

AVR1313 : XMEGA I/Oピンと外部割り込みの使い方

要点

- 柔軟なピン形態設定
- 同期(と/または)非同期の入力感知
- 非同期起動の合図
- 形態設定可能な出力駆動部と引き込み設定:
 - ・ コンプリメンタリ
 - ・ ワイヤードAND
 - ・ ワイヤードOR
 - ・ プルアップまたはプルダウン
 - ・ バス保持
 - ・ 反転入出力
- スリューレート制御
- 単一操作での複数ピン形態設定
- 出力と方向のレジスタ用の仮想の設定/解除/切り換えレジスタ
- 高速アクセス用のI/O空間の仮想ポートへのポート割り当て

1. 序説

本応用記述は高い形態設定能力のXMEGA®のI/Oピンと外部割り込みを使用するための導入を与えます。

2. 部署概要

本節はI/Oポート部署の概要を与え、利用可能な様々な形態設定任意選択を記述します。

2.1. 命名

XMEGAのI/Oピンは8ピンのI/Oポート内で群化されています。I/OポートはPORT x と命名され、ここでの x は文字で、例えばPORTA, PORTBなどです。ポート内の個別ピンはP xn と命名され、ここでの x はポート文字で、 n はピン番号を識別し、例えばPA0, PA1などです。

いくつかのレジスタはポート内の全ピンに影響を及ぼします。これらのレジスタではビット n がピン n に対応します。別の言葉では、最下位ビットがピン0に対応し、一方最上位ビットがピン7に対応します。

2.2. 基本的な形態設定と使い方

本項は形態設定任意選択とI/Oポートピンの基本的な機能を使うための必要なレジスタを記述します。

2.2.1. ポートピンの方向設定

ポートピンの方向はDIRレジスタを通じて制御されます。ポートピン n はDIRレジスタのDIR n ビットが'1'の場合に出力として形態設定されます。DIR n ビットが'0'なら、そのポートピンは入力として形態設定されます。

DIRレジスタは直接的に、またはDIRSET, DIRCLR, DIRTGLのストローブ(訳補:目的を起動する瞬間信号)レジスタを通してアクセスすることができます。DIRSETへのビット遮蔽値書き込みはDIRレジスタ内でビット遮蔽値(のビット)を設定(1)します。DIRCLRへのビット遮蔽値書き込みはDIRレジスタ内でビット遮蔽値(のビット)を解除(0)します。DIRTGLへのビット遮蔽値書き込みはDIRレジスタ内でビット遮蔽値のビットを切り替えさせます。

2.2.2. ポートピンの出力値制御

ポートピンの出力値はOUTレジスタを通じて制御されます。ポートピンの方向はどの効果も持つように、対応するOUTレジスタビットに関して出力に設定されなければなりません。ポートピン n はOUTレジスタのOUT n ビットが'1'の時、Highに駆動されます。OUT n が'0'の時はポートピンがLowに駆動されます。これは出力形態設定が変更されていない、ピン反転ビットが設定されていないとの仮定です。

OUTレジスタは直接的に、またはOUTSET, OUTCLR, OUTTGLのストローブレジスタを通してアクセスすることができます。OUTSETへのビット遮蔽値書き込みはOUTレジスタ内でビット遮蔽値(のビット)を設定(1)します。OUTCLRへのビット遮蔽値書き込みはOUTレジスタ内でビット遮蔽値(のビット)を解除(0)します。OUTTGLへのビット遮蔽値書き込みはOUTレジスタ内でビット遮蔽値のビットを切り替えさせます。



8ビット AVR®
マイクロコントローラ

応用記述

本書は一般の方々の便宜のため有志により作成されたもので、ATMEL社とは無関係であることを御承知ください。しおりのはじめにでの内容にご注意ください。

Rev. 8050A-02/08, 8050AJ4-03/14

2.2.3. ポートピンの論理状態読み込み

ポートピンの論理状態はINレジスタを読むことによって得られます。ポートピンの現在の論理状態は方向設定に拘らず、INビットを通して読むことができます。

2.3. ピン形態設定

各I/OピンはPINnCTRLと名付けられた自身の形態設定レジスタを持ち、ここでのnはピン番号を指示します。これらのレジスタを通して、いくつかの要素を基本的にピン単位で形態設定することができます。これらの要素は本項で説明されます。

2.3.1. 出力/引き込み形態設定

出力/引き込み形態設定ビットは出力駆動部動作形態設定と引き込み形態設定に使用されます。表2-1.は可能な形態設定要素概要を示します。

■ 出力形態

- “コンプリメンタリ”形態では、対応するOUTレジスタのビットによる指令として出力がVCCまたはGNDのどちらかに強く駆動されます。

- “ワイヤードOR”形態では、OUTnへの’1’書き込みがピンをVCCに強く駆動させます。OUTnへの’0’書き込みはピンを開放し、内部または外部のプルダウン抵抗でのGNDへの引き込みを許します。

- “ワイヤードAND”形態では、OUTnへの’0’書き込みがピンをGNDに強く駆動させます。OUTnへの’1’書き込みはピンを開放し、内部または外部のプルアップ抵抗でのVCCへの引き込みを許します。

■ 引き込み形態

- “入出力でのバス保持”形態はピンがどの論理状態へも駆動されない時にピンを同じ論理レベルに保つ弱いバス保持器を提供します。

- “プルダウン”形態はピンに対する内部プルダウン抵抗を許可します。

- “プルアップ”形態はピンに対する内部プルアップ抵抗を許可します。

2.3.2. 入力/感知形態設定

入力/感知形態設定ビットはI/Oピンのデジタル入力緩衝部と入力感知を制御します。可能な入力/感知形態設定は表2-2.で一覧にされます。

4つの感知形態が利用可能です。割り込みと事象である結果はピンから生成されます。ピンは両端、上昇端、下降端、またはLowレベルで感知することができます。Highレベルの感知がPINnCTRLの“反転I/O”ビットの設定(1)によって達成することに注意してください。ピンが事象生成に使用され、そして感知設定がLowレベルに設定される場合、そのピンは事象システムに対して透過にされ、ピンのレベルが事象線へ直接的に接続されることを意味します。

入力/感知領域の“デジタル入力緩衝部禁止”設定はピンのデジタル入力緩衝部を禁止します。これはピンが未使用、またはアナログ機能に対してだけ使用される時の消費電力削減に使用することができます。INレジスタの対応ビットはデジタル入力緩衝部が禁止された時、常に’0’を読みます。

2.3.3. 反転

PINnCTRLレジスタの“反転I/O”ビットはピンの極性を制御します。このビットが1を書かれた場合、本応用記述の記述に関して全ての入出力論理が反転されます。

ピンに対する“反転I/O”設定はピンを無効化するなどの周辺機能部署に対する入出力信号も反転します。例としてタイマ/カウンタ部署からのPWM出力信号は単に“反転I/O”ビットを設定(1)することによって反転することが可能です。

2.3.4. スルーレート制御

PINnCTRLの“スルーレート制御”ビットへの’1’書き込みはそのI/Oピンに対するスルーレート制限を許可します。これはポートピンでの論理レベルの切り換えによって起こるEMC問題を減らすのに使用することができます。スルーレート制限の特性についての情報に関してはデバイスのデータシートを参照してください。

表2-1. 出力/引き込み形成

| シンボル | 出力形態 | 引き込み形態 |
|--------------------------|----------|-----------|
| PORT_OPC_TOTEM_gc | コンプリメンタリ | (なし) |
| PORT_OPC_BUSKEEPER_gc | コンプリメンタリ | 入出力でのバス保持 |
| PORT_OPC_PULLDOWN_gc | コンプリメンタリ | 入力でのプルダウン |
| PORT_OPC_PULLUP_gc | コンプリメンタリ | 入力でのプルアップ |
| PORT_OPC_WIREDOR_gc | ワイヤードOR | (なし) |
| PORT_OPC_WIREDAND_gc | ワイヤードAND | (なし) |
| PORT_OPC_WIREDORPULL_gc | ワイヤードOR | プルダウン |
| PORT_OPC_WIREDANDPULL_gc | ワイヤードAND | プルアップ |

表2-2. 入力/感知形態

| シンボル | 形態 |
|---------------------------|---------------------|
| PORT_ISC_BOTHEDGES_gc | 両端感知 |
| PORT_ISC_RISING_gc | 上昇端感知 |
| PORT_ISC_FALLING_gc | 下降端感知 |
| PORT_ISC_LEVEL_gc | Lowレベル感知 (事象に対して透過) |
| PORT_ISC_INPUT_DISABLE_gc | デジタル入力緩衝部禁止 |

2.4. ポート割り込みの形態設定と使い方

I/Oポート割り込みはピン変化またはピンレベルでの割り込み生成と休止形態からデバイスを起すのに使用することができます。本項はI/Oポート割り込みの概要とその使用方法を与えます。

2.4.1. ポート割り込みの形態設定

XMEGAの各I/Oポートは2つの割り込みを持っています。それはI/Oポートのピンの任意の組み合わせによって起動されるように各割り込みを割り当てることが可能です。

ピン割り込みの初期設定は3つの段階で行われ、この例は割り込み0です。

1. 割り込みを起動し得る各ピンに対してPINnCTRLレジスタの入力/感知部分を形態設定してください。
2. 割り込みを起動し得るピンに対するビット遮蔽値をINTOMASKレジスタに書いてください。
3. INTLVLレジスタのINT0LVL部分を設定することによって割り込み優先権レベルを選択してください。

割り込み処理部が実行されるために、選択した割り込みレベルが設定可能な多段割り込み制御器(PMIC)で許可され、そして全体割り込み許可ビットが設定(1)されなければならないことに注意してください。XMEGAでの割り込みのより多くの情報に関してはAVR1305応用記述をご覧ください。

2.4.2. 非同期感知

XMEGAの非同期支援には2つのレベルがあります。全てのポートのピン2は完全な非同期支援を持ち、一方他のピンは制限された非同期支援を持ちます。完全と制限された非同期感知動作の要約が表2-3.と表2-4.で一覧にされます。

表2-3. 完全な非同期感知支援 (ピン2)

| 感知設定 | 支援有無 | 起動後の割り込み |
|--------|------|------------------------|
| 上昇端 | 有 | 常に起動 |
| 下降端 | 有 | 常に起動 |
| 両端 | 有 | 常に起動 |
| Lowレベル | 有 | ピン値は無変化を維持されなければなりません。 |

表2-4. 制限された非同期感知支援 (ピン2を除く全ポートピン)

| 感知設定 | 支援有無 | 起動後の割り込み |
|--------|------|------------------------|
| 上昇端 | 無 | - |
| 下降端 | 無 | - |
| 両端 | 有 | ピン値は無変化を維持されなければなりません。 |
| Lowレベル | 有 | ピン値は無変化を維持されなければなりません。 |

2.4.3. 休止形態からの起動に対するポート割り込みの使い方

どのポートピンも休止形態からXMEGAを起動するのに使用することができます。けれども、起動に使用されるピンの非同期感知支援レベルが、どの感知設定が使用できるのかと、起動後の割り込み要求フラグの状態を決めます。

表2-3.と表2-4.は完全と制限された非同期感知ピンに対して支援される非同期感知設定を示します。起動に使用するピンはデバイスで起動できる非同期感知によって支援される設定の1つでなければなりません。

表2-3.と表2-4.はデバイス起動後に設定(1)されるべき割り込み要求フラグが合致しなければならない条件も示します。

2.5. 効率的なI/Oポートの使い方

XMEGA系で利用可能な2つの機能はコード量削減と実行速度増加を行えます。それは複数ピン形態設定と仮想ポートです。

2.5.1. 複数ピン形態設定

各ポートピンに対して1つの形態設定レジスタを持つことは柔軟性を増しますが、一度に1つでの全てのピンの形態設定は沢山のコードを必要とします。1つのI/Oポート内の多数のピンが同じ形態設定を必要とするかもしれません。1つのI/Oポート内の多数のピンを同じ形態へ設定する処理は複数ピン形態設定処理を通して簡単化されます。最初に、形態設定されるべきピンに一致するビット様式がPORTCFG部内のMPCMSKレジスタに書かれます。そのポートのPINnCTRLレジスタの1つにピン形態設定が書かれる時に、その値がMPCMSKレジスタ内のビット様式に一致する全てのピンのPINnCTRLレジスタに書かれます。それはMPCMASKレジスタによって目的対象とされたレジスタの1つに書く必要はありません。書かれたレジスタがMPCMASKのビット遮蔽によって目的対象とされていない場合、それは無変化に留まります。

複数ピン形態設定がPINnCTRLレジスタを書く作業によって割り込まれないのが重要なことに注意してください。MPCMASKレジスタが既に書かれて、割り込みが実行され、そしてその割り込み処理ルーチン(ISR)が何れかのPINnCTRLレジスタに書く場合、意図と異なるI/Oポートに属するピンが形態設定されるでしょう。推奨される解決策は複数ピン形態設定を行う前に全体割り込みビットを保存して禁止し、形態設定が書かれた後で全体割り込みビットを回復することです。

複数ピン形態設定の使用法の例に関しては3.節をご覧ください。

2.5.2. 仮想ポート

AVR®命令一式内のいくつかの命令はAVRのI/O空間内のアドレスでだけ操作することができます。それらの等価なデータ空間の代わりにこれらの命令を使用することは、より速くそしてプログラムメモリをより少なく消費する両方です。XMEGAでの全てのI/OポートレジスタはI/O空間外のアドレスを持ちます。

これに対する解決策は仮想ポートを使用することです。4つまでのI/OポートがI/O空間内のレジスタを持つ仮想ポートに割り当てることができます。仮想ポートはDIR, OUT, IN, INTFLAGSレジスタをI/O空間内で利用可能な望むI/Oポートにします。使用されたI/Oレジスタ以外のその他は未だ通常のポート部署レジスタを通して利用可能です。

2.5.3. I/O空間とデータ空間の命令の違い

関連するデータ空間命令に対する特別なI/O空間命令の実行時間とコード量が表2-5.で示されます。

表2-5. I/O空間とデータ空間の命令に関する実行時間とコード量

| 命令 | クロック周期 | 量 (語) | 注釈 |
|---------|---------------|-------|-----------------------|
| IN | 1 | 1 | I/O空間のみ |
| OUT | 1 | 1 | I/O空間のみ |
| CBI | 1 | 1 | \$001F以下のアドレスのI/O空間のみ |
| SBI | 1 | 1 | \$001F以下のアドレスのI/O空間のみ |
| SBIC | 2/3/4 | 1 | \$001F以下のアドレスのI/O空間のみ |
| SBIS | 2/3/4 | 1 | \$001F以下のアドレスのI/O空間のみ |
| LD/LD+ | 1(I/O),2(RAM) | 1 | ポインタが初期化されなければなりません。 |
| LD-/LDD | 2(I/O),3(RAM) | 1 | ポインタが初期化されなければなりません。 |
| LDS | 2(I/O),3(RAM) | 2 | |
| ST/ST+ | 1 | 1 | ポインタが初期化されなければなりません。 |
| ST-/STD | 2 | 1 | ポインタが初期化されなければなりません。 |
| STS | 2 | 2 | |

I/O空間がデータ空間内にも割り当てられているので、データ空間でのデータ命令操作がI/O空間アドレスを持つレジスタに使用できることに注目してください。

実行時間とコード量の違いが大きいと思わないかもしれませんが、実際の例ではこの違いが巨大になり得ます。例証のため、他のピンの状態を変更せずにPC0ピンをHighに設定することを考察してください。以下の2つの例はPVIRT0(仮想ポート0)に割り当てられたPORTCをデータ空間でのPORTC直接アクセスと比較する時に、この課題を実行するのに必要とするアセンブリ言語例を示します。

PVIRT0 (I/O空間) 仮想ポート使用

```
SBI PVIRT0_OUT, 0
```

このコード量は1語で、実行時間は1クロック周期です。

PORTC (データ空間) 直接使用

```
SBR R16, 0b00000001
STS PORTC_base + PORT_OUTSET_offset, R16
```

このコード量は3語で、実行時間は3クロック周期です。

上の例はポートを仮想ポートに割り当てることによって大事なクロック周期とプログラムメモリの量を節約することが可能なことを示します。応用でタイミングの必要条件がきつい時やポートレジスタが頻繁にアクセスされる時に仮想ポート割り当てを使用することが推奨されます。

例えCPUが“SBI”命令を1クロック周期で完了するとしても、結果をI/Oポートで見ることが出来るまでに2クロック周期かかることに注意してください。

3. 開始に際して

本項はXMEGAのI/Oピンとポート割り込みでの準備と実行に対する基本段階を簡単に片付けます。

3.1. 基本デジタルI/O

課題: 8つのスイッチから入力を読むためにPORTCを初期設定して、PORTDに接続された8つのLEDにピン状態を出力してください。

1. **PORTD.DIR**レジスタを\$FFに設定することによってPORTDの全8ピンを出力に形態設定してください。
2. **PORTC.IN**レジスタからPORTCの状態を読んでください。
3. 段階2.からの値を**PORTD.OUT**に格納してください。
4. 段階2.からを繰り返してください。

3.2. 1操作での多数ピン形態設定

課題: PORTCのピン0~3をプルアップ付きのワイヤードAND動作に初期設定してください。

1. ピン0~3が形態設定を行われるのを選択するため、**PORTCFG.MPCMASK**に\$0Fを書いてください。
2. PORTCのピン0~3に対して**PINnCTRL**レジスタへの書き込みを起動するために、**PIN0CTRL**に**PORT_OPC_WIREDANDPULL_gc**を書いてください。

3.3. 仮想ポートへの実ポート割り当て

課題: PORTCを仮想ポート0へ、PORTDを仮想ポート1に割り当て、そして仮想ポートを使用して3.1.項と同じ課題を実行してください。

1. PORTCをPVIRT0に割り当てるために、**PORTCFG.VIRTMAP0_PORTC_gc**を**PORTCFG.VMAP0**へ書いてください。
2. PORTDをPVIRT1に割り当てるために、**PORTCFG.VIRTMAP1_PORTD_gc**を**PORTCFG.VMAP1**へ書いてください。
3. **PVIRT1.DIR**レジスタを\$FFに設定することによってPORTDの全8ピンを出力に形態設定してください。
4. **PVIRT0.IN**レジスタからPORTCの状態を読んでください。
5. 段階4.からの値を**PVIRT1.OUT**に格納してください。
6. 段階4.からを繰り返してください。

3.4. 割り込み生成用のI/Oピン形態設定

課題: PORTC割り込み0をPC0の上昇端による起動の中レベル割り込みとして初期設定してください。PORTDでの出力を切り換えるために割り込み処理ルーチンを使用してください。

1. ピン0での入力/感知を上昇端に形態設定してください。
2. 割り込み0に対する供給元としてPC0を選択するために、\$01を**PORTC.INTOMASK**に書いてください。
3. 中レベルで割り込み0を許可するために**PORTC.INTCTRL**の**IN0LVL**部分を**PORT_INT0LVL_MED_gc**に設定してください。
4. PMICで中レベル割り込みを許可してください。
5. 全体割り込みビットを許可してください。

4. ドライバ実装

内包されたドライバはI/Oポート部署の主な機能の全てを制御する関数を持ちます。全ての関数は最初の引数としてI/Oポート部署に対するポイントを取り、故に1つのXMEGAの全ポート部署に対して同じ関数を再使用することができます。

このドライバはANSI® Cで書かれていて、XMEGA支援を持つ全てのコンパイラでコンパイルできるでしょう。このドライバが高性能の考えで書かれていないことに注意してください。それはXMEGAのI/Oポートでの始めと素早い試作に対する容易な使用の雛形を得るためのライブラリとして設計されています。時間とコード量が重要な応用開発については、マクロまたはレジスタへの直接アクセスでの関数呼び出しへの置換を考慮してください。

4.1. ファイル

ソースコード一括は次のファイルから成ります。

- **port_driver.c** : I/Oポートドライバ ソース ファイル
- **port_driver.h** : I/Oポートドライバ ヘッダ ファイル
- **port_example.c** : I/Oポートドライバ使用例

4.2. Doxygen資料化

全てのソースファイルはDoxygenを使用する自動資料生成用に準備されています。Doxygenは特別なキーワードを使用してソースコードを分析することによって、ソースコードから資料を作成するツールです。Doxygenについてのより多くの詳細に関しては<http://www.doxygen.org>を訪ねてください。予めコンパイルされたDoxygen資料は本応用記述に伴うソースコードと共に供給され、ソースコードフォルダの**readme.html**ファイルから利用可能です。



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
東熱新川ビル 9F
アトメル ジャパン株式会社
TEL (81) 03-3523-3551
FAX (81) 03-3523-7581

製品窓口

ウェブサイト

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

お断り: 本資料内の情報はATMEL製品と関連して提供されています。本資料またはATMEL製品の販売と関連して承諾される何れの知的所有権も禁反言あるいはその逆によって明示的または暗示的に承諾されるものではありません。ATMELのウェブサイトに位置する販売の条件とATMELの定義での詳しい説明を除いて、商品性、特定目的に関する適合性、または適法性の暗黙保証に制限せず、ATMELはそれらを含むその製品に関連する暗示的、明示的または法令による如何なる保証も否認し、何ら責任がないと認識します。たとえATMELがそのような損害賠償の可能性を進言されたとしても、本資料を使用できない、または使用以外で発生する(情報の損失、事業中断、または利益の損失に関する制限なしの損害賠償を含み)直接、間接、必然、偶然、特別、または付随して起こる如何なる損害賠償に対しても決してATMELに責任がないでしょう。ATMELは本資料の内容の正確さまたは完全性に関して断言または保証を行わず、予告なしでいつでも製品内容と仕様の変更を行う権利を保留します。ATMELはここに含まれた情報を更新することに対してどんな公約も行いません。特に別の方法で提供されなければ、ATMEL製品は車載応用に対して適当ではなく、使用されるべきではありません。ATMEL製品は延命または生命維持を意図した応用での部品としての使用に対して意図、認定、または保証されません。

© Atmel Corporation 2008. 全権利予約済 ATMEL®、ロゴとそれらの組み合わせ、AVR®とその他はATMEL Corporationの登録商標、XMEGA®とその他は商標またはその付属物です。他の用語と製品名は一般的に他の商標です。

© HERO 2014.

本応用記述はATMELのAVR1313応用記述(doc8050.pdf Rev.8050A-02/08)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識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。