Field–Programmable Analog Arrays Enable Analog Signal Processing Education

Tyson S. Hall[1] and Paul Hasler[2]

Abstract–In a laboratory environment, the practicality and scope of experiments is constrained by time and financial resources. In the digital hardware design arena, the development of programmable logic devices, such as field–programmable gate arrays (FPGAs), has greatly enhanced the students ability to design and synthesize complete systems within a short period of time and at a reasonable cost. Unfortunately, analog circuit design and signal processing have not enjoyed similar advances. However, new advances in field–programmable analog arrays (FPAAs) have created many new opportunities in analog circuit design and signal processing education. This paper will investigate the usefulness of these FPAAs as viable pedagogical tools. It will also explore the new methodologies in analog signal processing education that are available when FPAAs are brought into the classroom.

Keywords: Field–programmable analog array, FPAA, analog signal processing, reconfigurable analog circuits.

I. Educational Impact of Programmable Analog ICs

In an academic setting, analog circuits are typically prototyped with discrete components, and larger designs are often tested by performing computer simulations only. Decisions on the size and scope of the experiments are often dictated by limited time available in the academic term and the–often minimal–laboratory budget. In digital laboratories, field–programmable gate arryas (FPGAs) have steadily decreased in price and increased in functionality, providing an ideal platform for digital experiments. Unfortunately, reconfigurable analog devices have not kept pace, and analog technologies available to classroom laboratories have remained relatively un-programmable. However, recent advances in analog circuit design have led to a novel class of field–programmable analog arrays (FPAAs) [1], [2], [3], [4]. These FPAAs are accurately programmable, continuous–time, analog devices that are capable of implementing full–scale analog signal processing systems.

FPAAs introduce new opportunities to improve analog circuit design and signal processing education. This is not to mean that FPAAs will replace all traditional analog circuit methodologies in the classroom. However, FPAAs can be an additional pedagogical resource. By providing students with an easily reconfigurable, general–purpose analog device, laboratory projects can grow in scope and functionality. Students can use FPAAs to implement and test multiple circuit designs within a single laboratory period. In minutes, a single FPAA device can be configured to implement several different circuit topologies that can be tested and compared. In addition, modern FPAAs can contain analog–to–digital converters that ease the interfacing of analog systems with digital logic implemented on FPGAs and/or microcontrollers.

New effort in FPAA research and development has created many new opportunities in analog circuit design and signal processing education. The devices discussed here are really a class of FPAAs based on the same underlying floating–gate technology and not a single commercial product. Several FPAAs of varying sizes, complexities, and design have been fabricated. This paper is motivated by a desire to utilize this emerging research within the classroom to enhance analog circuit and signal processing education. A number of different FPAA ideas and concepts will be mentioned with the hope that discussions within the engineering academic community can contribute to a refined FPAA design that is optimized for use in undergraduate and graduate laboratories. This paper will investigate the usefulness of these FPAAs as viable pedagogical tools, and it will also explore the new methodologies in analog signal processing education that are available when FPAAs are brought into the classroom.
The remainder of this paper is organized into four sections. An overview of large–scale FPAAs based on floating–gate transistors is included in Section 2. Section 3 discusses analog circuit curricula and the potential for augmenting

[1] Southern Adventist University, Collegedale, TN 37315–0370, tyson@southern.edu
[2] Georgia Institute of Technology, Atlanta, GA 30332–0250, phasler@ece.gatech.edu

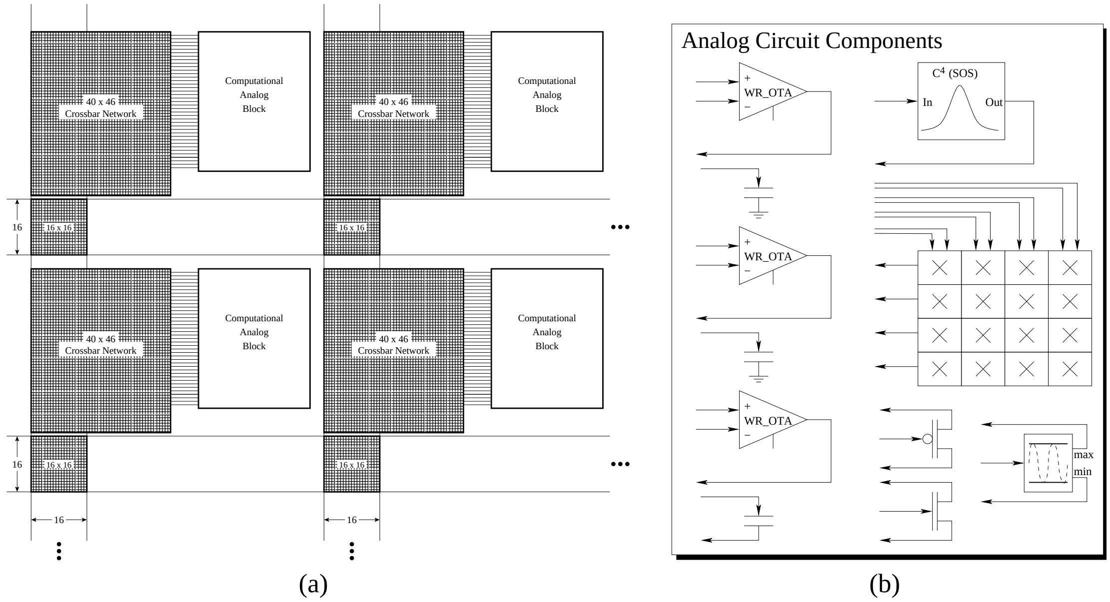


Fig. 1. (a) This is the overall block diagram for a large–scale FPAA. The switching interconnects are fully connectable crossbar networks built using floating–gate transistors. (b) There are two slightly different Computational Analog Blocks (CABs) on this FPAA. The large or regular CAB contains a four–by–four matrix multiplier, three wide–range operational transconductance amplifiers (OTAs), three fixed–value capacitors, a capacitively coupled current conveyor (C^4) second–order section (SOS), a peak detector, and two FET transistors. The small CAB is the same except it does not include the four–by–four matrix multiplier. This design includes large CABs at the top and bottom of each column and small CABs in–between.

traditional approaches with FPAAs. In Section 4, three different topical areas—embedded systems, neuromorphic VLSI, and cooperative analog/digital signal processing—are explored to show the breadth of educational arenas that can benefit from using FPAAs in the classroom. Finally, Section 5 concludes this paper with a summary of its discussion.

II. Field–Programmable Analog Arrays

Reconfigurable hardware has long been of interest to circuit designers and engineers. In the digital domain, programmable logic devices (PLDs) have made a large impact on the development of custom digital chips by enabling a designer to try custom designs on easily reconfigurable hardware. Since their conception in the late 1960s and early 1970s, PLDs have evolved into today's high–density field–programmable gate arrays (FPGAs) [5], [6], [7]. Modern FPGAs are widely used in the lab for rapidly prototyping digital hardware as well as in production goods to decrease time–to–market and to allow products to be easily upgraded after being deployed. Progress in reconfigurable analog hardware, however, has been much slower. While early analog integrated circuits (ICs) were often tunable with adjustable biases, truly reconfigurable analog circuitry in the form of field–programmable analog arrays (FPAAs) did not emerge until the late 1980s [8], [9], and commercial offerings did not reach the market until 1996 [10]. Having been in the marketplace for nearly a decade, current FPAAs offerings have struggled to establish a solid market base. They have been plagued by poor performance, small sizes, and a lack of generality/functionality.

Fundamentally, FPAAs include two functions: routing and computation. The routing elements are typically networks of switches connected together by signal lines with the network architecture and switch types varying dramatically across different FPAAs. The switch networks then connect to the computational elements of the system. If there is more than one type of computational element, the computational elements are usually grouped together to form a computational analog block (CAB) that is analagous to the computational logic blocks found on FPGAs. Recently, a new class of FPAAs has been introduced that promises increased functionality, a large number of computational elements on a single IC, and performance specifications well in excess of that needed in most educational laboratories [1], [4]. These FPAAs are continuous–time devices that utilize floating–gate transistors as the programmable analog element [2].

Many example CABs can be imagined using this technology. Figure 1b shows one example CAB, whose functionality is enhanced by a mixture of medium– and coarse–grained computational blocks similar to many modern FPGA designs. The computational blocks were carefully selected to provide a sufficiently flexible, generic architecture while optimizing certain frequently used signal processing blocks. For generality, three operational transconductance amplifiers (OTAs) are included in each CAB. OTAs have already been shown to be effective at implementing a large class of systems including amplification, integration, filtering, multiplication, exponentiation, modulation, and other linear and non-linear functions [11], [12], [13], [14]. In addition, the two FET devices provide the ability to perform logarithmic and exponential functions, as well as convert back and forth between current and voltage. The three capacitors are fixed in value to minimize the size of the CAB and are primarily used on the outputs of the OTAs; however, they will be available for any purpose. The variable capacitor and/or current mirror banks found in some designs are not needed here, because the use of floating–gate transistors in the OTAs will give the user sufficient control in programming the transconductance of the amplifiers [2], [12]. Eliminating the capacitor banks creates a large savings in the area required for each CAB.

The high–level computational blocks used in this design are a capacitively coupled current conveyor (C^4) used as a bandpass filter module and the 4 x 4 vector–matrix multiplier block. In general, the C^4 module provides a straightforward method of sub–banding an incoming signal. This allows Fourier analysis analagous to performing a Fast Fourier Transform (FFT) in the digital domain. The vector–matrix multiplier block allows the user to perform a matrix transformation on the incoming signals. Together these blocks can be used like a Fourier processor [15], [16]. In addition, a peak detector is added to each CAB.

III. Enhancing Analog Circuit Education

FPAAs are a natural fit in analog circuit courses. They can be used in early circuits courses to provide students with a hands–on experience learning measurement techniques, characterizing basic components, and even designing simple analog systems. In later courses, FPAAs can be used to experiment with concepts in analog system design, integrated circuit (IC) design, and signal processing. With the proper mix of different computational elements with each CAB, specialized CABs, and peripheral blocks such as analog–to–digital converters (ADCs) and digital–to–analog converters (DACs), analog multiplexers, and analog shift registers, a single FPAA can be used in a wide variety of courses and experiments.

A. Basic Circuit Curricula

Most electrical and computer engineering (ECE) students are attracted to ECE by the possibility of building systems. Often students see that signal processing offers the ability to accomplish a wide range of tasks from audio and speech processing to video, wireless communications, and more. However, their first circuit courses are something between applied math and rote, ”plug and chug”–style instrumentation experiments. In reality, after several courses, the student really only knows how to make ”little wiggles into big wiggles.”

Teaching analog design at the undergraduate level is still mostly focused on analysis and design techniques. Early courses usually emphasize these fundamental concepts without giving students the motivation (i.e., ”the big picture”) as to why these details are important. Laboratory projects and homework assignments often take the form of design an audio–style amplifier or analyze five different op-amp flavors, and students rarely get the opportunity to see larger systems until a senior design course or graduate–level courses. While these fundamental techniques are extremely important and useful, they only give a small perspective of the analog circuit world. It seems more beneficial to students to teach these approaches within a broader context showing students a wide range of circuits and systems, which can motivate the necessity for learning the circuit design details.

FPAAs can be the tool to enable this broader system approach to basic circuit curricula. FPAAs for these courses could be fairly small (8–16 CABs). Initial labs could step students through basic data acquisition techniques using an FPAA that has been pre-configured by the instructor. (A sample characterization acquisition is shown in Fig. 2. Students can be introduced to basic transistor I–V characteristics, basic switch effects, and the first several circuits typical in these courses. With the addition of ADCs and DACs on the FPAA, computer analysis and acquisition can be made easier for the early courses. Of course, standard bench instruments (voltage meters, ammeters, oscilloscopes) can also be used with the analog input and output pins.

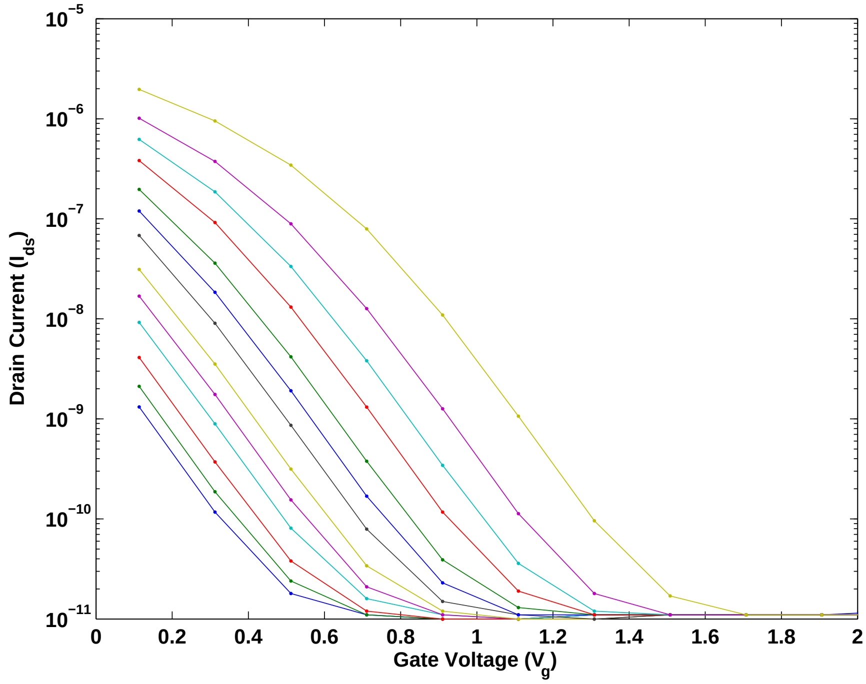


Fig. 2. This figure illustrates how the student can characterize basic devices on the FPAA over a wide range of operating conditions. Here, a typical I–V plot is taken on an floating–gate pFET switch residing on a test FPAA. Multiple plots are shown as the switch is programmed from the *off* state to the *on* state.

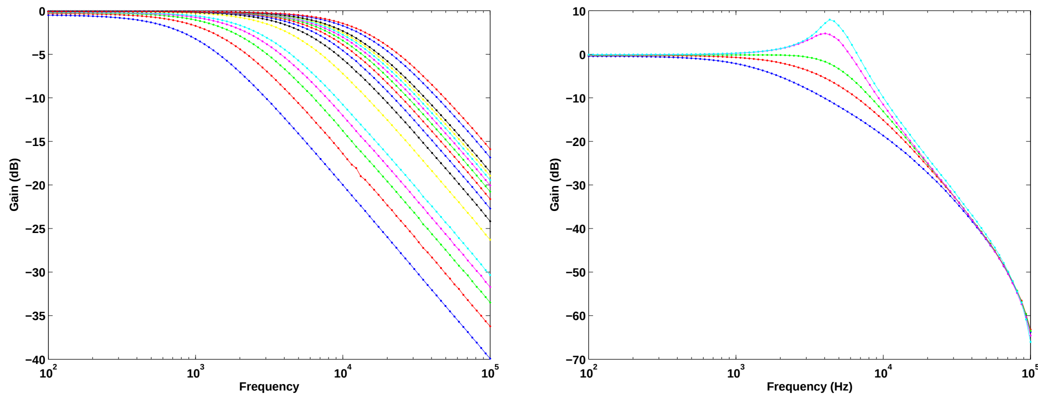
Later laboratories can continue to use the FPAAs. For example, when analog filtering is discussed simple RC filters can be compared to more complicated filter topologies synthesized on the FPAA. The instructor can download several different filter designs (showing varying topologies and filter orders) to a single FPAA and quickly illustrate the differences through both quantitative measurements and qualitative demonstrations (i.e., connect a portable CD player or computers sound card output to an input on the FPAA and an amplified speaker on an output). Figure 3 illustrates several example low–order filter designs that can be implemented on the FPAA. Biases on the FPAA are programmed to adjust the corner frequencies and Q peak of the different filters. The system also has enough flexibility to implement a graphic equalizer. Since the implementation is done on the FPAA, students should have plenty of time to experiment with system parameters and explore the effects of varying different biases, corner frequencies, Q peaks, etc.

B. **Integrated Circuit Curricula**

FPAA chips are important for training students specializing in analog circuit design at the Integrated Circuit (IC) level. The demand for analog IC engineers continues to increase; however, teaching this material is becoming increasingly more complex, resulting in the number of graduates not increasing significantly. Furthermore, courses based only on computer tools leave most students without basic IC fundamentals, thus shifting a greater responsibility for training to the employer. Typically, an analog IC engineer is considered trained only after a mentorship of two to four years.

The addition of having students test ICs–fabricated ahead of time–for a class, as well as participating in design projects directed towards fabricating ICs have shown positive results at the Georgia Institute of Technology. These activities have helped students start developing an intuition for analog integrated circuits. However, while intuition is the first step towards analog IC design, it still stops short of preparing students for creative IC design.

Students need to be involved in creative IC design projects that can be started and completed in a single semester, beginning in the first analog IC course. The combination of a large–scale FPAA IC, with an accompanying development board, would be a free–standing platform for students to design, develop, and experimentally test IC



(a) (b)

Fig. 3. The FPAA can implement a wide variety of different filter topologies. In addition, students can program the FPAA to modify key filter parameters, such as corner frequency and Q-peak. (a) A first–order filter (integrator) is shown with several different corner frequencies as set by computer software that programmed the FPAA. (b) A second–order filter is shown with variable Q peak and corner frequency. This plot shows the Q peak varying for a constant corner frequency.

circuits without any additional equipment. Of course, bench instruments can be added to enhance the measurement experience where desired. This approach enables rapid prototyping of analog IC circuits and helps develop student intuition and creativity in a fairly low–cost solution. Having a chip where the devices are very flexible, but with some constraints, allows students to gain an appreciation for key concepts in design.

A basic large–scale FPAA IC with a mixture of granularities allows for a range of design experiences, from single device behaviors to complex analog system circuits. A 16 CAB FPAA composed of 4 CABs of transistor elements, such as nFETs, pFETs, capacitors, and floating–gate transistors, and 12 CABs with higher granularity elements, like amplifiers and filters, should be sufficient for typical designs in a first or second semester IC course. These elements can be integrated with configurable DACs and ADCs. Since the FPAA is a sizable chip with a wide range of devices, students will get a clearer understanding of mismatch effects, as well as how to place devices to improve matching, and the effects of added parasitic capacitances due to the multiple paths through routing matrices. Enabling easier access to these IC design techniques empowers students to study these concepts earlier in their engineering curriculum.

IV. Enhancing Analog Signal Processing Education

Analog signal processing is a broad and diverse field of study. In this section, three different areas will be explored—embedded systems, neuromorphic VLSI, and cooperative analog/digital signal processing. These subjects are rarely linked together, but the inclusion of this diverse group of subjects is meant to demonstrate the wide–spread potential of introducing reprogrammable analog devices into classroom and laboratory activities.

A. Embedded Systems Curricula

The field of embedded systems is an increasingly popular area of study for students and an area of focus for educators [17], [18], [19]. Courses are emerging that focus on designing low–power systems that can contain reconfigurable hardware (FPGAs), embedded processors / software, digital signal processors, wireless communication, and an array of sensors. Pedagogically, these courses allow students to explore the trade–offs between different design modalities (i.e., hardware versus software, digital versus analog circuitry, dedicated versus reconfigurable hardware, etc.) and investigate an optimal solution given a set of constraints placed on the system (i.e., physical size, battery life, cost, limited memory capacity, etc.).

Analog circuitry is usually limited to the input and output sensors in typical embedded systems with ADCs and DACs placed as close to the sensors as possible leaving the majority of the system completely digital. This is understandable given the reprogrammable nature of digital technologies and ease of design. However, with the

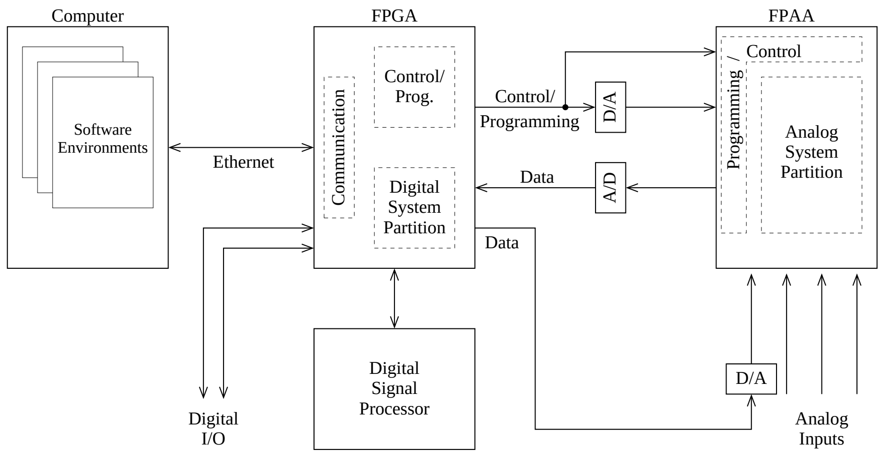


Fig. 4. This is a block diagram of an example mixed–signal development system. This system incorporates reconfigurable digital hardware, reconfigurable analog hardware, and a DSP microprocessor to provide a platform for experimenting with embedded systems. It allows a wide array of analog and digital hardware systems, as well as software systems to be prototyped.

appropriate CAD tools, FPAAs can bring these same advantages to analog circuit design. Adding analog circuitry (in the form of an FPAA) on the front–end of embedded systems can provide some significant advantages, particularly when power is a driving concern.

With the class of FPAAs introduced here, one can imagine implementing programmable filters, frequency decomposition, smoothing (and other signal conditioning), thesholding, peak detection, and more in the FPAA. With the addition of on–chip ADCs, the FPAA can be thought of as a ”smart ADC.” This term implies the FPAAs extensive computational capability in addition to traditional data conversion functionality. There is another subtlety at play in a smart ADC converter. By adding computational effort in the analog domain, the amount of data that needs to be converted may be drastically reduced thus requiring a simpler, smaller, slower, lower power ADC in the system.

Embedded systems curricula is often multi-disciplinary and can include students from electrical engineering, computer engineering, mechanical engineering, and computer science. In this environment, traditional analog circuitry is often shied away unless well understood by a particular student. However, FPAAs (with the proper CAD tools) can lower the learning barrier for analog design. Graphical software can allow students to configure the FPAA by connecting high–level functional blocks together and specifying design parameters such as cut–off frequency, Q peak, decay constants, etc. The low–level details of system implementation are thus abstracted away from the user. Analog CAD tools are still quite rudimentary, but research is currently under way to provide tools equivalent to those used to configure digital FPGAs.

B. Neuromorphic VLSI Curricula

The origins of Neuromorphic Engineering began as a class, and then a series of classes, taught at Caltech in the 1980s and 1990s, utilizing IC physics and circuit design as the bridge between neurobiology, mathematics, and engineering. In particular, neuromorphic engineering uses inspiration from the neural computations in analog IC design at various modeling levels, resulting in unique engineered systems and sometimes resulting in insights back to the neurobiological systems. Much of the current work in CMOS imagers (i.e., digital cameras), hearing aids, and cochlear implants has been affected by these approaches [20].

One would expect that the benefits of using FPAAs in teaching analog IC design would apply towards teaching neuromorphic engineering concepts, but requiring accessibility to a wider audience, including a range of engineers, neuroscientists, and mathematicians. Unlike typical analog IC design classes, where the circuit to be designed is known and its parameters specified, a neuromorphic design process often includes the design of several ICs to be tested to understand device, circuit, and basic system concepts. Because many of these concepts are hard to simulate or often not modeled by simulation tools, students typically fabricate several ICs, resulting in substantial time for

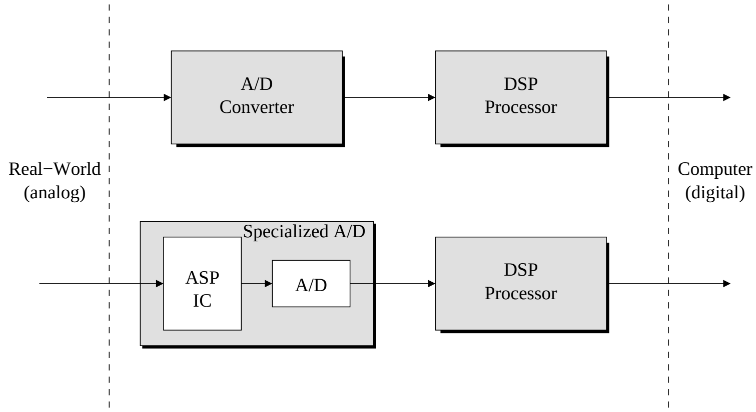


Fig. 5. In most traditional digital signal processing systems, the analog–to–digital converter (ADC) is placed as close to the real–world, analog inputs as possible. However, significant power savings can be achieved by moving some of the signal processing functionality into the analog domain (in front of the ADC). Conceptually, this analog signal processing can be combined with the ADC to form a "specialized ADC" as shown here.

each iteration due to IC fabrication times (approximately 3 months). A related issue is the amount of time and effort required for training students to be proficient at IC layout and to be skilled at building custom test setups to show system functionality.

Large–scale FPAA elements drastically reduces the learning and development time for designing these systems, even if the system eventually goes to a custom IC. An FPAA with 64 to 128 CABs, similar to CABs reported in [1], with ADCs, DACs, and on–chip current measurement for instrumentation as well as some specialized blocks to assist are sufficient to investigate most current neuromorphic engineering topics in a fraction of the time. One could further expand this concept to include specialized blocks including a reconfigurable set of biological channels [21], [22], [23], front–end cochlea models, on–chip sensors (e.g., pixel arrays for vision applications), and specialized digital interfaces, like Address Event Representation (AER) of neuron action potentials.

C. **Cooperative Analog/Digital Signal Processing**

Today's mobile computing environment has placed increased demands on low–power signal processing systems. This has created a renewed interest in analog signal processing functions. Cooperative Analog/Digital Signal Processing (CADSP) is a combined research and educational initiative aimed at investigating the partitioning of signal processing systems between the analog and digital domains. Most current signal processing systems that generate digital output place the ADC as close to the analog input signal as possible to take advantage of the computational flexibility available in digital processors (see Fig. 5). However, the development of large–scale FPAAs—and the CAD tools needed for their ease of use—would allow engineers the option of performing some of the computations in reconfigurable analog hardware prior to the ADC, resulting in both a simpler ADC and a substantially reduced computational load on the digital processors that follow. By leveraging the power efficiencies found in some analog circuitry, some analog signal processing systems have been shown to achieve as much as five orders of magnitude over typical DSP microprocessor implementations [24], [25], [26].

CADSP can provide an intriguing and valuable study for upper–divisional and graduate–level courses. As shown in Table I, a number of important signal processing systems and functions can be implemented in analog using FPAAs [1].

Functionality	DSP μP	Trad. Analog	Large–scale FPAA
Programmable	●	–	●
Monolithic Filters	○	●	●
Linear Scalar	●	○	●
Nonlinear Scalar	○	●	●
Vector–Matrix	○	○	●
Linear–phase Fltrs	●	–	○
Adaptivity	○	–	○
Tap Delay Lines	●	○	○

Key:
– = No or very limited support
○ = Possible
● = Efficient, well–suited to technology

TABLE I

SUMMARY OF SIGNAL PROCESSING FUNCTIONALITY

V. Conclusion

FPAAs provide an interesting enhancement to traditional analog circuit and signal processing education. They can provide the students with increased laboratory experiences earlier in their educational careers and can be used within a meaningful mixed–signal prototyping system for upper divisional and graduate–level courses.

A number of different FPAA ideas and concepts have been mentioned with the hope that discussions within the engineering academic community can contribute to a refined FPAA design that is optimized for use in educational laboratories. This paper has investigated a number of different ideas for using FPAAs within a full range courses from basic circuits classes to upper–divisional and graduate–level courses in embbeded systems, neuromorphic VLSI, and cooperative analog/digital signal processing.

References

[1] T. S. Hall, C. M. Twigg, P. Hasler, and D. V. Anderson, “Developing large–scale field–programmable analog arrays,” in *Proc. 18th International Parallel and Distributed Processing Symposium*, Santa Fe, New Mexico, April 2004.

[2] T. S. Hall, C. M. Twigg, P. Hasler, and D. V. Anderson, “Application performance of elements in a floating–gate FPAA,” in *Proceedings of the 2004 IEEE International Symposium on Circuits and Systems*, May 2004, pp. II.589–II.592.

[3] T. S. Hall, C. M. Twigg, P. Hasler, and D. V. Anderson, “Developing large–scale field–programmable analog arrays for rapid prototyping,” *International Journal of Embedded Systems*, 2004.

[4] T. S. Hall, C. M. Twigg, J. D. Gray, P. Hasler, and D. V. Anderson, “Large–scale field–programmable analog arrays for analog signal processing,” *IEEE Transactions on Circuits and Systems*, 2004.

[5] J.F. Wakerly, *Digital Design: Principles and Practices*, chapter 5: Combinational logic design practices, pp. 311–455, Prentice Hall, New Jersey, third edition, 1999.

[6] J. M. Birkner and H. T. Chua, “Programmable array logic circuit,” U.S. Patent No. 4,124,899.

[7] P. Chow, S. O. Seo, J. Rose, K. Chung, G. P. Monzon, and I. Rahardja, “The design of an SRAM–based field–programmable gate array—part i: architecture,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 7, no. 2, pp. 191–197, June 1999.

[8] M. A. Sivilotti, *Wiring Considerations in Analog VLSI Systems, with Application to Field-Programmable Networks (VLSI)*, Ph.D. thesis, California Institute of Technology, Pasadena, CA, 1991.

[9] P. G. Gulak, "Field–programmable analog arrays: past, present and future perspectives," in *IEEE Region 10 International Conference on Microelectronics and VLSI*, Nov. 1995, pp. 123–126.

[10] D. Marsh, "Programmable analogue ICs challenge spice–and–breadboard designs," in *EDN Europe*, pp. 30–36. Reed Business Information, http://www.ednmag.com, Oct. 2001.

[11] B. Ray, P. P. Chaudhuri, and P. K. Nandi, "Design of OTA based field programmable analog array," in *Proc. 13th International Conference on VLSI Design*, Jan. 2000, pp. 494–498.

[12] B. Pankiewicz, M. Wojcikowski, S. Szczepanski, and Y. Sun, "A field programmable analog array for cmos continuous–time ota-c filter applications," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 2, pp. 125–136, Feb. 2002.

[13] E. Sanchez-Sinencio, J. Ramirez-Angulo, B. Linares-Barranco, and A. Rodriguez-Vazquez, "OTA-based non-linear function approximations," in *ISCAS Proc.*, May 1989, vol. 1, pp. 96–99.

[14] Fast Analog Solutions Ltd., http://www.zetex.com, *Totally re-configurable analog circuit – TRAC®*, Mar. 1999.

[15] P. Hasler, B. A. Minch, and C. Diorio, "An autozeroing floating-gate amplifier," *IEEE Transactions on Circuits and Systems II*, vol. 48, no. 1, pp. 74–82, Jan. 2001.

[16] M. Kucic, A. Low, P. Hasler, and J. Neff, "A programmable continuous-time floating-gate Fourier processor," *IEEE Transactions on Circuits and Systems II*, vol. 48, no. 1, pp. 90–99, Jan. 2001.

[17] J. W. Bruce, J. C. Harden, and R. B. Reese, "Cooperative and progressive design experience for embedded systems," *Proceedings of the IEEE*, vol. 88, no. 1, pp. 23–30.

[18] W. Wolf and J. Madsen, "Embedded systems education for the future," *IEEE Transactions on Education*, vol. 47, no. 1, pp. 83–92, Feb.

[19] M. Leeman, F. Barat, V. De Florio, and G. Deconinck, "Bridging the educational gap in embedded systems curricula: developing an e–commerce audio streaming system," in *Proc. Ninth Annual IEEE International Conference and Workshop on the Engineering of Computer–Based Systems*, Lund, Sweden, April 2002, pp. 211–220.

[20] C. Mead, *Analog VLSI and Neural Systems*, Addison-Wesley, Reading, MA, 1989.

[21] E. Farquhar and P. Hasler, "A bio–physically inspired silicon neuron," in *Proceedings of the 2004 IEEE International Symposium on Circuits and Systems*, May 2004, pp. I.309–I.312.

[22] E. Farquhar, D. Abramson, and P. Hasler, "A reconfigurable bidirectional active 2–dimensional dendrite model," in *Proceedings of the 2004 IEEE International Symposium on Circuits and Systems*, May 2004, pp. I.313–I.316.

[23] C. Gordon, E. Farquhar, and P. Hasler, "A family of floating–gate adapting synapses based upon transistor channel models," in *Proceedings of the 2004 IEEE International Symposium on Circuits and Systems*, May 2004, pp. I.317–I.320.

[24] D. V. Anderson, P. Hasler, R. Ellis, H. Yoo, D. W. Graham, and M. Hans, "A low–power system for audio noise suppression: A cooperative analog–digital signal processing approach," in *Proc. 2002 DSP Workshop*, Pine Mountain, GA, Oct. 2002.

[25] R. Ellis, H. Yoo, D. Graham, P. Hasler, and D. Anderson, "A continuous–time speech enhancement front–end for microphone inputs," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, Phoenix, AZ, May 2002, vol. 2, pp. II.728–II.731.

[26] P. D. Smith, M. Kucic, R. Ellis, P. Hasler, and D. V. Anderson, "Mel–frequency cepstrum encoding in analog floating–gate circuitry," in *Proceedings of the International Symposium on Circuits and Systems*, 2002, vol. 4, pp. 671–674.

Tyson S. Hall

Tyson Hall received the Ph.D., MSECE, and BSCMPE degrees in electrical and computer engineering from the Georgia Institute of Technology in 2004, 2001, and 1999. He is currently an Assistant Professor in the School of Computing at Southern Adventist University in Collegedale, Tennessee. Dr. Hall's research interests include rapid prototyping of mixed–signal systems, cooperative analog/digital signal processing, reconfigurable computing, and embedded systems education.

Paul Hasler

Paul Hasler received the M.S. and B.S.E. degrees in electrical engineering from Arizona State University, Tempe, in 1991, and the Ph.D. degree from the California Institute of Technology, Pasadena, in computation and neural systems in 1997. He is an Associate Professor in the School of Electrical and Computer Engineering at Georgia Institute of Technology. His current research interests include low power electronics, mixed-signal system ICs, floating-gate MOS transistors, adaptive information processing systems, smart interfaces for sensors, cooperative analog/digital signal processing, device physics related to submicron devices or floating-gate devices, and analog VLSI models of on-chip learning and sensory processing in neurobiology. Dr. Hasler received the NSF CAREER Award in 2001 and the ONR YIP Award in 2002.