INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2014, VOL. 60, NO. 1, PP. 15–26
Manuscript received January 15, 2014; revised March, 2014. DOI: 10.2478/eletel-2014-0002

Analog Reconfigurable Circuits

Andrzej Malcher and Piotr Falkowski

***Abstract*—The aim of this paper is to present an overview of a new branch of analog electronics represented by analog reconfigurable circuits. The reconfiguration of analog circuits has been known and used since the beginnings of electronics, but the universal reconfigurable circuits called Field Programmable Analog Arrays (FPAA) have been developed over the last two decades. This paper presents the classification of analog circuit reconfiguration, examples of FPAA solutions obtained as academic projects and commercially available ones, as well as some application examples of the dynamic reconfiguration of FPAA.**

***Keywords*—analog signal processor, Configurable Analog Block (CAB), dynamic reconfiguration, Field Programmable Analog Array (FPAA), Switched Capacitor (SC) circuit**

I. Introduction

In order to define the analog circuit reconfiguration we have to define an analog circuit: the analog circuit is a circuit which converts an analog input signal or a set of analog signals into an analog output signal or into a set of analog output signals.

As an analog signal we understand a signal which may have continuous values within a specified range. On the opposite side we have quantized signals which may have only discrete values. The quantized signals are a subject of digital signal processing, which is not discussed in this paper.

In the time domain analog signals can be divided into two classes: continuous-time signals and discrete signals. For the discrete signals the value is important only in discrete “points in time”. Figure 1 presents a comparison of the continuous time and discrete time analog signals.

The division of analog signals into discrete time and continuous time signals is important because it determines the main classification of reconfigurable circuits. They can be divided into continuous time, discrete time and mixed analog reconfigurable circuits. The continuous time class of reconfigurable circuits contains devices based on Operational Transconductance Amplifiers (OTA) [1]–[3], current conveyors [4], [5] and also a wide class of so called Externally Linear (ELIN) circuits – e.g. log domain [6], [7], square root domain and others. The discrete time circuits are mainly based on the Switched Capacitance (SC) principle [8]–[11] or Switched Current (SI) principle [12]. The mixed reconfigurable circuits usually contain a continuous time tunable anti-aliasing filter at the input, a fully reconfigurable SC network as a processing unit and a continuous time smoothing filter at the output. However, sometimes they contain also both – continuous time and disrete time processing blocks.

The paper is structured as follows. Section II contains a general classification of the analog circuit reconfiguration proposed by the authors. In section III some practical realizations of FPAA are presented – commercially available as well as created as academic projects. Section IV presents in detail the most popular and commonly used family of FPAA – the dpASP circuits from Anadigm. In section V some practical applications of FPAA circuits are described. The last section contains final conclusions.

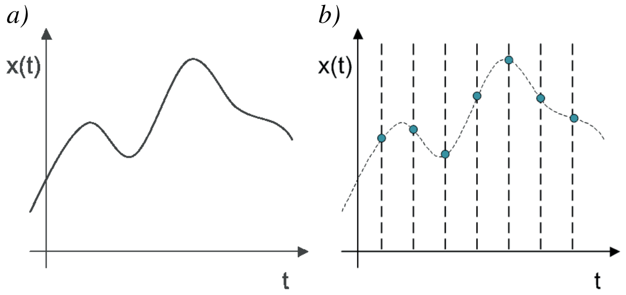


Fig. 1. Continuous time (a) and discrete time (b) analog signals.

II. Classification of the Analog Circuit Reconfiguration

Depending on their application, basic analog circuits can be described using different types of equations. There are several classes of analog circuits:

1) Static non-linear circuits. In these circuits the input and output signals are interrelated by a nonlinear dependency. We can point out multiplying, dividing, log, root, rectifiers and other circuits. The common feature of these circuits is the fact that the characteristics do not contain the time variable.
2) Dynamic linear continuous time circuits. In these circuits the input and output signals are described by a linear differential equation. The transfer function is commonly presented using the Laplace transformation. The examples of these circuits are analog filters, amplifiers and also sine wave oscillators.
3) Dynamic linear discrete time circuits. These circuits are usually described using the difference equations or the Z transformation. The examples are commonly used switched capacitance (SC) filters.
4) Other analog circuits. There are some circuits which cannot be included into one of the above classes. An example is the sample and hold circuit.

In the real world there are no pure static non-linear or dynamic linear circuits, but this slightly “artificial” classification is very useful in the design and analysis process.

Practical electronic systems contain usually a combination of several basic circuits presented above.

Independently from the class to which the circuit belongs, it can be described by a kind of an analytical equation:

$$\underline{Y(t)} = f(\underline{X(t)}, \underline{p}) \qquad (1)$$

A. Malcher is with the Institute of Electronics, Silesian University of Technology, Gliwice, Poland, (e-mail: amalcher@polsl.pl).

P. Falkowski was with the Institute of Electronics, Gliwice, Poland, now he is with TRW Engineering Center, Czestochowa, Poland, (e-mail: piotr.falkowski@trw.com).

where:
$\underline{X}$ – vector of the input signals;
$\underline{Y}$ – vector of the output signals;
$\underline{p} = [p_1, \ldots, p_n]$ – parameters or coefficients of the f function.

In classical analog circuits the forms of the f function and the $\underline{p}$ vector are constant. Reconfigurable analog circuits are circuits in which the form of the f function or the $\underline{p}$ vector can change. This leads to two classes of reconfiguration:

1) Parametric reconfiguration takes place when one or more parameters p_i of the f function can be tuned during the operation, but the main function remains unchanged. An example of this type of reconfiguration is tuning of a filter.
2) Functional reconfiguration takes place when the function of the circuit changes. As an example we can point out switching over the functions of a multimeter (e.g. from voltmeter to ohmmeter).

Parametric reconfiguration can be further divided into continuous and discrete.

- In the continuous parametric reconfiguration the parameters can take any value from the defined range. A simple example of it is gain regulation in an amplifier using a potentiometer.
- In the discrete parametric reconfiguration the parameters may have only predefined discrete values. It happens for example in a multimeter during switching the ranges.

Functional reconfiguration can also be divided into two groups:

- Without resources reuse – it happens when each function performed by the circuit uses its own dedicated hardware designed and tuned only for that specific application. An example is switching the radio receiver between the AM and FM. Each band uses usually its own dedicated hardware.
- With resources reuse – it happens when different functions can be performed using the same hardware resources.

The functional reconfiguration with resources reuse is the most desired form of reconfiguration. However, it is also difficult because it requires multi-functional cells and a configurable network of interconnections. The solutions to this problem are presented in subsequent paragraphs.

The analog circuit reconfiguration can be static or dynamic. In the dynamic reconfiguration the changes of parameters or even the whole function takes place “on the fly” without breaking the signal continuity. In the static reconfiguration each parameter change requires temporary stopping of the circuit, reloading and activating the new configuration.

In the analog reconfigurable circuits one has to distinguish a subsystem which can be called “reconfigurator”. It is a part of the system which controls the process of reconfiguration. Depending on the principle of operation of the reconfigurator, we can analyze two modes:

a) reconfiguration “on demand”, when the change of the parameters or function is initiated by the user, for example by means of electromechanical components like switches or potentiometers;
b) automatic reconfiguration, where the new configuration is activated according to an algorithm implemented for example in a microprocessor system.

Automatic reconfiguration can be further divided into:

- reconfiguration according to a scheduler;
- open-loop automatic reconfiguration;
- closed-loop automatic reconfiguration.

In the open-loop reconfiguration the reconfigurator observes the features of the input signals or other signals and, as a result of detection of the defined criterion, it switches the parameters or the function of the analog circuit. As an example we can point out the de-esser circuit discussed further.

In the closed-loop reconfiguration the reconfigurator observes the output signals of the circuit and controls the parameters of the circuit in order to obtain required features of the output signal. The example of the closed-loop reconfiguration is a commonly used Automatic Gain Control (AGC) circuit.

The comparison of the open-loop and closed-loop reconfiguration is presented in Fig. 2.

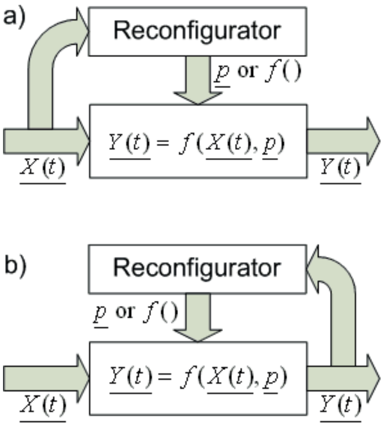


Fig. 2. Principle of the automatic reconfiguration – a) open loop reconfiguration, b) closed loop reconfiguration.

Figure 3 presents a complete classification of the reconfigurable circuits.

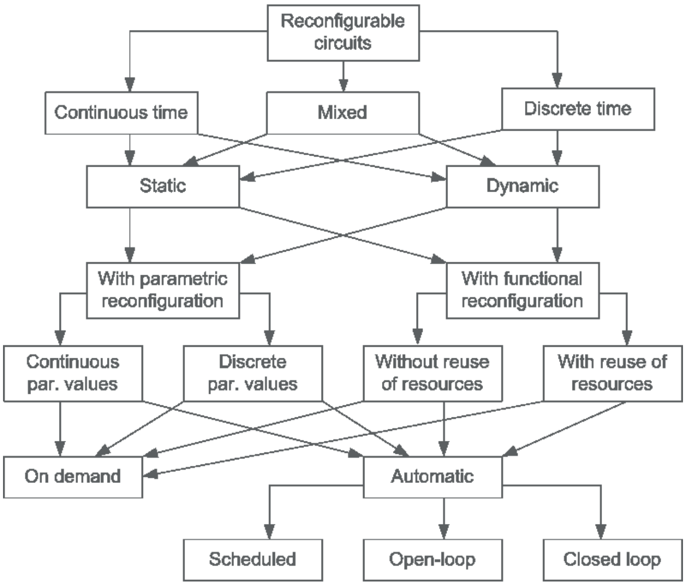


Fig. 3. Classification of the analog circuit reconfiguration.

This short summary of the reconfiguration classes helps us realize that most commonly used analog devices implement a kind of reconfiguration. However, most of these devices are dedicated to specific applications and their "area of reconfiguration" is fixed by the manufacturer.

Universal circuits may have a much wider range of applications as their functions can be configured by the user. These kinds of circuits can be included in a greater project and their function can be developed, tested, modified and tuned in the same way as the configuration of the commonly used FPGA circuits or the program of microcontrollers. The designer is obviously limited by the amount of resources.

The next section contains a short overview of available analog reconfigurable circuits.

III. Practical Inplementations of Analog Reconfigurable Circuits

Contrary to digital programmable circuits (e.g. FPGA or CPLD), which are commonly available and used, the analog programmable circuits are considered a new branch. There are few vendors that deliver commercially available FPAA circuits, while the number of academic projects on programmable analog circuits is still growing. Some of them will be presented below.

A. Academic Projects

Most devices encountered in literature are academic projects. According to the authors' knowledge, none of the devices described in this section is available on the market.

1) Reconfigurable Analog Signal Processor (RASP)

The RASP is an FPAA designed by Tyson S. Hall in his Ph.D. thesis [13]. Since then, it has been developed by the CADSP group at Georgia Institute of Technology [14].

It is a large-scale device consisting of 32 Configurable Analog Blocks (CABs) of two types – CAB 1 and CAB 2. CAB 1 includes 3 Operational Transconductance Amplifiers (OTAs) with programmable bias current sources, 3 floating capacitors, 2 multi-input floating-gate transistors (FGFET), a transmission gate and an nMOS/pMOS transistor array. CAB 2 is built of an FGFET-based current mirror, an OTA and 2 folded Gilbert multipliers.

The authors of [14] report that RASP is able to perform AM reception and speech processing. One of the previous generations of this FPAA was reported to be capable of modeling a perceptual model of an MP3 encoder [15].

2) FPAA by Pankiewicz et al.

Bogdan Pankiewicz *et al.* report an FPAA designed for filtering applications [1]. It was developed as a result of the main author's Ph. D. thesis.

The device comprises 40 CABs built of one OTA and one programmable capacitor each. The CABs are placed in an array of 5 rows and 8 columns with interconnections between the neighbouring CABs.

The authors report a design of a sixth-order Chebyshev bandpass filter with the center frequency of 60 kHz, tunable in a range of about 22 times. The reported maximum operating frequency of a single CAB in this FPAA is 9 MHz.

3) FPAA by Univ. of Freiburg and Univ. of Ulm researchers

A team of researchers from the University of Ulm and University of Freiburg presented an FPAA of different architecture [16]. It is also built as a continuous-time device (on OTAs and capacitances). Its authors introduce a hexagonal lattice CAB layout, which employs 6 interconnections to the neighboring blocks and one for self feedback. Tuning is realized using FGFET switches and current sources.

The latest report on this work [3] states that the designed circuit contains 55 CABs and its Gain-Bandwidth Product (GBP) is 186 MHz.

4) Translinear log-domain CMOS FPAA by Fernadez et al.

The FPAA circuit presented in [7] consists of 25 reconfigurable translinear cells based on the log-domain principle. They use MOS transistors as translinear elements. Thanks to a special "predistortion effect" the characteristics are exponential not only in the weak-inversion range but also in the moderate-inversion range. It gives about five decades of useful operating range. The bandwidth of the FPAA is about 20 MHz. Several practical applications like a four quadrant multiplier or a fourth order log-domain filter were successfully mapped and tested in the circuit.

5) Current conveyor based voltage mode FPAA by Soliman et al.

The FPAA circuit presented in [4] belongs to the continuous time class of reconfigurable circuits. It consists of 30 digitally programmable second generation current conveyors combined in 7 macrocells. The gain of each cell can be digitally switched over by a three-bit digital control word. The interconnections among the cells are fixed, but each output current of the cell can be switched off by the control logic, which allows to obtain different topologies. The authors presented an application of their circuit in a second order tunable universal (low/high/band-pass) filter. The cut-off frequency of the filter can be digitally tuned from 8.8 MHz to 11.6 MHz.

6) Current conveyor based FPAA by Gaudet et al.

The FPAA circuit [5] consists of 4 CAB blocks. Each of them contains a current conveyor, two transconductors working as tunable grounded resistors, two programmable capacitors and a buffer. The set of applicable functions contains amplification, integration, differentiation, addition, subtraction, comparison. The bandwidth of this circuit for various configurations was measured between 11 MHz and 13 MHz, so it can be applicable for analog video signal processing.

7) Op-amp based FPAA by Loobi and Lyden

The continuous time FPAA circuit presented in [17] contains the cells based on CMOS operational amplifiers. There are two types of cells. The first type is a programmable gain adder based on an *R-2R* DAC with 5 bit resolution. The second one is a subtractor cell with a switchable feedback capacitor. The cell can work as an differential integrator or a comparator. Approximately 60% of the area is occupied by the interconnection.

The authors report two applications of the FPAA: a PC power supply monitoring subcircuit and a programmable pulse generator. The bandwidth obtained in the prototype circuit was about 1 MHz.

8) Switched current (SI) FPAA by Chang et al.

The FPAA circuit presented in [12] works in discrete time. The multi-function switched current blocks are using four phase clock signals. The authors report an application of this circuit in a current controlled oscillator and a second order biquad filter.

B. Commercially Available Devices

There are several families of programmable analog circuits commercially available. They differ from one another in principle of operation, amount of resources and area of application.

1) PSoC family from Cypress Semiconductor [8]

PSoC® is a **programmable embedded system-on-chip** integrating configurable analog and digital peripheral functions, memory and a microcontroller on a single integrated circuit. The PSoC® architecture consists of configurable analog and digital blocks, a CPU subsystem and programmable routing and interconnect. PSoC allows to insert a predefined and tested analog and digital circuitry IP from the PSoC library into a project.

The analog system of PSoC 1 family is composed of 12 configurable blocks, each containing an opamp based circuit, which allows to create complex analog signal flows. Analog blocks are provided in columns of three, each including one Continuous Time (CT) and two Switched Capacitor (SC) blocks. Analog peripherals are very flexible and can be customized to support specific application requirements.

Some more common PSoC analog functions (mostly available as user modules) are: ADCs (selectable as incremental, delta sigma, and SAR), DACs (with 6- to 9-bit resolution), filters (2-, 4-, 6-, and 8-pole band pass, low pass, and notch), amplifiers (up to 4, with selectable gain to 48x), instrumentation amplifiers (up to 2, with selectable gain to 93x), comparators (up to 4, with 16 selectable thresholds), multiplying DACs (up to 4, with 6- to 9-bit resolution), high current output drivers (four with 30 mA drive as a core resource), 1.3 V reference, DTMF dialer, modulators, correlators, peak detectors and many other basic blocks.

Some of the PSoC circuits are equipped with a CapSense™ interface. Capacitive sensors like touch devices can be connected directly to these terminals.

Each of the analog blocks has to be parameterized and switched on by the CPU software. The program can be written in C or assembler languages. It is also possible to change parameters of particular blocks during runtime.

2) FPAA and dpASP family from Anadigm [9]–[11]

Anadigm offers a full range of 5 V and 3.3 V programmable analog arrays. They are divided into static programmable devices (FPAA), which require a reset before loading a new configuration bitstream, and so called dynamically programmable Analog Signal Processors (dpASP). Anadigm's dpASPs provide real time dynamic reconfigurability that allows the functionality of the dpASPs to be reconfigured in-system by the designer or "on-the-fly" by a microcontroller/processor. A dpASP can be programmed to implement multiple analog functions and/or adapt "on-the-fly" to maintain precise operation required by applications that have changing requirements in real time, such as signal conditioning, filtering, data acquisition, and closed-loop control.

By using AnadigmDesigner2 EDA software, the designer can construct complex analog functions using configurable analog modules (CAMs) as building blocks. With an easy-to-use drag-and-drop interface, the design process can be measured in minutes allowing complete analog systems to be built, immediately simulated, and then downloaded to the dpASP chip for testing and validation.

Programmable analog arrays from the Anadigm company are the most popular commercially available FPAA circuits.

3) ispPAC family from Lattice Semiconductor [18], [19]

The ispPAC family consists of 5 circuits. All of them work according to continuous time principle and can be programmed in-system. The circuits have different functionality. ispPAC10 is designed for linear signal processing – amplification and filtering. The corner frequency of the filters can be configured from 10 Hz to 250 kHz. ispPAC20 is intended for non linear signal processing. It contains comparators, multiplexers and configurable filters. 8 bit digital to analog converter can work as a precision reference voltage source. The interconnections between the blocks are completely programmable. ispPAC30 is a multi function signal conditioning system. It contains wide range programmable gain amplifiers with low offset voltage, and also programmable filters. It implements analog signal arithmetic – summing, subtraction. The circuit is reconfigurable on-the-fly. ispPAC80 is a programmable 5-th order continuous time active filter. The filter type (Butterworth, Chebychev, Legendre, Elliptica, Linear Phase, Bessel) and also the gain and corner frequency can be configured. All the ispPAC circuits can be programmed using Lattice's PAC-Designer design tools. Circuit designs are entered graphically and verified all within a single environment. The PAC-Designer schematic window provides access to all programmable features in ispPAC devices via a graphical user interface. Once the design file is complete, it can be saved and an industry-standard JEDEC file can be generated for device programming. The ispPAC circuits are intended for wide variety of analog signal conditioning.

In 2007 the production of ispPAC circuits was discontinued and there are no replacement parts for them.

4) Other devices

There are several more FPAA families the production of which is discontinued. An example is TRAC family from Zetex Semiconductors [20]. The circuits contained 20 simple configurable analog blocks working in continuous time. Another example is MPAA020 from Motorola [21]. It also contained 20 cells but working on the SC principle. The circuit was statically reconfigurable, however, there were some works on dynamic reconfiguration of systems based on this circuit [22].

The comparison of the main features of the commercially available FPAA circuits is presented in Tab. I.

TABLE I
COMMERCIALLY AVAILABLE FPAA FAMILIES

Manufacturer	Series	Mode	Bandwidth	Number of cells	Technology
Motorola	MPAA 020	SC	200 kHz 1 MHZ clk	20	CMOS
Zetec	TRAC	Voltage cont.	1 MHz	20	BiCMOS
Lattice	IspPAC	Voltage cont.	15 MHz	4	CMOS
Cypress	PSoC	Cont. /SC	8.9 MHz	12	CMOS
Anadigm	FPAA dpASP	SC	470 kHz 20 MHz clk	4	CMOS

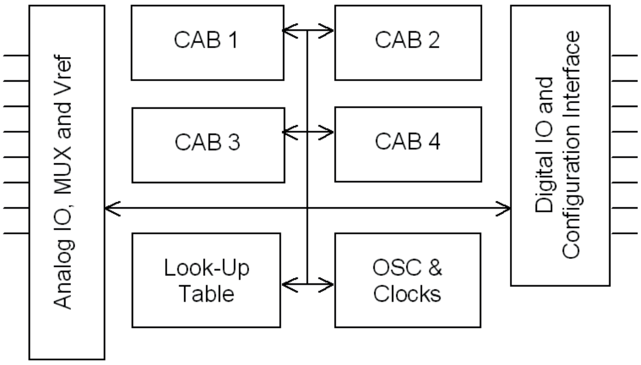


Fig. 4. Simplified architecture of FPAA circuits.

IV. OVERVIEW OF THE ANADIGM FAMILY OF THE PROGRAMMABLE ANALOG CIRCUITS

Most applications of the programmable analog arrays reported in literature are based on the dpASP circuits from the Anadigm company, therefore this section will present this family in detail. The products of this company are a continuation of the MPAA020 from Motorola. The first FPAA from Anadigm was marked AN10E40 and contained, like its predecessor, 20 configurable analog blocks. It belongs to the first generation of FPAA circuits, but is already not available.

At present, Anadigm offers two families of FPAA. One of them, marked as second generation programmable circuits, is called AnadigmVortex. It is powered from a 5V supply. The Vortex family contains 8 circuits and the most popular of them is AN221E04. The third generation AnadigmApex family is a successor of AnadigmVortex with the commonly used AN231E04 circuit. One of the most noticeable differences between the families is the supply voltage – the new Apex family is powered from a 3.3 V supply instead of the 5 V one. This results in easy interfacing with contemporary 3.3 V microcontrollers with no need of using level-shifters. The power consumption has not changed significantly. Both AN231E04 and AN221E04 with comparable configurations and 16 MHz clock require about 45 mA of current [10], [11].

Each member of these two families is offered in two versions: Dynamically Programmable Analog Signal Processor (dpASP) with the dynamic reconfigurability feature and Field Programmable Analog Array (FPAA) which is only statically configurable. For example AN231E04 is dynamically reconfigurable, and AN131E04 – statically. The analog part of these circuits is identical.

The configuration RAM of the dpASP chip is divided into the configuration memory and shadow memory. When the FPAA is working, it is set up in accordance with the configuration memory. New configuration data received by SPI interface is stored in the shadow memory. Configuration changes do not take effect until the FPAA receives an “Execute” signal, which can be generated externally or internally (by an event or on configuration data transfer completion).

Both families include 2x2 Configurable Analog Blocks (CAB) array. Each CAB consists of 2 operational amplifiers, 8 programmable capacitors, an 8 bit Successive-Approximation register and a comparator. All analog signal paths are fully differential. Chip architecture includes also a Lookup Table (LUT) for realizing nonlinear transfer functions.

A simplified block diagram of the circuits is presented in Fig. 4 and a diagram of the Configurable Analog Block (CAB) – in Fig. 5.

The structure of the CAB makes it possible to implement different functional units called Configurable Analog Modules (CAM). The user of the AnadigmDesigner software can choose the CAMs from the library, insert them in the workspace using easy to use the drag-and-drop tool and connect together. Thanks to the smart structure of the blocks one CAB can implement several CAMs at the same time – it can e.g. contain a low pass filter, an amplifier and a comparator.

The combination of two operational amplifiers and the bank of switches makes it possible to build different types of amplifiers, sum/dif circuits and also bilinear and biquadratic filters. The comparator can be used in rectifier circuits and in peak detectors. The combination of the 8 bit SAR and the Look Up Table allows to implement nonlinear functions. In multiplying and dividing circuits one signal is converted to a digital form in the SAR, then the conversion result is used as an address in the LUT of 256 elements. The data read from the LUT controls the gain of the amplifier fed by the second signal. The SAR with LUT enables also the implementation of the non-linear transfer function, variable gain amplifiers and arbitrary generators.

The list of the CAMs which can be implemented in the FPAA circuit contains more than 30 elements and can be extended. Customers can request some specific functions and the engineering staff from the Anadigm can prepare new library elements for those applications. Such elements can be imported to the AnadigmDesigner software and used in the projects. Usually such elements are marked in the library as “Unapproved” which means that they have not been fully tested.

A. *Configuration Procedure of the dpASP*

The configuration data of the Anadigm have to be loaded to the circuit via the SPI interface. The circuit does not contain any non-volatile memory, therefore the configuration has to be

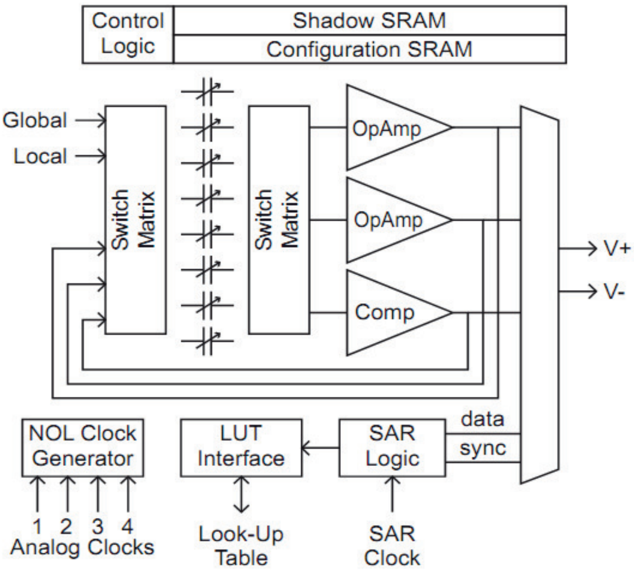


Fig. 5. Block diagram of a Configurable Analog Block [9].

stored externally. There are two modes of data downloading selectable by an external MODE pin. In Master mode the FPAA circuit during startup loads the configuration data stream from an external serial EPROM module through the SPI interface. In this mode there is no possibility of dynamic reconfiguration. In the Slave mode the data transmission is controlled by an external microcontroller. Depending on the header of the datastream the configuration can be treated as "Primary" and loaded directly to the configuration memory, or as "Reconfig data" and loaded to the shadow memory. The reconfig data can contain new parameters of particular blocks or even the whole configuration of the module. It is activated in one clock cycle after the Execute signal, which can be generated externally, internally or automatically after the write to the shadow RAM is completed. The digital interface of the dpASP circuits allows to connect several devices into a system. During the primary configuration download each device obtains a unique address. All modules can be connected to a single SPI bus and occupy only 3 lines of the host microcontroller (CS, SCLK and MOSI).

The primary configuration data is created by the AnadigmDesigner2 software as a result of mapping the user design into the physical structure of the device. The data can be loaded directly to the FPAA system through the COM interface, but the board must be equipped with a PIC microcontroller with the Anadigm Boot Kernel (ABK) software. The evaluation boards delivered by Anadigm already contain this system.

The configuration data can be also saved in a file in one of different formats – e.g. Intel HEX, binary, standard C file and others. The data can be stored in a boot EEPROM or downloaded to the FPAA from any kind of microcontroller through the SPI bus.

There are two methods of creating reconfiguration data. The first of them is called "State driven method". In this method the user creates a schematic of the application and specifies lists of possible parameter values of selected CAMs (gains, corner frequencies and others). For each combination of these parameter values AnadigDesigner2 prepares a separate dataset in a selected format (binary, HEX, C source). In order to perform the dynamic reconfiguration the user program has to select adequate data set and send it to FPAA. The reconfig data sets are usually short (dozens of bytes). This method does not require large computing power of the microcontroller, but only sufficient storage space for all possible combinations of the parameters of CAMs. In this mode the parameter values can be switched only to predefined discrete values. The state driven method has a variant in which the whole structure can be swapped to another one in a single clock cycle without the circuit reset – dynamic functional reconfiguration.

The second method of reconfiguration is called "algorithmic method". In this method AnadigmDesigner2 creates four source files in ANSI C. These files contain functions that allow to change parameters of any CAM block implemented in the FPAA. An example can be the function that changes the parameters of a second order bandpass filter. To change any parameter of this filter (or all of them together) one should call the SetBQBandPassFilterII function. Its declaration part looks as follows:

void an_SetBQBandPassFilterII(an_CAM cam, double Fo, double G, double Q);

The function has the following parameters: an_CAM – id of the CAM module, Fo – center frequency in kilohertz, G – maximum gain and Q – the quality factor. The function body is written in ANSII C language, so any C compiler that implements the floating point arithmetic should compile it correctly and include it into a larger project. As a result, this function generates a string of reconfig data which should be sent through the SPI link to the FPAA. AnadigmDesigner2® generates similar functions for all blocks used in the project and also for the configurable input and output cells. This method allows for dynamic parametric reconfiguration with continuous parameter values, but it requires much more computational power than the state driven method.

AnadigmDesigner2 contains also a Visual C++ application generator. It generates a complete template of the PC application which utilizes the dynamic reconfiguration according to the algorithmic method. The template contains functions for configuring the COM port, parallel port and also for data downloading. Functions which allow parameter reconfiguration of each used CAM are also included.

B. *Specific Features of the Vortex Family*

The second generation of the dpASP/FPAA circuits is powered from 5V. The main functional difference between Apex and Vortex families is the number and structure of IO cells. The Vortex family contains 4 input cells with an output option, 2 dedicated output cells and 1 auxiliary output cell. Each of the input cells can work with gain in the range from 16 to 128 or in the bypass mode. In addition, when a cell is programmed in the gain mode, the chopper amplifier can be enabled to nullify the input offset voltage. Each of the input cells has also a programmable lowpass anti-alias filter with corner frequency ranging from 76 kHz to 470 kHz. One of the input cells is connected to an input multiplexer, providing a possibility to select one of four input differential channels or eight single ended channels.

Output cells of Vortex-family FPAA can be configured as either analog or digital. Analog output cells have smoothing lowpass filters with the same frequency range as the input antialiasing filters. An output cell can also work in the bypass mode.

The input and output continuous time antialiasing filters were tested in detail in [23]. The measurement results show that they introduce a high level of harmonic distortion – especially when set to low corner frequency. The only way to avoid it is to switch the cells to the bypass mode and use external antialiasing filters, what generates additional costs.

Another interesting feature of the Vortex family is the fact that the CAB library contains blocks which allow to connect external capacitors. Thanks to that, one can build filters with very low corner frequencies.

C. Specific Features of the Apex Family

The Apex-family device has 7 IO cells in total and 1 auxiliary 2-output cell. There are 4 Type1 and 3 Type2 cells. Each of the Type1 cells can work as an input or output cell. When configured as an input, Type1 cell can work in the bypass mode as a fully-differential amplifier or as a sample-and-hold (S/H) circuit. In addition, two out of four Type1 cells can access a chopper amplifier.

Type1 cell used as an output can work in bypass mode, as an S/H circuit or as a digital output (for comparator signal output). It can also output a 1.5 V reference voltage (VMR).

Type2 cells can be set as a bypass input, digital input (for controlling internal comparators), bypass output, digital output or VMR output. In the digital output mode a Type2 cell can output a clock signal, a comparator output signal or a RAM-transfer pulse indicating a configuration change. In the input mode one of the Type2 cells can access a chopper amplifier.

The AN231E04 FPAA includes a single chopper amplifier with programmable gain feature accessible from one selected IO cell. The gain ranges from 0 to 60 dB. Besides that, OpAmp chopping can be enabled for each of the gain-stage CAMs in the FPAA.

The AN231 IO cells do not include anti-aliasing filters. It is up to the user to design an anti-aliasing and smoothing filter. The filters can be built using differential amplifiers of a Type1 cell configured as separate amplifiers and external resistors and capacitors. Anadigm provides the formulas to determine the values of components in the anti-aliasing filters with the required corner frequency.

D. Clock Generators

Another difference between Vortex and Apex families is the clocking system. The system is intended to create non overlapping signals for clocking SC cells of the FPAA. In the Vortex family the Master clock is divided by 1 to 510 and forms the System clock. The system clock is fed to 4 frequency dividers which supply 4 different clocking frequencies for different blocks of the application. In the Apex family there are two dividers of the master clocks and two different system clocks. There can be also six different clocking frequencies for the CABs. The clocks can be obtained by the division of the system clock frequencies.

V. Examples of Applications of the FPAA

As it was stated before, numerous applications reported in literature which utilize reconfiguration are based on the Anadigm FPAAs. There can be found three main areas of application:

1) Control and measurement systems,
2) Audio signal processing,
3) Biomedical signal processing.

This section presents some examples of such applications.

A. Programmable Analog Controller for Active Magnetic Levitation

An example of the dpASP application in control systems is the Programmable Analog Controller (PAC) presented in [24]–[27]. The controller is a modular system, which meets the industrial standards. It consists of the following modules: I/O, A/D, D/A, main Analog Processing Unit (APU), Power Driver (PD), Power Supply (PS), and Power Supply for Power Driver (PS-PD). The APU contains a chain of the dpASP circuits from Anadigm. They are connected to the Analog signal bus which may control the power modules. The dpASP modules are programmed and controlled by a microcontroller via a SPI digital bus. It allows the static and dynamic reconfiguration of the analog processors. The digital part can be also configured as an oscilloscope, data acquisition system or a digital controller. The communication between PAC and the host computer is carried via the USB interface.

The controller was tested in connection with a laboratory magnetic levitation system. It consists of a dual electromagnet and an optical displacement measurement system. Its task is to keep a ferromagnetic sphere levitated at a desired position.

The signal from the displacement sensor was connected to the analog input, while the control signal was supplied to the electromagnet coil through the power module.

The operation of the system was illustrated by an example of the PD controller. The authors tested the static and dynamic configuration of the dpASP. The static configuration takes place on the system start-up and when the structure of the signal path is changed. The authors show the histograms of the configuration time depending on the number of the dpASP used. The test results show that the data transmission from the host computer and the configuration download takes about 50 ms per processor.

As an example of the dynamic reconfiguration the authors show the step change of the target position. In this case the response time is much shorter then in start-up because the reconfiguration data stream occupies only 24 bytes. The switching between the old and new configuration takes only one system clock tick – in the presented application it takes 1.34 μs and the operation of the controller remains stable.

The tests with the dynamic reconfiguration showed that the system can work as an adaptive controller in which the parameters can be moderated during operation.

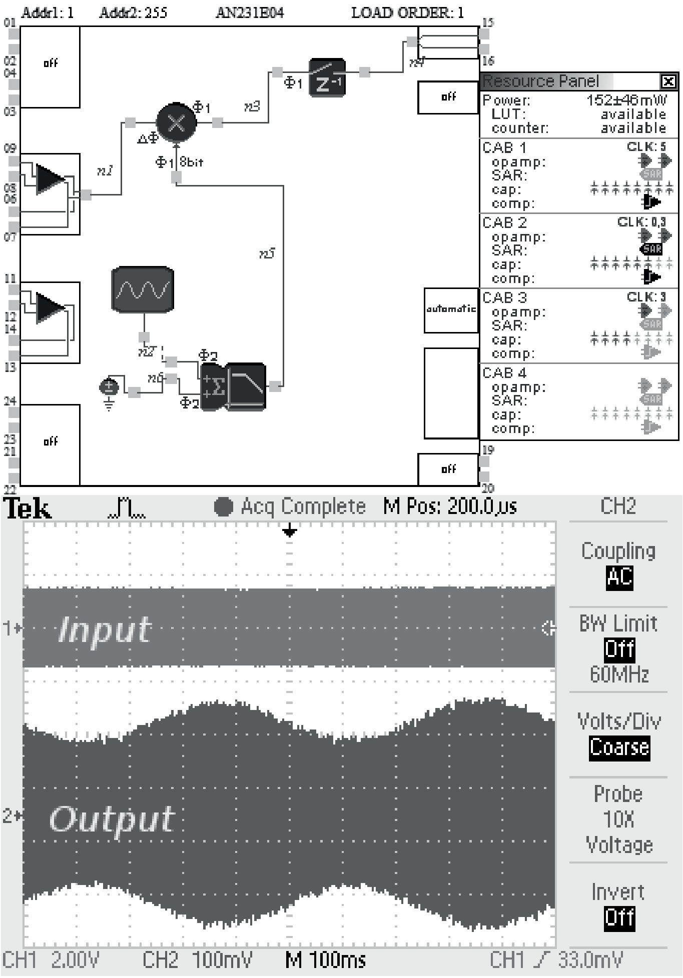
B. Ultrasonic Proximity Meter

Another example of application of the FPAA reconfiguration is a conditioning system for an ultrasonic transducer presented in [28]. The measurement system consists of the pulse train generator, analog front-end, the piezoelectric transducer and an FPAA circuit applied for the echo processing. The system is controlled by a PIC microcontroller.

The main task of the FPAA is to obtain the echo envelope. The signal path in the FPAA consists of: 1) a bandpass flter, whose 45 kHz central frequency is tuned to the ultrasonic sensor natural frequency with the aim of enhancing echo SNR; 2) a full wave rectifer followed by a peak detector in order to extract the echo envelope; 3) a low-pass flter with 1 kHz corner frequency used to smooth the output signal; and 4) an output amplifer employed to increase the level of the conditioned signal. The output signal of the FPAA is sampled by an A/D converter and stored in the microcontroller. The distance of the target is calculated according to position of the barycenter of the echo envelope. In order to obtain high measurement accuracy, full dynamic range of the converter should be used.

The experiments show that the shape of the echo envelope depends on the distance, so the optimum gain and quality factor of the filter should be also dependent on the distance. The authors decided to tune the filter parameters using dynamic reconfiguration of the FPAA. As a result of reconfiguration, the shape and amplitude of the echo envelope are independent on the target distance.

The authors compare meteorological features of the instrument working with and without the dynamic reconfiguration of the FPAA. They report that the INL error with the dynamic parameter tuning is less than 0.1% which corresponds to 1 mm distance measurement resolution in the range from 450 to 1150 mm.



Resource Panel
Power: 152±46mW
LUT: available
counter: available
CAB 1 CLK: 5
opamp:
SAR:
cap:
comp:
CAB 2 CLK: 0,3
opamp:
SAR:
cap:
comp:
CAB 3 CLK: 3
opamp:
SAR:
cap:
comp:
CAB 4
opamp:
SAR:
cap:
comp:

CH2
Coupling AC
BW Limit Off 60MHz
Volts/Div Coarse
Probe 10X Voltage
Invert Off

Fig. 6. Tremolo effect application – Schematic of the tremolo effect (upper); Input and output signals of the tremolo (lower).

C. FPAA in Audio Applications

The goal of the research work presented in [29], [30] was to determine if FPAAs can be utilized in audio processing applications [31]. To verify it, some audio effect circuits were designed and implemented using AN231E04 and AN221E04 FPAA. Due to the limitations of the development platform experiments were limited to standalone applications of the circuit. However, it is possible to build a microprocessor system combining several presented applications together. It is possible to switch over between them using the functional dynamic reconfiguration feature implemented in dpASP.

Every designed audio effect except AGC has been tested with an electronic keyboard, electric guitar and loudspeakers.

1) Tremolo effect

Tremolo is an effect of changing the amplitude of sound. It is implemented by using a simple amplitude modulator and a low-frequency oscillator (LFO). As the amplitude modulator, a multiplier CAM was used. LFO is realized using a sinewave oscillator CAM clocked with 100 Hz frequency. The schematic of the effect is presented in Fig. 6 (upper).

The lowest LFO output frequency obtained was 2 Hz. At lower frequencies the sinewave was distorted.

In Fig. 6 (lower) a time graph of 750 Hz tone modulated by the designed tremolo effect circuit is shown.

2) Infinite limiter (overdrive effect)

Limiting is a kind of dynamic range compression, limiting the gain of the circuit if the input signal is above a previously-set threshold. One of limiting variants is so-called “infinite limiting” or “clipping”.

A clipper circuit comprising a high-pass filter, a gain stage and a gain-limit stage has been realized. Its schematic is shown in Fig. 7.

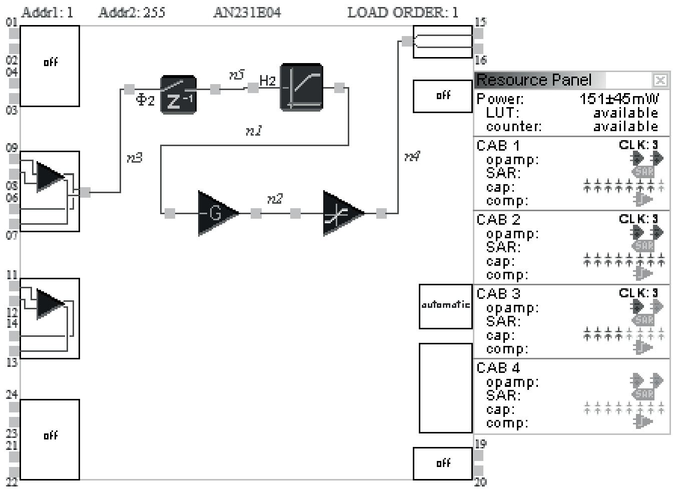
The HP filter is necessary to eliminate a DC component in the signal generated by an instrument. Without it, the DC component can overdrive the circuit.

There is no objective way of measuring the quality of the clipper effect. It can be rated only by hearing.

3) Automatic gain control (AGC) circuit

It is also possible to create other limiting or compressing circuits utilizing the AGC.

The automatic gain control circuit (AGC) is used to change the voltage gain in order to regulate the signal output level at the value possibly close to the preset one. There are also similar circuits that dynamically change the gain according to



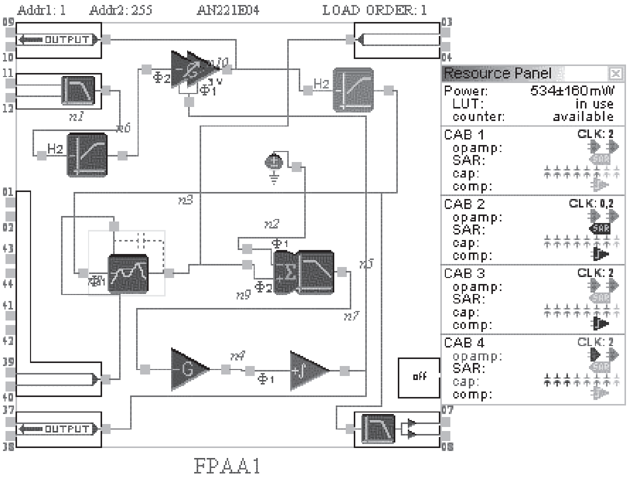
Resource Panel
Power: 151±45mW
LUT: available
counter: available
CAB 1 CLK: 3 opamp: SAR: cap: comp:
CAB 2 CLK: 3 opamp: SAR: cap: comp:
CAB 3 CLK: 3 opamp: SAR: cap: comp:
CAB 4 opamp: SAR: cap: comp:

Fig. 7. A schematic of the infinite limiter.

the input signal level – for example level compressors and level limiters commonly used in acoustic signal processing applications. The idea is always the same but the transfer characteristics differ.

An AGC circuit requires an analog multiplier or another amplifier with an electrically controlled voltage gain factor (Variable Gain Amplifier, VGA), for example a transconductance amplifier. The CAM library of the AN221E04 contains an SC amplifier block called GainVoltageControlled. In this circuit the capacitance values which determine the gain factor are set dynamically depending on the conversion result from the SAR analog-to-digital converter measuring the control voltage signal.

The AGC circuit realized using AN221E04 device works in the acoustic frequency range. The block diagram of the circuit created in AnadigmDesigner2 environment is shown in Fig. 8.



Resource Panel
Power: 534±160mW
LUT: in use
counter: available
CAB 1 CLK: 2 opamp: SAR: cap: comp:
CAB 2 CLK: 0,2 opamp: SAR: cap: comp:
CAB 3 CLK: 2 opamp: SAR: cap: comp:
CAB 4 CLK: 2 opamp: SAR: cap: comp:

Fig. 8. AGC circuit – block diagram.

The features and parameters of this circuit were studied in detail in [23].

The circuit gave a proper output level for the input signal amplitude over a two-decade range, from 40 mVpp up to 4 Vpp. These values correspond to the preset gain range of the voltage controlled amplifier (from 0.3 up to 50) stored in the LUT.

The output level and the response time constant could be tuned using the dynamic reconfigurability feature.

4) De-esser

De-essing is a process of removing excess sibilant sounds from a recording or a live audio mix. There are several ways to achieve such an effect. Split-band de-essing was chosen and examined.

The designed circuit consisted of two filters (lowpass and highpass), a peak detector, a variable-gain amplifier and a summing block. The filters split the input signal into two subbands around the frequency of 2 kHz.

Figure 9 presents spectra of the input and output signals of the de-esser. The spectra were measured when a previously recorded sample of an "s" consonant was played by a PC connected to the de-esser's input. It can be noticed in spectrum A that the "s" consonant has most of its energy concentrated above 4 kHz. In spectrum B the part above 4 kHz is attenuated about 10 dB compared to spectrum A. This is considered to be a success and a proof that a simple de-esser circuit can be implemented using the examined FPAA.

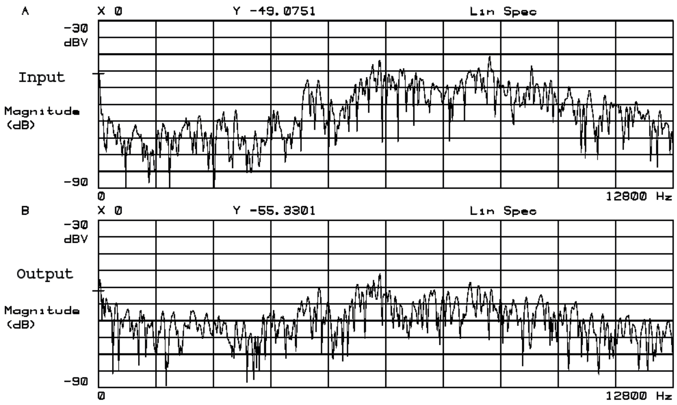


Fig. 9. Spectra of de-esser input (A) and output (B) signals.

D. FPAA Applications for Biomedical Signal Processing

There can be found numerous research works on biomedical signal processing in which the authors use the FPAA circuits. They are used in ECG signal filtering [32], fetal heart rate measurement [33], electromyografic (EMG) signal processing [34], phoncocardiographic acquisition system (smart stethoscope) [35] and others.

As an example of this group of applications we can briefly present a non-invasive fetal heart rate extraction system presented in [33]. Non-invasive fetal ECG (FECG) measurement is based on registration of the maternal ECG acquired from the skin electrodes over the mother's abdomen. The FECG is separated from the maternal signal using digital processing techniques. The fetal electrocardiogram signal maximum amplitude is about 600 μV and this signal is mixed with other biological electrical signals, such as maternal ECG, myographic signals, movement electrical artifacts and encephalograph signals from both the mother and fetus, apart from external interference sources such as power line interference.

In order to extract the FECG signal the authors decided to build a mixed analog-digital programmable acquisition system. The analog part contains a AN221E04 circuit working as a preconditioning module. The configuration of the FPAA consists of two similar paths – one for the abdominal electrodes and one for thoracic. Both paths use the input cell with a chopper stabilized amplifier and continuous time anti-aliasing filter. The next stage is a high pass filter, a band stop filter with the corner frequency of 50 Hz and the first order low pass filter. The filtered signal is fed to the output cell with a smoothing filter. Thanks to the dynamic reconfigurability of the FPAA the gains and the corner frequencies of the filters can be adapted to the features of the signals. The digital part is based on the Xilinx Spartan 3E FPGA device. The FPAA output signals are digitalized using two 14 bits A/D converters. The fetal ECG extraction is performed in the FPGA circuit using adaptive filtering. In the prototype solution presented in the paper the FPGA circuit is controlled by a LabView application. The solution was tested using signals form two benchmark databases.

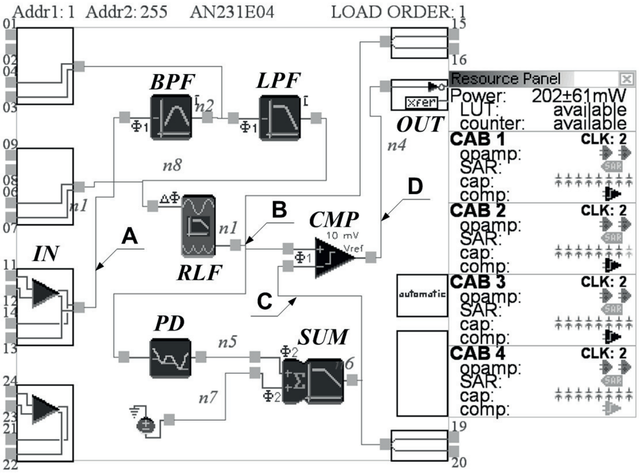
The authors report that the detection efficiency obtained in the reconfigurable analog-digital system is comparable to that obtained with other methods.

E. Hybrid System of the QRS Complexes Detection for Synchronization of Special Medical Devices

Some medical devices such as ventricular assist devices (VAD), pacemakers and defibrillators require synchronization with the appropriate phase of the cardiac cycle of the patient. In such applications the most important problem is to determine the position of the R wave in electrocardiogram. There are numerous offline algorithms which allow to locate the QRS complex with a very high accuracy. However, most of them introduce a large time delay of detection which is unacceptable in online operation like VAD synchronisation. There are numerous research works on the precise online QRS detectors, some of them use in the detection path an FPAA circuit [36]–[39]. Another problem is the fact that for the synchronization purpose the ECG signal is often obtained form the epicardial electrodes. As the result of the biochemical processes which take place on the electrodes the signal level is gradually decreasing and must be compensated by an AGC loop.

As an example of the solution of the online detection problem we can point the hybrid ECG acquisition and QRS detection system presented in [38]. The system uses modern programmable components – ADS129x multi-channel A/D converter with amplifiers and AN231E04 FPAA circuit. The construction fulfils the requirements for the medical equipment concerning the isolation and leakage currents, so it consists of two separate boards connected together by an optical isolator and isolated power supply circuits.

The first board contains the ADS circuit and an ARM microcontroller. The ADS129x circuit is specially designed for bipotentials measurements, so no conditioning circuits are required. The inputs are equipped only with ESD protection circuits. The gains of the ADS amplifiers are configurable, which is controlled by the first microcontroller. The 24 bit samples are transferred to the microcontroler, conditioned and standarized to 12 bit samples. These samples are transferred to the main microcontroller via an isolated serial interface. The QRS detection is performed in the second board. The samples are converted to the analog form by the D/A converter built in the main microcontroller and fed to the FPAA input. Figure 10 presents the block diagram of the FPAA application.


Addr1: 1 Addr2: 255 AN231E04 LOAD ORDER: 1

Resource Panel
Power: 202±61mW
LUT: available
counter: available
CAB 1 CLK: 2
opamp:
SAR:
cap:
comp:
CAB 2 CLK: 2
opamp:
SAR:
cap:
comp:
CAB 3 CLK: 2
opamp:
SAR:
cap:
comp:
CAB 4 CLK: 2
opamp:
SAR:
cap:
comp:

Fig. 10. The QRS detector implementation in the FPAA.

The principle of detection is based on a comparison of the filtered and rectified ECG signals with the threshold voltage obtained from the peak detector. The ECG signal (A) is provided to the input IN of the FPAA matrix. The signal is band-pass filtered (BPF block) in order to extract spectral components correspondent with QRS complexes. Additionally, a second order low-pass filter (LPF) was implemented to eliminate high frequency noise. The filtered signal is rectified by a full wave rectifier with a low-pass filter (RLF). Unipolar pulses (B) generated at its output are supplied to the peak detector (PD) with programmable decay time constant. A small DC voltage is added to the output signal of the peak detector in order to obtain better noise immunity (block SUM). QRS detection is performed in the comparator block (CMP), in which unipolar pulses (B) are compared with the exponentially falling wave at the peak detector output (C). The output signal of the comparator (waveform D) is supplied to the output cell, which is connected to the microcontroller digital input. One of the tasks of the microcontroller is standardization of the detection pulses. The waveforms registered in the selected nodes of the circuit are presented in Fig. 11.

As it was stated before, the amplitude of the ECG signal obtained from the epicardial electrodes can vary in a wide range, so the automatic gain control is required (AGC). The ECG signal is specific, so classical AGC loops (e.g. dedicated for the audio systems) are inefficient in this application. Instead of it, the authors decided to implement a hybrid AGC algorithm, which uses the dynamic reconfigurability of the FPAA circuit. The unipolar pulses (B) are supplied to the analog input of the main microcontroller in order to measure their amplitude. The amplitude of the pulses is filtered using

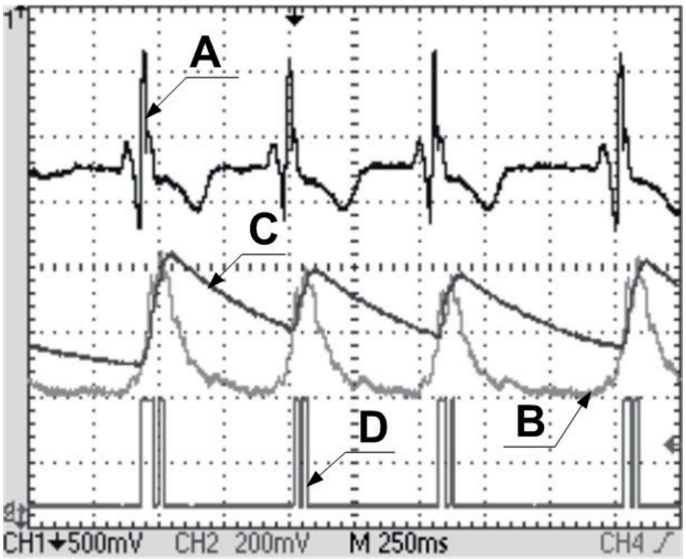


Fig. 11. Waveforms in the analog QRS detector.

a simple first order IIR digital filter to obtain the average amplitude. The filter output value is then compared to the setpoint and when the error is too large, the new configuration data is prepared and sent to the FPAA. It results in a change of the gain of the first stage in dynamic reconfiguration mode. The change of the gain does not disturb the detection process. If the maximum gain is reached and the signal level is still to low then the “reconfig” command is sent to the microcontroller which controls the ADS circuit. It increases the gain by the factor of 2. Similarly, the gain can be gradually increased in case of the rise of the ECG amplitude. This two stage AGC hybrid algorithm allowed to increase the acceptable range of signal amplitude changes to 1:72, which is sufficient for this application.

Another parameter which is important in the QRS detection efficiency is the time constant of the peak detector (PD). The optimum value of this parameter is dependent on the actual heart rate. The actual heart rate is measured in the microcontroller and according to its mean value new reconfiguration data is created and sent to the FPAA. The time constant correction also does not affect the proper operation of the detector blocks.

The QRS detection system was tested using real ECG signals from database and also using signals acquired from epicardial electrodes. The experimental results showed that for all tested data records more than 95% of the QRS complexes were detected with the delay less than 50 ms which is acceptable for the synchronization of the ventricular assist device.

F. Other Applications of the Anadigm FPAA Circuits

Besides the applications presented above, there can be found many others. The FPAA technology has been also used for Atomic Force Microscope control [40], a Self-Adapted PID [41], a reconfigurable filter for a rotating machine [42], state detection in contactless electrical energy transmission [43], emulation of power system dynamics [44], enhancing ADC resolution through dynamic reconfiguration of FPAA [45] and sigma-delta converters prototyping [46]. The number of applications in which dynamic reconfiguration of the FPAA is successfully used is still growing.

VI. Conclusions

This short overview of solutions and applications of programmable analog circuits shows that the reconfiguration is commonly used in modern electronics. Former solutions of the reconfigurable circuits applied usually manual “reconfiguration-of-demand” – e.g. tuning or switching using electromechanical controls. Present applications are usually automatically reconfigurable by means of digital control – microprocessors of FPGA circuits. The selected applications presented in this paper show that parametric reconfiguration is much more often used than functional reconfiguration.

It is also noticeable that most of academic FPAA projects are based on the continuous time principle, while the currently commercially available circuits (PSoC, dpASP) work on the discrete time SC principle. One reason for the popularity of SC technology is the fact that SC circuits allow to obtain a wide span of the parameter change by means of the switching frequency tuning. There is no need of precision tunable current sources required usually in continuous time circuits. Another advantage of the SC circuits is better repeatability of the parameters. Interfacing of the SC blocks to digital systems is also much easier than for the continuous time blocks.

We have to remark that in modern electronic devices the analog path is usually only a part of the mixed system. We can expect that this part will be gradually incorporated into the complex “system on chip”. We can observe this trend in subsequent generations of the Cypress Programmable System on Chip (PSoC).

However, the mixed systems based on PSoC or a combination of the FPAA circuits with a digital system are now popular in prototype applications. In a mass production of electronic systems most of solutions is based on Application Specific Integrated Circuits (ASIC). We can foresee that in the nearest future the dynamic reconfigurable analog cells will be parts of ASIC-s. The design tools of these systems will enclose libraries with IP cores containing dynamic reconfigurable analog cells available in the same way as other components – amplifiers or comparators. In order to obtain it, a method of description of the reconfigurable circuits should be developed. For this purpose VHDL-AMS (Analog and Mixed Systems) language could be adopted. This can be a subject of future research works.

Nowadays the FPAA circuits are a very comfortable and flexible platform for rapid prototyping.

References

[1] 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, 2002.

[2] E. A. Soliman and S. A. Mahmoud, “Digitally Controlled CMOS Balanced Output Transconductor and Application to Variable Gain Amplifier and Gm-C Filter on Field Programmable Analog Array,” *Journal of Circuits, Systems, and Computers*, vol. 14, no. 4, pp. 667–671, 2005.

[3] J. Becker, F. Henrici, S. Trendelenburg, M. Ortmanns, and Y. Manoli, “A Field-Programmable Analog Array of 55 Digitally Tunable OTAs in a Hexagonal Lattice,” *IEEE Journal of Solid-State Circuits*, vol. 43, no. 12, pp. 2759–2768, 2008.

[4] E. A. Soliman and S. A. Mahmoud, "Voltage-mode Field Programmable Analog Array using Second Generation Current Conveyor," in *2012 IEEE 55th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2012, pp. 710–713.

[5] V. C. Gaudet and P. G. Gulak, "CMOS implementation of a current conveyor-based field-programmable analog array," in *Proceedings of Conference of Signals, Systems and Computers*, November 1997, pp. 1156–1159, vol. 2.

[6] D. Abramson, J. D. Gray, S. Subramanian, and P. Hasler, "A field-programmable analog array using translinear elements," in *Proceedings of 5th International Workshop System-on-Chip for Real-Time Applications*, 2005, pp. 425–428.

[7] D. Fernandez, L. Martinez-Alvarado, and J. Madrenas, "A Translinear, Log-Domain FPAA on Standard CMOS Technology," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 2, pp. 490–503, 2012.

[8] Cypress homepage, "CY8C29466, CY8C29566, CY8C29666, CY8C29866: PSoC® Programmable System-on-Chip™," http://www.cypress.com/?docID=45257 (accessed 2014).

[9] Anadigm homepage, "AN121/221E04 User Manual," http://www.anadigm.com/dpasp.asp (accessed 2014).

[10] Anadigm homepage, "AN221E04 Datasheet – Dynamically Reconfigurable FPAA With Enhanced I/O," Datasheet for Anadigm, http://www.anadigm.com/_doc/DS030100-U006.pdf, 2003.

[11] Anadigm homepage, "AN231E04 Datasheet Rev.1.1 – Dynamically Reconfigurable dpASP," Datasheet for Anadigm, http://www.anadigm.com/_doc/D0S231000-U001.pdf , 2008.

[12] S. Chang, B. Hayes-Gill, and C. Paull, "Multi-function block for a switched current ?eld programmable analogue array," in *Proceedings of IEEE 39th Midwest Symposium Circuits Systems*, 1996, pp. 158–161, vol. 1.

[13] T. S. Hall, "Field Programmable Analog Arrays: A Floating-Gate Approach," Ph.D. dissertation, Georgia Institute of Technology, Atlanta, 2004.

[14] A. Basu, S. Brink, C. Schlottmann, and S. Ramakrishnan, "A Floating-Gate-Based Field-Programmable Analog Array," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 9, pp. 1781–1794, 2010.

[15] C. M. Twigg and P. Hasler, "A Large-Scale Reconfigurable Analog Signal Processor (RASP) IC," in *Proceedings of IEEE Custom Integrated Circuits Conference*, San Jose, USA, 2006, pp. 5–8.

[16] F. Henrici, J. Becker, S. Trendelenburg, D. DeDorigo, M. Ortmanns, and Y. Manoli, "A Field Programmable Analog Array using Floating Gates for High Resolution Tuning," in *Proceedings of IEEE International Symposium on Circuits and Systems*, Taipei, Taiwan, 2009, pp. 265–268.

[17] C. A. Looby and C. Lyden, "Op-amp based CMOS field-programmable analogue array," in *Proceedings Of IEE Circuits, Devices and Systems*, April 2000, pp. 93–95, vol. 147.

[18] E. Ramsden, "The ispPAC family of reconfigurable analog circuits," in *Proceedings of The Third NASA/DoD Workshop on Evolvable Hardware*, 2001, pp. 176–181.

[19] Lattice Semiconductor Corporation , "IspPAC30 – In-System Programmable Analog Circuit," Datasheet for Lattice Semiconductor, http://www.latticesemi.com/lit/docs/datasheets/pac/pac30.pdf, 2002.

[20] Zetex Semiconductors Ltd., "Totally reconfigurable analog circuit: TRAC," 1999.

[21] A. Bratt, "Motorola field programmable analogue arrays, present hardware and future trends," in *Proceedings of IEE Half-day Colloquium on Evolvable Hardware Systems*, London, UK, 1998, pp. 1/1–1/5, no. 1988/233.

[22] L. Znamirowski, O. Palusinski, and C. Reiser, "Optimization technique for dynamic reconfiguration of programmable analog/digital arrays," *Analog Integrated Circuits and Signal Processing*, vol. 31, pp. 19–30, 2002.

[23] A. Malcher and Z. Kidoń, "Some properties of FPAA-based Analog Signal Processing," in *Proceedings of 9th International IFAC Workshop on Programmable Devices and Embedded Systems*, Roznov pod Radhostem, Czech Republic, 2009, pp. 188–193.

[24] A. Piłat, "Control Toolbox for Industrial Programmable Analog Controller – Embedding State Feedback Controller," in *Proceedings of 2012 IEEE 17th Conference on Emerging Technologies & Factory Automation (ETFA)*, Kraków, 2012, pp. 1–4.

[25] A. Piłat, "The Programmable Analog Controller. Static and Dynamic Configuration, as exemplified for Active Magnetic Levitation," *Przeglad Elektrotechniczny (Electrical Review)*, vol. 88, no. 4b/2012, pp. 283–287, 2012.

[26] A. Piłat and K. Klocek, "Investigation of chained analog signal processors in programmable analog computer," in *Proceedings of Programmable Devices and Embedded Systems*, Brno, Czech Republic, 2012, pp. 289–293, vol. 11 Part 1.

[27] A. Piłat and J. Kloocek, "Programmable Analog Hard Real-Time Controller," *Przeglad Elektrotechniczny (Electrical Review)*, vol. 89, no. 3a/2013, pp. 38–46, 2013.

[28] A. Baccigalupi and A. Liccardo, "Field Programmable Analog Arrays for Conditioning Ultrasonic Sensors," *IEEE Sensors Journal*, vol. 7, no. 8, pp. 1176–1182, 2007.

[29] P. Falkowski and A. Malcher, "Audio signal processing based on dynamically programmable analog arrays," in *Proceedings of 2010 International Conference on Signals and Electronic Systems (ICSES)*, 2010, pp. 29–32.

[30] P. Falkowski and A. Malcher, "Dynamically programmable analog arrays in acoustic frequency range signal processing," *Metrology and Measurement Systems*, vol. 18, no. 1/2011, pp. 77–89, 2011.

[31] U. Zölzer, *DAFX: Digital Audio Effects*. New York: John Wiley & Sons, 2002.

[32] D. Morales, A. Garcia, E. Castillo, M. Carvajal, J. Banqueri, and A. Palma, "Flexible ECG acquisition system based on analog and digital reconfigurable devices," *Sensors and Actuators*, vol. 165, no. 2, pp. 261–270, 2011.

[33] D. Morales, A. Garcia, E. Castillo, M. Carvajal, J. Banqueri, and A. Palma, "An application of reconfigurable technologies for non-invasive fetal heart rate extraction," *Medical Engineering & Physics*, vol. 35, no. 7, pp. 1005–1014, July 2013.

[34] P. R. S. Sanches, A. F. Müller, L. Carro, A. A. Susin, and P. Nohama, "Analog reconfigurable technologies for EMG signal processing," *Revista Brasileira de Engenharia Biomédica*, vol. 23, no. 2, pp. 153–157, 2007.

[35] G. Domenech-Asensi, J. Martinez-Alajarin, R. Ruiz-Merino, and J. A. Lopez-Alcantud, "Synthesis on FPAA of a Smart Sthetoscope Analog Subsystem," in *Proceedings of FPL '06. International Conference on Field Programmable Logic and Applications*, 2006, pp. 1–5.

[36] A. Malcher, S. Pietraszek, and T. Przybyła, "Hybrid QRS Detection Circuit Based on Dynamic Reconfigurable Field Programmable Analog Array," in *Proceedings of 10th International IFAC Workshop on Programmable Devices and Embedded Systems*, Pszczyna-Gliwice, Poland, 2010, pp. 69–74.

[37] A. Malcher, S. Pietraszek, T. Przybyla, and Z. Kidon, "Real Time QRS Detector Based on Field Programmable Analog Array," *Journal of Medical Informatics and Technologies*, vol. 13/2009, pp. 183–188, 2009.

[38] D. Komorowski, A. Malcher, and S. Pietraszek, "Hybrid system of ECG signal acquisition and QRS complexes detection for special medical devices synchronization," *Journal of Medical Informatics and Technologies*, vol. 22, pp. 227–234, 2013.

[39] J. Mocha, A. Sobotnicki, M. Czerw, and P. Gibinski, "QRS complexes detector for synchronization of ventricular assist device," *Journal of Medical Informatics and Technologies*, pp. 35–41, 2010, vol. 16/2010.

[40] G. Schitter and N. Phan, "Field programmable analog array (FPAA) based control of an atomic force microscope," in *Proceedings of American Control Conference*, 2008, pp. 2690–2695.

[41] I. Lita, D. A. Visan, and I. B. Cioc, "FPAA based PID controller with applications in the nuclear domain," in *Proceedings of 32nd International Spring Seminar on Electronics Technology (ISSE)*, Brno, Czech Republic, 2009, pp. 1–4.

[42] Z. Wei and L. Yuanxiang, "A self-adapted pid system based on intrinsic evolvable hardware," in *Proceedings of International MultiConference of Engineers and Computer Scientists*, Hong Kong, March 2008, pp. 19–21.

[43] R. Stala and M. Gurbiel, "Practical realization of state detection of contactless electrical energy transmission system using field programmable analog array – FPAA," *Przeglad Elektrotechniczny (Electrical Review)*, vol. 81, no. 7-8, pp. 91–97, 2005.

[44] A. Deese, J. C. Jimenez, and C. O. Nwankpa, "Utilization of field programmable analog arrays (FPAA) to emulate power system dynamics," in *Proceedings of IEEE International Symposium on Circuits and Systems*, Taipei, Taiwan, 2009, pp. 1713–1716.

[45] D. P. Morales, A. Garcia, A. J. Palma, M. A. Carvajal, E. Castillo, and L. F. Capitan-Vallvey, "Enhancing ADC resolution through Field Programmable Analog Array dynamic reconfiguration," in *Proceedings of International Conference on Field Programmable Logic and Applications (FPL)*, Heidelberg, Germany, 2008, pp. 635–638.

[46] R. Suszynski and K. Wawryn, "Prototyping of higher order $\Sigma\Delta$ ADC based on implementation of a FPAA," in *Proceedings of 2012 International Conferenc Signals and Electronic Systems (ICSES)*, Wroclaw, Poland, 2012, pp. 1–4.