

## AVR057 : ATtiny4/5/9/10/20/40用内部RC発振器校正

Atmel 8ビット マイクロ コントローラ

## 要点

- Atmel® ATtiny4/5/9/10/20/40用8MHz内部RC発振器の校正
- Atmel AVR918(Atmel TPIを使用)とAtmel AVR911(AVROSP)で実装される校正
- ±1%精度で調整可能なRC周波数
- どの動作電圧と温度でもRC発振器調整
- 仕様内のどの周波数でもRC発振器調整

## 序説

ATtiny4/5/9/10/20/40デバイスには前置分周器と校正レジスタ(OSCCAL)を持つ8MHz内部RC発振器が特徴です。これらのデバイスの内部RC発振器は3Vの供給電圧と25°Cで±10%の精度に工場校正されます。しかし、応用の殆どはそれらの基本的な必要条件としてもっと正確なクロックを求めます。そのような応用に対してAVR®は内部RC発振器を校正する方法を提供します。内部RC発振器はOSCCALレジスタを変更することによって±1%精度に使用者校正することができます。

この応用記述はATtiny4/5/9/10/20/40デバイスの8MHz内部RC発振器を校正するための容易な手続を提供します。それはソフトウェア前処理部としてAVROSPでのプログラミングと校正のためにAVR918(Atmel Tinyプログラミング インターフェース(TPI)の使い方)を使います。

## 目次

---

1. 内部RC発振器	3
1.1. 使用者校正	3
2. 校正規約	3
2.1. 作業の流れ	4
3. ハードウェア構成設定	5
4. 即時開始の手引き	6
5. 更新履歴	8

# 1. 内部RC発振器

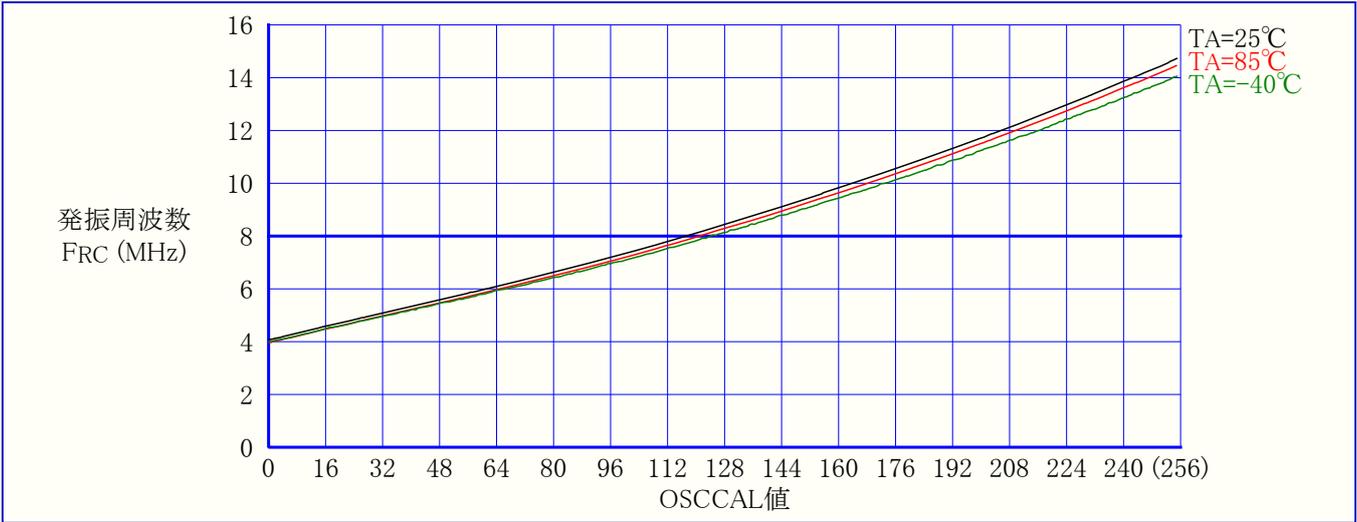
AVR MCUの大多数はクロック元の1つとして校正付き内部RC発振器を持ちます。この内部RC発振器はMCUへどんな外部部品も接続することなく動き、システムクロックとして使うことができます。

製造に於いてAtmel ATtiny4/5/9/10/20/40デバイスの内部RC発振器は3Vで校正されます。この工場校正の精度は±10%以内です。精度に対する設計者の要求がAtmelによる工場での標準校正によって提供され得るものを超える場合、RC発振器の第2校正を実行することが可能です。これを行うことによって±1%内の精度の周波数を得ることが可能です。従って第2校正は発振器の精度または周波数の改善または適合のために実行することができます。

## 1.1. 使用者校正

ATtiny4/5/9/10/20/40デバイスはOSCCAL(OSCillator CALibration:発振器校正)レジスタと呼ばれるレジスタを含みます。内部RC発振器の周波数はOSCCALレジスタに設定された値に応じて変わります。例として図1-1は3Vの供給電圧でのOSCCAL値に関するATtiny10の内部RC発振器周波数の代表的な変動を示します。

図1-1. OSCCAL値に関する内部RC発振器変動 (VCC=3VでのATtiny10に対して)



上図から内部RC発振器の周波数がOSCCAL値だけでなく温度でも変わることが推定できます。従って特定の動作環境に対して内部RC発振器はOSCCALレジスタを変えることによって±1%精度に校正することができます。

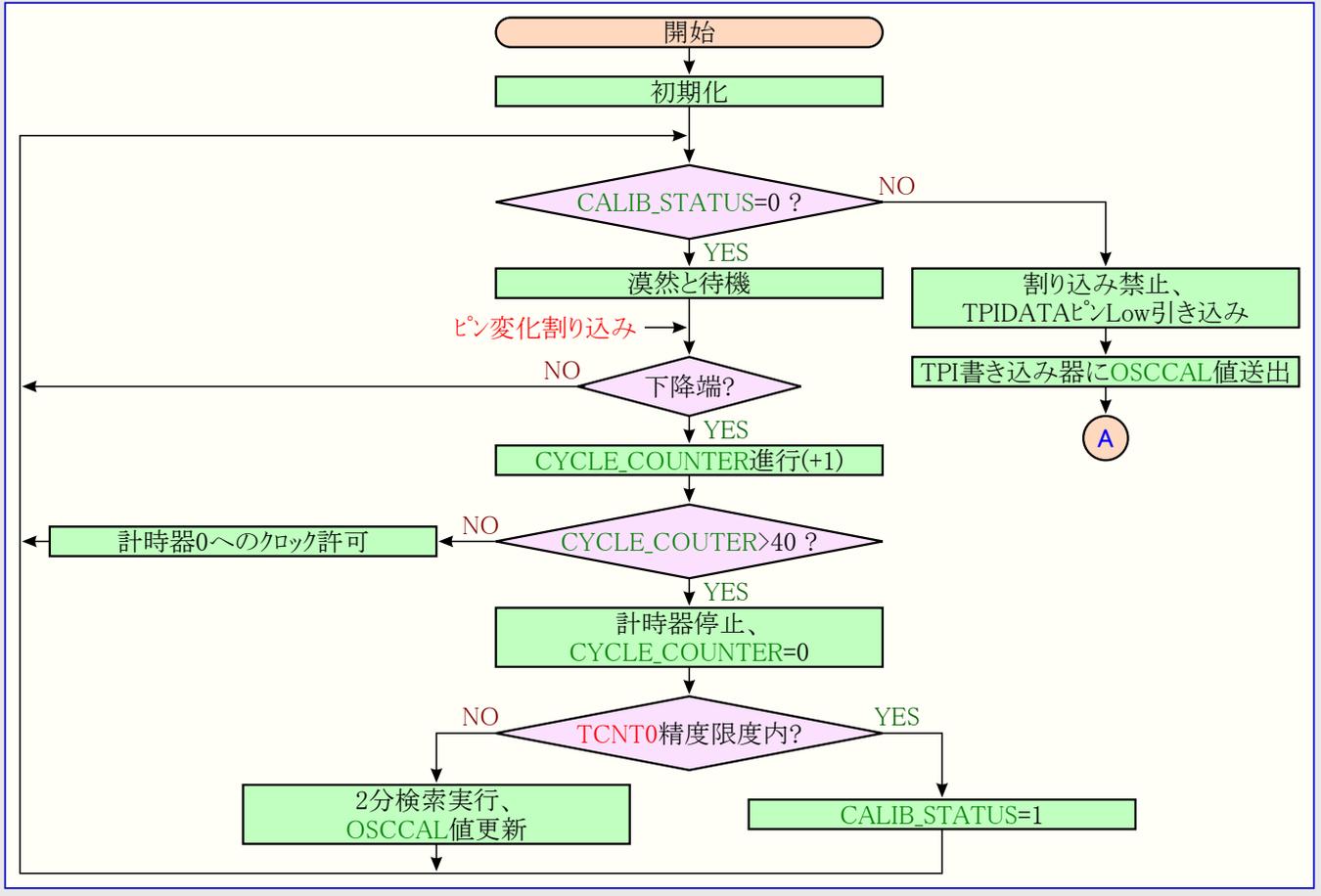
工場校正に対応するOSCCAL値はデバイスの製品識票列内に保存されます。従ってリセット後に毎回、デバイスは保存されたOSCCAL値を製品識票列からOSCCALレジスタ内へ設定します。これは既定によって内部RC発振器を±10%精度で走行させます。使用者応用はその後に指定された精度に達するまで継続的にOSCCALレジスタ値を変更することによって内部RC発振器を校正することができます。

## 2. 校正規約

この応用記述で実装される規約は外部的に印加される32kHz信号に基づいて内部RC発振器を校正します。校正処理はAtmel AVR 918のTPI書き込み器自身で達成されます。TPI書き込み器が校正にどう使われるかの詳細説明については来るべき項で説明されず。

本章は目的対象デバイスがそのTPICLKピンに印加された32kHz信号に基づく校正をどう実行するかを簡単に説明します。図2-1は目的対象MCU側で走行する主校正ルーチンの流れ図を提供します。

図2-1. 主校正ルーチン用流れ図



目的対象のTPIDATAピンに接続されるTPI書き込み器のI/Oピンは、許可された内部プルアップを持つ入力として構成設定されます。32 kHz信号は目的対象のTPICLKピンに印加されます。

## 2.1. 作業の流れ

処理の簡潔な概要のように、この規約はどの前置分周もなしにI/Oクロックで走行する目的対象の16ビット計時器を使います。Atmel ATtiny4/5/9/10については計時器(タイマ/カウンタ)0が使われ、Atmel ATtiny20/40については計時器(タイマ/カウンタ)1が使われます。規約は最初に初期値(代表的に最大の半分)でOSCCALレジスタを初期化し、TPICLKピンに印加された校正クロックの下降端に基づいて計時器を開始します。計時器は40校正クロック周期間走行を続けます。40周期後に計時器は停止され、TCNT値が精度の限界の内側かどうか調べられます。値が最大限度よりも大きければOSCCALが減らされ、値が最小限度よりも小さければOSCCALが増やされ、そして再び計時器が次の40校正クロック周期間走行します。この処理はTCNT値が精度限界内になるまで継続されます。

校正コードの段階的実行は下で説明されます。

- 目的対象デバイスは初期化ルーチンで活動を開始します。このルーチンは以下の段階を行います。
  - プログラムの変数のためにレジスタを割り当ててその変数を初期化します。
  - 指定された精度に対して周波数限度と計時器計数限度を計算します。
  - 段階量を最大OSCCAL値の半分で始めます。
  - クロック元として内部RC発振器を選びます(この設定はリセットの既定です)。
  - システムクロック前置分周器を1分周(分周なし)に構成設定します(既定構成設定は8分周です)。
  - TPIDATAピンを出力として、TPICLKピンを内部プルアップ付きの入力として構成設定します。
  - TPICLKピンでのピン変化割り込みを許可します。
  - 初期段階値(127)をOSCCAL値に設定します。
  - 全体割り込み許可(I)ビットを設定(1)します。
- 初期化後に主コードはCALIB\_STATUS変数が0かどうかを調べます。そうならばそれが非0になるまで漠然と待ちます。この変数が非0になった場合に校正処理を停止し、OSCCAL値をビット単位で送り始めます(これは直に説明されます)。
- TPICLKピンが32kHz信号を受け取り始めると、それは初期化中に許可されたピン変化割り込みを起動します。ピン変化割り込みが起動される時に必ず、目的対象はそれが下降端かどうかを調べます。それは上昇端を無視します。
- 下降端が検出されたなら、CYCLE\_COUNTER変数を増加し、それが最大値(40)に達したかどうかを調べます。そうでなければ、計時器へのクロックが許可され、CPUはmainへ戻されます。

5. CYCLE\_COUNTERが限度に達した場合、TCNT値はそれが精度限界内かどうか調べます。

- TCNTが最大限度よりも大きければOSCCAL値は段階量の半分によって減らされます
- TCNTが最小限度よりも小さければOSCCAL値は段階量の半分によって増やされます。
- TCNTが限度内なら、校正が完了されたことをTPI書き込み器に指示するためにTPIDATAピンがLowに引かれ、割り込みが禁止されます。一旦校正が終わると、CPUはTPI書き込み器へOSCCAL値を送るためのルーチンを実行します。

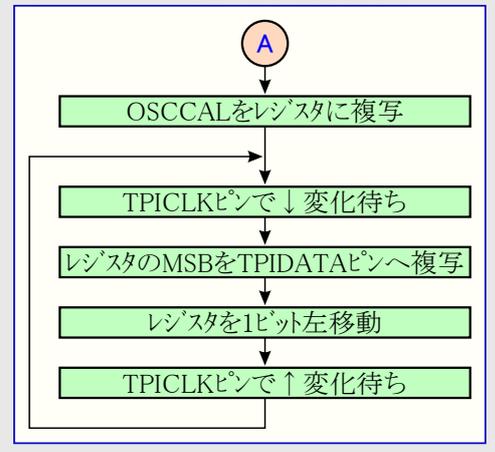
6. TCNTが限度内でなければ、TCNTレジスタが解除され、CYCLE\_COUNTERが解除されて段階4が続きます。

TPI書き込み器は常にTPIDATAピンのLowレベルを監視します。TPIDATAピンがLowに引かれたなら、目的対象での校正が完了されたと理解してTPICLKピンへの校正クロック提供を停止します。今や、OSCCAL値は目的対象から読み戻されなければなりません。目的対象(ATtiny4/5/9/10)はどの通信インターフェースも持たないため、目的対象のTPICLKピンに接続されたTPI書き込み器のI/OピンはTPIDATA線に於いてビット単位でOSCCAL値を出力するために(High/Lowに)切り替えられます。TPIDATAピンからのビットはTPI書き込み器でOSCCAL値の形式で累積されます。図2-2はこの処理の作業の流れを描きます。

校正の間、TPI書き込み器は目的対象デバイスのI/Oピンで信号を制御する通常のI/Oピン制御器として働きます。換言すると目的対象はプログラミング動作下ではなく、リセット線はTPI書き込み器の制御下ではありません。一旦校正が成功でTPI書き込み器がOSCCAL値を持つと、リセット線を(Low)に保持してプログラミング動作へ移行します。

校正が失敗の(AVROSP時間超過の前にTPIDATA線がLowに引かれられない)場合、AVROSP時間超過が起きて命令ウィンドウで異常メッセージが飛び出ででしょう。

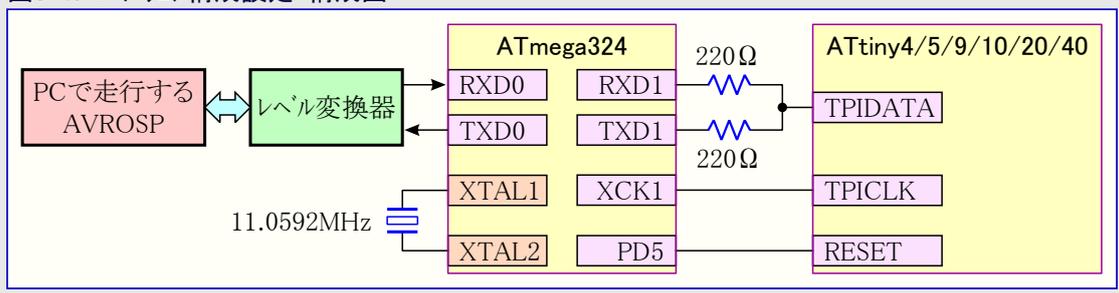
図2-2. OSCCAL読み戻し流れ図



### 3. ハードウェア構成設定

この応用記述はTPI書き込み器としてAtmel AVR918、ソフトウェア前処理部としてAtmel AVR911を用いて目的対象デバイスのプログラミングだけでなく目的対象内部RC発振器の校正も行います。図3-1はハードウェア構成設定の構成図を示します。

図3-1. ハードウェア構成設定-構成図



AVR918例コードはそれが2つのUSART単位部を持つためにTPI書き込み器としてAtmel ATmega324を使います。ATmega324は禁止されたCKDIV8ヒューズでのクロック元として働く外部11.0592MHzクリスタルと共に接続されます。USART0単位部は(正しいレベル変更後に)コンピュータのRS232ポートに接続されて非同期動作され、コンピュータで走行するAVROSPとの通信に使われます。USART1単位部はTPI通信と校正に使われ、非同期で動作されます。

目的対象プログラミング用の命令はAVR918応用記述での説明と同じです。校正については'-Y'命令が使われるべきです。AVROSPを通して'-Y'命令がTPI書き込み器に送られると、TPI書き込み器(ATmega324)は以下の段階を行います。

1. USART1に対応するPORTとDDRのレジスタの現在の値を保存し、USART1単位部を禁止します。
2. TXD1とRXD1のピンを内部プルアップ付きの入力として構成設定します。
3. 校正クロックを送るためにXCK1ピンを出力として構成設定します。
4. XCK1ピン(OC1Bピン、計時器1は比較一致で交互切り替えるOC1BでのCTC動作形態で動作)で32kHz信号を出力するために計時器1を開始します。
5. RXD1がLowに引かれるまで待ちます。
6. 一旦RXD1がLowに引かれると、計時器が停止され、XCK1ピンでの信号は目的対象にOSCCAL値を送らせるために交互切り替えられます。毎回の交互切り替えに対して書き込み器はRXD1ピンで見える値を累積します。これは8回の間起きます(1バイトのOSCCAL値)。
7. その後にUSART1単位部を許可し、PORTとDDRの値を回復してAVROSPにOSCCAL値を送り返します。

-Y命令はOSCCAL値が書かれるフラッシュメモリのバイトアドレスと共に提供されるべきです。また命令行に於いて-Y命令は最終応用HEXコードの書き込み命令に伴われるべきです。例えば-Y命令は以下のように使われるべきです。

```
avrosp.exe -dattiny40 -s -Y5FF -e -pf -vf -ifuser.hex
```

上の命令が実行されると、校正後にAVROSPはuser.hexファイルを読み、そのHEXファイルで目的対象デバイスを書きます。HEXファイル書き込み中にAVROSPはフラッシュメモリの指定アドレス(バイトアドレス\$05FF)に最終OSCCAL値も書きます。指定バイトアドレスがユーザーHEXファイル応用コード範囲内の場合、異常メッセージが投げられます。

## 4. 即時開始の手引き

1. AVR057.zipファイルをダウンロードして解凍してください。
2. AVR918\_HEX\_FILESフォルダを開いてください。このフォルダはTPI書き込み器のHEXファイルを含みます。このフォルダ内には利用可能な3つのHEXファイルがあります。Atmel ATtiny4/5/9/10デバイス用の1つ、Atmel ATtiny20デバイス用の1つ、そしてAtmel ATtiny40デバイス用の3つ目のものです。必要なHEXファイルを選んでそのHEXファイルでAtmel ATmega324を書いてください。
3. ATmega324のXTAL1とXTAL2のピン間に11.0592MHzのクリスタルを接続し、クロック元として外部クリスタル用発振器が選択されてCKDIV8ヒューズが禁止されるようなヒューズ設定に変更してください。
4. 図3-1の構成図で示されるように目的対象デバイスだけでなくコンピュータにTPI書き込み器も接続してください。
5. 一旦接続が終わったなら、コマンドプロンプトを開いてavrosp.exeファイルが存在するAVR057フォルダ内のDebugフォルダに誘導してください。それで以下の命令を実行してください。

```
mode com1 baud=115200 parity=n data=8
avrosp.exe -dattiny40 -Y0
```

最初の命令はコンピュータのCOMポートを構成設定し、2つ目の命令が校正を開始します。校正クロックは今やTPI書き込み器(ATmega324)のPD4(XCK1)ピンで利用可能です。オシロスコープを使い、このピンで(概ね32kHzであるべき)クロック信号の周波数を測定してください。

**注:** コマンドプロンプトでAVROSP時間超過異常が投げられます。この異常は目的対象に校正ファームウェアが設定されていないために起きます。この異常を無視してTPI書き込み器をリセットしてください。

6. さてzipファイルから解凍されたAVR057フォルダを開き、フォルダ内のAtmel Studio 6解決策ファイルを開いてください。
7. Solution explorerからAVR057.asmファイルを開き、コードの最初の行でデバイス名を指定してください。例えばデバイスがATtiny40なら、".equ ATTINY = 40"と記載してください。
8. ".equ CALIB\_CLOCK\_FREQ = 31800"行に於いて段階5.で測定した周波数で校正クロック周波数を指定してください。
9. Project⇒AVR057 Properties⇒Device⇒Change Deviceを使ってAtmel Studio 6プロジェクトでのデバイスを変更してください。
10. プロジェクトを保存してBuild⇒Rebuild solutionを使ってこのプロジェクトを再構築してください。Debugフォルダは利便性のためにavrosp.exeファイルだけでなくバッチファイル(avr057.bat)も含みます。
11. ノートパッドでavr057.batファイルを開き、CPU領域でデバイス名を指定してください。またCUSTOMERCODE領域で最終応用コードHEXファイル名を指定し(現在それはuser.hexに指定)、このバッチファイルを保存してください。
12. 最終応用コードHEXファイルをバッチファイルが存在するAVR057のDebugフォルダ内に複写してください。
13. バッチファイルは最初に目的対象デバイス内に校正ファームウェアを書きます。一旦書き込みが成功すると、目的対象デバイスは電源OFF/ONが必要です。図4-1.で示されるようにバッチファイルはこのメッセージを表示します。

図4-1. AVR057.hexファイルで目的対象デバイス書き込み後の出力

```

C:\WINDOWS\system32\cmd.exe
*****
Batch file for calibration of Atmel AVR TINY4/5/9/10/20/40
oscillator through the TPI interface with AVR918
- The internal RC is calibrated to value and accuracy
  given in AVR057.asm
- Programming calibration firmware is performed initially.
ECHO is off.
$Name$
$Revision: 3900 $
$RCSfile$
$Date: 2008-04-30 14:28:26 +0200 (on, 30 apr 2008) $
*****
-----
** S T A R T   P R O G R A M M I N G **
-----

Status for device COM1:
-----
Baud:          115200
Parity:        None
Data Bits:     8
Stop Bits:     1
Timeout:       OFF
XON/XOFF:      OFF
CTS handshaking: OFF
DSR handshaking: OFF
DSR sensitivity: OFF
DTR circuit:   ON
RTS circuit:   ON

AVR Open-source Programmer $Revision: 1163 $ (C) 2004 Atmel Corp.
Serial port timeout set to 5 sec.
Scanning COM ports for supported programmer...
COM1...
Found AVR ISP on COM1!
Entering programming mode...
Reading signature bytes: 0x1e 0x92 0x0e
Parsing XML file for device parameters...
Parsing '.\attiny40.xml'...
#####
Saving cached XML parameters...
Signature matches device!
Erasing chip contents...
Reading HEX input file for flash operations...
#####
Programming Flash contents...

Reading Flash contents...

Comparing Flash data...
Equal!
Leaving programming mode...
=====
Calibration Firmware successfully programmed to the target
=====
POWER CYCLE THE TARGET, RESET TPI PROGRAMMER AND PRESS ENTER!!!
Press any key to continue . . .

```

14. それで目的対象デバイスを電源OFF/ONし、TPI書き込み器をリセットしてコンピュータのキーボードでEnterキーを叩いてください。
15. 一旦これが行われると、使用者コードの書き込みが後続する校正が始まります。最終OSCCAL値が-Y命令と共に言及されるフラッシュメモリの位置に書かれます。図4-2は校正と使用者コードの書き込みが成功だった時の命令出力を示します。

図4-2. 表示されたOSCCAL値で校正後の出力

```

C:\WINDOWS\system32\cmd.exe
=====
Calibration Firmware successfully programmed to the target
=====

POWER CYCLE THE TARGET, RESET TPI PROGRAMMER AND PRESS ENTER!!!

Press any key to continue . . .

** Start Calibration followed by programming customer code
-----
AVR Open-source Programmer $Revision: 1163 $ (C) 2004 Atmel Corp.

Serial port timeout set to 5 sec.
Scanning COM ports for supported programmer...
COM1...
Found AVR ISP on COM1!
Sending Calibration Clock...
Calibration Done!
OSCCAL Value = 114
Entering programming mode...
Parsing XML file for device parameters...
Parsing '.\attiny40.xml'...
#####
Saving cached XML parameters...
Signature matches device!
Erasing chip contents...
Reading HEX input file for flash operations...
#####
Programming Flash contents...
#####
Reading Flash contents...
#####
Comparing Flash data...
Equal!
Leaving programming mode...

*****
          P R O G R A M M I N G   O K
*****
Press any key to continue . . . _
    
```

### 5. 改訂履歴

文書改訂	日付	注釈
42038A	2012年10月	初版文書公開



Enabling Unlimited Possibilities®

*Atmel Corporation*

1600 Technology Drive  
San Jose, CA 95110  
USA  
TEL (+1)(408) 441-0311  
FAX (+1)(408) 487-2600  
[www.atmel.com](http://www.atmel.com)

*Atmel Asia Limited*

Unit 01-5 & 16, 19F  
BEA Tower, Millennium City 5  
418 Kwun Tong Road  
Kwun Tong, Kowloon  
HONG KONG  
TEL (+852) 2245-6100  
FAX (+852) 2722-1369

*Atmel Munich GmbH*

Business Campus  
Parking 4  
D-85748 Garching b. Munich  
GERMANY  
TEL (+49) 89-31970-0  
FAX (+49) 89-3194621

*Atmel Japan G.K.*

141-0032 東京都品川区  
大崎1-6-4  
新大崎勸業ビル 16F  
アトメル ジャパン合同会社  
TEL (+81)(3)-6417-0300  
FAX (+81)(3)-6417-0370

© 2012 Atmel Corporation. 不許複製 / 改訂:42038A-AVR-10/2012

Atmel®、ロゴとそれらの組み合わせ、Enabling Unlimited Possibilities®、AVR®とその他はAtmel Corporationの登録商標または商標またはその付随物です。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。たとえば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。

© HERO 2021.

本応用記述はAtmelのAVR057応用記述(Rev.42038A-10/12)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。