Technical Review Paper Evaluation Form

(attach this form as the cover page for your report)

Student Name: Harrison Zhang

Project Advisor: Jennifer Hasler

Team Name: Music Synthesizer with FPAAs

Team Members: Chris Walds, Kristyn DiGiovanni, Jongheon Park, Sriram Pulavarty, Justin Kelley, Yewon Kim

_____ / 30 Technical Content

- Current state-of-the-art and commercial products
- Underlying technology
- Implementation of the technology
- Overall quality of the technical summary

_____ / 30 Use of Technical Reference Sources

- Appropriate number of sources (at least six)
- Sufficient number of source types (at least four)
- Quality of the sources
- Appropriate citations in body of text
- Reference list in proper format

_____ / 40 Effectiveness of Writing, Organization, and Development of Content

- Introductory paragraph
- Clear flow of information
- Organization
- Grammar, spelling, punctuation
- Style, readability, audience appropriateness, conformance to standards

_____ **/ 100 Total - Technical Review Paper**

USB Interface for Android Smartphone to FPAA Data Transfer

Introduction

In the design of modern technology, the ability to transfer data between a host and peripheral device is a necessity to develop systems of any scale. A large-scale embedded system might use SPI or I2C protocol to transfer data internally between chips, or a DMA might be used to transfer pixel graphic data to the screen of a mobile video game device; there are countless technologies that rely on a number of interfaces to transfer data between subcomponents of the device. Clearly, the ability to reliably communicate information within a system is an absolute necessity and it serves as the backbone behind the dataflow of any design before data can even be processed and used. This paper is a review of the use of a USB interface between an Android smartphone and FPAA for music synthesis applications.

Commercial Applications/Availability

FPGAs have offered a configurable medium for DSP applications, but more recently FPAAs have been studied as an alternative. While FPGAs rely on the conversion of data from real to integer for use in a digital setting, an FPAA bypasses this process while having the ability to rely on multiple functional blocks that already exist in the device fabric for the control system(s) and logic required to process the data [1]. However, FPAAs are still an early technology and not many units are actively manufactured and distributed (compared to FPGAs). Some research into the availability of FPAAs revealed that they are produced and sold by only a few companies like Anadigm and Okika Technologies. Anadigm's latest FPAA board, the AN231E04, offers a dynamically reconfigurable analog signal processor [2] for $200 (single evaluation kit). The options for commercially available FPAAs are already scarce, but additionally, none offer the I/O capabilities desired to interface with an Android smartphone.

Such an application-specific FPAA interface won't be found on the market, but it has been studied by Dr. Hasler's previous research groups. In the past, the group researched the development of a USB interface to allow students to program and interact with an FPAA in an educational setting. A laptop would be used to power and communicate with the device [3]. This application shares an on-board USB interface with our application, however there are key differences in the two designs. Rather than interfacing with a laptop, the team's design seeks to interface with an Android smartphone. Additionally, programmability of the FPAA would not fall under the functionality of our interface; the smartphone would provide software control for the synthesizer functionality, but hardware configuration would be isolated. Another design, while also interfacing with a laptop, abstracts FPAA control system logic and hardware onto a separate board which communicates with the actual FPAA IC board [4].

Technology Functionality

USB is a flexible and widely-used interface for communication and power supply between a host device and a peripheral device. For an Android smartphone-specific design, a micro USB port is expected to interface with the USB port on the FPAA control board. The most common application of a micro USB to USB interface would be the cable used to charge smartphones and transfer files between a host PC and the smartphone. For the team's application, an Android smartphone will act as host by controlling the music synthesizer on the FPAA.

The USB On-the-Go (OTG) protocol would be most suitable for the team's purposes as it "allows two USB devise to talk to each other without requiring the services of a personal computer" [5]. USB OTG also allows the two devices to swap host and peripheral roles as needed, so the smartphone could communicate synthesizer control to the FPAA which could in turn return audio data to the phone. For a device to act as host, it must support the ability to send start of frame (SOF), SETUP, IN, and OUT packets as well provide a USB signal reset and USB power management [5]. A Host Negotiation Protocol (HNP) is used to swap roles between the devices by using an additional ID pin which can be grounded and left floating to indicate which device is host and peripheral respectively. A Session Request Protocol (SRP) allows the peripheral to request power and communication from its host. To be OTG compatible, an OTG transceiver must have switchable pull-up and pull-down resistors (to switch host-peripheral functionality), power control circuitry, and an ID pin.

While USB OTG provides the infrastructure for data transfer between devices, the protocol will not operate correctly as is; for the smartphone the driver will have to be modified to properly interact with the device's software (an app for example) and for the FPAA control board, some internal logic must occur to map incoming and outgoing data bits to the proper pin connections at the USB interface. Commercially available USB OTG cables will come with software compatible with the device but there is also open-source code in C that defines certain parts of the driver software that would need to be modified to be compatible with the team's application. Important portions of the code would include status definitions, states, addressability of data, etc. [6].

Implementation

In a previous section, the latter design by Dr. Hasler's research group more closely matches the design for the team's intended application. In that design, a USB to serial converter IC is used to interface with the device [4]. Ideally, we would be able to treat the FPAA control board as any other digital peripheral via USB as a standard interface [8]. For this application, work must be done in both a software and hardware setting to reach a workable design. The proposed micro USB OTG to FPAA USB interface is an ideal situation, however in reality it is likely that by itself, the FPAA control board would not be able to support host functionality. In such a case, the hardware would have to be modified via internal

digital logic to support the HNP portion of the overall OTG protocol. To accomplish this, a mix of protocols may be used (for example SPI might be used to interface with other areas of the board) along with FPAA functional blocks to send analog data over a digital serial interface. Software will also have to be modified as a custom device driver would be needed to allow the smartphone to properly interface with the FPAA control board. As there is a mix of responsibilities involved with this technical aspect of the design project, there are numerous options available for EE or CE majors looking to get involved with hardware and/or software.

References

[1] R. Selow, H. S. Lopes, and C. R. E. Lima, "A comparison of FPGA and FPAA technologies for a signal processing application," in *2009 International Conference on Field Programmable Logic and Applications, Prague, Czech Republic,* August 31, 2009. [Online]. Available: https://ieeexplore.ieee.org/document/5272306/authors#authors [Accessed: Sept. 28, 2019].

[2] Anadigm, "AN231E04 Datasheet Rev 1.2," 2014. [Online]. Available: https://www.anadigm.com/_doc/DS231000-U001.pdf [Accessed: Sept. 27, 2019].

[3] J. Hasler, A. Natarajan, S. Shah, and S. Kim, "SoC FPAA Immeresed Junior Level Circuits Course," *MSE,* May 2017. [Online]. Available: http://hasler.ece.gatech.edu/SoCFPAA.html [Accessed: Sept. 28, 2019].

[4] J. Hasler, S. Kim, S. Shah, F. Adil, M. Collins, S. Koziol, and S. Nease, "Transforming Mixed-Signal Circuits Class through SoC FPAA IC, PCB, and Toolset," *European Workshop on Microelectronics Education, Southampton*. May 2016. [Online]. Available: http://hasler.ece.gatech.edu/SoCFPAA/European_MSE_conference_c.pdf [Accessed: Sept. 28, 2019].

[5] Maxim Integrated, "USB ON-THE-GO (OTG): USES AND SUPPORT," 2015. Available: https://www.maximintegrated.com/en/design/technical-documents/tutorials/1/1822.html [Accessed: Sept. 28, 2019].

[6] STMicroelectronics, "UM1734 User manual," Feb. 2019. [Online]. Available: https://www.st.com/content/ccc/resource/technical/document/user_manual/cf/38/e5/b5/dd/1d/4c/09/DM00108129.pdf/files/DM00108129.pdf/jcr:content/translations/en.DM00108129.pdf

[Accessed: Sept. 28, 2019].

[7] Z. Lin, D. Meng, and Q. Zhong, “Design and Implementation of OTG Communication System Terminal Based on USB,” *2011 International Conference on Control, Automation, and Systems Engineering (CASE), Singapore*, 30 August 2011. [Online]. Available: https://ieeexplore.ieee.org/document/5997526 [Accessed: Sept. 28, 2019].

[8] J. Hasler, S. Shah, S. Kim, I. Lal, and M. Collins, "Remote FPAA System Setup Enabling Wide Accessibility of Configurable devices," *Journal of Low Power Electronics Applications*, June 2016. [Online]. Available: http://hasler.ece.gatech.edu/SoCFPAA.html [Accessed: Sept. 28, 2019].