Wireless @ Virginia Tech

CWT

Leveraging Embedded Heterogeneous Processors for SDR Applications

Almohanad Fayez

Qinqin Chen

Jeannette Nounagnon

Charles Bostian

Outline

- Introduction
- Hardware Platform
- Software Tools
- GNU Radio DSP Extension
- Performance Testing
- Future Work
- Questions

- **Introduction**
- Hardware Platform
- Software Tools
- GNU Radio DSP Extension
- Performance Testing
- Future Work
- Questions

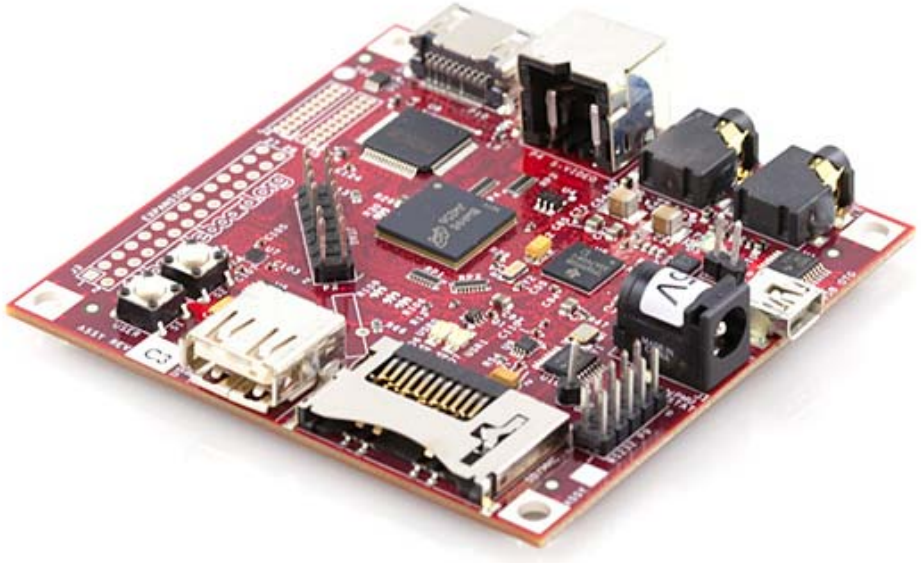
Introduction

- SDR applications vary in computational and power needs.
- Different computing devices needed for different scenarios.
  - General Purpose Processors (GPPs), Digital Signal Processors (DSP), and Field Programmable Gate Arrays (FPGAs)
- Targeting open source tools for SDR.

- Introduction
- **Hardware Platform**
- Software Tools
- GNU Radio DSP Extension
- Performance Testing
- Future Work
- Questions

Hardware Platform

- **Beagleboard**
  - **OMAP 3530 Processor**
    - **Arm Cortex A-8 (500 MHz)**
    - **C64x+ Fixed Point DSP (360 MHz)**
    - **NEON SIMD – media acceleration coprocessor**



- Introduction
- Hardware Platform
- **Software Tools**
- GNU Radio DSP Extension
- Performance Testing
- Future Work
- Questions

Software Tools

- OpenEmbedded
  - Build framework for embedded Linux.
- DSP/BIOS Link (DSPLink)
  - Library for inter-processor communication.
- C64x+ Compiler
  - DSP compiler.

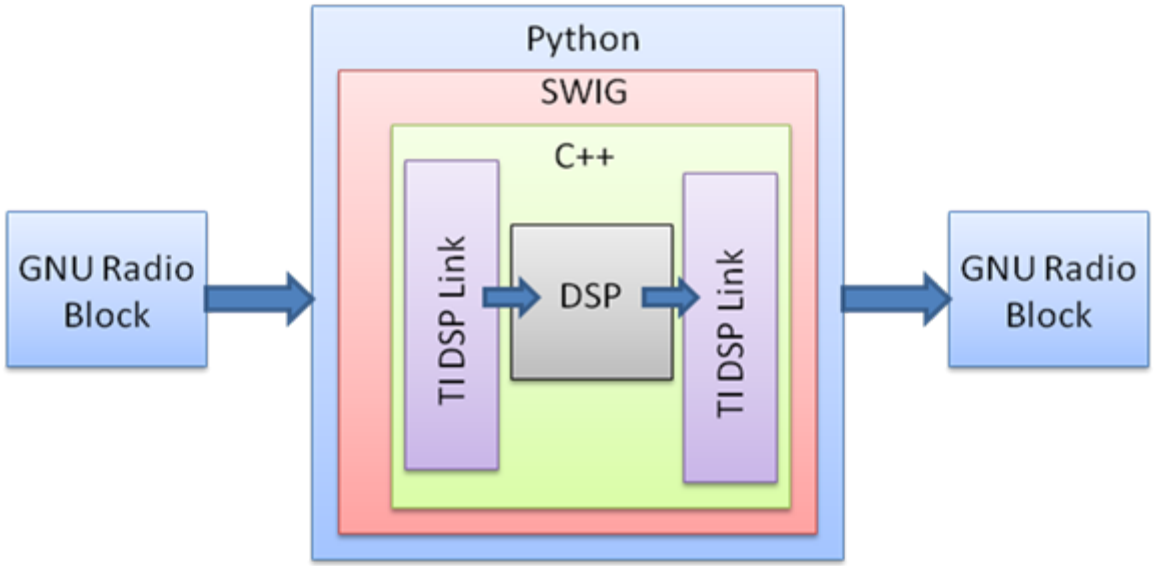
Software Tools

- DSP BIOS
  - DSP Real-Time Operating System.
- Code Composer
  - Texas Instrument IDE.
- GNU Radio
  - SDR software development kit.

- Introduction
- Hardware Platform
- Software Tools
- **GNU Radio DSP Extension**
- Performance Testing
- Future Work
- Questions

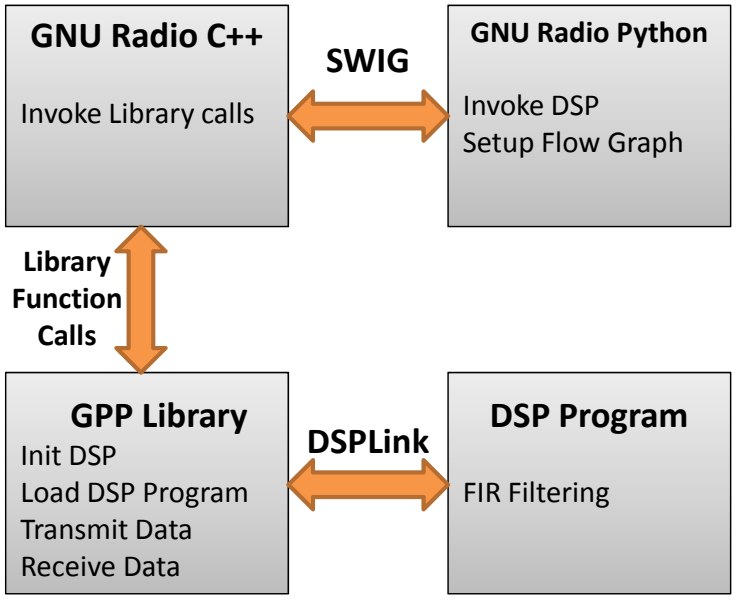
GNU Radio DSP Extension

- GNU Radio blocks which utilize the DSP.
- DSP blocks invoked using Python.
- DSP backend written in C/C++.
- SWIG allows C++ functions to be invoked from C++.



GNU Radio DSP Extension

- **GPP Library**
  - **Abstracts inter processor communication.**
- **DSP Program**
  - **Contains various DSP functions that can be invoked.**



GNU Radio DSP Extension

- Each GPP/DSP stream contains header tokens.
  - Identifies function to be implemented.
  - Unique ID.
  - Interpolation/decimation factors.
- Fixed/Floating point conversions.
  - GPP floating-point .
  - DSP fixed-point.

- Introduction
- Hardware Platform
- Software Tools
- GNU Radio DSP Extension
- **Performance Testing**
- Future Work
- Questions

Performance Testing

- **Test FIR filter implementation with IQ Input/Output and real coefficients.**
  - **300 Taps**
  - **Input data varying from 1.6k floating-point IQ pairs to 16M floating-point IQ pairs.**
- **Test program inputs IQ data from a file, filters, and saves them to another file.**

Performance Testing

- **DSP FIR implementation requires**
  - **Converting data between floating/fixed point format.**
  - **Physically copying data between GPP buffer to a shared GPP/DSP buffer.**

Performance Testing

- **GPP / DSP performance using Linux's *time* command.**
- **Time is measured as *real time* versus *CPU time*.**

Input (IQ pairs)	GPP Execution (seconds)	DSP Execution (seconds)	Speedup factor
$1.6*10^3$	1.916	2.109	0.91
$16*10^3$	3.63	2.19	1.66
$160*10^3$	20.519	2.494	8.23
$1.6*10^6$	189.308	6.237	30.35
$16*10^6$	1876.937	46.837	40.07

Performance Testing

- **Measuring DSP overheard.**

Input (IQ pairs)	Flowgraph setup IQ data I/O (seconds)	Loopback without conversion (seconds)	Loopback with Conversion (seconds)
$1.6*10^3$	1.718	2.11	2.128
$16*10^3$	1.713	2.153	2.151
$160*10^3$	1.782	2.361	2.487
$1.6*10^6$	3.159	5.085	5.735
$16*10^6$	22.987	37.919	42.11

- Introduction
- Hardware Platform
- Software Tools
- GNU Radio DSP Extension
- Performance Testing
- Future Work
- Questions

Future Work

- Pass pointers to DSP instead of explicitly copying buffers.
- Allocate GPP/DSP link interfaces on per block basis versus per DSP program basis.
- Provide feedback from DSP to GNU Radio scheduler.
- Create more blocks to run digital radios.

Questions

- **Thanks for listening … Questions ???**

Acknowledgements

National Institute of Justice
The Research, Development, and Evaluation Agency of the U.S. Department of Justice

–This project is support by Awards 2005-IJ-CX-K017 and 2009-SQ-B9-K011 awarded by the National Institute of Justice, Office of Justice Programs, US Department of Justice. The opinions, findings, and conclusions or recommendations expressed are those of the author and do not necessarily reflect the views of the Department of Justice.

National Science Foundation
WHERE DISCOVERIES BEGIN

–This material is based upon work supported by the National Science Foundation under Grant No. CNS-0519959. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation (NSF).