Design and Performance Tradeoffs in Digital Radio Processing Architectures

Mujun Song, Jason R. Pennington, Mark D. Silvius,
Ryan W. Thomas, Richard K. Martin, Charles W. Bostian

Outline

- Introduction
- Contributions
- Background
- Processor Architectures
- Experimental Comparison
- Conclusions

Introduction

- In the last few years, the software defined radio (SDR) world has produced an ever increasing number of variants
- Different processing architectures have a large tradespace in performance, cost, power, and ease-of-use
- For a researcher just entering into the SDR world, discerning what technologies are appropriate can be a daunting challenge

Contributions

- Investigate the tradeoffs between different processing architectures running in firmware and software
- Review current processing architectures and products on the market that are using them
- Define a series of metrics that can be used to evaluate and compare these architectures in a typical use case, e.g. spectrum sensing
- Use these results to show the potential advantages and disadvantages of these processing architectures

Processor Architectures

- In our paper, we define a category of digital radio that we term firmware defined radio (FDR):

  … a digital radio in which the operation is characterized by firmware as the primary functional actor. Firmware is defined as user-modifiable, read-only software that controls the hardware functionality

Background

- Typical use case: Spectrum Sensing
  - Filtering; Fast Fourier Transform (FFT); Computation of power spectral density (PSD)
- Applications to Dynamic Spectrum Access (DSA)
  - Indentify the presence of existing signals; then joining or avoiding communications
  - Collection of spectrum utilization information in a back to central database, e.g. “DSA Broker”
  - For military, the presence and locations of the signals, useful for exploiting or jamming adversaries

Processor Architectures

- Next, we review the processor hardware available for use in software defined radio and cognitive radios
  - General Purpose Processor (GPP)
  - Digital Signal Processor (DSP)
  - Field Programmable Gate Array (FPGA)

Processor Architectures

- General Purpose Processor (GPP)
  - The GPP is one of the foundational processor architectures in SDRs
  - High clock rate and large on-chip memories make GPPs appropriate for SDR
  - Because its main purpose is not only for signal processing, but also for performing various other system operations, it is less efficient and consumes more power than other devices.
  - However, since the GPP is continuously being developed and improved, some specialty processing features are now found on GPPs

Processor Architectures

- Digital Signal Processor (DSP)
  - DSPs are flexible and can be programmed with a HLL (high level language) such as C
  - Modifications and upgrades can be made easily by changing the HLL
  - Some DSPs have provisions for parallel execution of complex instructions; otherwise, complex instructions may need to be executed via a sequence of basic operations
  - Typically, run at slower clock rate than GPP
  - DSPs are comparatively cheaper than other hardware

Processor Architectures

- Field Programmable Gate Arrays (FPGAs)
  - Provide a matrix of reconfigurable logic resources on a single chip, programmable to user specified circuit
  - FPGAs can execute parallel operation; achieve a higher computational performance than a DSP
  - Lower engineering cost than an ASIC; reconfigurable and can leverage existing intellectual property
  - FPGAs are programmed using Hardware Description Languages (HDLs) which are often considered more difficult to learn

Processor Architectures

- Then, we highlight some of the most popular products, identify the processor architecture, and discuss some of the tradeoffs inherent in the products
  - Universal Software Radio Peripheral (USRP) v1/v2
  - Small Form Factor (SFF) SDR Platform by Lyrtech
  - Wireless Open Access Research Platform (WARP)
  - Kansas University Agile Radio (KUAR)
  - Berkeley BEE2/BEE3
  - TI Beagleboard
  - TMS320C6416 DSP Starter Kit

Processor Architectures

- Ettus Universal Software Radio Peripheral (USRP)
  - Revision 1 uses an Altera Cyclone 1 FPGA; Revision 2 uses a Xilinx Spartan FPGA
  - FPGAs used for basic, high sample-rate processing, like digital up-and-down conversion
  - The GPP on the host computer is used for lower-rate signal processing operations
  - In USRPv1, FPGA is not directing operation of platform
  - The USRPv2 can do more of both the high and low–rate processing operations in the FPGA, making it resemble more of a true FDR
  - $700 for the USRPv1; $1400 for the USRPv2

Processor Architectures

- Lyrtech Small Form Factor (SFF)
  - Uses both a Virtex 4 SX35 FPGA and a specialized embedded DSP called a digital media processor (DMP)
    - Consists of DSP and a GPP based on the ARM architecture
  - Platform has an FPGA, DSP and GPP, making it very difficult to categorize
  - Depending on how its processors are utilized, the SFF platform can be more SDR oriented or more FDR oriented
  - Development kit allows a user to develop C/C++ for DSP and GPP, or HDL code for FPGA
  - $3500 for board and $8500 for kit

Processor Architectures

- Wireless Open Access Research Platform (WARP) by Rice University
  - Version 2 uses Virtex 4 FX100 FPGA
  - Radio can communicate with a GPP-based PC, but the platform is designed to perform all radio calculation on the local FPGA
  - Virtex 4 FPGA model is chosen for high-performance logic applications
  - Board costs approximately $8500 for a basic single-antenna, single radio model

Processor Architectures

- Kansas University Agile Radio (KUAR)
  - SDR uses Virtex 2 Pro P30 FPGA in conjunction with a 1.4 GHz Pentium M GPP
  - Internal power supply with a battery pack, so it can operate self-contained
  - Higher level radio functionality is handled on the GPP, consider it a close to SDR than FDR
- Berkeley Emulation Engine (BEE)
  - BEE2 uses 5 FPGAs all Virtex 2 Pro P70s)
  - BEE3 uses 4 FPGAs all Virtex 5s, LXT/SXT/FXT varieties
  - Purpose of this platform is to make a FPGA base computer system, but a radio that utilizes a BEE would be considered a FDR

Processor Architectures

- TI Beagleboard
  - The Beagleboard has an OMAP3530 application processor featuring the ARM Cortex-A8
    - The OMAP processor contains both a GPP (ARM) and a DSP (TI architecture)
  - This GPP board is much cheaper than a host PC which is generally used with radio frontends such as the USR
  - Even though the performance of the GPP on the Beagleboard is lower than a GPP in a laptop, the DSP can make up the deficiency

Processor Architectures

- TMS320C6416 DSP Starter Kit (DSK)
  - A low-cost development platform used for the development of high performance digital signal processing applications
  - Uses the TMS320Cx DSP chipset
  - Features USB communications for off-board connectivity
  - Compatible with TI's Code Composer Studio IDE and eXpressDSP Software
  - Hardware board sells for $400-500

Processor Architectures

- Table summarizes these products
  - Except for TI DSK6416, every platform uses FPGAs in some capacity
  - Some platforms use not only FPGAs, but also include GPPs and/or DSPs.
  - The only platform that uses only FPGAs is the WARP

Table 2. Summary of Market Research

	USRP ½	Lyrtech	RICE/WARP	KUAR	BEE2/BEE3	TI DSK6416
Cost	$700 / $1400	$10,395	$3500	Not for sale	Not figured out	$495
FPGA	Cyclone 1 EP1C12 / Xilinx Spartan 3-2000	Virtex-4 SX35	Virtex-2 Pro P70 (V.1) / Virtex-4 FX100 (V.2)	Virtex-2 Pro V20	Virtex-2 Pro P70/ Virtex-5 (LX/SX)	•
Other elements (signal processing)	Host computer(GPP)	DM6446 DMP SoC from TI (DSP+MPU)	•	PowerPC 405		TMS320C6416

Experimental Comparison

- We define metrics to characterize and evaluate SDR and FDRs
  - Price of Hardware
  - Performance
  - Fidelity
  - Price of Software
  - Power
  - Development Difficulty

Experimental Comparison

- Price of Hardware
  - The approximate retail cost of a single instance of the processing chip in U.S. dollars
- Performance
  - The wall clock time required (in seconds) to complete one sense cycle (filter, FFT, and detect), once the A/D samples are in local registers (lower is better)
- Fidelity
  - The dB lost over the SNR of floating point operation resulting from rounding and truncation errors

Experimental Comparison

- Price of Software
  - The approximate commercial single seat license price in U.S. dollars of the standard development kit for the platform
- Power
  - The number of Watts consumed by the processing chip alone; it excludes the front end power and support hardware power requirements
- Development Difficulty
  - Qualitative measure; based on our experience in developing systems on the platforms as low, medium, or high

Experimental Comparison

- Architectures used in comparison
  - Linux PC laptop which used an Intel Core2Dou GPP
  - Texas Instruments 6416 DSP development board with a TMS320C6416 DSP processor,
  - WARP development board with Virtex 4 FX100 FPGA
- Use case: spectrum sensing, signal analysis
  - After collecting digital samples, we signal analysis
  - Filtering, Fast Fourier Transform (FFT), computation of power spectral density (PSD); signal detection via threshold test

Experimental Comparison

- Experiment Setup: Spectrum sensing signal analysis

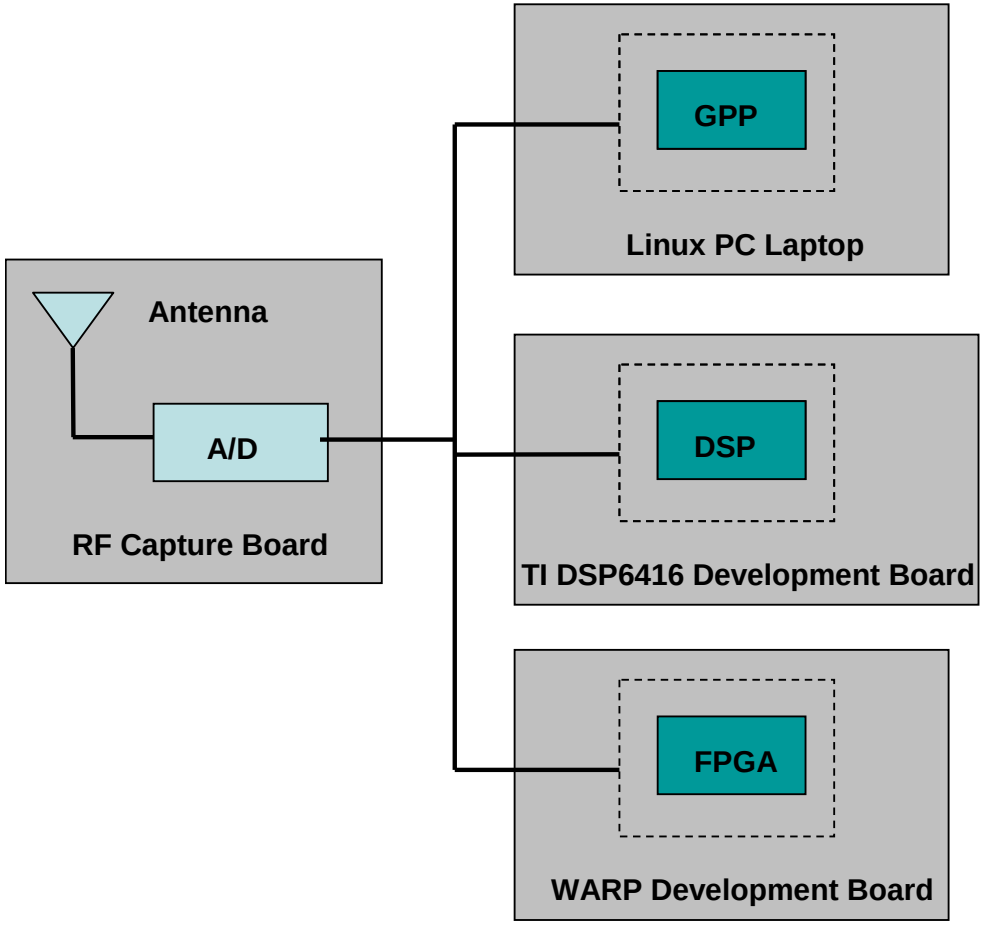


Figure 1. Experimental Setup

Experimental Comparison

- Table enumerates strengths/weaknesses
  - GPP system had high fidelity, low difficulty of development
  - DSP system had low cost, low power requirements, and moderate ease of use
  - FPGA had the highest performance, high fidelity, and low power constraints

Table 3. Architecture Evaluation

Evaluation	Price of Hardware (Dollars)	Performance (ns)	Fidelity (dB)	Price of Software (Dollars)	Power (Watts)	Development Difficulty (opinion)
GPP	2,000	100-200	-302	2,000	20-50	Low
DSP	231	406.111	-48	445	1.5	Medium
FPGA	2,188	7.4322	-302	4,000	1.44	High

Experimental Comparison

- Table enumerates strengths/weaknesses
  - GPP system had high fidelity, low difficulty of development
  - DSP system had low cost, low power requirements, and moderate ease of use
  - FPGA had the highest performance, high fidelity, and low power constraints

Table 3. Architecture Evaluation

Best

Evaluation	Price of Hardware (Dollars)	Performance (ns)	Fidelity (dB)	Price of Software (Dollars)	Power (Watts)	Development Difficulty (opinion)
GPP	2,000	100-200	-302	2,000	20-50	Low
DSP	231	406.111	-48	445	1.5	Medium
FPGA	2,188	7.4322	-302	4,000	1.44	High

Experimental Comparison

- Experimental Results
  - Next, we plotted these metrics in a series of spider plots in Figures 2-4, to visually represent the trends
  - The better, a more desirable metric value is represented by a larger quantity on the plot

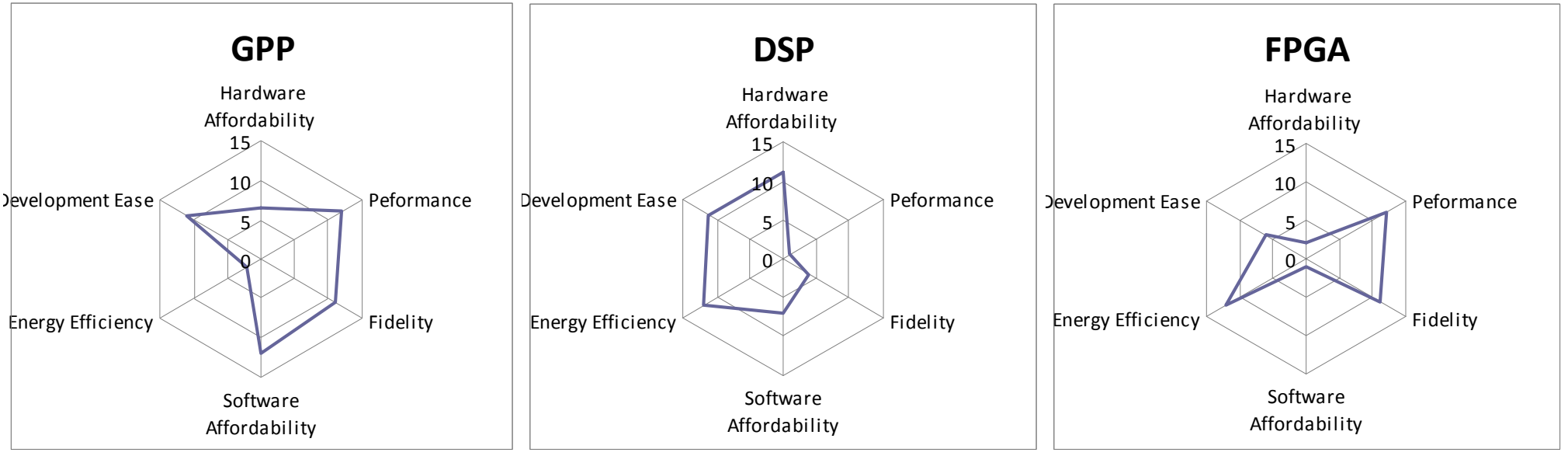


Figure 2-4. Experimental Results

Experimental Comparison

- Experimental Results
  - Next, we plotted these metrics in a series of spider plots in Figures 2-4, to visually represent the trends
  - The better, a more desirable metric value is represented by a larger quantity on the plot

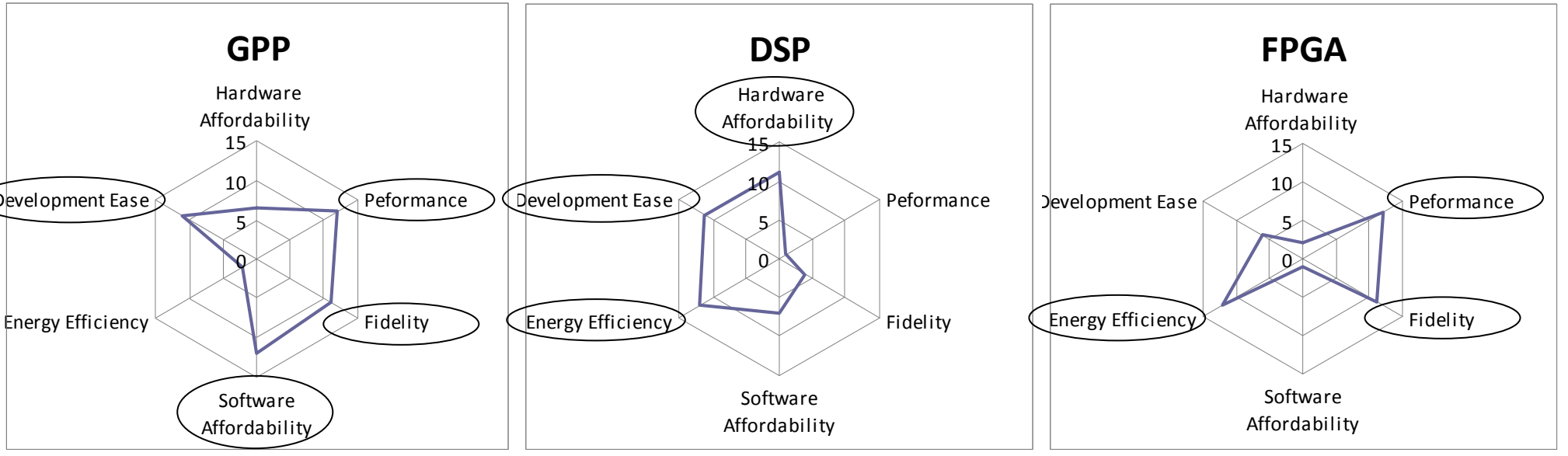


Figure 2-4. Experimental Results

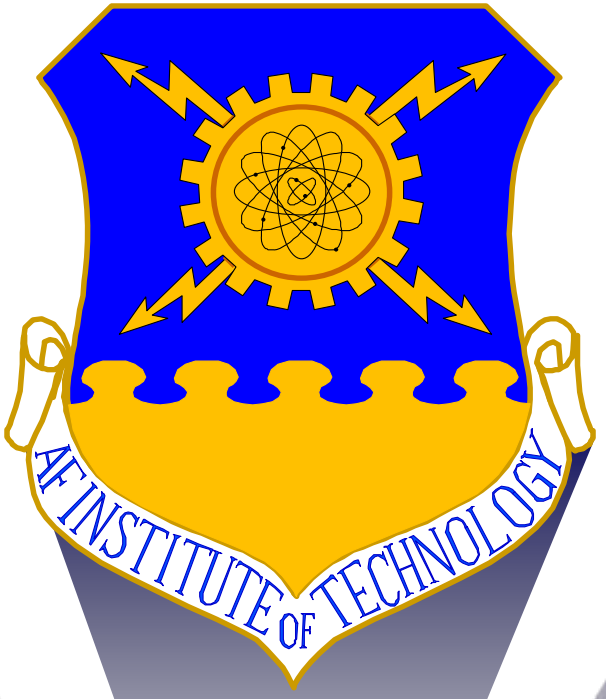
Conclusions

- Processor architectures each have their own strengths and weaknesses, and how the optimum choice of a processor for a system may depend on that system's application
- For example
  - The GPP architecture may be best suited for ground-based communication and spectrum sensing applications
  - In mobile or vehicle mounted devices FPGAs may be best suited
  - In battery-powered sensor network application, a DSP system may be the best choice
- Better understanding these design and performance tradeoffs will assist the designer when selecting a processor architecture for his/her specific system application

Acknowledgement

- This work funded in part by the Air Force Research Labs, Sensors Directorate and the Air Force Office of Scientific Research. The views expressed in this paper are those of the authors, and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government


AIR EDUCATION & TRAINING COMMAND


AF INSTITUTE OF TECHNOLOGY


AIR UNIVERSITY

Questions?

Extra

Processor Architectures

- FPGA Evolution Trends
  - Increasing resources (termed slices)
  - The use of specialized embedded blocks, serving to improve delay, power, and area if utilized by the application, but waste area and power if unused
  - Device families with additional mixes of features

Table 1. Evolution of FPGA Technology [8-11]

	LUTs (Per a slice)	Input capacity of LUT	Flip-flops (Per slice)	Slices (per CLB)	DSP	Capability of multiplier in DSP
Virtex-2	2	4-input	2	4	None	None
Virtex-4	2	4-input	2	4	32-512	18X18 multiplier
Virtex-5	4	6-input / 5 input-dual output	4	2	32-512	25X18 multiplier
Virtex-6	4	6-input / 5 input-dual output	8	2	288-2016	25X18 multiplier