

ZYBOで始めるリアルタイム 信号処理システム設計入門

① ARMベース SoC Zynq とは何かを理解する

岩田 利王
Toshio Iwata

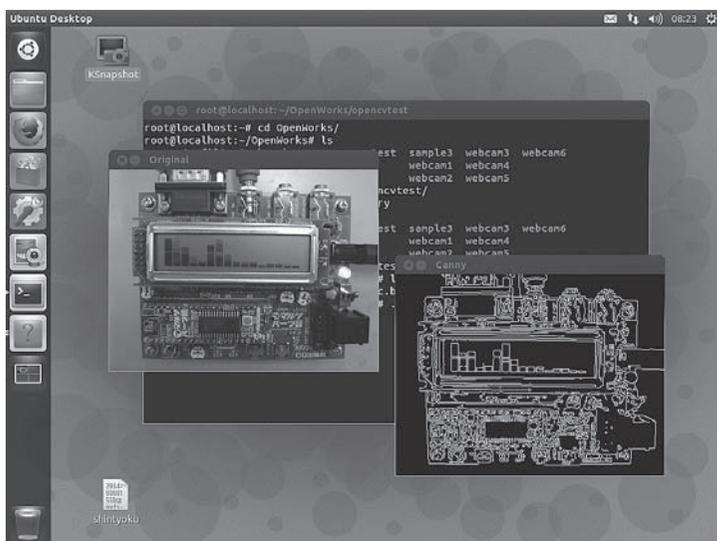


図1 こんなリッチなGUIでエッジ検出したい!

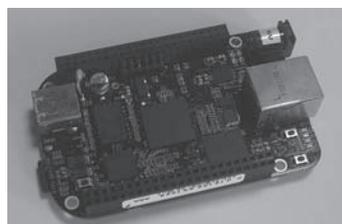


写真1 AM335x(ARM Cortex-A8コア)
搭載シングル・ボード・コンピュータ
BeagleBone Black

リッチなGUIと高速/リアルタイム信号処理を 同時にこなすシステムを作るには...

● 図1のような画像処理システムを作りたい... みなさんならどうしますか?

PCを使えば、図1のような画像のエッジ検出は比較的簡単にできそうです。でももっと小型で安価なボードで実現しなければならない... そんな状況での解決策を探ってみましょう。

▶ワンチップ・マイコンではパワー不足

図1のGUI(Graphical User Interface)環境はまるでPCのようです。その実現にはDDR(Double Data Rate)などの外部メモリが必要になるでしょうし、クロックも数百MHzは必要でしょう。従って、Cortex-M0クラスのワンチップ・マイコンではどう見ても間に合わなさそうです。

▶パワフルなシングル・ボード・コンピュータなら間に合いそう

写真1は、AM335x(ARM Cortex-A8コア)搭載の

BeagleBone Black(BeagleBoard.org製)です。クロックは1GHz、外部メモリは512MバイトDDR3なので何とかかなりそうです。実際、参考文献(1)ではこのボードにLinuxを搭載した例がいくつか紹介されています。

そして、Linux上にOpenCV(注1)などをインストールすれば、図1のような画像処理も可能になりそうです。

● 並行してリアルタイム信号処理も必要... さあ困った

さらに、図2(a)のようなシステムを考えます。Linuxを動かし、さらにリアルタイム信号処理をやるというものです。

▶次から次へとデータが入ってくる

扱うデータはリアルタイム、つまり音声信号のよう

注1: Open source Computer Vision library, インテルが開発するオープン・ソースの画像処理ライブラリ。言語はC/C++/Java/Pythonに対応。無償でダウンロードし使用できる。