

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。

M16C/80 グループ

C 言語での DMA(DMA0, DMA1 のみ)設定手順(NC308WA の Ver5 以降対応の例)

1. 要約

この資料では、C 言語での DMA(DMA0, DMA1)の設定手順を説明し、応用例を掲載しています。

2. はじめに

この資料で説明する応用例は、次のマイコン、条件での利用に適用されます。

- ・マイコン : M16C/80 グループ
- ・M32C/80, M16C/80, M16C/70 シリーズ用
C コンパイラ バージョン : M3T-NC308WA V.5.00 Release 1

M3T-NC308WA V.5.00 Release 1 未満では、M16C/80 グループアプリケーションノート DMA 設定手順(ASM 関数使用の例)を参照してください。

M16C/80 グループと同様の SFR(周辺装置制御レジスタ)を持つ他の M16C ファミリでも本プログラムを使用することができます。ただし、一部の機能を機能追加等で変更している場合がありますのでマニュアルで確認してください。このアプリケーションノートをご使用に際しては十分な評価を行ってください。

3. 応用例の説明

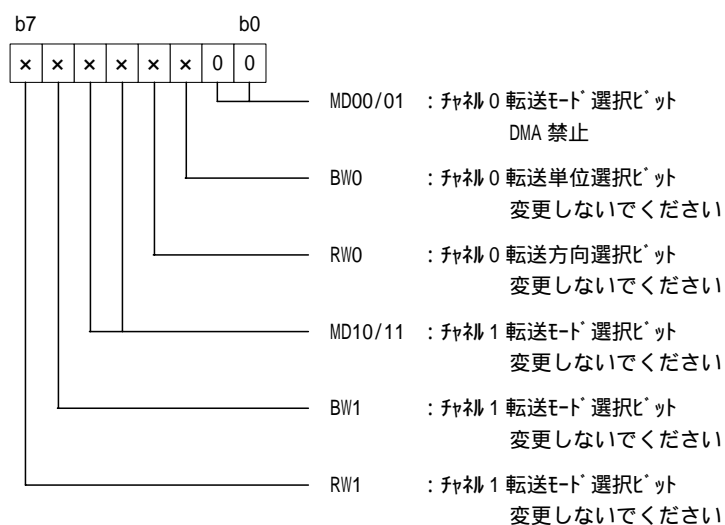
M32C/80, M16C/80, M16C/70 シリーズ用 C コンパイラ (M3T-NC308WA V.5.00 Release 1) では、
” #pragma DMAC ” を使用することにより、外部変数を DMAC 専用レジスタに割付けて、DMAC レジスタにアクセスできます。

3.1 設定方法

「3. 応用例の説明」を実現するための設定手順と設定値を DMA0 を例にして示します。各レジスタの詳細は、M16C/80 グループのデータシートを参照願います。

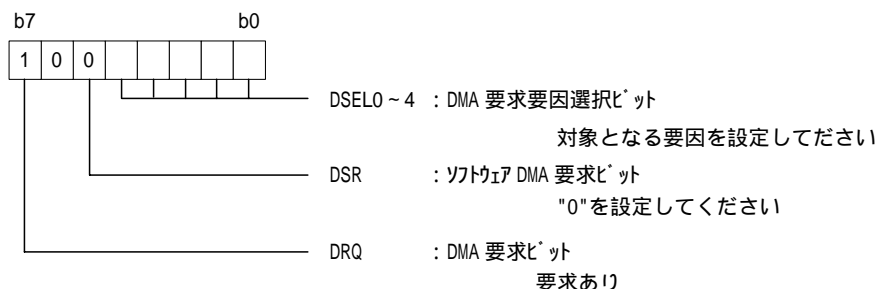
(1) DMA モードレジスタ 0(DMD0)を設定する。

DMA0 転送モード選択ビットを"00" (DMA 禁止)にする。

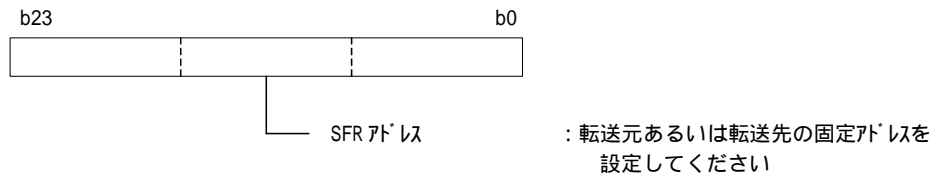


(2) DMA0 要因選択レジスタ (DMOSL)を設定する。

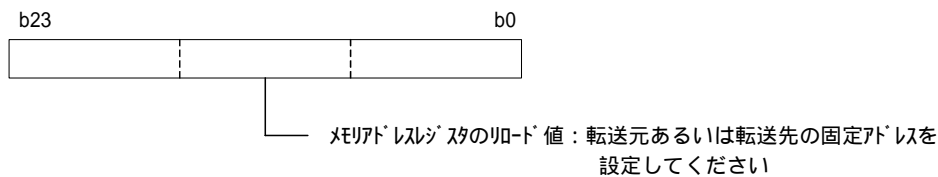
- DMA 要求要因選択ビットにて、DMA 転送のトリガとなる要因を設定する。
- DMA 要求ビットに"1" (要求あり)を設定する。



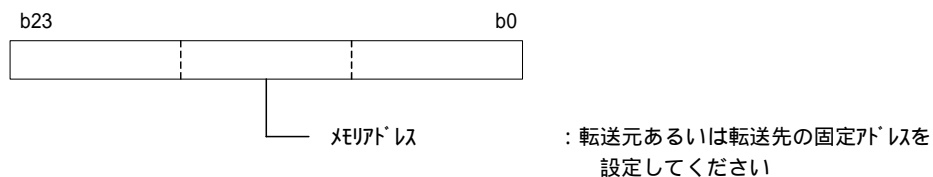
(3) DMA0SFR アドレスレジスタ(DSA0)を設定する。



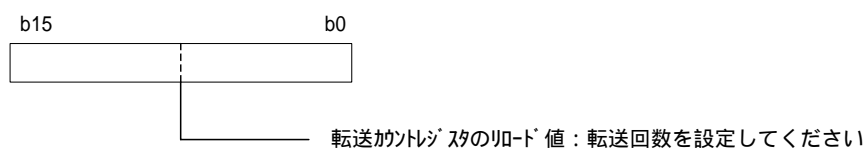
(4) DMA0 メモリアドレスリロードレジスタ(DRA0)を設定する。



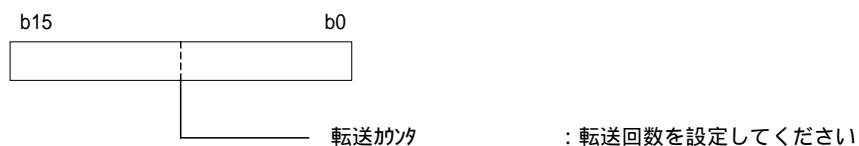
(5) DMA0 メモリアドレスレジスタ(DMA0)を設定する。



(6) DMA0 転送カウントリロードレジスタ(DRC0)を設定する。



(7) DMA0 転送カウントレジスタ(DCT0)を設定する。

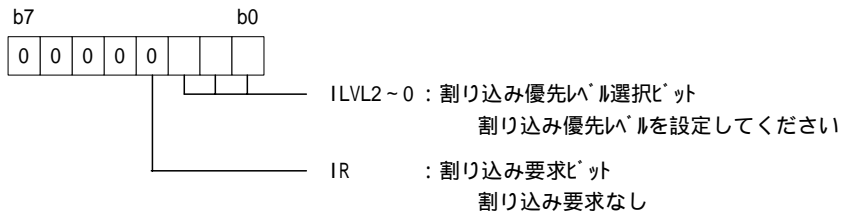


(8) ダミーサイクルを挿入する。

使用する DMA のチャネル数に応じて、下記の計算式で求めたサイクル数分 "NOP 命令" を挿入してください。

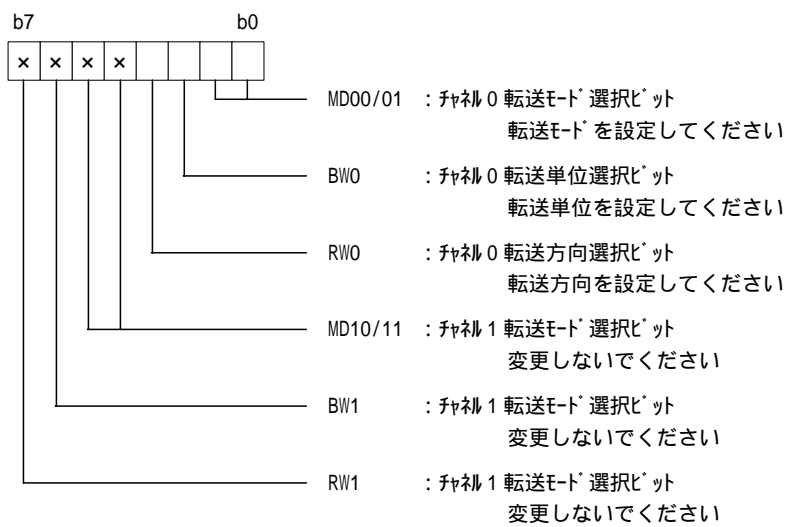
$$\text{ダミーサイクル数} = 8 + 6n \quad (n = \text{該当チャネル以外で DMA 要求要因が発生する可能性のある全てのチャネル数})$$

(9) DMA0 割り込み制御レジスタ(DMOIC)を設定する。



(10) DMA モードレジスタ 0(DMD0)を再設定する。

転送モード選択ビット、転送単位選択ビット、転送方向設定ビットを設定する。



(11) 割り込みを許可 (IFlag = "1") する。

(12) DMA0 要求要因となる周辺機能を開始する。

4. サンプルプログラム

DMA 要求要因 = "タイマ A1"、転送空間 = 16M バイトの任意の空間から固定アドレス、転送単位 = 8 ビット、転送モード = リピート転送とした場合のサンプルプログラムを以下に示します。

```

/*****/
/*  FILE NAME  :  rjj05b0288_src.c                               */
/*  Ver        :  1.00                                           */
/*  CPU        :  M16C/80                                         */
/*  FUNCTION   :  The DMA setting procedure in the C language.   */
/*-----*/
/*  Copyright(C)2003, Renesas Technology Corp.                   */
/*  Copyright(C)2003, Renesas Solutions Corp.                     */
/*  All rights reserved.                                          */
/*                                                                */
/*****/
/*****/
/*  include file                                                */
/*****/
#include "sfr80144.h" // Special Function Register Header File

/*****/
/*  #pragma declaration                                         */
/*****/
// CPU internal register

unsigned short  dmd0;
#pragma DMAC   dmd0   DMD0 // DMD0(DMA mode register)
unsigned short  dct0;
#pragma DMAC   dct0   DCT0 // DCT0(DMA0 transfer count register)
unsigned short  drc0;
#pragma DMAC   drc0   DRC0 // DRC0(DMA0 transfer count reload register)
void _far      *dma0;
#pragma DMAC   dma0   DMA0 // DMA0(DMA0 memory address register)
void _far      *dsa0;
#pragma DMAC   dsa0   DSA0 // DSA0(DMA0 SFR address register)
void _far      *dra0;
#pragma DMAC   dra0   DRA0 // DRA0(DMA0 memory address reload register)
void _far      *dma1;
#pragma DMAC   dma1   DMA1 // DMA1(DMA1 memory address register)
void _far      *dsa1;

```

```

#pragma DMAC   dsa1   DSA1           // DSA1(DMA1 SFR address register)
unsigned short dct1;
#pragma DMAC   dct1   DCT1           // DCT1(DMA1 transfer count register)
unsigned short drc1;
#pragma DMAC   drc1   DRC1           // DRC1(DMA1 transfer count reload register)
void _far      *dra1;
#pragma DMAC   dra1   DRA1           // DRA1(DMA1 memory address reload register)

unsigned short dmd1;
#pragma DMAC   dmd1   DMD1           // DMD1(DMA mode register1)

/*****/
/*   Function declaration           */
/*****/
void ta1_init(void);
void dma0_int(void);

/*****/
/*   Global variable declaration    */
/*****/
static char   data0[] = {0x1,0x3,0x7,0xf,0x1f,0x3f,0x7f,0xff};

/*****/
/*   main function                  */
/*****/
void main(void)
{
    short      dmd0_tmp;

    pd0 = 0xff;           // P0 is an output port.
    pd6 = 0xff;           // P6 is an output port.
    pd8 = 0x1f;           // P8_0-P8_4 are an output port.
    p0  = 0;              //
    p6  = 0;              //
    p8  = 0;              //

                                // A setup of DMA0

```



```

dmd0_tmp = dmd0;          // (1)DMA0 inhibit(DMD0)
dmd0_tmp &= 0x00fc;       //
dmd0      = dmd0_tmp;     //

ta1_init();              //
dm0sl = 0x84;            // (2)A setup of DMOSL(DRQ=1, TA1)
dsa0 = &p0;              // (3)A setup of DSA0(DMA0 SFR address register)
dra0 = &data0;          // (4)A setup of DRA0(DMA0 memory address reload register)
dma0 = &data0;          // (5)A setup of DMA0(DMA0 memory address register)
drc0 = 8;                // (6)A setup of DRC0(DMA0 transfer count reload register)
dct0 = 8;                // (7)A setup of DCT0(DMA0 transfer count register)
                        // (8)Dummy cycle insertion

asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");
asm("NOP      ");

dm0ic = 0x5;             // (9)A set of DMOIC(DMA0 interrupt control register)

dmd0_tmp |= 0x0B;       // (10)DMA0 permission(DMD0)  Repeat transfer(memory -> fixed
address)
dmd0 = dmd0_tmp;       //

asm("fset I    ");     // (11)Interruption permission.

ta1s = 1;               // (12)Operation of the circumference function of a DMA demand
factor is permitted.

```

```

    while(1)
    {
        p8_1 = 1;           // test end...
    }

}

/*****/
/*  Timer-A1 initialization          */
/*****/
void ta1_init(void)
{
    ta1mr = 0x80;          // Timer-mode(f32)
    ta1   = 0x8fff;       // Timer value setup
}

/*****/
/*  DMA0 interrupt routine          */
/*****/
#pragma INTERRUPT/B dma0_int
void dma0_int(void)
{
    p6_0 = !p6_0;
}

```

5. 参考ドキュメント

データシート

M16C/80 グループデータシート

(最新版をルネサス テクノロジホームページから入手してください。)

6. ホームページとサポート窓口

ルネサス テクノロジホームページ

<http://www.renesas.com/>

M16C ファミリ MCU 技術サポート窓口

E-mail: support_apl@renesas.com

7. 改定記録

| Rev. | 発行日 | 改訂内容 | |
|------|-----------|------|------|
| | | ページ | ポイント |
| 1.00 | 2002.8.29 | - | 初版発行 |
| | | | |

安全設計に関するお願い

1. 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故、火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご留意ください。

本資料ご利用に際しての留意事項

1. 本資料は、お客様が用途に応じた適切なルネサス テクノロジー製品をご購入いただくための参考資料であり、本資料中に記載の技術情報についてルネサス テクノロジーが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、ルネサス テクノロジーは責任を負いません。
3. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、ルネサス テクノロジーは、予告なしに、本資料に記載した製品または仕様を変更することがあります。ルネサス テクノロジー半導体製品のご購入に当たりましては、事前にルネサス テクノロジー、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジーホームページ(<http://www.renesas.com>)などを通じて公開される情報に常にご注意ください。
4. 本資料に記載した情報は、正確を期すため、慎重に制作したのですが万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジーはその責任を負いません。
5. 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。ルネサス テクノロジーは、適用可否に対する責任を負いません。
6. 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジー、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジーの事前の承諾が必要です。
8. 本資料に関し詳細についてのお問い合わせ、その他お気づきの点がございましたらルネサス テクノロジー、ルネサス販売または特約店までご照会ください。