AVR1502 : Xplain練習 – 直接ﾒﾓﾘ入出力制御器

前提条件

- 必要な知識
  - AVR1500:Xplain練習 – XMEGA®基礎の完了
- ｿﾌﾄｳｪｱ必要条件
  - ATMEL® AVR® Studio® 4.18またはそれ以降
  - WinAVR/GCC 20100110またはそれ以降
- ﾊｰﾄﾞｳｪｱ必要条件
  - JTAGICEmk Ⅱ
  - Xplain評価基板
- 予想完了時間
  - 2時間

1. 序説

この応用記述はATMEL XMEGAの直接ﾒﾓﾘ入出力制御器(DMAC)の基本的な機能を網羅します。この練習の目的はCPU時間を殆ど使用することなく簡単なﾒﾓﾘ転送を開始させ、そして殆どどんなCPUの介在もなく周辺機能に読み書きすることです。

個別の転送元、転送先、起動、塊容量を持つ4つのDMAﾁｬﾈﾙがあります。DMA制御器は或るﾒﾓﾘ領域から別のﾒﾓﾘ領域、ﾒﾓﾘと周辺機能間、周辺機能間でﾃﾞｰﾀを移動できます。

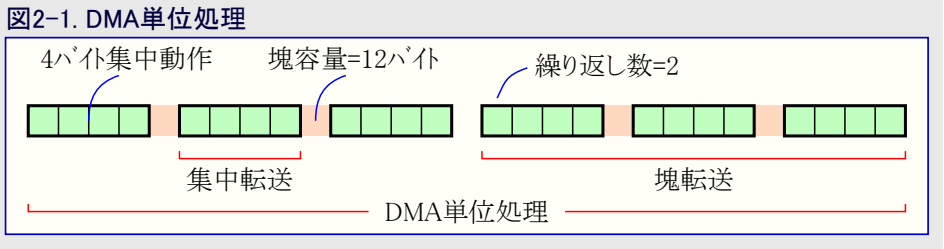
2. DMA制御器の構成設定

ATMEL XMEGAの直接ﾒﾓﾘ入出力制御器(DMAC)は最小のCPU介在でﾒﾓﾘと周辺機能間でﾃﾞｰﾀを転送する能力を持つ高い柔軟性の4ﾁｬﾈﾙDMA制御器です。CPUが低電力休止形態で時を費やす、または他の作業を実行している間に、XMEGAのDMACは或る領域から別の領域へ単にﾃﾞｰﾀの複写を処理することによってCPUの負荷を無くします。以降の副節はこの練習に於いて課題を行う時の理解を補完することを目論まれています。

2.1. DMA単位処理

ﾒﾓﾘや周辺機能間での完全なDMA読み書き操作はDMA単位処理と呼ばれます。単位処理はﾃﾞｰﾀの塊で行われ、単位処理の量(転送ﾊﾞｲﾄ数)はｿﾌﾄｳｪｱから選択可能で塊量と繰り返し計数器の設定によって制御されます。各塊転送はより小さな集中(転送)に分割されます(図2-1.をご覧ください)。

図2-1. DMA単位処理



DMAﾁｬﾈﾙがﾃﾞｰﾀ転送を要求すると、ﾊﾞｽ調停器はAVRｺｱがﾃﾞｰﾀ ﾊﾞｽを使用しないようになるまで待ってDMA制御器にﾃﾞｰﾀの転送を許します。転送は1,2,4または8ﾊﾞｲﾄの集中(転送)で行われます。ｱﾄﾞﾚｽ指定は静止、増加、減少にできます。各集中転送、塊転送後、転送完了時、または禁止時に、転送元や転送先のｱﾄﾞﾚｽ自動再設定を行うことができます。DMA転送は応用ｿﾌﾄｳｪｱ、周辺機能、それと事象によって起動することができます。

塊転送の容量は塊転送計数ﾚｼﾞｽﾀによって設定され、1ﾊﾞｲﾄから64Kﾊﾞｲﾄまでのどれにでもできます。繰り返し計数器は単位処理が完了する前の繰り返し塊転送数の設定を許すことができます。繰り返しは1から255までで、繰り返し回数を0に設定することによって無制限の繰り返し回数を達成することができます。

ﾊﾞｽ調停器はDMA制御器とAVRｺｱがﾊﾞｽを使用できる時を制御します。ｺｱが常に優先権を持ち、故にｺｱがﾊﾞｽへのｱｸｾｽを要求する限り、どの保留集中転送も待たなければなりません。ｺｱはSRAM、I/Oﾒﾓﾘ、EEPROM、外部ﾊﾞｽ ｲﾝﾀｰﾌｪｰｽとﾃﾞｰﾀを読み書きする命令を実行する時にﾊﾞｽのｱｸｾｽを要求します。

8ﾋﾞｯﾄ AVR® マイクロ コントローラ

応用記述

Rev. 8310A-06/10, 8310AJ1-03/14

2.2. ｱﾄﾞﾚｽ指定

転送元や転送先がSRAMの場合、使用者は多分、ｱﾄﾞﾚｽ指示子の増加または減少を望みそうです。従って、増加または減少の形態でDMACの転送元/転送先を設定することが可能です。

DMACは増加動作形態でｱﾄﾞﾚｽ指示子を自動的に増加します。

SRAMｱﾄﾞﾚｽ	値
$2066	34
$2067	76
$2068	13
$2069	113

増加形態ではDMACが転送元ｱﾄﾞﾚｽで始まります。最初のﾊﾞｲﾄが送られてしまった後、次のﾊﾞｲﾄが直前のｱﾄﾞﾚｽ+1から読まれます。

減少形態ではこれが逆で、DMACは与えられた転送元ｱﾄﾞﾚｽで始まり、前(-1)のｱﾄﾞﾚｽで進みます。

元の転送元と転送先のｱﾄﾞﾚｽはDMA制御器によって保存され、故に転送元と転送先のｱﾄﾞﾚｽは以下の時点で再設定するように個別に形態設定できます。

- 各集中転送の最後
- 各塊転送の最後
- 転送処理単位の最後
- 再設定なし

SPIのような周辺機能でDMACを使用する時はﾃﾞｰﾀ ﾚｼﾞｽﾀが固定で、ｱﾄﾞﾚｽ指定形態が静止であることが重要です。

2.3. 転送の起動

DMA起動の主な目的は転送速度で周辺機能を同期することです。例えば、USARTを用いて9600bpsで転送する時に、DMAは大抵転送完了ﾌﾗｸﾞの設定(1)で起動されるべきではありません。

DMA転送はDMA転送要求が検出された時にだけ開始することができます。転送要求はｿﾌﾄｳｪｱ、外部起動元(周辺機能)、または事象から起動することができます。各DMAﾁｬﾈﾙに対して専用の起動元選択があります。利用可能な起動元はﾃﾞﾊﾞｲｽ毎に変わるかもしれず、ﾃﾞﾊﾞｲｽに存在する部署や周辺機能に依存します(各種転送起動元についてはXMEGA手引書をご覧ください)。

2.4. 割り込み

DMA制御器はDMAﾁｬﾈﾙで異常が検出された時、またはDMAﾁｬﾈﾙに関して転送が完了した時に割り込みを生成することができます。各DMAﾁｬﾈﾙは独立した割り込みﾍﾞｸﾀを持ち、異常と転送処理単位完了に対して異なる割り込みﾌﾗｸﾞがあります。繰り返しが許可されていなければ、塊転送の最後で転送処理単位完了ﾌﾗｸﾞが設定(1)されます。無制限繰り返しが許可されている場合、各塊転送の最後でも転送処理単位完了ﾌﾗｸﾞが設定(1)されます。

3. 概要

ここはこの練習に於ける課題の短い概要です。

課題1. ﾒﾓﾘ複写

この課題はCPUの介在なしに或る位置から別の位置にﾒﾓﾘの塊を複写する方法を示します。この課題は以下のように2つの部分に分けられます。

1. 最初の部分はDMACでSRAMのﾒﾓﾘを複写する方法を示します。
2. 2つ目の部分は転送が終わったことを示すように割り込みを構成設定する方法を示します。

課題2. 記録器

この課題は10秒間のｽｲｯﾁ押下の流れを記録してLEDでそれらを再生する簡単な記録器の実装方法を示します。記録と再生の処理はCPUの介在なしで実行されます。

4. 課題1: ﾒﾓﾘ複写

DMA制御器は或る位置から別の位置へﾒﾓﾘの大きな塊を複写するのに理想的です。この課題はﾄﾞﾗｲﾊﾞ ﾌｧｲﾙからの塊ﾒﾓﾘ複写ﾙｰﾁﾝを実演します。転送が終わった時に割り込みを得るためにPMICを構成設定する方法も示します。

この課題の目的はあなたが以下の方法を知ることです。

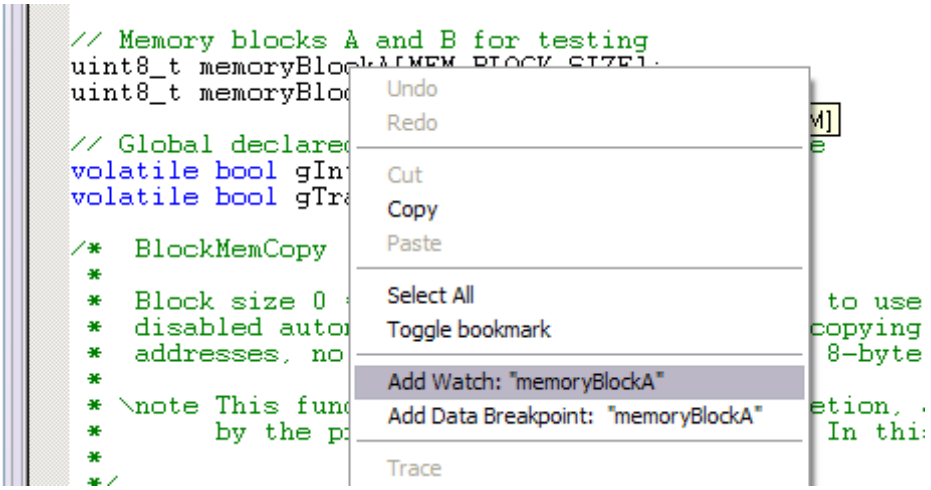
- DMACを構成設定するためにﾄﾞﾗｲﾊﾞ ｺｰﾄﾞを用いてﾒﾓﾘ複写操作を開始
- 転送が終わった時に割り込みを起動するようにPMICを構成設定

4.1. ﾒﾓﾘ複写: ﾎﾟｰﾘﾝｸﾞ手法

すること

1. AVR StudioでMemoryCopy.apsﾌﾟﾛｼﾞｪｸﾄ ﾌｧｲﾙを開き、task1.cﾌｧｲﾙを見てください。
2. ﾌﾟﾛｼﾞｪｸﾄを構築(F7を押)してﾃﾞﾊﾞｯｸﾞを始めて(Play釦をｸﾘｯｸして)ください。
3. memoryBlockAとmemoryBlockBの定義を探し出してそれらの両方に対してﾃﾞｰﾀ監視(Watch)を追加(右ｸﾘｯｸでのﾒﾆｭｰで"Add watch..."をしてください(図4-1.をご覧ください)。

図4-1. task1.cの先頭でのﾒﾓﾘ塊位置指示



4. main()でMemCopyに対する呼び出し上で右ｸﾘｯｸして右ｸﾘｯｸでのﾒﾆｭｰで"Run to Cursor"を選んでください(図4-2.をご覧ください)。
5. 監視(Watch)ｳｨﾝﾄﾞｳでmemoryBlockAとmemoryBlockBを展開してそれらの内容を調べてください。それらは等しいですか?(図4-3.をご覧ください)。

図4-2. ｶｰｿﾙ位置まで実行

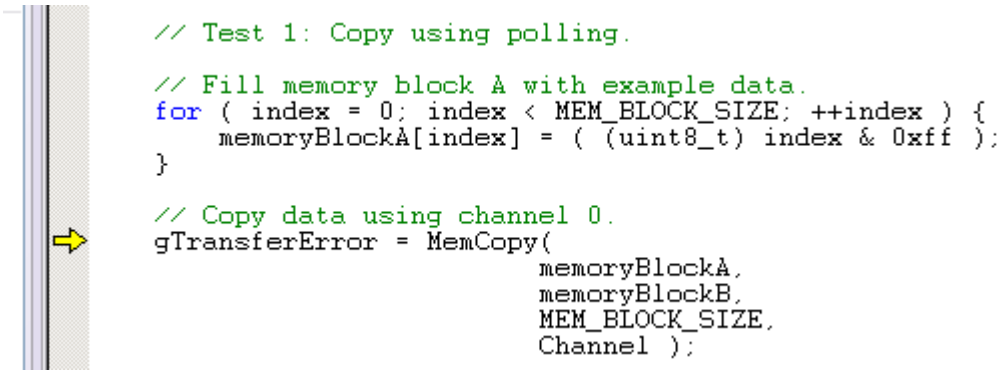
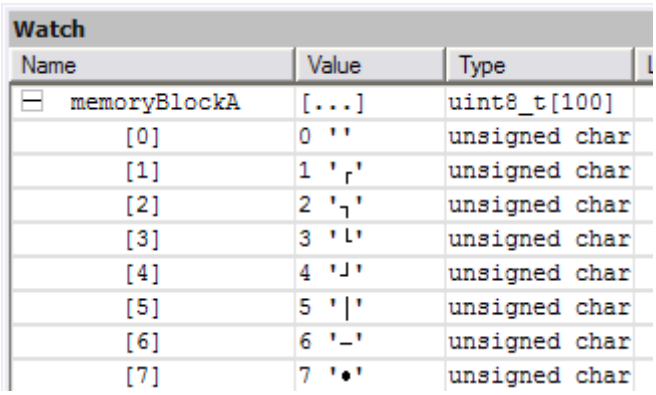


図4-3. memoryBlockAとmemoryBlockBの比較



6. MemCopy呼び出しを外側実行(Step over)して再びmemoryBlockAとmemoryBlockBの内容を調べてください。今はそれらが等しいですか?。
7. 実行をﾘｾｯﾄ(Shift+F5)して再びMemCopyを走らせてください。今回はこの関数をｱｸｾｽするのに内側実行(Step into: F11押下)を行ってください。更に、ﾄﾞﾗｲﾊﾞがDMACをどう形態設定するかを見るためにDMA_SetupBlockも内側実行してください。これらの問いを答えるためにあなたは導入節の参照を望むかもしれません。
   a) ﾒﾓﾘ複写は静止、増加、減少のどれとして構成設定されますか?。何故このように構成設定されるのでしょうか?。
   b) 別のDMAﾁｬﾈﾙを使用することができましたか?。
   c) このﾒﾓﾘ間転送に関して既定としてｿﾌﾄｳｪｱ起動元が使用されます。SRAMからSRAMへの複写時に何故これが便利と考えますか?。
   d) 集中転送長はいくつですか?。何故これが形態設定可能と考えますか?。

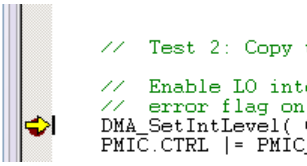
4.2. メモリ複写: 割り込み手法

課題:

1. 図4-4.で示されるように、試験2の開始へ中断点を追加してください。そして走行するためにF5を押してください。

すること

図4-4. ここに中断点追加



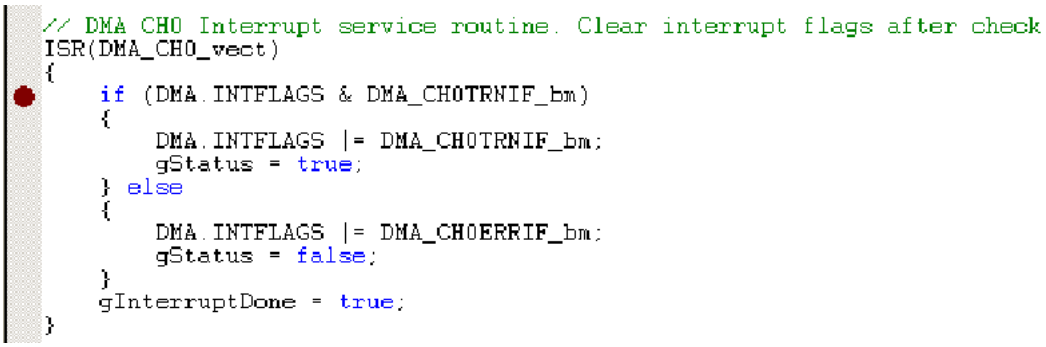
```
//  Test 2: Copy using interrupts.

//  Enable LO interrupt level for the complete transaction and
//  error flag on DMA channel 0.
DMA_SetIntLevel( Channel, DMA_CH_TRNINTLVL_LO_gc, DMA_CH_ERRINTLVL_LO_gc );
PMIC.CTRL |= PMIC_LOLVLEN_bm;
```

2. 直前の試験が快調(転送で異常なし)なら、次の試験の準備を整えます。いくつかの段階を1行実行(F10押下)し、次のDMA転送単位処理完了時に割り込みが起動するように構成設定されることに気をつけてください。
3. AsyncMemCopy関数を考察してください。MemCopyと比べて、DMA_ReturnStatus_non_blocking呼び出しがないことに注意してください。

DMA_ReturnStatus_non_blocking関数はDMA転送が終わるまでポーリングします。転送終了を示す代わりに割り込みを使用する時に、CPUは他の操作を行うように自由になります。

4. task1.cの下部に於いて、図4-5.で示されるように、ISR(DMA_CH0_vect)割り込み処理ルーチンで中断点を追加してください。走行するためにF5を押してください。

図4-5. 割り込み待機



```
// DMA CH0 Interrupt service routine. Clear interrupt flags after check
ISR(DMA_CH0_vect)
{
    if (DMA.INTFLAGS & DMA_CH0TRNIF_bm)
    {
        DMA.INTFLAGS |= DMA_CH0TRNIF_bm;
        gStatus = true;
    } else
    {
        DMA.INTFLAGS |= DMA_CH0ERRIF_bm;
        gStatus = false;
    }
    gInterruptDone = true;
}
```

5. 完了するために走行(F5を押)し、暫くして実行を中断(Ctrl+F5)してください。memoryBlockAとmemoryBlockBを比べることによって転送が上手く行われたことを検証してください。

5. 課題2: 記録器

DMACが実際に有用な応用はADCやDACのような周辺機能とメモリ間でデータを複写することです。この課題は或る時間の間スイッチを採取し、SRAM緩衝部にそれを格納し、その後に何度も何度もLEDでデータを再生する簡単な記録器の実装方法を示します。ADCで音を記録してDACでそれを再生する場合、正確に同じ原理が用いられます。

この課題の目的はあなたが以下の方法を知ることです。

- 周辺機能からSRAMへデータを複写するようにDMACを形態設定
- SRAMから周辺機能へデータを複写するようにDMACを形態設定
- DMAデータ転送用起動元として計時器の計時を使用

すること

1. ATMELのAVR Studioで課題2フォルダのDMA_Recorder.apsプロジェクト ファイルを開いてtask2.cを考察してください。
2. プロジェクトを構築(F7を押)し、デバッグを開始(Play釦をクリック)して、そして記録器を使用するためにコードを走らせて(F5を押して)ください。

a) 一度LEDが発光します。

b) どれかスイッチを押してください。

c) LEDが再び発光します。

d) DMACがスイッチの状態を記録する間にスイッチを叩くのに、概ね10秒あります。

e) LEDが再び発光します。

f) どれかスイッチを押してください。

g) LEDが再び発光します。

h) DMACはスイッチが押されるまでLED上でスイッチの記録を再生します。

i) この手順をやり直してください。

3. SAMPLE_COUNT定義または採取速度を変更してみてください。再ｺﾝﾊﾟｲﾙして走らせて、何が起きるかを見てください(**図5-1.**をご覧ください)。

図5-1. SAMPLE_COUNT変更試行

```
#define LEDPORT PORTD
#define SWITCHPORT PORTC

#define SAMPLE_COUNT 300 // How many switch state samples to store.
uint8_t samples[SAMPLE_COUNT]; // Store switch state samples here.


// The DMA channel is triggered by the Timer Overflow Interrupt Flag.
// and will be continuously re-triggered as long as the flag is set.
```

4. 何が起きたかを理解するためにｺｰﾄﾞを簡単に片付けてみてください。これらの問いに答える時に導入節を参照するかもしれません。

a) SetupReadChannelに於いて、何が転送元で何が転送先ですか?。

b) SetupWriteChannelに於いて、何が転送元で何が転送先ですか?。

c) ｿﾌﾄｳｪｱ起動元でDMACを構成設定したなら、その場合は何が起きますか?。

d) 読み込みﾁｬﾈﾙに関して、何故転送元ｱﾄﾞﾚｽが固定で転送先ｱﾄﾞﾚｽが増加なのですか?。

e) 集中(転送)長は4や8のような違う長さにできますか?。

6. 要約

- ﾒﾓﾘ複写、SRAMからSRAMへのDMAC。
- 転送の最後を示すための割り込み元の使用。
- DMACを用いたｽｲｯﾁ押下記録。この応用はﾎﾟｰﾄ ﾋﾟﾝの入力状態を記録してSRAMにその情報を保存しました。SRAMに保存された情報はその後にLEDに出力設定されました。起動元として計時器溢れが使用されました。

7. 資料

- XMEAの手引書とﾃﾞｰﾀｼｰﾄ
  - ○ http://www.atmel.com/xmega
- ATMELのﾍﾙﾌﾟ ﾌｧｲﾙ付きAVR studio
  - ○ http://www.atmel.com/products/AVR/
- WINAVR GCCｺﾝﾊﾟｲﾗ
  - ○ http://winavr.souceforge.net/
- ATMEL用IAR Embedded Workbench®ｺﾝﾊﾟｲﾗ
  - ○ http://www.iar.com/

8. ATMEL技術支援ｾﾝﾀｰ

ATMELは以下の利用可能な多数の支援ﾁｬﾈﾙを持ちます。

・ｳｪﾌﾞ入り口	: http://www.atmel.no/	全てのATMELﾏｲｸﾛ ｺﾝﾄﾛｰﾗ
・Eﾒｰﾙ	: avr@atmel.com	全てのATMEL AVR製品
・Eﾒｰﾙ	: avr32@atmel.com	全ての32ﾋﾞｯﾄAVR製品

以下のｻｰﾋﾞｽへのｱｸｾｽを得るにはｳｪﾌﾞ入り口で登録してください。

- 豊富なFAQﾃﾞｰﾀﾍﾞｰｽへのｱｸｾｽ
- 技術支援要請の容易な依頼
- あなたの過去の全支援要請の履歴
- ATMELﾏｲｸﾛ ｺﾝﾄﾛｰﾗ時事通信の受信のための登録
- 利用可能な練習と練習材料についての情報取得

本社

Atmel Corporation
2325 Orchard Parkway
San Jose, CA 95131
USA
TEL 1(408) 441-0311
FAX 1(408) 487-2600

国外営業拠点

Atmel Asia
Unit 1-5 & 16, 19/F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
Hong Kong
TEL (852) 2245-6100
FAX (852) 2722-1369

Atmel Europe
Le Krebs
8, Rue Jean-Pierre Timbaud
BP 309
78054 Saint-Quentin-en-
Yvelines Cedex
France
TEL (33) 1-30-60-70-00
FAX (33) 1-30-60-71-11

Atmel Japan
104-0033 東京都中央区
新川1-24-8
東熱新川ﾋﾞﾙ 9F
ｱﾄﾒﾙ ｼﾞｬﾊﾟﾝ株式会社
TEL (81) 03-3523-3551
FAX (81) 03-3523-7581

製品窓口

ｳｪﾌﾞｻｲﾄ
www.atmel.com

技術支援
avr@atmel.com

販売窓口
www.atmel.com/contacts

文献請求
www.atmel.com/literature

本応用記述はATMELのAVR1502応用記述(doc8310.pdf Rev.8310A-06/10)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意訳されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

必要と思われる部分には( )内に英語表記や略称などを残す形で表記しています。

青字の部分はﾘﾝｸとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。