【作業内容】

プログラムの理解と作成

【作業項目】

1. wiringPiのライブラリを使ったプログラミング

wiringPiのSPI用関数は、初期化関数としてwiringPiSPISetup(ch, speed)、データの送受信関数としてwiringPiSPIDataRW(ch, data, byte)を用いる。

・int wiringPiSPISetup(int ch, int speed)

Ch: 0 | 1 チャンネルを与える。

0はCE0（BCM＿GPIO8）、1はCE1（BCM＿GPIO7）と連動するので注意。

指定したチャンネルのCE0またはCE 1がアクセス時に０となる。

CE0をC S／SHDNに接続したなら、0を指定することになる。

　Speed: SPI通信のクロック周波数はHzで指定する。32Mhzまで対応可能。

MCP3008では、10k Hzが１０ビットA/D変換時の最大周波数とされている。よって10000を与えることとなる。

　戻り値：成功時はファイルディスクリプた、失敗時は-1

　　・int wiringPiSPIDataRW(int ch, unsigned char \*data, int len)

　Ch: 0 | 1 スレーブのチェンネルを与える。上と同じ。

Data: 送受信するデータのバッファとなる配列名を指定する。

なお、送受信終了時に、受信データによって送信データは上書きされる。

実行前：data[0]=”0000 0001”, data[1] = “SCCC XXXX”, data[2] = “XXXX XXXX”

* 例）S: 1（シングルエンド指定）、CCC：”000”（ch0を指定）

※Xは0でも１でもよいことを示す。

実行後：data[0] = “???? ????”, data[1] = “???? ?0BB”, data[2] = “BBBB BBBB”

* Data[1]の回２ビットをdata[2]の８ビット上位ビットとして連結

※？はそのビットの値が不定であることを示す。

　Len: 3 送受信するデータの長さ（バイトで指定）を指定する。

　戻り値：不明（記述がない）

　なお、wiringPiSPIDatRW()関数を使わなくても、システムコールread(), write()でもアクセス可能である。データ読み込みの必要がない、カラーLEDやシフトレジスタなどのアクセスにはwrite()を使うのが良い方法である。ただし、今回はA/D変換器に書き込みながら読み出しが必要なので、この関数を使用する。

【作業時間】

・作業時間：60分

・報告書作成時間：60分