



PIC18FXX8 Data Sheet

High Performance 28-Pin/40-Pin
Microcontrollers with CAN

19.3 CAN動作モード

PIC18FXX8は次の動作モードを持っています。これらのモードがそうです：

- コンフィギュレーションモード
- ディスエーブルモード
- 標準動作モード
- リスンオンリーモード
- ループバックモード
- エラーアクリッジモード (CANRXMビットを通じて選択された)

エラー認識モード以外、モードがREQOPビットをセットすることによってリクエストされます、そしてそれはCANRXMビットを通じてリクエストされます。OPMODEビットをモニターすることによって、モードへのエントリーがアクリッジされます。モードを変えるとき、すべてのペンディング(未決定)メッセージ送信が完了になるまで、モードは実際に変化しないでしょう。これのために、ユーザーはそれ以上のオペレーションが実行される前にデバイスが実際にリクエストモードに変わったことを確かめなくてはなりません。

19.3.1 コンフィギュレーションモード

CANモジュールは起動前に初期化されなければなりません。もしモジュールがコンフィギュレーションモードにあるなら、これはただ可能なだけです。REQOP2ビットを設定することによって、コンフィギュレーションモードはリクエストされます。ただステータスビットOPMODE2がハイレベルを持つときだけ、初期化は行なわれることができます。その後、コンフィギュレーションレジスタ、アクセプタンスマスクレジスタとアクセプタンスフィルターレジスタは書かれることができます。コントロールビットCFGREQをゼロに設定することによってモジュールは作動します。モジュールはプログラミングエラーを通じて偶然にCANプロトコルに違反することからユーザーを守るでしょう。モジュールがオンラインである間に、モジュールのコンフィギュレーションをコントロールするすべてのレジスタは修正されることができません。送信が起きている間にCANモジュールはコンフィギュレーションモードに入ることを可能にされないでしょう。CONFIGビットはロックとして次のレジスタを守るのに役立ちます。

- コンフィギュレーションレジスタ
- バスタイミングレジスタ
- アイデンテファイアーアクセプタンスフィルターレジスタ
- アイデンテファイアーアクセプタンスマスクレジスタ

コンフィギュレーションモードで、モジュールは送信あるいは受信をしないでしょう。エラーカウンタはクリアされます、そして割り込みフラグは変化していないままです。プログラマーは他のモードで限定されるアクセスであるコンフィギュレーションレジスタへのアクセスを持つでしょう。

19.3.2 ディスエーブルモード

ディスエーブルモードで、モジュールは送信あるいは受信をしないでしょう。モジュールはバスアクティビティのためにWAKIFビットをセットする能力を持っています、しかしながらどんな差し迫っている割り込みでも残っているでしょう、そしてエラーカウンターはそれらの値を維持するでしょう。もしREQOP <2:0>が001にセットされるならモジュールはモジュールディスエーブルモードに入るでしょう。

モジュールイネーブルをターンオフ(止める)ことによって、このモードは他の周辺モジュールをディスエーブルにすることに似ています。これはモジュール内部クロックを、モジュールがアクティブではないならストップさせます(すなわち、メッセージを受信するか、あるいは送信すること)。もしモジュールがアクティブであるなら、モジュールはCANバスで11のレセッシブビットを待ってアイドルバスとしてその条件を検出して、それからモジュールディスエーブルコマンドを受け入れるでしょう。OPMODE <2:0> = 001 はモジュールが成功裏にモジュールディスエーブルモードに入ったかどうかを示します。WAKIF割り込みはモジュールディスエーブルモードでまだアクティブな唯一のモジュール割り込みです。もしWAKIEが設定されるなら、CANバスが、SOFで起こるように、ドミナントステートを検出するときはいつでも、プロセッサは割り込みを受信するでしょう。モジュールがモジュールディスエーブルモードにあるとき、I/Oピンは標準I/Oファンクションに逆戻りするでしょう。

19.3.3 標準モード

これはPIC18FXX8の標準動作モードです。このモードで、デバイスはアクティブにすべてのバスメッセージをモニターして、そしてアクリッジビット、エラーフレームを生成するなどします。これは同じくPIC18FXX8がCANバス上にメッセージを送信するであろう唯一のモードです。

19.3.4 リスンオンリーモード

リスンオンリーモードはエラーで、メッセージを含めて受信するすべてのメッセージへのPIC18FXX8のための手段を提供する。このモードはバスモニターアプリケーションのために使われることができるか、あるいは「ホットブラッキング(通電接続)」状態でポーレートを検出することのために使われる。オートポー検出のために、お互いとコミュニケーションしている少なくとも2つの他のノードがあることは必要です。有効なメッセージが受信されるまで、ポーレートは経験的にテストの異なった値によって検出されることが出来ます。リスンオンリーモード、ただモードがエラーフラグあるいはアクリッジシグナルを含めてこのステートにある間にメッセージが送信されないであろうことを意味しているサイレントモードです。フィルターやマスクがただ特別のメッセージだけが受信レジスタにロードされることを可能にするために使われることができるか、あるいはフィルターマスクは全数字0にセットされることが出来るどんなアイデンティファイアでも通過するメッセージを可能にします。エラーカウンタはリセットされて、そしてこのステート非アクティブにされます。ただリスンオンリーモードだけはCANCONレジスタでモードリクエストビットをセットすることによりアクティブ化されます。

19.3.5 ループバックモード

CANバスで実際にメッセージを送信しないで、このモードは送信バッファから受信バッファまでメッセージの内部の送信を許すでしょう。このモードはシステム開発とテストで使われることができます。このモードで、ACKビットは無視されます、そしてただそれらがもう1つのノードから来るかのように、デバイスはそれ自身からの入ってくるメッセージを許すでしょう。ループバックモードはエラーフラグあるいはアクリッジシグナルを含めてこのステートにある間にメッセージが送信されないであろうことを意味しているサイレントモードです。デバイスがこのモードにある間に、TXCANピンはポートI/Oに逆戻りするでしょう。フィルターとマスクはただ特別のメッセージだけが受信レジスタにロードされることを可能にするために使われることができます。マスクはすべてのメッセージを受け入れるモードを提供するために全数字0にセットされることが出来ます。ループバックモードはCANCONレジスタでモードリクエストビットを設定することによってアクティブ化します。

19.3.6 エラー認識モード

モジュールはすべてのエラーと受信するどんなメッセージを無視するように設定されることができます。RXBnCONレジスタで11にRXMに<1:0>ビットを割り当てることによって、エラー認識モードはアクティブ化します。このモードで、エラー時間まで、メッセージセンブリバッファにあるデータは受信バッファでコピーされて、CPUインタフェースを経由して読まれることができます。加えるに、エラー時にCANバス内部サンプリング上にあったデータとプロトコルステートマシンのステートベクトルとビットカウンターCntCanはレジスタにストアされて、そして読まれることができます。

19.4 CAN メッセージ送信

19.4.1 送信バッファ

PIC18FXX8は3つの送信バッファを実装します。それぞれこれらのバッファについて14バイトのSRAMを占有しますそしてデバイスメモリマップにマップされます。MCUがメッセージバッファへの書き込みアクセスを持つためにTXREQビットはクリアされるに違いありません、メッセージバッファは送信されるべき色々な未決定(ペンディング)メッセージのクリアを示している。最小限において、TXBnSIDH、TXBnSIDLとTXBnDLCレジスタはロードされなくてはなりません。もしデータバイトがメッセージに存在しているなら、TXBnDmレジスタは同じくロードされなくてはなりません。もしメッセージが拡張アイデンティファイアを使うなら、TXBnEIDmレジスタは同じくロードされなくてはなりません、そして EXIDEビットはセットしました。メッセージを送る前に、メッセージが送られるとき、MCUはTXInEビットをイネーブルあるいはディスエーブル、割り込みの発生に初期化しなくてはなりません。MCUは同じくTXPプライオリティビットを初期化しなくてはなりません(セクション19.4.2参照)。

19.4.2 送信プライオリティ(優先順位)

送信プライオリティはPIC18FXX8内で未決定(ペンディング)の送信可能なメッセージのプライオリティゼーション(優先順位付け)です。これは独立しているから、そして関係ないからどんなプライオリティゼーション(優先順位付け)でも暗示的で、CANプロトコルにメッセージアービトラージ(通信調停)スキームでビルトされませんでした。SOFを送る前に、送信のために待ち行列に入れられるすべてのバッファのプライオリティは比較されます。最高優先を持っている送信バッファは最初に送られるでしょう。もし2つのバッファが同じプライオリティ設定を持っているなら、最高のバッファ数字を持っているバッファは最初に送られるでしょう。送信プライオリティの4つのレベルがあります。もし特定のメッセージバッファのためのTXPビットが11にセットされるなら、そのバッファは最も高い可能なプライオリティを持っています。もし特定のメッセージバッファのためのTXPビットが00であるなら、そのバッファは最も低い可能なプライオリティを持っています。

19.4.3 送信初期化

メッセージ送信を初期化するために、TXREQビットはそれぞれのバッファのために送信されるようにセットされなくてはなりません。TXREQがセットされるとき、TXABT、TXLARBTとTXERRビットはクリアされるでしょう。TXREQビットをセットすることはメッセージ送信を初期化しません、それはただメッセージバッファに送信のために準備ができているというフラグをつけます。デバイスはバスが利用可能であることを検知するとき送信が始まるでしょう。デバイスはそれで用意ができている最も高いプライオリティメッセージの送信を始めるでしょう。送信がサクセスフリー(成功裏)に完了した、TXREQビットがクリアされるであろうとき、TXBnIFビットはセットされるでしょう、そしてもしTXBnIEビットがセットされるなら、割り込みが生成されるでしょう。もしメッセー

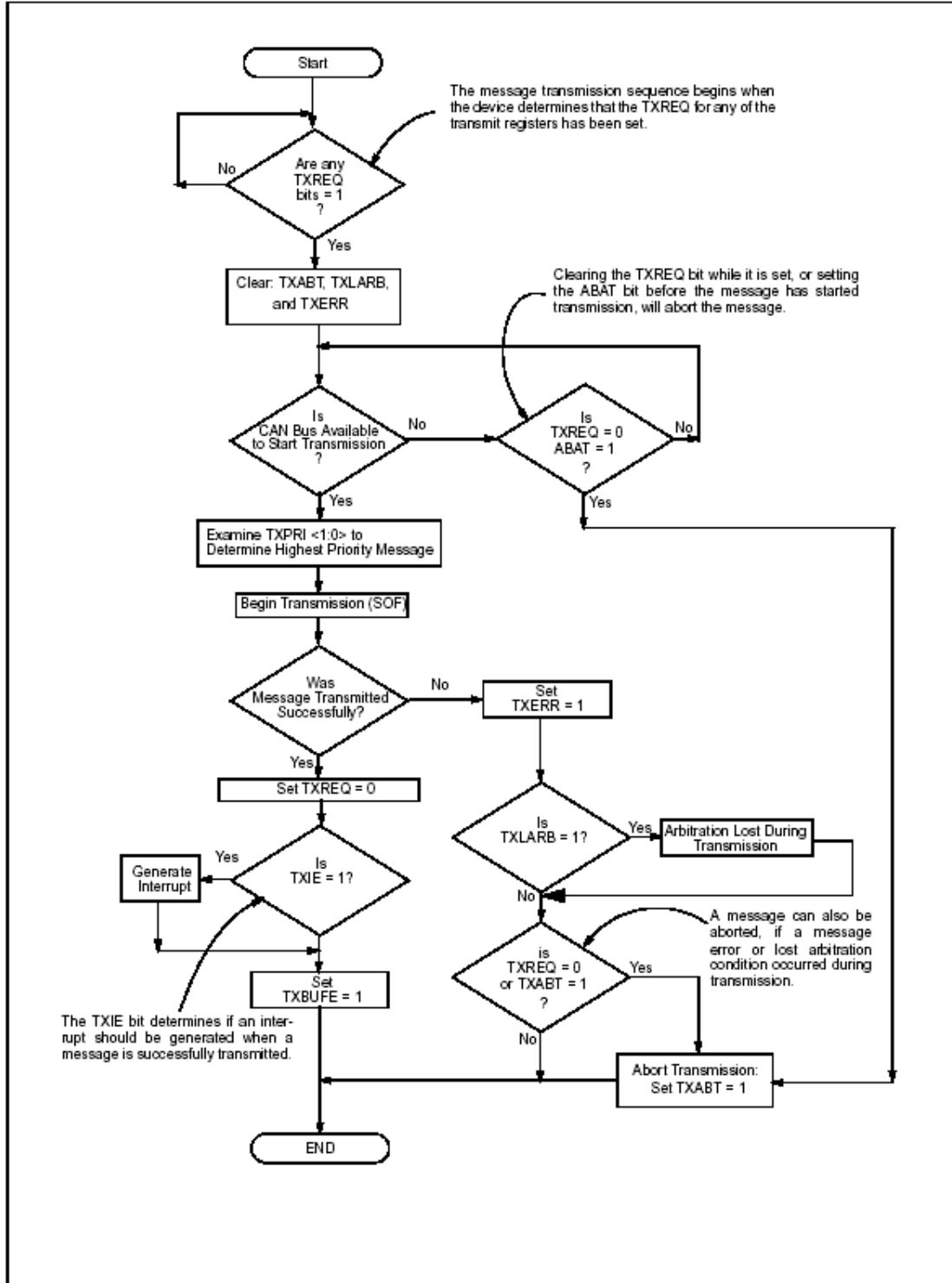
ジ送信が失敗するなら、TXREQはメッセージが送信のためにまだ未決定(ペンディング)である、そして次の状態フラグの1つがセットされるであろうことを示して、セットのままです。もしメッセージが送信し始めましたが、エラー条件に遭遇したなら、TXERRとIRXIFビットはセットされるでしょう、そして割り込みが生成されるでしょう。もしメッセージがアービトレーション(通信調停)を失敗したなら、TXLARB ビットはセットされるでしょう。

19.4.4 送信を中止する(アボート)

対応するメッセージバッファと結び付けられるTXBnCON.TXREQビットをクリアすることによって、MCUはアボート(停止)にメッセージをリクエストすることができます。CANCON.ABATビットをセットすることはすべてのペンディングメッセージのアボートをリクエストするでしょう。もしメッセージがまだ送信を始動させなかったなら、あるいはもしメッセージが始まった、しかしアービトレーション(通信調停)の失敗あるいはエラーによって割り込まれるなら、アボートが処理されるでしょう。モジュールがTXBnCON.ABTFビットをセットするとき、アボートは示されます。もしメッセージが送信し始めたなら、それは完全に現在のメッセージを送信しようと試みるでしょう。もし現在のメッセージが完全に送信されて、そしてアービトレーション(通信調停)失敗あるいはエラーなら、メッセージがサクセスフリー(成功裏)に送信されたならABTFビットはセットされないでしょう。同じく、もしメッセージがアボートの間に送信されているなら、リクエストしてください、そうすればメッセージはアービトレーション(通信調停)失敗あるいはエラーです、メッセージは再送信されないでしょう、そしてABTFビットはメッセージがサクセスフリー(成功裏)にアボート(中止)されたことを示して、セットされるでしょう。

PIC18FXX8

FIGURE 19-2: TRANSMIT MESSAGE FLOW CHART



詳細はホームページ <http://www5b.biglobe.ne.jp/~tekhanzo/>

をご確認ください。

不許複製

PIC18Fxx8 CAN モジュール章完全日本語訳 サンプル

Japanese Translation Copy Rights©2004-2010 Tech - Hanzougane Yoshiaki Morohashi

発行 2007年7月12日 初版発行

翻訳者 諸橋 義明

発行元 テック・ハンゾウガネ(個人事業者)

〒940-0213

新潟県長岡市栃尾山田町 6-53

TEL 0258 (53) 0082 E-mail tekhanzo@mta.biglobe.ne.jp

HP アドレス <http://www5b.biglobe.ne.jp/~tekhanzo/>
