S(oC)DR: Interfacing Between a System-on-Chip and Software Defined Radio

Ajay Thakkar
Electrical and Computer Engineering
Stevens Institute of Technology
Hoboken, New Jersey
athakka5@stevens.edu
Advised by Professor Bernard Yett

***Abstract*—Placeholder**

I. Introduction

Many modern spectrum tasks require portability and computational power to solve complex signal processing tasks in dynamic environments. Applications like remote sensing and tactical communications require self sufficient RF systems that can operate on the edge while employing modern spectrum tasks such as spectrum sensing and signal classification. Creating such devices usually requires custom hardware and software that are tailored to the application, which is both a costly and non-reusable solution. Technology such as Software Defined Radios (SDR) and System-on-Chips (SoC) can be leveraged in these applications to form fast, reprogrammable, portable, and cost effective systems.

SDRs are flexible RF systems that allow for digital reconfigurability to change its operational parameters while running. Their novelty lies in the interface between a General-Purpose Processor (GPP) and an RF front end, allowing the user to digitally control the front end's operating frequency, gain, sampling rate, etc. This GPP also allows for a personal computer to communicate with the system to send commands and retrieve return data for further processing. Software such as GNU Radio and MATLAB include packages to communicate with popular SDRs as well as signal processing and visualization blocks that can be tailored to a wide variety of RF tasks. This SDR ecosystem has been used extensively in modern research for prototyping RF systems, like Hu et al. utilizing the Pluto SDR platform as well as MATLAB's deep learning toolbox to implement a Convolutional Neural Network (CNN) for signal classification of the Pluto SDR's return signal. However, since many of these studies use a personal computer for their data processing which have a large form factor and utilize a lot of power, these systems are not viable for field testing or deployment.

To that end, many of these same signal processing applications such as CNNs have been implemented on programmable logic platforms like SoCs. SoCs contain both Programmable Logic (PL) and a Processing System (PS), essentially integrating both FPGA fabric and general purpose processors on the same chip and allowing them to work in tandem. Amin et al. were able to implement a CNN for signal classification on programmable logic, achieving an accuracy of 99.98% at 84139 frames per second with only 4.4W of power consumption, showing how aforementioned SDR research can utilize SoCs to lower power consumption and increase portability while still retaining the same processing power as a personal computer. This paper endeavors to create an interface between commercial SoCs and SDRs to provide a proof-of-concept for low-cost, low power, portable RF systems that can operate on the edge and be used in settings other than a lab environment.

The novel contributions of this paper can be summarized as follows:

- Generate a PetaLinux image that can be deployed on the ZYBO SoC Development Board with necessary libraries, packages, and SDR utilities
- Create a hardware platform in Vivado that can transfer HackRF I/Q data to and from the PS
- Compile a custom interface to be used in the Linux image which can control the HackRF, interface with the hardware platform, and communicate the data over a UDP server
- Write an upstream program that can connect to the UDP server on the ZYBO, collect and visualize data, and send commands for controlling the HackRF

This paper is organized as follows: Section 2 provides a formally defined problem statement that is specific to the proof-of-concept scope of this project. Section 3 presents the work done in the project, including block diagrams, photos of the lab setup, and the hardware and software that was created for the project.

II. Problem Statement

Most RF systems which include a software defined radio rely on personal computers and accompanying software for controlling the SDR and performing complex signal processing tasks. However, for applications such as tactical communications or remote sensing, portability and low power is required for these systems to be viable in field testing and deployment. Therefore, this paper aims to show the replacement of a personal computer with an SoC, which serves as a low power, low cost, and portable alternative with the same computational capability.

The Zybo is a cheap Zynq SoC development board made by Digilent that features Artix FPGA fabric, dual-core ARM A9 processors, and different peripherals that can be used for embedded applications. These Zynq SoC's can run embedded linux with the help of Xilinx's PetaLinux toolkit, allowing for community SDR applications like PySDR and the HackRF API to be leveraged when controlling the SDR. A user can also leverage the FPGA fabric within the SoC to create fast parallelized signal processing capabilities that not even a desktop can achieve. To this end, I propose S(oC)DR, a Zynq-based framework that abstracts the complexities of interfacing with and controlling the HackRF while enabling efficient parallel signal processing.

References

[1] D. M. Molla, H. Badis, L. George, and M. Berbineau, "Software defined radio platforms for wireless technologies," *IEEE Access*, vol. 10, pp. 26203–26229, 2022.

[2] Z. Zhang, W. Zhao, Y. Huang, J. An, and Y. Zhu, "Implementation of doa estimation system based on hackrf one," *International Journal of Antennas and Propagation*, vol. 2022, no. 1, p. 7901714, 2022.

[3] Y. Manasa, D. Dharun, U. Vamshi, and M. Gowtham, "Implementation of ofdm using gnu radio with hackrf one and rtl-sdr," in *2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS)*, pp. 1–8, 2024.

[4] U. M. Butt, S. A. Khan, A. Ullah, A. Khaliq, P. Reviriego, and A. Zahir, "Towards low latency and resource-efficient fpga implementations of the music algorithm for direction of arrival estimation," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 8, pp. 3351–3362, 2021.

[5] H. Nguyen and S. Nguyen, "Fpga-based implementation and evaluation of realtime ofdm phase compensation in 5g," in *2021 International Conference on Advanced Technologies for Communications (ATC)*, pp. 131–134, 2021.

[6] Y. Xia, Z. Ma, and Z. Huang, "Over-the-air radar emitter signal classification based on sdr," in *2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP)*, pp. 403–408, 2021.

[7] R. A. Amin, M. S. Amran Hossain, L. T. Schmid, V. Wiese, and R. Obermaisser, "Power efficient real-time traffic signal classification for autonomous driving using fpgas," in *2024 6th International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, pp. 1–5, 2024.

[8] L. Hu, H. Jiang, R. Lu, and C. Liu, "Signal classification in realtime based on sdr using convolutional neural network," in *2021 IEEE 2nd International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA)*, vol. 2, pp. 893–898, 2021.