書き込み器とﾃﾞﾊﾞｯｶﾞ

AVR ONE!

使用者の手引き

Atmel AVR ONE!ﾃﾞﾊﾞｯｶﾞ

Atmel® AVR ONE!はAtmel AVR® 8ﾋﾞｯﾄと32ﾋﾞｯﾄMCUﾃﾞﾊﾞｲｽのﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾁｯﾌﾟ上ﾃﾞﾊﾞｯｸﾞ用の強力な開発ﾂｰﾙです。こ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AVR UC3ﾏｲｸﾛ ｺﾝﾄﾛｰﾗとﾌﾟﾛｾｯｻのﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾁｯﾌﾟ上ﾃﾞﾊﾞｯｸﾞ
- AUXﾎﾟｰﾄを持つ全てのAVR UC3ﾏｲｸﾛ ｺﾝﾄﾛｰﾗとﾌﾟﾛｾｯｻのNexus AUX追跡でのﾁｯﾌﾟ上ﾃﾞﾊﾞｯｸﾞ
- JTAGと2線PDIの両ｲﾝﾀｰﾌｪｰｽで全てのAVR XMEGA®系ﾃﾞﾊﾞｲｽのﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾁｯﾌﾟ上ﾃﾞﾊﾞｯｸﾞ
- JTAGとﾃﾞﾊﾞｯｸﾞWIREのどちらかのｲﾝﾀｰﾌｪｰｽでのOCD支援を持つ全てのAtmel megaAVR®とtinyAVR®ﾏｲｸﾛ ｺﾝﾄﾛｰﾗの(JTAGとSPIの)ﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾃﾞﾊﾞｯｸﾞ

Rev. 32222A-06/2016, 32222AJ1-01/2020

目次

1. 序説

1.1. Atmel AVR ONE!の紹介

Atmel AVR ONE!は全てのAtmel AVR 8ﾋﾞｯﾄ/32ﾋﾞｯﾄ ﾏｲｸﾛ ｺﾝﾄﾛｰﾗのﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾁｯﾌﾟ上ﾃﾞﾊﾞｯｸﾞ用の強力な開発ﾂｰﾙです。これは以下を支援します。

- JTAGとaWireの両ｲﾝﾀｰﾌｪｰｽでの全てのAVR UC3ﾏｲｸﾛ ｺﾝﾄﾛｰﾗとﾌﾟﾛｾｯｻのﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾁｯﾌﾟ上ﾃﾞﾊﾞｯｸﾞ
- AUXﾎﾟｰﾄを持つ全てのAVR UC3ﾏｲｸﾛ ｺﾝﾄﾛｰﾗとﾌﾟﾛｾｯｻのNexus AUX追跡でのﾁｯﾌﾟ上ﾃﾞﾊﾞｯｸﾞ
- JTAGと2線PDIの両ｲﾝﾀｰﾌｪｰｽでの全てのAVR XMEGA系ﾃﾞﾊﾞｲｽのﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾁｯﾌﾟ上ﾃﾞﾊﾞｯｸﾞ
- JTAGやﾃﾞﾊﾞｯｸﾞWIREの両ｲﾝﾀｰﾌｪｰｽでのOCD支援を持つ全ての8ﾋﾞｯﾄAVRﾏｲｸﾛ ｺﾝﾄﾛｰﾗのﾌﾟﾛｸﾞﾗﾐﾝｸﾞ(JTAGとSPI)とﾃﾞﾊﾞｯｸﾞ

どのﾃﾞﾊﾞｲｽが現在支援されているかを見るにはAtmel Studioの公開記述/readmeﾌｧｲﾙを読んでください。

1.2. Atmel AVR ONE!の特徴

- Atmel Studio、あVR3Studio、AVR Studio 4、AVR Studio 5に完全適合
- 全てのAtmel AVR UC3とAVR XMEGAﾃﾞﾊﾞｲｽ、OCDを持つ全てのAtmel megaAVRとAtmel tinyAVRﾃﾞﾊﾞｲｽのﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾃﾞﾊﾞｯｸﾞを支援
- AUXﾎﾟｰﾄを持つ全てのUC3ﾃﾞﾊﾞｲｽでのAUX追跡を支援
- 循環、直線、または伸縮の追跡緩衝部として使うための基板上の128Mﾊﾞｲﾄ DDR-SDRAM
- 1.65～5.5Vの目的対象動作電圧範囲
- 動作中に目的対象のVTrefから1mA未満の引き出し(消費)
- 32kHz～33MHzの通信ｸﾛｯｸ周波数を支援
- USB2.0高速(High-speed)ﾎｽﾄ ｲﾝﾀｰﾌｪｰｽ
- MICTOR-38と10ﾋﾟﾝJTAGの両ｺﾈｸﾀだけでなく、ｱﾀﾞﾌﾟﾀを使ってaWire、PDI、SPI、ﾃﾞﾊﾞｯｸﾞWIREのｲﾝﾀｰﾌｪｰｽも支援

1.3. システム要件

Atmel AVR ONE!本体はｺﾝﾋﾟｭｰﾀにｲﾝｽﾄｰﾙされた前処理ﾃﾞﾊﾞｯｸﾞ環境(AVR32 StudioやAVR Studio 4.15またはそれ以降、またはAtmel Studio)と連携するﾕｰﾃﾘﾃｨが必要です。これら一括のｼｽﾃﾑ要件についてはwww.atmel.comで調べてください

AVR ONE!本体は提供されたUSBｹｰﾌﾞﾙを使ってﾎｽﾄ ｺﾝﾋﾟｭﾀに接続されなければなりません。AVR ONE!から絶え間なく続く追跡読み出しを達成するには、ｺﾝﾋﾟｭｰﾀのUSBﾎﾟｰﾄがUSB2.0高速(High-speed)適合でなければなりません。ｺﾝﾋﾟｭｰﾀがUSB1.1(または全速(Full-speed)USB)だけを支援する場合、AVR ONE!は未だﾃﾞﾊﾞｯｸﾞに使うことができますが(**注**: 制限適用、「**既知の問題**」をご覧ください)、追跡は緩衝動作での操作に制限されるでしょう。

AVR ONE!本体は本ｷｯﾄに含まれる12V外部電源に接続されなければなりません。

2. 公開履歴、新機能

2.1. 新規情報

本公開での新規情報

公開基盤	Atmel Studio 6.0
ﾌｧｰﾑｳｪｱ版	MCU : 5.21 ($0515) AVR32ｲﾒｰｼﾞ : 4.1 ($0401) XMEGAｲﾒｰｼﾞ : 3.2 ($0302) TMEGAｲﾒｰｼﾞ : 2.1 ($0201)
新機能	なし
修正	些細な内部ﾊﾞｸﾞ修正

2.2. ファームウェア公開履歴

表2-1. 過去の公開

公開基盤	AVR Studio 5.1	
ファームウェア版	MCU : 5.20 ($0514) AVR32イメージ : 4.1 ($0401) XMEGAイメージ : 3.2 ($0302) TMEGAイメージ : 2.1 ($0201)	
新機能	Atmel AVR XMEGAデバイスでの上位SUT値に対する支援	
修正	・aWire自動ボーレート計算改善 ・(低電圧で見られる)Atmel AVR XMEGAフラッシュ ページ プログラミング異常を修正 ・改善されたデバッグWIRE単一実行(シングル ステップ)性能	

公開基盤	AVR Studio 5.0
ファームウェア版	MCU : 5.13 ($050D) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)
新機能	なし
修正	Atmel AVR XMEGAソフトウェア リセット処理を修正

公開基盤	AVR Studio 5.0 β
ファームウェア版	MCU : 5.11 ($050B) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)
新機能	なし
修正	改善されたaWire速度

公開基盤	AVR Studio 5 β
ファームウェア版	MCU : 5.6 ($0506) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)
新機能	なし
修正	AVR Studio 5前処理部の支援を追加

公開基盤	AVR Studio 4.18 SP3	AVR32 Studio 2.6.0
ファームウェア版	MCU : 4.15 ($040F) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
ファームウェア版	MCU : 4.16 ($0410) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
新機能	なし	なし
修正	特別なHVE MUL命令の支援を追加	・停止動作形態(Atmel AVR UC3L)の支援を追加 ・(或るLinux®版で)FPGAイメージが読み込まれなかった時のデバイス切断を修正

次頁へ続く

表2-1 (続き). 過去の公開

公開基盤	AVR Studio 4.18 SP2	AVR32 Studio 2.5.0
ファームウェア版	MCU : 4.12 ($040C) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
ファームウェア版	MCU : 4.12 ($040C) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
新機能	なし	なし
修正	2語命令でのソフトウェア中断点からの走行を修正	なし

公開基盤	AVR Studio 4.18 SP1 PP1	AVR32 Studio 2.4.0
ファームウェア版	MCU : 4.11 ($040B) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
ファームウェア版	MCU : 4.11 ($040B) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
新機能	なし	なし
修正	なし	10ピンJTAGコネクタでのEVTO感知を修正

公開基盤	AVR Studio 4.18 SP1	AVR32 Studio 2.4.0
ファームウェア版	MCU : 4.11 ($040B) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
ファームウェア版	MCU : 4.11 ($040B) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
新機能	なし	なし
修正	EEPROMメモリ割り当て配置時のAVR XMEGA EEPROM読み書きを修正	10ピンJTAGコネクタでのEVTO感知を修正

公開基盤	–	AVR32 Studio 2.3.1
ファームウェア版	MCU : 4.9 ($0409) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
ファームウェア版	MCU : 4.9 ($0409) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
新機能	–	なし
修正	–	3.3V以上の電圧走行する目的対象への接続に関する問題を修正

次頁へ続く

表2-1 (続き). 過去の公開

公開基盤	AVR Studio 4.18	AVR32 Studio 2.3
ファームウェア版	MCU : 4.8 ($0408) AVR32イメージ : 3.1 ($0301) XMEGAイメージ : 2.1 ($0201) TMEGAイメージ : 1.7 ($0107)	
ファームウェア版	MCU : 4.8 ($0408) AVR32イメージ : 4.0 ($0400) XMEGAイメージ : 3.0 ($0300) TMEGAイメージ : 2.0 ($0200)	
新機能	より長いディージー チェーンに対する支援。最大ディージーチェーンは今や240命令レジスタ ビットです。	• より長いディージー チェーンに対する支援。最大ディージーチェーンは今や240命令レジスタ ビットです。 • XMEGAに対する発信器校正を支援
修正	• 低クロック周波数で走行する目的対象のデバッグに対する改善された支援 • ハードウェア中断点削除時にハードウェア中断点を失う結果となるバグを修正 • Atmel megaAVRのJTAGに対するデバッグ動作でフラッシュ読み込み時のソフトウェア中断点遮蔽を修正	改善されたaWireボーレート設定

公開基盤	AVR Studio 4.17
ファームウェア版	MCU : 3.16 ($0310) AVR32イメージ : 3.1 ($0301) XMEGAイメージ : 2.1 ($0201) TMEGAイメージ : 1.7 ($0107)
新機能	Atmel tinyAVRとAtmel megaAVRデバイスに対する支援
修正	リセット線の負荷問題修正。Atmel AVR ONE!は電源ONである限り、今や目的対象のリセット線の負荷になりません。

公開基盤	AVR32 Studio 2.2
ファームウェア版	MCU : 3.6 ($0306) AVR32イメージ : 3.0 ($0300) XMEGAイメージ : 2.0 ($0200) TMEGAイメージ : 1.3 ($0103)
新機能	aWireインターフェースに対する支援
修正	リセット線の負荷問題修正。Atmel AVR ONE!は電源ONである限り、今や目的対象のリセット線の負荷になりません。

公開基盤	AVR Studio 4.16
ファームウェア版	MCU : 2.0B AVR32イメージ : 1.09 ($0109) XMEGAイメージ : 1.04 ($0104)
新機能	なし
修正	• 8MHzに設定したJTAGクロックでのJTAGプログラミングの問題を修正 • 幻の中断点に帰着するソフトウェア中断点を修正 • Atmel AVR XMEGAに対する単一実行(シングル ステップ、外側実行)のバグを修正

公開基盤	AVR Studio 4.15	AVR32 Studio 2.1.0
ファームウェア版	MCU : 2.07 ($0207) MCU : 2.06 ($0206) AVR32イメージ : 1.09 ($0109) XMEGAイメージ : 1.03 ($0103)	
新機能	Atmel AVR XMEGA系統を支援	なし
修正	なし、これが新規公開	微小なバグ修正

公開基盤	AVR32 Studio 2.0
ファームウェア版	MCU : 1.01 ($0101) AVR32イメージ : 1.01 ($0101)
新機能	Atmel AVR 32ビット マイクロ コントローラを支援
修正	なし

3. 既知の問題

3.1. 全般

- USB1.1またはUB2.0の全速(Full-speed)使用時にﾌｧｰﾑｳｪｱ更新が動作しません。将来のGNUﾂｰﾙﾁｪｰﾝAVR Studio 4公開で修正版が配布されるまでの対策は、更新を実行するのにUSB2.0ﾎｽﾄを使うことです。
- AVR ONE!が目的対象に接続されているが、給電されていない時に、それはRESET線の受動性負荷になるでしょう。これは意図せずに目的対象ﾃﾞﾊﾞｲｽをﾘｾｯﾄに保持させるかもしれません。目的対象ﾃﾞﾊﾞｲｽに接続されている時は常にｴﾐｭﾚｰﾀに通電してください。RESETはﾌｧｰﾑｳｪｱ更新実行時や、目的対象系統型式変更時のFPGA再構成時にも僅かに負荷を与えられます。或る場合に於いて、これは目的対象ﾃﾞﾊﾞｲｽをﾘｾｯﾄにさせるかもしれません。
- ATmega128が連鎖内の先頭ﾃﾞﾊﾞｲｽの場合を除き、ATmega128がﾃﾞｨｰｼﾞｰ ﾁｪｰﾝの一部の時にﾃﾞｨｰｼﾞｰ ﾁｪｰﾝの自動検出は動かないでしょう。これはJTAG命令のIDCODEがこのﾃﾞﾊﾞｲｽで正しく動かないためです。より多くの情報についてはATmega128ﾃﾞｰﾀｼｰﾄの障害情報章をご覧ください。
- AVR Studio 4.15とそれ以降で公開されるAVR ONE!ﾌｧｰﾑｳｪｱはAVR32 Studio 2.1とそれ以降版とでだけ動作します。AVR Studio 4.15から更新したAVR ONE!を持ち、そしてAVR32 Studio 2.0版とでの動作を必要とする場合、ﾌｧｰﾑｳｪｱはAVR32 Studioでの更新を手動で開始することによって後退更新することできます。再びAVR Studio 4とでの作業を望む時はﾌｧｰﾑｳｪｱ更新が自動的に開始されます。

3.2. Atmel AVR XMEGA OCD特有の問題

- ATxmegaA1系列に関して、改訂Gまたはそれ以降だけが支援されます。
- Atmel AVR XMEGA系統に関するJTAGでのLiveﾃﾞﾊﾞｯｸﾞ支援が時々失敗します(目的対象がﾘｾｯﾄになります)。Liveﾃﾞﾊﾞｯｸﾞに対してはPDIが好ましいｲﾝﾀｰﾌｪｰｽです。離脱(切断)時、目的対象は外部ﾘｾｯﾄを必要とする停止動作状態のままになるかもしれません。

3.3. Atmel megaAVR OCDとAtmel tinyAVR OCD特有の問題

- ﾃﾞﾊﾞｯｸﾞ作業中のATmega32U6での電源ON繰り返しはﾃﾞﾊﾞｲｽとの同期消失を引き起こすかもしれません。
- GCCで生成されたｺｰﾄﾞのソース ﾚﾍﾞﾙでの単一実行(ｼﾝｸﾞﾙ ｽﾃｯﾌﾟ)は常に可能でないかもしれません。最良の結果のために最適化段階を最低に設定し、必要とする時に逆ｱｾﾝﾌﾞﾘ ｳｨﾝﾄﾞｳを使ってください。
- 低すぎる(代表的に100kHz以下の)目的対象ｸﾛｯｸ(JTAG用のﾃﾞﾊﾞｯｸﾞとﾌﾟﾛｸﾞﾗﾐﾝｸﾞのｸﾛｯｸ)、または低い周波数でのﾃﾞﾊﾞｯｸﾞWIRE目的対象の走行は、それらの低周波数使用時にAtmel AVR ONE!ｴﾐｭﾚｰﾀで或る命令を完了するのにかかる時間を処理に対して制限時間が短くなるので、Atmel Studioで異常に帰着するかもしれません。この問題は目的対象ﾃﾞﾊﾞｲｽのﾒﾓﾘ容量増加や活性なｿﾌﾄｳｪｱ中断点数の増加で増します。

3.4. Atmel AVR 32ﾋﾞｯﾄ ﾏｲｸﾛ ｺﾝﾄﾛｰﾗ特有の問題

- ありません。

4. 始める前に

4.1. ｷｯﾄ内容

Atmel AVR ONE!ｷｯﾄはこれらの品目を含みます。

- 探針付きAVR ONE!本体
- USBｹｰﾌﾞﾙ (1.8m、高速(High-speed))
- 欧州と米国用主電力ｹｰﾌﾞﾙ
- 電源
- 付着した10ﾋﾟﾝ ｹｰﾌﾞﾙ付きAVR ONE!試験ｱﾀﾞﾌﾟﾀ
- 個別ｱﾀﾞﾌﾟﾀ (10ﾋﾟﾝ 100mil、6ﾋﾟﾝ 100mil、10ﾋﾟﾝ 50mil、6ﾋﾟﾝ 50mil)
- 38ﾋﾟﾝMICTORｺﾈｸﾀ試供品
- 10ﾋﾟﾝ ﾊﾞﾗ線ｹｰﾌﾞﾙ
- Atmel AVR技術ﾗｲﾌﾞﾗﾘDVD
- 開始に際しての手引き

図4-1. Atmel AVR ONE!ｷｯﾄ内容

4.2. Atmel AVR ONE!の給電

Atmel AVR ONE!は12V DCで15Wを供給する能力がある(提供された)外部電源によって給電されなければなりません。DCｼﾞｬｯｸの極性は中心が正(+)です。AVR ONE!電源投入時、電源LEDが直ちに点灯するでしょう。LEDが点灯しない場合、正しい電源が使われていること、または違う電源が使われているなら、極性が中心+であることを調べてください。

4.3. ﾎｽﾄ ｺﾝﾋﾟｭｰﾀへの接続

最初にAVR ONE!を接続する前に、ﾎｽﾄ ｺﾝﾋﾟｭｰﾀでUSBﾄﾞﾗｲﾊﾞを必ずｲﾝｽﾄｰﾙしてください。これはAtmelから無償で提供された前処理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。更なる情報について、または最新の前処理ｿﾌﾄｳｪｱをﾀﾞｳﾝﾛｰﾄﾞするには、www.atmel.comをご覧ください。

AVR ONE!は提供されたUSBｹｰﾌﾞﾙを使ってﾎｽﾄ ｺﾝﾋﾟｭｰﾀで利用可能なUSBﾎﾟｰﾄに接続されなければなりません。AVR ONE!はUSB2.0準拠制御器を含み、例え全速(Full-speed)動作時に流し追跡機能が利用可能でなくても、全速(注:制限適用、「既知の問題」をご覧ください)と高速(High-speed)の両方で動作することができます。最良の結果のために、AVR ONE!を(外部ﾊﾌﾞ経由ではなく)直接ﾎｽﾄ ｺﾝﾋﾟｭｰﾀに接続し、(提供された)USB2.0高速保証ｹｰﾌﾞﾙを使ってください。

4.4. USBﾄﾞﾗｲﾊﾞのｲﾝｽﾄｰﾙ

4.4.1. Windows

Microsoft® Windows®が走行するｺﾝﾋﾟｭｰﾀでAtmel AVR ONE!をｲﾝｽﾄｰﾙすると、AVR ONE!が最初に通電(接続)される時にUSBﾄﾞﾗｲﾊﾞが読み込まれます。

注: 最初に通電(接続)する前に、前処理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。

新しいﾊｰﾄﾞｳｪｱ ｳｨｻﾞｰﾄﾞ(Found New Hardware Wizard)を通して既定(″推奨″)任意選択で処理を進めてください。

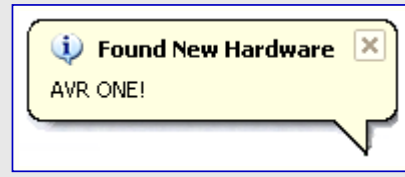
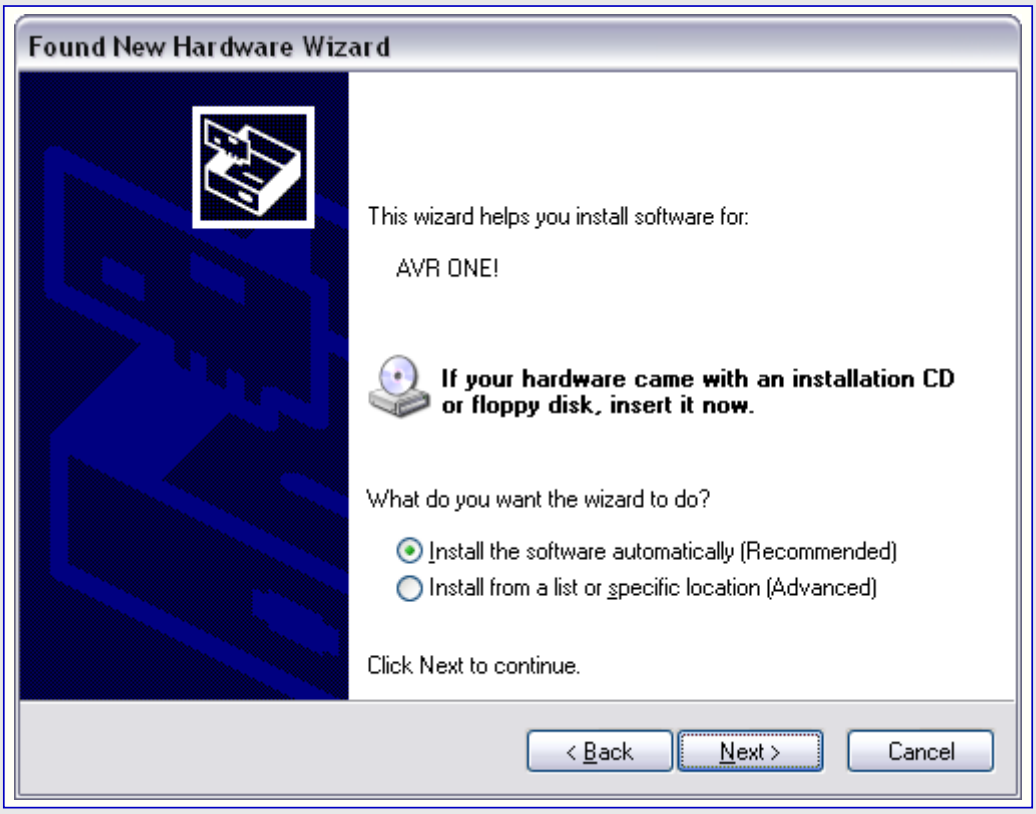


図4-2. AVR ONE! USBﾄﾞﾗｲﾊﾞのｲﾝｽﾄｰﾙ

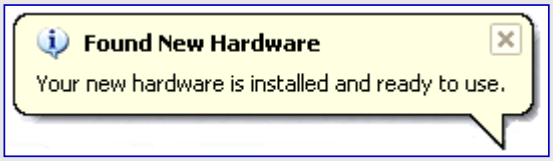
図4-3. AVR ONE! USBﾄﾞﾗｲﾊﾞのｲﾝｽﾄｰﾙ



自動的に検出されなかった場合、ｳｨｻﾞｰﾄﾞを<windowsﾙｰﾄ>¥infﾌｫﾙﾀﾞ内に格納されているavrone.infと呼ばれる(Jungoによって提供される)ﾃﾞﾊﾞｲｽﾄﾞﾗｲﾊﾞへ指示してください。

一旦ｲﾝｽﾄｰﾙが成功すると、AVR ONE!はﾃﾞﾊﾞｲｽ ﾏﾈｰｼﾞｬで"Jungo"ﾃﾞﾊﾞｲｽとして現れます。

今やAVR ONE!は使用準備が整いました。



4.5. ﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾃﾞﾊﾞｯｸﾞ

Atmel Studioを使用してAtmel AVR ONE!とで始めるための最も簡単な方法はAtmel Studioと共に含まれる多数のﾌﾟﾛｼﾞｪｸﾄ例の1つを構築することです。

5. Atmel AVR ONE!の接続

5.1. JTAG目的対象への接続

Atmel AVR ONE!探針はJTAGのﾃﾞﾊﾞｯｸﾞとﾌﾟﾛｸﾞﾗﾐﾝｸﾞを支援する2つの目的対象ｺﾈｸﾀを持ちます。AUX追跡を持つ目的対象ﾃﾞﾊﾞｲｽをﾃﾞﾊﾞｯｸﾞする時に、38ﾋﾟﾝのNexus(Mictor)ｺﾈｸﾀが使用されるべきです。AUX追跡またはﾌﾟﾛｸﾞﾗﾐﾝｸﾞなしでのJTAGﾃﾞﾊﾞｯｸﾞについてはMictorまたは10ﾋﾟﾝのどちらのｺﾈｸﾀでも使用することができます。

5.1.1. JTAG Mictorｺﾈｸﾀの使い方

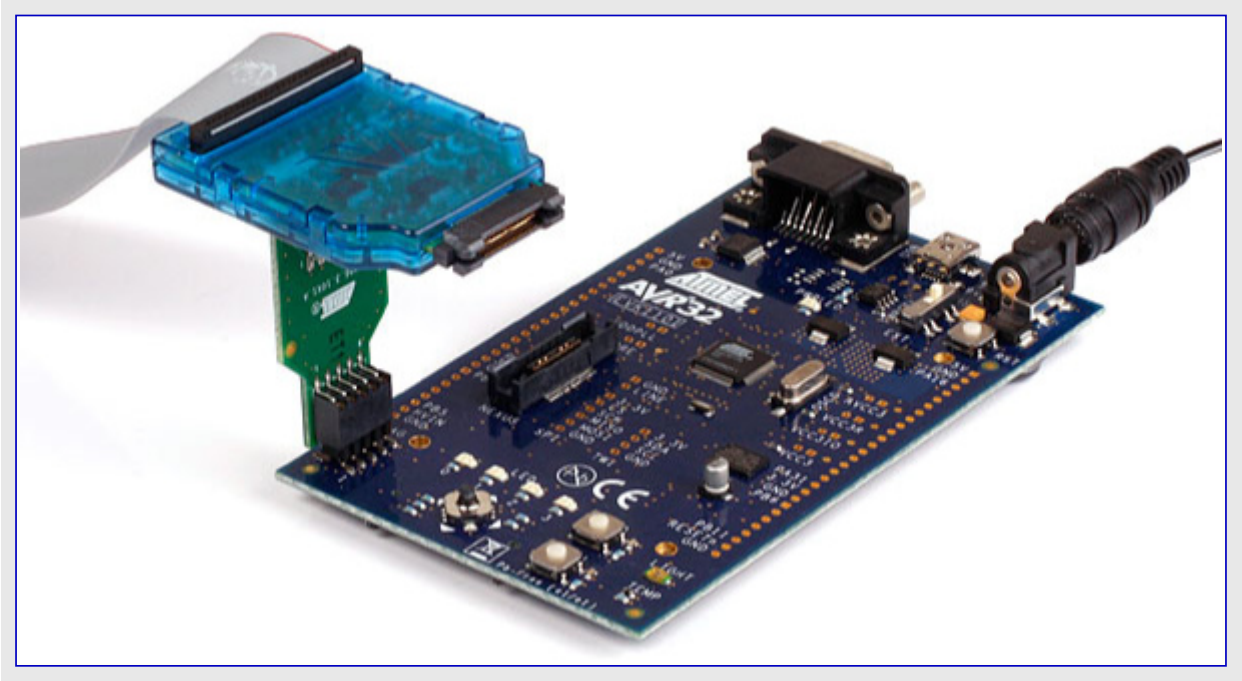
38ﾋﾟﾝ Mictorｺﾈｸﾀ用のﾋﾟﾝ説明図は図6-4. Mictorｺﾈｸﾀ説明図で示されます。

Atmel AVR ONE!探針ｺﾈｸﾀを単に目的対象応用のMictorｺﾈｸﾀに挿入し、接続が確実にまるまで圧力を加えてください。不正な嵌合の向きを防ぐために、このｿｹｯﾄは極性化されています。

5.1.2. JTAG 10ﾋﾟﾝ ｺﾈｸﾀの使い方

10ﾋﾟﾝ JTAGｺﾈｸﾀ用のﾋﾟﾝ説明図は図6-2. JTAGﾍｯﾀﾞ ﾋﾟﾝ説明図で示されます。

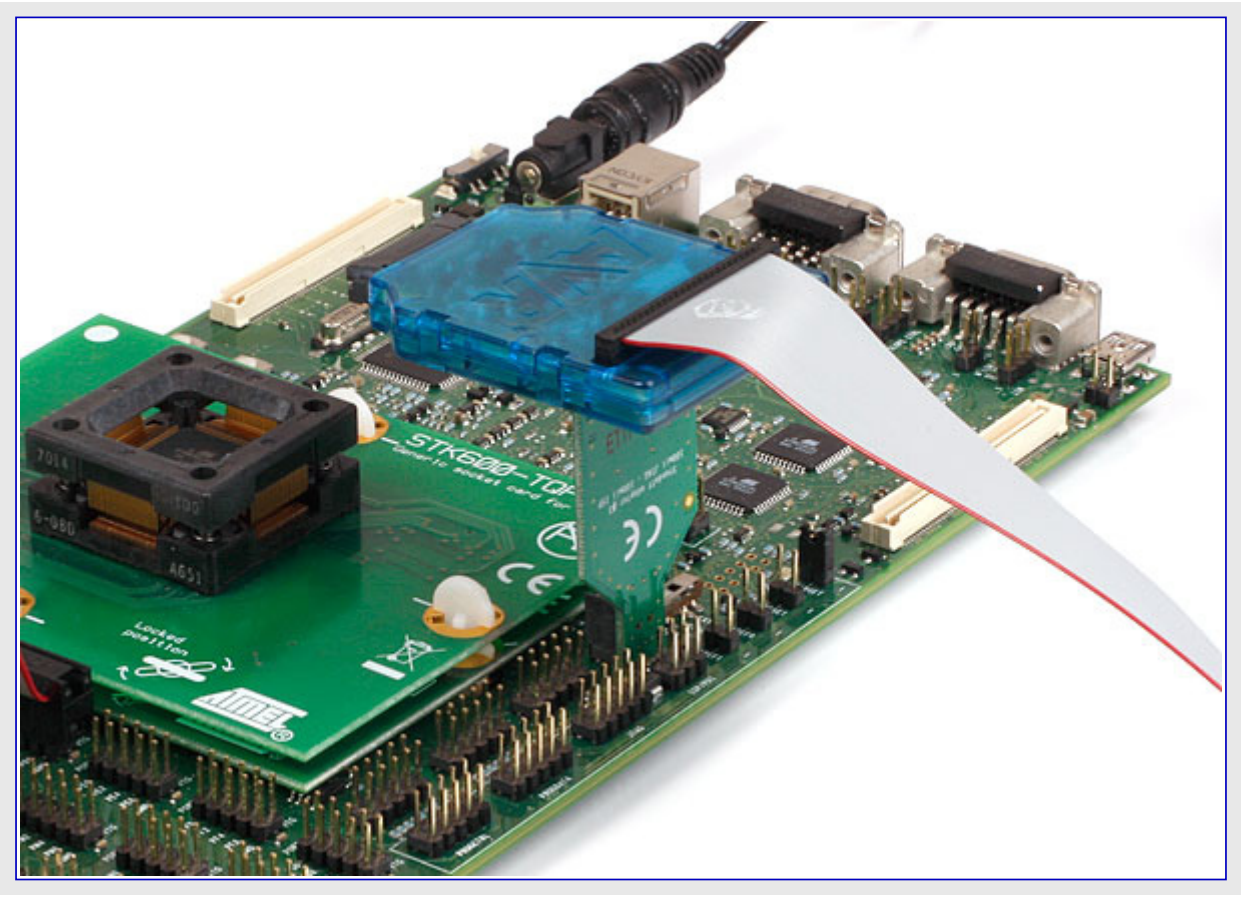
Atmel AVR ONE!を目的対象応用PCBに接続する時に正しい10ﾋﾟﾝ ﾍｯﾀﾞの向きで使うことに注意してください。AVR ONE!探針を100milと50milの両方の目的対象応用ｺﾈｸﾀへ接続するのに、(提供された)個別ｱﾀﾞﾌﾟﾀを使うことができます。



5.2. aWire目的対象への接続

Atmel AVR ONE!は単線’aWire’ｲﾝﾀｰﾌｪｰｽを使用するAtmel AVR UC3L系統のﾃﾞﾊﾞｲｽとｲﾝﾀｰﾌｪｰｽすることができます。aWireｲﾝﾀｰﾌｪｰｽ用の推奨ﾋﾟﾝ説明図は図6-5. aWireﾍｯﾀﾞ ﾋﾟﾝ説明図で示されます。

AVR ONE!を目的対象応用PCBに接続する時に正しい6ﾋﾟﾝ ﾍｯﾀﾞの向きで使うことに注意してください。AVR ONE!探針を100milと50milの両方の目的対象応用ｺﾈｸﾀへ接続するのに、(提供された)個別ｱﾀﾞﾌﾟﾀを使うことができます。



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。推奨6ピン ピン配列はAVR ONE!デバガガそれ自身の資源だけでなく、既存のAVRインアターフェースにも基づきます。標準6ピン ヘッダを持たない目的対象への接続時、AVR ONE!探針上の10ピン JT AGコネクタと目的対象基板間にバラ線ケーブルを使うことができます。右表で記述されるように3つの接続が必要とされます。

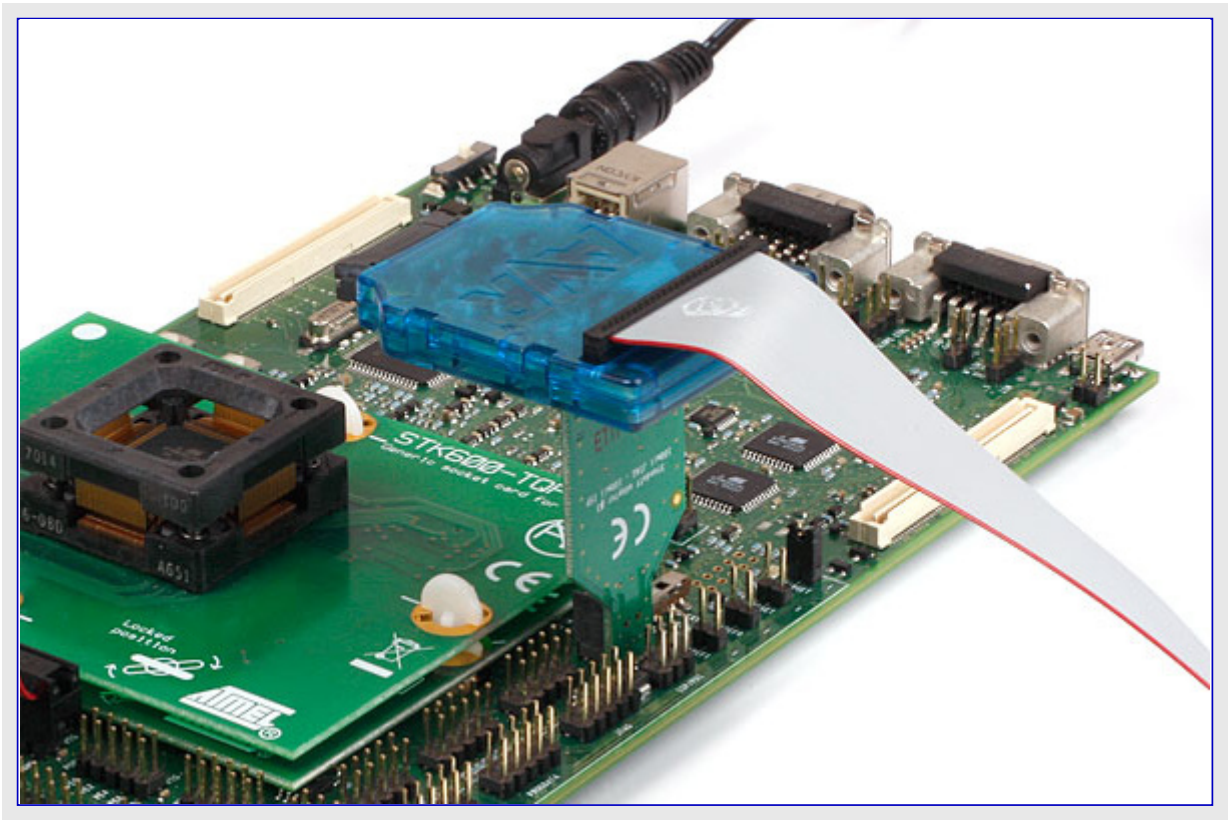
表5-1. バラ線ケーブルを使ったaWireへの接続

AVR ONE! JTAG探針	目的対象ピン	バラ線ケーブル色	aWireピン
1 (TCK)		黒	
2 (GND)	GND	白	6
3 (TDO)	DATA	灰	1
4 (VTref)	VTref	紫	2
5 (TMS)		青	
6 (nSRST)		緑	
7 (未接続)		黄	
8 (nTRST)		橙	
9 (TDI)		赤	
10 (GND)		茶	

5.3. PDI目的対象への接続

6ピン PDIコネクタについてのピン説明図は図6-6. PDIヘッダ ピン説明図で示されます。

Atmel AVR ONE!を目的対象応用PCBへ接続する時に正しい6ピン ヘッダの向きで使うことに注意してください。AVR ONE!探針を100milと50milの両方の目的対象応用コネクタへ接続するのに、(提供された)個別アダプタを使うことができます。



標準6ピン ヘッダを持たない目的対象へ接続する時は、探針上のAVR ONE! 10ピンJTAGコネクタと目的対象基板間にバラ線ケーブルを使うことができます。4つの接続が必要とされ、右表がそれらを接続する場所を記述します。

注: PDI_DATAが9番ピンに接続されるJTAGICEmk IIのJTAG探針とは違いがあります。

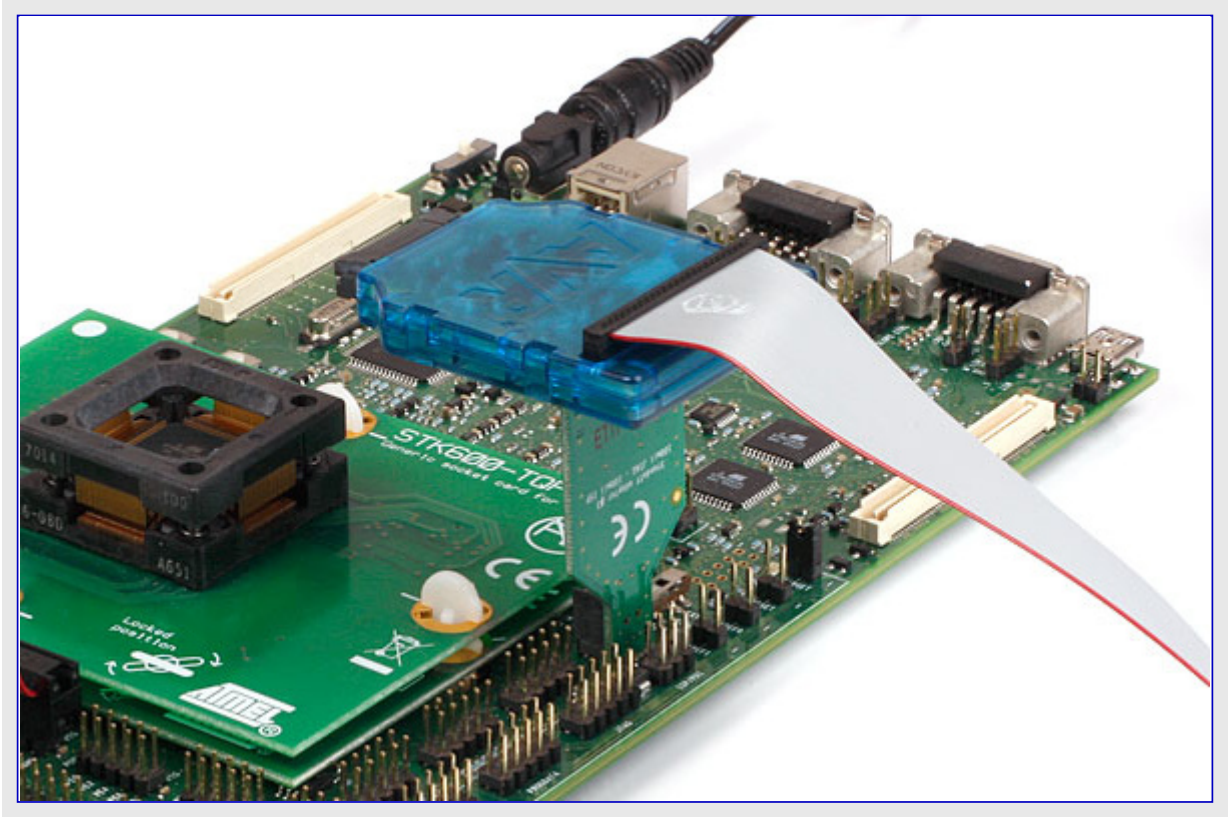
表5-2. バラ線ケーブルを使ったPDIへの接続

AVR ONE! JTAG探針	目的対象ピン	バラ線ケーブル色	STK600 PDIピン
1 (TCK)		黒	
2 (GND)	GND	白	6
3 (TDO)	PDI_DATA	灰	1
4 (VTref)	VTref	紫	2
5 (TMS)		青	
6 (nSRST)	PDI_CLK	緑	5
7 (未接続)		黄	
8 (nTRST)		橙	
9 (TDI)		赤	
10 (GND)		茶	

5.4. ﾃﾞﾊﾞｯｸﾞWIRE目的対象への接続

6ﾋﾟﾝ ﾃﾞﾊﾞｯｸﾞWIRE(SPI)ｺﾈｸﾀについてのﾋﾟﾝ説明図は図6-7. ﾃﾞﾊﾞｯｸﾞWIRE(SPI)ﾍｯﾀﾞ ﾋﾟﾝ説明図で示されます。

Atmel AVR ONE!を目的対象応用PCBへ接続する時に正しい6ﾋﾟﾝ ﾍｯﾀﾞの向きで使うことに注意してください。AVR ONE!探針を100milと50milの両方の目的対象応用ｺﾈｸﾀへ接続するのに、(提供された)個別ｱﾀﾞﾌﾟﾀを使うことができます。



例えﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽが正しく動作するために単一信号線(RESET)とVCC、GNDだけが必要とは言え、ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽがSPIﾌﾟﾛｸﾞﾗﾐﾝｸﾞを使って許可と禁止ができるように完全なSPIｺﾈｸﾀへの接続を持つことが推奨されます。

DWENﾋｭｰｽﾞが許可されると、OCD部がRESET線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。ﾃﾞﾊﾞｯｸﾞWIRE OCDは(Atmel Studioのﾌﾟﾛﾊﾟﾃｨ ﾀﾞｲｱﾛｸﾞ内のﾃﾞﾊﾞｯｸﾞ ﾀﾌﾞ上の釦を使って)一時的にそれ自身を禁止する、故にRESET線の制御を開放する能力があります。その後にSPIｲﾝﾀｰﾌｪｰｽは(SPIENﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(0)されている場合にだけ)再び利用可能で、SPIｲﾝﾀｰﾌｪｰｽを使って非ﾌﾟﾛｸﾞﾗﾑ(1)にされることをDWENﾋｭｰｽﾞに許します。DWENﾋｭｰｽﾞが非ﾌﾟﾛｸﾞﾗﾑ(1)にされる前に電源がOFF/ONされた場合、ﾃﾞﾊﾞｯｸﾞWIRE部は再びRESETﾋﾟﾝの制御を取るでしょう。単純にAtmel StudioにDWENﾋｭｰｽﾞの設定(0)と解除(1)を処理させることが強く推奨されます。

目的対象AVR上の施錠ﾋﾞｯﾄがﾌﾟﾛｸﾞﾗﾑ(0)されている場合、ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽを使うのは不可能です。常にDWENﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(0)される前に施錠ﾋﾞｯﾄが解除(1)されているのを確実にし、DWENﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(0)されている間に施錠ﾋﾞｯﾄを決して設定(0)しないでください。ﾃﾞﾊﾞｯｸﾞWIRE許可(DWEN)ﾋｭｰｽﾞと施錠ﾋﾞｯﾄの両方が設定(0)された場合、ﾁｯﾌﾟ消去を行う、故に施錠ﾋﾞｯﾄを解除(1)するのに、高電圧ﾌﾟﾛｸﾞﾗﾐﾝｸﾞを使うことができます。施錠ﾋﾞｯﾄが解除(1)されると、ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽが再び許可されるでしょう。SPIｲﾝﾀｰﾌｪｰｽはDWENﾋｭｰｽﾞが非ﾌﾟﾛｸﾞﾗﾑ(1)にされる時に、ﾋｭｰｽﾞ読み出し、識票読み出し、ﾁｯﾌﾟ消去の実行の能力だけがあります。

表5-3. ﾊﾞﾗ線ｹｰﾌﾞﾙを使ったﾃﾞﾊﾞｯｸﾞWIREへの接続

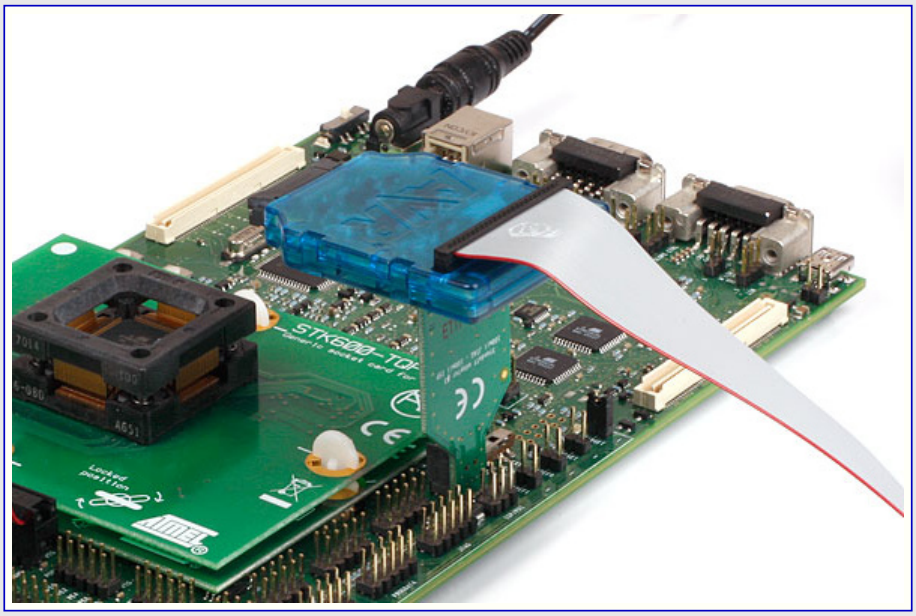
AVR ONE! JTAG探針	目的対象ﾋﾟﾝ	ﾊﾞﾗ線ｹｰﾌﾞﾙ色
1 (TCK)		黒
2 (GND)	GND	白
3 (TDO)		灰
4 (VTref)	VTref	紫
5 (TMS)		青
6 (nSRST)	RESET	緑
7 (未接続)		黄
8 (nTRST)		橙
9 (TDI)		赤
10 (GND)		茶

5.5. SPI目的対象への接続

6ﾋﾟﾝ SPIｺﾈｸﾀについてのﾋﾟﾝ接続図は図6-8. SPIﾍｯﾀﾞ ﾋﾟﾝ接続図で示されます。

Atmel AVR ONE!を目的対象応用PCBへ接続する時に正しい6ﾋﾟﾝ ﾍｯﾀﾞの向きで使うことに注意してください。AVR ONE!探針を100milと50milの両方の目的対象応用ｺﾈｸﾀへ接続するのに、(提供された)個別ｱﾀﾞﾌﾟﾀを使うことができます。

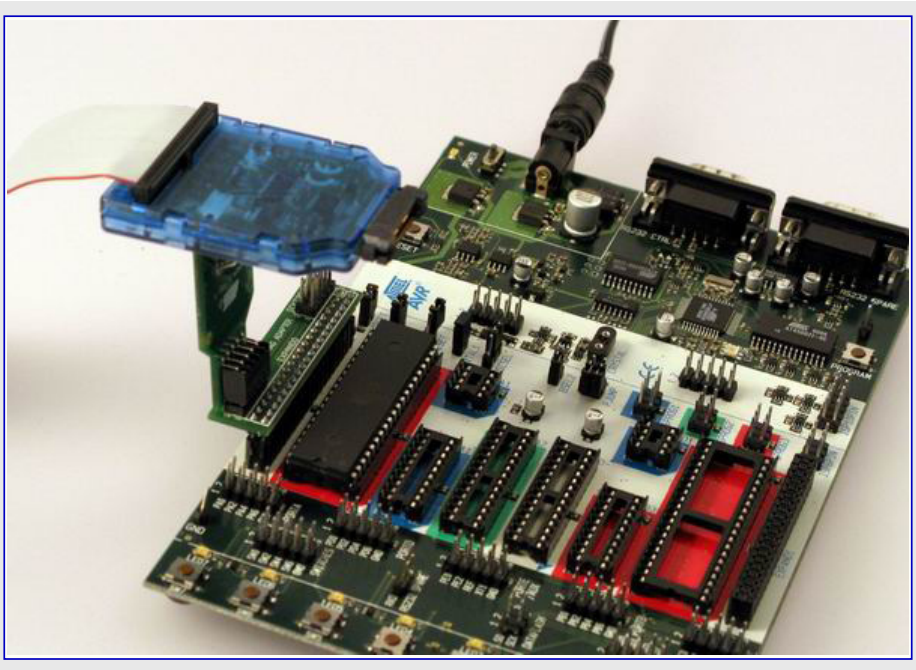
注: 例えSPIENﾋｭｰｽﾞもﾌﾟﾛｸﾞﾗﾑ(0)されていても、ﾃﾞﾊﾞｯｸﾞWIRE許可(DWEN)ﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(0)される時にSPIｲﾝﾀｰﾌｪｰｽが事実上禁止されます。SPIｲﾝﾀｰﾌｪｰｽを再び許可するには、ﾃﾞﾊﾞｯｸﾞWIREﾃﾞﾊﾞｯｸﾞ作業中に'ﾃﾞﾊﾞｯｸﾞWIRE禁止'命令が発行されなければなりません。この方法でのﾃﾞﾊﾞｯｸﾞWIRE禁止はSPIENﾋｭｰｽﾞが既に(0)されていることが必要です。Atmel StudioがﾃﾞﾊﾞｯｸﾞWIREの禁止に失敗した場合、それは多分、SPIENﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(0)されていないことです。この場合、SPIENﾋｭｰｽﾞをﾌﾟﾛｸﾞﾗﾐﾝｸﾞするのに高電圧ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ｲﾝﾀｰﾌｪｰｽを使うことが必要です。DWENﾋｭｰｽﾞの設定と解除を単純にAtmel Studioに処理させることが強く推奨されます。



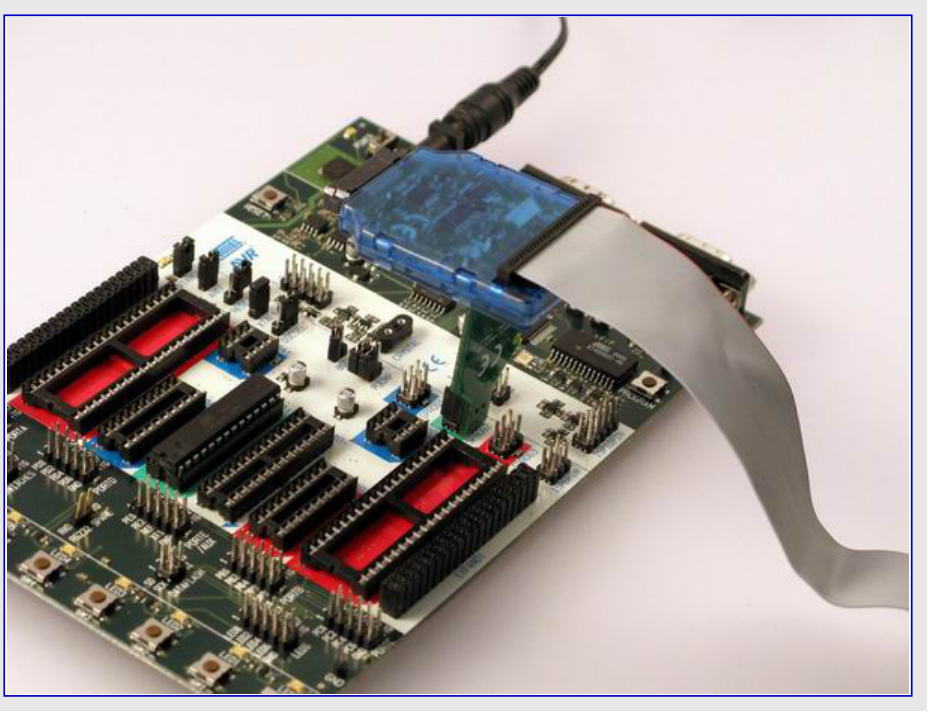
5.6. STK500でのAVR ONE!の使用

Atmel STK®500ｽﾀｰﾀ ｷｯﾄはJTAG、ﾃﾞﾊﾞｯｸﾞWIRE、SPIのｲﾝﾀｰﾌｪｰｽを通してAVR ONE!を接続できるAtmel AVRﾃﾞﾊﾞｲｽを収容するのに用いることができます。

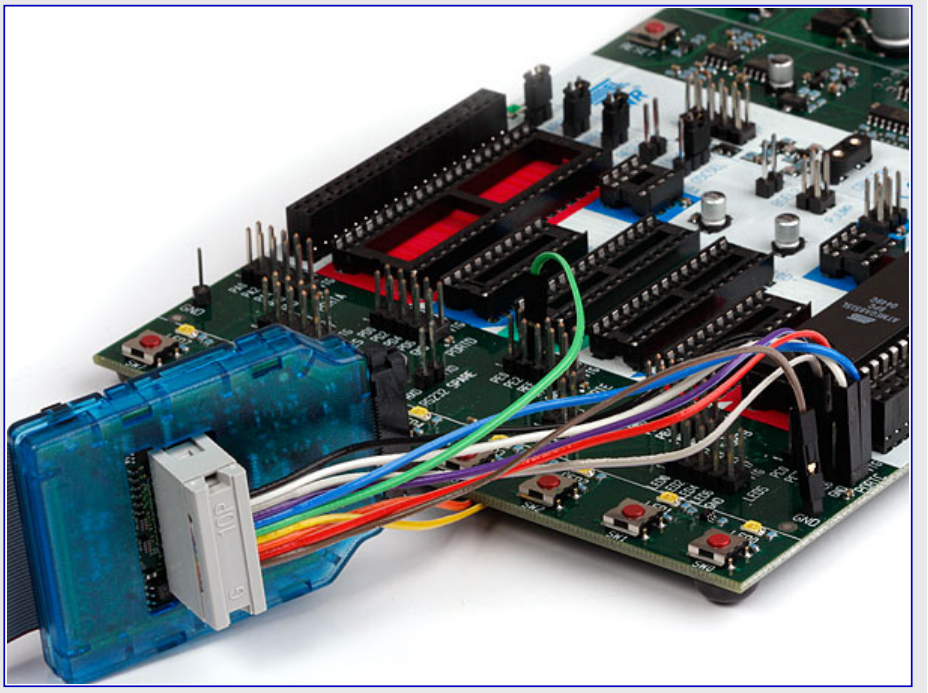
JTAG目的対象接続時、単純にATSTK500_JTAG_ADAPTERを使ってください。利用可能なSTK500 JTAGｱﾀﾞﾌﾟﾀを持っていない場合、STK500のPORTC[5～2]でのﾃﾞﾊﾞｲｽのJTAGﾎﾟｰﾄへ直接的に接続するのに10ﾋﾟﾝの多色"ﾊﾞﾗ線"ｹｰﾌﾞﾙも使うことができます。



ﾃﾞﾊﾞｯｸﾞWIREとSPIの目的対象への接続は同じ個別ｱﾀﾞﾌﾟﾀを用いて行われます。ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽ使用時、ﾘｾｯﾄ線に必要とされる駆動を許すために、STK500のRESETｼﾞｬﾝﾊﾟの取り外しを確実にしてください。

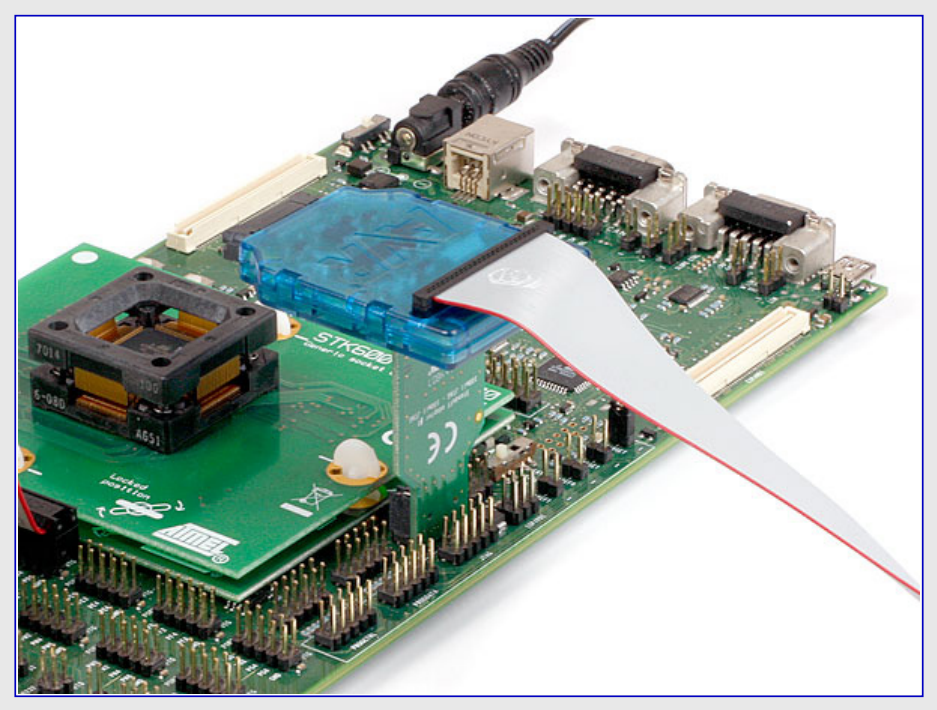


代わりに、AVR ONE!は(提供された)10ﾋﾟﾝ"ﾊﾞﾗ線"ｹｰﾌﾞﾙを用いてどの目的対象ｲﾝﾀｰﾌｪｰｽへも接続することができます。

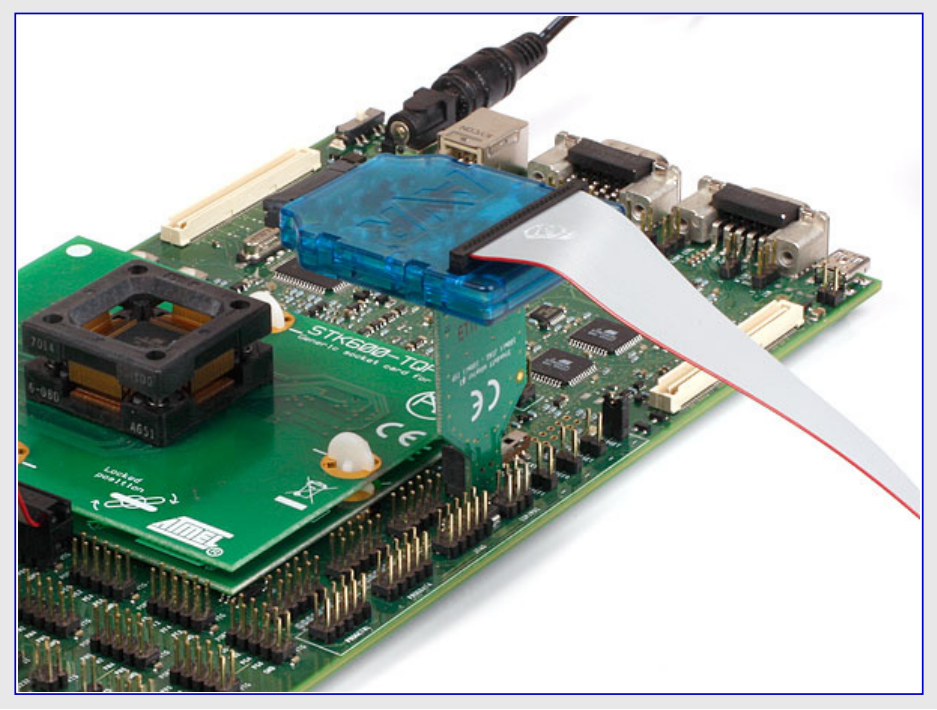


5.7. STK600でのAVR ONE!の使用

Atmel STK600ｽﾀｰﾀ ｷｯﾄはJTAGｲﾝﾀｰﾌｪｰｽを通してAVR ONE!を接続できるAtmel AVRﾃﾞﾊﾞｲｽを収容するのに用いることができます。



JTAG目的対象への接続時、STK600のJTAGｺﾈｸﾀへ接続するには単純に(提供された)10ﾋﾟﾝの100mil個別ｱﾀﾞﾌﾟﾀを使ってください。



PDI、ﾃﾞﾊﾞｯｸﾞWIRE、またはISP目的対象への接続時、ISP/PDIｺﾈｸﾀへ接続するには単純に(提供された)6ﾋﾟﾝの100mil個別ｱﾀﾞﾌﾟﾀを使ってください。

6. チップ上デバッグ

6.1. 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 AVR ONE!は伝統的なエミュレータではありません。代わりに、AVR ONE!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Atmel AVR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AVR目的対象デバイス上で実際に実行されます。

OCDシステムとで、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気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AVR ONE!と完全に独立です。AVR ONE!は中断条件が発生したかを知るために目的対象AVR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

AVR 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AVR目的対象に実装されるOCD部の大きさに依存します。通常、デバッガ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 AVR 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 AVR 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「特別な考慮」章をご覧ください。

6.2. 物理インターフェース

Atmel AVR ONE!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。

6.2.1. 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 AVR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AVRを含む応用PCB設計時、図6-2. JTAGヘッダ ピン接続図で示されるようなピン配列を使うことが推奨されます。Atmel AVR ONE!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されます。

図6-1. JTAGインターフェースの基本

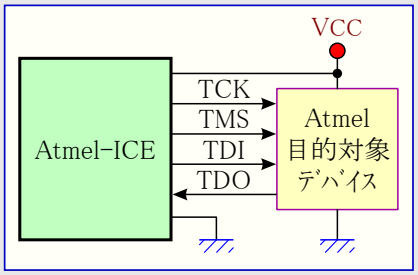


図6-2. JTAGヘッダ ピン接続図

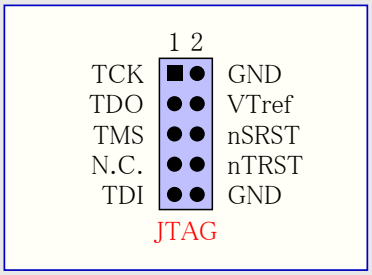


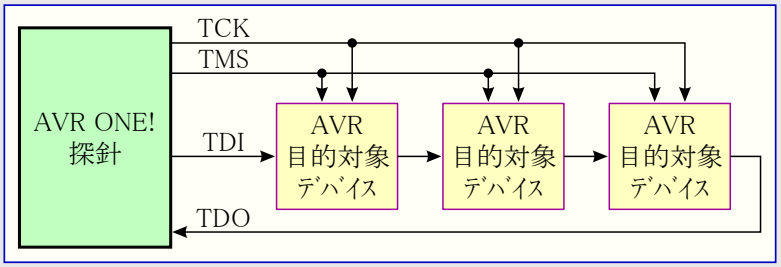
表6-1. JTAGﾋﾟﾝ説明

名前	ﾋﾟﾝ番号	説明
TCK	1	検査ｸﾛｯｸ (AVR ONE!から目的対象ﾃﾞﾊﾞｲｽへのｸﾛｯｸ信号)
TMS	5	検査種別選択 (AVR ONE!から目的対象ﾃﾞﾊﾞｲｽへの制御信号)
TDI	9	検査ﾃﾞｰﾀ入力 (AVR ONE!から目的対象ﾃﾞﾊﾞｲｽへ送出されるﾃﾞｰﾀ)
TDO	3	検査ﾃﾞｰﾀ出力 (目的対象ﾃﾞﾊﾞｲｽからAVR ONE!へ送出されるﾃﾞｰﾀ)
nTRST	8	検査ﾘｾｯﾄ (任意、いくつかのAVRﾃﾞﾊﾞｲｽのみ)。JTAG TAP制御器のﾘｾｯﾄに使用
nSRST	6	供給元ﾘｾｯﾄ (任意)。目的対象ﾃﾞﾊﾞｲｽのﾘｾｯﾄに使用。或る筋書でのﾃﾞﾊﾞｯｸﾞを必要とし得る、目的対象ﾃﾞﾊﾞｲｽをﾘｾｯﾄ状態で保持することをAVR ONE!に許すために、このﾋﾟﾝの接続が推奨されます。
VTref	4	目的対象基準電圧。AVR ONE!は正しくﾚﾍﾞﾙ変換器を給電するために、このﾋﾟﾝで目的対象ﾃﾞﾊﾞｲｽ電圧を採取します。AVR ONE!はこのﾋﾟﾝから1mA未満を引き出します。
GND	2,10	接地。AVR ONE!と目的対象ﾃﾞﾊﾞ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番ﾋﾟﾝとGND間に雑音分離(ﾃﾞｶｯﾌﾟ)ｺﾝﾃﾞ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ﾃﾞﾊﾞｲｽに対してﾃﾞｨｰｼﾞｰ チｪｰﾝ構成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。目的対象ﾃﾞﾊﾞ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6-3. JTAGﾃﾞｨｰｼﾞｰ 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6-3. JTAGﾃﾞｨｰｼﾞｰ チｪｰﾝ



ﾃﾞｨｰｼﾞｰ チｪｰﾝでのﾃﾞﾊﾞｲｽ接続時、以下の点が考慮されなければなりません。

- 全てのﾃﾞﾊﾞｲｽはAVR ONE!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。
- 全てのﾃﾞﾊﾞｲｽは同じ目的対象電圧で動作しなければなりません。AVR ONE!のVTref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。
- ﾁｪｰﾝ内のﾃﾞﾊﾞｲｽの何れかがJTAGﾎﾟｰﾄを禁止する場合、AVR ONE!探針のnSRSTはﾃﾞﾊﾞｲｽのRESETに接続されなければなりません。
- ”Devices before”はTDI信号が目的対象ﾃﾞﾊﾞｲｽに到達する前にﾃﾞｨｰｼﾞｰ チｪｰﾝ内を通って通過しなければならないJTAGﾃﾞﾊﾞｲｽ数を参照します。同様に”Devices after”は信号がAVR ONE!のTDOﾋﾟﾝに到達する前に目的対象ﾃﾞﾊﾞｲｽの後を通って通過しなければならないﾃﾞﾊﾞｲｽ数です。
- ”Instruction bits before”と”Instruction bits after”はﾃﾞｨｰｼﾞｰ チｪｰﾝ内で目的対象ﾃﾞﾊﾞｲｽの前後に接続される全てのJTAGﾃﾞﾊﾞｲｽの命令ﾚｼﾞｽﾀ(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240ﾋﾞｯﾄに制限されます。

ﾃﾞｨｰｼﾞｰ チｪｰﾝ例：TDI ⇒ ATmega1280 ⇒ ATxmega128A1 ⇒ ATUC3A0512 ⇒ TDO

Atmel AVR XMEGAﾃﾞﾊﾞｲｽに接続するためのﾃﾞｨｰｼﾞｰ チｪｰﾝ設定は次のとおりです。

Devices before : 1
Devices after : 1
Instruction bits before : 4 (8ﾋﾞｯﾄ AVRﾃﾞﾊﾞｲｽは4ﾋﾞｯﾄのIRを持ちます。)
Instruction bits after : 5 (32ﾋﾞｯﾄ AVRﾃﾞﾊﾞｲｽは5ﾋﾞｯﾄのIRを持ちます。)

6.2.2. (JTAGを含む)補助(AUX)物性

補助ﾎﾟｰﾄが特徴のAtmel AVR目的対象ﾃﾞﾊﾞｲｽのﾃﾞﾊﾞｯｸﾞ時、JTAGとAUXの両ﾎﾟｰﾄへのｱｸｾｽを提供する38ﾋﾟﾝ ｺﾈｸﾀを使うことが推奨されます。AUXﾎﾟｰﾄはﾌﾟﾛｸﾞﾗﾑ追跡のような高度なﾃﾞﾊﾞｯｸﾞ機能を容易にします。

この38ﾋﾟﾝ ｺﾈｸﾀのﾋﾟﾝ接続図は図6-4. Mictorｺﾈｸﾀ ﾋﾟﾝ接続図で示され、表6-2. Mictorｺﾈｸﾀ接続表で一覧にされます。

MictorｺﾈｸﾀはTyco Electronics (部品番号:2-5767004-2)で入手可能です。

図6-4. Mictorｺﾈｸﾀ接続図

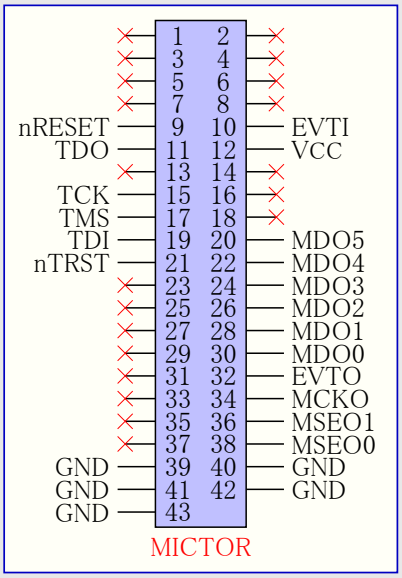


表6-2. Mictorｺﾈｸﾀ接続表

名前	ﾋﾟﾝ番号	説明
TCK	15	検査ｸﾛｯｸ
TMS	17	検査種別選択
TDI	19	検査ﾃﾞｰﾀ入力
TDO	11	検査ﾃﾞｰﾀ出力
nTRST	21	検査ﾘｾｯﾄ
nRESET	9	供給元ﾘｾｯﾄ
EVTI	10	事象入力
EVTO	32	事象出力
MCKO	34	ﾒｯｾｰｼﾞｸﾛｯｸ出力
MSEO0	38	ﾒｯｾｰｼﾞ開始/終了出力0
MSEO1	36	ﾒｯｾｰｼﾞ開始/終了出力1
MDO0	30	ﾒｯｾｰｼﾞﾃﾞｰﾀ出力0
MDO1	28	ﾒｯｾｰｼﾞﾃﾞｰﾀ出力1
MDO2	26	ﾒｯｾｰｼﾞﾃﾞｰﾀ出力2
MDO3	24	ﾒｯｾｰｼﾞﾃﾞｰﾀ出力3
MDO4	22	ﾒｯｾｰｼﾞﾃﾞｰﾀ出力4
MDO5	20	ﾒｯｾｰｼﾞﾃﾞｰﾀ出力5
VREF	12	目的対象電圧基準
GND	39～43	接地

6.2.3. aWire

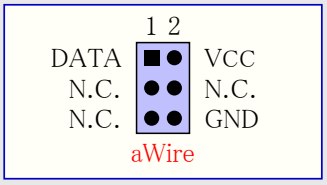
aWireｲﾝﾀｰﾌｪｰｽはﾌﾟﾛｸﾞﾗﾐﾝｸﾞをﾃﾞﾊﾞｯｸﾞの機能を許すためにAtmel AVRﾃﾞﾊﾞｲｽの/RESET線を利用します。Atmel AVR ONE!によって、ﾋﾟﾝの既定/RESET機能を禁止する特別な許可手順が送出されます。

aWireｲﾝﾀｰﾌｪｰｽを持つAVRを含む応用PCBの設計時、**図6-5. aWireﾍｯﾀﾞ ﾋﾟﾝ接続図**で示されるようなﾋﾟﾝ配列を使うことが推奨されます。AVR ONE!はこのﾋﾟﾝ配列を支援する100milと50milの両ｱﾀﾞﾌﾟ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、方向変更時に開始ﾋﾞｯﾄ検出の失敗を避けるため、/RESET線上に約47kΩのﾌﾟﾙｱｯﾌﾟ抵抗が推奨されます。

aWireｲﾝﾀｰﾌｪｰｽはﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾃﾞﾊﾞｯｸﾞの双方のｲﾝﾀｰﾌｪｰｽとして使うことができ、10ﾋﾟﾝ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6-5. aWireﾍｯﾀﾞ ﾋﾟﾝ接続図

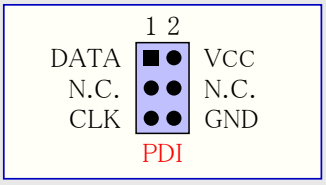


6.2.4. PDI物性

ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ/ﾃﾞﾊﾞｯｸﾞ用ｲﾝﾀｰﾌｪｰｽ(PDI:Program and Debug Interface)はﾃﾞﾊﾞｲｽの外部ﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾁｯﾌﾟ上ﾃﾞﾊﾞｯｸﾞ用のAtmel専有ｲﾝﾀｰﾌｪｰｽです。PDI物性は目的対象ﾃﾞﾊﾞｲｽとの双方向半二重同期通信を提供する2ﾋﾟﾝ 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6-6. PDIﾍｯﾀﾞ ﾋﾟﾝ接続図**で示されるﾋﾟﾝ配列が使われるべきです。そしてAVR ONE!探針を応用PCBへ接続するのに、AVR ONE!ｷｯﾄと共に提供される6ﾋﾟﾝ個別ｱﾀﾞﾌﾟﾀを使うことができます。

図6-6. PDIﾍｯﾀﾞ ﾋﾟﾝ接続図



6.2.5. ﾃﾞﾊﾞｯｸﾞWIRE

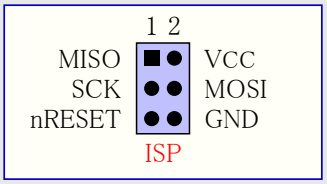
ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽは小ﾋﾟﾝ数ﾃﾞﾊﾞｲｽで使うためにAtmelによって開発されました。4ﾋﾟﾝを使うJTAGｲﾝﾀｰﾌｪｰｽとは異なり、ﾃﾞﾊﾞｯｸﾞWIREはﾃﾞﾊﾞｯｸﾞ ﾂｰﾙと共に双方向半二重非同期通信用に単一(RESET)ﾋﾟﾝだけを利用します。

ﾃﾞﾊﾞｯｸﾞ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6-7. ﾃﾞﾊﾞｯｸﾞWIRE(SPI)ﾍｯﾀﾞ ﾋﾟﾝ接続図**で示されるﾋﾟﾝ配列が使われるべきです。

注: ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽはﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ｲﾝﾀｰﾌｪｰｽとして使うことができません。これは目的対象をﾌﾟﾛｸﾞﾗﾐﾝｸﾞするために、(**図6-8. SPIﾍｯﾀﾞ ﾋﾟﾝ接続図**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。

ﾃﾞﾊﾞｯｸﾞWIRE許可(DWEN)ﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(0)され、施錠ﾋﾞｯﾄが非ﾌﾟﾛｸﾞﾗﾑ(1)の時に、目的対象内のﾃﾞﾊﾞｯｸﾞWIREｼｽﾃﾑが活性(有効)にされます。RESETﾋﾟﾝはﾌﾟﾙｱｯﾌﾟ許可のﾜｲﾔｰﾄﾞAND(ｵｰﾌﾟﾝ ﾄﾞﾚｲﾝ)双方向入出力ﾋﾟﾝとして構成設定され、目的対象とﾃﾞﾊﾞｯｶﾞ間の通信中継器になります。

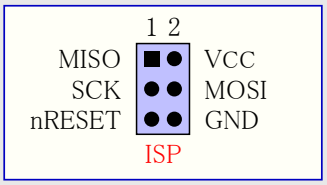
図6-7. ﾃﾞﾊﾞｯｸﾞWIRE(SPI) ﾍｯﾀﾞ ﾋﾟﾝ接続図



6.2.6. SPI

実装書き込み(ISP:In-System Programming)はﾌﾗｯｼｭ ﾒﾓﾘとEEPROM内にｺｰﾄﾞを書き込むのに目的対象のAtmel AVRの内部SPI(Serial Peripheral Interface)を使います。これはﾃﾞﾊﾞｯｸﾞ 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6-8. SPIﾍｯﾀﾞ ﾋﾟﾝ配列**」で示されるﾋﾟﾝ配列が使われるべきです。

図6-8. SPIﾍｯﾀﾞ ﾋﾟﾝ接続図



6.3. Atmel AVR OCD実装

6.3.1. 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ﾋﾞｯﾄ ﾏｲｸﾛ ｺﾝﾄﾛｰﾗ用のﾁｯﾌﾟ上ﾃﾞﾊﾞｯｸﾞ規格である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適合ﾃﾞﾊﾞｯｸﾞ解決策
- どのCPU速度も支援するOCD
- 6つのﾌﾟﾛｸﾞﾗﾑ ｶｳﾝﾀ ﾊｰﾄﾞｳｪｱ中断点(ﾌﾞﾚｰｸ ﾎﾟｲﾝﾄ)
- 2つのﾃﾞｰﾀ中断点
- 監視点として構成設定できる中断点
- 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ﾄﾞｳｪｱ中断点
- 実時間ﾌﾟﾛｸﾞﾗﾑ ｶｳﾝﾀ分岐追跡、ﾃﾞｰﾀ追跡、処理追跡

このﾃﾞﾊﾞｯｸﾞ ｲﾝﾀｰﾌｪｰｽに関する特別な考慮については「**特別な考慮**」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。

6.3.2. 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ﾃﾞﾊﾞｲｽ内で同じ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。

- 完全なﾌﾟﾛｸﾞﾗﾑの流れ制御
- 1つの専用ﾌﾟﾛｸﾞﾗﾑ ｱﾄﾞﾚｽ比較器またはｼﾝﾎﾞﾘｯｸ中断点(予約)
- 4つのﾊｰﾄﾞｳｪｱ比較器
- (BREAKを使う)無制限数の使用者ﾌﾟﾛｸﾞﾗﾑ中断点
- ｼｽﾃﾑ ｸﾛｯｸ周波数での制限なし

このﾃﾞﾊﾞｯｸﾞ ｲﾝﾀｰﾌｪｰｽに関する特別な考慮については「**特別な考慮**」をご覧ください。

6.3.3. 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。

- 完全なﾌﾟﾛｸﾞﾗﾑの流れ制御
- 4つのﾌﾟﾛｸﾞﾗﾑ ﾒﾓﾘ (ﾊｰﾄﾞｳｪｱ)中断点 (1つは予約)
- ﾃﾞｰﾀ中断点形式に結合することができるﾊｰﾄﾞｳｪｱ中断点
- (BREAKを使う)無制限数のﾌﾟﾛｸﾞﾗﾑ中断点 (ATmega128[A]を除く)

このﾃﾞﾊﾞｯｸﾞ ｲﾝﾀｰﾌｪｰｽに関する特別な考慮については「**特別な考慮**」をご覧ください。

6.3.4. Atmel megaAVR/tinyAVR OCD (ﾃﾞﾊﾞｯｸﾞWIRE)

ﾃﾞﾊﾞｯｸﾞWIRE OCDは小ﾋﾟﾝ数のAtmel AVRﾃﾞﾊﾞ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。

- 完全なﾌﾟﾛｸﾞﾗﾑの流れ制御
- (BREAKを使う)無制限数の使用者ﾌﾟﾛｸﾞﾗﾑ中断点
- 目的対象ｸﾛｯｸに基づく自動転送速度構成設定

このﾃﾞﾊﾞｯｸﾞ ｲﾝﾀｰﾌｪｰｽに関する特別な考慮については「**特別な考慮**」をご覧ください。

7. Atmel AVR ONE!ﾊｰﾄﾞｳｪｱ説明

7.1. LED

Atmel AVR ONE!の前面ﾊﾟﾈﾙは現在のﾃﾞﾊﾞｯｸﾞまたはﾌﾟﾛｸﾞﾗﾐﾝｸﾞの作業の状態を示す4つのLEDを持ちます。

図7-1. AVR ONE! LED位置

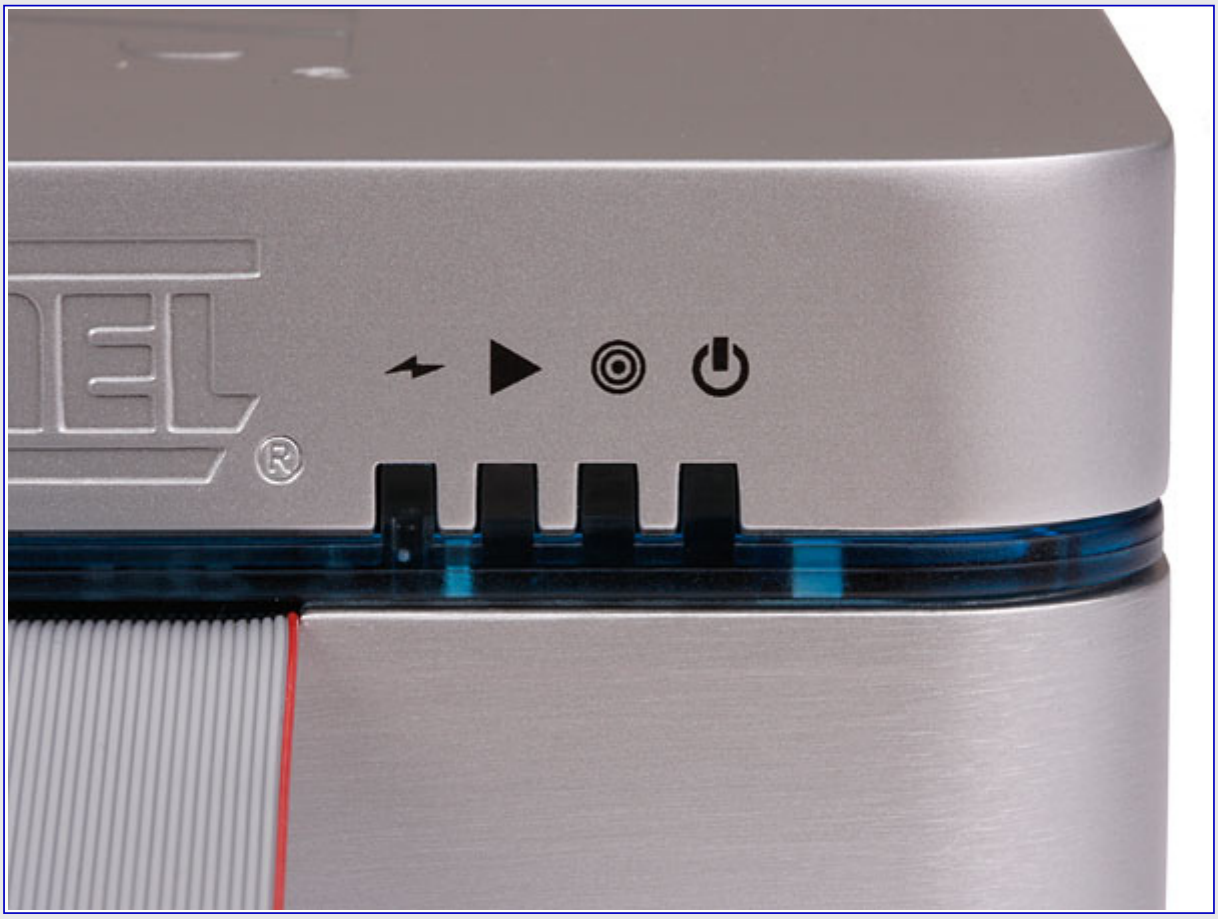


表7-1. LED

LED	ｱｲｺﾝ	説明
主電源		主基板の電源がOKの時に赤
目的対象電源		目的対象の電源がOKの時に緑。点滅は目的対象の電源異常を示します。ﾃﾞﾊﾞｯｸﾞ作業接続が試みられている以外は全く点灯しません。
目的対象走行		目的対象が走行している時に緑。目的対象が停止している時に橙。
通信		ﾎｽﾄ ｺﾝﾋﾟｭｰﾀとの作業が活性の時に緑。
青帯		この帯はﾃﾞﾊﾞｯｶﾞ内部のFPGAが読み込まれる時に必ず点灯します。これは今のところ機能的な重要性を持ちません。Atmel Studioがこれに接続するまでFPGAがﾌﾟﾛｸﾞﾗﾐﾝｸﾞされないため、この帯が更新直後に点灯しないことに注意してください。

7.2. 背面ﾊﾟﾈﾙ

Atmel AVR ONE!の背面ﾊﾟﾈﾙはDCｼﾞｬｯｸ、電源ｽｲｯﾁ、USBｺﾈｸﾀを収容します。上部の張り紙は通番と製造日付を示します。技術支援を求める時にこれらの詳細を含めてください。

図7-2. AVR ONE! 背面ﾊﾟﾈﾙ

7.3. 探針

Atmel AVR ONE!本体部は以下の能力を持つ探針と共にやって来ます。

- 10ﾋﾟﾝ ｺﾈｸﾀ　　：目的対象電圧範囲：1.65～5.5V、ｲﾝﾀｰﾌｪｰｽ周波数：32kHz～33MHz
- MICTOR38ｺﾈｸﾀ ：目的対象電圧範囲：1.65～3.6V、AUXｲﾝﾀｰﾌｪｰｽ周波数：最大200MHz

代替の探針を別に買うことができます。この探針は元の探針と同じに見えますが、以下の能力を持ちます。

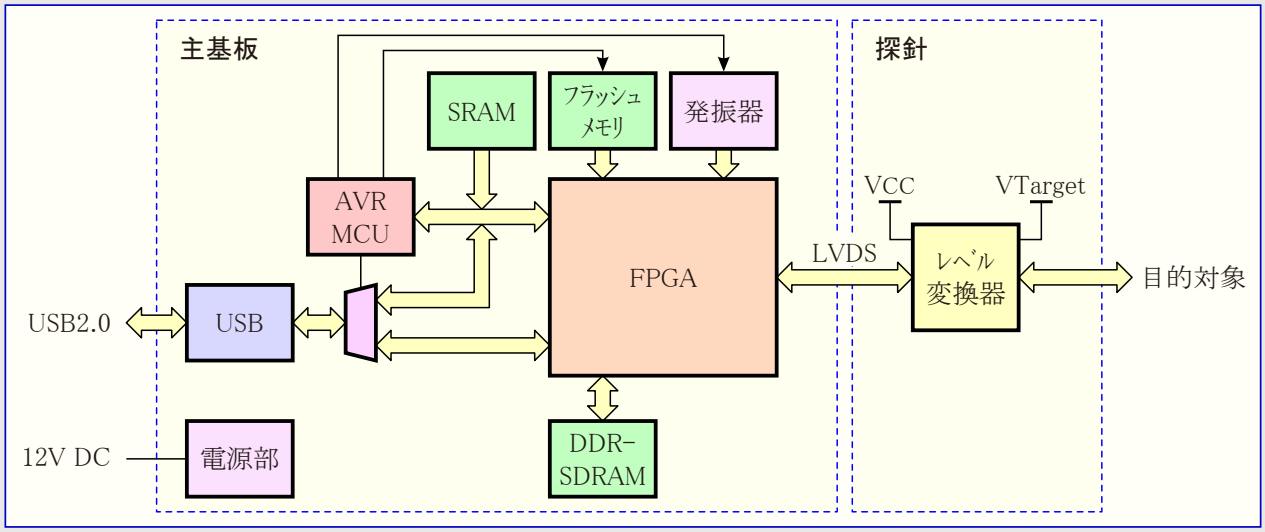
- 10ﾋﾟﾝ ｺﾈｸﾀ　　：目的対象電圧範囲：1.65～5.5V、ｲﾝﾀｰﾌｪｰｽ周波数：32kHz～33MHz
- MICTOR38ｺﾈｸﾀ ：目的対象電圧範囲：1.65～5.5V、AUXｲﾝﾀｰﾌｪｰｽ周波数：最大75MHz

AVR ONE!ﾌｧｰﾑｳｪｱは探針が接続されたことを自動的に検出して、元の3.3V探針が使われた時にMICTOR38ｺﾈｸﾀを通した5.5V目的対象への接続を拒否します。JTAGの10ﾋﾟﾝ ｺﾈｸﾀが両探針で同じ能力を持つことは注意に値します。違うのはMICTOR38ｺﾈｸﾀだけです。

7.4. 基本構造説明

Atmel AVR ONE!の基本構造は図7-3. AVR ONE!構成図で示されます。

図7-3. AVR ONE!構成図



7.4.1. Atmel AVR ONE!主基板

電力は12V DCコネクタを経由して供給されます。USBポートはホスト通信にだけ使われ、ホストから電流を引き出しません。Atmel AVR ONE!主基板の心臓部は目的対象インターフェース信号処理用のFPGAと対にされたATmega1280 AVRマイクロ コントローラです。目的対象インターフェースは概ね1kHz～64MHzの範囲で周波数を供給する能力の外部クロック発生器によってクロック駆動されます。小容量の外部SRAMはAVR MCUに接続され、デバッグ作業中のシンボル情報の格納に使われ、一方より大きくより高速なDDR-SDRAMは追跡メッセージ緩衝部としてFPGAだけによって使われます。

7.4.2. Atmel AVR ONE!探針

Atmel AVR NOE!主基板と探針間の通信は探針ケーブルを伝うLVDS信号によって行われます。LVDS送受信部は目的対象の動作電圧とAVR ONE!の内部電圧レベル間で信号を移動するレベル変換器に接続されます。レベル変換器の外側は物理的に目的対象コネクタへ接続されます。レベル変換器はVTrefから直接ではなく、VTref電圧に一致する緩衝された電源から給電されます。JTAGチャネルは1.65～5.5Vの範囲で33MHzまで動作することができ、一方AUXチャネルは3.6Vで200MHzまでの動作能力です。

最良の結果のために目的対象応用PCB上の高速AUX信号を正しく終端することが推奨されます。

目的対象応用への探針接続方法の更なる情報については「AVR ONE!の接続」章をご覧ください。

8. ソフトウェア統合

8.1. Atmel Studio

Atmel AVR ONE!は全てのAtmel AVR 8/32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のためにAtmel Studioと共に使うことができます。

より多くの情報についてはAtmel Studio使用者の手引きを調べてください。

9. コマンド行ユーティリティ

Atmel StudioはAVR ONE!を使って目的対象をプログラミングするのに使うことができるatprogramと呼ばれるコマンド行ユーティリティと共にやって来ます。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 7.0 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 7.0¥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次の命令を入力してください。

```
atprogram --help
```

10. 高度なデバッグ技術

10.1. Atmel AVR 32ビット マイクロ コントローラ

10.1.1. EVTI/EVTOの使い方

EVTIとEVTOのピンは或る状況で気楽なデバッグに使うことができる機能を提供します。EVTIは目的対象デバイスに事象を合図するのに使われ、一方EVTOは目的対象デバイスからの出力を合図する事象です。38ピンのMICTORコネクタは目的対象デバイス上のこれらのピンへのアクセスをAtmel AVR ONE!に許す、これら2つの信号を含みます。EVTOはそれを読めるべくAVR ONE!用の10ピンJTA Gコネクタの7番ピンにも接続されます。ハードウェア0版の探針に対して目的対象電圧が3.6V以下の場合、EVTOがJTAG 10ピン コネクタ上でだけ利用可能なことに注意してください。デバッグ中で外部的にこれらのピンを使うために、応用基板上のAVR ONE! MICTORコネクタからそれらを切断することが推奨されます。

EVTIは以下の目的で使うことができます。

- 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。これはAtmel AVR 32ビット マイクロ コントローラによって必要とされないNexus準拠機能です。EVTIでのHighからLowへの遷移は追跡同期メッセージを生成するように(DC内のEICビットを使って)構成設定することができます。AVR ONE!はそれらが追跡の再構成に必要とされないため、それらのメッセージを無視するでしょう。

EVTOは以下の目的で使うことができます。

- 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うことができます。
- 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。AVR目的対象はメッセージが追跡送出待ち行列に追加される毎に交互変化するように構成設定することができます。これは追跡出力に対してもっと正確なタイミング情報を得るのに使うことができます。この機能は現在AVR ONE!で支援されません。

10.2. Atmel megaAVR目的対象

10.2.1. I/Oデバッグ レジスタ (IDR)

OCDデバッガはこれが停止動作形態の間にAtmel AVR目的対象デバイスの内部をアクセスするためにメモリ割り当てされたOCDRレジスタを利用します。走行動作形態時、目的対象で走行する応用はこのレジスタへ値を書くことができます。OCDシステムはその後にこれをデバッガへ合図し、そしてデータを取得してそれが表示されるGUI前処理部へ渡します。故に応用はデバガへ素朴なデバッグ メッセージを与えることができます。

注: これよりも高い頻度でこれを書くことが信頼に足る結果を生じないようにIDR値は固定間隔(100ms)でポーリングされます。

注: AVR目的対象デバイスがデバッグ中に電力喪失状況を体験した場合、偽のIDRメッセージに帰着するかもしれません。これは電圧低下がそれの最小動作電圧以下になる時にデバッガがAVRのポーリングを続けるためです。

10.3. デバッグWIRE目的対象

10.3.1. 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、デバッガは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、デバッガ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、デバッガ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。極端な場合では、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。

11. 特別な考慮

11.1. 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ｸﾛｯｸが使われます。OCDｸﾛｯｸはJTAGｲﾝﾀｰﾌｪｰｽが使われている場合にJTAGのTCK、またはPDIｲﾝﾀｰﾌｪｰｽが使われている場合にPDI_CLKのどちらかです。

停止動作でのSDRAM再活性(ﾘﾌﾚｯｼｭ)

OCDが停止動作の時に、MCUは上の項で記述されるように、PDIまたはJTAGのｸﾛｯｸによってｸﾛｯｸ駆動されます。ﾃﾞﾊﾞｯｶﾞやOCDによってこの周波数を知られることがないため、低再活性周期($20)が自動的に使われます。この値は使用者によって変更することはできず、OCDｸﾛｯｸ周波数とSDRAMﾃﾞﾊﾞｲｽの全ての組み合わせに合致しないかもしれません。停止動作でSDRAMの問題が観測される場合、OCDｸﾛｯｸ周波数の調整を試みてください。

停止動作での入出力単位部

初期のAtmel megaAVRﾃﾞﾊﾞｲｽと異なり、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。

ﾊｰﾄﾞｳｪｱ中断点

2つのｱﾄﾞﾚｽ比較器と2つの値比較器で4つのﾊｰﾄﾞｳｪｱ中断点(ﾌﾞﾚｰｸ ﾎﾟｲﾝﾄ)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(ﾌﾟﾛｸﾞﾗﾑまたはﾃﾞｰﾀ)でなければなりません。
- 全てのﾃﾞ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。
- ｱﾄﾞﾚｽ範囲が使われる場合は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つの単一ﾃﾞｰﾀまたはﾌﾟﾛｸﾞﾗﾑ ｱﾄﾞﾚｽの中断点
- 1つのﾃﾞｰﾀまたはﾌﾟﾛｸﾞﾗﾑ ｱﾄﾞﾚｽの範囲中断点
- 単一値比較を持つ2つの単一ﾃﾞｰﾀ ｱﾄﾞﾚｽ中断点
- ｱﾄﾞﾚｽ範囲、値範囲、または両方を持つ1つのﾃﾞｰﾀ中断点

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、ﾃﾞｰﾀ中断点がﾌﾟﾛｸﾞ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います。ﾃﾞﾊﾞｯｸﾞ時、ﾘｾｯﾄのﾌﾟﾙｱｯﾌﾟは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 ｺﾝﾃﾞ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ﾃﾞﾊﾞｯｸﾞ

ATxmegaA1系の初期版にはﾃﾞﾊﾞｲｽが或る休止動作中に許可されたOCDが妨げられるﾊﾞｸﾞが有りました。ﾃﾞﾊﾞｯｸﾞに戻るのに使う2つの方法があります。

- Atmel AVR ONE!任意選択ﾀﾞｲｱﾛｸﾞへ移行し、”Always activate external reset when reprogramming device”を許可してください。
- ﾁｯﾌﾟ消去を実行してください。

このﾊﾞｸﾞを引き起こす休止動作は以下です。

- ﾊﾟﾜｰﾀﾞｳﾝ
- ﾊﾟﾜｰｾｰﾌﾞ
- ｽﾀﾝﾊﾞｲ
- 拡張ｽﾀﾝﾊﾞｲ

このﾊﾞｸﾞの結果は目的対象がこれらの休止動作形態の時に、Liveﾃﾞﾊﾞｯｸﾞを使う目的対象への取り付けが動かないことです。ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ﾀﾞｲｱﾛｸﾞはこのﾊﾞｸﾞによって影響を及ぼされません。

11.2. Atmel megaAVR OCDとﾃﾞﾊﾞｯｸﾞWIRE OCD

入出力周辺機能

殆どの周辺機能は例えﾌﾟﾛｸﾞﾗﾑ実行が中断点(ﾌﾞﾚｰｸ ﾎﾟｲﾝﾄ)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ﾋﾞｯﾄが設定されます。送信完了(TXC)ﾌﾗｸﾞが設定(1)され、例え実際のﾃﾞﾊﾞｲｽでは通常、より遅くに起こるとしても、ｺｰﾄﾞ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って構成設定可能)
- ｳｫｯﾁﾄﾞｯｸﾞ ﾀｲﾏ (ﾃﾞﾊﾞｯｸﾞ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、或るﾀｲﾐﾝｸﾞの問題を避けるために注意が払われるべきです。例えば次のｺｰﾄﾞです。

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

このｺｰﾄﾞの通常走行時、ﾃﾞｰﾀはIN操作によって採取される時に物理的に未だﾋﾟﾝにﾗｯﾁされていないため、TEMPﾚｼﾞｽﾀは$AAを読み戻しません。PINﾚｼﾞ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、このｺｰﾄﾞは常に$AAを与えます。

単一段実行とﾀｲﾐﾝｸﾞ

或るﾚｼﾞ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、そのようなｺｰﾄﾞを通した単一段実行はﾀｲﾐﾝｸﾞ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ｸﾞ必要条件で成功裏のﾚｼﾞｽﾀを読みまたは書きを行うには、全速でﾃﾞﾊﾞ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。これはｺｰﾄﾞを実行するのにﾏｸﾛまたは関数を使う、またはﾃﾞﾊﾞｯｸﾞ環境でｶｰｿﾙまで実行の機能を使うことによって行うことができます。

16ﾋﾞｯﾄ ﾚｼﾞｽﾀのｱｸｾｽ

Atmel AVR周辺機能は代表的に8ﾋﾞｯﾄ ﾃﾞｰﾀ ﾊﾞｽ経由でｱｸｾｽすることができる多数の16ﾋﾞｯﾄ ﾚｼﾞｽﾀ(例えば、16ﾋﾞｯﾄ ﾀｲﾏ/ｶｳﾝﾀのTCNTn)を含みます。16ﾋﾞｯﾄ ﾚｼﾞｽﾀは2つの読みまたは書きの操作を用いてﾊﾞｲﾄ ｱｸｾｽされなければなりません。16ﾋﾞｯﾄ ｱｸｾｽの中間での中断、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ﾚｼﾞｽﾀ ｱｸｾｽ

或るﾚｼﾞ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。このようなﾚｼﾞｽﾀは読むことによって解除(0)されるﾌﾗｸﾞを含むそれらや、緩衝されたﾃﾞｰﾀ ﾚｼﾞｽﾀ(例えば、UDR)を含みます。OCDﾃﾞﾊﾞｯｸﾞの意図された非侵入的性質を守るため、ｿﾌﾄｳｪｱ前処理部は停止動作時にこれらのﾚｼﾞｽﾀの読み込みを防ぎます。加えて、いくつかのﾚｼﾞ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。それらのﾚｼﾞｽﾀは読み出し専用です。例えば、以下です。

- 何れかのﾋﾞｯﾄへの'1'書き込みによって解除(0)されるﾌﾗｸﾞのﾌﾗｸﾞ ﾚｼﾞｽﾀ
- 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。これらのﾚｼﾞｽﾀはｱｸｾｽ不能です。

11.3. Atmel megaAVR OCD (JTAG)

ｿﾌﾄｳｪｱ中断点

ATmega128[A]は初期のOCD単位部を含むため、ｿﾌﾄｳｪｱ中断点(ﾌﾞﾚｰｸ ﾎﾟｲﾝﾄ)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ﾃﾞﾊﾞｯｸﾞ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ﾃﾞﾊﾞｯｸﾞ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ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ時、TCK周波数は目的対象ﾃﾞﾊﾞｲｽの最大周波数評価によって制限され、使わ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、周波数がﾃﾞﾊﾞｲｽ毎に変わ得て、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。

JTAGENとOCDENのﾋｭｰｽﾞ

JTAGｲﾝﾀｰﾌｪｰｽは既定によってﾌﾟﾛｸﾞﾗﾑ(0)されているJTAGENﾋｭｰｽﾞによって許可されます。これはJTAGﾌﾟﾛｸﾞﾗﾐﾝｸﾞ 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ﾋｭｰｽﾞをﾌﾟﾛｸﾞﾗﾑ(0)することができます(既定でのOCDENは非ﾌﾟﾛｸﾞﾗﾑ(1))。これはﾃﾞﾊﾞｲｽのﾃﾞﾊﾞｯｸﾞ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ﾋｭｰｽﾞが非ﾌﾟﾛｸﾞﾗﾑ(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ﾋｭｰｽﾞが意図せずに禁止された場合、SPIまたはPPのﾌﾟﾛｸﾞﾗﾐﾝｸﾞ法を使ってのみ再許可することができます。

JTAGENﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(0)されたなら、JTAGｲﾝﾀｰﾌｪｰｽはJTDﾋﾞｯﾄを設定(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。これはﾃﾞﾊﾞｯｸﾞ不可ｺｰﾄﾞにし、ﾃﾞﾊﾞｯｸﾞ作業を試みる時に実行されべきではありません。ﾃﾞﾊﾞｯｸﾞ作業開始時にこのようなｺｰﾄﾞが既に実行された場合、Atmel AVR ONE!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ﾃﾞﾊﾞ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ﾋﾟﾝは代替ﾋﾟﾝ機能に使うことができません。それらはﾌﾟﾛｸﾞﾗﾑ ｺｰﾄﾞからJTDﾋﾞｯﾄの設定(1)、またはﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ｲﾝﾀｰﾌｪｰｽを通してJTAGENﾋｭｰｽﾞの解除(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専用ﾋﾟﾝに留まります。

IDR事象

応用ﾌﾟﾛｸﾞﾗﾑがﾃﾞﾊﾞｯｸﾞされつつあるAVRﾃﾞﾊﾞｲｽのOCDRﾚｼﾞｽﾀにﾊﾞｲﾄ ﾃﾞｰﾀを書く時に、AVR ONE!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ｼﾞ ｳｨﾝﾄﾞｳに表示します。IDRﾚｼﾞｽﾀは100ms毎にﾎﾟｰﾘﾝｸﾞされ、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。ﾃﾞﾊﾞｯｸﾞ中にAVRﾃﾞﾊﾞ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AVR ONE!が未だﾃﾞﾊﾞｲｽをﾎﾟｰﾘﾝｸﾞし得るために起こります。

11.4. ﾃﾞﾊﾞｯｸﾞWIRE OCD

ﾃﾞﾊﾞｯｸﾞWIRE(dW)ﾋﾟﾝは物理的に外部ﾘｾｯﾄ(RESET)と同じﾋﾟﾝに配置されます。従ってﾃﾞﾊﾞｯｸﾞ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。

ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽが機能するためには、目的対象ﾃﾞﾊﾞｲｽでﾃﾞﾊﾞｯｸﾞWIRE許可(DWEN)ﾋｭｰｽﾞが設定(0)されなければなりません。このﾋｭｰｽﾞはAtmel AVRﾃﾞﾊﾞｲｽが工場から出荷される時に既定によって非ﾌﾟﾛｸﾞﾗﾑ(1)にされます。ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽそれ自身はこのﾋｭｰｽﾞを設定することができません。DWENﾋｭｰｽﾞを設定するにはSPI動作が使わ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ﾋﾟﾝが接続されていればこれを自動的に処理します。Atmel Studioのﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ﾀﾞｲｱﾛｸﾞからSPIﾌﾟﾛｸﾞﾗﾐﾝｸﾞを使うこともできます。以下の2つのどちらかを行います。

- ﾃﾞﾊﾞｯｸﾞWIRE部でﾃﾞﾊﾞｯｸﾞ作業の開始を試みてください。ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽが許可されていなければ、Atmel Studioは再試行を提供するか、またはSPIﾌﾟﾛｸﾞﾗﾐﾝｸﾞを使ってﾃﾞﾊﾞｯｸﾞWIREを許可しようと試みます。完全なSPIﾍｯﾀﾞ接続があれば、ﾃﾞﾊﾞｯｸﾞ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ｽﾞを変更するのに必要とされます。
- SPI動作でﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ﾀﾞｲｱﾛｸﾞを開き、識票が正しいﾃﾞﾊﾞｲｽと一致することを確認してください。ﾃﾞﾊﾞｯｸﾞWIREを許可するためにDWENﾋｭｰｽﾞをﾁｪｯｸしてください。

注: SPIENﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(0)され、RSTDISBLﾋｭｰｽﾞが非ﾌﾟﾛｸﾞﾗﾑ(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ﾃﾞﾊﾞｯｸﾞWIRE動作でﾃﾞﾊﾞｲｽを動作停止にし、DWEN設定を元に戻すのに高電圧ﾌﾟﾛｸﾞﾗﾐﾝｸﾞが必要とされます。

ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽを禁止するには、DWENﾋｭｰｽﾞを非ﾌﾟﾛｸﾞﾗﾑ(1)にするのに高電圧ﾌﾟﾛｸﾞﾗﾑを使ってください。代わりに、ﾃﾞﾊﾞｯｸﾞWIREを一時的に禁止するのにﾃﾞﾊﾞｯｸﾞWIREそれ自身を使い、SPIENﾋｭｰｽﾞが設定(0)されていれば、SPIﾌﾟﾛｸﾞﾗﾐﾝｸﾞ実行を許します。

注: SPIENﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(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、高電圧ﾌﾟﾛｸﾞﾗﾐﾝｸﾞが使用されなければなりません。

- ﾃﾞﾊﾞｯｸﾞ作業の間、'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。ﾃﾞﾊﾞｯｸﾞWIREが一時的に禁止され、Atmel StudioはDWENﾋｭｰｽﾞを非ﾌﾟﾛｸﾞﾗﾑ(1)にするのにSPIﾌﾟﾛｸﾞﾗﾐﾝｸﾞを使います。

ﾌﾟﾛｸﾞﾗﾑ(0)されたDWENﾋｭｰｽﾞ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。従ってﾃﾞﾊﾞｯｸﾞWIREが使われない時は常にDWENﾋｭｰｽﾞが禁止されるべきです。

ﾃﾞﾊﾞｯｸﾞWIREが使われる目的対象応用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)線のﾌﾟﾙｱｯﾌﾟ抵抗は10kΩよりも小さく(強力で)あってはなりません。ﾌﾟﾙｱｯﾌﾟ抵抗はﾃﾞﾊﾞｯｶﾞ ﾂｰﾙがこれを提供するため、機能的にﾃﾞﾊﾞｯｸﾞWIREに必要ではありません。
- /RESETﾋﾟﾝの直接VCC接続はﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽを失敗させます。
- /RESETﾋﾟﾝに接続されるどの安定化ｺﾝﾃﾞﾝｻも、それらがｲﾝﾀｰﾌｪｰｽの正しい動作を妨げるため、ﾃﾞﾊﾞｯｸﾞ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線上の他の活性な駆動部は、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。

目的対象ﾃﾞﾊﾞｲｽの施錠ﾋﾞｯﾄを決してﾌﾟﾛｸﾞﾗﾑ(0)してはなりません。ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽは正しく機能するために施錠ﾋﾞｯﾄの解除(1)が必要です。

11.5. Atmel AVR UC3 OCD

いくつかのAtmel AVR UC3ﾃﾞﾊﾞｲｽでJTAGﾎﾟｰﾄは既定で許可されません。これらのﾃﾞﾊﾞｲｽ使用時、Atmel AVR ONE!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。

/RESETﾋﾟﾝに接続されるどの安定化ｺﾝﾃﾞﾝｻも、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。この線上の弱い外部ﾌﾟﾙｱｯﾌﾟが推奨されます。

aWire通信のﾎﾞｰﾚｰﾄはﾃﾞｰﾀがそれら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。AVR ONE!はｼｽﾃﾑ ｸﾛｯｸがより低いことを自動的に検知し、それに従ってﾎﾞ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。ﾃﾞﾊﾞｯｸﾞ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のﾎﾞｰﾚｰﾄはﾂｰﾙ ﾁｪｰﾝで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。

11.6. Atmel AVR UC3停止動作

UC3 Lにような、いくつかのAtmel AVR UC3ﾃﾞﾊﾞｲｽは停止(ｼｬｯﾄﾀﾞｳﾝ)と呼ばれる特別な休止動作形態を支援します。この動作形態を使う時に、ﾃﾞﾊﾞｲｽはVDDINで3.3Vによって給電され、その後に内部調整器がｺｱに1,8Vを供給します。これに加えて1.8VはVDDIOに給電するのに使うことができます。ﾃﾞﾊﾞｲｽが停止動作形態になる時にｺｱとほとんどの入出力の両方が電力断に帰着する内部調整器をOFFに切り替えます。ﾃﾞﾊﾞｲｽを再び起こすにはﾘｾｯﾄ ﾋﾟﾝで外部ﾘｾｯﾄによってﾘｾｯﾄされなければなりません。ﾘｾｯﾄ ﾋﾟﾝはVDDINによって給電され、一方JTAG線はVDDIOによって給電されます。停止動作形態のより多くの情報については使うﾃﾞﾊﾞｲｽのﾃﾞｰﾀｼｰﾄを参照してください。このような構成設定での目的対象ﾃﾞﾊﾞｯｸﾞ時、いくつかの特別な考慮が必要とされます。

aWare

aWareを使うﾃﾞﾊﾞｯｸﾞ時、ﾘｾｯﾄ線は使う信号線だけです。VTref(Atmel AVR ONE!のJTAGｺﾈｸﾀの4番ﾋﾟﾝ)はaWareが動作するようにVDDIN(3.3V)に接続されなければなりません。

JTAG

JTAGを使うﾃﾞﾊﾞｯｸﾞ時、AVR ONE!が停止動作形態から目的対象を起こすことができるように、ﾘｾｯﾄ線をAVR ONE!のJTAGｺﾈｸﾀの6番ﾋﾟﾝのnSRSTに配線することも重要です。AVR ONE!のJTAGｺﾈｸﾀの4番ﾋﾟﾝのVTrefはVDDIO(1,8V)に接続されなければなりません。JTAG使用時の問題は目的対象のﾘｾｯﾄ ﾋﾟﾝが3.3VのVDDINによって給電される一方で、AVR ONE!探針が1.8VのVTrefによって給電されることです。この構成設定はﾊｰﾄﾞｳｪｱを大変危険にするかもしれませんが、AVR ONE!はﾘｾｯﾄ線をLowにだけ駆動し、決してHighに駆動せず、故にどんな競合もないでしょう。けれども、ﾘｾｯﾄ線を1.8Vに引っ張るAVR ONE!探針上の強力なﾌﾟﾙｱｯﾌﾟがあります。このﾌﾟﾙｱｯﾌﾟはUC3目的対象内の内部ﾌﾟﾙｱｯﾌﾟよりも強力で、故にﾘｾｯﾄ線は約2Vに留まります。目的対象がこれをﾘｾｯﾄでのLow値として読むのを避けるため、いくつかの行動が取られなければなりません。ﾘｾｯﾄ線での外部3.3kΩ追加または目的対象のVDDINを2.2Vかより低くする、のどちらかの2つの任意選択があります。

12. 障害対策

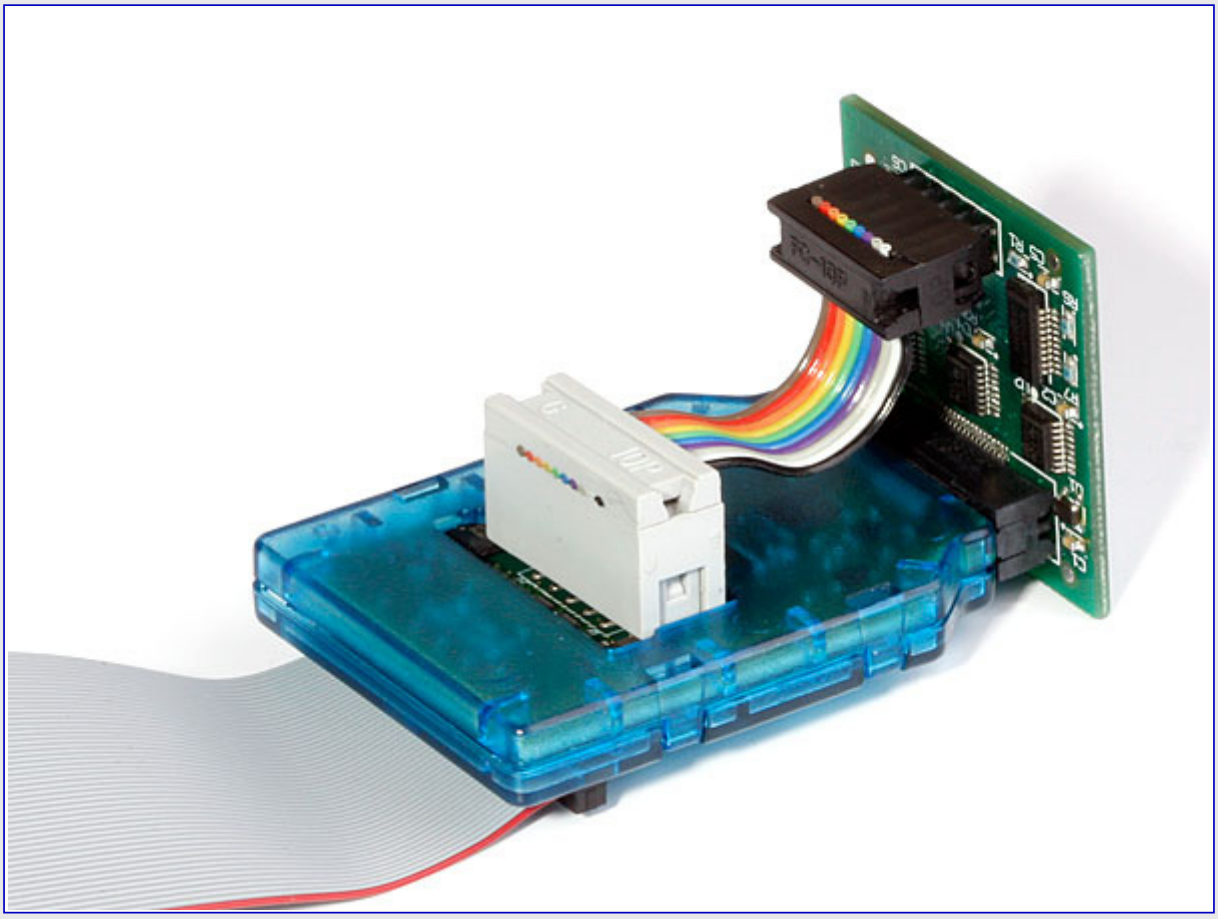
12.1. 自己検査

Atmel AVR ONE!はIDCｹｰﾌﾞﾙを持つ自己検査ｱﾀﾞﾌﾟﾀと共に出荷されます。何らかの理由のためにAVR ONE!ｹｰﾌﾞﾙと探針が正しく機能していないと疑われる場合、自己検査を走らせることができます。自己検査ｱﾀﾞﾌﾟﾀは目的対象の通信回路での診断実行を基板上のMCUで可能にする環状戻りｱﾀﾞﾌﾟﾀです。これは探針、ｹｰﾌﾞﾙ、主基板の部品と暗黙的に検査ｱﾀﾞﾌﾟﾀそれ自身を含みます。

12.1.1. 接続

検査を走らせるにはｱﾀﾞﾌﾟﾀとｹｰﾌﾞﾙが**図12-1. AVR ONE!検査ｱﾀﾞﾌﾟﾀの接続**で示されるように接続されなければなりません。ｴｯｼﾞ ｺﾈｸﾀは検査ｱﾀﾞﾌﾟﾀ上のMictor38ｿｹｯﾄに直接勘合します。良好な接続を保証するためにしっかりと押してください。検査ｱﾀﾞﾌﾟﾀ ｹｰﾌﾞﾙのｵｽ部分は探針に接続されます。例えｹｰﾌﾞﾙに関してｺﾈｸﾀが完全に正しい向きでだけ挿入するように穴が鍵にされているとは言え、不正な向きを使った場合に未だ中途半端に挿入している可能性があることに注意してください。ｹｰﾌﾞﾙのﾒｽ部分は検査ｱﾀﾞﾌﾟﾀ上のﾋﾟﾝ ﾍｯﾀﾞと勘合します。これが正しく勘合されていることを確認してください。

図12-1. AVR ONE!検査ｱﾀﾞﾌﾟﾀの接続



12.1.2. 開始

自己検査はｺﾏﾝﾄﾞ行(未実装)またはAtmel Studioのどちらからでも開始することができます。自己検査はAtmel StudioからR利用可能なﾂｰﾙ ﾋﾞｭｰ(View⇒Available Atmel AVR Tools)を開いて、その後に検査を必要とするAtmel AVR ONE!を右ｸﾘｯｸして引き落としﾒﾆｭｰで自己検査(Self Test)を選んでください。

12.1.3. 診断結果の使用法

自己検査は異常を見つけたなら、失敗信号を含むMICTOR38ｺﾈｸﾀと10ﾋﾟﾝ ｺﾈｸﾀの両方、またはMICTOR38ｺﾈｸﾀだけのどちらかで告げます。これが起きた場合、接続を2重に調べて検査を再試行してください。接続が緩いのかもしれず、またJTAGｺﾈｸﾀの場合はﾋﾟﾝ ﾍｯﾀﾞへ不正に接続されているかもしれません。異常ﾒｯｾｰｼﾞが存続する場合、何処かにﾊｰﾄﾞｳｪｱの問題があるようです。Atmelｽﾄｱ(store.atmel.com)で新しい探針とｹｰﾌﾞﾙを別に注文することができます。

最もありそうなﾊｰﾄﾞｳｪｱ異常の元は探針上のMictor38または10ﾋﾟﾝ ｺﾈｸﾀ内の何れかの信号をAtmel AVR ONE!探針の最大限度を超える電圧に接続した後です。探針ﾊｰﾄﾞｳｪｱの最大定格は「Atmel AVR ONE!の特徴」で与えられます。

注: 自己検査は何が悪いのかの助言を与え得るだけです。これはどの部品が悪いのか正確に告げることはできません。自己検査の主な目的はﾊｰﾄﾞｳｪｱが動くか否かのどちらかかを明らかにすることです。

12.2. 障害対策の手引き

表12-1. 障害対策の手引き

目的対象 型式/系統	問題	有り得る原因	解決策
–	電源LEDが点灯しない。	DC供給電圧が不十分または不正な極性。	ｷｯﾄと共に提供された正しい電源を使っていることを調べてください。他の電源を使っている場合、それが中心+のｺﾈｸﾀを持つことを確実にしてください。
megaAVR AVR XMEGA	JTAGﾃﾞﾊﾞｯｸﾞ開始、その後、突然に失敗。	MCU(C)SRﾚｼﾞｽﾀのJTAG禁止ﾋﾞｯﾄが応用によって偶然に書かれた。	制御を回復するためにﾘｾｯﾄをLowに保ち、JTAG禁止ﾋﾞｯﾄが書かれないようにｺｰﾄﾞを変更してください。
megaAVR AVR XMEGA	ﾃﾞﾊﾞｲｽにｺｰﾄﾞを書き込むのにAtmel AVR ONE!を使用した後にｴﾐｭﾚｰﾀがもはや動かない。	1. JTAG許可ﾋｭｰｽﾞが禁止されている。 2. ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ｲﾝﾀｰﾌｪｰｽが未だ活性。OCDとﾌﾟﾛｸﾞﾗﾐﾝｸﾞの同時使用は不可能です。	1. JTAG許可ﾋｭｰｽﾞをﾌﾟﾛｸﾞﾗﾑ(0)してください。 2. ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ｲﾝﾀｰﾌｪｰｽを閉じ、その後にｴﾐｭﾚｰﾀ動作形態へ移行してください。
tinyAVR megaAVR AVR XMEGA	Atmel AVR ONE!がAtmel Studioまたは他の前置ｿﾌﾄｳｪｱによって検出されたが、目的対象ﾃﾞﾊﾞｲｽに接続しない。	JTAG：JTAG許可ﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(0)されていない。 ﾃﾞﾊﾞｯｸﾞWIRE：DWENﾋｭｰｽﾞがﾌﾟﾛｸﾞﾗﾑ(0)されていない。	JTAG：JTAG許可ﾋｭｰｽﾞをﾌﾟﾛｸﾞﾗﾑ(0)するのに他のﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ｲﾝﾀｰﾌｪｰｽを使ってください。 ﾃﾞﾊﾞｯｸﾞWIRE：DWENﾋｭｰｽﾞをﾌﾟﾛｸﾞﾗﾑ(0)するのに他のﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ｲﾝﾀｰﾌｪｰｽを使ってください。
megaAVR AVR XMEGA AVR UC3	JTAGのﾌﾟﾛｸﾞﾗﾐﾝｸﾞとﾃﾞﾊﾞｯｸﾞが不安定、または全てで動かない。	いくつかの目的対象基板に関して、ｲﾝﾀｰﾌｪｰｽ線上に幾許かのﾘﾝｷﾞﾝｸﾞが有るかもしれない。	JTAG線、特にTCKに直列抵抗を追加してください。しかし、直列抵抗はTMSとTDIでも有用で有り得ます。殆どの場合で約68Ωの値が適合するでしょう。直列抵抗追加後はJTAGｸﾛｯｸ周波数を減らさなければならないかもしれないことに注意してください。
–	電圧が存在しないとのﾒｯｾｰｼﾞをAtmel Studioが与える。	1. 目的対象基板で電力なし。 2. VTrefが接続されていない。 3. 目的対象電圧が低すぎる。	1. 目的対象基板に電力を印加してください。 2. JTAGｺﾈｸﾀがVTref信号を含むことを確実にしてください。 3. 目的対象電源が充分な電力を提供できることを確実にしてください。
megaAVR	OCDﾋｭｰｽﾞが禁止されているが、Atmel AVR ONE!を使用して、OCDが未だ可能。	AVR ONE!はOCDﾋｭｰｽﾞが禁止されている場合、自動的にそれをﾌﾟﾛｸﾞﾗﾑ(0)します。	これは正しい動作です。
tinyAVR megaAVR	Atmel StudioのI/OｳｨﾝﾄﾞｳでいくつかのI/Oﾚｼﾞｽﾀが正しく更新されない。	邪魔しない読み戻しが不可能な時に、AVR ONE!はAtmel StudioのI/Oｳｨﾝﾄﾞｳでその位置を更新しません。	このI/O位置を一時ﾚｼﾞｽﾀ内に読んで、ﾃﾞﾊﾞｯｸﾞ中にそれを見てください。これによってどのﾚｼﾞｽﾀが影響を及ぼされるかについての情報に関しては「特別な考慮」章をご覧ください。
ﾃﾞﾊﾞｯｸﾞWIRE支援を持つtinyAVR megaAVR	ﾃﾞﾊﾞｯｸﾞWIRE作業後のSPIﾌﾟﾛｸﾞﾗﾐﾝｸﾞが不可能。	ﾃﾞﾊﾞｯｸﾞWIREｲﾝﾀｰﾌｪｰｽが許可されると、SPIｲﾝﾀｰﾌｪｰｽは禁止されます。	「ﾃﾞﾊﾞｯｸﾞWIRE目的対象への接続」項で記述されるようにSPIｲﾝﾀｰﾌｪｰｽを再許可してください。SPIｲﾝﾀｰﾌｪｰｽを再許可するのにｺﾏﾝﾄﾞ行ｿﾌﾄｳｪｱを使ってください。

次頁へ続く

表12-1 (続き). 障害対策の手引き

目的対象 型式/系統	問題	有り得る原因	解決策
ﾃﾞﾊﾞｯｸﾞWIRE支援を持つtinyAVR megaAVR	SPIまたはﾃﾞﾊﾞｯｸﾞWIREのどちらの接続も動かない。	SPIとﾃﾞﾊﾞｯｸﾞWIREのｲﾝﾀｰﾌｪｰｽが禁止されています。ﾃﾞﾊﾞｯｸﾞWIREは施錠ﾋﾞｯﾄがﾌﾟﾛｸﾞﾗﾑ(0)されている場合に動きません。	高電圧ﾌﾟﾛｸﾞﾗﾐﾝｸﾞで目的対象に接続してください。SPIまたはﾃﾞﾊﾞｯｸﾞWIREを許可し、ﾃﾞﾊﾞｯｸﾞWIREを使う場合は施錠ﾋﾞｯﾄを解除(1)してください。
tinyAVR megaAVR AVR XMEGA AVR UC3	ﾃﾞﾊﾞｯｸﾞWIREまたはJTAGの使用時に異常ﾒｯｾｰｼﾞまたは他のおかしな動き。	目的対象が(最大周波数対Vccの)安全動作領域外で走行。	目的対象が実際のﾃﾞﾊﾞｲｽ用のﾃﾞｰﾀｼｰﾄの「**電気的特性**」章で記述されるような安全動作領域内で走行することを確実にしてください。周波数をより低くしたり、電圧を増してください。

13. 製品適法性

13.1. RoHSとWEEE

AVR ONE!と全ての付属品は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。

13.2. CEとFCC

AVR ONE!本体は以下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。

- 2004/108/EC(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われます。

- 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わ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。

14. 資料改訂

資料改訂	日付	注釈
32222A	2016年6月	初版資料公開

Atmel® | Enabling Unlimited Possibilities®

Atmel Corporation 1600 Technology Drive, San Jose, CA 95110 USA TEL:(+1)(408) 441-0311 FAX: (+1)(408) 436-4200 | www.atmel.com

---

本応用記述はAtmelのAVR ONE!使用者の手引き(改訂42222A-6/2016)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。