# Design of Serial Analyzer for Characterization of IoT Devices

Ameen Kubra
PG Student
Department of Electronics and Communication Engineering
GSSSIETW, Mysuru

Sanket N Shettar
Assistant professor
Department of Electronics and Communication Engineering
GSSSIETW, Mysuru

***Abstract*-One of the major challenges of today's embedded system design is that the debugging, which consumes most of the design point in time and as such there is a demand to reduce the debugging time. The proposed Serial Analyzer is designed to restore the traditional time -consuming debugging and provide a low cost and efficient means to debug the serial communication protocol like UART, I2C, SPI. This paper proposes the debug needs of IoT devices and systems. The proposed Serial Analyzer also acts as a tool that allows numerous serial data to be acquired simultaneously. To acquire different digital waveforms and data a Serial Protocol Analyzer is a multi-channel device. The major logic for Serial Analyzer lies on a FPGA board with a powerful embedded processor. A graphical user interface has been build to communicate with the Serial Analyzer. The firmware is built for the processor to interact with GUI and the RTL logic in FPGA. Serial Analyzer is designed and developed to debug, analyze and decode serial protocols in IoT systems.**

***Keywords: Internet of things (IoT), serial protocols, FPGA, serial analyzer, GUI.***

## I. INTRODUCTION

One of the major challenges of today's embedded system design is that the debugging, which consumes most of the design point in time and as such there is a demand to reduce the debugging time. The proposed Serial Analyzer is designed to restore the traditional time -consuming debugging and provide a low cost and efficient means to debug the serial communication protocol like UART, SPI, I2C.

The design of Serial Analyzer is use to debug serial communication protocols like UART, SPI, I2C. This tool will assist out designers with debugging their complex embedded systems. The Serial Analyzer comes with 16 channels, multiple trigger options and a easy to use front end GUI. It has the ability to both analyze and decode popular serial protocols.

### *A. FPGA (Field Programmable Gate Array)*

FPGA (Field Programmable Gate Array) has become one of the current implementation technologies for complex digital systems in both operational and prototyping systems. It is used extensively and became a popular prototype platform while develop customer electronic product such as designing of a digital circuit. Very often they are used as asort of core processor for existing processors, for example, as the PC-bus add-on boards, or the boards for the other bus types. In addition, FPGAs are always became one of the recommended devices used to implement simple interface

circuit or complex state machine to satisfy a different system requirement. It is because it can improve the system integration, reliability and reduce power consumption. It also contains a programmable logic components as known as "logic block" and wired together in a special hierarchy of configurable interconnect. Besides that, it can configure into a logic blocks to perform complex combinational functions in simple logic gates such as AND and XOR.

## II. . HARDWARE DEVELOPMENT

### *A. SERIAL ANALYZER HARDWARE ARCHITECTURE*

Serial Analyzer Hardware architecture consists of many blocks as shown in the Figure 1. The Figure shows the major block which are discussed in next sections in details.

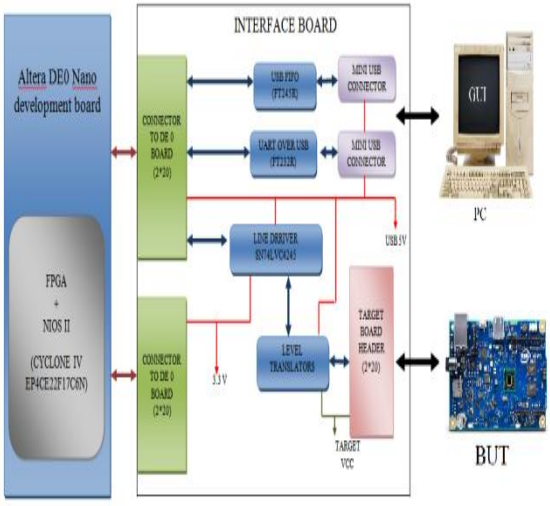


Figure1. Hardware Architecture

The hardware design of Serial Protocol Analyzer is explained into four sub modules as follows

- Processing Module
- USB Module
- Buffer Module
- Power Supply Module

Processing Module: All the logic for Serial Protocol Analyzer is centered on Cyclone IV FPGA series from Altera. The crucial part of logic processing takes place in FPGA.The part number for the same is EP4CE22F17C6N.

USB Module: The USB section consists of USB parallel FIFO IC and USB UART IC.

Buffer Module: This section enables the user to connect target boards of different voltage levels (1.2V to 5V).FPGA will be working with 3.3V ,but the target board can work anywhere with in the range of 1.2V to 5V.So there is a need for level translation and buffer for protection and compatibility.
Power Supply Module: The power to hardware unit comes from USB port and target board.

### *B. HARDWARE ARCHITECTURE SCHEMATICS*

The Design starts with design specification, which lists the required components for the design. The components symbols which are not in library are requested through a centralized library manager AGILE. The Figure 2 shows the design flow.

The schematics is designed using CADANCE CAPTURE CIS , detailed procedure is discussed in next chapter. The Netlist is generated which is imported into CADANCE PCB editor for layout. Also Bill of Material (BOM) is generated that listes components based on cost, specs ,part number etc.

The layout is designed involving placement and routing of components based on requirements. A brd file is generated and checked for DRC (Design rule Check) and LVS ( Layout versus Schematic) errors. Then it is sent for fabrication. Figure 1 shows the basic design Flow.

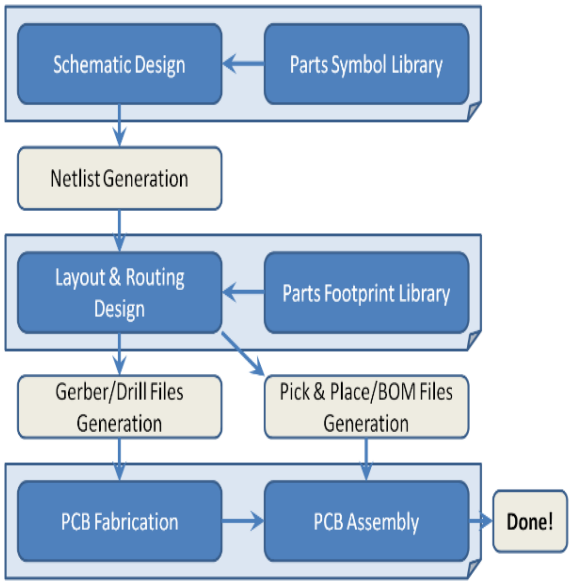


Figure 2. Hardware Design Flow

## III. RTL DESIGN

There is lot of advantages in using FPGA for designing embedded system. The main attraction is the flexibility in design. Interfacing with different modules is easy in FPGA as several IP cores are available. Speed and integrity of FPGAs are adorable.

The logic for decoding and capturing serial data line will be implemented in FPGA . FPGA logic design will interpret the commands given by the user and capture necessary signals.

Cyclone 4 FPGA from ALTERA is utilized in design. It has a speed grade of 6. For the purpose of processing data a soft NIOS Processor core is utilized. IP core helps to in add much functionality with less effort.

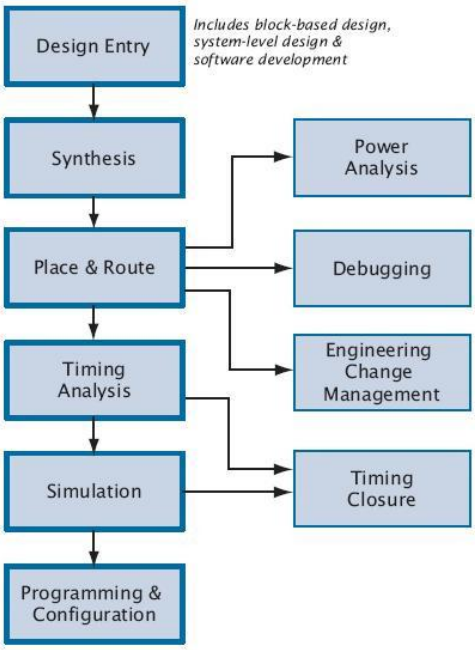


Figure 3 Basic design Flow

The following steps describe the basic design flow is shown in figure 3 for using the Quartus II GUI:

- To create a new project and specify a target device or device family, on the File menu, click New Project Wizard
- Use the Text Editor to create a Verilog HDL, VHDL, or Altera Hardware Description Language (AHDL) design.
- Use the Block Editor to create a block diagram with symbols that represent other design files, or to create a schematic.
- Use the MegaWizard® Plug-In Manager to generate custom variations of megafunctions and IP functions toinstantiate in your design, or create a system-level design by using SOPC Builder or DSP Builder.
- Specify any initial design constraints using the Assignment Editor, the Pin Planner, the Settingsdialog box, the Device dialog box, the Chip Planner, the Design Partitions window, or the Design Partition Planner
- (Optional) Perform an early timing estimate to generate early estimates of timing results before fitting.
- Synthesize the design with Analysis & Synthesis.
- (Optional) If your design contains partitions and you are not performing a full compilation, merge the partitions with partition merge.
- (Optional) Generate a functional simulation netlist for your design and perform a functional simulation with an EDA simulation tool.
- Place and route the design with the Fitter.
- Perform a power estimation and analysis with the PowerPlay Power Analyzer.
- Use an EDA simulation tool to perform timing simulation for the design.
- Use the TimeQuest Timing Analyzer to analyze the timing of your design
- (Optional) Use physical synthesis, the Chip Planner, LogicLock™ regions, and the Assignment Editor to correct timing problems.
- Create programming files for your design with the Assembler, and then program the device with the Programmer and Altera programming hardware.
- (Optional) Debug the design with the SignalTap®II Logic Analyzer, an external logic analyzer, the SignalProbe feature, or the Chip Planner.
- (Optional) Manage engineering changes with the Chip Planner, the Resource Property Editor, or the Change Manager.

## IV. DEVELOPMENT OF SOFTWARE TOOLS

This Chapter describes Software functionalities for Serial Analyzer. It contains the explanation for the PC application software / Graphical User Interface and the firmware . The communication in the target board can be easily monitored with the help of this GUI. The user will interact with the Serial Analyzer through this GUI. This document gives an details about PC application software for user interaction and firmware that drives the RTL logic as per commands from the user. The software design of Serial Analyzer system can be explained two parts - PC application software and Firmware on NIOS II core.

GUI is the main part of PC application software. Various applications of Serial Analyzer can be initiated from GUI and result will be displayed on the GUI for analysis. Tool used for developing GUI is Microsoft Visual C#. The Default screen for Serial Analyzer is shown in Figure 4. The user can select the target board protocol by clicking the respective buttons given at the left side of the screen.When a protocol has been selected, the next step is to configure the channels. 16 channels are available and the user can select any channel amongst these.

In case of UART , the channels for TX and RX are selected from the available 16 channels . The next step is defining the baud rate . Proper care must be taken here to match the baud rate value to that of target board failing which an erroneous data will be displayed on GUI.

The following baud rates are available

- 9600
- 14400
- 19200
- 28800
- 31250
- 38400
- 57600
- 115200

Trigger can be configured using the trigger options given on the GUI. Data trigger value, Trigger channel, pre trigger & post trigger value and capture size can be defined using these options.

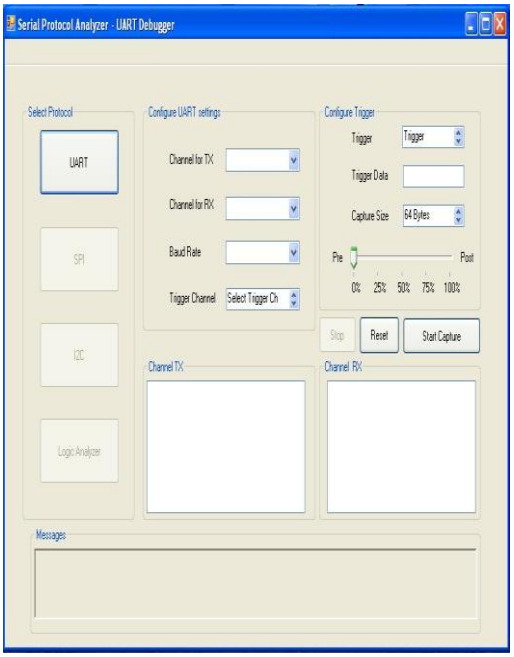


Figure 4. GUI at startup

## III. CONCLUSION

The design and Implementation of Serial Analyzer is a diverse project which would aid most of the designers to reduce the debugging time and cost .The RTL logic for the Serial Analyzer is based on a FPGA centered around a powerful NIOS II processor. Therefore the project can be extended to include all popular serial protocols like USB, CAN and so on.

Serial Analyzer has been studied to find its potential and necessary trade -offs when used as a logic analyzer. Within the goals of having a complete system based on FPGA to do logic analysis and serial debugging, the entire project has been developed with utmost optimization for speed, memory, RTL logic, power and size.

## REFERENCE

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," FutureGener. Comput. Syst.,vol.29, no. 7, pp. 1645–1660, Sep. 2013.

[2] J. Buckley, "From RFID to the Internet of Things pervasive networked systems,"Conference Centre Albert Borschette (CCAB), Brussels, Belgium, Mar. 2006.[Online].Available:ftp://ftp.cordis.europa.eu/pubist/docs/ka4/au_conf670306_buckley_en.pdf.[accessed on 2-1-2017].

[3] D. Evans, "The Internet of things: How the next evolution of the Internet is changing everything," Cisco IBSG, San Francisco, CA, USA, Apr. 2011. [Online]. Available: http://www.cisco.com/web/about/ac79/ docs/innov/IoT_IBSG_0411FINAL.pdf. [accessed on 5-1-2017].

[4] The Zettabyte Era-Trends and Analysis. Cisco, May 2013. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/VNI_Hyperconnectivity_WP.html. [accessed on 5-1-2017].

[5] D. Lake, A. Rayes, and M. Morrow, "The Internet of Things," Internet Protocol J., vol. 15, no. 3, pp. 10–19, Sep. 2012. [Online]. Available: http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_153/153_Internet.html[accessed on 9-1-2017].

[6] ARM targets Internet of Things with New Low-Power Chip. Institute of Nanotechnology.[Online].Available:http://www.Instituteofnanotechnology.co.uk/arm-targets-Internet-of-things-withnewlow-power-chip.[accessed on 9-1-2017].

[7] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2012–2017.Cisco,Feb.2013.[Online].Available:http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html. [accessed on 9-1-2017]

[8] O. Bello and S. Zeadally, "Communication issues in the Internet of Things," in Next Generation Wireless Technologies: 4G and Beyond. London, U.K.: Springer-Verlag, 2013, pp. 189–219.

[9] A. Liotta, Why the Internet Needs Cognitive Protocols. New York, NY, USA: IEEE Spectrum, May 2013. [Online]. Available: http://spectrum.ieee.org/computing/networks/why-the-internet-needs-cognitive-protocols. [accessed on 12-1-2017].

[10] Amirhossein Shahshahani, Andrey Tolstikhin, Zeljko Zilic, "Enabling Debug in IoT Wireless Development and Deployment With Security Considerations", 2016 IEEE

[11] M. W. Chiang, Z. Zilic, K. Radecka, and J. S. Chenard, "Architectures of increased availability wireless sensor network nodes," in Test Conference, 2004. Proceedings. ITC 2004. International, Oct 2004, pp.1232–1241.

[12] R. Zhang, Z. Zilic, and K. Radecka, "Energy efficient software-based self-test for wireless sensor network nodes," in VLSI Test Symposium, 2006. Proceedings. 24th IEEE, April 2006, pp. 6 pp.–191.

[13] J. Daniel, S. Greenfeld and J. Schmalzel, "Serial Analysis with a PalmTM Organizer", 2000 IEEE

[14] Shouqian Yu Lili Yi Weihai Chen Zhaojin Wen, "Implementation of a Multi-channel UART Controller Based on FIFO Technique and FPGA", 2007 IEEE.

[15] Nurul Fatihah Jusoh, Azlina Ibrahim, Muhamad Adib Haron and Fuziah Sulaiman, "An FPGA Implementation of Shift Converter Block Technique on FIFO for UART", 2011 IEEE.

[16] V.Vijaya, Rama Valupadasu, B.Rama Rao Chunduri, "FPGA Implementation of RS232 to Universal serial bus converter", 2011 IEEE.

[17] Ken Bryan F. Fabay, John Cris F. Jardin, Kervin John C. Jocson, Bernard Raymond D. Pelayo Anastacia B. Alvarez, Joy Alinda R. Madamba, Louis P. Alarcon, "A Test Port for Interfacing and Debugging ARM9 Processors Implemented in FPGA", 2012 IEEE.

[18] Jie Yan, Xiaosu Xu, Lihui Wang, "Implementation Of High Speed Serial Communication Based On Fpga", 2013 IEEE.

[19] Jingwen Fu, Tianle Zhang, Jiaqi Li, "Method and Implementation of FPGA Configuration Based on Serial Communication", 2015 IEEE.

[20] Ioana Mot, Oana Boncalo, Alexandru Amaricai, "Performance enhancement of serial based FPGA probabilistic fault emulation techniques", 2015 IEEE.