A Control System for Automated Evaluation and Tuning of ASIC Parameters

Adam Hudec, Richard Ravasz, Daniel Arbet and Viera Stopjakova
Institute of Electronics and Photonics
Faculty of Electrical Engineering and Information Technology
Slovak University of Technology
Bratislava, Slovakia

***Abstract*—This paper deals with design of a control system for automated post-fabrication setting/tuning of ASIC parameters. A new computer application that easily and effectively measures and sets the functional parameter values using graphical user interface (GUI) was developed. A non-standard communication protocol has been implemented on-chip, and the microcontroller (MCU) as a communication protocol converter was employed. Transformation of the substandard communication protocol into the standard one closes the communication channel between the computer and the Circuit Under Test (CUT). The whole control system was verified using a Field Programmable Gate Array (FPGA), MCU and a Logic analyzer.**

***Keywords*—computer-controlled tuning; FDDA; microcontroller; graphical user interface**

I. Introduction

The ongoing development of advanced electronics is leading to greater complex electronic systems integrated on a chip. Also new possibilities of smaller and more robust electronic systems become a part of wearable electronics or Internet of Things (IoT) [1]. On the other hand, this development in electronic system properties brings higher demands on power management, especially in battery-powerred devices or systems using energy harvesters. The guarantee that integrated circuits (IO) with an ultra-low supply voltage value will still work correctly is rather low [2], which is due to process, voltage and temperature fluctuations. The correct operation can be still reached, for example by trimming methods such as fuse or laser trimming [3]. Better and more convenient way is the application of digital circuitry as tuning logic to calibrate/tune selected circuit's parameters by switching signal ways or by connecting banks of resistors or capacitors. Furthermore, the use of tune logic is not destructive as trimming process and any changes could be returned to the previous state. Another advantage of the calibration circuit is a possibility to communicate with a Field Programmable Gate Array [4], a Microcontroller Unit or a computer (PC) via standard or custom communication protocols. Each of these mentioned devices can send or receive data from the tune registers. Data frames could be generated by MCU based on hardware parts like buttons, rotary encoders actions or by a computer application with graphical user interface. Moreover, it could be a part of the fully automated measuremnt and calibration system.

II. Computer control system

In our work, the ultra-low voltage Fully Differential Difference Amplifier (FDDA) [5] with the supply voltage of 0.4 V was used as the CUT. The block diagram of CUT is shown in Fig. 1. Digital tune registers are employed to ensure the digital tuning of selected (critical) parameters of the FDDA. The Offset Calibration (OC) block was used to calibrate the input offset voltage while the Frequency Compensation (FC) block allows tuning of frequency compensation towards ensuring the stability of the whole FDDA. These registers content can be set or get externally using the communication protocol implemented on-chip. Changing values in the tuning registers can be

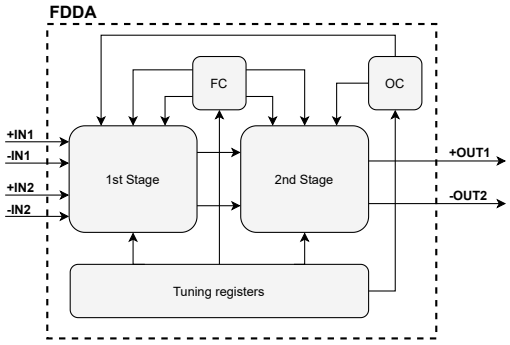


Fig. 1: FDDA block diagram.

realized in various ways. As the introduction describes, control tools like buttons, rotary encoders and other peripherals implemented directly on the test board, or on extended board connected by wires to the testing board, can be used to change registers values. Much better and safer option to properly change the properties of the tested circuit is setting the tune registers values by a computer, which can communicate with the tuning logic. In addition, this solution prevents the occurrence of potential risks during the measurement like ESD, cold junction, physical destruction etc. These arguments forced a development of a control system for simple and automatic measurement with more accurate results. The whole control system block diagram is depicted in Fig. 2. It consists of the following three blocks:

- Computer
- Microcontroller
- Tuning logic

2021 International Conference on Applied Electronics (AE) | 978-80-261-0973-0/21/$31.00 ©2021 10.23919/AE51540.2021.9542883

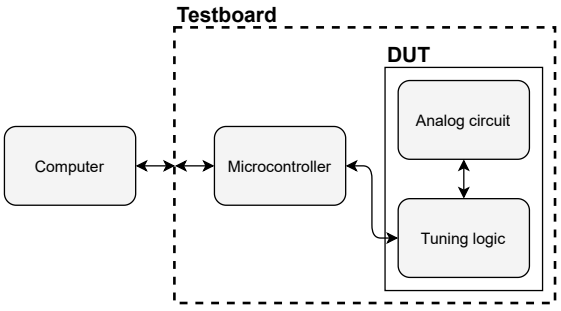


Fig. 2: The control system block diagram.

III. Control Application

The operation of the FDDA depends on fourty two different tuning registers with variable data width. The PC application with graphical user interface for simplified handling measured circuit registers has been developed. Clear and modern GUI is created with Python language using TkInter package with basic components (e.g. buttons, combo boxes, radio buttons and labels). The application functionality in Fig. 3 is divided into three main parts - the Communication Parameters, the FDDA Settings and the Action panel.

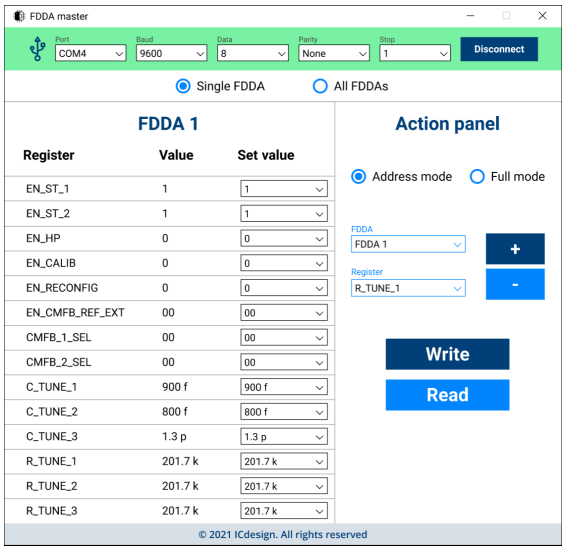
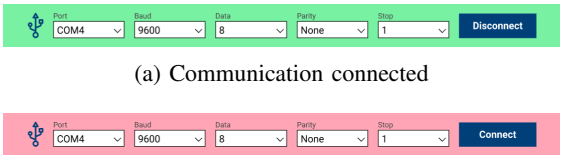


Fig. 3: The control application.

A. Communication Parameters

The Universal Asynchronous Receiver-Transmitter (UART) [6]–[8] communication protocol is used for direct communication with tuning logic via the communication protocol converter. It is an asynchronous serial communication that has to have precisely defined parameters (such as Communication port, Baud rate, Data width, Parity and number of Stop bits) before creation of the communication channel. These individual parameter values are selected from a combo box. The first one control parameter ”Port” is automatically updated list of available communication ports depending on connected devices. Another combo boxes contain basic parameter values for UART protocol (e.g. ”Baud rate” which is a speed of UART communication) could be set to standard baud rate values such as 9600, 57600 or 115200 bauds per second. Sending data width could be set in box ”Data” from 5-bits to 9-bits. Parameters in ”Parity” drop-down list could be one of Odd, Even or None value. ”Stop” combo box control can acquire a value of 1 or 2 bits. Opening and closing of the communication channel is controlled by pressing ”Connect or Disconnect” button. As we can see in Fig. 4 green or red background color indicates the current state of the connection between computer and the tuning logic.



(a) Communication connected

(b) Communication disconnected

Fig. 4: Connection bar.

B. FDDA Settings

The tested analog circuit consists of three FDDAs labeled as FDDA 1, FDDA 2 and FDDA 3 inside the chip. Each FDDA has 14 tuning registers with individual read and write access. A separate snipped working panel with current FDDA selection information and list of tuning registers is shown in Fig. 5. Each register row is divided into the three columns and the first one called ”Register” reffers to the register name. The following column shows the regster current value. However, read values in the middle column are not updated automatically and require click action on the ”Read” button in the Action panel. A combo box with options which varies in every register type is situated in the last column. The first eight registers values have 1-bit or 2-bit width binary number with enabling or selecting function. The last six registers values are coded or decoded as a real values of resistors and capacitors banks implemented on-chip.

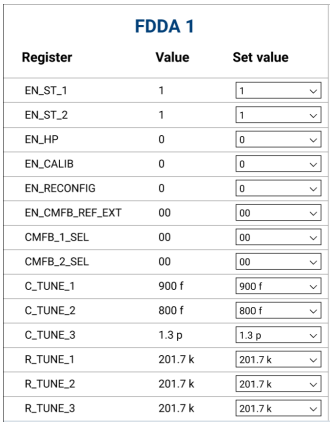


Fig. 5: Panel of registers.

C. Action Panel

Reading or writing registers content is not automatic process but it is executed by buttons in Action panel, which are also depicted in Fig. 6. At the top of the panel, there are two radio buttons for selecting type of sending configuration to the one or all selected FDDA registers. In case of Address mode selection, the FDDA index could be choosed from the combo box labeled ”FDDA”. The following ”Register” drop-down menu specifies type of register which will be written or read by ”Write” and ”Read” buttons. Moreover, when the R_TUNE_X or C_TUNE_X register is selected, the ”Plus” and ”Minus” buttons will appear. Their functionallity is incrementing or decrementing value by one and send the current configuration to the tuning register and update value in the ”Value” column. Switching the radio button to the ”Full mode” will reduce the number of the interactive components in the Action panel. The ”Plus”, ”Minus” buttons and ”FDDA” combo box will disappear and ”Write”, ”Read” buttons deal with all selected FDDA registers.

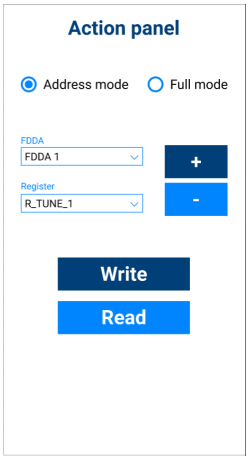


Fig. 6: Action panel.

IV. Communication Protocol Converter

As mentioned in previous chapters, the FDDA circuit properties can be changed by tuning registers. The data in these registers are filled via a custom half-duplex serial communication bus designed at Department of IC Design and Test. It contains three input ports: clock (CLK), reset (RST_N), data in (DATA_IN) and one data output port (DATA_OUT). The computer direct connectivity to the tuning logic is not possible because each of them runs on a different communication protocol. Therefore, a communication protocol converter has to be implemented in between. In our case, the MCU is used to receive data from PC and generate a proper data frame to the tuning logic by Bit Banging technique. It is a method for serial communication provided by MCU General Purpose Input Output (GPIO) pins where the whole communication is created and controlled by software instead of hardware [9]. Software controlled communication does not reach the speeds of hardware-driven one because it is limited by instruction executing time, but in this case, the high communication speed is not required.

V. Experimental Verification

A complete control system has been verified by development boards. Data flow between the MCU and FPGA was sniffed on individual signal lines by 8-channel digital analyzer with 24 MHz sample rate as shown in Fig. 7. Tuning logic is sensitive on positive clock edge and the registers are initialized to default values with synchronous reset. Communication data flow example shown in Fig. 8 presents write/read data from a specific tuning register.

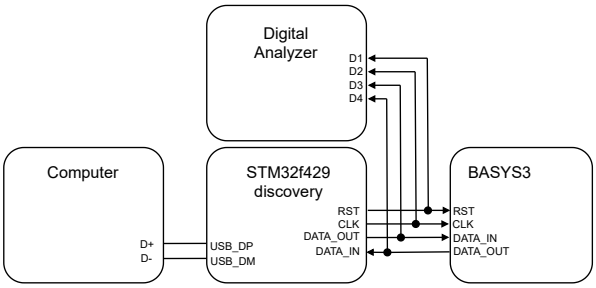
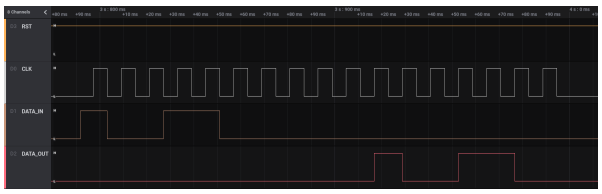


Fig. 7: Block diagram of the verification system.

(a) Writing data.



(b) Reading data.

Fig. 8: Data frames captured.

VI. Conclusion

In this paper, the control system for verification and tuning the FDDA circuit parameters has been presented. The computer application with dedicated graphical user interface to write or read tuning registers was developed. Hardware description of the tuning logic with registers was implemented into the FPGA BASYS 3 board and verified by STM32F4-Discovery as a communication protocol converter applying the Bit Banging method. Used development kits are shown in Fig. 9. The achieved experimental results prove that the designed communication system as well as the tuning logic work correctly. The developed tuning logic control system will be used for verification of analog ASICs by the test board shown in Fig. 10.

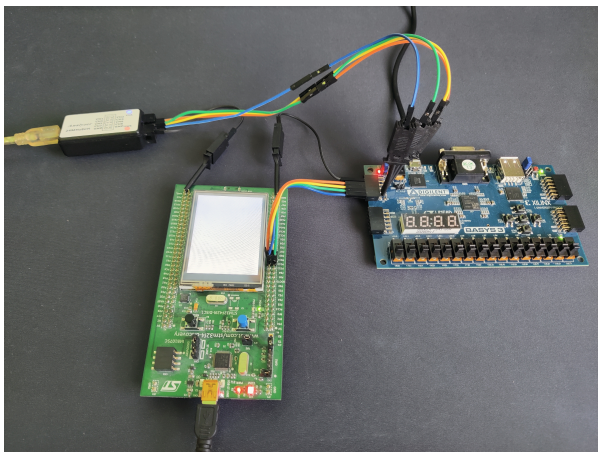


Fig. 9: The verification setup

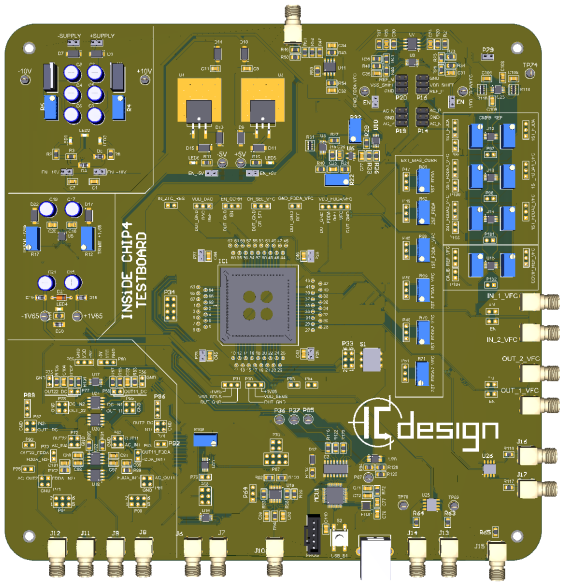


Fig. 10: Developed test board.

VII. Acknowledgement

This work was supported in part by the Ministry of Education, Science, Research and Sport of the Slovak Republic under grants VEGA 1/0731/20 and VEGA 1/0760/21, and the Slovak Research and Development Agency under grant APVV-19-0392.

References

[1] M. Alhawari, B. Mohammad, H. Saleh, and M. Ismail, *Energy harvesting for self-powered wearable devices*. Springer, 2018.

[2] W. Greig, *Integrated circuit packaging, assembly and interconnections*. Springer Science & Business Media, 2007.

[3] M. Alafogianni, R. Penlington, and M. Birkett, "Resistor trimming geometry; past, present and future," in *IOP Conference Series: Materials Science and Engineering*, vol. 104, no. 1. IOP Publishing, 2016, p. 012002.

[4] S. Ramachandran, *Digital VLSI systems design: a design manual for implementation of projects on FPGAs and ASICs using Verilog*. Springer Science & Business Media, 2007.

[5] D. Arbet, M. Kováč, G. Nagy, and V. Stopjaková, "High dynamic range and low distortion fully differential difference amplifier in cmos," in *2015 25th International Conference Radioelektronika (RADIOELEKTRONIKA)*, 2015, pp. 114–117.

[6] A. Gupta, *The IoT Hacker's Handbook*. Apress, Berkeley, CA., 2019.

[7] M. Sharma, N. Agarwal, and S. R. N. Reddy, "Design and development of daughter board for usb-uart communication between raspberry pi and pc," in *International Conference on Computing, Communication Automation*, 2015, pp. 944–948.

[8] U. Nanda and S. K. Pattnaik, "Universal asynchronous receiver and transmitter (uart)," in *2016 3rd International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 01, 2016, pp. 1–5.

[9] E. Odunlade, "Introduction to bit banging: Spi communication via bit banging," https://circuitdigest.com/article/introduction-to-bit-banging-spi-communication-in-arduino-via-bit-banging, 2018.