

計測コラム emm87 号用

デジタル計測の基礎 – 第 15 回「浮動小数点形式とデータ転送」

今回は、前回の続きとして、いろいろな浮動小数点形式のデータとそれを外部 PC で読み取る方法などについてお話しします。

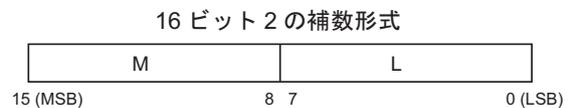
FFT アナライザなどのデジタル計測器から得られるデジタルデータを PC などで読み取る（変数に代入する）ためには、そのデータの形式と並び順を理解する必要があります。

計測/分析された結果のデジタルデータ形式は、大きく固定小数点形式と浮動小数点形式に分けることができます。固定小数点形式としては、主に A/D コンバータからのデジタル時間データ列で、例えば、16 ビット 2 の補数形式（2 バイト）です。これを GP-IB インターフェイスなどでバイト単位ごとに読み取った場合、求める数値（整数バイナリデータ）を x とすると以下のように変換します。

$M \geq 128$ ならば
 $x = 2^8 M + L - 2^{16}$

$M < 128$ ならば
 $x = 2^8 M + L$

(注) M は上位バイト、L は下位バイトを意味する



このとき問題になるのは、1 バイトデータを読み取る順番がどうなるか、すなわち計測機器側からどのような順番で送られてくるかです。

一般に、1 点が多バイトで構成されるデータをバイト単位でメモリアクセスできるコンピュータのデータメモリに格納する方式は、上位バイト (M) を下位番地に格納する方式 (ビッグエンディアン) と下位バイト (L) を下位番地に格納する方式 (リトルエンディアン) の 2 種類あります。

例えば上記の 2 バイト (16 ビットデータでその数値を 16 進数で 1234H とする) データが 1000H 番地から格納されているとすると、図 1 のようになります。



図 1 . 多バイトデータの格納方式の違い (2 バイトの時)

1 データ 4 バイト (32 ビットデータでその数値を 16 進数で 12345678H とする) の時も同様です (図 2) 。



図 2 . 多バイトデータの格納方式の違い (4 バイトの時)

2 方式の比較は、表 1 となります。最近では、両方式をサポートするバイエンディアンを採用する CPU (PowerPC など) もあります。また、LAN のネットワーク (TCP/IP プロトコル) 上での多バイトデータはビッグエンディアン方式で転送されるのがルールのようなのです。

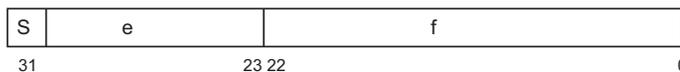
	特徴	主な対応 CPU
ビッグエンディアン	メモリを表示させた時、その並びが人間にとって分かりやすい	モトローラ系の CPU (68 系)
リトルエンディアン	コンピュータにとって処理が容易となり、スピードが速くなる。 メモリ効率もよい	インテル系の CPU (x86 系)

表 1 . 格納方式の比較

上記の例のように、インターフェイスから転送されて来るデータを 1 バイトずつ読み取り変数に入れる場合は、データの形式と転送順さえ分かれば 1 データずつ数値変数に変換すればよいことになります。

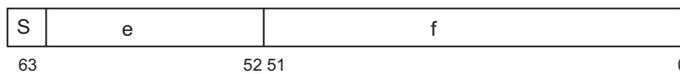
計測機器では、A/D コンバータからの生の時間データを演算処理しますので、その結果は普通浮動小数点形式のデータとなります。前回の計測コラムに説明しました IEEE 規格の形式を含め、主な浮動小数点形式は、図 3 となります。

① IEEE 規格単精度浮動小数点形式 (32bit)



S: 符号 e: 指数部 f: 仮数部
 数値: $(-1)^S \times 2^{e-127} \times (1.f)$

② IEEE 規格倍精度浮動小数点形式 (64bit)



S: 符号 e: 指数部 f: 仮数部
 数値: $(-1)^S \times 2^{e-1023} \times (1.f)$

③ MSBASIC 単精度浮動小数点形式 (32bit)



S: 符号 e: 指数部 f: 仮数部
 数値: $(-1)^S \times 2^{e-129} \times (1.f)$

④ Am9511A 単精度浮動小数点形式 (32bit)



S: 符号 e: 指数部 (2の補数) f: 仮数部
 数値: $(-1)^S \times 2^e \times (0.f)$

図 3 . いろいろな浮動小数点形式

図 3 の 番目と 番目は、IEEE754 規格の浮動小数点形式 (単精度と倍精度) で、最近の CPU やプログラム言語でサポートする標準のフォーマットとなっています。

番目は、以前のパソコンでよく使われていた米国マイクロソフト社系の BASIC の浮動小数点形式です。番目は、8/16 ビットマイコンと一緒に使われていた浮動小数点演算ユニット Am9511A (AMD 社製) で採用されていた形式です。浮動小数点形式のデータはもちろん多バイト (単精度では 4 バイト、倍精度では 8 バイト) ですので、データの並び順も重要となります。

C 言語などの最近の高級プログラミング言語では、外部の機器などからバイナリデータをまとめて読み取りそれを指定した配列変数に自動的に格納してくれるコマンドや関数などが用意されています。この場合、データの並び順とデータの形式が読み取る側の PC と一致していることが重要となります。

固定小数点形式の場合では、普通データの並び順だけを問題とすればよいですが、浮動小数点形式のデータでは、データの並び順だけでなくデータ形式（どの形式か、単精度か倍精度か）も気にする必要があります。機器側としては、データ転送順や浮動小数点形式を指定することができるコマンドを用意しています。

近年発売する計測機器は、現在主流の Windows PC に合うようにデータ転送順はリトルエンディアン、浮動小数点形式は IEEE754 規格対応というのは主流となっています。

以上

(追記) 2009年1月9日

当社の FFT アナライザ (DS/CF シリーズ) の標準バイナリファイル (DAT ファイル) を他のソフトで読み込む場合も、データの並び (エンディアン) や浮動小数点の形式が重要となります。DAT ファイルは、今までのアナライザと互換性のため、すべてビッグエンディアンで保存されています。

- FAQ サイトの「CF 標準バイナリファイル仕様」を参照してください。

http://www.onosokki.co.jp/HP-WK/c_support/faq/ds2000/binaryfileformat.htm