
AVR®マイクロ コントローラでの実時間クロックの校正と補償

序説

著者: Eivind Berntsen, Per Andreas Gulbrandsen, Microchip Technology Inc.

組み込み時間管理応用は内部発振器と比べてそれらのより高い精度のためにクロック元として度々32.768kHz外部クリスタル用発振器を使います。けれども、精度はいくつかの要素によって悪い影響を及ぼされます。この資料はPCB設計での温度に渡る経時変化と変動に目を向けます。加えて、クリスタル製造許容誤差が簡単に説明されます。

高い精度の時間管理に対して発振器の精度を改善することができます。これは発振器の誤差を測定してそれを補償することによって行われます。周波数測定は外部参照基準とtinyAVR 1系とmegaAVR 0系の内部周辺機能を使うことによって行うことができ、または外部部品を使ってそれを行うことができます。両方の手法がこの資料で概説されません。

誤差が既知の時はその誤差を減らすようにソフトウェアに基づく補償を実装することができます。ソフトウェアに基づく補償の効率的な実装のためのいくつかの有用な技術と共に少しの例が示されます。

要点

紹介と測定:

- ・ クリスタル周波数不正確さの貢献要因に対する簡単な紹介
- ・ tinyAVR 1系とmegaAVR 0系での32.768kHzクリスタル クロックの周波数測定方法の説明

例:

- ・ RTCが静的なクリスタル誤差に対してどう補償することができるかを示す例
- ・ 静的なクリスタル誤差と温度変動のための誤差を補償するのにRTC、内部温度感知器、ADCがどう使われ得るかを示す例
- ・ 周波数測定での2つの例と補償算法を示す2つの例の、Atmel | STARTで利用可能な4つの例

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

目次

序説	1
要点	1
1. 関連デバイス	3
1.1. tinyAVR® 1系統	3
1.2. megaAVR® 0系統	3
2. 不正確さと変動の元	4
2.1. 容量性負荷不整合	4
2.2. クリスタル製造許容誤差	4
2.3. 32.768kHzクリスタル発振器温度変動	4
2.4. 32.768kHzクリスタル発振器経時変動	5
3. 周波数測定	5
3.1. 始動時間	5
3.2. 外部的な周波数測定	5
3.3. 内部クロック測定	7
4. RTC補償	7
4.1. クリスタルと負荷容量の偏差に対する補償	7
4.2. 温度変動に対する補償	7
4.3. クリスタル補償参照表例	8
5. Atmel STARTからのソースコード取得	8
6. 改訂履歴	9
Microchipウェブ サイト	10
お客様への変更通知サービス	10
お客様支援	10
Microchipデバイス コード保護機能	10
法的通知	10
商標	11
DNVによって認証された品質管理システム	11
世界的な販売とサービス	12

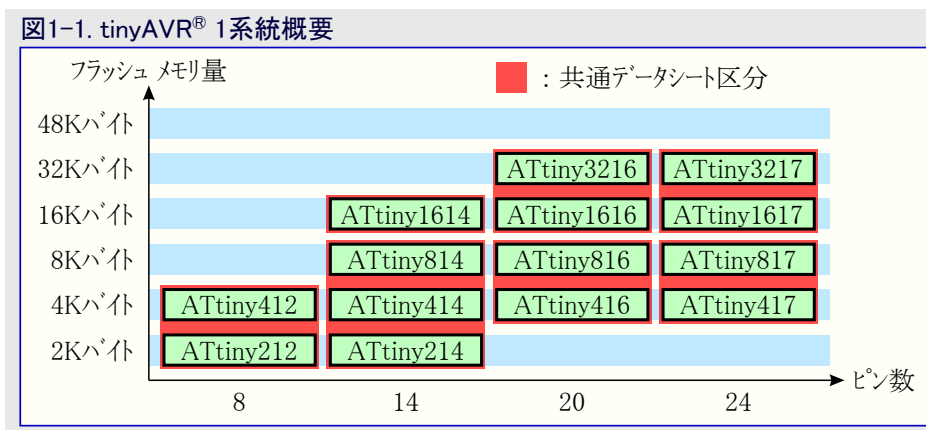
1. 関連デバイス

本章はこの資料に関連するデバイスを一覧にします。

1.1. tinyAVR[®] 1系統

下図はピン配置変種とメモリ量を展開してtinyAVR[®] 1系統デバイスを示します。

- これらのデバイスがピン互換で同じまたはより多くの機能を提供するため、垂直上方向移植はコード変更なしに可能です。下方向移植はより少ない利用可能ないくつかの周辺機能の実体のためにコード変更が必要かもしれません。
- 左への水平方向移植はピン数、従って利用可能な機能を減らします。

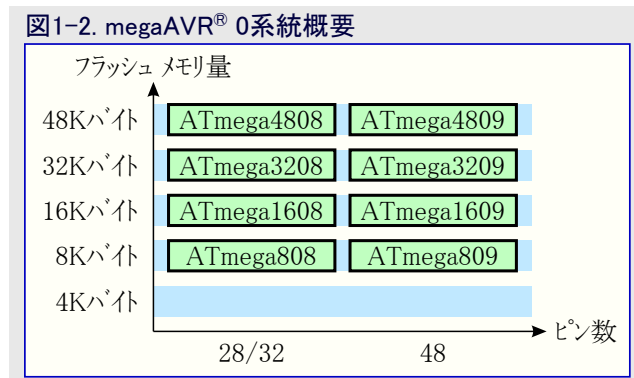


異なるフラッシュメモリ量を持つデバイスは一般的に異なるSRAMとEEPROMの量を持ちます。

1.2. megaAVR[®] 0系統

右図はピン配置変種とメモリ量を展開してmegaAVR[®] 0系統デバイスを示します。

- これらのデバイスが完全にピンと機能が互換のため、垂直方向移植はコード変更なしで可能です。
 - 左への水平方向移植はピン数、従って利用可能な機能を減らします。
- 異なるフラッシュメモリ量を持つデバイスは一般的に異なるSRAMとEEPROMの量を持ちます。



2. 不正確さと変動の元

32.768kHzクリスタル用発振器はRC発振器よりも高い精度を持ちます。けれども、クリスタル発振器は周波数での誤差も前提にしています。これはPCB設計と製造誤差のためにそれが主である静的誤差を含みます。周波数誤差は温度に渡る変動によっても影響され、動作温度の変化のために出力周波数が変わることの意味します。

2.1. 容量性負荷不整合

クリスタル製造業者はクリスタルに対して印加するための外部容量性負荷を指定します。容量性負荷の不整合はクリスタルのより低い精度や、発振失敗にさえ帰着します。クリスタル発振子のデータシートで見つけることができるように、周波数対負荷容量の代表的な特性曲線が図2-1.で示されます。

図2-2.は代表的な32.768kHzクリスタル回路用の回路図を示します。クリスタルはMCUのTOSC1とTOSC2のピンに接続されます。C1とC2は負荷容量を平衡するのに使われるコンデンサです。

図2-2. 32.768kHzクリスタル回路

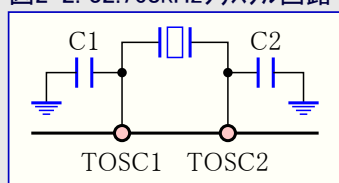
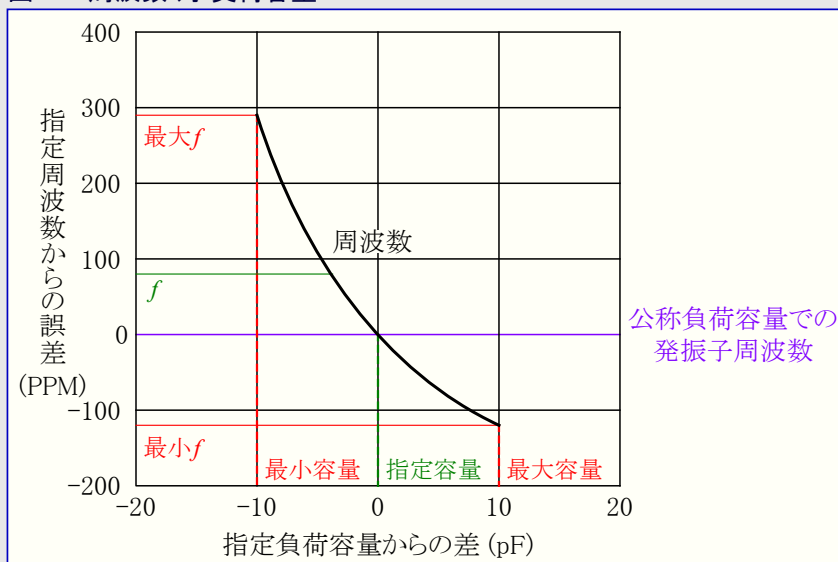


図2-1. 周波数 対 負荷容量



クリスタル回路で負荷容量に寄与する複数の供給元は以下のとおりです。

- PCB寄生容量、これはPCB布線間の容量です。
- MCU寄生容量。更なる詳細についてはデバイスのデータシートの電気的特性を参照してください。
- 実装されたC1とC2のコンデンサ、これは各々、MCUのTOSC1ピンとGND間とMCUのTOSC2ピンとGND間に配置されます。

C1とC2の値を計算する時にMCU寄生容量とPCB寄生容量が考慮されなければなりません。PCB寄生容量はPCB設計に依存します。例えば、布線の長さや幅の選択が容量に影響します。これの最善の処置のより多くの情報については「AN2648 - AVR4100:Microchip AVR®マイクロコントローラ用32kHzクリスタルの選択と試験」応用記述を参照してください。

全ての部品は製造許容誤差を前提にしています。これは正しいC1とC2の選択が製造された全ての個体に対して同じ精度を保証しないことを意味します。このため、経時管理での最適精度を達成するためにソフトウェアに基づく補償が実行されなければなりません。

2.2. クリスタル製造許容誤差

正しく設定されると、32.768kHzクリスタル発振器の精度は一般的に±20ppmの範囲です。1ppmの誤差は1年当たり30.5秒の誤差と等価です。20ppmの誤差は1年当たり概ね10分の誤差と等価です。一般的に、製造許容誤差に寄与するものはクリスタルカットの精度とクリスタルの純度です。

2.3. 32.768kHzクリスタル発振器温度変動

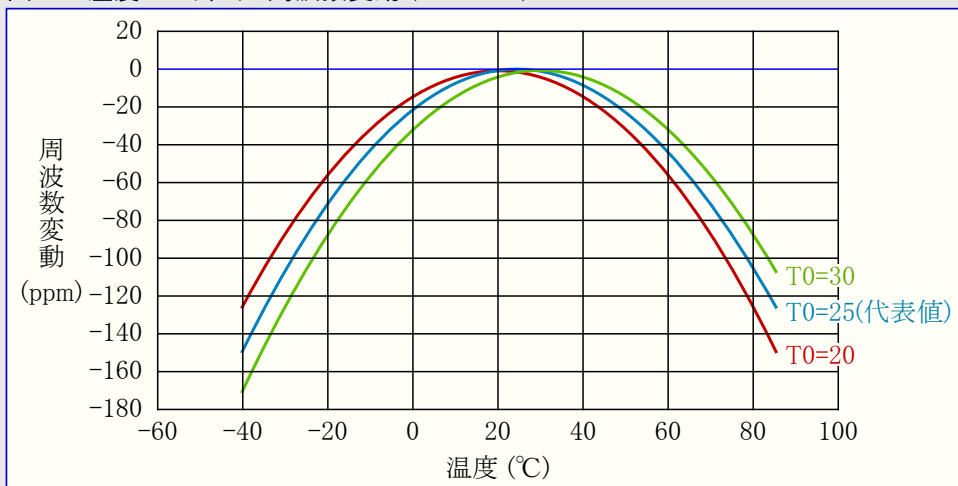
安価な32.768kHzクリスタル発振器はクリスタル製造業者が次のように指定するので、図2-3.で示されるように、一般的に温度に対して放射線状の周波数依存性を持ちます。

- ppm/°C²での温度係数(B)
- °Cでの転換温度(T0)

温度係数(B)は負で、クリスタル発振器が低温と高温で落ちることを示します。温度係数は許容誤差を持ち、代表的にこれは±15%です。転換温度(T0)は代表的に±5%の範囲の許容誤差付きで代表的に25°Cです。温度(T)での周波数変動は以下によって与えられます。

$$\Delta f/f_0(\text{ppm}) = B \times (T - T_0)^2$$

図2-3. 温度でのクリスタル周波数変動 (B=-0.035)



例として、 $B=-0.035\text{ppm}/^{\circ}\text{C}^2\pm 15\%$ 、 $T_0=25^{\circ}\text{C}\pm 5\%$ 、 $T=85^{\circ}\text{C}$ と仮定してください。これは以下の代表的な変動を引き起こします。

$$\Delta f/f_0 = 126\text{ppm}\pm 40\text{ppm}$$

2.4. 32.768kHzクリスタル発振器経時変動

殆どのクリスタル製造業者は1年動作後のクリスタルの経時変化を指定します。代表的な範囲は最初年に対して $\pm 3\text{ppm}$ です。この仕様はJクリスタルの変動が方向と大きさに於いて予測不能であることを意味します。殆どの周波数変動が最初の年に起こるとは言え、いくつかの変動は最初の年の後でも起こります。周波数変動に寄与する要因のいくつかは酸化で、クリスタルに付着する容器内の残留物です。これはクリスタル総質量、従って共振周波数を変える効果を持ちます。これのもっと詳細な説明についてはhttps://en.wikipedia.org/wiki/Crystal_oscillator#Stability_and_agingを参照してください。

負荷容量は何年にも渡って変化し得ます。そうならば、周波数変動が起きますが、この変化の大きさのためとクリスタルピン間で対称的になる変化のため、この影響は極僅かです。PCB寄生容量経時変化、更にMCU寄生容量の経時変化は実質的に存在しません。NP0/C0G型コンデンサの経時変化はコンデンサの寿命全体に対して ± 0.1 未満、即ち、10pFのコンデンサに対して10fF未満です。ppm誤差を考察すると、負荷コンデンサの変化は0.1ppm未満の変動を与えます。

PCBの寄生容量が一般的にあまり経時変化しないとは言え、耐震性や耐湿性を付加するためにPCBにシリコンやウレタンを付け足すことは、それらの材料が空気と比べて異なる誘電率を持つため、PCB布線の容量を僅かに変えるかもしれません。

3. 周波数測定

それを補償することができるように周波数誤差の大きさを知ることが必要です。従って、クリスタルの周波数が測定されなければなりません。測定はクリスタルとデバイスがPCBに実装されている時に実行されなければなりません。

温度の範囲に渡る複数回の測定は一般的に、製造業者のデータを信頼するのと比べてより良い結果を与えるでしょう。

32.768kHzクリスタル発振器の周波数は外部測定ツールを用いて、または内部資源を使うことによって測定することができます。外部ツール使用時、そのツールは入力としてクリスタルの周波数を取ります。測定に内部資源、例えば、タイマ/カウンタを使う時に高精度の参照基準が供給されなければなりません。32.768kHzクリスタル周波数が比較される対象が参照基準なので、測定の精度は参照基準の精度よりも決して高くなりません。

両方の場合に於いて、周波数が測定されるのに先立ってクリスタルが始動と安定のための時間を与えられることが重要です。

3.1. 始動時間

全てのクリスタルはクリスタル出力に対して安定になるのにかかる時間である始動時間があります。与えられたクリスタルの始動時間はそのクリスタルのデータシートで指定されます。tinyAVR 1系とmegaAVR 0系では65535となる最大値を持つ始動時間設定に関して複数の選択があります。これはデバイスが2秒の始動時間に換算される32.768kHzの周波数での65536周期を計数することを意味します。始動時間が経過されると、主クロック状態レジスタの32.768kHzクリスタル用発振器状態(MCLKSTATUS.XOSC32KS)が設定(1)されます。

注: 32.768kHzクリスタルは32.768kHzクリスタル用発振器制御レジスタの許可(XOSC32CTRLA.ENABLE)ビットが'1'を書かれる時に自動的に開始されません。それは32.768kHzクロックが周辺機能、例えば、主クロックまたはRTCによって要求された時、またはXOSC32CTRLAのスタンバイ時走行(XOSC32CTRLA.RUNSTDBY)ビットが設定(1)される場合にだけ開始されます。

注: MCLKSTATUS.XOSC32KSは32.768kHzクロックが周辺機能によって要求されない限り設定(1)されません。これはXOSC32CTRLA.RUNSTDBYビットが設定(1)される時にも真(1)です。

3.2. 外部的な周波数測定

周波数測定を行うのに外部測定ツールを使う時に、32.768kHzクリスタル発振器出力はピンで利用可能でなければなりません。これはRTCまたはPITからの事象を使ってピンを使用部として構成するか、または主クロックとして32.768kHzクリスタルを使うことによるかのどちらかを行うことができます。

許可されると、RTCは以下の出力事象を生成します。

- ・ 溢れ(OVF) : 計数器が頂上値に達して0に丸められる時に生成されます。生成された瞬発(ストロブ)はCLK_RTCと同期し、1 CLK_RTC周期留まります。
- ・ 比較(CMP) : 計数器値と比較レジスタ間の一致を示します。生成された瞬発(ストロブ)はCLK_RTCと同期し、1 CLK_RTC周期留まります。

許可されると、PITはそれの事象出力で以下の50%デューティサイクルのクロック信号を生成します。

- ・ 事象0 : クロック間隔 = 8192 RTCクロック周期
- ・ 事象1 : クロック間隔 = 4096 RTCクロック周期
- ・ 事象2 : クロック間隔 = 2048 RTCクロック周期
- ・ 事象3 : クロック間隔 = 1024 RTCクロック周期
- ・ 事象4 : クロック間隔 = 512 RTCクロック周期
- ・ 事象5 : クロック間隔 = 256 RTCクロック周期
- ・ 事象6 : クロック間隔 = 128 RTCクロック周期
- ・ 事象7 : クロック間隔 = 64 RTCクロック周期

事象使用部は事象システム(EVSY)によって構成設定されます。

主クロック元として32.768kHzクリスタルを使う時は以下のようにピンのクロック出力配線するいくつかの方法があります。

- CLKOUTピンへ直接的にクロックを配線
- 主クロック周波数を出力するのにTCA、TCB、TCD、またはSPIを使用

周波数を測定するのにクリスタル回路にプローブを直接的に付着することは推奨されません。これは回路の特性を変え、クリスタル周波数を変えます。

以下の2つの例は32.768kHzクリスタルをどうピンに出力するかを示します。1つ例はシステムクロックを使う方法を示し、もう1つはRTCと事象システムを使う方法を示します。

3.2.1. システムクロックとしてクリスタルクロックを使用

この例は32.768kHzクリスタルをシステムクロックになるようにどう構成設定し、システムクロックをCLKOUTピンに出力としてどう設定するかを示します。プロジェクトへのリンクについては「[5. Atmel | STARTからのソースコード取得](#)」を参照してください。

32.768kHzクロックがCPUをクロック駆動するには以下が推奨されます。

1. `main()`で実行される活動の1つとしてスタンバイ時実行でクリスタルを許可してください。
2. もしあれば、他の製造検査と回路の校正を走らせてください。
3. 主クロックをクリスタルに切り替えてください。
4. システムクロックをCLKOUTピンでの出力として構成設定してください。

この手順は他の製造検査が実行されつつある間にクリスタルに対する安定にするための時間を許します。

下のコード例では最初の保護された書き込みによってクリスタルが開始されます。同時にスタンバイ時実行(`RUNSTDBY`)ビットが書かれることに注意してください。これは例えば周辺機能やCPUがクロックを要求しない時でもクリスタルが開始することを保証するためです。何かがクロックを要求しない限り、`XOSC32KS`ビットが設定(1)されないため、クリスタルが安定なことを調べるのに先立って主クロックが切り替えられます。

```
// レジスタに対する変更保護を無効にし、ここでXOSC32K開始、始動時間は最大の64K(2秒)に設定されます。
_PROTECTED_WRITE(CLKCTRL_XOSC32CTRLA, CLKCTRL_ENABLE_bm |
    CLKCTRL_CSUT_64K_gc | CLKCTRL_RUNSTDBY_bm);

// レジスタに対する変更保護を無効にし、主クロックを32K発振器に切り替え、クロックをピンに出力
_PROTECTED_WRITE(CLKCTRL_MCLKCTRLA, CLKCTRL_MCLKCTRLA =
    (CLKCTRL_CKSEL1_bm) | CLKCTRL_CLKOUT_bm);

// 安定として報告されるXOSC32K待機
while(!(CLKCTRL_MCLKSTATUS & CLKCTRL_XOSC32KS_bm));

// レジスタに対する変更保護を無効にし、前置分周器が禁止されるのを確実にします。
_PROTECTED_WRITE(CLKCTRL_MCLKCTRLB, 0);
```

32.768kHzクロックを出力する場合、出力ピンの容量性負荷を考慮することが重要です。高い負荷は検査ピンで良好な信号出力を得ることを難しくします。また、クロック信号は検査の時間でピンに接続された他の部品に逆に影響してはなりません。ピンがこの信号を扱うのに利用可能なことを保証するために、製造検査は設計工程で初期に計画されるべきです。

3.2.2. RTCと事象システムを使用

この例では事象システムと共にRTCの周期的割り込み計時器(PIT:Periodic Interrupt Timer)が使われます。この場合、システムクロックは低速発振器に切り替えられてはなりません。プロジェクトへのリンクについては「[5. Atmel | STARTからのソースコード取得](#)」を参照してください。

PITで生成された周期的事象は事象システムを使って出力することができます。上で見られるように、利用可能な多くの異なる間隔があります。この例では'`EVSY_ASYNCCH3 = EVSY_ASYNCCH3_PIT_DIV64_gc;`'行で見ることができるよう、最低間隔PIT事象出力が使われます。RTCクロック周波数の分周を64への設定で、これは1.95msの間隔で選んだ出力ピンでのパルスを生成します。

可能な限り早く32.768kHzクリスタルを許可するための保護された書き込みを実行してください。より良い時間利用のため、クリスタルが安定になる間に他の製造検査を実行することが推奨されます。下のコードは事象生成部としてPITを構成設定する方法とピンで信号を出力する方法を示します。

```
// レジスタに対する変更保護を無効にし、ここでXOSC32K開始、始動時間は最大の64K(2秒)に設定されます。
_PROTECTED_WRITE(CLKCTRL_XOSC32CTRLA, CLKCTRL_ENABLE_bm |
    CLKCTRL_CSUT_64K_gc | CLKCTRL_RUNSTDBY_bm);

// PIT事象用経路として事象システムのチャンネル3を使用
EVSY_ASYNCCH3 = EVSY_ASYNCCH3_PIT_DIV64_gc;
// チャンネル3を非同期使用部8(EVOUT0)に接続
EVSY_ASYNCUSER8 = EVSY_ASYNCUSER8_ASYNCCH3_gc;
```

```

VOOUT0多重器出力をPA2へ
PORTMUX_CTRLA = PORTMUX_EVOUT0_bm;

// RTCでPITを許可
RTC_PITCTRLA = RTC_PITEN_bm;
// RTC用クロック元としてクリスタルを設定
RTC_CLKSEL = RTC_CLKSEL_TOSC32K_gc;
// RTC許可
RTC_CTRLA = RTC_RTCEN_bm;

```

3.3. 内部クロック測定

内部周波数測定は製造だけでなく何時でも行うことができます。内部測定を行うのに参照基準クロックが必要とされます。高くまたは低い周波数のクロックを、またはGPS単位部からの1秒の刻時のような刻時を使うことができます。

2つの方法の内、高い周波数参照基準を使う方は最短時間で最良の結果を与えます。

3.3.1. 高い周波数入力での校正

考えはクリスタル発振器から走行するRTCによって与えられる設定期間内で高速参照基準周波数を計数するのにTCDを使うことです。TCDの溢れ数と最終捕獲値を予期される値と比較することができます。プロジェクトへのリンクについては「[5. Atmel | STARTからのソースコード取得](#)」を参照してください。

この方法は以下の段階から成ります。

1. 外部の参照基準周波数をデバイスのEXTCLKピンに印加してください。
2. クロック元としてこれを使うようにTCDを構成設定ください。
3. XOSC32Kクロックを使うようにRTCを構成設定ください。
4. TCDからの捕獲値を得るためにRTCからの溢れ信号を送るように事象システムを構成設定してください。

TCDは最初の捕獲後に0からの計数を開始し、従って最初の捕獲値は破棄されるべきです。RTC期間内でTCD溢れ数を数えてください。RTC期間が終わると、2つ目の捕獲値が格納され、測定された値としてこれとTCD溢れ数を使ってください。理想予測値と測定値間の違いがクリスタルの周波数誤差を示します。

例に於いて、測定されるRTC期間はクリスタルが正確ならば1秒に対応するように設定されます。この例で使われる外部参照基準は20MHzの周波数を持ちます。クリスタルが正確に32.768kHzで走行し、期間が1秒に設定される場合、TCDは20,000,000計数に達するべきです。TCDが12ビット幅だけのため、この計数値に達することができません。これは4882回の溢れ、それと捕獲値3328になるべきです。

さて、最終捕獲値がこの理想値に達しないが、その代わりに溢れ数が未だ4882と同時に2296に達したと仮定してください。TCDはあるべきそれよりも3328-2296=1032少ない計数された周期を持ちます。クリスタルはあるべきそれよりも速く動いていて、RTC期間を予想よりも短くさせます。秒での時間の違いは $1032/20,000,000=51.6\mu\text{s/s}$ です。

4. RTC補償

「[2.3. 32.768kHzクリスタル発振器温度変動](#)」で見られるように、代表的な32.768kHzクリスタルは25°C周辺で対称的に変動します。RTCに補償算法を適用することにより、変動に対して調整することが可能です。

4.1. クリスタルと負荷容量の偏差に対する補償

上の例に於いて、クリスタルは $51.6\mu\text{s/s}$ 速すぎるのが見つけられました。1秒当たりRTCで2つの余分な周期を数えるなら、RTCは $2 \times (1/32768)=60\mu\text{s/s}$ より遅くなります。誤差は今や $60-51.6=8.6\mu\text{s/s}$ 遅すぎになるように減らされます。各秒で累積された誤差の経緯を保つのにソフトウェア変数を使うことができます。累積された誤差が32.768kHzクリスタルクロックの1周期を溢れると、RTC周期は1秒に対して1周期減らされます。

Atmel | STARTで見つかるクリスタル補償例では補償変数の精度が $0.1\mu\text{s/s}$ です。誤差として小数点以下の桁数を増やすことが可能です。温度補償例では補償の精度が $0.01\mu\text{s/s}$ に増えています。プロジェクトへのリンクについては「[5. Atmel | STARTからのソースコード取得](#)」を参照してください。

4.2. 温度変動に対する補償

クリスタルの容量性負荷を制御することができるようにしてクリスタル発振器回路へ誘導される雑音を減らすため、32.768kHzクリスタルはデバイスの近くに配置されるべきです。これはデバイスとクリスタルが同じ温度に晒されることも意味します。

tinyAVR 1系とmegaAVR 0系内のデバイスは内部温度感知器を持ちます。対応するデバイスのデータシートの特性章から、温度感知器が相対的に貧弱な精度を持つことを見ることができます。例えばデバイスの識票列に置かれて供給された校正値を使う時でも、シリコンが未だウェハー上の時にこれらの値が見つけられるため、これは真実です。切断して外装することがシリコンでの影響を持ち、従って温度読み出しを変えます。校正値はデバイスがPCBに半田付けされた時に更に偏るかもしれません。

製造で新しい利得と変位(オフセット)の値が測定されるなら、内部温度感知器でより良い結果を達成することが可能です。これは校正手順を必要とし、デバイスがPCBに半田付けされてしまった後に行われるのが好ましいです。広い温度範囲に渡って良好な性能が必要とされるなら、2または3点校正が実行されるべきです。検査点の2つは精度が必要とされる温度範囲の僅かに上と下で、1つは中央近くであるべきです。クリスタル周波数がこれら3点で測定されるなら、クリスタルのデータシートの特性で与えられるものよりもっと正確な放物線状温度曲線を得ることが可能です。

注: 高精度温度感知器を持つことが最適とは言え、温度補償を行うのに低精度温度感知器を使うことは未だ有用です。これは不正確な読み取りに基づいて選ばれた補償値が25°C用に与えられた値よりもより正確なため、特に動作温度が25°Cから遠い場合に真実です。

4.3. クリスタル補償参照表例

温度変化に対処するため、継続的に温度を測定してRTC割り込み処理ルーチン(ISR)でこれに対する補償を適用することが可能です。これはクロックをもっと正確に保つのを助けます。不経済な浮動や固定の小数点演算を避けるため、クリスタル温度補償値を参照表に格納することができます。クリスタル製造業者は一般的に温度の関数として変動を大きさを記述するための図や式を与えます。製造業者の仕様に比べてより高い精度が望まれる場合、各PCBに対して違う温度で3点測定を行うことができます。この方法は各クリスタルと回路に対して正しい曲線を見つけることが可能です。同時に内部温度感知器の測定が行われるなら、データ点はクリスタル変動と測定温度間でよく整列されるべきです。

Atmel | STARTで見つかる温度補償例では放物線状曲線が以下の算術関数に由来します。

$$\Delta f/f_0 = -0.04 \times (T-25)^2$$

RTC ISRで使われる補償値は上の関数で生成された表から直接取られます。RTCが毎秒溢れて表からの誤差値は各秒で総累積誤差の経緯を保つ変数に加算されます。この累積された誤差変数が32.768kHzの1周期よりも大きな時にRTC期間から1以上の周期を減算します。累積した誤差が1周期よりも小さい時はRTC期間値が通常の1秒期間値に設定されます。

式が放射線で表の値が頂上点周辺で対称なため、25~105°Cまでの値だけを格納し、25~-40°C間の温度に対して25~90°Cまでの値の部分を使うことが可能です。これは必要とされるフラッシュメモリの大きさを減らします。

例では殆どの計算がRTCとADCのISRで行われ、実行されるコードはそんなに時間を浪費しません。

RTC ISRでは累積された誤差が計算されます。これは未だ補償されていない累積された誤差、秒毎の静的なクリスタル誤差、秒毎の温度補償誤差から成ります。温度補償はADC ISRで参照表を用いて見つかります。ADC変換はRTC ISRから開始されます。

注: RTC定期(PER)レジスタはRTC ISRで書かれます。書かれる値は32.768kHz(クロック)領域でレジスタに同期されなければなりません。以前に同じレジスタへ書かれた値が同期されてしまっている限り、CPUはこれを行うのに待つ必要はありません。このコードでこれは問題ではありません。このレジスタがISRの外側で書かれた場合、またはPER値が非常に低かった(1~3)場合、ISRが立て続けに実行されるため、問題になり得ます。各レジスタに対してRTCが独立した同期機構を持つことに注意してください。RTCで立て続けの2つの違うレジスタへの書き込みは問題ではありません。

ADC ISRではADCの結果が読み出され、利得と変位(オフセット)の補償が実行され、結果はケルビン(K)に変換されます。その後に結果は温度補償参照表から使うための補償値を見つけるのに使われます。

例のコードはデバイスの内部温度感知器を使います。もっと正確な外部温度感知器を使うことも可能です。これを行う場合、クロック精度が増されるべきです。

5. Atmel | STARTからのソースコード取得

コード例は画像ユーザーインターフェース(GUI)を通して応用コードの形態設定を許すウェブに基づくAtmel | STARTを通して利用可能です。コードは下の直接コード例リンクまたはAtmel | START先頭頁のBROWSE EXAMPLES(例検索)鉤経由Atmel StudioとIAR Embedded Workbench®の両方に対してダウンロードすることができます。

Atmel | STARTウェブ ページ: [ATMEL START](#)

コード例

- 周波数出力 (Frequency Output)
 - http://start.atmel.com/#example/Atmel:rtc_calibration_and_compensation:1.0.0::Application:RTC_Crystal_Frequency_Output:
- 内部周波数校正 (Internal Frequency Calibration)
 - http://start.atmel.com/#example/Atmel:rtc_calibration_and_compensation:1.0.0::Application:RTC_Internal_Frequency_Calibration:
- クリスタル誤差補償 (Crystal Error Compensation)
 - http://start.atmel.com/#example/Atmel:rtc_calibration_and_compensation:1.0.0::Application:RTC_Crystal_Error_Compensation:
- クリスタル補償参照表 (Crystal Compensation Look-Up Table)
 - http://start.atmel.com/#example/Atmel:rtc_calibration_and_compensation:1.0.0::Application:RTC_Crystal_Compensation_Look-Up_Table:

例プロジェクトについての詳細と情報に関してはAtmel | STARTでUser guide(使用者の手引き)を押下してください。User guide鉤はAtmel | STARTプロジェクト形態設定部内の一覧画面でプロジェクト名をクリックすることにより、例閲覧部で見つけることができます。

Atmel Studio

DOWNLOAD SELECTED EXAMPLE(選んだ例をダウンロード)をクリックすることにより、Atmel | STARTで例閲覧部からAtmel Studio用.**.atzip**ファイルとしてコードをダウンロードしてください。Atmel | START内からファイルをダウンロードするには、**EXPORT PROJECT**(プロジェクトをエクスポート)に続いて**DOWNLOAD PACK**(一括ダウンロード)をクリックしてください。

ダウンロードした**.atzip**ファイルをダブルクリックしてください。プロジェクトがAtmel Studio 7.0に導入されます。

IAR Embedded Workbench

IAR Embedded Workbenchでプロジェクトをインポートする方法の情報についてはAtmel | START使用者の手引きを開き、**Using Atmel Start Output in External Tools**(外部ツールでAtmel START出力を使用)と**IAR Embedded Workbench**を選んでください。Atmel | START使用者の手引きへのリンクは共に頁の右上隅に置かれたAtmel | START先頭頁から**About**(これについて)またはプロジェクト形態設定部内の**Help And Support**(手助けと支援)をクリックすることによって見つけることができます。

6. 改訂履歴

資料改訂	日付	注釈
A	2018年7月	初版資料公開
B	2018年10月	「 関連デバイス 」章を8/16KバイトmegaAVR 0系デバイスを含めるように更新

Microchipウェブ サイト

Microchipは<http://www.microchip.com/>で当社のウェブ サイト経由でのオンライン支援を提供します。このウェブ サイトはお客様がファイルや情報を容易に利用可能にする手段として使用されます。お気に入りのインターネット ブラウザを用いてアクセスすることができ、ウェブ サイトは以下の情報を含みます。

- **製品支援** – データシートと障害情報、応用記述と試供プログラム、設計資源、使用者の手引きとハードウェア支援資料、最新ソフトウェア配布と保管されたソフトウェア
- **一般的な技術支援** – 良くある質問(FAQ)、技術支援要求、オンライン検討グループ、Microchip相談役プログラム員一覧
- **Microchipの事業** – 製品選択器と注文の手引き、最新Microchip報道発表、セミナーとイベントの一覧、Microchip営業所の一覧、代理店と代表する工場

お客様への変更通知サービス

Microchipのお客様通知サービスはMicrochip製品を最新に保つのに役立ちます。加入者は指定した製品系統や興味のある開発ツールに関連する変更、更新、改訂、障害情報がある場合に必ず電子メール通知を受け取ります。

登録するには<http://www.microchip.com/>でMicrochipのウェブ サイトをアクセスしてください。”Support”下で”Customer Change Notification”をクリックして登録指示に従ってください。

お客様支援

Microchip製品の使用者は以下のいくつかのチャネルを通して支援を受け取ることができます。

- 代理店または販売会社
- 最寄りの営業所
- 現場応用技術者(FAE:Field Application Engineer)
- 技術支援

お客様は支援に関してこれらの代理店、販売会社、または現場応用技術者(FAE)に連絡を取るべきです。最寄りの営業所もお客様の手助けに利用できます。営業所と位置の一覧はこの資料の後ろに含まれます。

技術支援は<http://www.microchip.com/support>でのウェブ サイトを通して利用できます。

Microchipデバイスコード保護機能

Microchipデバイスでの以下のコード保護機能の詳細に注意してください。

- Microchip製品はそれら特定のMicrochipデータシートに含まれる仕様に合致します。
- Microchipは意図した方法と通常条件下で使用される時に、その製品系統が今日の市場でその種類の最も安全な系統の1つであると考えます。
- コード保護機能を破るのに使用される不正でおそらく違法な方法があります。当社の知る限りこれらの方法の全てはMicrochipのデータシートに含まれた動作仕様外の方法でMicrochip製品を使用することが必要です。おそらく、それを行う人は知的財産の窃盗に関与しています。
- Microchipはそれらのコードの完全性について心配されているお客様と共に働きたいと思います。
- Microchipや他のどの半導体製造業者もそれらのコードの安全を保証することはできません。コード保護は当社が製品を”破ることができない”として保証すると言ったことを意味しません。

コード保護は常に進化しています。Microchipは当社製品のコード保護機能を継続的に改善することを約束します。Microchipのコード保護機能を破る試みはデジタル ミレニアム著作権法に違反するかもしれません。そのような行為があなたのソフトウェアや他の著作物に不正なアクセスを許す場合、その法律下の救済のために訴権を持つかもしれません。

法的通知

デバイス応用などに関してこの刊行物に含まれる情報は皆さまの便宜のためにだけ提供され、更新によって取り換えられるかもしれません。皆さまの応用が皆さまの仕様に合致するのを保証するのは皆さまの責任です。Microchipはその条件、品質、性能、商品性、目的適合性を含め、明示的にも黙示的にもその情報に関連して書面または表記された書面または黙示の如何なる表明や保証もしません。Microchipはこの情報とそれの使用から生じる全責任を否認します。生命維持や安全応用でのMicrochipデバイスの使用は完全に購入者の危険性で、購入者はそのような使用に起因する全ての損害、請求、訴訟、費用からMicrochipを擁護し、補償し、免責にすることに同意します。他に言及されない限り、Microchipのどの知的財産権下でも暗黙的または違う方法で許認可は譲渡されません。

商標

Microchipの名前とロゴ、Mcirochipロゴ、AnyRate、AVR、AVRロゴ、AVR Freaks、BitCloud、chipKIT、chipKITロゴ、CryptoMemory、CryptoRF、dsPIC、FlashFlex、flexPWR、Heldo、JukeBlox、KeeLoq、KeeLoqロゴ、Kleer、LANCheck、LINK MD、maXStylus、maXTouch、MediaLB、megaAVR、MOST、MOSTロゴ、MPLAB、OptoLyzer、PIC、picoPower、PICSTART、PIC32ロゴ、Prochip Designer、QTouch、SAM-BA、SpyNIC、SST、SSTロゴ、SuperFlash、tinyAVR、UNI/O、XMEGAは米国と他の国に於けるMicrochip Technology Incorporatedの登録商標です。

ClockWorks、The Embedded Control Solutions Company、EtherSynch、Hyper Speed Control、HyperLight Load、IntelliMOS、mTouch、Precision Edge、Quiet-Wireは米国に於けるMicrochip Technology Incorporatedの登録商標です。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、BodyCom、CodeGuard、CryptoAuthentication、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、EtherGREEN、In-Circuit Serial Programming、ICSP、INICnet、Inter-Chip Connectivity、JitterBlocker、KleerNet、KleerNetロゴ、memBrain、Mindi、MiWi、motorBench、MPASM、MPF、MPLAB Certifiedロゴ、MPLAB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、PowerSmart、PureSilicon、QMatrix、REAL ICE、Ripple Blocker、SAM-ICE、Serial Quad I/O、SMART-I.S.、SQI、SuperSwitcher、SuperSwitcher II、Total Endurance、TSHARC、USBCheck、VariSense、View Sense、WiperLock、Wireless DNA、ZENAは米国と他の国に於けるMicrochip Technology Incorporatedの商標です。

SQTPは米国に於けるMicrochip Technology Incorporatedの役務標章です。

Silicon Storage Technologyは他の国に於けるMicrochip Technology Inc.の登録商標です。

GestICは他の国に於けるMicrochip Technology Inc.の子会社であるMicrochip Technology Germany II GmbH & Co. KGの登録商標です。

ここで言及した以外の全ての商標はそれら各々の会社の所有物です。

© 2018年、Microchip Technology Incorporated、米国印刷、不許複製

DNVによって認証された品質管理システム

ISO/TS 16949

Microchipはその世界的な本社、アリゾナ州のチャンドラーとテンペ、オレゴン州グラシャムの設計とウェハー製造設備とカリフォルニアとインドの設計センターに対してISO/TS-16949:2009認証を取得しました。当社の品質システムの処理と手続きはPIC[®] MCUとdsPIC[®] DSC、KEELOQ符号飛び回りデバイス、直列EEPROM、マイクロ周辺機能、不揮発性メモリ、アナログ製品用です。加えて、開発システムの設計と製造のためのMicrochipの品質システムはISO 9001:2000認証取得です。

日本語© HERO 2018.

本応用記述はMicrochipのAN2711応用記述(DS00002711B-2018年10月)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。



MICROCHIP

世界的な販売とサービス

米国	亜細亜/太平洋	亜細亜/太平洋	欧州
本社 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 技術支援: http://www.microchip.com/support ウェブアドレス: www.microchip.com	オーストラリア - シドニー Tel: 61-2-9868-6733 中国 - 北京 Tel: 86-10-8569-7000 中国 - 成都 Tel: 86-28-8665-5511 中国 - 重慶 Tel: 86-23-8980-9588 中国 - 東莞 Tel: 86-769-8702-9880 中国 - 広州 Tel: 86-20-8755-8029 中国 - 杭州 Tel: 86-571-8792-8115 中国 - 香港特别行政区 Tel: 852-2943-5100 中国 - 南京 Tel: 86-25-8473-2460 中国 - 青島 Tel: 86-532-8502-7355 中国 - 上海 Tel: 86-21-3326-8000 中国 - 瀋陽 Tel: 86-24-2334-2829 中国 - 深圳 Tel: 86-755-8864-2200 中国 - 蘇州 Tel: 86-186-6233-1526 中国 - 武漢 Tel: 86-27-5980-5300 中国 - 西安 Tel: 86-29-8833-7252 中国 - 廈門 Tel: 86-592-2388138 中国 - 珠海 Tel: 86-756-3210040	インド - ハンガロール Tel: 91-80-3090-4444 インド - ニューデリー Tel: 91-11-4160-8631 インド - フネー Tel: 91-20-4121-0141 日本 - 大阪 Tel: 81-6-6152-7160 日本 - 東京 Tel: 81-3-6880-3770 韓国 - 大邱 Tel: 82-53-744-4301 韓国 - ソウル Tel: 82-2-554-7200 マレーシア - クアラルンプール Tel: 60-3-7651-7906 マレーシア - ペナン Tel: 60-4-227-8870 フィリピン - マニラ Tel: 63-2-634-9065 シンガポール Tel: 65-6334-8870 台湾 - 新竹 Tel: 886-3-577-8366 台湾 - 高雄 Tel: 886-7-213-7830 台湾 - 台北 Tel: 886-2-2508-8600 タイ - バンコク Tel: 66-2-694-1351 ベトナム - ホーチミン Tel: 84-28-5448-2100	オーストラリア - ウェルズ Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 デンマーク - コペンハーゲン Tel: 45-4450-2828 Fax: 45-4485-2829 フィンランド - エスポー Tel: 358-9-4520-820 フランス - パリ Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 ドイツ - ガルピング Tel: 49-8931-9700 ドイツ - ハーン Tel: 49-2129-3766400 ドイツ - ハイムブロン Tel: 49-7131-67-3636 ドイツ - カールスルーエ Tel: 49-721-625370 ドイツ - ミュンヘン Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 ドイツ - ローゼンハイム Tel: 49-8031-354-560 イスラエル - ラーナナ Tel: 972-9-744-7705 イタリア - ミラノ Tel: 39-0331-742611 Fax: 39-0331-466781 イタリア - ハドバ Tel: 39-049-7625286 オランダ - デルネン Tel: 31-416-690399 Fax: 31-416-690340 ノルウェー - トロンハイム Tel: 47-72884388 ポーランド - ワルシャワ Tel: 48-22-3325737 ルーマニア - ブカレスト Tel: 40-21-407-87-50 スペイン - マドリード Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 スウェーデン - イェテボリ Tel: 46-31-704-60-40 スウェーデン - ストックホルム Tel: 46-8-5090-4654 イギリス - ウォーキングム Tel: 44-118-921-5800 Fax: 44-118-921-5820
アトランタ Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455			
オースチン TX Tel: 512-257-3370			
ボストン Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088			
シカゴ Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075			
ダラス Addison, TX Tel: 972-818-7423 Fax: 972-818-2924			
デトロイト Novi, MI Tel: 248-848-4000			
ヒューストン TX Tel: 281-894-5983			
インディアナポリス Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380			
ロサンゼルス Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800			
ローリー NC Tel: 919-844-7510			
ニューヨーク NY Tel: 631-435-6000			
サンホセ CA Tel: 408-735-9110 Tel: 408-436-4270			
カナダ - トロント Tel: 905-695-1980 Fax: 905-695-2078			