AVR463：ATAVRBC100でのﾆｯｹﾙ水素電池の充電

要点

- ﾆｯｹﾙ水素電池充電用の完全な機能設計
- 10ﾋﾞｯﾄA/D変換器での高精度測定
- 単位部品化した"C"ｿｰｽ ｺｰﾄﾞ
- 容易に調整可能な電池と充電のﾊﾟﾗﾒｰﾀ
- 外部主装置との通信用直列ｲﾝﾀｰﾌｪｰｽ
- 電池EEPROMとの通信用単線ｲﾝﾀｰﾌｪｰｽ
- 電池IDと温度の読み込み用ｱﾅﾛｸﾞ入力
- 強化した温度管理用の内部温度感知器
- 電池の記憶と走行時ﾊﾟﾗﾒｰﾀ用のﾁｯﾌﾟ上EEPROM

8ﾋﾞｯﾄ AVR® マイクロ コントローラ

応用記述

1. 序説

この応用記述はATAVRBC100電池充電器参照基準設計(BC100)に基き、ﾆｯｹﾙ水素(NiMH)電池を充電するのに参照設計をどう使うのかに集中します。ﾌｧｰﾑｳｪｱは(IAR SystemsのEnbedded Workbenchを用いて)全体的にC言語で書かれ、他のAVR®ﾏｲｸﾛ ｺﾝﾄﾛｰﾗへの移設が簡単です。

この応用はATtiny861ﾏｲｸﾛ ｺﾝﾄﾛｰﾗに基きますが、ﾋﾟﾝ互換ﾃﾞﾊﾞｲｽのATtiny261やATtiny461のような他のAVRﾏｲｸﾛ ｺﾝﾄﾛｰﾗへ設計を移植することが可能です。ATtiny25/45/85のような少ﾋﾟﾝ数のﾃﾞﾊﾞｲｽも使えますが、機能的に減らされてです。

2. 動作の理屈

電池の充電は化学系に於いてｴﾈﾙｷﾞｰを再格納する可逆化学反応によって可能にされます。使われる化学物質に依存して、電池は或る特性を持ちます。電池に障害を与えるのを避けるために、これらの詳細な知識が必要とされます。

2.1. NiMH電池の技術

ﾆｯｹﾙ水素(NiMH)電池の技術はﾆｯｶﾄﾞ(NiCd)の後継で、ﾘﾁｳﾑに基く電池への踏み台と考えられます。その特性はNiCdにかなり類似していますが、減った繰り返し寿命(15頁の「参考文献」の1.をご覧ください)を犠牲にして概ね50%高いｴﾈﾙｷﾞｰ密度を提供します。NiMHはNiCdのようなﾒﾓﾘ効果からあまり被害を受けず、従って保管の間により少ない訓練(充放電)周回しか必要としません。加えて、NiMH電池はそれらがｶﾄﾞﾐｳﾑを含まないので環境的により優しく、従って処分に関する問題を提起しません。現在、ｶﾒﾗ、PDA、電動工具からﾊｲﾌﾞﾘｯﾄﾞ車まで全ての物に使われつつあります(訳補:原書執筆時点に於いて。訳時点では既に殆どがﾘﾁｳﾑ ｲｵﾝへ移行しています)。

ﾘﾁｳﾑ ｲｵﾝ電池が2倍のｴﾈﾙｷﾞｰ密度、より高いｾﾙ電圧、より短い充電時間を提供するとは言え、それらはﾆｯｹﾙに基く電池ほど安定ではありません。また、ﾆｯｹﾙに基く電池が通常の1.5Vのﾏﾝｶﾞﾝとｱﾙｶﾘの電池に近い電圧のため、それらはこのような電池によって給電される既存設備での使用に関してより向いています。

NiCdと比べて、NiMH電池の欠点は僅かにより高い自己放電率(室温に於いて月当たり概ね20%)、より低い過充電許容力、より短い繰り返し寿命(300～500充電周回)、それともっと熱を発するより長い充電時間を含みます。

2.1.1. 安全性

ﾆｯｹﾙに基く電池は殆ど安定ですが、内圧と発熱性のために激しく誤って扱われた(回路の短絡、不正な充電器での充電など)場合にｶﾞｽを発散または爆発するかもしれません。殆どの電池はこれを防ぐのを助けるために防爆弁、温度ﾋｭｰｽﾞ/ｽｲｯﾁ、または圧力ｽｲｯﾁと共にやって来ます。

Rev. 8098A-09/07, 8098AJ3-01/21

2.2. NiMH電池の充電

ニッケルに基く電池を再充電するための最も一般的な方法は定電流で行うことです。通常、遅い充電がメモリ効果や小充電効率(「参考文献」の7.)としても知られる結晶化を引き起こすため、充電電流は(急速充電の場合に)0.5～1.0C(脚注)の範囲が推奨されます。他方でこれよりも大きな電流は障害を引き起こすかもしれません。低い充電電流ではNiMH電池の満充電の検出もより難しくなります。

NiMH電池は70～90%間の充電効率で、与えられた充電電流と電池容量から計算するものよりもより長い間充電しなければならないことを意味します。

2.2.1. 安全性

充電周期の最後の方で、電池の正電極はまず満充電に達して酸素を生成させます。セル内の圧力増加を避けるため、酸素は負電極に於いて水を生成するように再結合されます。この処理は熱を発生しますが、これは制限されるべき充電電流を頼るので、酸素生成速度は再結合速度よりも大きくなることはありません。

圧力増加発生で、殆どの電池は安全弁か、または一旦圧力が高すぎになると充電電流を切断する同等の圧力スイッチを持ちます。ガスの排出は電解質が失われるために電池を劣化させます。もっと進んだ電池は圧力や温度が一旦閾値を越えると活動する保安電子回路を持ちます。

2.2.2. 電池温度

言及したように、NiMH電池は充電中に熱を発生し、従って損傷を防ぐために電池温度を監視することが重要です。急速充電用に示唆される温度範囲は5～45℃の領域ですが、充電の最後の短時間に関して45～60℃間の温度が受け入れ可能です(「参考文献」の5.と6.)。

普通、大きな電流は温度をより早く増加させます。高温はセル電圧と充電効率を減らし、より高い自己放電速度にさせます。より低い温度が酸素の再結合を遅くし、多分より速い圧力増加を引き起こすとは言え、最大効率のためにファンでの電池冷却が時々行われます。

2.2.3. 充電時間

充電がどの位の長さかを決めるのは電池の充電状態がその電圧から確かな判断ができないため、そう容易ではありません。加えて、それは電池の容量と充電電流に依存します。

無効充電のため、典型的な充電入力は最大繰り返し寿命(120%)または容量(150%)が望まれるのかに依存して、120～150%の領域に広がります。従って1.0C充電電流との仮定で、空の電池の満充電は1.2～1.5時間かかるでしょう。NiMH電池がNiCdのような過充電許容力がないため、満充電を検出するための測定が必要とされなければなりません。

2.2.4. 満充電検出

約60%充電入力に於いてNiMHセル内の熱生成は増加が始まり、満充電を検知するのに使われるかもしれません。温度増加速度が分当たり1℃に達すれば充電の終りとのことが推奨されます(「参考文献」の5.と6.)。

満充電の別の合図は、これが高温または低充電電流ではっきりしないとしても、酸素再結合のための電圧に於ける低下です。示唆される限度はセル当たり6～20mVの範囲です。下限は過充電の量を減らしますが、電気的雑音や充電中の他の一般的な電圧に於ける低下のため、早まった充電終了の危険も高めます。

2.2.5. 代表的な急速充電特性

電池の仕様は製造業者のデータシートで常に確かめられるべきです。右は代表的なニッケル水素電池の急速充電特性の要約です。

表2-1. 急速充電の代表的な充電特性

項目	代表的な値
充電時間	1.5～3時間
充電電流	0.5～1C
充電効率	70～90%
充電電圧	1.4～1.6V
温度範囲	5～60℃(最大)

2.2.6. 代表的な電池特性

右の表はNiMH電池に関する標準的なデータを要約します(「参考文献」の5.と6.)。

表2-2. 代表的なNiMH電池特性

項目	代表的な値
標準電圧	1.2V
開回路電圧	1.25～1.35V
代表終止電圧	1.0V

脚注: Cは時間当たりの電池容量、例えばmAを表します。

2.2.7. 3段階の急速充電

この応用記述のために選んだ充電手順は文献でと、電池と充電器の製造業者によって推奨されるような、最高級充電に従った急速充電です。けれどもそうでなければそれらが分からないため、これはｺﾝﾊﾟｲﾙ時に電池の仕様が定義されている必要があります。この手順は以下のとおりです。

- 予備検定 – 電池電圧が1.0Vに達するまで最大2(TBD)分間、低電流(0.1C)で充電します。時間超過は電池がかなり損傷しているようであることを意味します。
- 急速充電 – 温度増加速度が分当たり1℃に達するか、または電圧がｾﾙ当たり15mV低下するかのどちらかになるまで最大1.5時間の間、高電流(1C)で充電します。
- 低速充電 – 満充電を保証するため、30分間、低電流(0.1C)で電池を一杯に充電します。この電流は過充電の場合に酸素再結合を維持するために充分に低い電流です。

安全性のために、後の2段階に対して50℃の温度制限、最初に対しては35℃が設定されています。

自己放電を相殺するために不定期間ﾄﾘｸﾙ充電(0.003～0.05C)に電池を置くことも推奨されます。これは実装されていませんが、Charge()内で最後の充電段階を充電することによって容易にそのように行われるかもしれません。一定期間で電池を切り替えることは有益で、それは未充電電池の検出も許すでしょう。

2.3. 電池充電器

この応用記述はAtmelによるATAVRBC100電池充電器参照基準設計が目的対象です。この参照基準設計はむしろ複雑で特徴的な負荷を持ちますが、この応用は設計の核機能、降圧変換器だけに集中します。BC100のより多くの情報については「**AVR451:BC 100 ﾊｰﾄﾞｳｪｱ使用者の手引き**」(「**参考文献**」の**2.**)を参照してください。

2.3.1. ﾏｲｸﾛ ｺﾝﾄﾛｰﾗ

BC100は主(ATmega644)と従(ATtiny85またはATtiny861(既定))の2つのﾏｲｸﾛ ｺﾝﾄﾛｰﾗを持ちます。主ﾏｲｸﾛ ｺﾝﾄﾛｰﾗはこの応用の範囲外ですが、何時でも主装置が従装置からﾃﾞｰﾀを要求するかもしれないように、ﾏｲｸﾛ ｺﾝﾄﾛｰﾗがお互いに通信する能力があることが留意されるかもしれません。

従ﾏｲｸﾛ ｺﾝﾄﾛｰﾗは電池充電に関連する全ての作業を処理する完全な能力があり、主ﾏｲｸﾛ ｺﾝﾄﾛｰﾗの存在を必要としません。それは電池用のｺﾈｸﾀを継続的に走査し、(電池を)見つけたなら、必要とされる時にそれらを充電します。従ﾏｲｸﾛ ｺﾝﾄﾛｰﾗは何らかの変則に関してﾊｰﾄﾞｳｪｱも継続的に監視します。

2.3.2. 電源

この応用記述は電源に注目しません。けれども動作が信頼に足ることを保証するためにﾌｧｰﾑｳｪｱが継続的に入力電圧水準を監視することに留意されるかもしれません。

2.3.3. 降圧切り替え

従ﾏｲｸﾛ ｺﾝﾄﾛｰﾗ上のﾌｧｰﾑｳｪｱはBC100基板上の3つの降圧変換器のどれかを制御します。既定は電圧を調整して電池に電流を流すためにﾏｲｸﾛ ｺﾝﾄﾛｰﾗの高速PWM出力を使うことです。降圧変換器の電圧(と電流)はPWM信号のﾃﾞｭｰﾃｨ ｻｲｸﾙに直接比例します。

3. 電池充電器ﾊｰﾄﾞｳｪｱ

この応用記述はATAVRBC100電池充電器参照基準設計に基きます。詳細なﾊｰﾄﾞｳｪｱ説明はこの資料で提供されません。詳細情報については「**AVR451:BC100ﾊｰﾄﾞｳｪｱ使用者の手引き**」をご覧ください。

3.1. 構成設定

ATAVRBC100電池充電器参照基準設計は下で詳述されるように構成設定されなければなりません。

3.1.1. ﾏｲｸﾛ ｺﾝﾄﾛｰﾗ

ﾊｰﾄﾞｳｪｱは以下のようにされるべきです。

- SC300ｿｹｯﾄが空であることを確認してください。
- ATtiny861をSC301ｿｹｯﾄに居させてください。

他のAVRﾏｲｸﾛ ｺﾝﾄﾛｰﾗを使うことが可能ですが、この応用はATtiny861の使用に最適化されています。ｺﾝﾊﾟｲﾙされたｺｰﾄﾞ量が減少されるなら、ATtiny261とATtiny461(「**参考文献**」の**3.**)のようなﾋﾟﾝ互換の置換が使われるかもしれません。これはｺﾝﾊﾟｲﾗの最適化を増すのと、ﾌｧｰﾑｳｪｱから不要な機能を取り去ることによって行うことができます。

他のﾏｲｸﾛ ｺﾝﾄﾛｰﾗ選択にはATtiny25,ATtiny45,ATtiny85を含みます(「**参考文献**」の**4.**)。(他の8ﾋﾟﾝAVRﾏｲｸﾛ ｺﾝﾄﾛｰﾗと同じ様に)これらはBC100上のSC300ｿｹｯﾄを使います。少ﾋﾟﾝ数のために8ﾋﾟﾝのﾏｲｸﾛ ｺﾝﾄﾛｰﾗが既定の20ﾋﾟﾝよりも少ない機能しか提供しないことが注意されるべきです。

3.1.2. ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ ｺﾈｸﾀ

ﾏｲｸﾛ ｺﾝﾄﾛｰﾗはSPIまたはﾃﾞﾊﾞｯｸﾞWIREのどちらかを使って6ﾋﾟﾝのJ301ｺﾈｸﾀ経由でﾌﾟﾛｸﾞﾗﾐﾝｸﾞすることができます。

BC100の或るﾊｰﾄﾞｳｪｱ改訂版に於いて、R303を取り去ってU202の15番ﾋﾟﾝをHi-Zに(ATmega644からの/OEをLowに設定)する必要があるかもしれません。この手続きは外部の書き込み器やﾃﾞﾊﾞｯｶﾞによる使用のために/RESET線を自由にしますが、主ﾏｲｸﾛ ｺﾝﾄﾛｰﾗが従装置をﾘｾｯﾄする可能性を取り去ります。必要とされる限り、基板を変えないでください。代わりに、ﾏｲｸﾛ ｺﾝﾄﾛｰﾗは基板外で常にﾌﾟﾛｸﾞﾗﾐﾝｸﾞすることができます。

3.1.3. ｼﾞｬﾝﾊﾟ

ｼﾞｬﾝﾊﾟは以下のように構成設定されるべきです。

- J405とJ406：ｼﾞｬﾝﾊﾟを1/4に設定してください(最大測定可能電圧=10V)。
- 300mAhの電池に対して：J401,J404,J408は降圧変換器C(20V/1A)を許可するように設定されるべきです。
- 1300mAhの電池に対して：J400,J401,J403,J407,J408は降圧変換器B(30V/2.5A)を許可するように設定されるべきです。

他の構成設定も可能ですが、ﾌｧｰﾑｳｪｱの変更が必要かもしれません。ADC.hﾌｧｲﾙでVBAT_RANGE変数をご覧ください。

3.1.4. 電池

この応用の試験のために、MINAMOTOの2つの標準3ｾﾙNiMH電池が使われます。利用可能なﾃﾞｰﾀは指定容量と公称電圧だけです。両電池は各々300mAhと1300mAhの容量で3.6Vで評価されます。これらの電池は登録ID、EEPROM、NTCを含みません。

3.1.5. 電池登録ID

いくつかの電池は電池形式を認識するのに使われる登録ID(RID)を含むかもしれません。そして充電器はこれらの項目を固定的に保持する代わりに、容量、最大充電電流、充電時間のようなﾃﾞｰﾀを知ることができます。RIDが使われる場合、登録値と関連するﾃﾞｰﾀは電池製造業者から入手可能です。この応用はこれを支援しますが、RIDを必要とせず、使われるなら、RID参照表が更新されるべきです。登録関係は次のように接続されるべきです。

登録IDが接続されない、またはその値がA/D変換の飽和を引き起こす場合、既定の電池ﾃﾞｰﾀが使われるかもしれません。13頁の「構成設定」をご覧ください。

表3-1. 充電器への電池登録ID接続

電池ﾋﾟﾝ	充電器ｺﾈｸﾀ
RID	SCL
GND	BATTERY-

3.1.6. NTC

NiMH電池の急速充電中の温度測定は重要で、従っていくつかの製造業者は度々それらの電池内にｻｰﾐｽﾀを内包します。通常、これらは負性温度係数(NTC)です。

この応用のために、三菱のRH16-3H103FB NTCが使われ、これに応じてNTC参照表が置かれます。NTCは次のように接続されるべきです。

この応用ではNTCが接続されない場合にA/D変換が飽和され、温度が-1℃として記録されます。これは0未満の最低電池温度が設定されていない限り、充電を停止するでしょう。

表3-2. 充電器へのNTCｻｰﾐｽﾀ接続

電池ﾋﾟﾝ	充電器ｺﾈｸﾀ
NTC	NTC/RID
GND	BATTERY-

3.1.7. ﾃﾞｰﾀEEPROM

いくつかの電池は充電と製造のﾃﾞｰﾀを格納するための組み込みEEPROMが装備されています。この応用は単線ｲﾝﾀｰﾌｪｰｽ経由でEEPROMの読み込みを支援します。既定は次のように接続されたDS2502 EEPROMです。

EEPROMが電池充電器に接続されない場合、応用は単にその不在を無視するでしょう。

表3-3. 充電器へのDS2502外部EEPROM接続

電池ﾋﾟﾝ	充電器ｺﾈｸﾀ
DATA	1-WIRE/SDA
GND	BATTERY-

3.1.8. 供給電圧

より高い供給電圧とより高い最小電流を降圧切り換えは提供することができます。例えば、供給電圧が約9Vで4.20Vで電池を充電するのに降圧変換器Cが使われる場合、得られる最小電流は約80mAです。この点に於けるPWMﾃﾞｭｰﾃｨ ｻｲｸﾙでの最小減少(換言すると、1によるOCR1B内容の減少)は実際上、電池への電流をOFFにするでしょう。

電池充電電圧より約3V高い供給電圧を使うことが推奨されます。この応用では充電中に電池電圧が代表的に最大4.5Vを出力し、故に供給電圧は7.5Vが推奨されます。

ﾊｰﾄﾞｳｪｱが提供できる最小充電電流をより低めるための別の方法はより大きなｲﾝﾀﾞｸﾀを降圧変換器に使うことです。BC100でこれは降圧変換器Aを意味します。

4. 電池充電器ソフトウェア

ファームウェアはIAR SystemsのEmbedded Workbench™を使ってC言語で書かれています。ファームウェアが全体的にCで書かれているため、他のAVR Cコンパイラへ移設することは難しくないでしょう。いくつかのコンパイラ特有の項目は多分書き直される必要があるでしょう。

下表はコンパイラのプロジェクトに関連するファイルを一覧にします。

表4-1. プロジェクト ファイル

ファイル名	形式	注記
ADC.c	Cソース ファイル	A/D変換に関連する関数
ADC.h	ヘッダ ファイル	
battery.c	Cソース ファイル	電池制御とデータ獲得に関連する関数と、既定電池データの定義
battery.h	ヘッダ ファイル	
chargefunc.c	Cソース ファイル	充電に関連する関数
chargefunc.h	ヘッダ ファイル	
enums.h	ヘッダ ファイル	(SPI通信)主/従両方で使われるtime.cとUSI.c用の列挙
main.c	Cソース ファイル	主プログラム/プログラム入口
main.h	ヘッダ ファイル	
menu.c	Cソース ファイル	状態機構定義
menu.h	ヘッダ ファイル	
NiMHcharge.c	Cソース ファイル	NiMH電池充電用の充電状態関数
charge.h	ヘッダ ファイル	
NiMHspecs.h	ヘッダ ファイル	NiMHセルと電池仕様の定義
OWI.c	Cソース ファイル	単線インターフェースに関連する関数
OWI.h	ヘッダ ファイル	
PWM.c	Cソース ファイル	パルス幅変調出力の生成に関連する関数
PWM.h	ヘッダ ファイル	
statefunc.c	Cソース ファイル	各種状態関数
statefunc.h	ヘッダ ファイル	
structs.h	ヘッダ ファイル	(SPI通信)主/従両方で使われるADC.cとbattery.c用の構造体
time.c	Cソース ファイル	時間維持と時間測定に関連する関数
time.h	ヘッダ ファイル	
USI.c	Cソース ファイル	直列インターフェース(SPI通信)に関連する関数
USI.h	ヘッダ ファイル	

4.1. 概要

ファームウェアは2つのNiMH電池を充電するのに必要とする全ての関数を統合します。電池は一方が充電され得る間に他方がアイドルとなるように独立したポートに接続されます。ファームウェアは完全に自動化され、自立した電池の監視と充電の能力がありますが、またBC100で実装されているもののように、主マイクロ コントローラと共に使われるかもしれません。

既定により、ファームウェアはATtiny861(構築任意選択:デバッグ)またはATtiny461(構築任意選択:提出)に適合します。ファームウェアのメモリ必要条件は下表で要約されます。

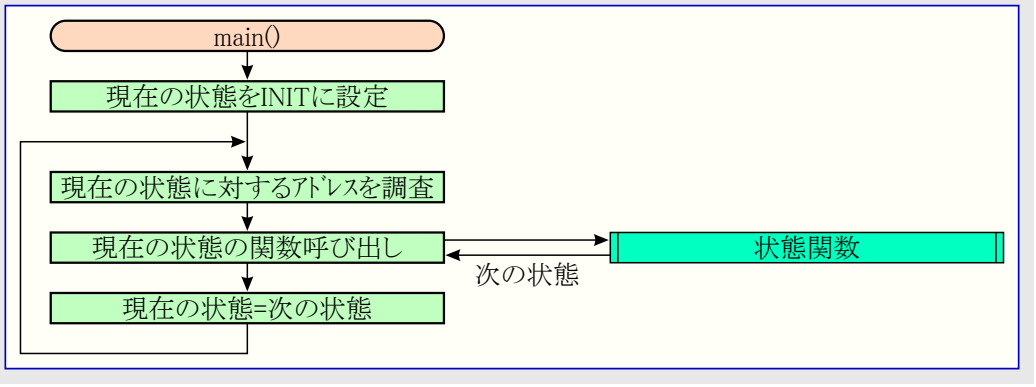
表4-2. ファームウェアのメモリ必要条件

構築任意選択	メモリ	概ねの値 (バイト)
デバッグ	CODE (フラッシュ メモリ)	5800
	DATA (SRAM)	270
	XDATA (EEPROM)	130
提出	CODE (フラッシュ メモリ)	3900
	DATA (SRAM)	270
	XDATA (EEPROM)	130

4.2. 状態機構

状態機構はかなり簡単でmain()関数に属します。単純に実行するための次の関数のｱﾄﾞﾚｽを調べて、そしてその関数へ飛びます。状態機構の流れ図が下図で図解されます。

図4-1. 状態機構を含むmain関数の流れ図



戻りに於いて、状態機構は戻り引数として次の状態を示すことを期待します。承認された戻り符号が下表で記述されます。

表4-3. 状態機構符号(ｿｰｽ ｺｰﾄﾞとmenu.hをご覧ください)

ﾗﾍﾞﾙ名 (注1)	関連する関数 (注2)	説明
INIT	Initialize()	入口の状態
BATCON	BatteryControl()	ﾊｰﾄﾞｳｪｱと電池を調査
PREQUAL	Charge()	電池電圧上昇、安全性検査
SLEEP	Sleep()	低消費電力動作形態
FASTCHARGE	Charge()	1.0C定電流で充電
TRICKLECHARGE	Charge()	0.1C定電流で充電
ENDCHARGE	Charge()	充電成功終了
DISCHARGE	Discharge()	
ERROR	Error()	可能ならば、異常を解決

注1: 先行する"ST_"を除いたﾗﾍﾞﾙ名

注2: ｿｰｽ ｺｰﾄﾞで宣言された関数名

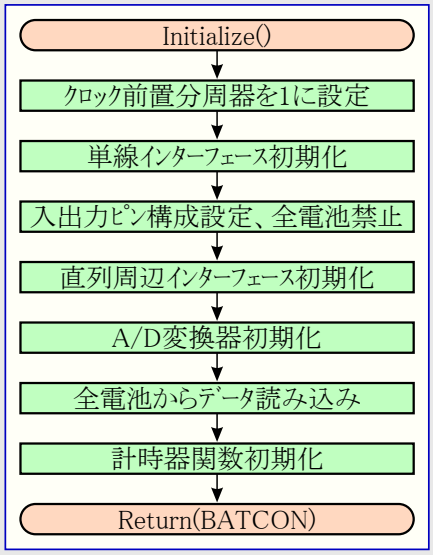
状態関数は以降の項で記述されます。

4.2.1. Initialize()

初期化関数はﾃﾞﾊﾞｲｽ ﾘｾｯﾄ後に実行される最初の状態関数です。この関数の流れ図は右図で示されます。

初期化関数は常に電池制御のための状態関数を示す同じ戻り符号で抜け出します。

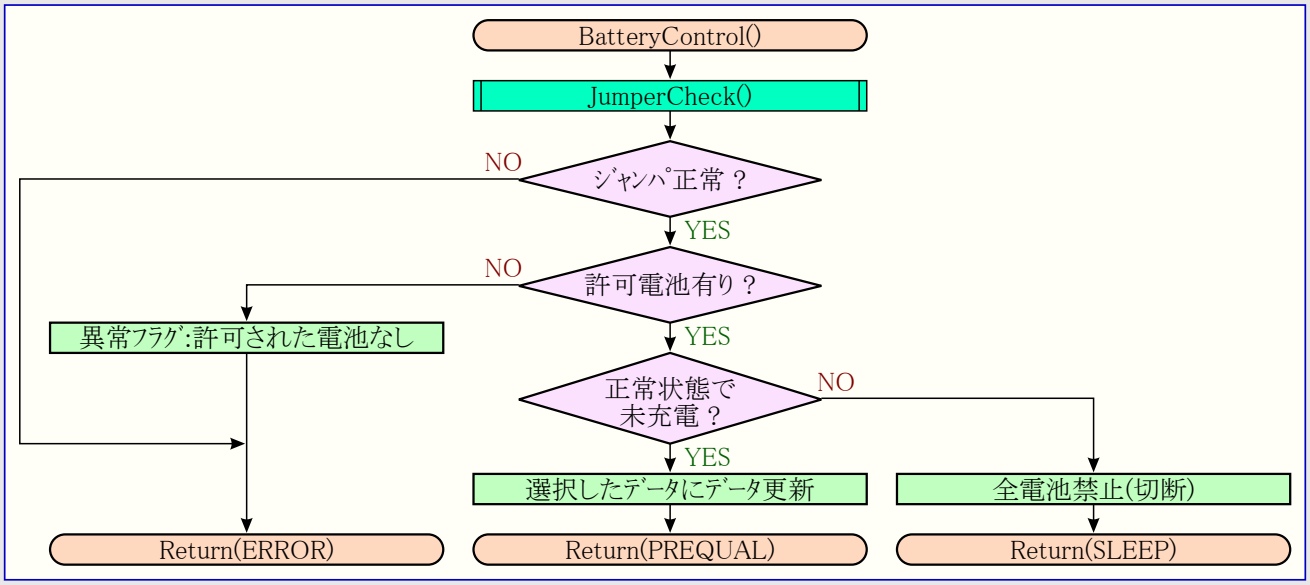
図4-2. 初期化関数の流れ図



4.2.2. BatteryControl()

電池制御関数はｼﾞｬﾝﾊﾟが正しく設定されていること確かめ、そして充電を必要とする何れかの許可された電池が存在するかを確かめるために調べます。ﾌﾟﾛｸﾞﾗﾑの流れは下図で図解されます。

図4-3. 電池制御関数の流れ図



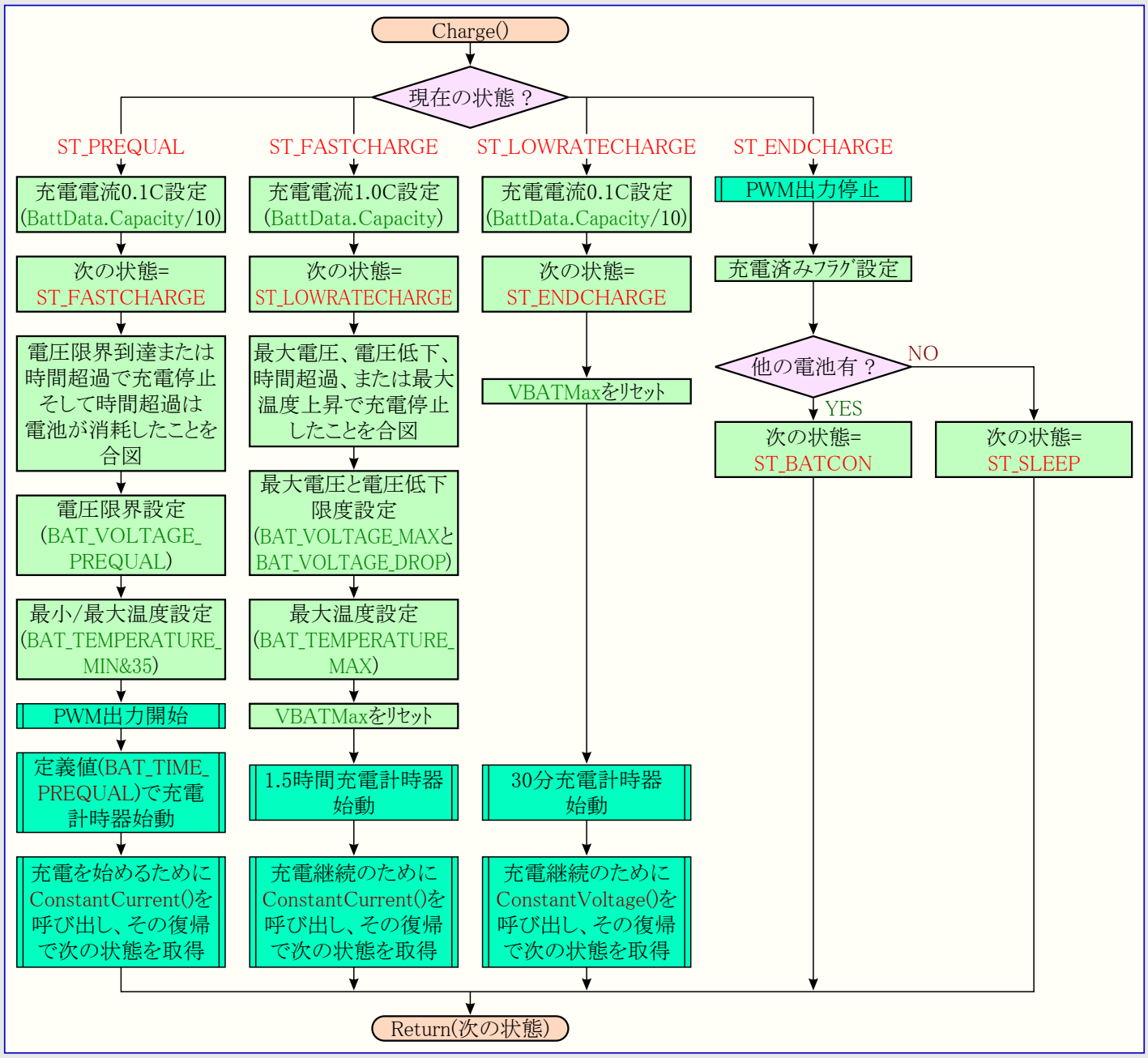
4.2.3. Charge()

充電関数は段階に分けられる充電算法を含みます。この応用に関して、それは4つの段階を持ちます。

- 予備検定 – 電池電圧が0.8～1.0Vなら、2分間0.1Cのような低速で充電します。電池が制限時間内に1.0Vへ上昇しない場合、電池はおそらく損傷されています。最大温度を35℃に制限します。
- 急速充電 – 最大で1.5時間1.0Cで充電し、電圧がｾﾙ当たり10～15mV落ちる、または温度増加速度が分当たり1℃に達した時に終了します。最大温度を50℃に制限します。
- 満充電 – 30分間0.1C充電で電池を満たします。最大温度を50℃に制限します。
- 充電終了 – 休止状態へ行くのか、または他の電池の充電を試みるのかのどちらかを決めます。

ChargeParametersとHaltParametersはこの関数に於ける中心的な変数です。各段階後、関数は次に望む状態をmain()に返します。この状態関数のﾌﾟﾛｸﾞﾗﾑの流れは次図で図解されます。

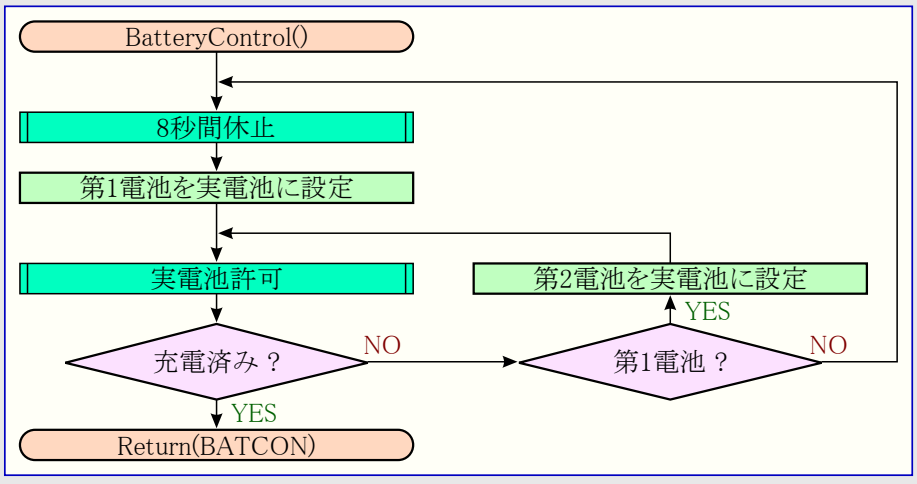
図4-4. 充電状態関数の流れ図



4.2.4. Sleep()

応用は電池が完全に充電されてしまった時に休止形態へ移行します。それは電池の現在の状態を調べるために一定間隔で起き上がります。休止形態は何れかの電池が充電を必要とすると直ぐに終了されます。

図4-5. 休止関数の流れ図

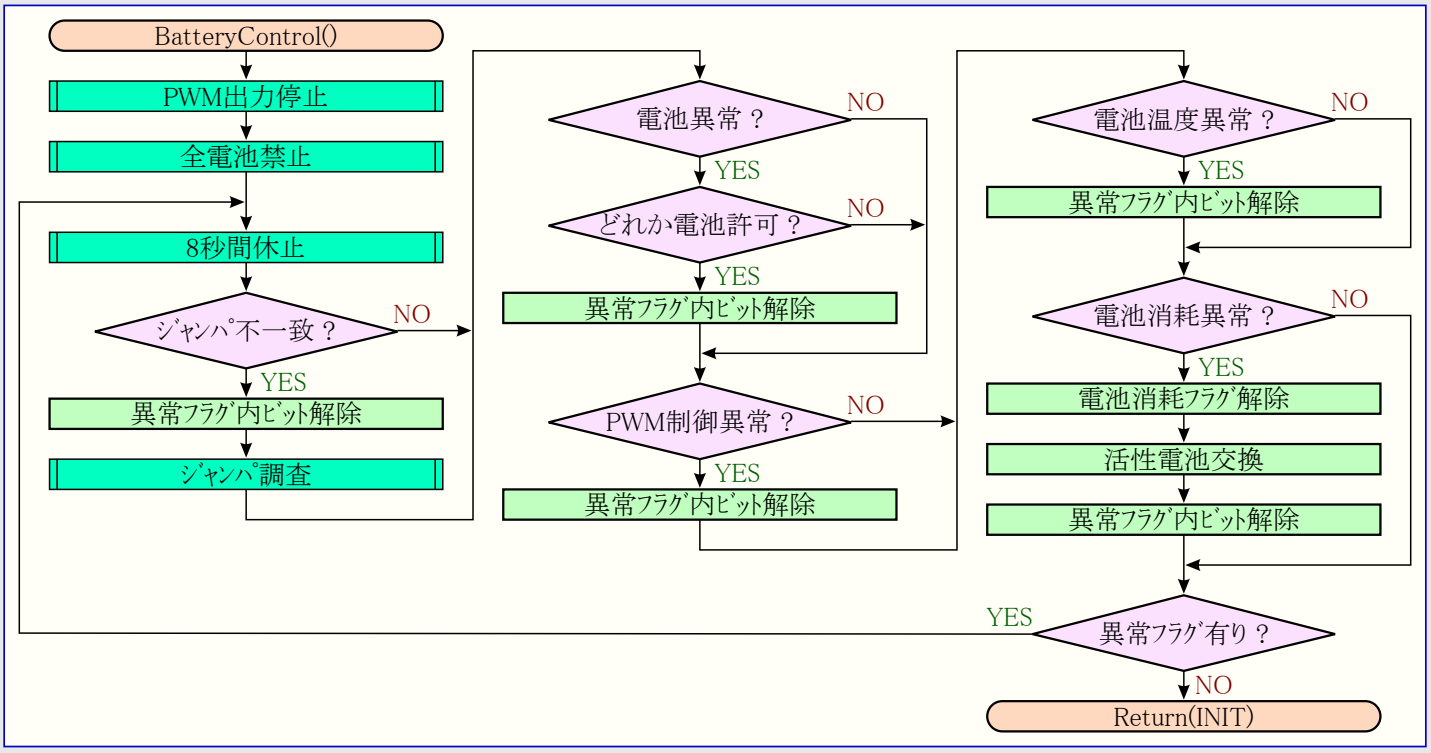


4.2.5. Error()

異常が起きた時にﾌﾟﾛｸﾞﾗﾑの流れはここで転換されます。異常処理部は殆どの一般的な問題の解決を試みるためのいくつかの簡単な算法を含みます。ﾌﾟﾛｸﾞﾗﾑ実行は全ての異常元が解除された時に異常処理部を抜け出します。

このﾌﾟﾛｸﾞﾗﾑの流れは下図で図解されます。

図4-6. 異常処理部の流れ図



4.3. 充電関数

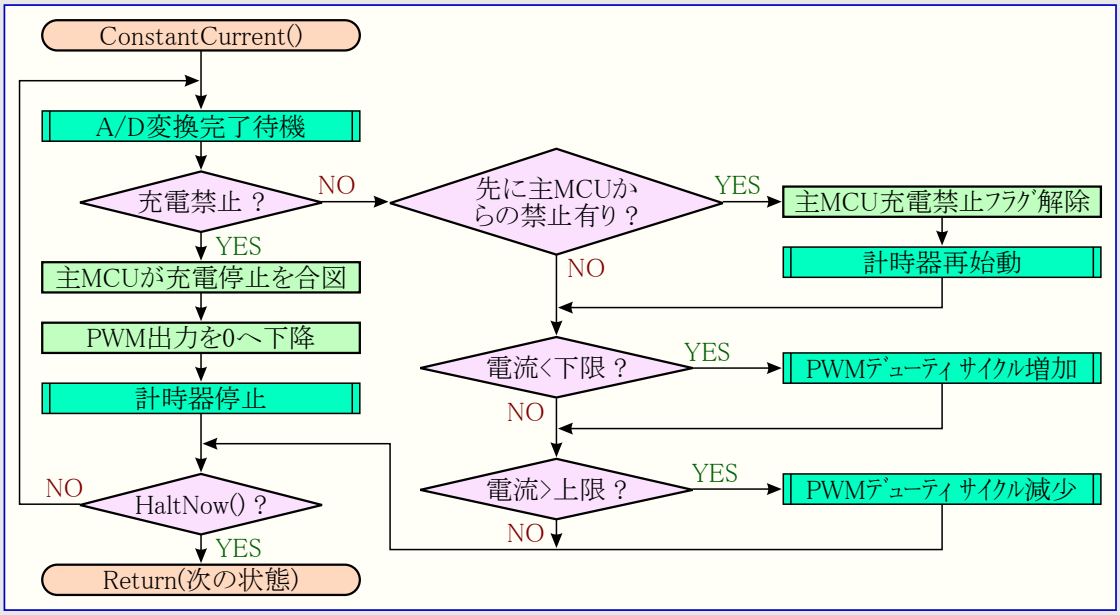
これらの関数は全てのﾊﾟﾗﾒｰﾀが設定された後でCharge()によって呼ばれます。

4.3.1. 定電流と定電圧

それらが限度内に保持を試みるためのA/D変換測定が何かを別にして、これら2つの関数は同じです。従って、ConstantCurrent()の流れ図だけが下図で図解されます。これら両方ともChargeParameter変数を使います。

主ﾏｲｸﾛ ｺﾝﾄﾛｰﾗが存在する場合、充電禁止の合図によって一時的に充電を停止するかもしれません。これは長い直列転送中に於ける電池の損傷を防ぐためです。

図4-7. ConstantCurrent()用流れ図



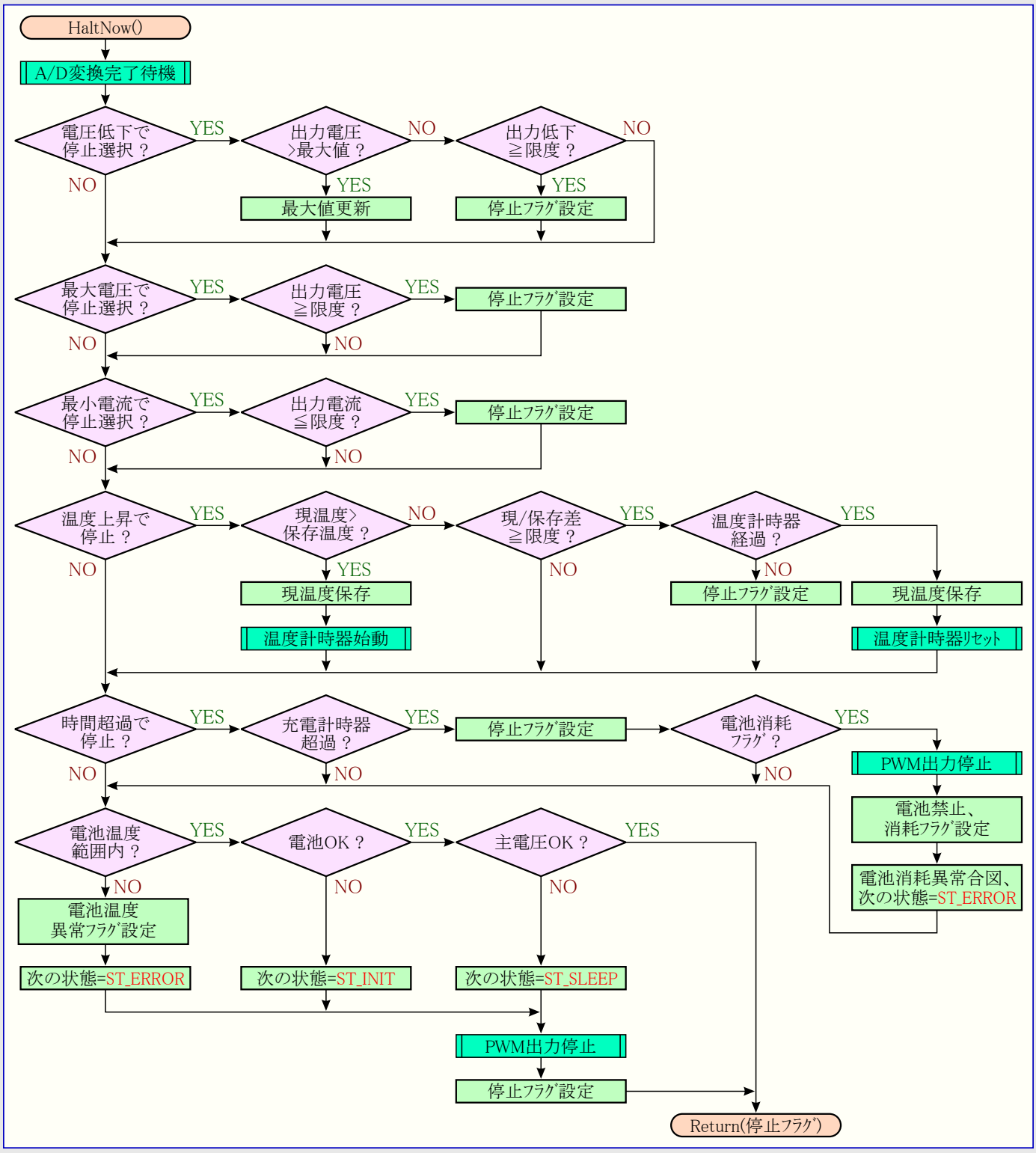
4.3.2. 充電停止判定

充電停止はHaltNow()によって判断されます。この関数は充電の段階が終わったかどうかを決めるために、それらが繰り返す度毎にConstantCurrent()とConstantVoltage()によって呼び出されます。

HaltParameters変数で使用者は何の項目で充電が停止されるべきで、例えば制限時間経過の場合に異常が合図されるべきかを指定することができます。また異常ﾌﾗｸﾞは次の状態で設定されるST_ERRORに帰着し、これによって充電を中止します。何も異常が合図されなければ、Charge()内で先に設定された次に望まれる状態が適用されます。

最後に、関数は温度が限度内か、電池が正常か、主電圧が最小以上かを調べます。これらの検査のどれかが誤りなら、次の状態は適切な異常処理部(ST_ERROR,ST_INIT,ST_SLEEP)に設定され、充電は中止されます。

図4-8. HaltNow()用流れ図



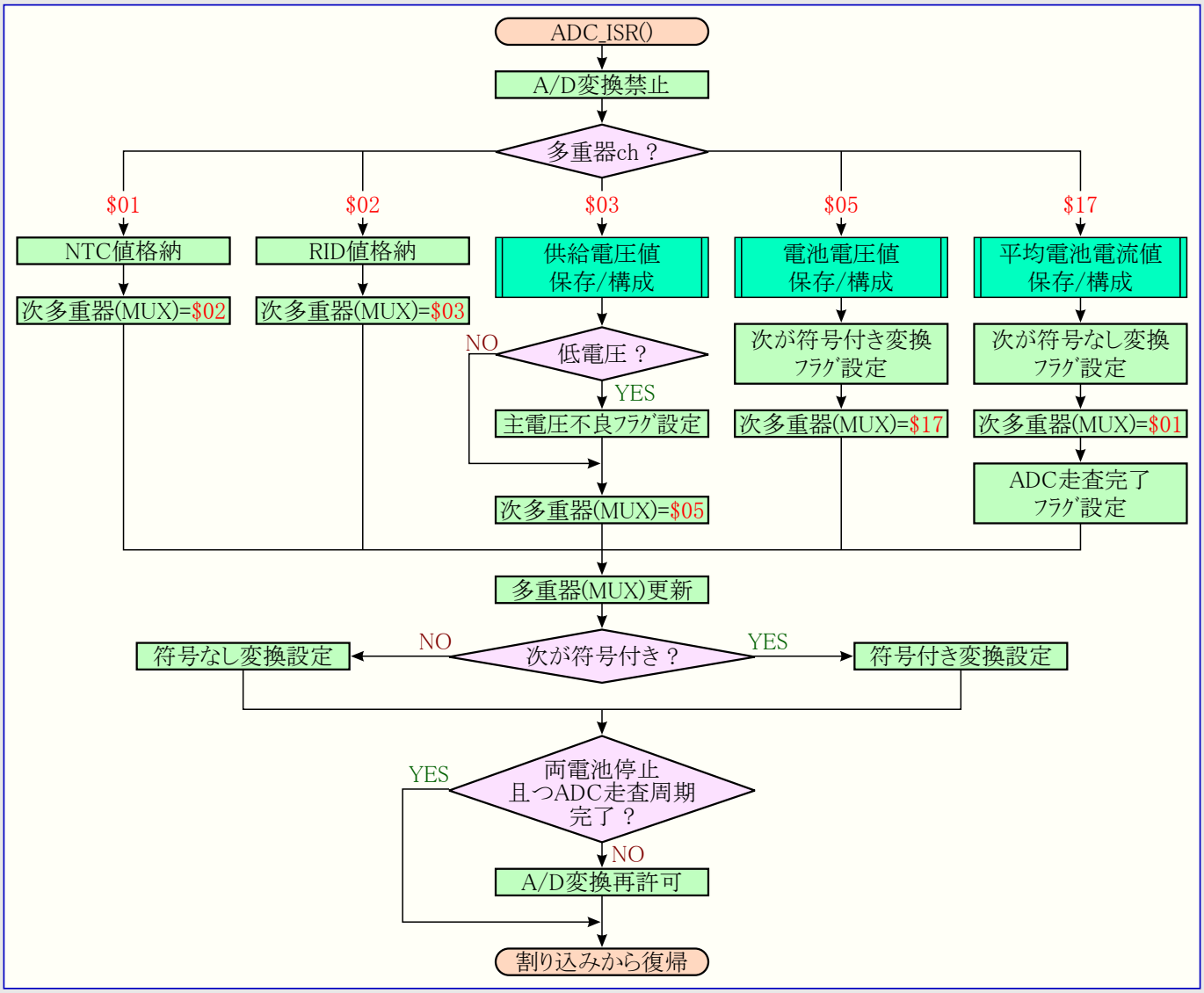
4.4. その他の関数

ﾌﾟﾛｸﾞﾗﾑの流れは主に状態に基きますが、いくつかの処理は背面で行われます。これはA/D変換、時間維持、直列ｲﾝﾀｰﾌｪｰｽの処理を含みます。これらの関数の全てが割り込み駆動です。

4.4.1. A/D変換

A/D変換は多数のﾁｬﾈﾙからﾃﾞｰﾀを読むのに多重器を用います。下の流れ図で図解されるように、変換の最後でA/D変換割り込み処理ﾙｰﾁﾝ(ISR)が呼ばれます。ISR完了後、ﾌﾟﾛｸﾞﾗﾑ実行は通常に戻ります。

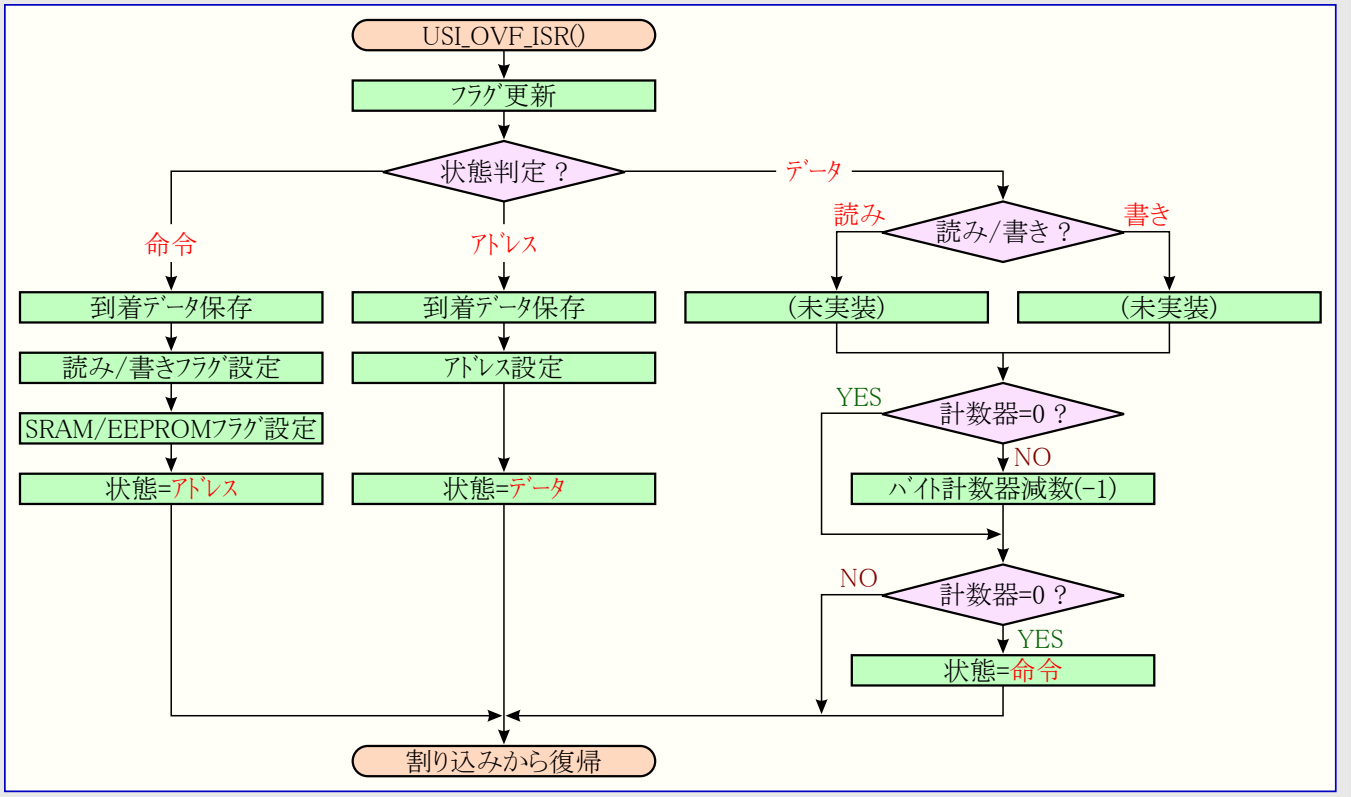
図4-12. A/D変換割り込み処理ﾙｰﾁﾝの流れ図



4.4.2. 主/従通信

この応用は自立型として動くように設計されていますが、他のマイクロ コントローラでの協力動作も支援します。多用途直列インターフェース(USI)はマイクロ コントローラ間の通信に使うことができます。このインターフェースに対する基本規約は開発されていますが、完成させるにはいくつかの機能が必要です。

図4-13. USI溢れ割り込み処理ルーチンの流れ図



5. 即時開始の手引き

この章はｿﾌﾄｳｪｱの構成設定、作成、書き込み(ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ)方法を記述します。

5.1. 構成設定

ｺﾝﾊﾟｲﾙ時に重要な殆どの定数が下表で一覧にされます。

表5-1. 電池に関連するｺﾝﾊﾟｲﾙ時定数 (battery.c,battery.h,NIMHspecs.hｿｰｽ ﾌｧｲﾙをご覧ください。)

ﾗﾍﾞﾙ名	説明
BAT_CELL_NUMBER	電池のｾﾙ数。BAT_VOLTAGE_MAX、_LOW、_MIN、_PREQUALを定義するために、定義されたｾﾙ電圧の各々がこれで乗算されます。
CELL_VOLTAGE_SAFETY	充電されるべき電池が不一致の場合に、電池内の追加ｾﾙ毎、換言するとBAT_CELLNUMBER-1に関して、この定数がCELL_VOLTAGE_MAXから引かれます。
CELL_VOLTAGE_MAX	充電されるべきｾﾙに対する最大電圧。
CELL_VOLTAGE_LOW	ｾﾙが充電されたと見做される最低電圧。充電は電圧がこの水準以下に落ちた時に始まります。
CELL_VOLTAGE_MIN	充電が始められる最低電圧。一般的に電池の更なる放電が損傷を引き起こす下限電圧に設定されるべきです。
CELL_VOLTAGE_PREQUAL	予備検定(充電)中に充電されるべき電池の電圧。
BAT_TEMPERATURE_MAX	電池許容最高温度。これ越えで充電は停止されるか、または開始されません。
BAT_TEMPERATURE_MIN	電池許容最低温度。これ未満で充電は停止されるか、または開始されません。
BAT_CURRENT_PREQUAL	予備検定(充電)中の充電電流。
BAT_CURRENT_HYST	充電電流ﾋｽﾃﾘｼｽ。目標±この値内の時に電流は調整されません。
BAT_VOLTAGE_HYST	充電電圧ﾋｽﾃﾘｼｽ。目標±この値内の時に電圧は調整されません。
BAT_VOLTAGE_PREQUAL	予備検定(充電)段階中の目標電圧。この電圧未到達の場合に電池は消耗と記されます。
BAT_TIME_PREQUAL	予備検定(充電)段階で費やされる最大時間。
DEF_BAT_CAPACITY	既定電池容量。
DEF_BAT_CURRENT_MAX	既定最大充電電流。
DEF_BAT_TIME_MAX	既定最大充電時間。
DEF_BAT_CURRENT_MIN	既定中止充電電流。
ALLOW_NO_RID	定義されていれば、RIDなしの(または参照表に一致しない)電池は既定電池(情報)を用いて充電させます。
RID[].LowとRID[].High	値がこれらの限度内の場合にRID抵抗一致と仮定します。
RID[].Capacity	与えられたRIDに対する電池容量。
RID[].Icharge	与えられたRIDに対する充電電流。
RID[].tCutOff	与えられたRIDに対する最大充電時間。
RID[].IcutOff	与えられたRIDに対する充電終了電流。
NTC[]	温度参照表。

5.2. ｺﾝﾊﾟｲﾙ

ｺｰﾄﾞをｺﾝﾊﾟｲﾙする前に以下の構成設定が行われるべきです。

表5-2. ｺﾝﾊﾟｲﾗ構成設定

分類	ﾀﾌﾞ	領域	値
General Options	Target	Processor configration	ATtiny861 (注)
		Memory model	Small
	System	Data stack	0x40
		Return address stack	24
		Enable bit definitions ...	ﾁｪｯｸ
C/C++ Com+iler	Language	Require prototypes	ﾁｪｯｸ
Linker	Output	Format	Other: ubrof8
	Extra Options	Command Line	-y(CODE) -Ointel-extended,(DATA)=EXE_DIR¥$PROJ_FNAME$_data.hex -Ointel-extended,(XDATA)=EXE_DIR¥$PROJ_FNAME$_eeprom.hex

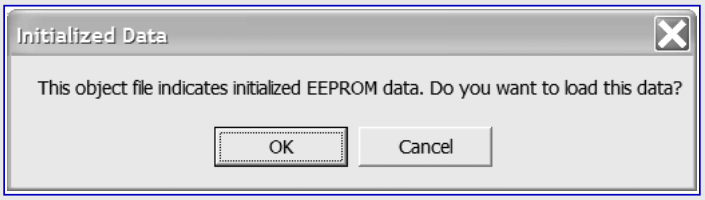
注: 他の選択も可能です。より多くの情報については3頁の3.1.1.項をご覧ください。

5.3. ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ

ｺﾝﾊﾟｲﾙしたｺｰﾄﾞはAVR StudioとJTAICEmk Ⅱのような選りすぐりのﾃﾞﾊﾞｯｶﾞまたは書き込みﾂｰﾙを用いて便利に目的対象ﾃﾞﾊﾞｲｽへ書かれます。

ｿﾌﾄｳｪｱが動くために、EEPROMﾃﾞｰﾀを含むｺﾝﾊﾟｲﾙされたｺｰﾄﾞが目的対象に格納され(書かれ)なければならないことに注意してください。EEPROM内容が格納されるべきかをAVR Studioが尋ねる時にOKで答えてください。これは下図で図解されます。

図5-1. EEPROMへの初期化されたﾃﾞｰﾀ格納



ﾌﾟﾛｸﾞﾗﾑは内部発振器の使用とｸﾛｯｸ信号が前置分周されないことを期待します。正しいﾌﾟﾛｸﾞﾗﾑ実行を保証するために、いくつかのﾋｭｰｽﾞﾋﾞｯﾄが設定されなければなりません。既定から外れるﾋｭｰｽﾞ設定が右図で一覧にされます。

表5-3. 既定でないﾋｭｰｽﾞﾋﾞｯﾄ設定

ﾋｭｰｽﾞﾋﾞｯﾄ	設定	説明
CKDIV8	1(非ﾌﾟﾛｸﾞﾗﾑ)	8分周を行わない
CKSEL3～0	0010	内部発振器を使用

6. 既知の制限

ここは一覧にされた設計の既知の制限です。

6.1. 電池存在検出

現在、電池の存在は各電池ｺﾈｸﾀ上の電圧を測定することによって検出されます。完全に放電された電池の場合に、回路要素を検知するために継電器がそれを接続すると直ぐに電圧が0Vへ落ち込み、充電器がそれらを検出しない原因になります。

1つの解決策は電圧を僅かに上げるために、約0.1Cの電力供給で数分間このような電池を充電することです。

また、充電電圧を印加することも可能で、電流が流れない場合に不在として合図します。

完全に放電した電池、特にそれらが複数ｾﾙの電池の場合に、それらを損傷するかもしれないことに注意してください。

6.2. 電流ﾋｽﾃﾘｼｽと電圧低下

只1ｾﾙまたは2ｾﾙを充電する時に、電圧低下のため、電流ﾋｽﾃﾘｼｽは満充電検出を誤らせるかもしれません。ﾋｽﾃﾘｼｽを狭めることがこれを助けますが、降圧変換器出力が無限の分解能でないことを覚えて置いてください。

6.3. 電池電流測定

電池電流は非常に低い抵抗値を持つ分圧抵抗器を用いて感知されます。これは測定される信号に於いて雑音が容易に拾われ、非常に低い振幅を持つ雑音でさえ、測定を乱すかもしれないことを意味します。救済策として、測定された電池電流は4採取に渡って平均化されます。

更に、1または2LSBの程度の変動に出会うのは珍しくありません。既定では(3.1.3.項をご覧ください)、これが7または14mAの測定誤差を意味します(ADC.cﾌｧｲﾙのScale()関数をご覧ください)。実際問題として、これは早まった充電周期の終りに帰着するかもしれません。

提案される解決策は分圧抵抗(R410:より大きな方が良い)と抵抗分圧器(R400～410,R427,R428,R446,R447)の値を最適化することです。

6.4. RID感知

電池識別抵抗はPA2(ADC2)ﾋﾟﾝ経由で感知されます。この線上の既定ﾌﾟﾙｱｯﾌﾟ抵抗(ATAVRBC100電池充電器参照基準設計のR305)は4.7kΩです。これは感知抵抗値を最大約14.7kΩに制限します。

Varta PlLiFlex電池使用時、これは信頼に足る感知ができる最大電池容量が1000mAhであることを意味します。より大きな感知抵抗/電池容量については、BC100上のﾌﾟﾙｱｯﾌﾟ抵抗が変更されなければなりません。加えて、ｿﾌﾄｳｪｱは新しいﾌﾟﾙｱｯﾌﾟ抵抗値を反映するように更新されなければなりません。

6.5. 降圧充電器

降圧充電器(と供給電圧)の選択は最小充電電流をどの位低くできるかの限度を設定します。より高い供給電圧とより小さい高圧切替ｲﾝﾀﾞｸﾀは最小充電電流をより高くするでしょう。これは或る構成設定が早まった充電周期の終りに帰着するかもしれないことを意味します。

救済策は低い供給電圧と大きなｲﾝﾀﾞｸﾀを持つ降圧切り替えを使うことです。

7. 参考文献

1. "What's the best battery?" – Battery Universityから2007年8月1日に取得:
   http://www.batteryuniversity.com/partone-3.htm
2. "AVR451:BC100ﾊｰﾄﾞｳｪｱ使用者の手引き" – Atmelのｳｪﾌﾞｻｲﾄから入手可能:
   http://www.atmel.com/products/avr/
3. "ATtiny261/461/861ﾃﾞｰﾀｼｰﾄ" – Atmelのｳｪﾌﾞｻｲﾄから入手可能:
   http://www.atmel.com/products/avr/
4. "ATtiny25/45/85ﾃﾞｰﾀｼｰﾄ" – Atmelのｳｪﾌﾞｻｲﾄから入手可能:
   http://www.atmel.com/products/avr/
5. "Duracell Ni-MH Rechargeable Batteries Technical Bulletin" – Duracellから2007年8月1日に取得:
   http://duracell.com/oem/Pdf/others/TECHBULL.pdf
6. "Handbook of Batteries, Third Edition" – McGraw-Hillから.
   http://www.mhprofessional.com/product.php?isbn=0071359788
6. "Charging nickel-based batteries" – Battery Universityから2007年8月1日に取得:
   http://www.batteryuniversity.com/partone-11.htm

本社

Atmel Corporation
2325 Orchard Parkway
San Jose, CA 95131
USA
TEL 1(408) 441-0311
FAX 1(408) 487-2600

国外営業拠点

Atmel Asia
Unit 1-5 & 16, 19/F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
Hong Kong
TEL (852) 2245-6100
FAX (852) 2722-1369

Atmel Europe
Le Krebs
8, Rue Jean-Pierre Timbaud
BP 309
78054 Saint-Quentin-en-
Yvelines Cedex
France
TEL (33) 1-30-60-70-00
FAX (33) 1-30-60-71-11

Atmel Japan
104-0033 東京都中央区
新川1-24-8
東熱新川ﾋﾞﾙ 9F
ｱﾄﾒﾙ ｼﾞｬﾊﾟﾝ株式会社
TEL (81) 03-3523-3551
FAX (81) 03-3523-7581

製品窓口

ｳｪﾌﾞｻｲﾄ
www.atmel.com

技術支援
avr@atmel.com

販売窓口
www.atmel.com/contacts

文献請求
www.atmel.com/literature

本応用記述はAtmelのAVR463応用記述(doc8098.pdf Rev.8098A-09/07)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。