

AVR316 : TWI部を使うSMBus従装置

要点

- 9つの異なるSMBus規約を支援
- パケット異常検査(PEC)
- 割り込み駆動SMBus従装置ドライバ
- 全支援規約の実演での簡単な実装

1. 序説

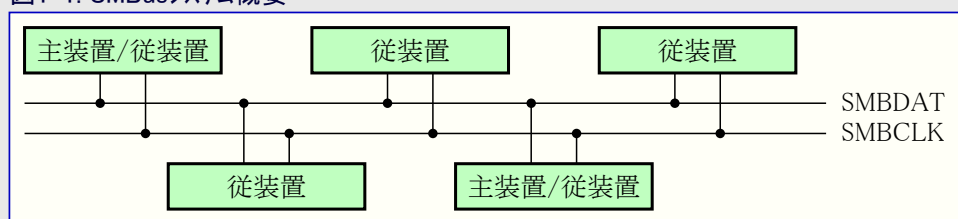
システム管理バス(SMBus)は装置がシステムと電力管理の情報を交換するための通信をできるバスにするためにI²Cバスの原則を使います。

SMBusはスマート バッテリ、温度制御、その他の低帯域システム管理通信に関して主にパーソナルコンピュータで使われます。

SMBus仕様改訂2.0は装置への動的アドレス割り当てに使われるアドレス解決規約(ARP)に加え、装置間通信に関して11の規約を載せています。各規約は基本的にI²C命令の連続です。従装置は手元の作業に関して比べた実装を考慮に入れて、定義された規約の完全な1式または部分1式を実装することができます。

この応用記述はSMBus仕様とAVRのTWI単位の背景情報と、割り込み駆動SMBus従装置ドライバ試供実装を提供します。

図1-1. SMBusシステム概要



2. SMBus仕様

SMBus仕様は2線バスで多数の装置間での通信を可能にするためにI²Cの原則を基にしていますが、電気的パラメータとタイミングパラメータの両方でいくつかの重要でない小さな違いもあります。

2.1. SMBusとI²C間の違い

以降の比較はSMBus2.0低電力装置に対してです。

まず最初に、I²CはVCCの百分率として入力電圧を定義し、一方SMBusは固定の入力電圧レベルで動きます。標準動作のI²CとSMBusに対する入力電圧レベルの最小と最大の定格が表1で示されます。

表1. I²CとSMBusに関する入力電圧レベル

| | I ² C(標準動作) | | SMBus | |
|-----------------|------------------------|-------------------------|-------|-------|
| | Min | Max | Min | Max |
| V _{IL} | -0.5 V | 0.3 VCC | - | 0.8 V |
| V _{IH} | 0.7 VCC | VCC _{max} +0.5 | 2.1 V | 5.5 V |

AVRは5Vに代わる3Vの供給電圧に於いて、SMBusの最低V_{IH}に最も良く対応します。

I²Cは最大バス容量(C_b)での制限を定めています。SMBus仕様はこのような制限を持ちませんが、100~350μAの範囲のプルダウン電流が必要です。

バス信号の上昇と下降の時間がSMBus仕様で定義されています。I²C仕様は最大バス容量(C_b)だけを定義し、上昇と下降の時間を定義していません。

I²Cバスに接続する各装置の最大漏れ電流は10μAとして指定されていますが、一方SMBusに関して対応する必要条件は5μAです。

SMBusは(バスがアイドルの時を除き、)10kHzの最小バス クロック周波数が必要です。I²Cはこのような必要条件を持ちません。SMBusの最大周波数は標準動作に於けるI²C動作の最大速度に等しい100kHzです。



8ビット AVR[®]
マイクロコントローラ

応用記述

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。

Rev. 2583A-10/05, 2583AJ3-04/21

SMBusはSMBCLK(SCL)の下降端後300ns間無変化に留まるSMBDAT(SDA)が必要です。I²C仕様によってこのような必要条件是全く課されていません。

SMBusはクロックLow時間の最大拡張で多くの制限を持ちます。I²Cはバス全体を封鎖する悪い装置を許す、任意の時間長への伸長をクロックLow時間に許します。

SMBusは装置が他の何の作業を実行中かにかかわらず、常にその従装置アドレスに応答することを装置に要求します。I²Cはこのような必要条件を持ちません。

SMBusは従装置が多忙、または異常が起きたことを示すのにNACK信号を用います。例えそれが転送の最終バイトであっても、従装置は各バイトの転送後にNACK信号を生成できなければなりません。

2.2. パケット誤り符号(PEC)

SMBus仕様はパケット誤り符号と呼ばれるCRCに基づく誤り検出法を含みます。SMBus装置はPECの使用を必要とされませんが、PEC能力を持つどの装置もPECを実装しない他のSMBus装置と通信ができなければなりません。

PECは多項式 X^8+X^2+X+1 で8ビットCRCとして計算されます。PECのCRCは送受信されるビット順で計算されます。SLA+R/Wを含む規約の全バイトがこの計算に含まれなければならないが、一方開始条件、再送開始条件、停止条件、ACK、NACKの制御信号は除外されます。

PECに対して使われるCRC計算の実装には多くの方法があります。CRCは移動とXOR操作を使って計算することができ、または参照表を利用することもできます。完全な参照表は最大256バイトのフラッシュメモリを使いますが、CRC演算は数クロック周期だけで実行されます。計算法は非常に少ないメモリ必要条件ですが、参照法よりも大きな実装になり、もっと遅く実行します。

2.3. SMBus規約

SMBus仕様2.0は装置間通信用に11の規約を定義しています。各装置は完全な1式または部分的な1式の規約を実装することができます。装置が規約を実装しない場合、操作の結果は未定義にされます。

各規約はI²C転送形式の厳密な連続で構成されます。SMBus規約は連続構成図によって表されます。構成図に於ける各種記号の意味は表2.で一覧にされます。

加えて、領域の直上に置かれた数値はその領域のビット幅(数)を指定し、領域の直下に置かれた数値はその領域がこの値を持つ必要があることを指定します(訳補:実際には直下の数値は使われていません)。

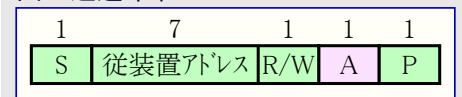
表2. SMBus規約構成図凡例

| 記号 | 意味 |
|--------------|---------------------------|
| S | 開始条件 |
| Sr | 再送開始条件 |
| R | 読み込み(1) |
| W | 書き込み(0) |
| A/ \bar{A} | 応答(ACK)(0)または無応答(NACK)(1) |
| P | 停止条件 |
| PEC | パケット誤り符号 |
| | 主装置→従装置 |
| | 従装置→主装置 |

2.3.1. 迅速命令

“迅速命令”規約はR/Wビットを通じて、従装置へ1ビットを送るのに使うことができます。

図2. 迅速命令



2.3.2. バイト送付

“バイト送付”規約は従装置へ1バイトのデータを送るのに使うことができます。

図3. バイト送付

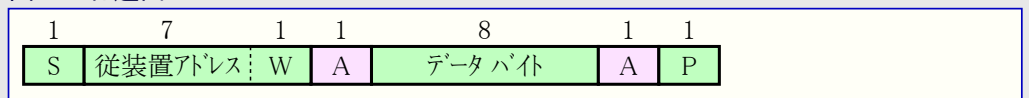
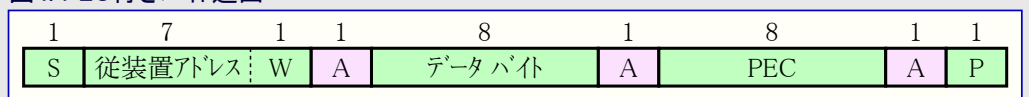


図4. PEC付きバイト送付



2.3.3. バイト受取

“バイト受取”規約は従装置からの1バイトのデータを要求するのに使うことができます。

図5. バイト受取

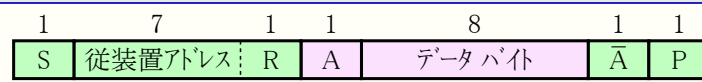
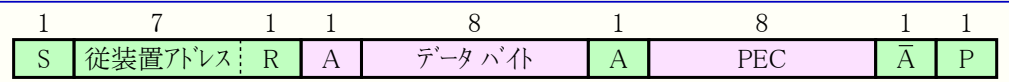


図6. PEC付きバイト受取



2.3.4. バイト/語書き込み

“バイト/語書き込み”規約は1つの命令符号と1つのバイト/語データを送るのに使うことができます。命令符号は例えばデータを配置するためのメモリ位置を指定することができます。

図7. バイト書き込み

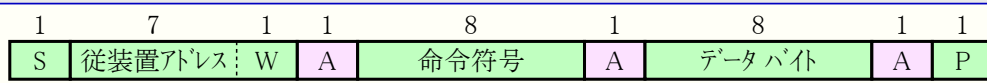


図8. PEC付きバイト書き込み

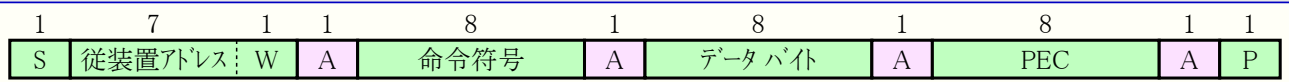


図9. 語書き込み

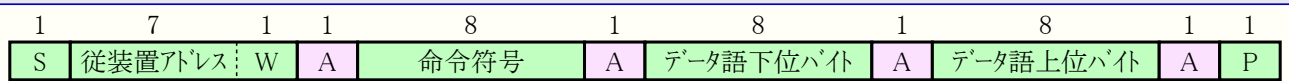
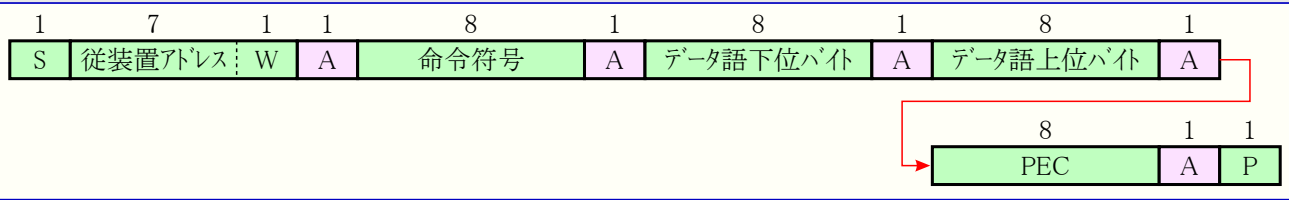


図10. PEC付き語書き込み



2.3.5. バイト/語読み込み

“バイト/語読み込み”規約は従装置から1つのバイト/語データを読むのに使うことができます。読むデータは命令符号により、例えば従装置の内部レジスタをアドレス指定することによって制御することができます。

図11. バイト読み込み

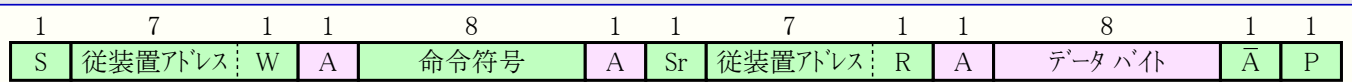


図12. PEC付きバイト読み込み

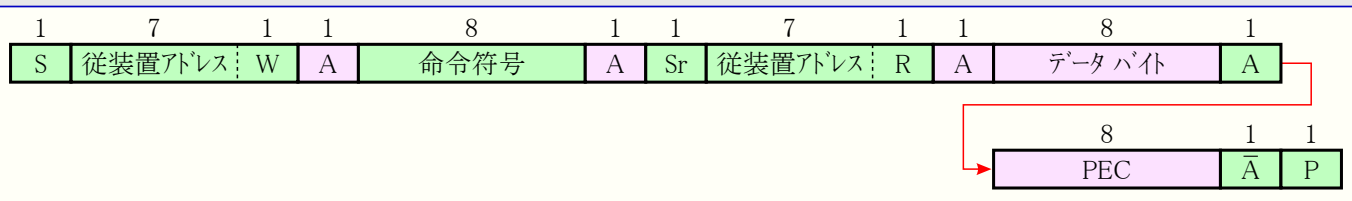


図13. 語読み込み

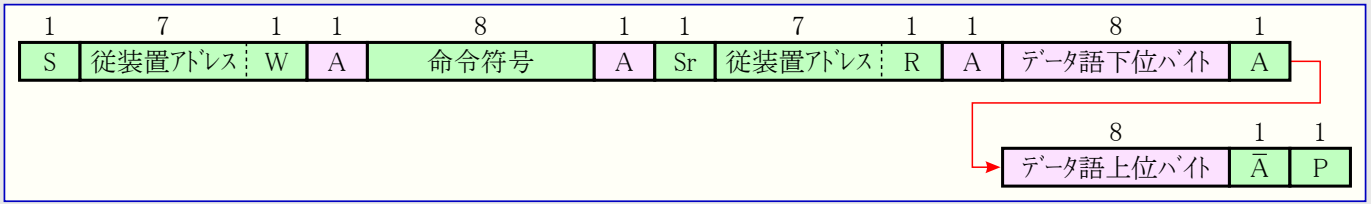
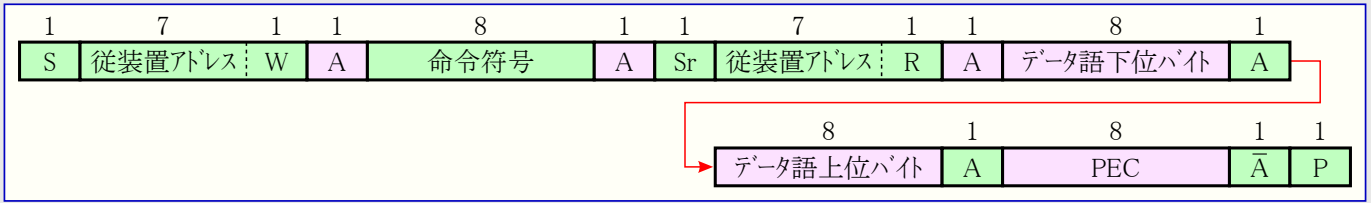


図14. PEC付き語読み込み



2.3.6. 処理呼び出し

“処理呼び出し”規約は命令符号と2バイトのデータの両方を従装置へ送り、そして2バイトのデータを受け取ることを主装置に許します。これは例えば1つの命令で従装置上の関数を引数付きで呼び出して戻り値を受け取るのに使うことができます。

図15. 処理呼び出し

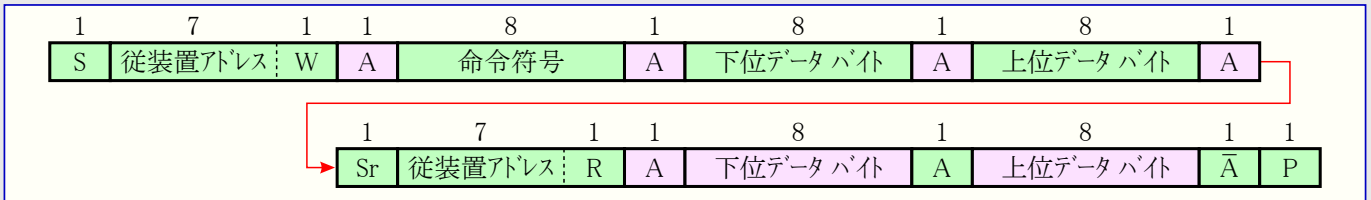
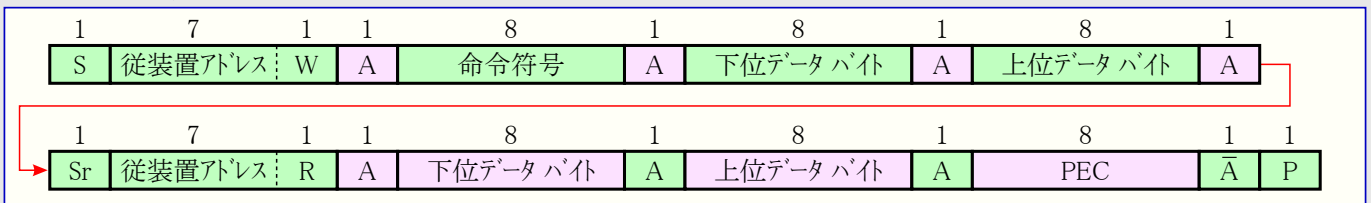


図16. PEC付き処理呼び出し



2.3.7. 塊読み書き

“塊読み書き”規約は従装置との32バイトまでのデータの送付/受け取りに使うことができます。命令符号と読み書きバイト数が最初に送信されます。バイト数は1~32の範囲でなければならず、PECを含みません。

図17. 塊書き込み

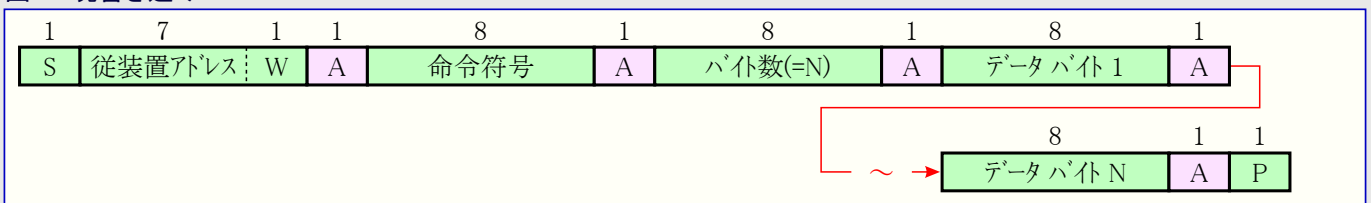


図18. PEC付き塊書き込み

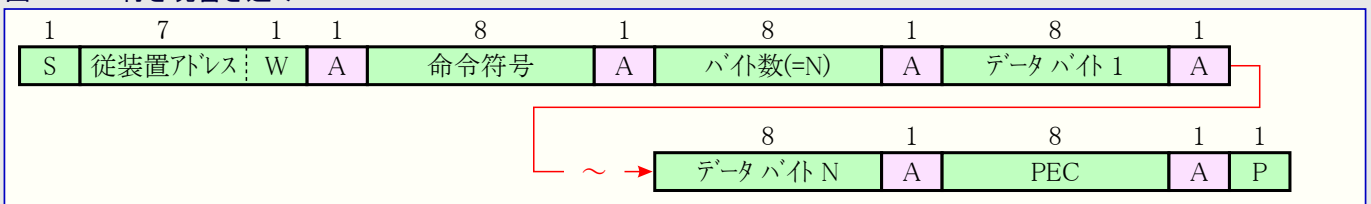


図19. 塊読み込み

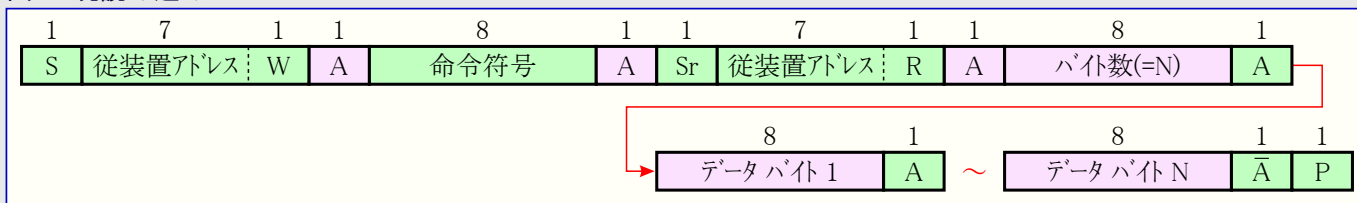
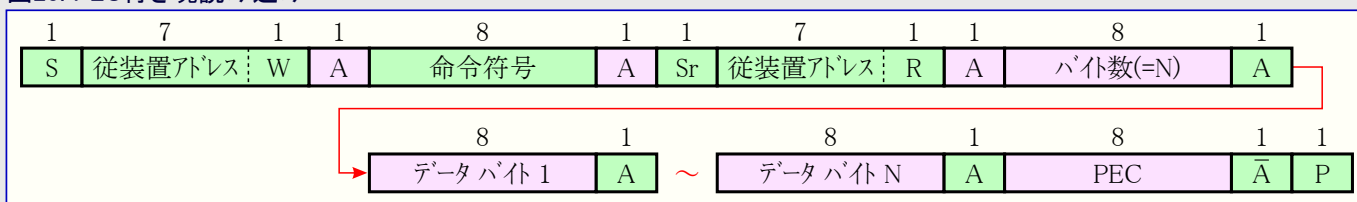


図20. PEC付き塊読み込み



2.3.8. 塊書き読み処理呼び出し

“塊書き読み処理呼び出し”規約はSMBus 2.0版で導入され、1つの操作で従装置へ命令符号と最大32バイトを送り、従装置から最大32バイトを受け取るのに使うことができます。これは“処理呼び出し”と同じように用いることができますが、引数と戻り値の長さが可変です。

図21. 塊書き読み処理呼び出し

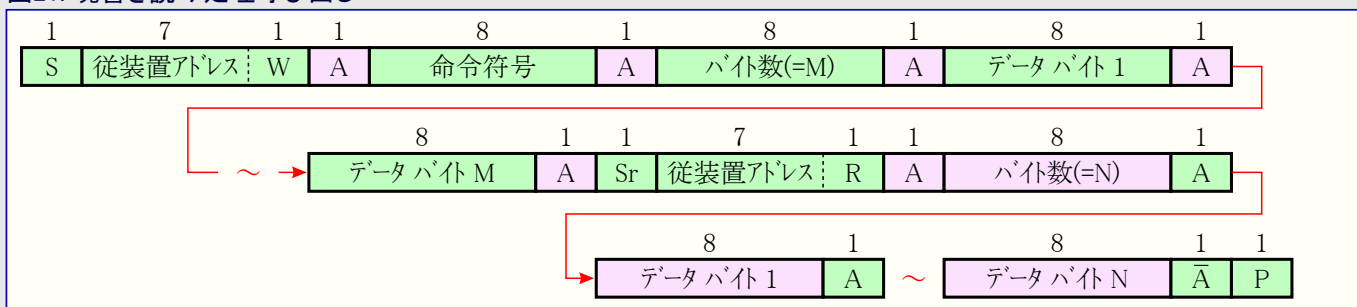
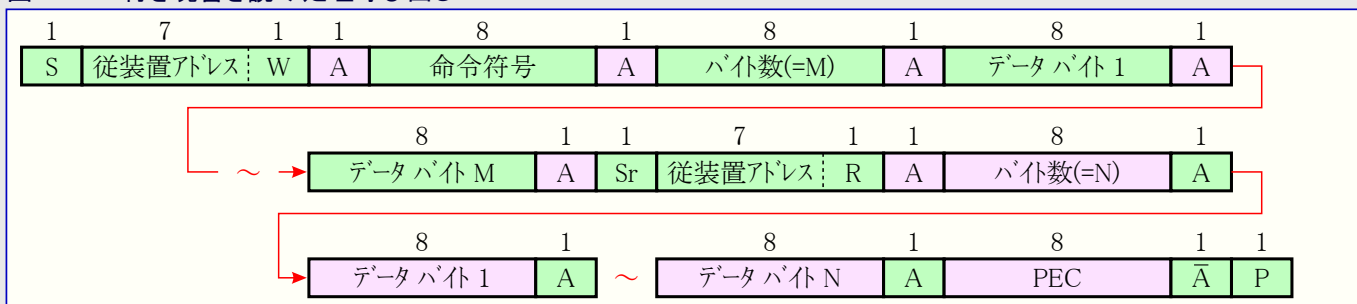


図22. PEC付き塊書き読み処理呼び出し



2.3.9. SMBusホスト通知規約

“SMBusホスト通知”規約は装置がホストになることを欲する時に使われます。ここで従装置応用が考察されるだけなので、SMBusホスト通知規約は必要とされません。

2.4. アドレス解決規約(ARP)

アドレス解決規約はSMBus 2.0版で導入され、バス上の各従装置への動的な固有アドレス割り当てに使われます。ARPは自身のアドレスとSMBus装置既定アドレスの両方を聴取する従装置を必要とします。SMBus装置既定アドレスは一斉呼び出しアドレスと違います。AVR TWI単位部はそれが一斉呼び出しアドレスでなければ2つの異なる従装置アドレスを聴取するように設定することができません。従ってARPは支援されません。

3. AVRマイクロ コントローラでのSMBus実装

多くのAVRで見られるTWI単位部が本応用記述に於けるSMBusドライバ実装に用いられます。TWI単位部の操作のための序説がここで与えられます。TWI単位部の全部の説明についてはデバイスのデータシートを調べてください。

3.1. TWI単位部

従装置動作に於けるTWI単位部の基本的な操作は次のとおりです。

AVRの従装置アドレスはTWIアドレスレジスタ(TWAR)に格納されます。TWI論理回路は進行中の通信のアドレスがTWARと一致するかどうかを判断するために、バス上に送信されたアドレスを装置に格納されたアドレスと比較します。

その決定が進行中の通信を継続させなければならない時は常にTWI割り込み要求(TWINT)フラグが設定(1)されます。そして割り込みの理由を判断するためにTWI状態(TWSR)レジスタが調べられなければなりません。そしてプログラムは何が起きたかに基いて或る動きを取ることができます。最後に、通信を継続または停止するために、TWI制御(TWCR)レジスタが操作されなければなりません。通信が終了されるまで、これが何度も何度も起きます。従装置動作に関連するTWSRの状態符号が表3.で一覧にされます。

表3. TWSR状態符号

| 状態符号 | TWIの状態 |
|------|--|
| \$60 | 自宛SLA+W受信, ACK応答 |
| \$68 | 主装置としてSLA+R/Wで調停喪失(敗退), 自宛SLA+W受信, ACK応答 |
| \$70 | 一斉呼び出し受信, ACK応答 |
| \$78 | 主装置としてSLA+R/Wで調停喪失(敗退), 一斉呼び出し受信, ACK応答 |
| \$80 | 自宛SLA+Wアドレス指定状態で、データ受信, ACK応答 |
| \$88 | 自宛SLA+Wアドレス指定状態で、データ受信, NACK応答 |
| \$90 | 一斉呼び出しアドレス指定状態で、データ受信, ACK応答 |
| \$98 | 一斉呼び出しアドレス指定状態で、データ受信, NACK応答 |
| \$A0 | 未だ従装置としてアドレス指定状態で、停止条件または再送開始条件を受信 |
| \$A8 | 自宛SLA+R受信, ACK応答 |
| \$B0 | 主装置としてSLA+R/Wで調停喪失(敗退), 自宛SLA+R受信, ACK応答 |
| \$B8 | TWDR内のデータバイト送信, ACK受信 |
| \$C0 | TWDR内のデータバイト送信, NACK受信 |
| \$C8 | TWDR内の最終データバイト送信(TWEA='0'), ACK受信 |
| \$F8 | 関連状態情報なし |
| \$00 | 不法な開始条件または停止条件のためのバス異常 |

TWI単位部でSMBusを実装する時に注意すべきいくつかの実装の詳細があります。

- AVRが従装置受信部として動くとき、データを受信する前にそのデータに対してACKまたはNACKのどちらかを決めなければなりません。例えば停止条件または再送開始条件が予期される時にデータが受信された場合、従装置が異常を検知した時にACKが既に送られてしまっています。同様に、これはPEC誤りを合図するためのACKとNACKの使用をAVRに関して重要にさせます。SMBus仕様はこれを行うことを従装置に要求しておらず、PECの照合はより高位の網規約層で行い得ることを示唆しています。
- 停止条件または再送開始条件が受信されたことをTWSRが示すとき、どれで起こったかを知る方法がありません。これはバイト送出またはバイト読み込み、語読み込みまたは塊読み込みのどちらが進行中かを知ることを難しくします。
- 従装置送信部として動いている時にPECが使われるべきか否かを予め知らない場合、データバイトに対する返答でNACKを予期する時を知る方法がありません。
- AVRのTWI単位部は迅速命令をあまり上手く支援しません。

上の点のどれでもないものがAVRでSMBus従装置を実装する時の主要な問題です。SMBusホストは誤りなしの通信完了かどうかを決めるのにACK/NACKだけを信頼すべきです。単一ビットのACK/NACKは雑音に敏感です。PECは支援される必要がなく、またこの処理はより高位の規約で達成することができます。従装置実装は実装に対してどの規約を選ぶかが自由です。主装置には正しい命令を発行することと、未支援規約の結果がSMBus仕様に従って未定義にされることに関して責任があります。

3.2. 提唱実装

前の検討に基き、多方面の実装がここで提案されます。実装はAVRで未だあいまいになり得る規約を避け、何の種類の規約が進行中かを判断するのに必要なコードを最小にします。その上更に必要ならばPECを支援することができます。提唱実装は以下のSMBus規約を支援します。

- ・ バイト受取
- ・ バイト書き込み
- ・ 語書き込み
- ・ バイト読み込み
- ・ 語読み込み
- ・ 塊書き込み
- ・ 塊読み込み
- ・ 処理呼び出し
- ・ 塊書き読み処理呼び出し

この削減した規約1式は何の規約が進行中かの判断を非常に容易にします。“バイト受取”だけがSLA+Rで始まる規約です。他の全ての規約は従装置への最初のデータバイトとして送られる命令符号を含みます。命令符号は多くの方法で利用することができます。1つの方法は規約が常に命令符号から配給され得るような方法で、操作毎に対して固有の命令符号を持つことです。代わりに、命令符号はレジスタやメモリ位置のアドレスを指定することもできます。

3.3. 含まれているドライバと試供応用の説明

この応用記述と共に含まれている試供ドライバと応用に於いて、命令符号はSMBus規約と必要とする従装置動作の両方を固有に認識します。

実際の一般的なSMBus実装はただ緩衝部での送受信を処理し、主プログラムとの通信にフラグを使うことができます。けれども、代表的なSMBus実装ではレジスタ値の設定や状態情報の読み戻しにSMBus命令が用いられます。各命令の正確な意味は予め適切に定義されます。これに留意し、含まれているSMBusドライバはTWI割り込みルーチンに実装され、ドライバを全体的に自律型にします。試供応用の機能が表4.で示されます。

表4. 命令符号割り当て

| 命令符号 | 動作 | 規約 |
|------|----------------------------|-------------|
| なし | スイッチ押下(PIND)状態読み込み | バイト読取 |
| \$10 | 供給者識別文字列読み込み | 塊読み込み |
| \$20 | スイッチ押下(PIND)状態読み込み | バイト読み込み |
| \$30 | EEPROM位置指示子設定 | 語書き込み |
| \$40 | EEPROMデータバイト読み込み | バイト読み込み |
| \$41 | EEPROMデータ語読み込み | 語読み込み |
| \$50 | LED(PORTB)へバイト出力 | バイト書き込み |
| \$51 | LED(PORTB)へ2つの切り替え模様として語出力 | 語書き込み |
| \$52 | LED(PORTB)へ模様の連続出力 | 塊書き込み |
| \$60 | 受信語の2倍値を返送 | 処理呼び出し |
| \$70 | 受信したバイトの合計を語として返送 | 塊書き読み処理呼び出し |

含まれている例はSTK500開発基板とでの考えで書かれていますが、これは使用に関する必要条件ではありません。STK500使用时、“PORTB”ヘッダは“LEDS”ヘッダに接続され、“PORTD”ヘッダは“SWITCHES”ヘッダに接続されるべきです。「スイッチ押下状態読み込み」(\$20)はPINDの1の補数を返します。PORTB/LEDSにデータを出力する命令は配列内にそのデータを置きます。タイマ/カウンタ溢れ割り込みが配列内を1つずつ連続的に出力します。

PECに対する任意の支援がドライバに含まれます。TWI割り込みルーチン内の多くの場所でCRC計算を行う必要があります。既定では、それが最も小さく、最も速い実装なので、参照表法が用いられます。フラッシュメモリ内の256バイトの表が大きすぎる場合、これはCRC計算(法)に変更することが可能です。PECが必要なければこれを禁止することができ、より小さくより速い実装での結果になります。

PEC支援が任意なので、PECだけに関連する全ての操作は流れ図に於いて赤背景色(訳注:原書は灰色)で記されます。PECが許可されないなら、プログラムの流れは直線的に通して、または判定部で“YES”方向を通して進むでしょう。

CRC参照表法の実装は次のように動きます。

1. これが新しいCRC計算の始めなら、CRC値を0に設定します。
2. 旧CRC値を新データとXORします。
3. 参照表内の指標として2.からの値を使います。
4. この指標で取得した値が新CRCです。

5. 終了まで2.~4.を繰り返します。

データ列(そのものに対する)のCRC値がそのデータ列自体に追加されると、誤りが検出されない時に総合的なデータ列のCRCは0であるべきです。これは含められているドライブで使われています。PECが主装置から受信されると、そのPECもCRC計算を通して送られます。このようにして、PEC変数が0であることを調べることによってPEC照合が実行されます。

進行中の通信の状態を追跡し得るには次の変数1式が必要とされます(())内が変数名)。

- 送信長 (txLength)
- 送信計数器 (txCount)
- 受信計数器 (rxCount)
- SMBus状態フラグ (state)
- SMBus許可フラグ (enable)
- SMBus異常フラグ (error)
- PEC (pec) (許可の場合)
- 受信緩衝部 (rxBuffer)
- 送信緩衝部 (txBuffer)

送受信の緩衝部は一時記憶としてドライブによってのみ使われます。一旦規約が成功裏に完了したならば、受信緩衝部の内容は予定された場所へ複写されます。これは不正データを引き起こす通信失敗を防ぎます。応用の他の部分はSMBus送受信緩衝部を少しもアクセスすべきではありません。これらの緩衝部の長さは支援する最長規約に対応するように設定されなければなりません。

送受信の計数器は現在の規約の進行の経緯を保持します。受信計数器は1つのデータバイトが受信された時毎に増加され、受信緩衝部内のポインタとして働きます。同様に、送信計数器は1つのデータバイトが送信された時毎に増加され、送信緩衝部内のポインタとして働きます。送信長は送信されるべき総バイト数に設定されます。

SMBus状態フラグは使われつつある規約に関して決定をするために命令符号との組み合わせで使われます。フラグは以下の4つの値の1つに設定することができます。

- IDLE : SMBus規約の開始待機
- READ_REQUESTED : 最初の命令としてSLA+Rを受信
- WRITE_REQUESTED : 最初の命令としてSLA+Wを受信
- WRITE_READ_REQUESTED : 最初の命令としてSLA+Wを受信、再送開始、そしてSLA+Rを受信

SMBus許可フラグはSMBusインターフェースを許可または禁止するために応用の他の部分によって使うことができます。このビットが設定(1)されると、SMBusドライブは主装置によって開始されたどの送信も完了を試みます。このフラグが解除(0)されると、TWI割り込みは未だ生成されますが、SMBusドライブはSLA+R/Wに答えるだけで、ホストへそれが多忙であることを合図することによってその通信の中止を試みます。どのSMBus従装置もそれ自身のアドレスにいつも応答できるべきであることをSMBus仕様が要求することに注意してください。

SMBusドライブを完全に禁止してTWI割り込みを生成されることから防ぐには、TWCRレジスタ内のTWENビットが解除(0)されなければなりません。TWI単位部が禁止された場合、SCLとSDAのピンは標準出力ピンとして構成設定されるでしょう。SMBusでの衝突を避けるため、これらは内部プルアップ禁止の入力として構成設定されなければなりません。SMBusに接続した現在の全てのAVRデバイスがどのバス操作をも可能とするために給電を必要とすることに注意してください。

SMBus異常フラグはSMBus通信中に異常が起きた時に必ず設定(1)されます。このフラグは例実装内のどんな有用な目的に対しても決して使われません。これは上位規約で処理されなければなりません。

PECが許可されている場合、pec変数はそこまでにこの通信に対して計算された現在のPEC値を保持します。例に於いて2つのCRCルーチンが含められています。両方ともこの変数を直接操作し、従ってSMBusドライブによってのみ使われなければなりません。

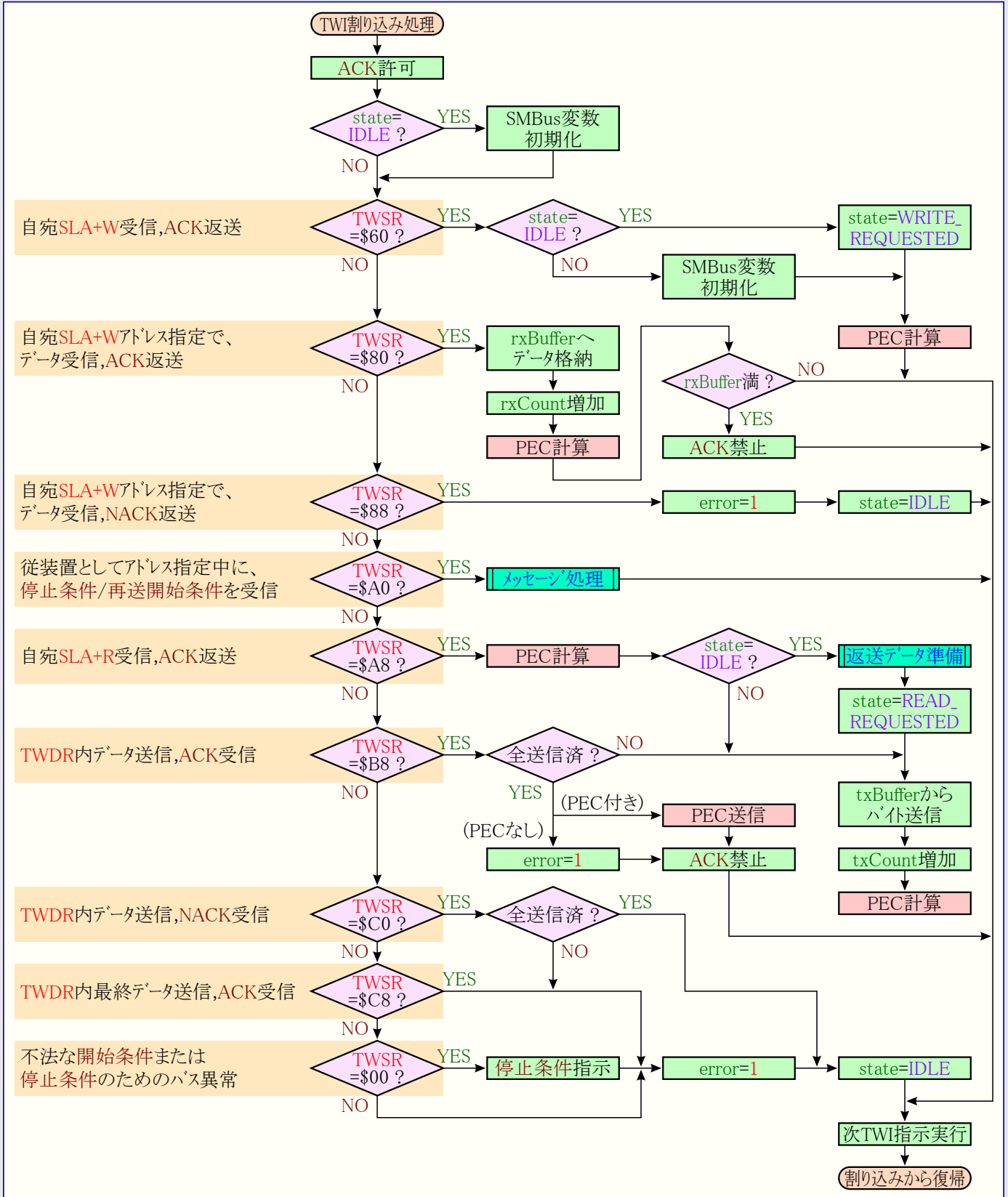
図23.は1つのSMBus規約実行中に於けるSMBusドライブの一般的なプログラムの流れを示します。これがTWI割り込みルーチンの流れ図でなく、図23.の流れ図の1つの周回を完了するために多数のTWI割り込みが起動されることに注意してください。TWI割り込みが実行される全ての場所に於いて、()内16進値(訳注:原書は破線の箱)がその方向でのプログラムの流れに対応するTWSRの状態符号を示します。

図24.はTWI割り込みルーチンを示します。この流れ図はTWI割り込みの反復に分類される図23.内の1つとして同じ様に実行します。

図26.は図24.で“メッセージ処理”と呼ばれる部分に対する流れ図を示します。これは実装されたSMBus従装置の応用特有部分の殆どの処です。これが走行する間にSMBCLKはLowを保持され、従って25msのSMBus t_{LOW:SEXT}に違反しないように関数を充分短く保つのが重要なことに注意してください。これが起きた場合、主装置は時間超過して線を開放し、AVRのTWI単位部はI²C適合のため、TWI単位部での時間超過がありません。そしてAVRはSMBDAT線をクロックを待つLowに保持し、故にバスを無期に妨げ得ます。これは応用に応じて処理されるべきなので、本実装はこの状況を修正する備えを持ちません。

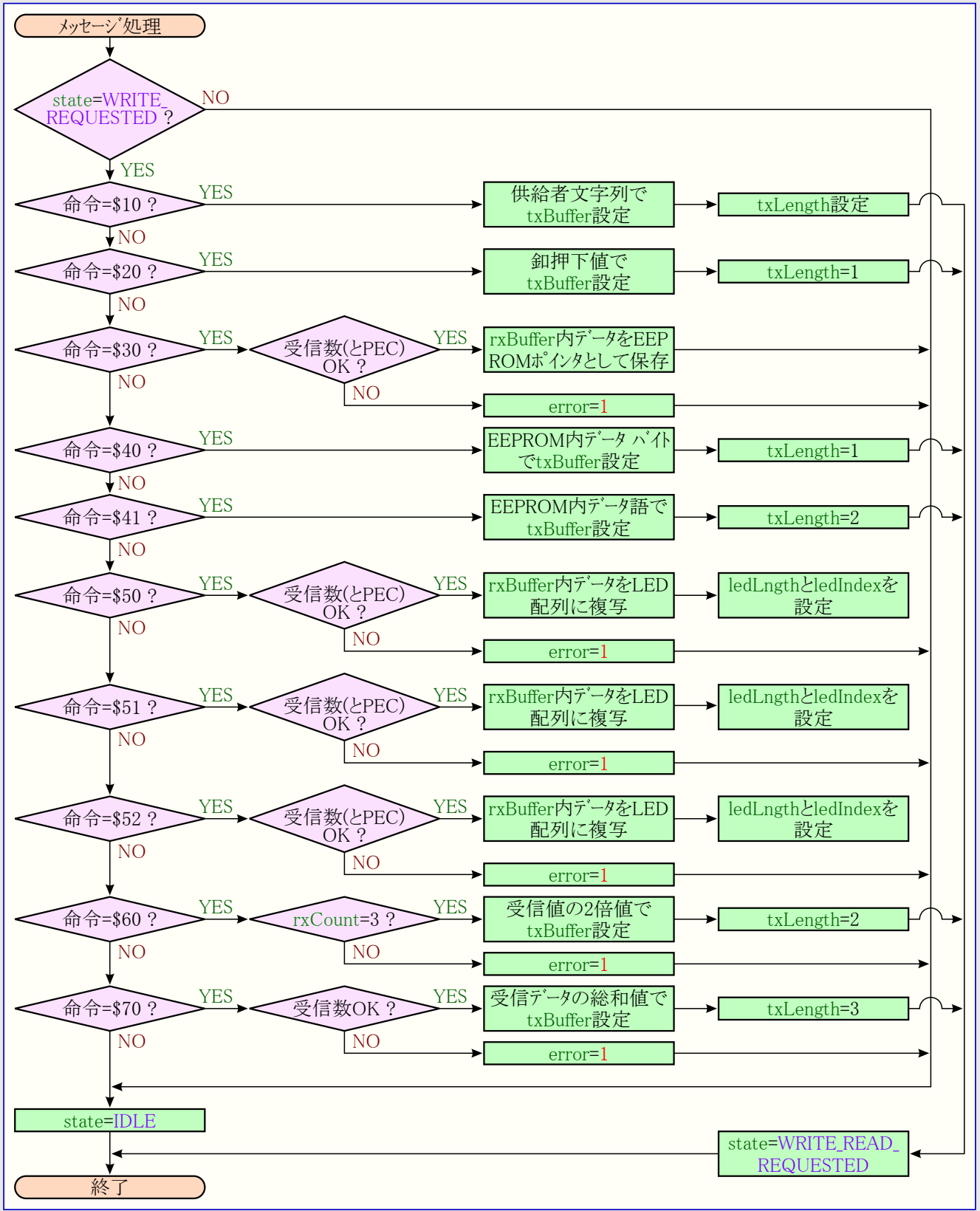
加えて、タイマ/カウンタ1が固定間隔で割り込みを生成するように構成されます。この割り込みはSTK500開発基板のLEDでの連続的なデータ表示に使用されます。最大32の連続値を'leds'全体配列内に置くことができます。'ledLength'全体変数がこの配列からの表示値数を定義し、同時に'ledIndex'が表示される現在値を制御します。

図24. TWI割り込み処理ルーチン



(訳注) 原書の図24.と図25.は図24.として纏めました。

図26. メッセージ処理流れ図



3.4. ソースコード

ZIPファイルとしてのダウンロードでソースコードが利用可能です。これはDoxygen適合注釈で資料化されており、コンパイルされた資料が'readme.html'ファイル内のソースと共に含まれています。支援されるコンパイラとファイルのコンパイル方法についての情報も含まれます。

3.5. 接続

AVRの'SDA'ピンを'SMBDAT'に、AVRの'SCL'を'SMBCLK'に接続してください。STK500開発基板が使われるなら、'PORTB'を'LED S'に、'PORTD'を'SWITCHES'に接続してください。例を動かすのに'PORTB'と'PORTD'の接続は必要ありませんが、機能が制限されます。

3.6. SMBus例の改造

試供実装実演は違う支援規約のように実装することができます。試供応用を望むように改造するためには、'ProcessRecvByte'と'ProcessMessage'の関数を変えられなければなりません。これはdoxygen資料またはソースコード内の注釈を通して提供されます。

'ProcessRecvByte'と'ProcessMessage'の実行中にTWI単位部によってクロックがLowに保持されることに注意してください。これらの関数がSMBus仕様の'クロックLow延長'章に適合すべく充分速く実行することを確実にしてください。

3.7. 新デバイス用支援追加

コンパイル中に「このデバイスはSMBusドライバで支援されません。」と言う異常メッセージが示される場合、そのデバイス用の支援を手動で追加することができます。デバイスはTWI単位部と16ビットのタイマ/カウンタ(LED用にだけ必要)が必要です。これは'device_specific.h'ファイルを編集することによって行うことができます。

'device_specific.h'はデバイス特性定義の長い一覧を含みます。ATmega16定義は次のように見えます。

```
#elif defined(__ATmega16__)
#define TIMER_OVF_VECT TIMER1_OVF_vect
#define TWI_VECT TWI_vect
#define TIMSK1 TIMSK
```

新デバイス用の支援追加は上のような新規登録を必要とするだけです。各記名の意味は表5.で説明されます。

表5. デバイス特性記名

| 記名 | 意味 |
|----------------|---------------------------------------|
| TIMER_OVF_VECT | タイマ/カウンタ1溢れ割り込みベクタの名前 |
| TWI_VECT | TWI割り込みベクタの名前 |
| TIMSK1 | タイマ/カウンタ1溢れ割り込み許可(TOIE1)ビットを含むレジスタの名前 |

3.8. コード量

コード量については資料をご覧ください。

4. 目次

| | |
|---------------------------------|----|
| AVR316 : TWI部を使うSMBus従装置 | 1 |
| 要点 | 1 |
| 1. 序説 | 1 |
| 2. SMBus仕様 | 1 |
| 2.1. SMBusとI ² C間の違い | 1 |
| 2.2. パケット誤り符号(PEC) | 2 |
| 2.3. SMBus規約 | 2 |
| 2.4. アドレス解決規約(ARP) | 5 |
| 3. AVRマイクロコントローラでのSMBus実装 | 6 |
| 3.1. TWI単位部 | 6 |
| 3.2. 提唱実装 | 7 |
| 3.3. 含まれているドライバと試供応用の説明 | 7 |
| 3.4. ソースコード | 12 |
| 3.5. 接続 | 12 |
| 3.6. SMBus例の改造 | 12 |
| 3.7. 新デバイス用支援追加 | 12 |
| 3.8. コード量 | 12 |
| 4. 目次 | 13 |
| お断り | 14 |



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

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、または利益の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。

© Atmel Corporation 2005. 不許複製 Atmel®、ロコとそれらの組み合わせ、AVR®とその他はAtmel Corporationの登録商標または商標またはその付属物です。他の用語と製品名は一般的に他の商標です。

© HERO 2021.

本応用記述はAtmelのAVR316応用記述(doc2583.pdf Rev.2583A-10/05)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。