

オーディオ/モータ/信号処理…タスク数個のゴリゴリリアルタイム用途にピタッ!

並列処理プロセッサ xCORE 徹底研究

MCUでも
FPGAでもない
ヘンテコッ!?

第6回 リアルタイム・スペアナの並列FFT処理
(最終回)

圓山 宗智

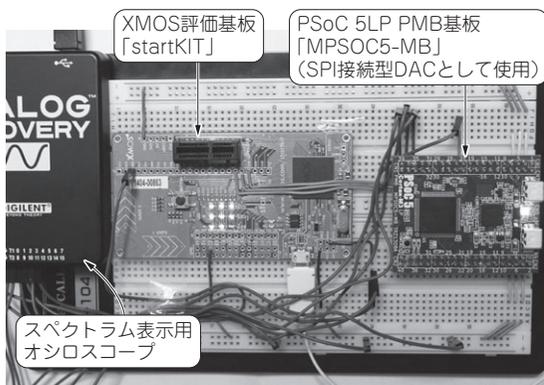


写真1 前回製作した125kHzリアルタイム並列FFT型スペアナ

前回は、スペクトラム・アナライザ(スペアナ)を製作し(写真1)、実現方法やハードウェア設計について説明しました。今回は、FFT演算を並列処理により高速化する技術など、ソフトウェアの工夫について解説します^{注1}。

FFTの並列処理方法

● FFTの並列化へのアイデア

FFT処理を並列演算させる方式はたくさん考案されています。今回は、図1に示すようなFFTのデータ・フロー図における各ステージに着目してみました。

この中で、バタフライ演算はいずれも独立しており、これらを複数のコアに配分すれば、並列処理できます。今回のFFT処理には4コアを割り振り、4並列で実行させます。

注1: 製作した並列FFT型スペアナのxCOREのプロジェクト「P03_FFT」(xTIMEcomposer用)と、シリアルD-A変換器として使ったPSoC5LPのプロジェクト「PSoC5LP_XMOSFFT」(PSoC Creator用)は、それぞれ本誌のWebページからダウンロードできる。

● ステージ間のデータ受け渡しを工夫

図1の各ステージでは32個の複素数バッファに対してバタフライ演算を行っています。個々の演算におけるバッファ入出力位置は、ステージごとによって変わってくるため、ステージ内処理を複数のコアに分散させつつ、いかにステージ間でデータを効率良く受け渡すかを工夫しました。

サンプル数 $N=32$ の場合を例として、今回実現したxCOREによる並列FFT処理方法を図2に示します。

● コア間のデータ受け渡し用バッファを8種類用意

まず、複素数の実部と虚部を格納できるバッファを用意します。バッファ内データは32ビット幅固定小数点であり、その実部と虚部を交互に格納しますので、バッファのサイズは要素数 $\times 2$ になります。

バッファは分割された8種類を用意し、それぞれ「00」「01」「10」「11」「20」「21」「30」「31」と名前を付けておきます。この例では、各バッファには $N/8$ サンプルずつデータが入ります。

バッファを8種類用意したのは、FFTの各ステージの処理を4個のコアに分担させるとき、ステージをまたぐときにコア間で受け渡すデータ列のうち、連続した固まり個数が最大8種類に収まる性質があったからです。これについては、図1をひたすら眺めながら図2を描いて理屈なく考えた、というのが正直なところです。実際、図1と図2のデータ・フローが同一になっていることを確認してみてください。

なお、製作したFFT型スペアナでは、 $N=1024$ に設定しています。

xCOREによる並列FFT型スペアナの実現

● タスクTask_ADCで入力波形をサンプリング

図2を使って、実際のxCOREによる並列FFT型スペアナの実現方法を説明します。

タスクTask_ADCで N 個のデータを250kHzの