FPGA firmware framework for MTCA.4 AMC modules

Lukasz Butkowski, Tomasz Kozak, Bin Yang, DESY, Hamburg, Germany
Paweł Prędki, DMCS, Lodz University of Technology, Lodz, Poland
Radoslaw Rybaniec, ISE, Warsaw University of Technology, Warsaw, Poland

INTRODUCTION

In MTCA.4 architecture all boards communicate with the central processing unit (CPU) over PCI Express (PCIe), send data to each other using Multi-Gigabit Transceivers (MGT), use the same backplane resources and have the same Zone3 IO or FPGA mezzanine card (FMC) connectors. All those interfaces are connected and implemented in Field Programmable Gate Array (FPGA) chips.

The MTCA.4 firmware framework introduces an additional abstraction layer that separates the hardware dependent logic from user application logic. The framework specifies universal interfaces on this layer. This allows the same firmware and software components to be reused, irrespective of the type of the used hardware.

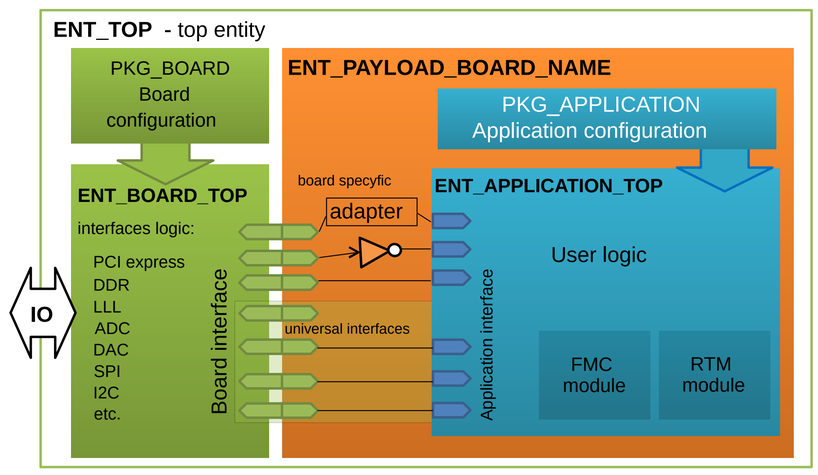
GENERAL VHDL CODE STRUCTURE

The code for the FPGA firmware was divided into three main parts:

Board: hardware dependent and is specific for a given board,has all components for configuration of peripherals on the board, such as clocks, I/O buffers, Analog to Digital Converters (ADCs), Digital to Analog Converters (DACs), provice communication interfaces

Board payload: middle layer section, connecting the board interfaces with the application

Application: section contains user specific algorithms, uses interfaces which are available in hardware



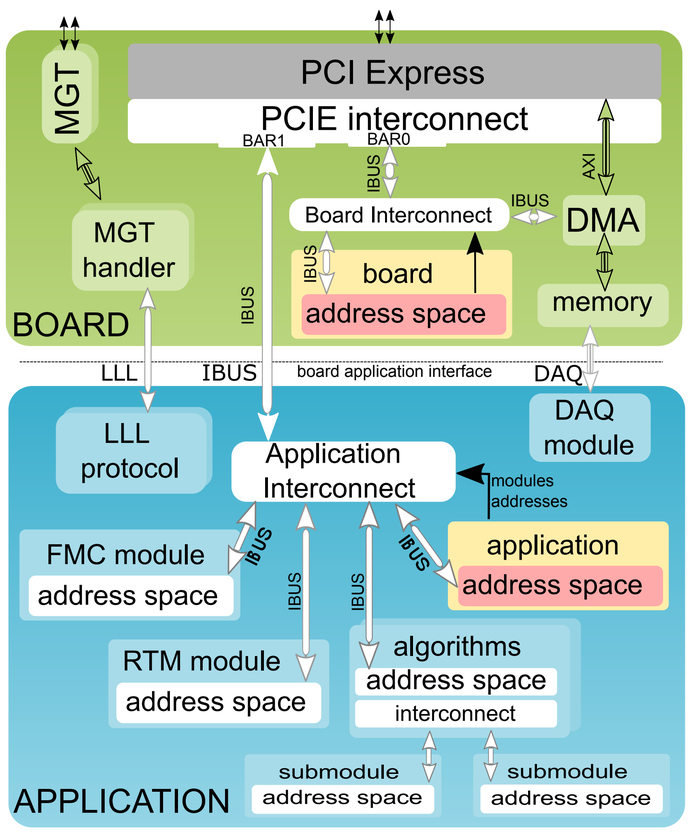
VHDL code block structure.

INTERFACES

Between board and application universal interfaces can be defined. Definitions of universal interfaces are common for all boards.

The include:

- **IBUS** – internal memory access bus, used to access registers and memory areas in FPGA from CPU, always connected to communication interface
- **DAQ** – data acquisition bus, allows to write stream of data to external memory. This data is accessed using Direct Memory Access (DMA) over communication interface
- **LLL** – low latency links bus, middle layer of point to point communication between boards using MGT
- **FMC** – FPGA Mezzanine Card I/O
- **RTM** – Rear Transition Modules I/O
- **AMCIO** – backplane differential signals for clocks and triggers
- **CLK** – clock resources



Block diagram of address space and interfaces connection of standard MTCA.4 firmware

AUTOMATION

Project generation follows the idea that components are independent and have separate source code files list. The creation of projects is automated and done using scripts.

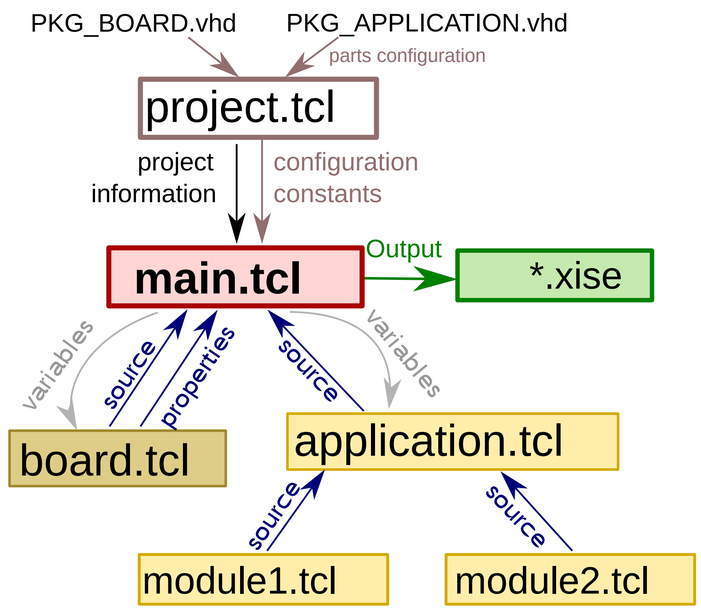
For firmware framework automation Tcl scripting language was chosen. Every part in firmware has it own Tcl script.

main.tcl – main Tcl script shared for all projects, main work is done here

project.tcl – information about board, application and its configuration packages

board.tcl – defines hardware platform such as FPGA chip type and project properties, adds all board support package files to projects

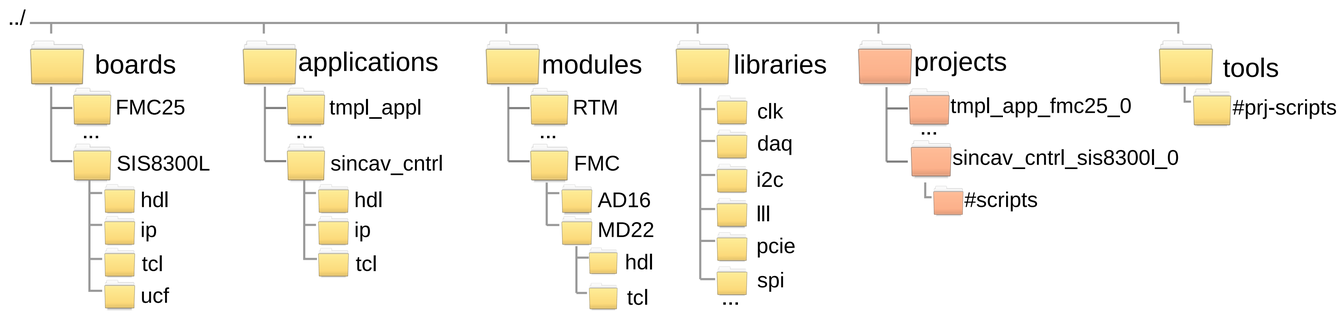
application.tcl, module.tcl, lib.tcl – modules and libraries source files to be added to the project



Project generation diagram

FRAMEWORK STRUCTURE

Firmware files are grouped in folders based on code structure and functionality.



MTCA.4 firmware framework folder structure

USE CASES

Presented MTCA.4 firmware framework is used for control systems at FLASH and European XFEL accelerators at DESY in Hamburg. Mainly used for Low Level Radio Frequency (LLRF) control systems and optical synchronization.

Application	*RTM*	*AMC*
LLRF controller field detection part	DWC10	SIS8300L
Single cavity LLRF controller	DWC8VM1	
Toroid detection application	SIS8900	

Different applications on the same board

CONCLUSION

Proper structure and interfaces were defined as well as automation scripts were written. The MTCA.4 firmware developed using this framework was successfully deployed for XFEL and FLASH accelerators. The framework by its modularity brings significant improvement in terms the development time.

WEPGF074

ICALEPCS 2015
International Conference on Accelerator & Large Experimental Physics Control Systems
17-23 OCTOBER 2015 MCEC MELBOURNE