

AVR103 : EEPROMプログラミング動作種別の使用法

要点

- EEPROM入出力説明
- 50%までに削減されたEEPROM更新時間
- 過去互換のコード

序説

この応用記述はATmega48/88/168, ATtiny13, ATtiny2313, ATmega256xのようないくつかのAVRデバイスのEEPROMに対して使用可能なプログラミング動作種別を利用するドライバ実装を与えます。このドライバはデータをEEPROMに書く時に最も効率的なプログラミング動作種別を常に使用することによって、電力と時間の両方を節約します。これはEEPROMセルがそれを必要とされる時にだけ消去と書き込みをされることを保証します。

EEPROMの消去と書き込みの操作の詳細な内容はプログラミング動作種別とそれらの恩恵を理解するために与えられます。いくつかの計画では、多くの応用、例えばEEPROM内へのフラグやパラメータの記憶で、本応用記述内のドライバ使用は重要な性能向上をすることができます。

動作理論

EEPROM入出力説明

EEPROMは単一ビットを表す各々の独立したセルから成ります。このセルはフローティングゲートトランジスタ技術に基きます。トランジスタのゲート上に捕らえられた電荷がセルの論理レベルを決めます。次の方が僅かに平易でしょう。セルの動きは以下のように説明できます。セルを消去すると、ゲートに充電が行われてセルは論理1として読まれます。セルのプログラミング(0書き込み)は論理レベルを0に持って来る、ゲート放電と当価です。消去(充電)されているセルのプログラミング(放電)だけが可能です。

例えEEPROMがバイトで入出力されるとしても、ビットが個別にプログラミングできることに注意してください。プログラム(0)されるべきビットだけが放電されるため、残りの非プログラム(1)ビットは未だ充電されています。どの非プログラム(1)ビットも後の段階でプログラム(0)にできます。従って、既に書かれたことのあるバイトを、その間での消去なしで書くことは、旧値と新値間のビット単位ANDの結果になるでしょう。

EEPROMプログラミング動作種別

選択可能なEEPROMプログラミング動作種別を持たないAVRデバイスでは、'書き込み'操作が実際には組み合わせられた消去とプログラムの操作です。書かれるべきメモリバイトは、例え消去されるべき必要があるビットがなくても、新しい値をプログラミングするのに先立って常に消去されます。例えば、値\$00を既に値\$0Fを含むEEPROMバイトにプログラミングするとき、最初にバイトを消去する必要がありません。0から1に変更されるべき必要なビットがありません。この場合の消去動作は時間と電力の消費を除いて何の影響もありません。組み合わせられた消去/プログラム操作は単独プログラミング操作時間の2倍かかります。

選択可能なEEPROMプログラミング動作種別を持つAVRデバイス、例えばATmega48/88/168で設計するとき、EEPROM書き込みを独立した消去とプログラムの操作に分割することが可能です。これは消去が必要とされない場合でEEPROM更新時間を50%削減するでしょう。利用可能なプログラミング動作種別は以下の表1.で一覧されます。

本資料では語'プログラム'が消去なしでの更新を記述するために使用され、'更新'はEEPROM内容を変更する何れかの操作に対して使用されます。

表1. EEPROMプログラミング動作種別

プログラミング動作種別	動作	代表的な更新時間
0	組み合わせられた消去とプログラム	3.4ms
1	消去のみ	1.8ms
2	プログラムのみ	1.8ms

プログラミング動作種別はEEPROM制御レジスタ(EECR)のEEPROM1とEEPROM0ビットを使用して設定されます。始動での既定動作種別は動作種別0で、従ってレジスタはEEPROMプログラミング動作種別を使用しないソフトウェアと互換です。



8-bit **AVR**[®]
マイクロコントローラ

応用記述

暫定

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、ATMEL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。

効率的なEEPROM入出力

プログラミング動作種別からの利益を得るため、EEPROMソフトウェアは必要とされる操作だけが実行されるように、この動作種別を使用すべきです。例えば、EEPROMのバイトを更新するとき、0から1に変更すべき必要のビットがなければ、消去の必要はありません。

EEPROMのバイトを更新するとき、ソフトウェアは新しい値のビットに対して既存の値のビットを比較して、どのプログラミング動作種別を使用するかを決めます。書かれる値に依存してどの動作種別を使用するかを決めるのに使用される追加のCPU周期は、平均更新時間の削減に関してより重要です。

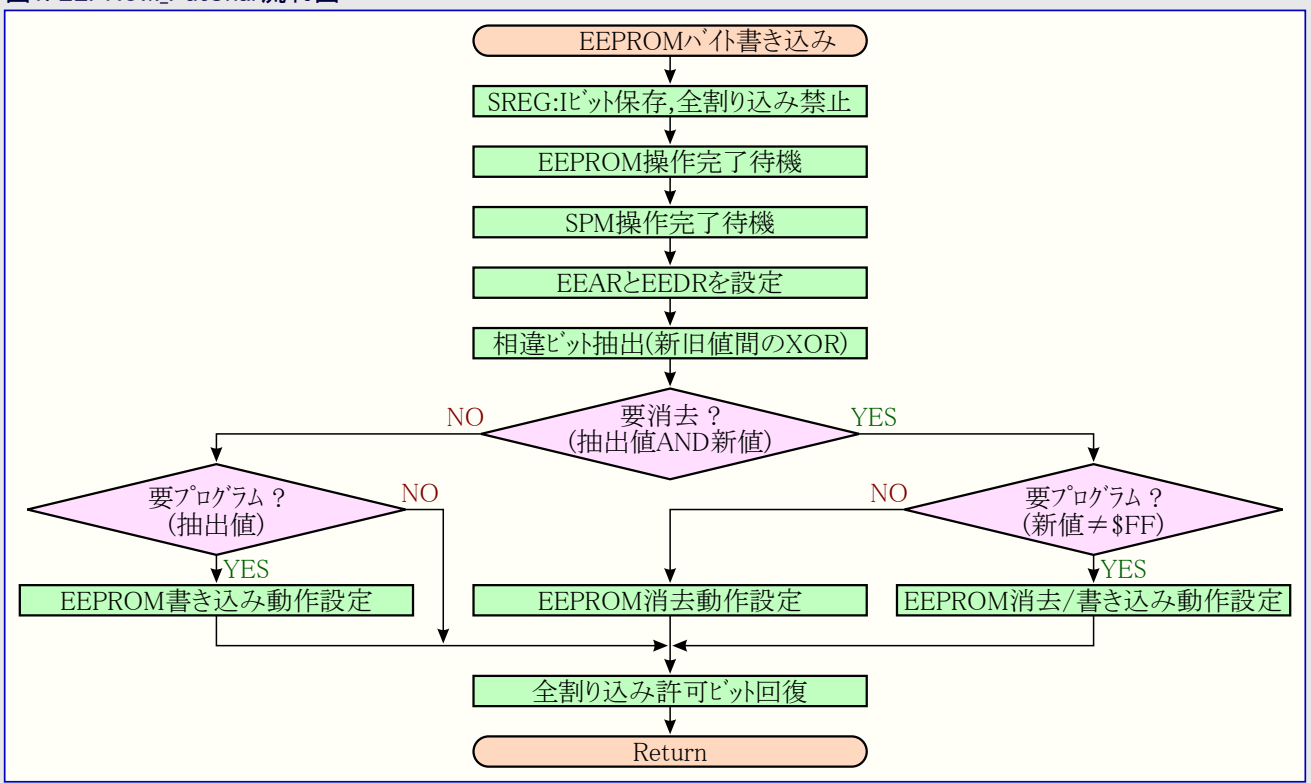
実装

この応用記述はプログラミング動作種別を使用する方法を示すための小さな試験応用を提供します。このソースコードはEEPROMの読み込みと更新の関数を実装し、各種プログラミング動作種別を実演するために幾らかのEEPROM更新を実行します。EEPROM更新関数だけが詳細に記述されます。

EEPROM_PutChar()関数

これはEEPROM更新関数です。これは新旧バイトを比較して更新操作のための最も効率的なプログラミング動作種別を使用します。この流れ図は以下の図1.で示されます。

図1. EEPROM_PutChar流れ図



代わりの近接手法は新旧値間でビット単位ANDを実行することでしょう。この結果が新しい値に等しければ、'プログラム'操作だけが必要とされます。異なる場合は組み合わされた消去と'プログラム'動作種別を使用してください。これは或る程度のCPU周期とコード容量を節約しますが、新旧値が等しい時、または消去操作だけが必要とされる時の場合からの恩恵がないでしょう。

実行時間

最も一般的な場合に関して、フラッシュメモリの自己プログラミング操作がこの実装で考慮されています。けれども、Cコンパイラの`#define`擬似命令を使用することで、SPMに関するコード部分を省けます。より多くの詳細については注釈されたソースコードを参照してください。

下の表2.では一般的な場合に対する実行時間が示されます。最大速度最適化設定に対する実行時間も示されます。この値は関数の呼び出しと復帰の命令を含み、直前のEEPROMとSPMの書き込みが完了されていたと仮定します。'旧EEPROM書き込み'操作はデータシート内の例として得られるプログラミング動作種別を使用しない標準的なEEPROM書き込みルーチンへの参照です。

表2. 実行時間

操作	非最適化	最大速度最適化
消去のみ	41	35
書き込みのみ	41	35
消去と書き込み	41	35
無操作	36	30
旧EEPROM書き込み	26	24

注: ・ 値は直前のEEPROMとSPM書き込みが完了されていたとの仮定です。
 ・ 総実行時間については表1.内の値を参照してください。

他応用内でのEEPROM_PutChar()使用

EEPROM_PutChar()の実装はプログラミング動作種別を使用しない既存実装を置換できます。この関数は最良のプログラミング動作種別選択を内部的に処理します。例えば「AVR104: 緩衝された割り込み制御EEPROM書き込み」応用記述は、本応用記述からのEEPROM_PutChar()コードでAVR104の図4.内の「EEMWE設定」と「EWE設定」部分を置き換えることにより、プログラミング動作種別を使用することで強化できます。

多くの応用では、EEPROMが使用される方法を注意深く計画することによって、重大な恩恵を得られます。できるなら、0から1への変更を避けることを試みるべきでしょう。例えば、EEPROM内のパラメータ記憶は、「AVR105: フラッシュメモリでの電力効率的な高耐久性パラメータ記憶」応用記述で利用されるフラッシュメモリと同じ方法の循環緩衝部で強化できます。フラッシュメモリのセルの消去と'プログラム'はEEPROMのセルと非常に似ています。

参考文献

- ・ ATmega48/88/168/328データシート
- ・ AVR104: 緩衝された割り込み制御EEPROM書き込み
- ・ AVR105: フラッシュメモリでの電力効率的な高耐久性パラメータ記憶
- ・ “Application-Specific Integrated Circuits” Michael John Sebastian Smith著



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
東熱新川ビル 9F
アトメル ジャパン株式会社
TEL (81) 03-3523-3551
FAX (81) 03-3523-7581

製造拠点

Memory

2325 Orchard Parkway
San Jose, CA 95131, USA
TEL 1(408) 441-0311
FAX 1(408) 436-4314

Microcontrollers

2325 Orchard Parkway
San Jose, CA 95131, USA
TEL 1(408) 441-0311
FAX 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3
France
TEL (33) 2-40-18-18-18
FAX (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards

Zone Industrielle
13106 Rousset Cedex
France
TEL (33) 4-42-53-60-00
FAX (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
TEL 1(719) 576-3300
FAX 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR
Scotland
TEL (44) 1355-803-000
FAX (44) 1355-242-743

RF/Automotive

Theresienstrasse 2
Postfach 3535
74025 Heilbronn
Germany
TEL (49) 71-31-67-0
FAX (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
TEL 1(719) 576-3300
FAX 1(719) 540-1759

Biometrics

Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex
France
TEL (33) 4-76-58-47-50
FAX (33) 4-76-58-47-60

文献請求

www.atmel.com/literature

© Atmel Corporation 2005.

ATMEL製品は、ウェブサイト上にあるATMELの定義、条件による標準保証で明示された内容以外の保証はありません。本製品は改良のため予告なく変更される場合があります。いかなる場合も、特許や知的技術のライセンスを与えるものではありません。ATMEL製品は、生命維持装置の重要部品などのような使用を認めておりません。

本書中の®、™はATMELの登録商標、商標です。

本書中の製品名などは、一般的に商標です。

© HERO 2014.

本応用記述はATMELのAVR103応用記述(doc2578.pdf Rev.2578A-03/05)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。