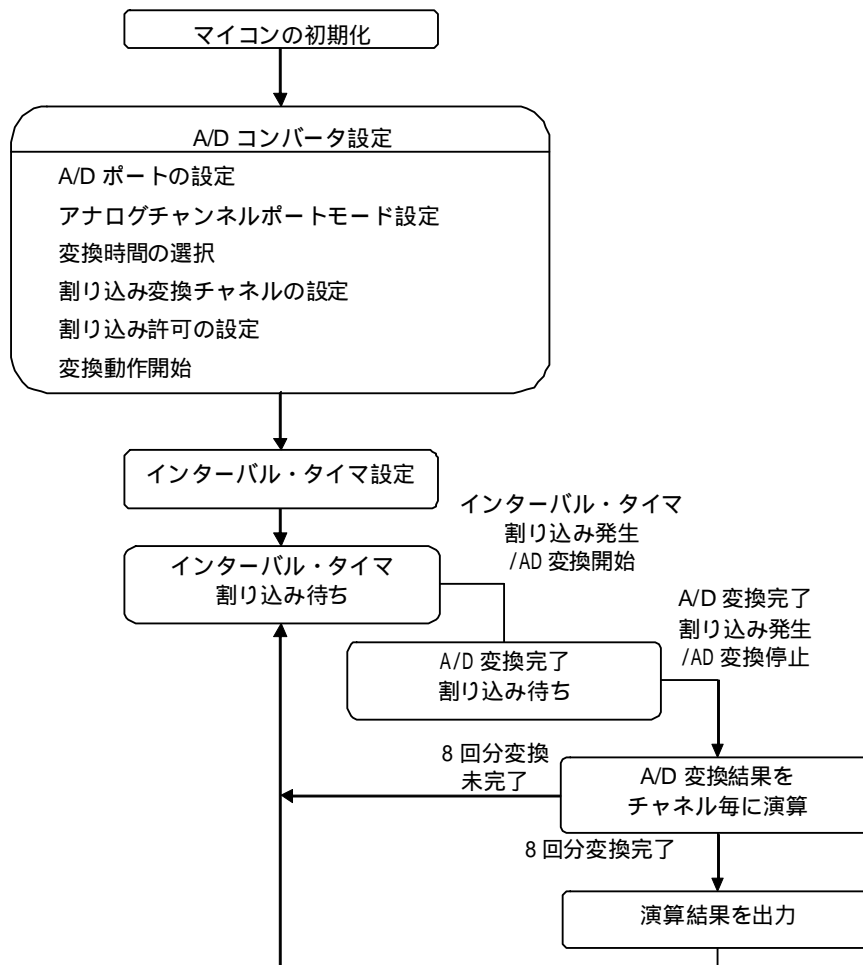
- ・入力ポート(アナログ入力用) : P20/ANI0, P21/ANI1, P22/ANI2, P23/ANI3
- ・出力ポート(LED駆動用) : P10, P11, P12, P13

### 3.3 初期設定と動作概要

このサンプル・プログラムでは、A/Dコンバータ設定にて、A/Dポートや、アナログチャンネルポートモード、A/D変換時間の設定を行います。

A/Dコンバータ設定後は、インターバル・タイマで1ms毎にA/D変換を開始します。A/D変換が開始されると、アナログチャンネルからの入力値をA/D変換し、変換結果をチャンネル毎に8回分加算後、平均値を算出します。

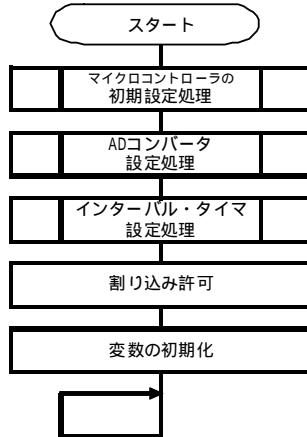
以下に、状態遷移図(ステート・チャート)に示します。詳細については、次のフローチャートを参照ください。



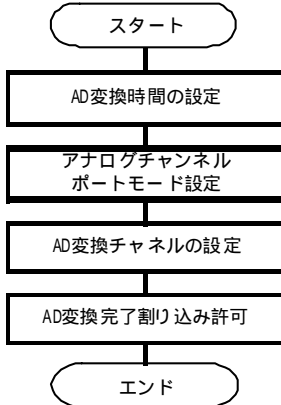
### 3.4 フロー・チャート

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。

処理全体 (C言語版:Kx2\_Ad.c アセンブリ言語版:Kx2\_Ad.asm)



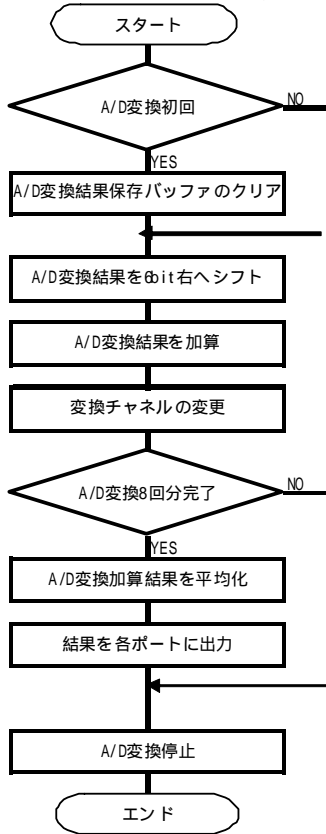
ADコンバータ設定処理 (C言語版:Kx2\_Ad.c アセンブリ言語版:Kx2\_Ad.asm)



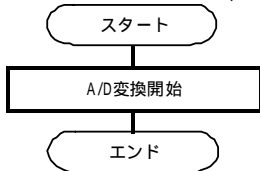
インターバル・タイマ設定処理 (C言語版:Kx2\_Ad.c アセンブリ言語版:Kx2\_Ad.asm)

インターバル・タイマの設定に関しては、インターバル・タイマ\_TM00編を参照ください。

AD変換完了割り込み処理 (C言語版:Kx2\_Ad.c アセンブリ言語版:Kx2\_Ad.asm)



インターバル・タイマ割り込み処理 (C言語版:Kx2\_Adc.c アセンブリ言語版:Kx2\_Adc.asm)



## 第4章 設定方法について

この章では、ADコンバータの設定について説明します。

レジスタ設定方法の詳細については、[78K0/Kx2 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、[78K0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 前処理指令

C言語において、SFR名称に関する操作、CPU制御命令、割り込み関数などを使用するためには、#pragma指令にてソース・プログラムの冒頭に前処理指令を記述する必要があります。本サンプル・プログラムで使用する前処理指令は以下のとおりです。ここではADC版のみを例として説明します。

【C言語】 (Kx2\_Adc.c)

/*=====	
前処理指令(#pragma指令)	
=====*/	
#pragma sfr	特殊機能レジスタ(SFR)名称を記述可能にします。
#pragma di	DI命令を記述可能にします。
#pragma ei	EI命令を記述可能にします。
#pragma nop	NOP命令を記述可能にします。
#pragma interrupt INTTM00 fn_intTimerInterval	インターバル・タイマ割り込み関数を宣言します。
#pragma interrupt INTAD fn_intAdConverter	A/D変換完了割り込み関数を宣言します。

### 4.2 インターバル・タイマの設定

インターバル・タイマは、次の項目を設定します。

- 1) タイマ・クロック選択
- 2) タイマ・モードの設定
- 3) タイマ・コンペア・レジスタの設定
- 4) タイマの動作許可の設定

インターバル・タイマの詳細は、78K0/Kx2サンプル・プログラム、インターバル・タイマ編を参照ください。

### 4.3 A/Dコンバータの初期設定

A/Dコンバータは、次の項目を設定します。

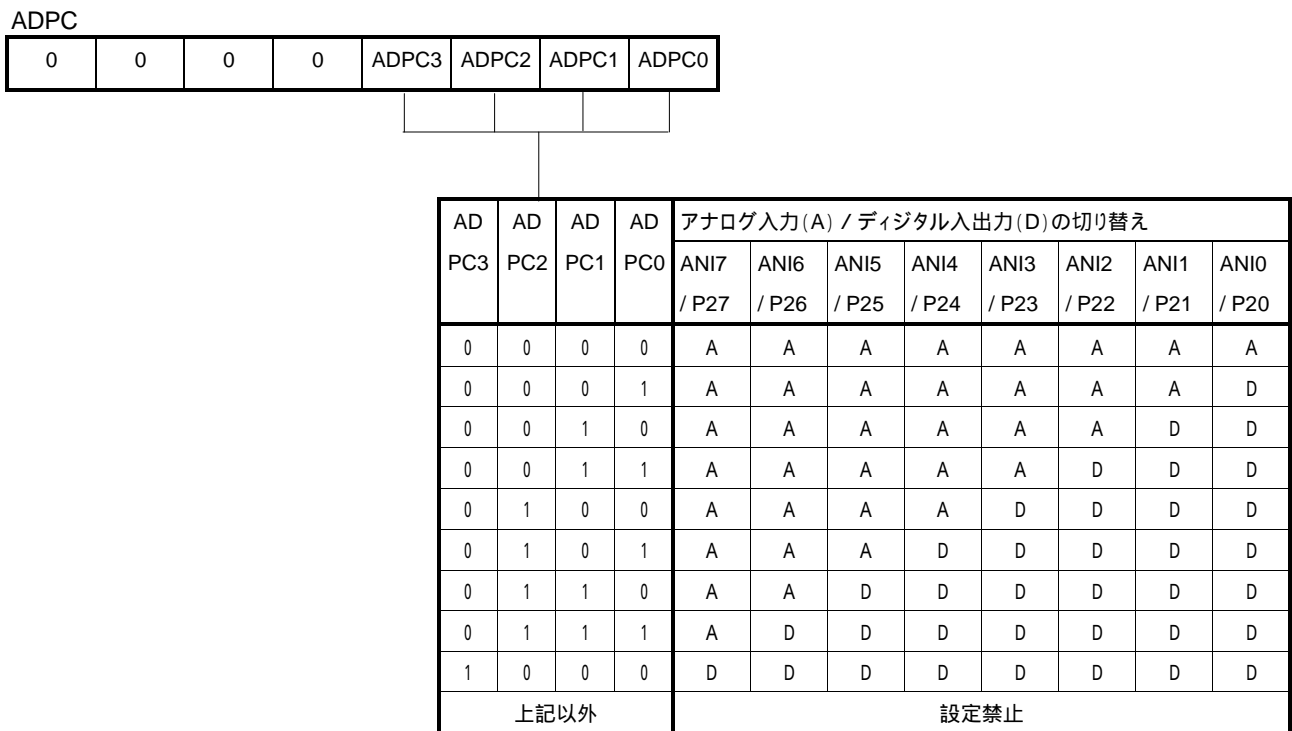
- 1) アナログチャンネルポートの設定
- 2) ポート2のモード設定
- 3) アナログ入力チャンネルの設定
- 4) A/Dコンバータの動作設定

本サンプル・プログラムでは、後述の【例 1】の内容で設定しています。

#### (1) アナログチャンネルポートの設定

P20/ANI0 ~ P27/ANI7端子のアナログ入力 / デジタル入出力を切り替えます。

図4 - 1 A/Dポート・コンフィギュレーション・レジスタ(ADPC)

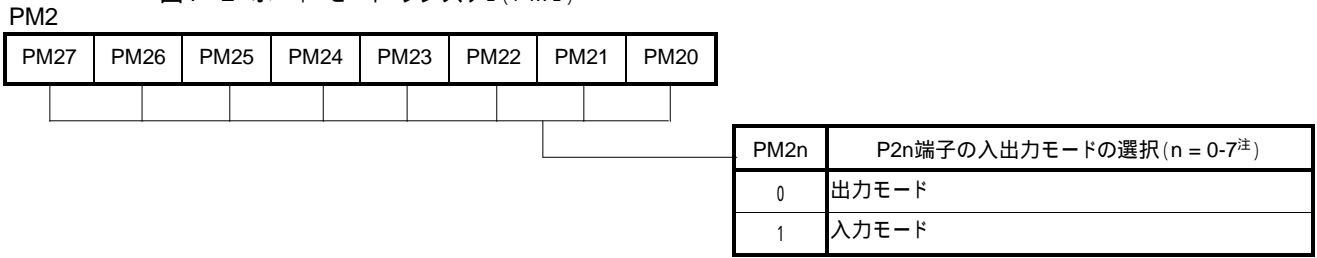


- 注意1. A/D変換で使用するチャンネルは、ポート・モード・レジスタ2(PM2)で入力モードに選択してください。
2. ADPCでデジタル入出力として設定した端子を、ADSで設定しないでください。
  3. ADPCにデータを書き込むと、ウェイトが発生します。またCPUがサブシステム・クロックで動作し、かつ周辺ハードウェア・クロックが停止しているときに、ADPCにデータを書き込まないでください。
  4. 78K0/KB2では、ADPC3は存在せず(0固定)、ADPC2が1の場合、ADPC1またはADPC0に1を設定することはできません。

(2)ポート2のモード設定

使用するアナログ入力端子を入力端子に設定します。

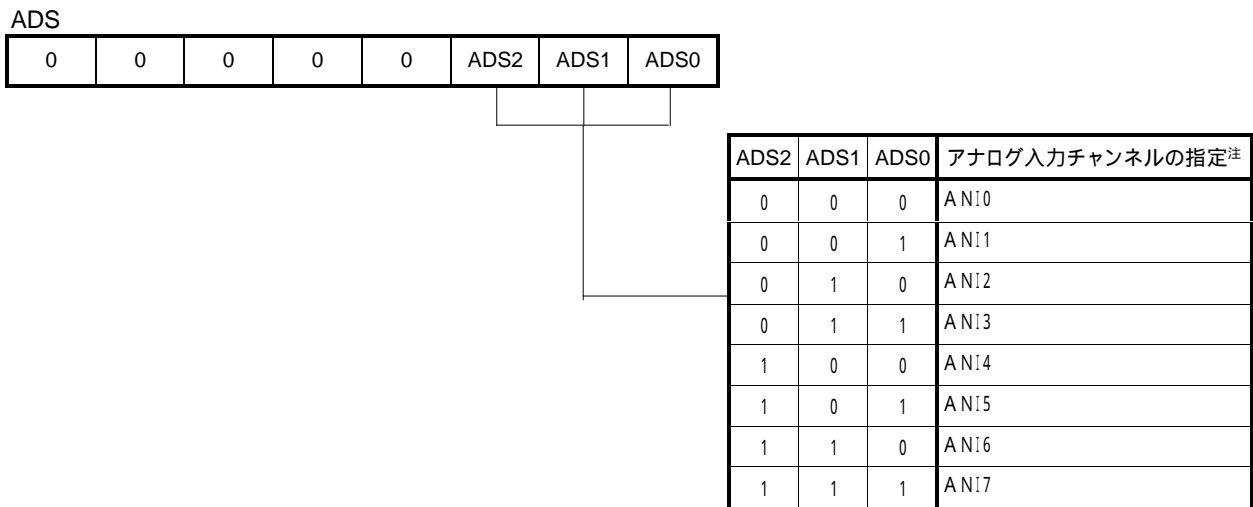
図4 - 2 ポート・モード・レジスタ2 (PM2)



(3)アナログ入力チャンネルの設定

ADSでA/D変換するアナログ電圧の入力チャンネルを指定します。

図4 - 3 アナログ入力チャンネル指定レジスタ (ADS)



- 注意1. A/D変換で使用するチャンネルは、ポート・モード・レジスタ2(PM2)で入力モードに選択してください。
- ADPCでデジタル入出力として設定する端子を、ADSで設定しないでください。
  - ADSにデータを書き込むと、ウェイトが発生します。またCPUがサブシステム・クロックで動作し、かつ周辺ハードウェア・クロックが停止しているときに、ADSにデータを書き込まないでください。
  - 78K0/KB2では、ADS2は存在しません。
  - ビット3-7には必ず0を(78K0/KB2ではビット2-7)設定してください。

(4) A/Dコンバータの動作設定

ADMでA/Dコンバータの動作を設定します。

図4 - 4 A/Dコンバータ・モード・レジスタ(ADM)

ADM

ADCS	0	FR2	FR1	FR0	LV1	LV0	ADCE
------	---	-----	-----	-----	-----	-----	------

ADCE	コンパレータの動作制御
0	コンパレータの動作停止
1	コンパレータの動作許可

2.7V AVREF 5.5V の場合

FR2	FR1	FR0	LV1	LV0	変換時間の選択	変換クロック (fAD)
0	0	0	0	0	264/fPRS	fPRS/12
0	0	1	0	0	176/fPRS	fPRS/8
0	1	0	0	0	132/fPRS	fPRS/6
0	1	1	0	0	88/fPRS	fPRS/4
1	0	0	0	0	66/fPRS	fPRS/3
1	0	1	0	0	44/fPRS	fPRS/2
上記以外					設定禁止	

2.3V AVREF < 2.7V の場合

FR2	FR1	FR0	LV1	LV0	変換時間の選択	変換クロック (fAD)
0	0	0	0	1	480/fPRS	fPRS/12
0	0	1	0	1	320/fPRS	fPRS/8
0	1	0	0	1	240/fPRS	fPRS/6
0	1	1	0	1	160/fPRS	fPRS/4
1	0	0	0	1	120/fPRS	fPRS/3
1	0	1	0	1	80/fPRS	fPRS/2
上記以外					設定禁止	

ADCS	A/D変換動作の制御
0	変換動作停止
1	変換動作許可

- 注意1. コンパレータはADCSとADCEで動作制御され、動作開始から安定するまでに、1  $\mu$ s かかります。  
このため、ADCEに1を設定してから1  $\mu$ s以上経過したあとに、ADCSに1を設定することで、最初の変換データより有効となります。  
1  $\mu$ s以上ウエイトしないでADCSに1を設定した場合は、最初の変換データを無視してください。
- FR2-FR0, LV1, LV0を同一データ以外に書き換える場合は、いったんA/D変換動作を停止させたのちに行ってください
  - 変換時間は、次の条件で設定してください。
    - ・4.0 V  $AV_{REF}$  5.5 Vの場合 :  $f_{AD} = 0.6 \sim 3.6$  MHz
    - ・2.7 V  $AV_{REF} < 4.0$  Vの場合 :  $f_{AD} = 0.6 \sim 1.8$  MHz
    - ・2.3 V  $AV_{REF} < 2.7$  Vの場合 :  $f_{AD} = 0.6 \sim 1.48$  MHz
  - クロック周波数の誤差を考慮して、変換時間を選択してください。

備考  $f_{PRS}$ : 周辺ハードウェア・クロック周波数



- 【例 1】 ・アナログ・チャンネル0から入力  
 ・2.7V 基準電圧(AVREF) 5.5V  
 ・変換時間: 22 $\mu$ s (176/fPRS)  
 (サンプル・プログラムの設定と同内容)

ADPC

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

AD PC3	AD PC2	AD PC1	AD PC0	アナログ入力(A) / デジタル入出力(D)の切り替え							
				ANI7 / P27	ANI6 / P26	ANI5 / P25	ANI4 / P24	ANI3 / P23	ANI2 / P22	ANI1 / P21	ANI0 / P20
0	0	0	0	A	A	A	A	A	A	A	A

PM2

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

PM2n	P2n端子の入出力モードの選択 (n = 0-7)
1	入力モード

ADS

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

ADS2	ADS1	ADS0	アナログ入力チャンネルの指定
0	0	0	ANI0
0	0	1	ANI1
0	1	0	ANI2
0	1	1	ANI3

ADM

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

ADCE	コンパレータの動作制御
0	コンパレータの動作停止
1	コンパレータの動作許可

2.7V AVREF 5.5V の場合

FR2	FR1	FR0	LV1	LV0	変換時間の選択	変換クロック (fAD)
0	0	1	0	0	176/fPRS	fPRS/8

ADCS	A/D変換動作の制御
0	変換動作停止
1	変換動作許可

サンプル・プログラムでは以下のようになります。

【C言語】 (Kx2\_Adc.c)

```
void fn_InitAd(void)
{
    ADM    =    0b00001001;    /*[A/Dコンバータ・モード・レジスタ]*/
    PM2    =    0b11111111;    /*[ポート2の入出力モード選択]*/
    ADPC   =    0b00000000;    /*[A/Dポート・コンフィギュレーション・レジスタ]*/
    ADS    =    0b00000000;    /*[アナログ入力チャネル指定レジスタ]*/
}
```

【アセンブリ言語】 (Kx2\_Adc.asm)

```
MOV    ADM,    #00001001B    ;[A/Dコンバータ・モード・レジスタ]
MOV    PM2,    #11111111B    ;[ポート・モード・レジスタ]
MOV    ADPC,   #00000000B    ;[A/Dポート・コンフィギュレーション・レジスタ]
MOV    ADS,    #00000000B    ;[アナログ入力チャネル指定レジスタ]
```

## 4.4 A/D変換値の取得

A/D変換値を保持するレジスタには以下の2種類があります。

- 1) 10ビットA/D変換結果レジスタ(ADCR)
- 2) 8ビットA/D変換結果レジスタ(ADCRH)

### (1) 10ビットA/D変換結果レジスタ

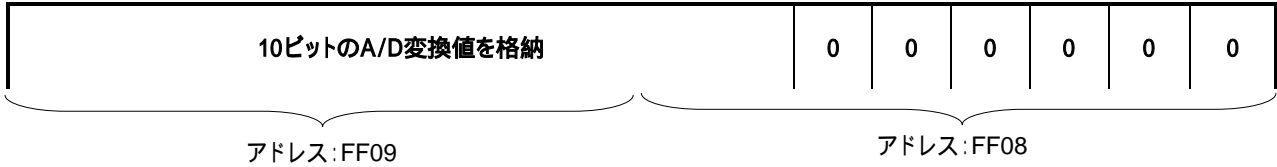
A/D変換結果を保持する16ビットのレジスタです。

下位6ビットは“0”固定です。

A/D変換が終了するたびに、逐次変換レジスタから変換結果がロードされます。

変換結果の上位8ビットがFF09Hに、下位2ビットがFF08Hの上位2ビットに格納されます。

ADCR



**注意** A/Dコンバータ・モード・レジスタ(ADM)、アナログ入力チャンネル指定レジスタ(ADS)、A/Dポート・コンフィギュレーション・レジスタ(ADPC)に対して書き込み動作を行ったとき、ADCRの内容は不定となることがあります。変換結果は、変換動作終了後、ADM、ADS、ADPCに対して書き込み動作を行う前に読み出してください。上記以外のタイミングでは、正しい変換結果が読み出されないことがあります。

### (2) 8ビットA/D変換結果レジスタ

A/D変換結果を保持する8ビットのレジスタです。10ビット分解能の上位8ビットを格納します。

ADCRHは、8ビット・メモリ操作命令で読み出せます。

リセット信号の発生により、00Hになります。

ADCRH



**注意** A/Dコンバータ・モード・レジスタ(ADM)、アナログ入力チャンネル指定レジスタ(ADS)、A/Dポート・コンフィギュレーション・レジスタ(ADPC)に対して書き込み動作を行ったとき、ADCRHの内容は不定となることがあります。変換結果は、変換動作終了後、ADM、ADS、ADPCに対して書き込み動作を行う前に読み出してください。上記以外のタイミングでは、正しい変換結果が読み出されないことがあります。

サンプル・プログラムでは以下のようにA/D変換値を取得しています。

【C言語】 (Kx2\_Adc.c)

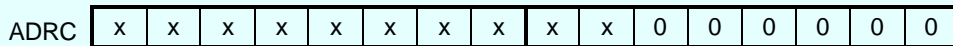
```
g_ucAdData[g_ucAdCh] += ADCR >> 6;
/*A/D変換結果の下位6bitを削除*/
```

【アセンブリ言語】 (Kx2\_Adc.asm)

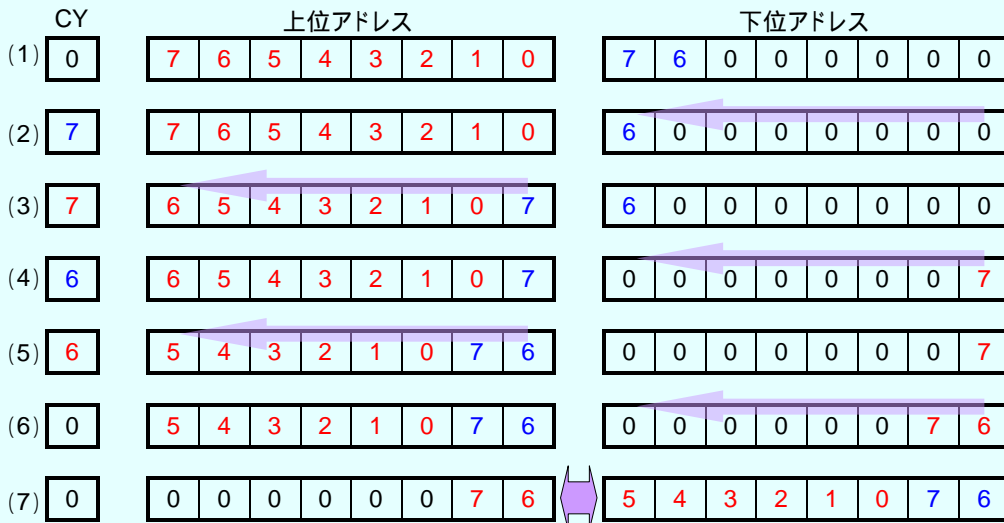
```
MOVW AX, ADCR ;AD変換結果取得
XCH A, X ;
ADD A, A ;下位8ビットを桁上げ
XCH A, X ;
ADDC A, A ;上位8ビットを桁上げとキャリーを取り込み
XCH A, X ;
ADDC A, A ;下位8ビットを桁上げ
XCH A, X ;
ADDC A, A ;上位8ビットを桁上げとキャリーを取り込み
XCH A, X ;下位8ビット演算完了
ADDC A, A ;上位8ビット演算完了
XCH A, X ;下位8ビット変換結果を合計
ADD A, [HL] ;
MOV [HL], A ;
INCW HL ;次回上位アドレスを変更
XCH A, X ;上位8ビット変換結果を合計
ADDC A, [HL] ;
MOV [HL], A ;
INCW HL ;次チャンネルの下位アドレスに変更
```




【コラム】 10bitAD変換結果の右詰めについて





10bitAD変換結果格納レジスタのADRCレジスタは左詰で格納される。しかし、このままでは後段で演算を行う場合、非常に不便であり、右シフトを6回行ってから変換結果を蓄積する必要がある。また、アセンブラ8bit命令でこの処理を行うと、相当な処理時間が必要になる。以下に、効率的に処理を行うイメージ例を示す。処理プログラムについては、上のアセンブリ言語を参照。



## 第5章 新統合開発環境Cube Suite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プログラム・ファイルを用い、サンプル・プログラムが、シミュレータでどのように動作するかを説明します。

### 5.1 サンプル・プログラムのビルド

サンプル・プログラムをシミュレータ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、シミュレータを起動する必要があります。ここでは、  でダウンロードし  プログラム・ファイルを用いて、新統合開発環境CubeSuiteにてビルドしてから、シミュレータを起動するまでの動作の一例を説明します。

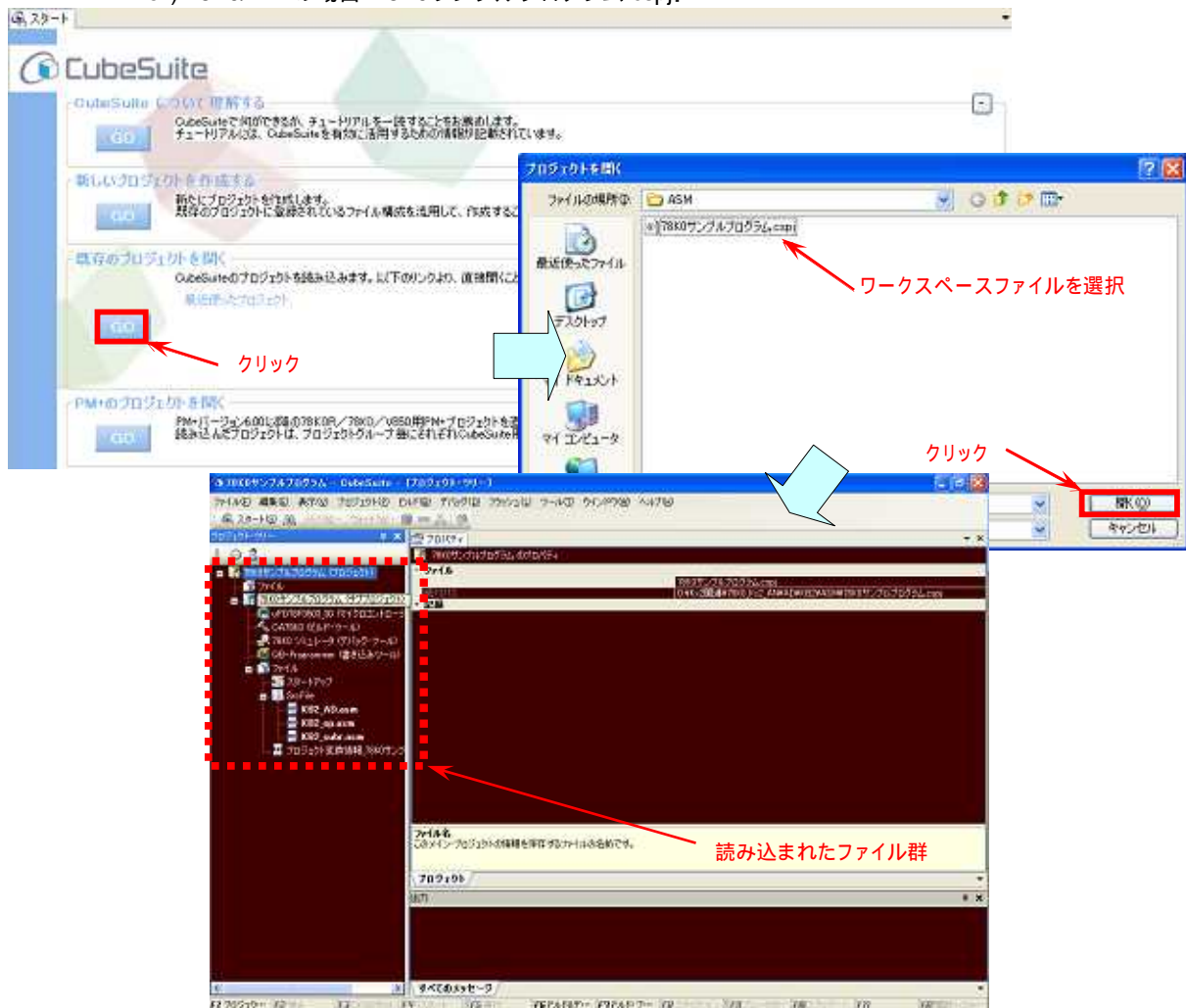
CubeSuite操作方法の詳細については、[CubeSuite ユーザーズ・マニュアル](#)を参照してください。

(1) CubeSuiteを起動してください。

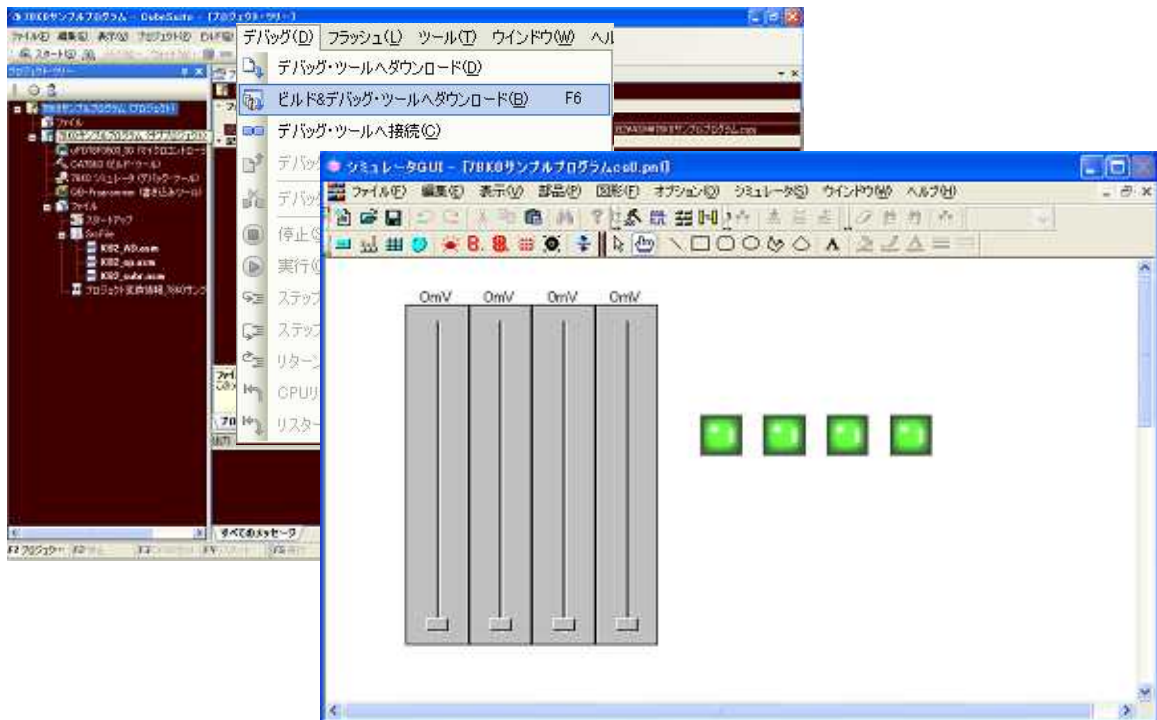
(2) [ スタート ] [ 既存のプロジェクトを開く ] から、「78K0サンプルプログラム.cspj」<sup>注</sup>を選択し、[ 開く ] ボタンをクリックしてください。ワークスペースが作成され、その中にソース・ファイルが自動的に読み込まれます。

注: 対象デバイスによりKx2の名称が変化します。

ex) 78K0/KB2の場合「78K0サンプルプログラム.cspj」



- (3) 「ビルド&デバッグ・ツールヘダダウンロード」をクリックしてください。ソース・ファイルの「Kx2\_AD.asm」と「Kx2\_subr.asm」と「Kx2\_op.asm」が正常にビルドされると、自動でシミュレータGUIが立ち上がります。

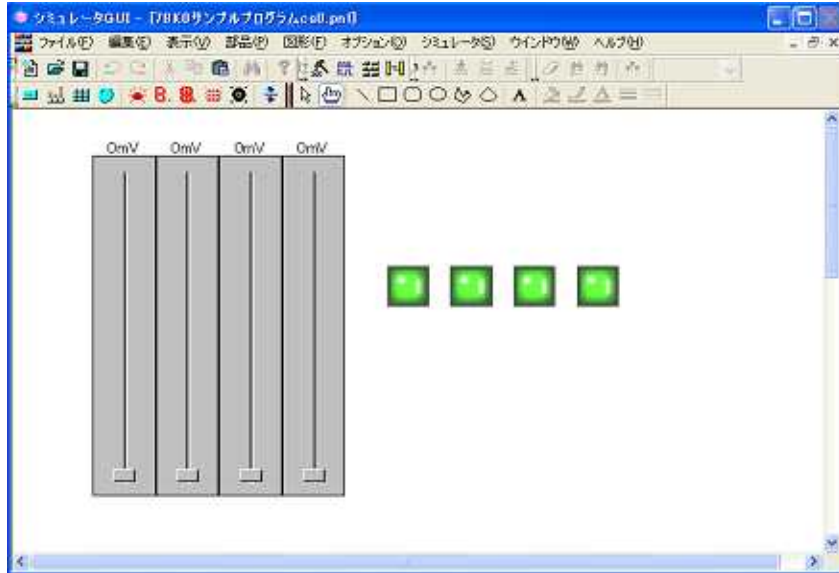


## 5.2 シミュレータでの動作

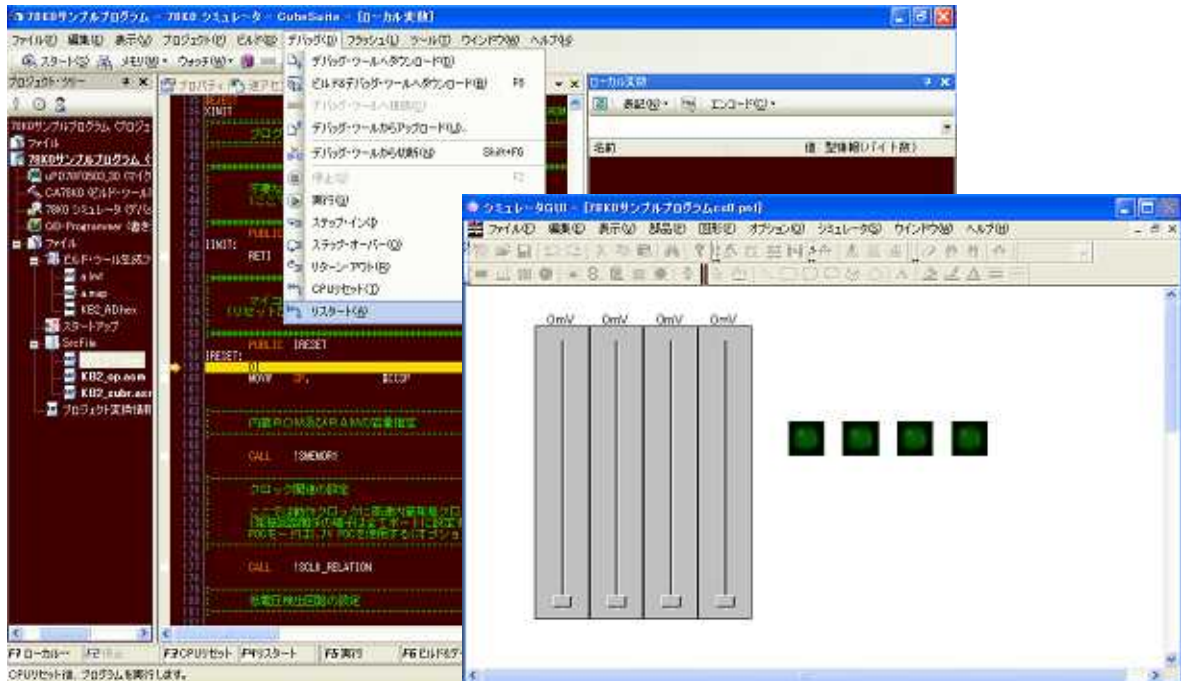
ここでは、SM+の入出力パネル・ウィンドウやタイミング・チャート・ウィンドウ上での動作確認の例を説明します。

シミュレータ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。

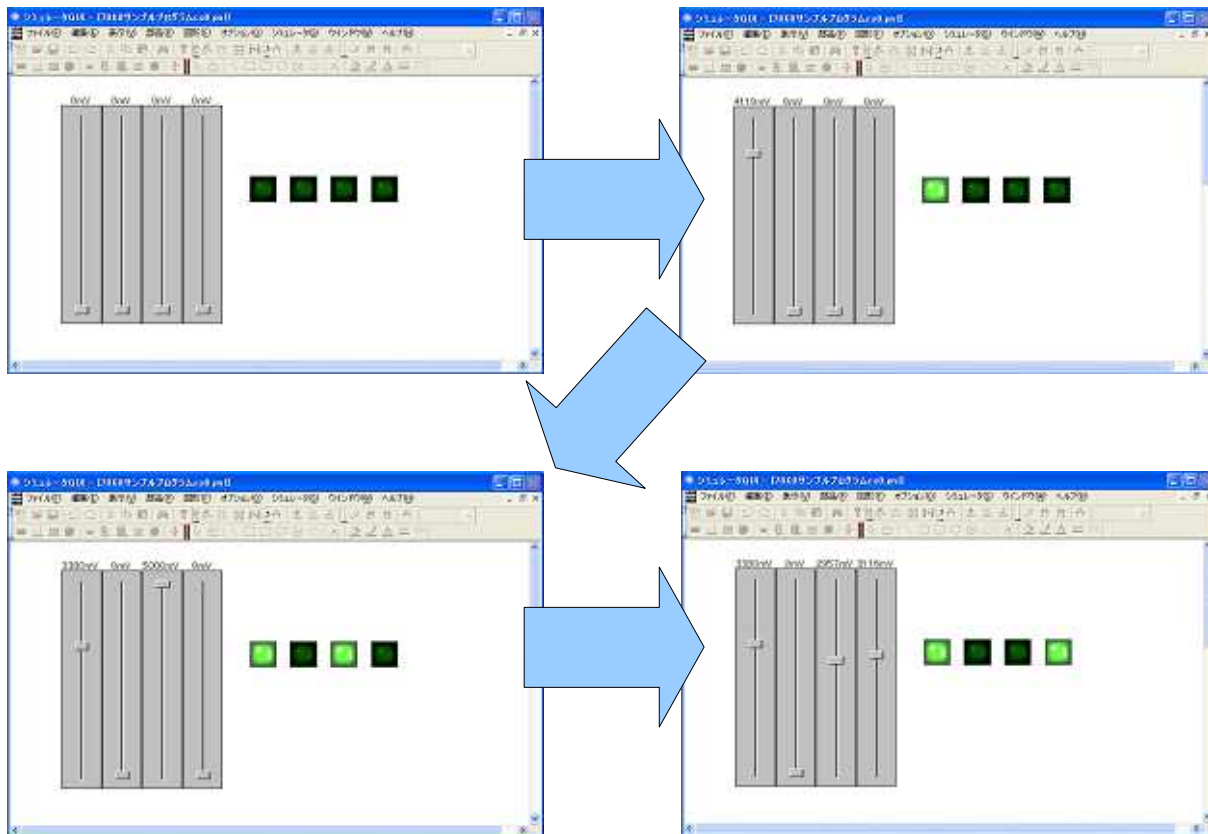
- (1) CubeSuiteの「ビルド&デバッグ・ツールヘダダウンロード」からシミュレータGUIを起動(5. 1を参照)すると、次のような画面になります。



- (2) 「リスタート」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。



- (3) プログラム実行中に, 入出力パネル・ウィンドウ上の [ スライダ ] をクリックしながら上下に動かしてください。  
 [ スライダ ] の位置により, 入出力パネル・ウィンドウ上の [ LED ] の点灯状況が変化することを確認してください。





## 5.3 オン・チップ・デバッグ時の注意

ここでは、サンプルプログラムを用いて、オン・チップ・デバッグを行う際の手順を説明します。

オン・チップ・デバッグ機能についてはユーザーズ・マニュアルを参照ください。

### (1) オプションバイトの設定

本サンプルプログラムはオプションバイトの初期設定でオン・チップ・デバッグ禁止になっています。

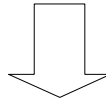
オプションバイトを設定しなおして、オン・チップ・デバッグを許可します。

オプションバイト設定は Kx2\_Mem.inc で行っています。

```

;-----
;               アドレス:0084H
;-----
; オンチップ・デバッグ動作制御
;CIOC DEN      EQU      00000000B;動作禁止
;CIOC DEN EQU   00000001B;設定禁止
;CIOC DEN EQU   00000010B;動作許可,セキュリティID認証失敗時にフラッシュ・メモリのデータを消去しない
;CIOC DEN EQU   00000011B;動作許可,セキュリティID認証失敗時にフラッシュ・メモリのデータを消去する

```



```

コメントアウト -----
;               アドレス:0084H
;-----
; オンチップ・デバッグ動作制御
;CIOC DEN      EQU      00000000B;動作禁止
;CIOC DEN EQU   00000001B;設定禁止
;CIOC DEN EQU   00000010B;動作許可,セキュリティID認証失敗時にフラッシュ・メモリのデータを消去しない
;CIOC DEN EQU   00000011B;動作許可,セキュリティID認証失敗時にフラッシュ・メモリのデータを消去する

```

コメント解除

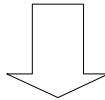
(2) オン・チップ・デバッグ使用領域の確保(アセンブリ言語版のみ)

本サンプル・プログラムのアセンブリ言語版はオン・チップ・デバッグ使用領域を確保する必要があります。

以下のように Kx2\_Init.asmを改造し、オン・チップ・デバッグ使用領域を確保します。

```

;=====
;
;      ベクタテーブル
;=====
TVECTTBL      CSEG      AT      0000H
;
;      DW      IRESET      ;0000H RESET入力, POC, LVI, WDT
;      DW      IINIT       ;
;      DW      IINIT       ;0004H INTLVI
;      DW      IINIT       ;0006H INTPO
    
```



```

;=====
;      ベクタテーブル
;=====
TVECTTBL      CSEG      AT      0000H
;
;      DW      IRESET      ;0000H RESET入力, POC, LVI, WDT
;      DW      IINIT       ;
;      DW      IINIT       ;0004H INTLVI
;      DW      IINIT       ;0006H INTPO
TEST         CSEG      AT      0004H
;
;      DW      IINIT       ;0004H INTLVI
;      DW      IINIT       ;0006H INTPO
    
```

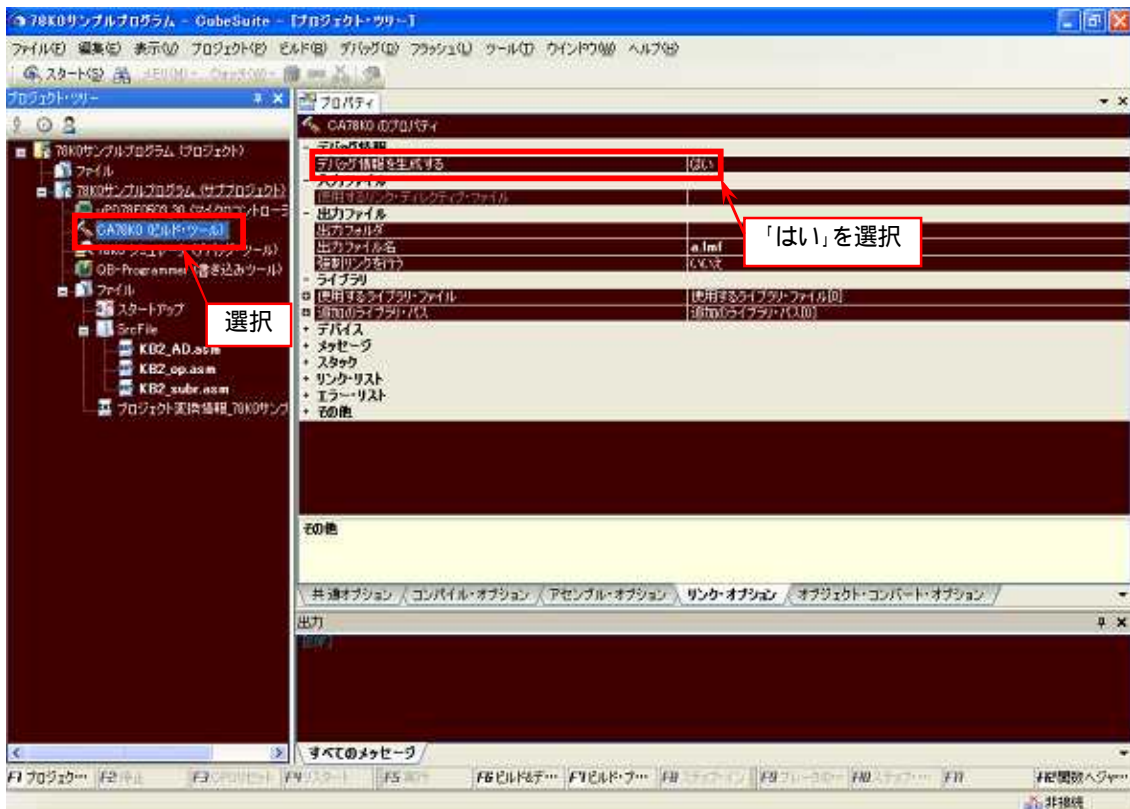
このコードでは、元のコードの2行目「DW IINIT ;」が「;DW IINIT」と変更され、また「TEST CSEG AT 0004H」が追加されています。赤い矢印とボックスは、これらの変更を強調しています。

C言語版では不要です。

## (3)リンクの設定

オン・チップ・デバッグを行う場合、ビルドの際、リンク・オプションの設定を行う必要があります。

CubeSuiteのプロジェクト・ツリーからCA78K0(ビルド・ツール)を選択し、リンク・オプションタブをクリックしてください。



デバッグ情報野の「デバッグ情報を生成する」を「はい」に変更してください。

以上で、リンク・オプションの設定は完了です。

## 第6章 関連資料

資料名		和文 / 英文
78K0/Kx2 ユーザーズ・マニュアル		<a href="#">PDF</a>
78K/0シリーズ 命令編 ユーザーズ・マニュアル		<a href="#">PDF</a>
RA78K0 アセンブラ・パッケージ ユーザーズ・マニュアル	言語編	<a href="#">PDF</a>
	操作編	<a href="#">PDF</a>
CC78K0 Cコンパイラ ユーザーズ・マニュアル	言語編	<a href="#">PDF</a>
	操作編	<a href="#">PDF</a>
Cube Suite キューブ・スイート ユーザーズ・マニュアル		<a href="#">PDF</a>
PM+ プロジェクト・マネージャ ユーザーズ・マニュアル		<a href="#">PDF</a>
SM+ システム・シミュレータ 操作編 ユーザーズ・マニュアル		<a href="#">PDF</a>

## 付録A 改版履歴

版数	発行年月	改版箇所	改版内容
第1版	Jan 2010	初版	-

## 【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）

電話：044-435-9494

E-mail：[info@necel.com](mailto:info@necel.com)

### 【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

E-mail：[toolsupport-micom@ml.necel.com](mailto: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約店へお申し付けください。

---