

著者マイクロチップ社 Stephen Bowking & Naveen Raj

## はじめに

多くの PIC マイコンには同期シリアルポート (SSP)や主同期シリアルポート(MSSP)が搭載されています。この周辺回路は SPI や i2c プロトコルがが使用できるようになっています。この ap 資料の目的は i2c のプロトコルを読者により深め、i2c バス上でスレーブとしてこれらのモジュールを働かせるためのやり方を示します。更に i2c の仕様とか SSP, MSSP について情報が必要でしたら、巻末の参考節に示された元資料を参照してください。

## i2c バスの仕様について

この資料では i2c バスの仕様を語ることは範囲外ですが、ここで基本的なことだけ触れておきます。インター・インテグレイテッド・サーキット(i2c)は PHILIPS 社により当初 PCB 内で IC 間をつなぐデータ転送のために開発されました。物理的な信号線としてはオープンコレクタのクロック (SCL) とデータ (SDA) の 2 本です。バスは 1 マスター/多スレーブ構成か、あるいは複数マスターになります。マスターはスレーブとのやり取りを行うクロック源の責任があります。

i2c プロトコルでは 7 ビットと 10 ビットのアドレッシングモードがあり、128 あるいは 1024 の素子をバス上に許容できます。実用的には仕様上取り置きアドレスがあり、もっと少な目の数になります。たとえば、7 ビットだと 112 が

最大使えるアドレスです。バス上のすべての転送はマスターが仕切り、いつもバス上にクロックを供給します。データ転送は一度に 8 ビットを MSB から先にバス上へ送り出し実行されます。1 転送サイクルで送るデータ量に制限はありません。

i2c プロトコルにはハンドシェイキング機構が含まれています。8 ビットの各転送後 9 番目のクロックが送られます。このとき送信側のデバイスは SDA 信号を開放し、受信側のデバイスは送信側からのデータに対する応答を受信側が受け取ります。もし受信データが良ければ ACK(SDA を L にする)あるいは NACK(SDA を H にする)を送ります。SDA のレベル遷移は SCL が L の時に行わねばなりません。

中略

FIGURE 1: TYPICAL I<sup>2</sup>C™ WRITE TRANSMISSION (7-BIT ADDRESS)

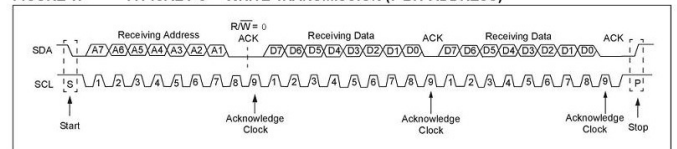


FIGURE 2: TYPICAL I<sup>2</sup>C™ READ TRANSMISSION (7-BIT ADDRESS)

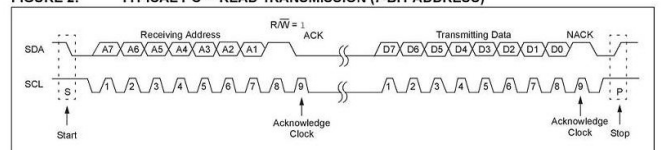
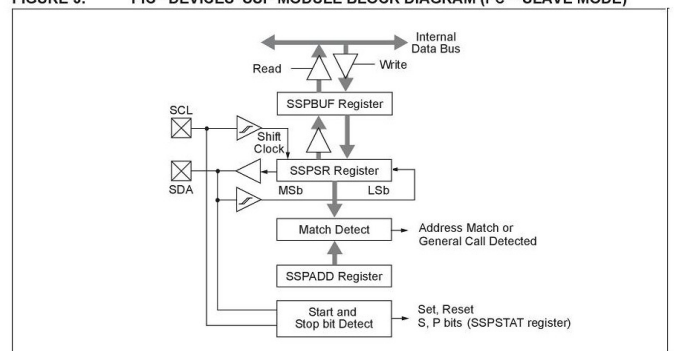


FIGURE 3: PIC® DEVICES' SSP MODULE BLOCK DIAGRAM (I<sup>2</sup>C™ SLAVE MODE)



## SSP モジュールの説明

i2c スレーブモードでの SSP モジュールのブロック図を図 3 に示しました。i2c スレーブ通信に必要とされるキーの制御とステータスビットは次のようなスペシャルファンクションレジスタで提供されます。

- SSPSTAT
- SSPCON
- PIR1 (割り込みフラグビット)
- PIE1 (割り込み許可ビット)

これらのレジスタにおけるビット機能のいくつかは SSP モジュールを i2c で使うのか SPI で使うのかで異なります。ここでは i2c モードでの各ビットの機能を示します。各ビットの完全な機能表記については各デバイスのデータシートを見てください。

### モジュールの動作状態を示す SSP ビットについて

#### BF(SSPSTAT<0>)

バッファフル(BF)ビットは今 SSPBUF 内にデータバイトがあるかどうかをユーザに知らせます。SSPBUF を読み取ったり、バイトが送信され全部シフトアウトすると自動的にこのビットはクリアされます。この BF ビットは次のような状況でセットされます。

- LSB まできれいにアドレスが受信されたとき。これは i2c におけるライト動作中にマスターデバイスが送る初めてのバイトです。
- 各バイトが i2c 書き込み動作でスレーブにて受信されたとき。
- SSPBUF に書かれたデータバイトがマスター側に送信されたたび。SSPBUF からマスター側にすべてのビットが送り出されたときに BF は自動的にクリアされます。

リード動作でアドレス受信の LSB がセットされたときに BF フラグがセットされる場合も有ります。詳しくは付録 C 「PIC16 と PIC18 のデバイスにより異なる i2c 状態について」を参照してください。

#### UA(SSPSYTAT<1>)

UA(アドレス更新)ビットは 10 ビットアドレスモードのときだけに使います。10 ビットアドレスモードでは i2c アドレスは 2 バイトに分けて送ります。10 ビットアドレスの上位半分は(1111 0 A9 A8 0)は最初に一致をみるために SSPADD に取り込まれます。この特別なアドレスは 10 ビット上位アドレスのために i2c プロトコルの中でリザーブされています。アドレス一致があると SSP モジュールは UA ビットをセットして一致判定のために下位アドレスを取り込むことを示します。

#### R/~W (SSPSTAT<2>)

R/~W (読み取り/書き込み) ビットはマスターデバイスがスレーブに対して読み取りなのか書き込みなのかを連絡します。このビットはマスターにより送られるアドレスの LSB 状態を示しています。R/~W ビットは該当の i2c メッセージの間のみ有効であり、STOP コンディションや START コンディションやマスターからの NACK 送信でリセットされます。

#### S(SSPSTAT<3>)

S(スタート)ビットはバス上でスタートコンディションが今発生したところだと知らせる。このビットは初めてモジュールがイニシャライズされてどちらのビットもクリアなときをのぞけば P(ストップ) ビットの逆である。

#### P(SSPSTAT<4>)

P(ストップ)ビットはバス上でストップコンデションが今発生したところだと知らせる。このビットは初めてモジュールがイニシャライズされてどちらのビットもクリアなときをのぞけばS(スタート)ビットの逆である。このPビットはバスがアイドル状態であることを決めるのに使える。

#### D/~A(SSPSTAT<5>)

D/~A(データ/アドレス)ビットは SSP モジュールが今受信したデータバイトがデータなのかアドレスなのかを示している。読み取り動作ではマスターデバイスに送られる最後のバイト D/~A ビットがセットされたらデータバイトだったと示している。

#### WCOL(SSPSTAT<7>)

WCOL(書き込み衝突)ビットは一つ前に SSPBUF に書かれたワードが送信中に書き込みが行われたことを示す。書き込み衝突が起こった際には一つ前の SSPBUF の内容は変化しない。WCOL ビットが発生したらソフトでクリアする必要がある。

#### SSPOV(SSPSTAT<6>)

SSPOV(SSP オーバーフロー) ビットは SSPBUF がまだ前の受信データが残っているところに新しいバイトを受信したことを示す。この場合、SSP モジュールは ACK パルスを返さず、また SSPBUF に新しいデータを取り込まない。データを使うかどうかに関わらず、ユーザは BF ビットがセットされたらこの SSP オーバーフロー状態にならないよう SSPBUF を読み取っておく必要がある。ユーザは SSPBUF を読み取り、オーバーフロー状態をクリアして SSPOV をクリアせねばならない。もしユーザが BF ビットをクリアするために SSPBUF を

読んでも、しかし SSPOV ビットがクリアしないときは次のデータバイトが SSPBUF に入っており、ACK パルスが出ていないことになる。

#### SSPIF(PIR1<3>)

SSPIF(SSP 割り込みフラグ) ビットは i2c イベントが完了したことを示す。ユーザはどのイベントが起こり、次のアクションは何をするかを定めるためにここでステータスビットをポーリングせねばならない。SSPIF ビットはユーザによってクリアする必要がある。

#### モジュール制御のための SSP ビット

##### SSPEN(SSPCON<5>)

SSPEN(SSP イネーブル) ビットは SSP モジュールを動かし、指定 I/O ピンをシリアルポートに構成する。

##### CKE(SSPSTAT<6>)

CKE(クロックエッジ)ビットは SSP モジュールを i2c 設定で使うときには機能は無い。クリアしておくこと。

##### SMP(SSPSTAT<7>)

SMP(サンプル位相)ビットは SSP モジュールを i2c 設定で使うときには機能は無い。クリアしておくこと。

##### CKP(SSPCON<4>)

CKP(クロックの極性)ビットは i2c プロトコルの時にはクロック延長のために使う。CKP ビットがクリアされていると、スレーブデバイスは SCL ピンを L 状態に保ち、マスターがバス上にクロックパルスを送り出せなくする。クロック延長中にはマスターはスレーブ側によりクロック線が開放されるまでクロックを送り出そうと待っている。クロック延長が便利なのはスレー

ブ側が入力バイトに対しすばやく応答できないときや SSPBUF がマスター側へ送信すべきデータの用意に時間を要するときなどである。SSP モジュールはデータがマスター側で読み取られる時は自動的にクロック延長を行う。CKP ビットはアドレスバイトや夫々連続するバイトが読まれた後、モジュール内でクリアされる。SSPBUF を読み出し後、クロックをリリースし、次のバイトの転送を許可するためにソフトウェアにより CKP ビットはセットせねばならない。

#### SSPM3:SSPM0(SSPCON<3:0>)

SSPM3:SSPM0(SSP モード)ビットは SSP モジュールを SPI や i2c プロトコルに対応させるために使われる。特定するための値は各該当のデバイスのデータシーを参照のこと。

#### SSPIE(PIE1<3>)

SSPIE(SSP の割り込み許可)ビットは SSP 割り込みを許可する。適宜 GIE とか PEIE 許可ビットもこのビットによる割り込みを有効にするためにセットせねばならない。

#### SSP を i2c スレーブモードに構成するには

モジュールを有効にする前にまず SCL,SDA に使用する信号端子を TRIS ビットで入力に設定したかを確認願いたい。これは I/O ピンを i2c プロトコルで駆動するために要求される構成事項である。

SSP モジュールは SSPCON レジスタを使用して構成し有効にする。SSP モジュールは次のような i2c モードに設定できる。

- ・ 7 ビットアドレスの i2c スレーブモード
- ・ 10 ビットアドレスの i2c スレーブモード
- ・ 7 ビットアドレスの i2c スレーブモードでス

タート、ストップコンデション割り込みも許可

- ・ 10 ビットアドレスの i2c スレーブモードでスタート、ストップコンデション割り込みも許可

この 4 構成の中で始めの 2 つは最も良く使われるスレーブの使い方である。残り 2 つはバス上でスタートとストップコンデション発生したときに割り込み発生し、バスがアイドルのときを検出するのに便利である。バスがアイドルであることを検出できれば、スレーブ側はバス上でマスターになることもできる。SSP モジュールには i2c 通信のマスターに対してのハードウェアサポートは無いのでマスター側通信はファームウェアで盛り込み必要である。

#### スレーブアドレスの設定の仕方

スレーブ側にアドレスは SSPADD レジスタ(図 3 参照) に書きこまなければならない。7 ビットアドレッシングだとビット<7:1>がアドレス値になる。LSB はアドレスには使われず、このビットはバス上でのやり取りが読み込みか書き込みかを定める。よって、SSPADD はいつも偶数値となる。その結果、スレーブ側は書き込み動作と読み込み動作で 2 つのアドレスを持つことになる。

#### ソフトウェアによる SSP イベントの扱い方

SSP モジュールを i2c のスレーブ通信に使うには一般的に各 i2c イベントの後にファームウェアによる操作をシーケンシャルに必要なになる。SSPIF ビットはバス上での動作終了イベントを通知する。SSPIF ビットをソフトでポーリングするか、割り込み要因に割り当てるか構成できる。SSPIF ビットがセットされるたびにどの i2c イベントかを SSPSTAT レジスタの各種ビ

ットを確かめて特定せねばならない。この説明のために考えられる状態を区別して、夫々について検討するのが役立つ。次に示すような i2c イベントあと SSP モジュールは合計 5 種の状態になる。SSP モジュールはイベントを覚えているわけにはいかないので、各 i2c イベント夫々の新規の SSPIF 割り込み毎に決めなければいけない。各イベントは割り込みを発生するので、コードでは i2c バスで丁度何が発生したかを確定するために SSPSTAT レジスタの各種重要 i2c ビットを確かめるそしてモジュールがその状態にあるかを確定する。コードのサンプルを付録 A 「i2c スレーブのソースコード例」と付録 B 「i2c スレーブのソースコード例(新規 PIC18 用修正)」の中にどうするかを示した。

#### 状態 1 : マスター側は書き込みで、最新バイトはアドレスだったとき

バス上のマスター側がバス上にスタートかリスタートコンデションを起こして、新しい書き込み動作を始めようとして i2c アドレスバイトを送ったとする。アドレスバイトの LSB はマスター側がスレーブに書き込みをしたいので、'0'である。SSPSTAT レジスタのビット類は次のようになる。

- S=1 (スタートコンデションが起こった)
- R/~W=0 (マスターはスレーブにデータを書く)
- D/~A=0 (最新バイトはアドレスだった)
- BF=1 (バッファがフル)

このとき、SSP バッファはフルで送られてきたアドレスが入っている。BF ビットをクリアするためにこのアドレスを無視するにしても SSPBUF を読まないといけない。もしここで SSPBUF を読まなかったらマスターが次に送ってくるバイトが追突して SSP オーバーフロ

ーを起こしてしまい、SSP モジュールは NACK を送り返す。

#### 状態 2: マスターが書き込みで最新バイトはデータ

ライト動作でアドレスが送られてきた後(状態 1)にはマスター側はスレーブに対し 1 バイト以上のデータをスレーブ側へ送ってくるはずである。もし書き込みの前フルでなかったら、SSP モジュールは ACK パルスを 9 番目のクロックに載せて返す。そうでない場合は SSPOV ビットをセットして受け取りバイトの NACK を示す。SSPSTAT レジスタ内のビット類はマスターがデータバイトをスレーブに書き込んだ後、次のように値になる。

- S=1 (スタートコンデションが起こった)
- R/~W=0 (マスターはスレーブにデータを書く)
- D/~A=1 (最新バイトはデータだった)
- BF=1 (バッファがフル)

#### 状態 3: マスターが読み込みで最新バイトはアドレスのとき

バス上のマスター側がバス上にスタートかリスタートコンデションを起こして、新しい読み出し動作を始めようとして i2c アドレスバイトを送ったとする。アドレスバイトの LSB はマスター側がスレーブに書き込みをしたいので、'1'である。SSPSTAT レジスタのビット類は次のようになる。

- S=1 (スタートコンデションが起こった)
- R/~W=1 (マスターはスレーブからデータを読む)
- D/~A=0 (最新バイトはアドレスだった)

この時点で SSP バッファはマスターに送るべ

きデータを取り込む準備ができています。CKP ビットは SCL 信号を L にするためクリアされる。SSPBUF に入ったスレーブからのデータがマスターに送られるよう CKP ビットは SCL 信号を開放するためにセットされる。

#### 状態 4: マスターが読み込みで、最新バイトはデータ

状態 4 が起こるのはマスターが前のデータをスレーブから受け取って、次のバイトを読み取りたいときである。SSPSTAT レジスタのビット類は次のようになる。

- S=1 (スタートコンデションが起こった)
- R/~W=1 (マスターはスレーブからデータを読む)
- D/~A=1 (最新バイトはデータだった)
- BF=0 (バッファが空)

このとき SSPBUF はマスターに送るべきデータを受け取り可能である。CKP ビットは SCL 信号を L にするためクリアされる。スレーブからのデータがマスターに送られるよう SSPBUF に入ったら CKP ビットは SCL 信号を開放するためにセットされる。

#### 状態 5: マスターが NACK を出す

状態 5 はスレーブ側から受信したデータに対する応答として NACK を送るときに発生する。この動作はマスターがスレーブからのデータをもう要らないことを示すためのものである。NACK 信号は i2c メッセージの最後の信号であり、スレーブの i2c 回路をリセットする効果がある。SSPSTAT レジスタの関連ビットは次のようになる。

- S=1 (スタートコンデションが起こった)
- D/~A=1 (最新バイトはデータだった)
- BF=0 (バッファが空)

- CKP=1 (クロックは開放した)

NACK イベントは CKP をセットすることで区別される。特にこの場合の状態ビットはマスターからデータビットが受信されて、バッファが空になったと言う状態である。

#### SSP エラーの取り扱い

スレーブのファームウェアで SSPBUF を読み取る時は必ず SSPOV ビットをチェックして受信オーバーフローが起きていない事確かめる必要があります。もしオーバーフローが起きているならば、SSPOV ビットをソフトウェアでクリアしないといけないし、SSPBUF は更に発生する受信のために読み取られねばなりません。

SSP オーバーフローが起こった後の対処は使い方にも拠ります。スレーブ側論理回路がオーバーフロー発生でマスターに NACK を送ります。通常の使い方ではマスターはスレーブからの ACK が返るまでデータを再送し続けるでしょう。SSPBUF にデータを書き込んだ後、ユーザは WCOL ビットををチェックして書き込み衝突が発生していないことを確かめます。実用上に考えて、各状態遷移で BF ビットがクリアされから SSPBUF にファームウェアが書き込みし、スレーブ側がマスターにデータ送信するなら書き込み衝突が起こるわけがありません。