Bidirectional Packet Protocol for FPGA Communication

31.03.2020

—

Apoorva Arora

About Me

I am a graduate student at TU Delft, specialising in Embedded Software and Networking. I enjoy developing creative and novel solutions for problems impacting business and lives. I have interest and experience in embedded software development, FPGA based real time systems design, MAC protocol and algorithmic design for IoT networks/LPWAN technologies (LoRaWAN). I am also interested in wireless and cellular communication technologies and Artificial Intelligence. I have a good experience and knowledge of programming tools like VHDL, C, C++, Python. I aim to become an expert in my field with excellent leadership skills. I want to extend my experience to include full stack network design from Data Link to Application layer.

The project offered by Apertus gives me an excellent opportunity to use my knowledge, learn and enhance my experience in FPGA based real time system design for data link/ MAC layer communication while contributing to the open source community.

Contact Info

Email: apoorvaarora.ms@gmail.com
Alternate Email: A.Arora-1@student.tudelft.nl
Phone No.: +31-626-466-875
FreeNode IRC Nick: apoorva_arora
Github Link: https://github.com/Apoorva-ar/GSOC_VHDL_TASK
CV: CV link

Past Projects / related works

- **Communication Engineer: Silverwing Dream Team, TU Delft-** Worked on networking architecture design for on-board sensor communication with the heterogeneous Flight Controller Board (Lattice MACHXO3) through various protocols, mainly SPI, I2C and CAN.
- **Student Researcher, KPN, The Netherlands -** Designed an algorithm for effective and efficient beacon packet scheduling for LoRaWAN Class B operation. Designed a system level simulator using Simpy with all the MAC layer details of LoRaWAN protocol. Received **KPN Inventor award** and a **Patent** Filed for the algorithm at the European Patent Office.
- **Graduation Project, TNO, The Netherlands -** Working on 5G cellular communication technology and artificial intelligence. The work involves designing a resource scheduling algorithm for Radio Access Network Slicing using AI.
- **Quadrotor Drone Embedded Software Design -** As a part of the project, designed and implemented a communication protocol for error free communication between the flight control board and the PC over USB. The communication allowed full flight control and stabilization of a quadrotor drone. Used Embedded C, concepts of Kalman filters, butterworth filters, PID controllers.

Project Idea

Apertus-Axiom Beta utilizes two Lattice MachXO2s to expand the IOs interfaces for the main processing unit (ZYNQ-SoC). The channel comprises two LVDS pairs sharing a common clock with ZYNQ.
The main idea behind designing a bidirectional packet protocol is to utilize the bandwidth efficiently and support various bus protocols on the MACHXO2 namely SPI, I2C and GPIO-based, ensuring predictable latency.

Major Goals:-

- Defining the protocol that works over a single LVDS pair.
- Implement the bus mapping/ circuit switching for various bus protocols mentioned above.
- Optimize the serial high speed link with relevant SERDES encoding scheme.

Challenges:-

- Maintaining predictable delays for high priority real time services like controlling time sensitive signals.
- Scheduling the packets adhering to priority of the service as well as efficiently utilizing the capacity of the link.

Project Approach

In order to access the sysIO of two MACHXO2s via single LVDS channel efficiently, accurately and with predictable latency, a communication architecture needs to be designed in order to maintain the above mentioned constraints.

Hence, a packet protocol is proposed which works in Asynchronous transfer Mode (ATM) with fixed packet packet length and hence predictable latency. This communication protocol also supports different types of services with different latency requirements.

The packets can be categorised mainly into three different categories based on priorities:-

- First Priority : These packets carry payload for real time services, mainly used to trigger sensors which require deterministic delay.
- Second Priority : These packets carry payload (control/data) for all the devices connected to remote bus systems like I2C, SPI etc. which need to be accessed via routing fabrics.
- Third Priority : These packets usually carry payloads with status information and hence have lowest priority.

Therefore, the packets are scheduled based on the priority of the service requested by the different processes. The data is sliced into these packets and padded with relevant source and destination addresses and data order information for smooth and accurate routing. The packets communicate with the external bus protocols embedded in MACHXO2 as well as various sensors via wishbone interface.

To support this protocol, a streaming architecture is proposed to handle the data and control signals accurately. The physical LVDS layer is supported via SERDES CODEC for high speed and fault tolerant serial data communication.

Packet protocol

The architecture of the packet is described below in figure P.1. The payload is padded with 6 headers namely:-

1. Priority : Contains information about the priority of the service
2. Checksum/Parity : For error handling
3. Source Address : source address of the process
4. Destination Address : Destination address of the peripheral
5. Packet Type : Contains the type of information in the payload :-
    a. Configuration
    b. Write Data
    c. Read Data
    d. Status
6. Data sequence number : Indicates the data sequence

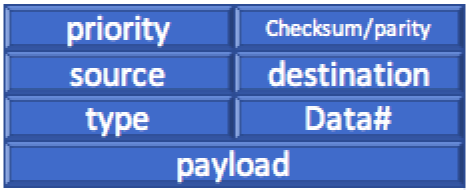

Figure: P1
Packet architecture

System Architecture

The system architecture is shown in figures S1 and S2. Different processes can demand and transmit data to the MACHXO2 peripherals via the architecture in a time deterministic manner. The FSM running on ZYNQ is responsible for construction and storing of packets in respective FIFOs based on the priority of the service. The FSM then transmits the packet (S1) or receives the packet (S2) from the LVDS channel via SERDES.
On the other hand, The FSM in MACHXO2 is responsible for analysing the packet, storing the information into relevant FIFO based on priority and communicating with the peripherals via wishbone bus interface.

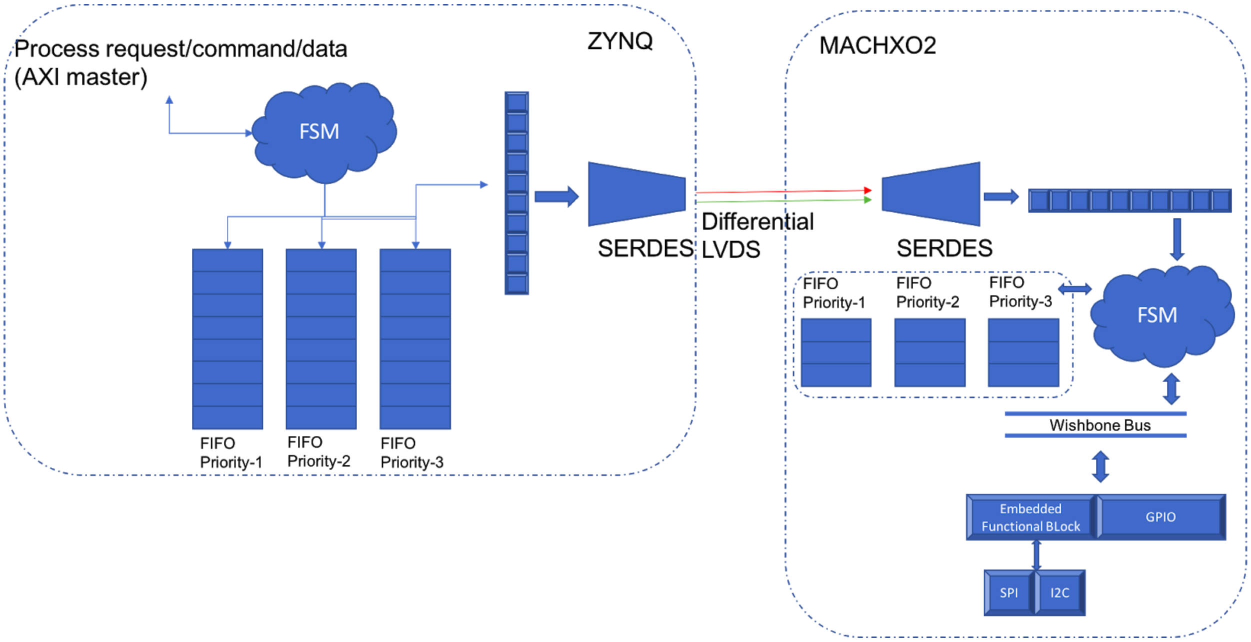

Figure: S1
Data and control flow (ZYNQ to MACHXO2)

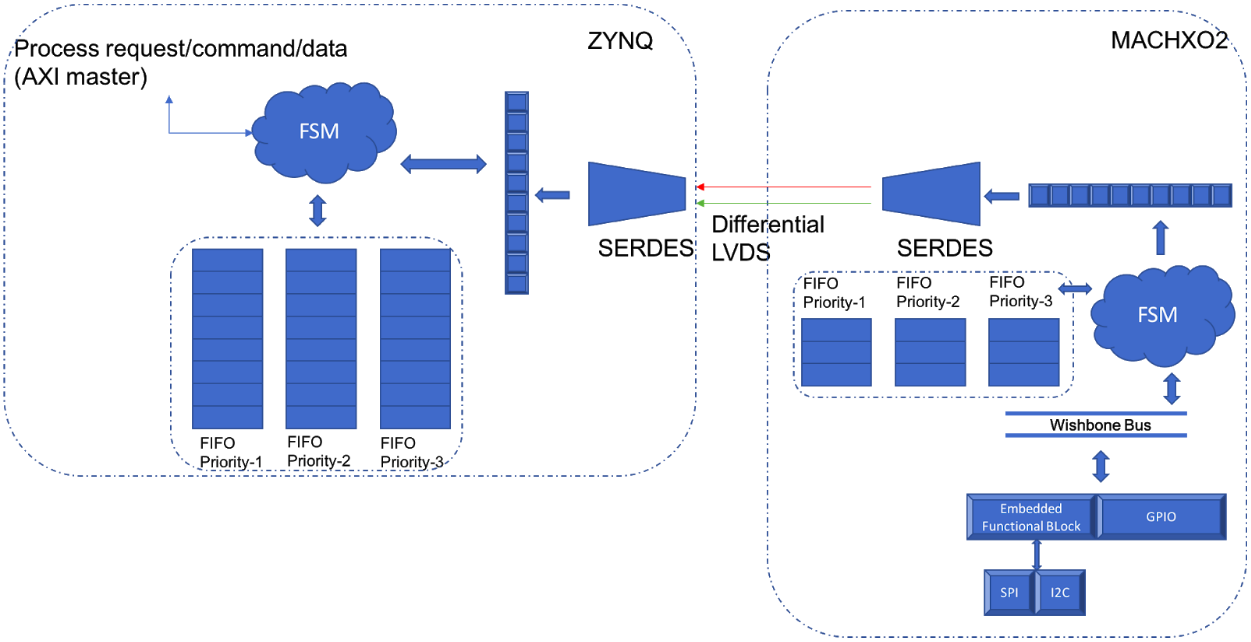

Figure: S2
Data and control flow (MACHXO2 to ZYNQ)

Timeline

Pre GSoC Result

I will gather and equip myself with more information on packet switching, scheduling and routing techniques for ATM type of communication relevant for the project.

Community Bonding Period

My main aim will be interaction with the mentors over details of the protocol and the architecture and discuss implementation strategies for different modules involved in the project.

Week 1

- Study AXIOM image processing pipeline.
- Conduct timing analysis of the data and control signals in different blocks of the pipeline relevant to the project.
- Discuss system architecture, packet protocol architecture as well as I/O specifications with the mentors and decide a final design strategy for the project.
- Improve current SERDES CODEC with testbench simulations.

Week 3

- Implement the packet based protocol for unidirectional path (ZYNQ to MACHXO2) with test bench simulations.

Week 4

- Implement user wishbone FSM to access hardened SPI and I2C on the MACHXO2 with test bench simulations.

Week 5

- Deliverables in the first evaluation:-
  - VHDL code that can transmit data requested by a process (over axi lite) to external peripherals on MACHXO2 via the packet protocol over unidirectional LVDS pair.
  - Simulation results for data and control/configuration transfer to wishbone emulating SPI and I2C embedded functional block for the MACHXO2.
  - Registering total system latency and performance.
- Discuss mentor's feedback and make changes accordingly.

Week 6 (University exams)

- Implementing and testing the architecture with high priority services.

Week 7

- Implement the packet based protocol for unidirectional path (MACHXO2 to ZYNQ) with test bench simulations.
- Simulate data transfer from wishbone slave on MACHXO2 to processes running on ZYNQ.

Week 8

- Discuss the simulation results with the mentors.
- Execute changes (if required).
- Buffer time.

Week 9

- Deliverables in the Second evaluation:-
  - VHDL code that can receive data requested by a process (over axi lite) to external peripherals on MACHXO2 via the packet protocol over unidirectional LVDS pair.
  - Simulation results for data transfer from wishbone slave on MACHXO2 to processes running on ZYNQ.
  - Registering total system latency and performance.
- Discuss mentor's feedback and make changes accordingly.

Week 10

- Implement complete bidirectional protocol architecture and simulate

Week 11

- Integrating the architecture with Axiom Beta pipeline and testing on various sensors.
- Documenting the results.

Week 12

Buffer week for unforeseen circumstances and final report generation and presentations.

Week 13

- Final Deliverables:-
  - Top level VHDL module of the IP
  - System level test bench
  - Detailed report of the project.
  - Future aspects of the project and strategies.

What interests you most about the apertus AXIOM project?

The project enables the main processing unit to expand its interfaces efficiently and handle the traffic in time deterministic manner. This architecture itself is required in various industries which require low powered, real time and cost effective embedded system design. I believe that the concepts learnt and experience acquired by working on this project will help me in my career at various stages of Digital system design.

Further, the very idea of an open source hardware/software camera intrigues me. It is a brilliant initiative for both- the photography enthusiasts and the research community. It allows the global community of tech savvys to play with the camera and develop new applications, accelerating technology development in a cost-effective manner. As a tech savvy myself, contributing to the development of this project is a great learning and self-exhilarating opportunity for me.

As mentors of the project. How can we get best out of you?

Contribution of mentors in improving, shaping and concretizing my ideas, for instance, in system design and protocol design will keep the work time-efficient. Weekly update meetings would play an important role to remain connected and solve any problems I might face.

Is there anything that you’ll be studying or working on whilst working alongside us?

Yes, I will be having a mini project and a course exam to take in the sixth week of the GSOC .

Are there any techniques and tools which you use to keep yourself organised?

I keep a record of deadlines (sometimes self made) of the projects/tasks and sync them with my Google calender. I categorise the tasks according to their importance and regularly keep switching tasks to stay efficient and motivated. Further, I keep documenting work related information gathered through different sources such as casual reading or meetings in google docs so that I can access it anytime and anywhere.