

Atmel-ICEデバッグ



Atmel-ICEはチップ上デバッグ能力を持つARM® Cortex®-Mに基づくAtmel® SAMとAtmel AVR® マイクロ コントローラのプログラミングとデバッグ用の強力な開発ツールです。

- JTAGとaWireの両インターフェースで全てのAtmel 32ビットAVRマイクロ コントローラのプログラミングとチップ上デバッグ
- JTAGと2線PDIの両インターフェースで全てのAtmel AVR XMEGA®系デバイスのプログラミングとチップ上デバッグ
- JTAGとデバッグWIREのどちらかのインターフェースでOCD支援を持つ全てのAtmel 8ビットAVRマイクロ コントローラ(JTAGとSPIの)プログラミングとデバッグ
- SWDとJTAGの両インターフェースでARM Cortex-Mに基づく全てのAtmel SAMマイクロ コントローラのプログラミングとデバッグ
- TPIインターフェース用支援を持つ全てのAtmel 8ビット tinyAVR®マイクロ コントローラのプログラミング(TPI)

デバイスの完全な一覧とこの公開ファームウェアによって支援されるインターフェースについてはAtmel Studio使用者に手引きで支援デバイス一覧を調べてください。

Atmel-ICEデバッグ	1	7. コマンド行ユーティリティ	17
1. 序説	3	8. 高度なデバッグ技術	17
1.1. Atmel-ICEの紹介	3	8.1. Atmel AVR UC3目的対象	17
1.2. Atmel-ICEの特徴	3	8.1.1. EVTI/EVTOの使い方	17
1.3. システム要件	3	8.2. デバッグWIRE目的対象	18
2. Atmel-ICEでの開始に際して	3	8.2.1. ソフトウェア中断点	18
2.1. 完全キット内容	3	9. 特別な考慮	18
2.2. 基本キット内容	4	9.1. Atmel AVR XMEGA OCD	18
2.3. PCBAキット内容	4	9.2. Atmel megaAVR OCDとデバッグWIRE OCD	19
2.4. 予備部品キット	4	9.2.1. Atmel megaAVR OCD (JTAG)	19
2.5. キット概要	5	9.2.2. デバッグWIRE OCD	20
2.6. Atmel-ICEの組み立て	5	9.3. Atmel AVR UC3 OCD	20
2.7. Atmel-ICEを開ける	6	9.4. SAM / CoreSight OCD	21
2.8. Atmel-ICEの給電	6	10. ファームウェア格上げ更新	21
2.9. ホストコンピュータへの接続	6	11. 公開履歴と既知の問題	21
2.10. USBドライバのインストール	6	11.1. 新規情報	21
2.10.1. Windows	6	11.2. ファームウェア公開履歴	21
3. Atmel-ICEの接続	7	11.2.1. Atmel Studio 6.2	21
3.1. 概要:AVRとSAM目的対象デバイスへの接続	7	11.2.2. Atmel Studio 6.2β	21
3.2. JTAG目的対象への接続	7	11.3. Atmel-ICEに関する既知の問題	21
3.3. aWire目的対象への接続	8	11.3.1. Atmel AVR XMEGA OCD特有問題	21
3.4. PDI目的対象への接続	8	11.3.2. Atmel megaAVR/tinyAVR OCD特有問題	21
3.5. デバッグWIRE目的対象への接続	9	11.4. 支援デバイス	21
3.6. SPI目的対象への接続	9	12. 製品適法性	22
3.7. TPI目的対象への接続	10	12.1. RoHSとWEEE	22
3.8. SWD目的対象への接続	10	12.2. CEとFCC	22
4. チップ上デバッグ	11	13. 資料改訂	22
4.1. チップ上デバッグ(OCD)の序説	11		
4.2. 物理インターフェース	11		
4.2.1. JTAG	11		
4.2.2. aWire	13		
4.2.3. PDI物性	13		
4.2.4. デバッグWIRE	13		
4.2.5. SPI	14		
4.2.6. TPI	14		
4.2.7. SWD	14		
4.3. Atmel OCD実装	14		
4.3.1. Atmel AVR UC3 OCD (JTAGとaWire)	14		
4.3.2. Atmel AVR XMEGA OCD (JTAGとPDI物性)	14		
4.3.3. Atmel megaAVR OCD (JTAG)	15		
4.3.4. Atmel megaAVR/tinyAVR OCD (デバッグWIRE)	15		
4.3.5. ARM CoreSight構成部	15		
5. ハードウェア説明	15		
5.1. LED	15		
5.2. 背面	15		
5.3. 底覆い	15		
5.4. 基本構造説明	15		
5.4.1. Atmel-ICE主基板	16		
5.4.2. Atmel-ICE目的対象コネクタ	16		
5.4.3. Atmel-ICE目的対象コネクタ部品番号	16		
6. ソフトウェア統合	16		
6.1. Atmel Studio	16		
6.1.1. Atmel Studioでのソフトウェア統合	16		
6.1.2. プログラミング任意選択	16		
6.1.3. デバッグ任意選択	17		

1. 特徴

1.1. Atmel-ICEの紹介

Atmel-ICEはチップ上デバッグ能力を持つARM Cortex-Mに基づくAtmel SAMとAtmel AVRマイクロコントローラのプログラミングとデバッグ用の強力な開発ツールです。

それは以下を支援します。

- JTAGとaWireの両インターフェースで全てのAtmel AVR UC3マイクロコントローラのプログラミングとチップ上デバッグ
- JTAGと2線PDIの両インターフェースで全てのAVR XMEGA系デバイスのプログラミングとチップ上デバッグ
- JTAGとデバッグWIREの両インターフェースでOCD支援を持つ全ての8ビットAVRマイクロコントローラのプログラミング(JTAGとSPI)とデバッグ
- JTAGとSWDの両インターフェースでARM Cortex-Mに基づく全てのAtmel SAMマイクロコントローラのプログラミングとデバッグ
- このインターフェースを支援する全てのAtmel 8ビットtinyAVRマイクロコントローラのプログラミング(TPI)

1.2. Atmel-ICEの特徴

- Atmel Studioに完全適合
- 全てのAtmel AVR UC3 32ビットマイクロコントローラのプログラミングとデバッグを支援
- 全ての8ビットAVR XMEGAデバイスのプログラミングとデバッグを支援
- OCDを持つ全てのAtmel 8ビット megaAVR®とtinyAVRデバイスのプログラミングとデバッグを支援
- ARM Cortex-Mに基づく全てのSAMマイクロコントローラのプログラミングとデバッグを支援
- 1.62~5.5Vの目的対象動作電圧範囲
- デバッグWIRE使用時に目的対象のVTrefから3mA未満、他の全てのインターフェースに対して1mA未満の引き出し
- 32kHz~7.5MHzのJTAGクロック周波数を支援
- 32kHz~7.5MHzのPDIクロック周波数を支援
- 4kビット/s~0.5Mビット/sのデバッグWIREホーレートを支援
- 7.5kビット/s~7Mビット/sのaWireホーレートを支援
- 8kHz~5MHzのSPIクロック周波数を支援
- 32kHz~2MHzのSWDクロック周波数を支援
- USB高速(High-speed)ホストインターフェース
- 最大3MB/sでのITM連続追跡捕獲
- AVRとCortexの両方のピン配列を持つ10ピン50mil JTAGコネクタを支援。標準探針ケーブルは10ピン50milだけでなく、AVRの6ピンISP/PDI/TPI 100milヘッダも支援します。6ピン50mil、10ピン100mil、20ピン100milヘッダを支援するアダプタが利用可能です。多数のキット任意選択が各種アダプタと配線で利用可能です。

1.3. システム要件

Atmel-ICE本体はコンピュータに前置デバッグ環境のAtmel Studio 6.2版またはそれ以降がインストールされる必要があります。Atmel-ICEは提供されたUSBケーブルまたは保証されたUSBマイクロケーブルを使用してホストコンピュータに接続されるべきです。

2. Atmel-ICEでの開始に際して

2.1. 完全キット内容

Atmel-ICE完全キットはこれらの品目を含みます。

- Atmel-ICE本体
- USBケーブル (1.8m、高速(High-speed)、マイクロ-B)
- 50mil AVR、100mil AVR/SAM、100mil 20ピンSAMのアダプタを含むアダプタ基板
- 10ピン50milコネクタと6ピン100milコネクタを持つIDCフラットケーブル
- 10×100milソケット付き50mil 10ピンミニバラ線ケーブル

図2-1. Atmel-ICE完全キット内容



2.2. 基本キット内容

Atmel-ICE基本キットはこれらの品目を含みます。

- Atmel-ICE本体
- USBケーブル (1.8m、高速(High-speed)、マイクロ-B)
- 10ピン50milコネクタと6ピン100milコネクタを持つIDCフラットケーブル

図2-2. Atmel-ICE基本キット内容

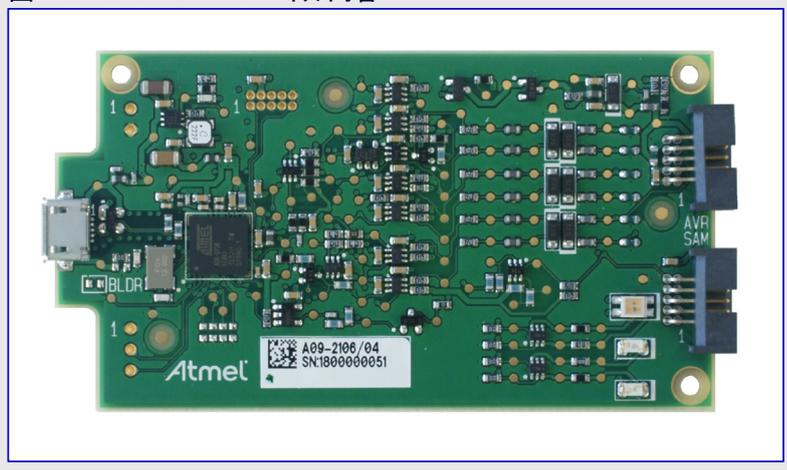


2.3. PCBAキット内容

Atmel-ICE PCBAキットは以下の品目を含みます。

- プラスティック筐体なしのAtmel-ICE本体

図2-3. Atmel-ICE PCBAキット内容



2.4. 予備部品キット

以下の予備部品キットが入手可能です。

- アダプタキット
- ケーブルキット

図2-4. Atmel-ICE アダプタキット内容



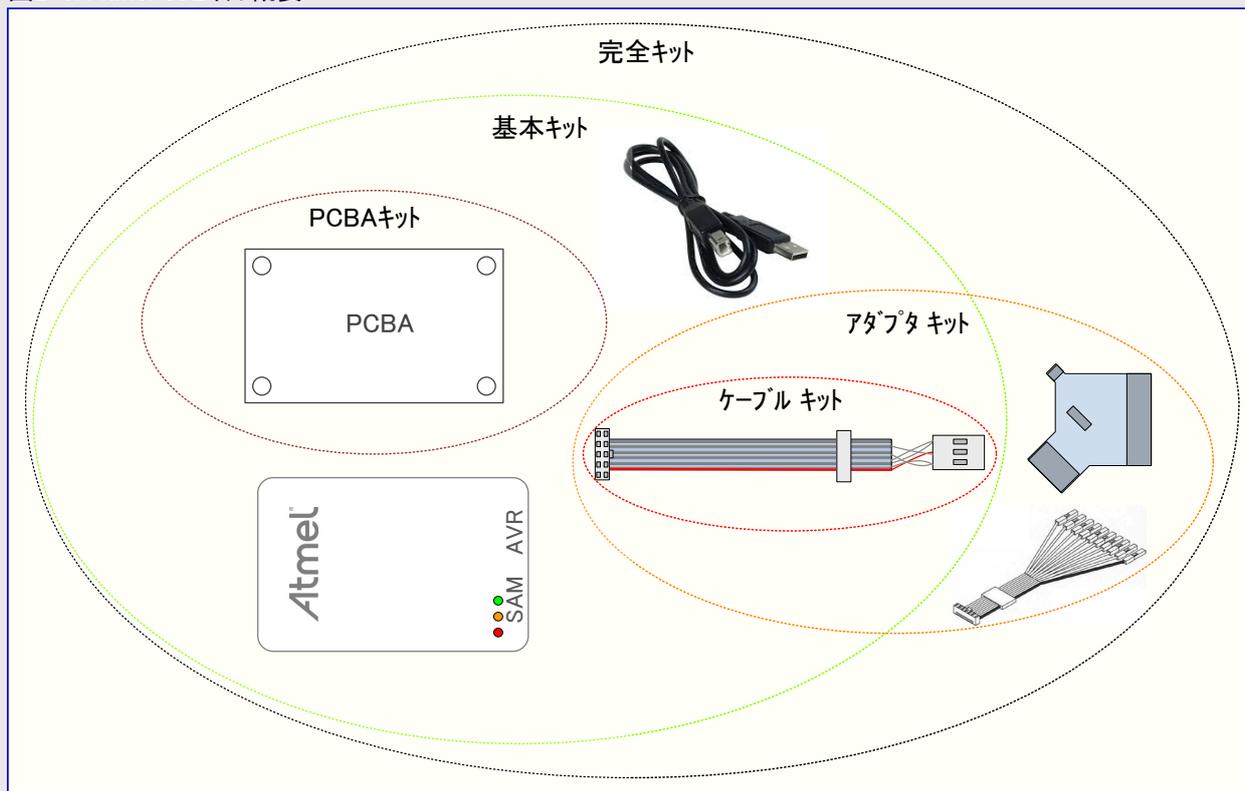
図2-5. Atmel-ICE ケーブルキット内容



2.5. キット概要

Atmel-ICEキット任意選択はこの図表で示されます。

図2-6. Atmel-ICEキット概要



2.6. Atmel-ICEの組み立て

Atmel-ICE本体は付随されるケーブルなしで出荷されます。完全キット内で次のような2つのケーブル任意選択が提供されます。

- ・ 6ピンISPと10ピン コネクタ付きの50mil 10ピンIDCフラット ケーブル
- ・ 10×100milソケット付き50mil 10ピン ミニ バラ線ケーブル

殆どの目的に関し、自然にその10ピンまたは6ピンのどちらかのコネクタに接続するか、またはアダプタ基板経由で接続するのに50mil 10ピンIDCフラット ケーブルを使用することができます。3つのアダプタが1つの小さなPCBAで提供されます。以下のアダプタが含まれます。

- ・ 100mil 10ピン JTAG/SWDアダプタ
- ・ 100mil 20ピン SAM JTAG/SWDアダプタ
- ・ 50mil 6ピン SPI/デバッグWIRE/PDI/aWireアダプタ

注: 50mil JTAGアダプタが提供されないことに注意してください。これは50mil JTAGヘッダに直接的に接続するのに50mil 10ピンIDCケーブルを使用することができるためです。50mil 10ピン コネクタに使用される部品の部品番号については「[Atmel-ICE 目的対象コネクタ部品番号](#)」をご覧ください。

10ピンIDCケーブルの部品として6ピンISP/PDIヘッダが含まれます。

Atmel-ICEをその既定形態設定に組み立てるには、右で示されるように本体に10ピンの50mil IDCケーブルを接続してください。ケーブルの赤い線(1番ピン)が筐体の青い帯の三角目印と整列するよう、ケーブルの向きに注意してください。ケーブルは本体から上側へ接続されるべきです。目的対象のAVRまたはSAMのピン配列に従ってポートへの接続を確実にしてください。

図2-7. Atmel-ICEケーブル

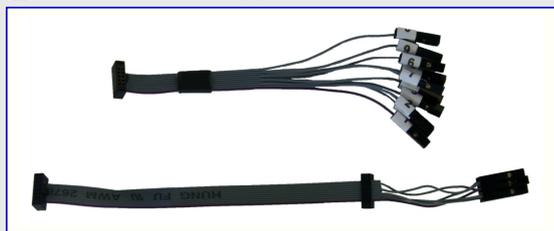


図2-8. Atmel-ICEアダプタ



図2-9. Atmel-ICEケーブル接続

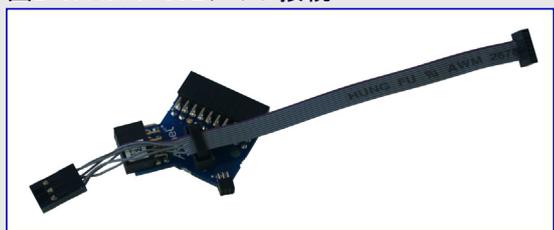


図2-10. Atmel-ICE AVR探針接続



図2-10. Atmel-ICE SAM探針接続



2.7. Atmel-ICEを開ける

注: 通常操作に関して、Atmel-ICE本体が開かれてはなりません。本体を開けることはあなた自身の責任に於いて行われます。静電防止予防策が取られるべきです。

Atmel-ICEの筐体は、組み立て中に共に嵌められる上覆い、底覆い、それと青い帯の分離した3つのプラスチック部品から成ります。本体を開けるには単に大きなマイナス(-)ドライバを青い帯の隙間に挿入し、或る程度内側へ圧力を印加して、優しく捻ってください。他の嵌め穴でこれを繰り返し、上覆いを取り上げてください。

図2-12. Atmel-ICEを開ける (訳注:原書での図2-12,13,14.は図2-12.として纏めました。)



本体を再び閉じるには、単に上覆いと底覆いを正しく整列し、共に固く押してください。

2.8. Atmel-ICEの給電

Atmel-ICEはUSBバス電圧によって給電されます。それは動作するのに100mA未満が必要で、故にUSBハブを通して給電することができます。電力LEDは本体が接続された時に点灯します。活動するプログラミングまたはデバッグの作業で接続されない時は、コンピュータの電池を保護するために本体が低電力消費へ移行します。Atmel-ICEは電力断にすることができず、未使用時に接続を外されるべきです。

2.9. ホスト コンピュータへの接続

Atmel-ICEは主に標準HIDインターフェースを使用して通信し、ホスト コンピュータで特別なドライバを必要としません。Atmel-ICEの高度なデータ中継器機能を使用するため、ホスト コンピュータでUSBドライバのインストールを確実にしてください。これはAtmelによって無料で提供される前置ソフトウェアをインストールする時に自動的に行われます。更なる情報や最終版の前置ソフトウェアのダウンロードについてはwww.atmel.comをご覧ください。

Atmel-ICEは提供されたUSBケーブルまたは適切で保証されたマイクロUSBケーブルを使用してホスト コンピュータで利用可能なUSBポートへ接続されなければなりません。Atmel-ICEはUSB 2.0適合制御器を含み、全速(Full-speed)と高速(High-speed)の両動作で動作することができます。最良の結果のために、提供されたケーブルを使用してホスト コンピュータのUSB 2.0適合高速ハブへ直接接続してください。

2.10. USBドライバのインストール

2.10.1. Windows

Microsoft® Windows®が走行するコンピュータでAtmel-ICEをインストールすると、Atmel-ICEが最初に接続される時にUSBドライバが読み込まれます。

注: 最初に本体を接続する前に、前置ソフトウェア一括のインストールを確実にしてください。

一旦成功裏にインストールされると、Atmel-ICEはデバイス マネージャで”対人インターフェース装置”として現れます。

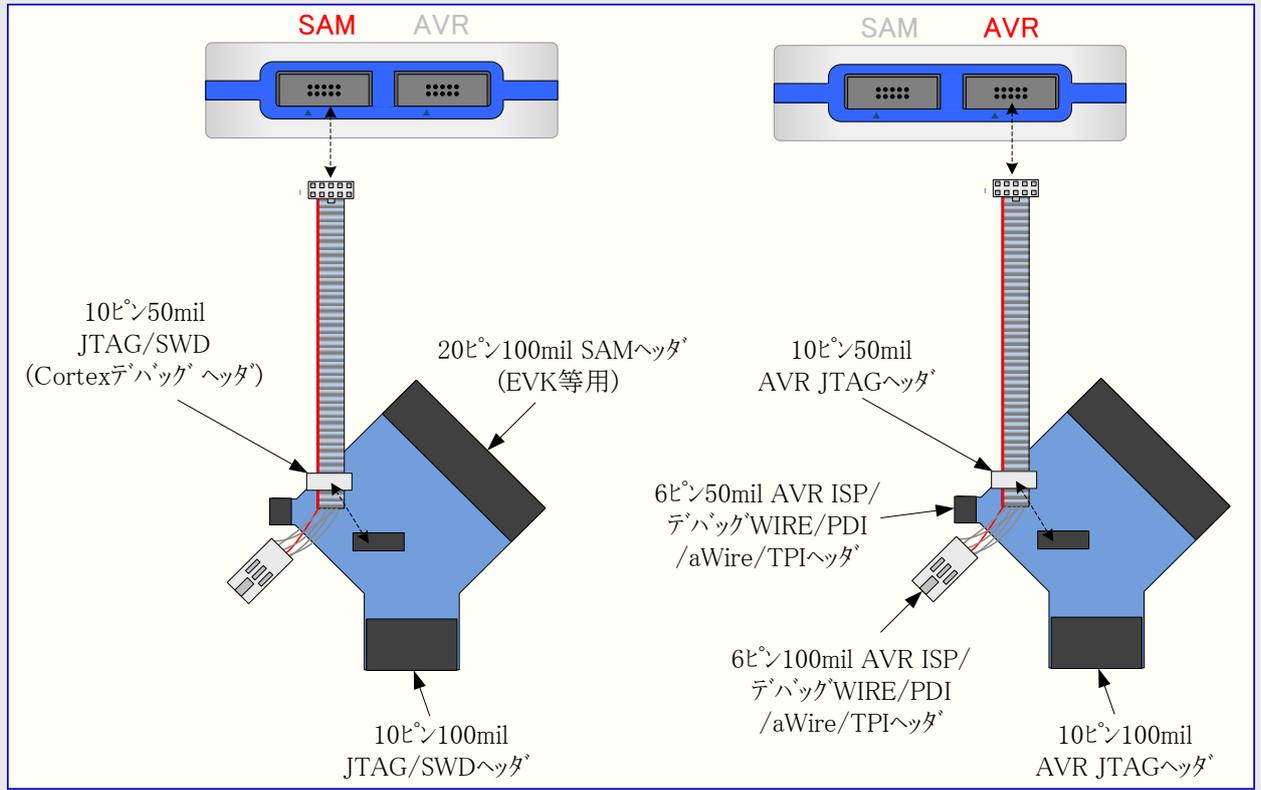
3. Atmel-ICEの接続

3.1. 概要:AVRとSAM目的対象デバイスへの接続

Atmel-ICE探針はツールの筐体の前面で利用できる50mil 10ピンの2つのJTAGコネクタを持ちます。両コネクタは電氣的に直接的に接続されていますが、AVR JTAGヘッダとARM Cortexデバッグヘッダの2つの異なるピン配列を確認してください。コネクタは目的対象MCU型式ではなく、目的対象基板のピン配列に基づいて選択されるべきです。例えばAVR STK600階層に装着されたSAMデバイスにはAVRヘッダが使用されるべきです。

各種Atmel-ICEキットで様々な配線やアダプタが利用可能です。接続任意選択の概要が示されます。

図3-1. Atmel-ICE接続任意選択



3.2. JTAG目的対象への接続

Atmel-ICE探針はツールの筐体の前面で利用できる50mil 10ピンの2つのJTAGコネクタを持ちます。両コネクタは電氣的に直接的に接続されていますが、AVR JTAGヘッダとARM Cortexデバッグヘッダの2つの異なるピン配列を確認してください。コネクタは目的対象MCU型式ではなく、目的対象基板のピン配列に基づいて選択されるべきです。例えばAVR STK600階層に装着されたSAMデバイスにはAVRヘッダが使用されるべきです。

10ピンAVR JTAGコネクタの推奨ピン配列は11頁の「[図4-2. AVR JTAGヘッダ ピン配列](#)」で示されます。

10ピンARM Cortexデバッグコネクタの推奨ピン配列は12頁の「[図4-3. SAM JTAGヘッダ ピン配列](#)」で示されます。

標準10ピン50milヘッダへの直接接続

このヘッダ形式を支援する基板へ直接接続するには(いくつかのキットに含まれる)50mil 10ピンフラットケーブルを使用してください。AVRピン配列で配置されたヘッダ用のAtmel-ICEのAVRコネクタポートとARM Cortexデバッグヘッダに従うヘッダ用のSAMコネクタポートを使用してください。

10ピンの両コネクタのピン配列は以降で示されます。

標準10ピン100milヘッダへの接続

100milヘッダに接続するには標準50mil⇒100milアダプタを使用してください。(いくつかのキットに含まれる)アダプタ基板がこの目的に使用することができ、または代わりにAVR目的対象に対してJTAGICE3のアダプタを使用することができます。

注: アダプタの2と10番ピン(AVR GND)が接続されるため、JTAGICE3の100milアダプタはSAMコネクタポートで使用することはできません。

独自100milヘッダへの接続

目的対象基板が50milか100milの適合10ピンJTAGヘッダを持たない場合、10個の個別100milソケットへの入出力を与える(いくつかのキットに含まれる)10ピンの”ミニバラ線”ケーブルを使用して独自ピン配列に割り当てることができます。

20ピン100milヘッダへの接続

20ピン100milヘッダを持つ目的対象へ接続するには(いくつかのキットに含まれる)アダプタ基板を使用してください。

表3-1. Atmel-ICE JTAGピン説明

名前	ポートのピン番号		説明
	AVR	SAM	
TCK	1	4	検査クロック (Atmel-ICEから目的対象デバイスへのクロック信号)
TMS	5	2	検査種別選択 (Atmel-ICEから目的対象デバイスへの制御信号)
TDI	9	8	検査データ入力 (Atmel-ICEから目的対象デバイスへ送出されるデータ)
TDO	3	6	検査データ出力 (目的対象デバイスからAtmel-ICEへ送出されるデータ)
nTRST	8	-	検査リセット(任意、いくつかのAVRデバイスのみ)。JTAG TAP制御器のリセットに使用
nSRST	6	10	リセット (任意)。目的対象デバイスのリセットに使用。或る筋書でのデバッグを必要とし得る、目的対象デバイスをリセット状態で保持することをAtmel-ICEに許すために、このピンの接続が推奨されます。
VTG	4	1	目的対象基準電圧。Atmel-ICEは正しくレベル変換器を給電するために、このピンで目的対象電圧を採取します。Atmel-ICEはこのピンからデバッグWIRE動作で3mA未満、他の動作で1mA未満を引き出します。
GND	2,10	3,5,9	接地。Atmel-ICEと目的対象デバイスが同じ接地基準を共用するのを保証するために全てが接続されなければなりません。

3.3. aWire目的対象への接続

aWireインターフェースはVCCとGNDに加えて1つのデータ線だけが必要です。例えデバッグがデータ線としてJTAG TDOを使用しても、目的対象でこの線はnRESET線です。

6ピンaWireコネクタ用の推奨ピン配列は13頁の「[図4-5. aWireヘッダピン配列](#)」で示されます。

6ピン100mil aWireヘッダへの接続

標準100mil aWireヘッダへ接続するには(いくつかのキットに含まれる)フラット ケーブル上の6ピン100mil引き出しを使用してください。

6ピン50mil aWireヘッダへの接続

標準50mil aWireヘッダへ接続するには(いくつかのキットに含まれる)アダプタ基板を使用してください。

独自100milヘッダへの接続

Atmel-ICE AVRコネクタ ポートと目的対象基板を接続するには10ピンのミニパラ線ケーブルが使用されるべきです。右表で記述されるように3つの接続が必要とされます。

表3-2. Atmel-ICEのaWireピン割り当て

Atmel-ICE AVRポートピン	目的対象ピン	ミニパラ線ピン	aWireピン
1 (TCK)		1	
2 (GND)	GND	2	6
3 (TDO)	DATA	3	1
4 (VTG)	VTG	4	2
5 (TMS)		5	
6 (nSRST)		6	
7 (未接続)		7	
8 (nTRST)		8	
9 (TDI)		9	
10 (GND)		0	

3.4. PDI目的対象への接続

6ピンPDIコネクタ用の推奨ピン配列は13頁の「[図4-6. PDIヘッダピン配列](#)」で示されます。

6ピン100mil PDIヘッダへの接続

標準100mil PDIヘッダへ接続するには(いくつかのキットに含まれる)フラット ケーブル上の6ピン100mil引き出しを使用してください。

6ピン50mil PDIヘッダへの接続

標準50mil PDIヘッダへ接続するには(いくつかのキットに含まれる)アダプタ基板を使用してください。

独自100milヘッダへの接続

Atmel-ICE AVRコネクタ ポートと目的対象基板を接続するには10ピンのミニパラ線ケーブルが使用されるべきです。次表で記述されるように4つの接続が必要とされます。

注: PDI_DATAが9番ピンに接続されるJTAGICE mk IIのJTAG探針とは違いがあります。Atmel-ICEはJTAGICE3、AVR ONE!、AVR Dragon製品によって使用されるピン配列と互換性があります。

表3-3. Atmel-ICEのPDIピン割り当て

Atmel-ICE AVRポートピン	目的対象ピン	ミニパラ線ピン	STK600 PDIピン
1 (TCK)		1	
2 (GND)	GND	2	6
3 (TDO)	PDI_DATA	3	1
4 (VTG)	VTG	4	2
5 (TMS)		5	
6 (nSRST)	PDI_CLK	6	5
7 (未接続)		7	
8 (nTRST)		8	
9 (TDI)		9	
10 (GND)		0	

3.5. デバッグWIRE目的対象への接続

6ピン デバッグWIRE(SPI)コネクタ用の推奨ピン配列は13頁の「[図4-7. デバッグWIRE\(SPI\)ヘッダピン配列](#)」で示されます。

6ピン100mil SPIヘッダへの接続

標準100mil SPIヘッダへ接続するにはいくつかのキットに含まれるフラットケーブル上の6ピン100mil引き出しを使用してください。

6ピン50mil SPIヘッダへの接続

標準50mil SPIヘッダへ接続するにはいくつかのキットに含まれるアダプタ基板を使用してください。

独自100milヘッダへの接続

Atmel-ICE AVRコネクタポートと目的対象基板を接続するには10ピンのミニパラ線ケーブルが使用されるべきです。下表で記述されるように3つの接続が必要とされます。

例え正しく動作するためにデバッグWIREインターフェースがVCC、GNDと1つの単線(RESET)だけが必要でも、SPIプログラミングを使用してデバッグWIREインターフェースを許可/禁止することができるように完全なSPIコネクタへの入出力を持つことが推奨されます。

DWENヒューズが許可されると、OCD単位部がRESETピン上の制御を持つために、内部的にSPIインターフェースが無効にされます。デバッグWIRE OCDは(Atmel Studioのプロパティダイアログ内のデバッグタブ上の鉤を使用して)一時的にそれ自身を禁止する能力があり、故にRESET線の制御を解放します。(SPIENヒューズだけがプログラム(0)されるなら、)その後SPIインターフェースを使用してDWENヒューズを非プログラム(1)にすることを許すSPIインターフェースが再び利用可能です。DWENヒューズが非プログラム(1)にされる前に電源がOFF/ONされた場合、デバッグWIRE単位部が再びRESETピンの制御を取ります。

注: 単純にAtmel StudioにDWENヒューズの設定と解除を扱わせることを強く勧告します。

目的対象Atmel AVRの施錠ビットがプログラム(0)される場合、デバッグWIREインターフェースを使うのは不可能です。常にDWENヒューズをプログラム(0)する前に施錠ビットが解除(1)されていることを確実にして、DWENヒューズがプログラム(0)されている間に施錠ビットを決して設定(0)しないでください。デバッグWIRE許可(DWEN)ヒューズと施錠ビットの両方が設定(0)される場合、チップ消去を行うのに高電圧プログラミングを使用することができ、そのようにして施錠ビットを解除(1)してください。施錠ビットが解除(1)されると、デバッグWIREインターフェースが再び許可されます。DWENヒューズが非プログラム(1)の時のSPIインターフェースはヒューズ読み出し、識票読み出し、チップ消去実行の能力だけです。

表3-4. Atmel-ICEのデバッグWIREピン割り当て

Atmel-ICE AVRポートピン	目的対象ピン	ミニパラ線ピン
1 (TCK)		1
2 (GND)	GND	2
3 (TDO)		3
4 (VTG)	VTG	4
5 (TMS)		5
6 (nSRST)	RESET	6
7 (未接続)		7
8 (nTRST)		8
9 (TDI)		9
10 (GND)		0

3.6. SPI目的対象への接続

6ピンSPIコネクタ用の推奨ピン配列は14頁の「[図4-8. SPIヘッダピン配列](#)」で示されます。

6ピン100mil SPIヘッダへの接続

標準100mil SPIヘッダへ接続するにはいくつかのキットに含まれるフラットケーブル上の6ピン100mil引き出しを使用してください。

6ピン50mil SPIヘッダへの接続

標準50mil SPIヘッダへ接続するにはいくつかのキットに含まれるアダプタ基板を使用してください。

独自100milヘッダへの接続

Atmel-ICE AVRコネクタポートと目的対象基板を接続するには10ピンのミニパラ線ケーブルが使用されるべきです。次表で記述されるように6つの接続が必要とされます。

注: デバッグWIRE許可(DWEN)ヒューズがプログラム(0)されている時に、例えばSPIENヒューズもプログラム(0)されていても、SPIインターフェースは事実上禁止されます。SPIインターフェースを再許可するには、デバッグWIREでのデバッグ作業中に' disable debugWIRE ' 命令が発行されなければなりません。この方法でのデバッグWIRE禁止はSPIENヒューズが既にプログラム(0)されていることが必要です。Atmel StudioがデバッグWIRE禁止を失敗する場合、多分SPIENヒューズがプログラム(0)にされていないでしょう。その場合、SPIENヒューズをプログラム(0)するのに高電圧プログラミングインターフェースを使うことが必要です。

表3-5. Atmel-ICEのSPIピン割り当て

Atmel-ICE AVRポートピン	目的対象ピン	ミニパラ線ピン	SPIピン
1 (TCK)	SCK	1	3
2 (GND)	GND	2	6
3 (TDO)	MISO	3	1
4 (VTG)	VTG	4	2
5 (TMS)		5	
6 (nSRST)	/RESET	6	5
7 (未接続)		7	
8 (nTRST)		8	
9 (TDI)	MOSI	9	4
10 (GND)		0	

3.7. TPI目的対象への接続

6ピンTPIコネクタ用の推奨ピン配列は14頁の「[図4-9. TPIヘッダピン配列](#)」で示されます。

6ピン100mil TPIヘッダへの接続

標準100mil TPIヘッダへ接続するには(いくつかのキットに含まれる)フラットケーブル上の6ピン100mil引き出しを使用してください。

6ピン50mil TPIヘッダへの接続

標準50mil TPIヘッダへ接続するには(いくつかのキットに含まれる)アダプタ基板を使用してください。

独自100milヘッダへの接続

Atmel-ICE AVRコネクタポートと目的対象基板を接続するには10ピンのミニパラ線ケーブルが使用されるべきです。右表で記述されるように5つの接続が必要とされます。

表3-6. Atmel-ICEのTPIピン割り当て

Atmel-ICE AVRポートピン	目的対象ピン	ミニパラ線ピン	TPIピン
1 (TCK)	CLOCK	1	3
2 (GND)	GND	2	6
3 (TDO)	DATA	3	1
4 (VTG)	VTG	4	2
5 (TMS)		5	
6 (nSRST)	/RESET	6	5
7 (未接続)		7	
8 (nTRST)		8	
9 (TDI)		9	
10 (GND)		0	

3.8. SWD目的対象への接続

ARM SWDインターフェースはTCKとTMSのピンを利用するJTAGインターフェースの一部分で、SWDデバイスに接続する時に技術的に10ピンJTAGコネクタを使用できることを意味します。けれどもARM JTAGとAVR JTAGのコネクタはピン互換ではなく、故にこれは使用する目的対象基板の配置に依存します。STK600使用またはAVR JTAGピン配列利用時、Atmel-ICEのAVRコネクタポートが使用されなければなりません。ARM JTAGピン配列を利用する基板への接続時、Atmel-ICEのSAMコネクタポートが使用されなければなりません。

10ピンCortexデバッグコネクタ用の推奨ピン配列は14頁の「[図4-10. ARM SWD/JTAGヘッダピン配列](#)」で示されます。

10ピン50mil Cortexヘッダへの接続

標準50mil Cortexヘッダへ接続するには(いくつかのキットに含まれる)フラットケーブルを使用してください。

10ピン100mil Cortexピン配列ヘッダへの接続

100mil Cortexピン配列ヘッダへ接続するには(いくつかのキットに含まれる)アダプタ基板を使用してください。

20ピン100mil SAMヘッダへの接続

20ピン100mil SAMヘッダへ接続するには(いくつかのキットに含まれる)アダプタ基板を使用してください。

独自100milヘッダへの接続

Atmel-ICE AVRまたはSAMコネクタポートと目的対象基板を接続するには10ピンのミニパラ線ケーブルが使用されるべきです。下表で記述されるように6つの接続が必要とされます。

表3-7. Atmel-ICE SWDピン割り当て

名前	ポートのピン番号		説明
	AVR	SAM	
SWDCLK	1	4	直列線デバッグ(SWD:Serial Wire Debug)クロック
SWDIO	5	2	直列線デバッグ データ入出力
SWO	3	6	直列線デバッグ出力 (任意-全てのデバイスで実装されるとは限りません。)
nSRST	6	10	リセット
VTG	4	1	目的対象電圧基準
GND	2,10	3,5	接地

4. チップ上デバッグ

4.1. チップ上デバッグ(OCD)の序説

伝統的なエミュレータは目的対象デバイスの正確な動きを真似ようとする道具です。より近いこの動きは実際のデバイスの動きで、より良い模倣です。

Atmel-ICEは伝統的なエミュレータではありません。代わりに、Atmel-ICEはデバイスの実行を監視して制御するための機構を提供する目的対象デバイス内の内部チップ上デバッグシステム(OCD:On Chip Debug system)とインターフェースします。この方法ではデバッグをされつつある応用がエミュレートされませんが、現実の目的対象デバイス上で実際に実行されます。

OCDシステムとて、応用は伝統的なエミュレータでは技術的に実現可能ではない何か、即ち目的対象システムで正確な電氣的及びタイミングの特性を保って静かに実行することができます。

走行動作

走行動作時、コードの実行はAtmel-ICEと完全に独立です。Atmel-ICEは中断条件が発生したかを知るために目的対象デバイスを継続的に監視します。これが発生すると、OCDシステムはデバッグインターフェースを通してデバイスに質問し、使用者にデバイスの内部状況を見ることを許します。

停止動作

中断点(ブレークポイント)到達時、プログラム実行は停止されますが、全ての入出力は中断点が起きなかったように走行を継続します。例えば中断点発生時に丁度USART送信が初期化(/設定)されたと仮定します。この場合は例えコアが停止動作であっても、USARTは送信を完了する全速で走行を継続します。

ハードウェア中断点

目的対象OCD部はハードウェアで実装された多数のプログラムカウンタ比較器を含みます。プログラムカウンタが比較器レジスタの1つに格納された値と一致すると、OCDは停止動作へ移行します。ハードウェア中断点はOCD部で専用のハードウェアを必要とするため、利用可能な中断点数は目的対象AVRに実装されるOCD部の大きさに依存します。通常、デバッグによって内部使用のために1つのこのようなハードウェア比較器が”予約”されます。様々なOCD部で利用可能なハードウェア中断点のより多くの情報については「[Atmel OCD実装](#)」項をご覧ください。

ソフトウェア中断点

ソフトウェア中断点は目的対象デバイス上のプログラムメモリに配置されたBREAK命令です。この命令が読み込まれると、プログラム実行が中断され、OCDは停止動作へ移行します。実行を継続するには、OCDから”start”命令を与えなければなりません。全てのAVRがBREAK命令を支援するOCD部を持っている訳ではありません。様々なOCD部で利用可能なソフトウェア中断点のより多くの情報については「[Atmel OCD実装](#)」項をご覧ください。

OCDシステム使用時の考慮と制限の更なる情報については18頁の「[特別な考慮](#)」をご覧ください。

4.2. 物理インターフェース

Atmel-ICEはこれ以降の項で記述されるように多数のハードウェアインターフェースを支援します。

4.2.1. JTAG

JTAGインターフェースはIEEE 1149.1規格に適合する4線検査入出力ポート(TAP:Test Access Port)制御器から成ります。IEEE規格は回路基板の接続性(境界走査)を効率的に検査する工業標準的な方法を提供するために開発されました。AtmelのAVRとSAMのデバイスは完全なプログラミングとチップ上デバッグの支援を含むように拡張されたこの機能を持ちます。

JTAGインターフェースを持つAtmel AVRを含む応用PCB設計時、「[図4-2. AVR JTAGヘッダピン配列](#)」で示されるようなピン配列を使用することが推奨されます。Atmel-ICEはこのピン配列の100milと50milの両変種に接続することができます。

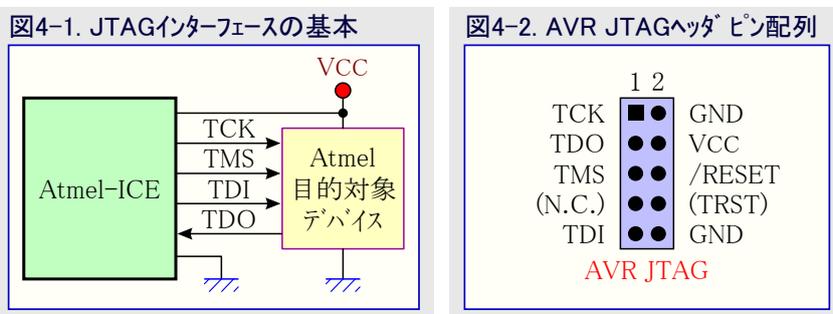


表4-1. AVR JTAGピン説明

名前	ピン番号	説明
TCK	1	検査クロック (Atmel-ICEから目的対象デバイスへのクロック信号)
TMS	5	検査種別選択 (Atmel-ICEから目的対象デバイスへの制御信号)
TDI	9	検査データ入力 (Atmel-ICEから目的対象デバイスへ送出されるデータ)
TDO	3	検査データ出力 (目的対象デバイスからAtmel-ICEへ送出されるデータ)
nTRST	8	検査リセット (任意、いくつかのAVRデバイスのみ)。JTAG TAP制御器のリセットに使用
nSRST	6	リセット (任意)。目的対象デバイスのリセットに使用。或る筋書でのデバッグを必要とし得る、目的対象デバイスをリセット状態で保持することをAtmel-ICEに許すために、このピンの接続が推奨されます。
VTG	4	目的対象基準電圧。Atmel-ICEは正しくレベル変換器を給電するために、このピンで目的対象デバイス電圧を採取します。Atmel-ICEはこのピンからデバッグWIRE動作で3mA未満、他の動作で1mA未満を引き出します。
GND	2,10	接地。Atmel-ICEと目的対象デバイスが同じ接地基準を共用するのを保証するために両方が接続されなければなりません。

助言: 4番ピンとGND間に雑音分離(デカップ)コンデンサを含むことを覚えて置いてください。

JTAGインターフェースを持つAtmel SAMを含む応用PCB設計時、「**図4-3. SAM JTAGヘッダピン配列**」で示されるようなピン配列を使用することが推奨されます。Atmel-ICEはこのピン配列の100milと50milの両変種に接続することができます。

図4-3. SAM JTAGヘッダピン配列

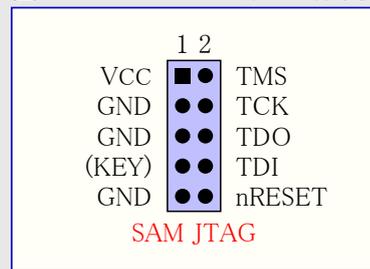


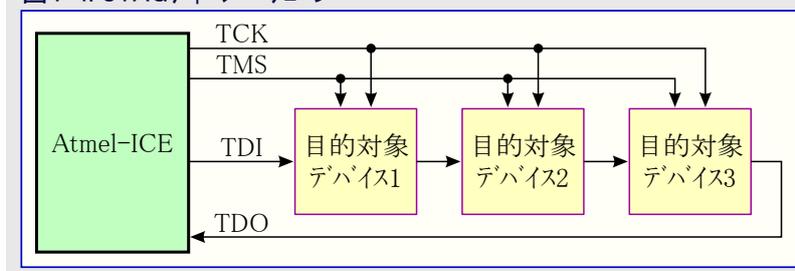
表4-2. SAM JTAGピン説明

名前	ピン番号	説明
TCK	4	検査クロック (Atmel-ICEから目的対象デバイスへのクロック信号)
TMS	3	検査種別選択 (Atmel-ICEから目的対象デバイスへの制御信号)
TDI	8	検査データ入力 (Atmel-ICEから目的対象デバイスへ送出されるデータ)
TDO	6	検査データ出力 (目的対象デバイスからAtmel-ICEへ送出されるデータ)
nRESET	10	リセット (任意)。目的対象デバイスのリセットに使用。或る筋書でのデバッグを必要とし得る、目的対象デバイスをリセット状態で保持することをAtmel-ICEに許すために、このピンの接続が推奨されます。
VTG	1	目的対象電圧基準。Atmel-ICEは正しくレベル変換器を給電するために、このピンで目的対象デバイス電圧を採取します。Atmel-ICEはこのピンからデバッグWIRE動作で3mA未満、他の動作で1mA未満を引き出します。
GND	3,5,9	接地。Atmel-ICEと目的対象デバイスが同じ接地基準を共用するのを保証するために全てが接続されなければなりません。
KEY	7	内部的にAVRコネクタのTRSTピンに接続。無接続としてが推奨されます。

助言: 1番ピンとGND間に雑音分離(デカップ)コンデンサを含むことを覚えて置いてください。

JTAGインターフェースは多数のデバイスに対してデジーチェーン形態設定内で単一インターフェースへ接続することを許します。目的対象デバイスは全てが同じ供給電圧によって給電され、共通接地節を共用しなければならず、「**図4-4. JTAGデジーチェーン**」で示されるように接続されなければなりません。

図4-4. JTAGデジーチェーン



デバッグチェーンでデバイス接続時、以下の点が考慮されなければなりません。

- 全てのデバイスはAtmel-ICE探針のGNDに接続された共通接地(GND)を共用しなければなりません。
- 全てのデバイスは同じ目的対象電圧で動作しなければなりません。Atmel-ICEのVTGはこの電圧に接続されなければなりません。
- TMSとTCKは並列で接続されます。TDIとTDOは直列連鎖で接続されます。
- チェーン内のデバイスの何れかがJTAGポートを禁止する場合、Atmel-ICE探針のnSRSTはデバイスのRESETに接続されなければなりません。
- "Devices before"はTDI信号が目的対象デバイスに到達する前にデバッグチェーン内を通過しなければならないJTAGデバイス数を参照します。同様に"Devices after"は信号がAtmel-ICEのTDOピンに到達する前に目的対象デバイスの後を通過しなければならないデバイス数です。
- "Instruction bits before"と"Instruction bits after"はデバッグチェーン内で目的対象デバイスの前後に接続される全てのJTAGデバイスの命令レジスタ(IR)長の総合計を参照します。
- 総IR長(Instruction bits before+Atmel AVR IR長+Instruction bits after)は最大256ビットに制限されます。チェーン内のデバイス数は前が15、後ろが15に制限されます。

助言: デバッグチェーン例: TDI ⇒ ATmega1280 ⇒ ATxmega128A1 ⇒ ATUC3A0512 ⇒ TDO

Atmel AVR XMEGAデバイスに接続するためのデバッグチェーン設定は次の通りです。

- Devices before : 1
- Devices after : 1
- Instruction bits before : 4 (8ビット AVRデバイスは4ビットのIRを持ちます。)
- Instruction bits after : 5 (32ビット AVRデバイスは5ビットのIRを持ちます。)

4.2.2. aWire

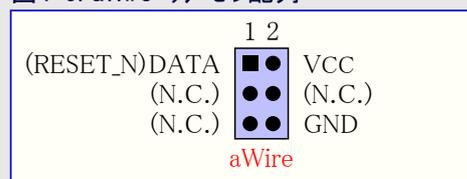
aWireインターフェースはプログラミングをデバッグの機能を許すためにAVRデバイスのRESET線を利用します。Atmel-ICEによって、ピンの既定RESET機能を禁止する特別な許可手順が送出されます。

aWireインターフェースを持つAtmel AVRを含む応用PCBの設計時、「[図4-5. aWireヘッダピン配列](#)」で示されるようなピン配列を使用することが推奨されます。Atmel-ICEはこのピン配列を支援する100milと50milの両アダプタと共に出荷します。

助言: aWireが半二重インターフェースなので、方向変更時に開始ビット検出の失敗を避けるため、RESET線上に約47kΩのプルアップ抵抗が推奨されます。

aWireインターフェースはプログラミングとデバッグの両方のインターフェースとして使用することができ、10ピンJTAGインターフェースを通して利用可能なOCDシステムの全機能はaWireを通してアクセスすることもできます。

図4-5. aWireヘッダピン配列

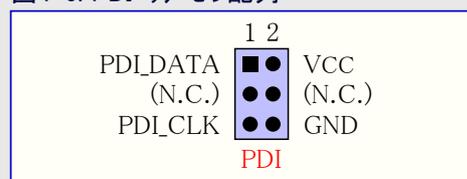


4.2.3. PDI物性

プログラミング/デバッグ用インターフェース(PDI: Program and Debug Interface)はデバイスの外部プログラミングとチップ上デバッグ用のAtmel専有インターフェースです。PDI物性は目的対象デバイスとの双方向半二重同期通信を提供する2ピンインターフェースです。

PDIインターフェースを持つAtmel AVRを含む応用PCBの設計時、「[図4-6. PDIヘッダピン配列](#)」で示されるピン配列が使用されるべきです。そしてAtmel-ICE探針を応用PCBへ接続するのに、Atmel-ICEキットと共に提供される6ピンアダプタの1つを使用することができます。

図4-6. PDIヘッダピン配列



4.2.4. デバッグWIRE

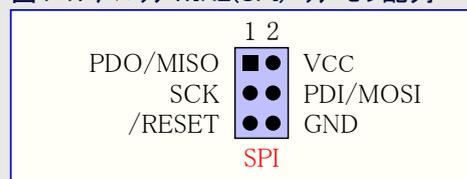
デバッグWIREインターフェースは小ピン数デバイスで使用するためにAtmelによって開発されました。4ピンを使用するJTAGインターフェースとは異なり、デバッグWIREはデバッグツールと共に双方向半二重非同期通信に単一(RESET)ピンだけを利用します。

デバッグWIREインターフェースを持つAtmel AVRを含む応用PCBの設計時、「[図4-7. デバッグWIRE\(SPI\)ヘッダピン配列](#)」で示されるピン配列が使用されるべきです。

注: デバッグWIREインターフェースはプログラミングインターフェースとして使用することができません。これは目的対象をプログラミングするために、「[図4-8. SPIヘッダピン配列](#)」で示されるようなSPIインターフェースも利用可能でなければならないことを意味します。

デバッグWIRE許可(DWEN)ヒューズがプログラム(0)され、施錠ビットが非プログラム(1)の時に、目的対象内のデバッグWIREシステムが活性(有効)にされます。RESETピンはプルアップ許可のワイヤードAND(オープンドレイン)双方向入出力ピンとして形態設定され、目的対象とデバッグ間の通信中継器になります。

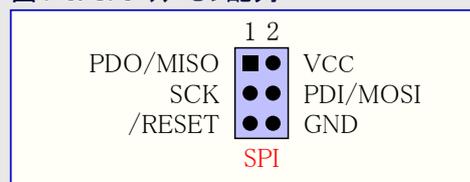
図4-7. デバッグWIRE(SPI)ヘッダピン配列



4.2.5. SPI

実装書き込み(ISP:In-System Programming)はフラッシュメモリとEEPROM内にコードを書き込むのに目的対象のAtmel AVRの内部SPI(Serial Peripheral Interface)を使用します。これはデバッグインターフェースではありません。SPIインターフェースを持つAVRを含む応用PCBの設計時、「[図4-8. SPIヘッダピン配列](#)」で示されるピン配列が使用されるべきです。

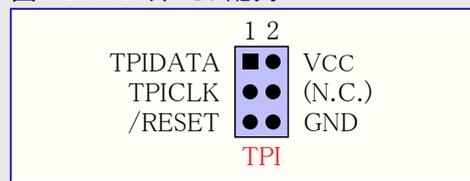
図4-8. SPIヘッダピン配列



4.2.6. TPI

TPIはいくつかのAVR ATtinyデバイス用のプログラミング専用インターフェースです。これはデバッグインターフェースではなく、これらのデバイスはOCD能力を持ちません。TPIインターフェースを持つAVRを含む応用PCBの設計時、「[図4-9. TPIヘッダピン配列](#)」で示されるピン配列が使用されるべきです。

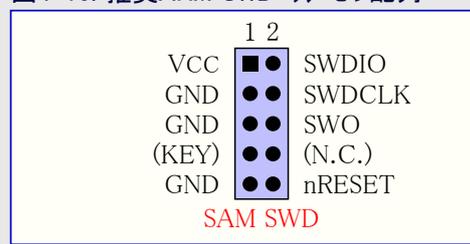
図4-9. TPIヘッダピン配列



4.2.7. SWD

ARM SWDインターフェースはJTAGインターフェースの一部で、TCKとTMSのピンを利用します。けれどもARM JTAGとAVR JTAGのコネクタはピン互換ではなく、故にSWDまたはJTAGのインターフェースを持つSAMデバイスを使用する応用PCBを設計する時に、「[図4-10. 推奨ARM SWDヘッダピン配列](#)」で示されるARMピン配列の使用が推奨されます。Atmel-ICEのASMコネクタポートはこのピン配列に直接接続することができます。

図4-10. 推奨ARM SWDヘッダピン配列



Atmel-ICEはUART形式ITM追跡をホストコンピュータへ流す能力があります。追跡は10ピンヘッダのTRACE/SWO(JTAGのTDO)ピンで捕獲されます。データはAtmel-ICEで内部的に緩衝され、HIDインターフェース上でホストコンピュータへ送られます。信頼できる最大データ速度は約3MB/sです。

4.3. Atmel OCD実装

4.3.1. Atmel AVR UC3 OCD (JTAGとaWire)

Atmel AVR UC3 OCDシステムは高い柔軟性と強力で開かれた32ビットマイクロコントローラ用のチップ上デバッグ規格であるNexus 2.0規格(IEEE-ISTO 5001™-2003)に従って設計されています。これは以下の機能を支援します。

- Nexus適合デバッグ解決策
- どのCPU速度も支援するOCD
- 6つのプログラムカウンタハードウェア中断点(ブレイクポイント)
- 2つのデータ中断点
- 監視点として形態設定できる中断点
- 範囲での中断を与えるように結合することができるハードウェア中断点
- (BREAK命令を使用する)無制限数の使用者プログラム中断点
- 実時間プログラムカウンタ分岐追跡、データ追跡、処理追跡(Atmel-ICEによって不支援)

このデバッグインターフェースに関する特別な考慮については20頁の「[Atmel AVR UC3 OCD](#)」をご覧ください

UC3 OCDシステムに関するより多くの情報についてはwww.atmel.com/uc3に置かれているAVR32UC技術参考書を調べてください。

4.3.2. Atmel AVR XMEGA OCD (JTAGとPDI物性)

Atmel AVR XMEGA OCDは他にPDI(Program and Debug Interface)として知られます。(JTAGとPDI物性の)2つの物理インターフェースがデバイス内で同じOCD実装へのアクセスを提供します。これは以下の機能を支援します。

- 完全なプログラムの流れ制御
- 1つの専用プログラムアドレス比較器またはシンボリック中断点(予約)
- 4つのハードウェア比較器
- (BREAKを使用する)無制限数の使用者プログラム中断点
- システムクロック周波数での制限なし

このデバッグインターフェースに関する特別な考慮については18頁の「[特別な考慮](#)」をご覧ください

4.3.3. Atmel megaAVR OCD (JTAG)

Atmel megaAVR OCDはJTAG物理インターフェースに基づきます。これは以下の機能を支援します。

- 完全なプログラムの流れ制御
- 4つのプログラム メモリ (ハードウェア) 中断点 (1つは予約)
- データ中断点形式に結合することができるハードウェア中断点
- (BREAKを使用する)無制限数のプログラム中断点 (ATmega128[A]を除く)

このデバッグ インターフェースに関する特別な考慮については19頁の「Atmel megaAVR OCD (JTAG)」をご覧ください

4.3.4. Atmel megaAVR/tinyAVR OCD (デバッグWIRE)

デバッグWIRE OCDは小ピン数のAtmel AVRデバイス用に特に設計された制限された機能一式を持つ特殊化したOCD部です。これは以下の機能を支援します。

- 完全なプログラムの流れ制御
- (BREAKを使用する)無制限数の使用者プログラム中断点
- 目的対象クロックに基づく自動転送速度形態設定

このデバッグ インターフェースに関する特別な考慮については19頁の「Atmel megaAVR OCD (JTAG)」をご覧ください

4.3.5. ARM CoreSight構成部

Cortex-Mに基づくAtmel ARMマイクロ コントローラはOCD構成部に適合するCoreSight™を実装します。これらの構成部の機能はデバイス毎に変わり得ます。更なる情報についてはデバイスのデータシートを調べてください。

5. ハードウェア説明

5.1. LED

Atmel-ICEの上覆いは現在のデバッグまたはプログラミングの作業の状態を示す3つのLEDを持ちます。

表5-1. LED

LED	機能	説明
左	目的対象電源	目的対象の電源がOKの時に緑。点滅は目的対象電源異常を示します。プログラミング/デバッグ作業の接続が開始するまで点灯しません。
中央	主電源	主基板の電源がOKの時に赤
右	状態	目的対象走行時に緑。目的対象停止時に橙



5.2. 背面

Atmel-ICEの背面はUSBマイクロコネクタを収めます。



5.3. 底覆い

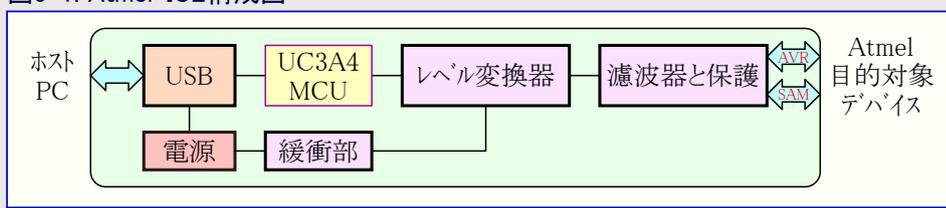
Atmel-ICEの底覆いは通番と製造日付を示す張り札を持ちます。技術支援を求める時にこれら詳細を含めてください。



5.4. 基本構造説明

Atmel-ICEの基本構造は「図5-1. Atmel-ICE構成図」で示されます。

図5-1. Atmel-ICE構成図



5.4.1. Atmel-ICE主基板

電力はUSBバスからAtmel-ICEへ供給され、降圧スイッチング調整器によって3.3Vに調整されます。VTGピンは基準電圧入力としてだけ使用され、基板上のレベル変換器の可変電圧側に独立した電力を供給します。Atmel-ICE主基板の心臓部は処理される作業に依存して1~60MHz間で走行するAtmel AVR UC3マイクロコントローラのAT32UC3A4256です。このマイクロコントローラはデバッグとの高いデータ単位処理量を許す、チップ上のUSB 2.0高速(High speed)部を含みます。

Atmel-ICEと目的対象デバイス間の通信は目的対象の動作電圧とAtmel-ICEの内部電圧レベル間の信号をシフトする一揃いのレベル変換器を通して行われます。また、信号経路には過電圧保護ツェナーダイオード、直列終端抵抗、誘導性濾波器、ESD保護ダイオードがあります。例えAtmel-ICEヘドウェアが5.0Vよりも高い電圧駆動できなくとも、全ての信号チャネルは1.62~5.5Vの範囲で動作することができます。最大動作周波数は使用する目的対象インターフェースに従って変わります。

5.4.2. Atmel-ICE目的対象コネクタ

Atmel-ICEはアクティブ探針を持ちません。直接またはいくつかのキットに含まれるアダプタのどちらかで目的対象に接続するのに50mil IDCケーブルが使用されます。配線とアダプタのより多くの情報については「[ATmel-ICEの組み立て](#)」項をご覧ください。

5.4.3. Atmel-ICE目的対象コネクタ部品番号

Atmel-ICE 50mil IDCケーブルを目的対象基板へ直接接続するため、どの標準50mil 10ピンヘッダも充分であるべきです。それらがこのキットに含まれるアダプタ基板で使用されるように、目的対象への接続時に正しい向きを保証するため、込み栓付きヘッダの使用が勧告されます。

SAMTECからのこのヘッダの部品番号はFTSH-105-01-LDV-K-A-Pです。

6. ソフトウェア統合

6.1. Atmel Studio

6.1.1. Atmel Studioでの統合

Atmel StudioはWindows環境でAtmel AVR応用を書いてデバッグするための統合開発環境(IDE: Integrated Development Environment)です。Atmel Studioはプロジェクト管理ツール、ソースファイルエディタ、シミュレータ、アセンブラとC/C++用前処理部、エミュレーション、チップ上デバッグを提供します。

Atmel Studio 6.2またはそれ以降版はAtmel-ICEと共に使用されなければなりません。

6.1.2. プログラミング任意選択

Atmel StudioはAtmel-ICEを使用してAtmel AVRとAtmel SAM ARMデバイスのプログラミングを支援します。プログラミングダイアログは選択した目的対象デバイスに従ってJTAG、aWire、SPI、PDI、TPI、SWDの動作を使用するように形態設定することができます。

クロック周波数形態設定時、各種インターフェースと目的対象システムに対して以下の各種規則が適用されます。

- SPIプログラミングは目的対象クロックを利用します。そのクロック周波数を目的対象デバイスが現在走行している周波数の1/4未満に形態設定してください。
- Atmel megaAVRデバイスでのJTAGプログラミングは書き込み器によってクロック駆動されます。これはプログラミングクロック周波数がデバイスそれ自身の最大動作周波数(通常、16MHz)に制限されることを意味します。
- JTAGとPDIの両インターフェースでのAVR XMEGAプログラミングは書き込み器によってクロック駆動されます。これはプログラミングクロック周波数がデバイスそれ自身の最大動作周波数(通常、32MHz)に制限されることを意味します。
- JTAGインターフェースでのAVR UC3プログラミングは書き込み器によってクロック駆動されます。これはプログラミングクロック周波数がデバイスそれ自身の(33MHzに制限される)最大動作周波数に制限されることを意味します。
- aWireインターフェースでのAVR UC3プログラミングは書き込み器によってクロック駆動されます。最適周波数は目的対象デバイスのSABバス速度によって与えられます。Atmel-ICEデバッグはこの判定基準に合うようにaWireのホーレートを自動的に調節します。通常必要ないとしても、使用者は(例えば、雑音環境で)必要な場合に最大ホーレートを制限することができます。
- SWDインターフェースでのSAM ARMプログラミングは書き込み器によってクロック駆動されます。Atmel-ICEで支援される最大周波数は2MHzです。周波数は目的対象CPU周波数 $\times 10$ ($f = 10 \times f_{sysclk}$)を超えるべきではありません。

6.1.3. デバッグ任意選択

Atmel Studioを使用してAtmel AVRデバイスをデバッグする時に、プロジェクトプロパティウィンドウ内の'Tool'タブはいくつかの重要な形態設定任意選択を含みます。更なる説明が必要な任意選択は以下です。

- 目的対象クロック周波数(Target Clock Frequency)

JTAGインターフェースでAtmel megaAVRデバイスの信頼に足るデバッグを達成するには目的対象クロック周波数の正確な設定が重要です。この設定はデバッグされる応用の目的対象AVRデバイスの最低動作周波数の1/4未満であるべきです。より多くの情報については19頁の「Atmel megaAVR OCD (JTAG)」をご覧ください。

デバッグWIRE目的対象デバイスでのデバッグ作業は目的対象デバイスそれ自身によってクロック駆動され、故に周波数設定は必要ありません。Atmel-ICEはデバッグ作業の開始で通信用の正しいボーレートを自動的に選択します。けれども、雑音性のデバッグ環境に関連した確実性の高い問題を経験した場合、デバッグWIRE速度をその'推奨(recommended)'設定の何分の1かに強制することが可能です。

AVR XMEGA目的対象デバイスのデバッグ作業はデバイスそれ自身の最大速度(通常32MHz)までクロック駆動することができます。

JTAGインターフェースでのAVR UC3目的対象デバイスのデバッグ作業はデバイスそれ自身の(33MHzに制限される)最大速度までクロック駆動することができます。けれども、最適周波数は目的対象デバイスの現在のSABクロックより僅かに低くなります。

aWireインターフェースでのUC3目的対象デバイスのデバッグ作業はAtmel-ICEそれ自身によって最適ボーレートへ自動的に調節されます。けれども、雑音性のデバッグ環境に関連した確実性の高い問題を経験した場合、aWire速度を形態設定可能限度以下に強制することが可能です。

SWDインターフェースでのSAM ARM目的対象デバイスのデバッグ作業はCPUクロックの10倍(最大2MHzに制限)までクロック駆動することができます。

- EEPROM保護(Preserve EEPROM)

デバッグ作業前に目的対象の書き込み中にEEPROMの消去を避けるにはこの任意選択を選択してください。

- 外部リセット使用(Use External Reset)

目的対象がJTAGインターフェースを禁止する場合、プログラミング中に外部リセットがLowに引かれなければなりません。外部リセットを使用するかどうかを繰り返し問われるのを避けるにはこの任意選択を選択してください。

7. コマンド行ユーティリティ

Atmel StudioはAtmel-ICEを使用して目的対象をプログラミングするのに使用することができるatprogramと呼ばれるコマンド行ユーティリティと共に来ます。Atmel Studioのインストール中にスタートメニューのAtmelフォルダ内にAtmel Studio 6.2 Command Promptと呼ばれるショートカットが作成されます。このショートカットのダブルクリックにより、コマンドプロンプトが開き、プログラミング命令を入力することができます。コマンド行ユーティリティはAtmel Studioインストールパス内のAtmel¥Atmel Studio 6.2¥atbackend¥フォルダ内にインストールされます。

コマンド行ユーティリティのより多くのヘルプを得るには命令: `atprogram --help` を入力してください。

8. 高度なデバッグ技術

8.1. Atmel AVR UC3目的対象

8.1.1. EVTI/EVTOの使い方

EVTIとEVTOのピンはAtmel-ICEでアクセスできません。けれども、それらは他の外部装置と共に未だ使用することができます。

EVTIは以下の目的で使用することができます。

- 目的対象は外部事象への応答で実行の停止を強制することができます。DCレジスタ内の制御での事象(EIC:Event In Control)ビットが'01'を書かれる場合、EVTIピンでのHigh⇒Low遷移が中断点(ブレークポイント)状態を生成します。これが起こる時にDS内の外部中断点(EXB:External Breakpoint)ビットが設定(1)されます。
- 追跡同期メッセージ生成。Atmel-ICEによって使用されません。

EVTOは以下の目的で使用することができます。

- CPUがデバッグ動作へ移行したことを指示。DC内のEOSビットの'01'設定は、目的対象デバイスがデバッグ動作移行時に1CPUクロック周期間Lowへ引かれることをEVTOピンに起こします。この信号は外部オシロスコープ用の起動元として使用することができます。
- CPUが中断点または監視点に到達したことを指示。対応する中断点/監視点制御レジスタ内のEOCビットの設定により、中断点または監視点の状態がEVTOピンで示されます。DC内のEOSビットはこの機能を許可するために'10'に設定されなければなりません。そして監視点タイミングを調べるためにEVTOピンは外部オシロスコープに接続することができます。
- 追跡タイミング信号生成。Atmel-ICEによって使用されません。

8.2. デバッグWIRE目的対象

8.2.1. ソフトウェア中断点

デバッグWIRE OCDはAtmel megaAVR (JTAG) OCDと比べた時に徹底的に縮小されています。これはデバッグ目的に使用者に対して利用可能などんなプログラム カウンタ中断点(ブレークポイント)比較器も持たないことを意味します。カーソルまで実行や一段実行の目的用にこのような比較器が1つ存在しますが、使用者中断点はハードウェアで支援されません。

代わりに、デバッグはAtmel AVRのBREAK命令を利用しなければなりません。この命令はフラッシュメモリに置かれ、それが実行のために読み込まれた時にAVR CPUを停止動作へ移行させます。デバッグ中に中断点を支援するため、デバッグは使用者が求める中断点の点でフラッシュメモリ内にBREAK命令を挿入しなければなりません。元の命令は後で置き換えるために貯えられなければなりません。BREAK命令での一段実行時、デバッグはプログラムの動きを維持するために貯えられた元の命令を実行しなければなりません。極端な場合には、BREAKがフラッシュメモリから取り去られて後で置換されなければなりません。これら全ての筋書は中断点からの一段実行時に明白な遅延を引き起こし得て、目的対象クロック周波数が非常に低い時に悪化されるでしょう。

故に可能な以下の指針を守ることが推奨されます。

- デバッグ中、常に可能な限り高い周波数で目的対象を動かしてください。デバッグWIRE物理インターフェースは目的対象クロックからクロック駆動されます。
- 各1つが目的対象で置換されるべきフラッシュ ページを必要とするため、中断点の追加と削除を最小とするよう試みてみてください。
- フラッシュ ページの書き込み操作数を最小にするため、同時に少数の中断点を追加または削除するよう試みてください。
- 可能なら、2語命令に中断点を配置することを避けてください。

9. 特別な考慮

9.1. Atmel AVR XMEGA OCD

OCDとクロック駆動

MCUが停止動作へ移行すると、MCUクロックとしてOCDクロックが使用されます。OCDクロックはJTAGインターフェースが使用されている場合にJTAGのTCK、またはPDIインターフェースが使用されている場合にPDI_CLKのどちらかです。

停止動作での入出力単位部

初期のAtmel megaAVRデバイスと対照的に、XMEGAの入出力単位部は停止動作で停止されます。これはUSART送信が中断され、計時器が停止されることを意味します。

ハードウェア中断点

2つのアドレス比較器と2つの値比較器で4つのハードウェア中断点(ブレークポイント)比較器があります。これらは或る制限を持ちます。

- 全ての中断点は同じ形式(プログラムまたはデータ)でなければなりません。
- 全てのデータ中断点は同じメモリ領域(I/O、SRAM、またはXRAM)でなければなりません。
- アドレス範囲が使用される場合は1つの中断点だけしかできません。

以下は設定し得る各種組み合わせです。

- 2つの単一データまたはプログラム アドレスの中断点
- 1つのデータまたはプログラム アドレスの範囲中断点
- 単一値比較を持つ2つの単一データ アドレス中断点
- アドレス範囲、値範囲、または両方を持つ1つのデータ中断点

Atmel Studioは中断点を設定できない場合に何故かを告げます。ソフトウェア中断点が利用可能な場合、データ中断点がプログラム中断点より上の優先権を持ちます。

外部リセットとPDI物性

PDI物性インターフェースはクロックとしてリセット線を使用します。デバッグ時、リセットのプルアップは10kΩに、より大きく、または取り外されるべきです。どのリセット コンデンサも取り去られるべきです。他の外部リセット元は切断されるべきです。

ATxmegaA1改訂Hとそれ以前版に対する休止でのデバッグ

ATxmegaA1系の初期版にはデバイスが或る休止動作中に許可されたOCDが妨げられるバグが有りました。デバッグに戻るのに使用する2つの方法があります。

- ToolsメニューでAtmel-ICE Optionsへ行き、”Always activate external reset when reprogramming device”を許可してください。
- チップ消去を実行してください。

このバグを引き起こす休止動作は以下です。

- パワーダウン
- パワーセーブ
- スタンバイ
- 拡張スタンバイ

9.2. Atmel megaAVR OCDとデバッグWIRE OCD

入出力周辺機能

殆どの周辺機能は例えプログラム実行が中断点(ブレークポイント)によって停止されても、走行を続けます。例: UART送信中に中断点に到達した場合、その送信は完了されて対応するビットが設定されます。送信完了(TXC)フラグが設定(1)され、例え実際のデバイスでは通常、より遅くに起こるとしても、コードの次の単一段実行で利用可能になります。

全ての入出力単位部は次の2つの例外付きで停止動作で走行を継続します。

- ・ タイマ/カウンタ (ソフトウェア前処理部を使用して形態設定可能)
- ・ ウォッチドッグ タイマ (デバッグ中のリセットを防ぐため、常に停止)

単一段実行の入出力アクセス

入出力が停止動作で走行を続けるため、或るタイミングの問題を避けるために注意が払われるべきです。例えば次のコードです。

```
OUT    PORTB, $AA
IN     TEMP, PINB
```

このコードの通常走行時、データはIN操作によって採取される時に物理的に未だピンにラッチされていないため、TEMPレジスタは\$AAを読み戻しません。PINレジスタに正しい値が存在するのを保証するには、OUTとINの命令間にNOP命令が置かれなければなりません。

けれども、OCDを通してこの機能を単一段実行すると、例え単一段実行中にコアが停止されていても、入出力が全速で走行するので、このコードは常に\$AAを与えます。

単一段実行とタイミング

或るレジスタは制御信号許可後に与えられた周期数内に読みまたは書きが必要です。停止動作でI/Oクロックと周辺機能が全速度走行を続けるため、そのようなコードを通した単一段実行はタイミング必要条件に合致しないでしょう。2つの単一段実行間で、I/Oクロックは100万周期走行するかもしれません。このようなタイミング必要条件で成功裏のレジスタを読みまたは書きを行うには、全速でデバイスを走らせる非分断操作として読みまたは書きの全体手順が実行されるべきです。これはコードを実行するのにマクロまたは関数を使用する、またはデバッグ環境でカーソルまで実行の機能を使用することによって行うことができます。

16ビットレジスタのアクセス

Atmel AVR周辺機能は代表的に8ビットデータバス経由でアクセスすることができる多数の16ビットレジスタ(例えば、16ビットタイマ/カウンタのTCNTn)を含みます。16ビットレジスタは2つの読みまたは書きの操作を用いてバイトアクセスされなければなりません。16ビットアクセスの間での中断、またはこの状況を通しての単一段実行は誤った値に帰着するかもしれません。

制限されたI/Oレジスタアクセス

或るレジスタはそれらの内容に影響を及ぼさずに読むことができません。このようなレジスタは読むことによって解除(0)されるフラグを含むそれらや、緩衝されたデータレジスタ(例えば、UDR)を含みます。OCDデバッグの意図された非侵襲的性質を守るため、ソフトウェア前処理部は停止動作時にこれらのレジスタの読み込みを防ぎます。加えて、いくつかのレジスタは副作用を起こすことなく、安全に書くことができません。これらのレジスタは読み出し専用です。例えば、以下です。

- ・ 何れかのビットへの'1'書き込みによって解除(0)されるフラグのフラグレジスタ
- ・ UDRとSPDRは単位部の状態に影響を及ぼすことなく読むことができません。これらのレジスタはアクセス不能です。

9.2.1. Atmel megaAVR OCD (JTAG)

ソフトウェア中断点

ATmega128[A]は初期版のOCD単位部を含むため、ソフトウェア中断点(ブレークポイント)用BREAK命令の使用を支援しません。

JTAGクロック

目的対象クロック周波数はデバッグ作業を始める前にソフトウェア前処理部で正確に指定されなければなりません。同期化の理由に関して、JTAGのTCK信号は信頼性に足るデバッグのために目的対象クロック周波数の1/4未満でなければなりません。JTAGインターフェース経由のプログラミング時、TCK周波数は目的対象デバイスの最大周波数評価によって制限され、使用される実際のクロック周波数ではありません。

内部RC発振器使用時、周波数がデバイス毎に変わって、温度やVCCの変化によって影響を及ぼされることに注意してください。目的対象クロック周波数を指定する時は控え目に(確実性を高く)してください。

ソフトウェア前処理部を使用して目的対象クロック周波数を設定する方法の詳細については17頁の「デバッグ任意選択」をご覧ください。

JTAGENとOCDENのヒューズ

JTAGインターフェースは既定によってプログラム(0)されているJTAGENヒューズによって許可されます。これはJTAGプログラミングインターフェースへのアクセスを許します。この機構を通して、OCDENヒューズをプログラム(0)することができます(既定でのOCDENは非プログラム(1))。これはデバイスのデバッグを容易にするためにOCDへのアクセスを許します。ソフトウェア前処理部は作業終了時にOCDENヒューズが非プログラム(1)にさせられることを常に保証し、それによってOCD単位部による不必要な電力消費を制限します。JTAGENヒューズが意図せずに禁止された場合、SPIまたはPPのプログラミング法を使用してのみ再許可することができます。

JTAGENヒューズがプログラム(0)されたなら、JTAGインターフェースはJTDビットを設定(1)することによって未だファームウェアで禁止することができます。これはデバッグ不可コードにし、デバッグ作業を試みる時に実行されべきではありません。デバッグ作業開始時にこのようなコードが既に実行された場合、Atmel-ICEは接続中にRESET線を有効にします。この線が正しく配線されているなら、目的対象AVRデバイスをリセットに強制し、それによってJTAG接続を許します。

JTAGインターフェースが許可された場合、JTAGピンは代替ピン機能に使用することができません。それらはプログラムコードからJTDビットの設定(1)、またはプログラミングインターフェースを通してJTAGENヒューズの解除(1)のどちらかによってJTAGインターフェースが禁止されるまでJTAG専用ピンに留まります。

注: JTDビットの設定(1)によってJTAGインターフェースを禁止するコードを走行するデバイスでRESET線を有効にしてJTAGインターフェースを再許可することをAtmel-ICEに許すために、プログラミングダイアログとデバッグ任意選択ダイアログの両方で”use external reset”チェックボックスのチェックを確実にしてください。

IDR事象

応用プログラムがデバッグされつつあるAVRデバイスのOCDレジスタにバイトデータを書く時に、Atmel-ICEはこの値を読み出してそれをソフトウェア前処理部のメッセージウィンドウに表示します。IDRレジスタは50ms毎にポーリングされ、故により高い繰り返しでの書き込みは信頼に足る結果を生じません。デバッグ中にAVRデバイスが電力を失うと、偽のIDR事象が報告され得ます。これは目的対象電圧がAVRの最低動作電圧以下に落ちる時にAtmel-ICEが未だデバイスをポーリングし得るために起こります。

9.2.2. デバッグWIRE OCD

デバッグWIRE(dW)ピンは物理的に外部リセット(RESET)と同じピンに配置されます。従ってデバッグWIREインターフェースが許可される時に外部リセット元は支援されません。

デバッグWIREインターフェースが機能するためには、目的対象デバイスでデバッグWIRE許可(DWEN)ヒューズが設定(0)されなければなりません。このヒューズはAtmel AVRデバイスが工場から出荷される時に既定によって非プログラム(1)にされます。デバッグWIREインターフェースそれ自身はこのヒューズを設定することができません。DWENヒューズを設定するにはSPI動作が使用されなければなりません。ソフトウェア前処理部は必要なSPIピンが接続されていればこれを自動的に処理します。Atmel StudioのプログラミングダイアログからSPIプログラミングを使用することもできます。以下の2つのどちらかを行います。

- デバッグWIRE部でデバッグ作業の開始を試みてください。デバッグWIREインターフェースが許可されていない場合は、Atmel Studioは再試行を提供するか、またはSPIプログラミングを使用してデバッグWIREを許可しようと試みます。完全なSPIヘッダ接続があれば、デバッグWIREが許可され、目的対象で電源のOFF/ONを問われるでしょう。これは効果を得るべくヒューズを変更するのに必要とされます。
- SPI動作でプログラミングダイアログを開き、識票が正しいデバイスと一致することを確認してください。デバッグWIREを許可するためにDWENヒューズをチェックしてください。

注: SPIENヒューズがプログラム(0)され、RSTDISBLヒューズが非プログラム(1)のままにされていることが重要です。これを行わないことはデバッグWIRE動作でデバイスを動作停止にし、DWEN設定を元に戻すの高電圧プログラミングが必要とされます。

デバッグWIREインターフェースを禁止するには、DWENヒューズを非プログラム(1)にするのに高電圧プログラムを使用してください。代わりに、デバッグWIREを一時的に禁止するのにデバッグWIREそれ自身を使用し、SPIENヒューズが設定(0)されていれば、SPIプログラミング実行を許します。

注: SPIENヒューズがプログラム(0)のままにされていない場合、Atmel Studioはこの操作を緩衝することができず、高電圧プログラミングが使用されなければなりません。

- デバッグ作業の間、'Debug'メニューから'Disable debugWIRE and Close'メニュー任意選択を選んでください。デバッグWIREが一時的に禁止され、Atmel StudioはDWENヒューズを非プログラム(1)にするのにSPIプログラミングを使用します。

プログラム(0)されたDWENヒューズがあることは全ての休止動作で走行することをクロック系のいくつかの部分に許します。これは休止動作間のAVRの消費電力を増やします。従ってデバッグWIREが使用されない時は常にDWENヒューズが禁止されるべきです。

デバッグWIREが使用される目的対象応用PCBの設計時、正しい動作のために以下の考慮をしなければなりません。

- dW(RESET)線のプルアップ抵抗は10kΩよりも小さく(強力で)あってはなりません。プルアップ抵抗はデバッグツールがこれを提供するため、機能的にデバッグWIREに必要ではありません。
- RESETピンの直接VCC接続はデバッグWIREインターフェースを失敗させ、Atmel-ICDへのハードウェア障害に帰着するかもしれません。
- RESETピンに接続されるどの安定化コンデンサも、それらがインターフェースの正しい動作を妨げるため、デバッグWIRE使用時に切断されなければなりません。
- 全ての外部リセット元またはREEST線上の他の活性化駆動部は、それらがインターフェースの正しい動作を妨げるため、切断されなければなりません。

目的対象デバイスの施錠ビットを決してプログラム(0)してはなりません。デバッグWIREインターフェースは正しく機能するために施錠ビットの解除(1)が必要です。

9.3. Atmel AVR UC3 OCD

JTAGインターフェース

いくつかのAtmel AVR UC3デバイスでJTAGポートは既定で許可されません。これらのデバイス使用時、Atmel-ICEがJTAGインターフェースを許可することをできるようにRESET線を正しくすることが重要です。

aWireインターフェース

aWire通信のホーレートはデータがそれら2つの領域間で同期されなければならないので、システムクロックの周波数に依存します。Atmel-ICEはシステムクロックがより低いことを自動的に検知し、それによってホーレートを再校正します。自動校正は8kHzのシステムクロック周波数へ落とすように動くだけです。デバッグ作業中のより低いシステムクロックへの切り替えは目的対象との交信を失わせるかもしれません。

必要とされるなら、aWireのポーレートはaWireクロック項目を設定することによって制限することができます。自動検出は未だ動きますが、上限値は結果を強制されるでしょう。

RESETピンに接続されるなどの安定化コンデンサも、それらがインターフェースの正しい動作を妨げるため、aWire使用時に切断されなければなりません。この線上の弱い(10kΩまたはより高い)外部プルアップが推奨されます。

遮断休止動作

いくつかのAVR UC3デバイスは1.8Vに調整された入出力線とで3.3V供給動作で使用することができる内部調整器を持ちます。これは内部調整器がコアと殆どの入出力の両方に給電することを意味します。Atmel-ICEはこの調整器がOFFにされる遮断休止動作(Shutdown sleep mode)を支援しません。換言すると、この休止動作はデバッグ中に使用することができません。デバッグ中にこの休止動作の使用が必要条件なら、代わりにAtmel AVR ONE!デバッグを使用してください。

9.4. SAM/CoreSight OCD

いくつかのSAMデバイスは完全なチップ消去と保護ビットが設定されたデバイスの解錠の実行を有効にするERASEピンを含みます。このピンはどのデバッグヘッダにも配線されず、故にAtmel-ICEはデバイスの解錠が不可能です。このような場合、使用者はデバッグ作業を開始する前に消去を実行すべきです。

JTAGインターフェース

RESET線はAtmel-ICEがJTAGインターフェースを許可することができるよう、常に接続されるべきです。

SWDインターフェース

RESET線はAtmel-ICEがSWDインターフェースを許可することができるよう、常に接続されるべきです。

10. ファームウェア格上げ更新

ファームウェアを格上げ更新する方法の情報についてはAtmel Studio(使用者の手引き)で[Atmel Studio使用者の手引き](#)をご覧ください。

11. 公開履歴と既知の問題

11.1. 新規情報

Atmel-ICEは新製品!

11.2. ファームウェア改訂履歴

11.2.1. Atmel Studio 6.2

表11-1. 本公開での新規事項

公開基盤	Atmel Studio 6.2 (最終版)
ファームウェア版	1.13
新機能	なし
修正	<ul style="list-style-type: none">発振器校正命令を修正デバッグWIREの信頼性を改善

11.2.2. Atmel Studio 6.2 β

表11-2. 本公開での新規事項

公開基盤	Atmel Studio 6.2 (β)
ファームウェア版	1.09
新機能	Atmel-ICEの初版公開
修正	N/A

11.3. Atmel-ICEに関する既知の問題

11.3.1. Atmel AVR XMEGA OCD特有問題

・ ATxmegaA1系については改訂Gまたはそれ以降版だけが支援されます。

11.3.2. Atmel megaAVR/tinyAVR OCD特有問題

・ デバッグ作業中のATmega32U6での電源ON繰り返しはデバイスとの通信中断を引き起こすかもしれません。

11.4. 支援デバイス

全てのAtmelツールに対する完全な支援デバイス表についてはAtmel Studio(使用者の手引き)で”[支援デバイス](#)”をご覧ください。

12. 製品適法性

12.1. RoHSとWEEE

Atmel-ICE(全キット)とその付属品はRoHS指令(2002/95/EC)とWEEE指令(2002/96/EC)の両方に従って製造されます。

12.2. CEとFCC

Atmel-ICE本体は以下のような必須の必要条件とその他の関連指令条項に従って検査されています。

- 2004/108EC(B級)指令
- FCC区分15副区分B
- 2002/95/EC(RoHS,WEEE)

評価には以下の規格が使用されます。

- EN 61000-1 (2007)
- EN 61000-3 (2007) + A1 (2011)
- FCC CFR 47区分15 (2013)

技術構成ファイルは以下に置かれます。

Atmel Norway
Vestre Rosten 79
7075 Tiller
Norway

全ての努力がこの製品からの電磁放射を最小にさせました。けれども、或る条件下で、システム(目的対象応用回路に接続された本製品)は前述の規格によって許される最大値を超える個別電磁成分周波数を放射するかもしれません。この放射の周波数と大きさは使用される製品と目的対象応用の配置と配線を含む様々な要素によって決められます。

13. 資料改訂

資料改訂	日付	注釈
42330A	2014年7月	初版資料公開

Atmel®, Atmelロゴとそれらの組み合わせ、Enabling Unlimited Possibilities®, AVR®, AVR Studio®, megaAVR®, tinyAVR®, XMEGA®とその他は米国及び他の国に於けるAtmel Corporationの登録商標または商標です。ARM®, ARM Connected®, Cortex®ロゴとその他はARM Ltd.の登録商標または商標です。Windows®は米国と他の国に於けるMicrosoft Corporationの登録商標です。他の用語と製品名は一般的に他の商標です。

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のウェブサイトに表示する販売の条件と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、または利益と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。

安全重視、軍用、車載応用のお断り: Atmel製品はAtmelが提供する特別に書かれた承諾を除き、そのような製品の機能不全が著しく人に危害を加えたり死に至らしめることがかなり予期されるどんな応用(“安全重視応用”)に対しても設計されず、またそれらとの接続にも使用されません。安全重視応用は限定なしで、生命維持装置とシステム、核施設と武器システムの操作の装置やシステムを含みます。Atmelによって軍用等級として特に明確に示される以外、Atmel製品は軍用や航空宇宙の応用や環境のために設計も意図もされていません。Atmelによって車載等級として特に明確に示される以外、Atmel製品は車載応用での使用のために設計も意図もされていません。

© HERO 2015.

本応用記述はAtmelのAtmel-ICE使用者の手引き(Rev.42330A-7/2014)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。