

ソフトウェアで制御するSPIマスタ/SPIスレーブ・プログラムの作成

ソフトウェア制御によるSPIマスタ/スレーブ (PIC/H8)

本章では、専用ハードウェアを使わずにソフトウェア制御のSPIマスタおよびスレーブ制御プログラムを作成します。プログラムは、章末にPIC用(リスト12-1)とH8用(リスト12-2)に示します。

12-1 ソフトウェア制御のSPIマスタ

● SPIマスタ・ドライバの動作

SPIの制御、とくにマスタの場合は簡単です。送信データ用の変数“SPITxData”のMSBから1ビットずつ順番にビット値を取り出し、それをSDOポートに出力し、SCKパルスを出力します。SCKパルスの出力終了時に受信データのビット値をSDIポートから入力して、それをMSBから順番に受信データ用の変数“SPIRxData”へ格納して行きます。この送受信の処理を8回繰り返します。

図12-1に、マスタ・ドライバのフローチャートを示します。

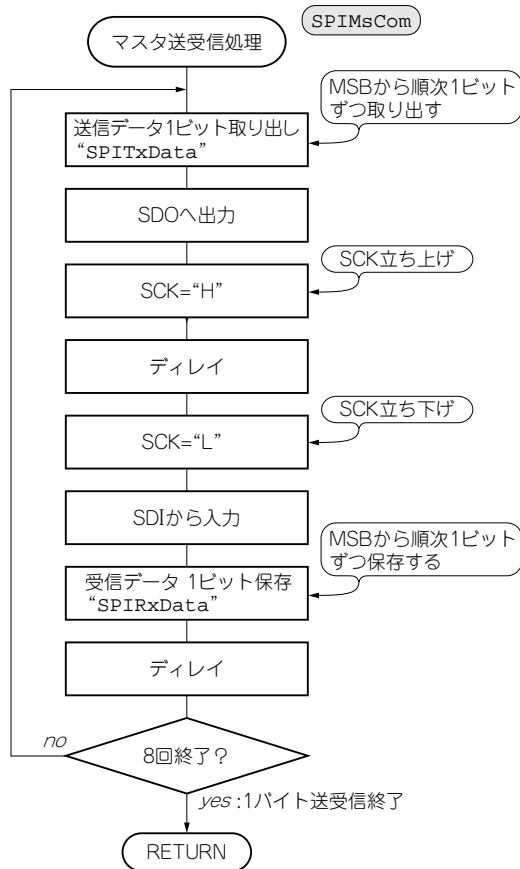
● 実験機#1(SPIマスタ)アプリケーション

実験用マスタ・アプリケーションは、トグル・スイッチでマスタ送信するデータを選択し送受信ボタンを押すと通信が始まり、マスタ受信したデータをLCDへ表示します。PIC版のプログラムは、ドライバがソフトウェア制御に替わるだけで、それ以外は第11章と同じです。フローチャートは、図11-6を参照してください。

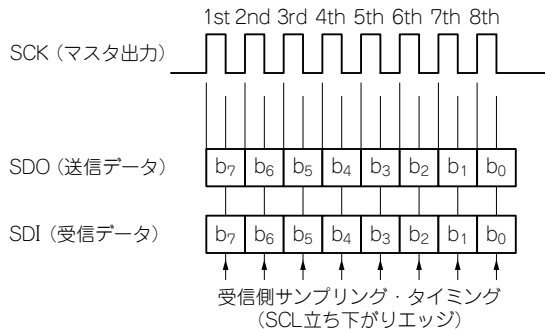
使用する主なファイルを次に示します。

◆ファイル構成(PIC版)

```
[PIC]
[121]
main121.asm ..... マスタ・アプリケーション ソース・ファイル
main121.mcp ..... プロジェクト・ファイル
main121.hex ..... HEXAファイル
[Common]
ComDef.inc ..... 共通定義ファイル
SPISMS.asm ..... SPIソフトウェア制御マスタ/スレーブ・ドライバ
```



(a) SPIソフトウェア制御マスタ・ドライバ



(b) SPI信号の入出力タイミング

図12-1 SPIソフトウェア・マスタ・ドライバの処理の流れ

ソフトウェアで制御するSPI通信のマスタの制御フローチャート。スループレートはディレイ時間で調整する。

- Lcd.asm LCDドライバ
- Key.asm SW入力ドライバ
- Hexa.asm Hexaユーティリティ
- AppFnc.asm アプリケーション共通サブルーチン
- Delay.asm ディレイ・ルーチン

※その他、ソース・ファイル“*.asm”と同名のヘッダ・ファイル“*.inc” (EXTERN宣言の定義)がソース・ファイルごとにある。

◆ファイル構成 (H8版)

- [H8]
- [121]

