

A GIGABIT ETHERNET LINK FOR AN FPGA BASED BEAM LOSS MEASUREMENT SYSTEM

M. Kwiatkowski, M. Alsdorf, B. Dehning, W. Vigano, C. Zamantzas,
CERN, Geneva, Switzerland

Abstract

A new Beam Loss Monitoring (BLM) system is under development at the European Organisation for Nuclear Research (CERN) within the LHC Injector Upgrade (LIU) project. The multi-channel system will have the ability to measure beam losses from various types of detectors with high precision and wide dynamic range. Several modes of data acquisition are supported. The data rate in the single-channel mode is 16 Mbps and in the multi-channel mode 128 Mbps. The Gigabit Ethernet link is implemented in an FPGA, which allows both a high throughput and a quick validation of the digital data processing algorithms using standard PCs in the initial stages of the development. Both TCP and UDP protocols were explored. The implementation of the Ethernet link is flexible and proved to be highly reliable, leading to its planned use in other measurement systems developed at CERN. The implementation details of the Ethernet link and the results achieved will be described in this paper.

INTRODUCTION

During the upgrade of each of the LHC Injectors at the European Organisation for Nuclear Research (CERN), an up-to-date Beam Loss Monitoring (BLM) system will need to be included for the monitoring of the beam losses and machine protection. For this reason, a new BLM system is under design [1].

The BLM Dual Polarity (BLEDP) module is the first stage of that system with the purpose to collect and digitise the current output from the detector. The acquisition crate is able to host up to eight of these modules. The BLEDP module is able to digitise in parallel the current from eight input channels each having a wide range from 10 pA up to 200 mA. To accomplish this, the range is split into two partially overlapping sub-ranges and for each of them a different circuit and measurement method is used. The current in the range from 10 pA to 10 mA is measured by making use of a Fully Differential Frequency Converter (FDFC). The current in the upper range, i.e. from 100 μ A up to 200 mA, is measured directly by the ADC as a voltage drop on the input resistor and it is referred as the Direct Analogue to Digital Conversion (DADC) method. More about the analogue front-end and the used measurement methods can be found in [2].

The BLEDP module is equipped with an FPGA device which is responsible for processing the FDFC data [3] and controlling the analogue circuitries. In the default configuration, the BLEDP card is expected to provide digitised

data to the processing electronics through two point-to-point connections with bidirectional multi-gigabit optical links. For the development phases as well as later to serve as a stand-alone measurement system, a Gigabit Ethernet link was implemented in the FPGA device. In this configuration, each of the modules in the acquisition crate have a separate link to the network. The measurement data collected is sent to a dedicated supervision application which is used for on-line view as well as off-line data storage and analysis. This paper will focus on the implementation of the Ethernet firmware for the BLEDP card but it will also show similar variants of the firmware which may be reused in other measurement systems developed at CERN.

BLEDP DATA FRAMES

The data frame, shown in Fig. 1, was designed to satisfy both of the measurement methods which are automatically selected by the BLEDP module depending on the input current level observed. In addition, the protocol must be capable of transmitting acquisition data from a single or multi-channels. The channel number or multi-channel transmission is requested by a command send by the connected client. The measurement method and the channel information are attached as a header to the data sample. That is, each “BLEDP frame” is 32 bits wide in total and it consists of the 6 bit header, 6 bit sequence number and the measurement data. The sequence number is used by the client application to verify data continuity, which could potentially be broken due to not sufficient throughput of the network. The size of one data sample in both measurement modes is 20 bit. The BLEDP is sampling the current flowing to each of its inputs at 500 kHz rate therefore the data rate is 16 Mbit/s per channel. The data samples are collected in so called “BLEDP bundle” which contains 350 BLEDP frames. This size of the BLEDP bundle was selected to

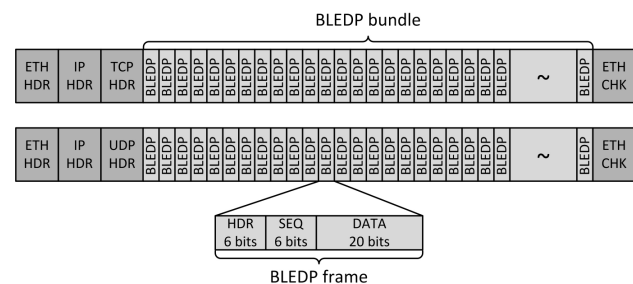


Figure 1: Structure of the BLM Data Frame transmitted over an Ethernet packet.

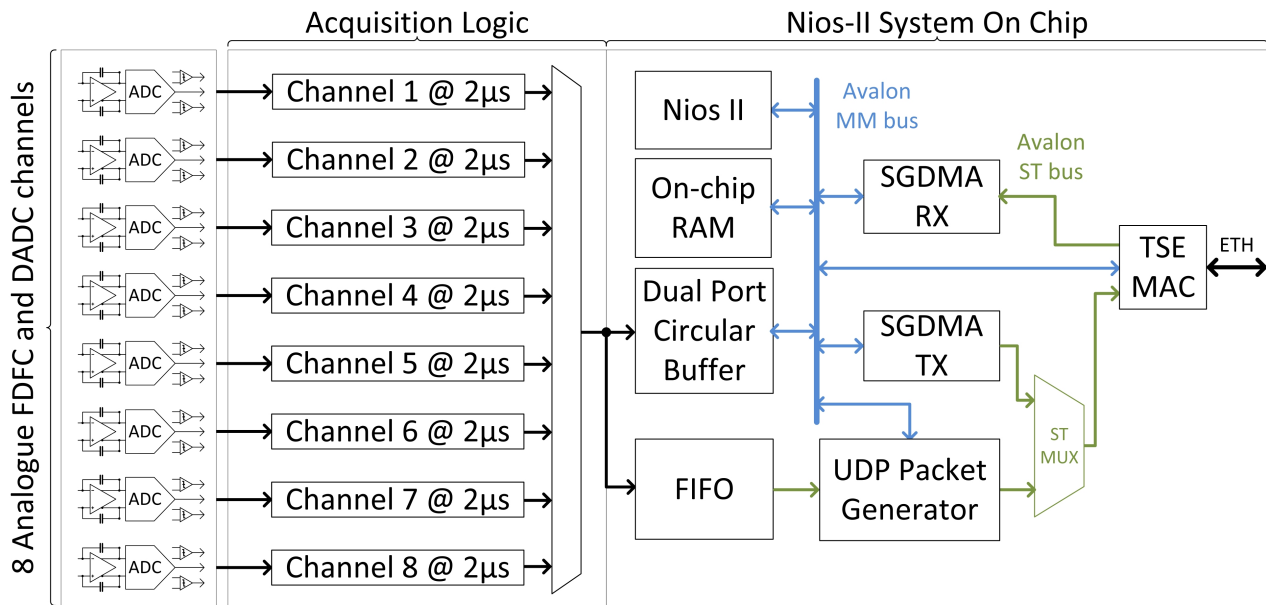


Figure 2: Architecture of the standalone BLEDP firmware with the Gigabit Ethernet link.

avoid the IP fragmentation, which may affect the performance of the transmission. The data bundles of the BLEDP can be encapsulated either in TCP/IP or in UDP/IP frames. The speed of the TCP protocol implemented in the FPGA is limited therefore it is possible to transmit the data stream from only one channel. The UDP protocol implementation achieves higher bandwidth and therefore allows multi-channel data transmission at 128 Mbit/s.

BLEDP FIRMWARE

The FPGA firmware realised for the BLEDP module is comprised by two distinct parts. One part is the acquisition logic and the second is the System-On-Chip (SOC) components generated with the assistance of the Altera development tools. A block diagram of the BLEDP firmware is shown in Fig. 2.

Acquisition Logic

The acquisition logic consists of eight instances of the data combination entities for each input channel. The instance of each channel is continuously sampling the input current with $2 \mu\text{s}$ intervals and therefore it must include a FIFO queue which buffers the data. Writing to that FIFO can be disabled and by this the channels can be selected individually for the transmission. The sampling is realised by use of either FDFC or DADC method depending on the input current of the given channel.

A multiplexer is used to interleave the data read from the FIFOs from all the channel instances and to write it to a dual-port circular buffer mapped into the memory space of the Nios-II processor. The second port of the circular buffer is equipped with the Avalon Memory-Mapped (MM) bus and it can be accessed by the software running on the

CPU. The same data is written in parallel to a FIFO queue included into the SOC.

System-On-Chip with Custom Components

The Nios-II SOC was created using standard IP cores as well as custom components imported into the Altera's system integration tool, i.e. QSYS. The processor is using only internal on-chip RAM memory for the program and data. This becomes possible due to the small memory footprint of the used software components and the large memory resources of the employed FPGA device. The fastest version of the Nios-II core was selected to achieve the highest possible performance of the TCP/IP software stack. The SOC is clocked using a 100 MHz internally generated clock.

The Gigabit Ethernet functionality is realised by a Triple Speed Ethernet (TSE) IP core provided by Altera Corp. The TSE requires the usage of two Scatter-Gather Direct Memory Access (SGDMA) IP cores and is configured to include the 1000BASE-X/SGMII Physical Coding Sub-layer (PCS) function with an embedded Physical Medium Attachment (PMA).

Every time the complete BLEDP bundle is written to the circular buffer, the interrupt for the Nios-II is generated. The custom server, as soon as it receives the interrupt, is transmitting the contents of the buffer. The data is read from the buffer by the SGDMA TX and it is passed to the TSE via the Avalon Streaming (ST) bus. The received packets are passed from the TSE to the Nios-II memory by the SGDMA RX. The server is also responsible for detecting faulty situations, e.g. data overwritten by the acquisition logic before being transmitted to the client.

The SOC contains a fully hardware UDP data path alternative to the standard TSE configuration. The concept of the system architecture was inspired by the Altera's Nios-II

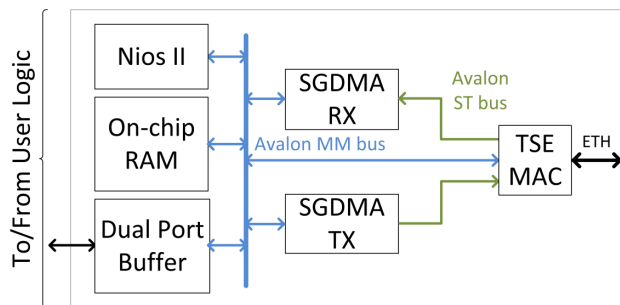


Figure 3: Standard configuration using only Altera's IP cores and software based TCP/IP stack.

UDP offload example available with the source codes under [4]. The UDP data path is one directional (output only) and it is built of custom components i.e. a UDP packet generator and a multiplexer of the Avalon ST bus used to switch sources, which are streaming data packets to the TSE. The UDP packet generator is retrieving the BLEDP bundles written to the FIFO queue included in the SOC and it is inserting the UDP header as configured by the software process. The configuration of the components in the SOC is realised by accessing their registers via the Avalon MM bus. The UDP packet generator requires that all the UDP header information (source and destination ports and IP addresses) is set before the transmission is enabled.

Server Process

The embedded software includes mainly ready made components provided by Altera Corp. and other vendors i.e. the μ C/OS-II real-time kernel ported for the Nios-II, the device drivers and the NicheStack TCP/IP stack. The only custom software implemented for the BLEDP module is the server application. It was written with the C language as a single-threaded process.

The server creates a standard socket provided by the software TCP/IP stack. That socket listens for the incoming connections and it is used to receive commands and transmit acquisition data from a single channel. The TCP protocol was selected for that purpose due to its inherited reliability. The software based TCP path is bi-directional. Only one client is allowed to be connected at a time and any additional connection requests from more clients are refused. The server is parsing incoming commands and it disables or enables the requested channels of the BLEDP module. The Interrupt Service Routine (ISR) was implemented in the server. The server accesses the content of the circular buffer and it transmits the BLEDP bundles written by the acquisition logic. In the case of not sufficient network throughput, it is possible that the buffer will become full with data that were not possible to be transmitted. In that case, the acquisition channel is stopped until the data are transmitted. At the same time, the client receives an alarm.

The software server is also responsible for setting up and enabling the fully hardware UDP data path which is used only for one direction transmission. The parameters which

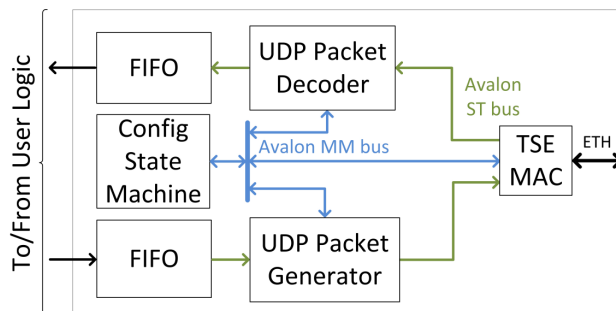


Figure 4: Fully hardware configuration using Altera's and custom IP cores.

are required by the UDP packet generator are the source and destination port and the source and destination IP address. The data count can be also set or the generator can be enabled in a continuous mode. The destination port is requested by the client by transmitting an appropriate command via the TCP path. All the parameters are written to the generator's registers by its device driver which can access it via the Avalon MM bus. After the UDP path is enabled it is streaming the acquisition data from all the channels available in the FIFO queue without any interventions from the software. The server can stop the UDP data transmission before it reaches the required data count or when it was enabled in the continuous mode.

OTHER VARIANTS OF THE ETHERNET FIRMWARE

A generic board with an Altera FPGA is currently under development in the Beam Instrumentation (BI) group at CERN. The Ethernet firmware can be reused in the other projects realised by the BI. For this reason other variants of the Ethernet implementation were proposed.

Software Based TCP and UDP Protocols

Standard configuration, shown in Fig. 3, using only standard Altera IP cores and software components (Nios-II, TSE MAC, SGDMAs, μ C/OS-II real-time kernel, device drivers, NicheStack TCP/IP stack). It allows implementing TCP and UDP based communication but its speed is limited to around 20-30 Mbit/s by the software running on the Nios-II processor. Since it is based on available components it can be implemented in short time periods.

Fully Hardware UDP Protocol

Fully Hardware UDP communication shown in Fig. 4. It does not require use of the Nios-II or any software, but it requires additional custom components. The UDP packet generator and decoder components for inserting and removing UDP headers with appropriate UDP ports and IP addresses. The other custom component should be implemented which will be responsible for the configuration of the UDP custom components and the TSE MAC IP core. The configurator component is replacing the role of the

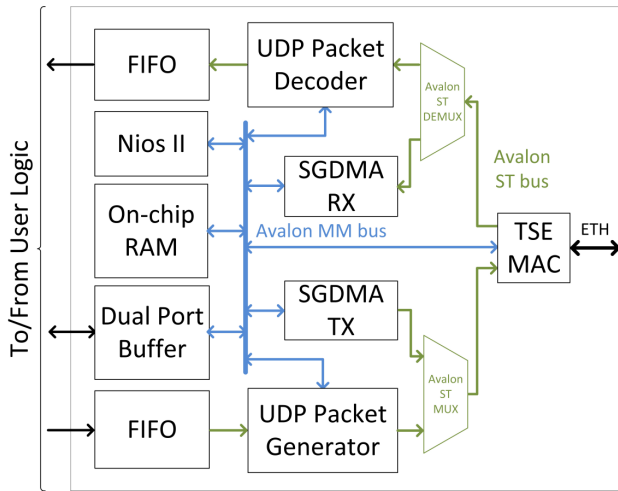


Figure 5: Hybrid containing both the software based TCP/IP and fully hardware UDP/IP data paths.

software running on the Nios-II processor. The configuration is done via the Avalon MM bus of the components. The advantage of the fully hardware UDP path is the transmission speed which can reach the 1Gbit/s level. The implementation time is longer due to the custom components which need to be implemented.

Mixed Software and Hardware Based Solutions

The hybrid containing both the software based TCP/UDP and the hardware based UDP data path shown in Fig. 5. In this solution two additional custom component should be implemented i.e. a multiplexer and a demultiplexer of the Avalon ST bus. These switches are responsible for redirecting the data stream from/to appropriate sources and destinations.

In all three cases shown in this section, the data paths can be trimmed to one direction only depending on the needs of a given design, e.g. only output UDP data stream like in the BLEDP firmware.

CONCLUSIONS

The TSE IP core is capable to reach the throughput of the Gigabit Ethernet but the software TCP/IP stack running on the Nios-II processor is limiting it to only 20-30 Mbit/s. For this reason the TCP protocol was used only to transmit acquisition data from a single channel. The multi-channel acquisitions are implemented by use of the fully hardware UDP data path which works in parallel to the bi-directional TCP data path.

The multi-channel acquisition capability of the newly developed card allowed analysis of a real data acquired in the Proton Synchrotron (PS) accelerator at CERN. The hardware was also verified in the laboratory where cross-talk tests were performed by the analysis of acquired data in the Matlab software. The Ethernet firmware shown in this paper was proven to be reliable and therefore it will be also used in the standalone version of the acquisition crate.

REFERENCES

- [1] C. Zamantzas, M. Alsdorf, B. Dehning, S. Jackson, M. Kwiatkowski, W. Vigano, "System Architecture for Measuring and Monitoring Beam Losses in the Injector Complex at CERN", 1st International Beam Instrumentation Conference, IBIC2012, Tsukuba, Japan, Oct 1-4, 2012.
- [2] W. Vigano, M. Alsdorf, B. Dehning, M. Kwiatkowski, G. Venturini, C. Zamantzas, "10 Orders of Magnitude Current Measurement Digitisers for the CERN Beam Loss Systems", Topical Workshop on Electronics for Particle Physics Conference, TWEPP2013, Perugia, Italy, Sept 23-29, 2013.
- [3] M. Kwiatkowski, C. Zamantzas, M. Alsdorf, B. Dehning, W. Vigano, "A Real-Time FPGA Based Algorithm for the Combination of Beam Loss Acquisition Methods Used for Measurement Dynamic Range Expansion", 1st International Beam Instrumentation Conference, IBIC2012, Tsukuba, Japan, Oct 1-4, 2012.
- [4] Altera Wiki, "Nios II UDP Offload Example", http://www.alterawiki.com/wiki/Nios_II_UDP_Offload_Example