Redstone Test Center

Test & Training Architecture On A Field Programmable Gate Array (TOAF)

Prepared by Mr. Scott C. Wolfson
November 2015

U.S. Army Test and Evaluation Command

Topics

- Test & Training Enabling Architecture (TENA) Overview
- Field Programmable Gate Array (FPGA) Overview
- Why TOAF?
- TOAF Instantiation Details
- Preliminary Lab Test Results
- Future Work

TENA Overview

What is an Architecture?

- **An architecture is a bridge from requirements to design that defines for each component its:**
  - Purpose
  - Function
  - Interfaces to other components / systems
  - Relationships to other components / systems
  - Guidelines for evolution over time
- **Architectures put constraints on developers. These constraints make possible the achievement of higher level goals.**
  - These higher-level goals are called the system's driving requirements

TENA Overview

TENA Object Models

- **Enable semantic interoperability among range resource applications**
- **Provide the "common language" that all range resource applications use to communicate**
- **Object Model Stages**
  - **User-Defined Objects** – objects defined solely for the purpose of a given logical range by TENA users
  - **TENA Candidate Objects** – objects defined as potential standards, which are undergoing test and evaluation by the community prior to standardization
  - **TENA Standard Objects** – objects developed and supported by the TENA SDA, which have been approved for standardization by the AMT

19

TENA Overview

How do we use TENA on a particular system?

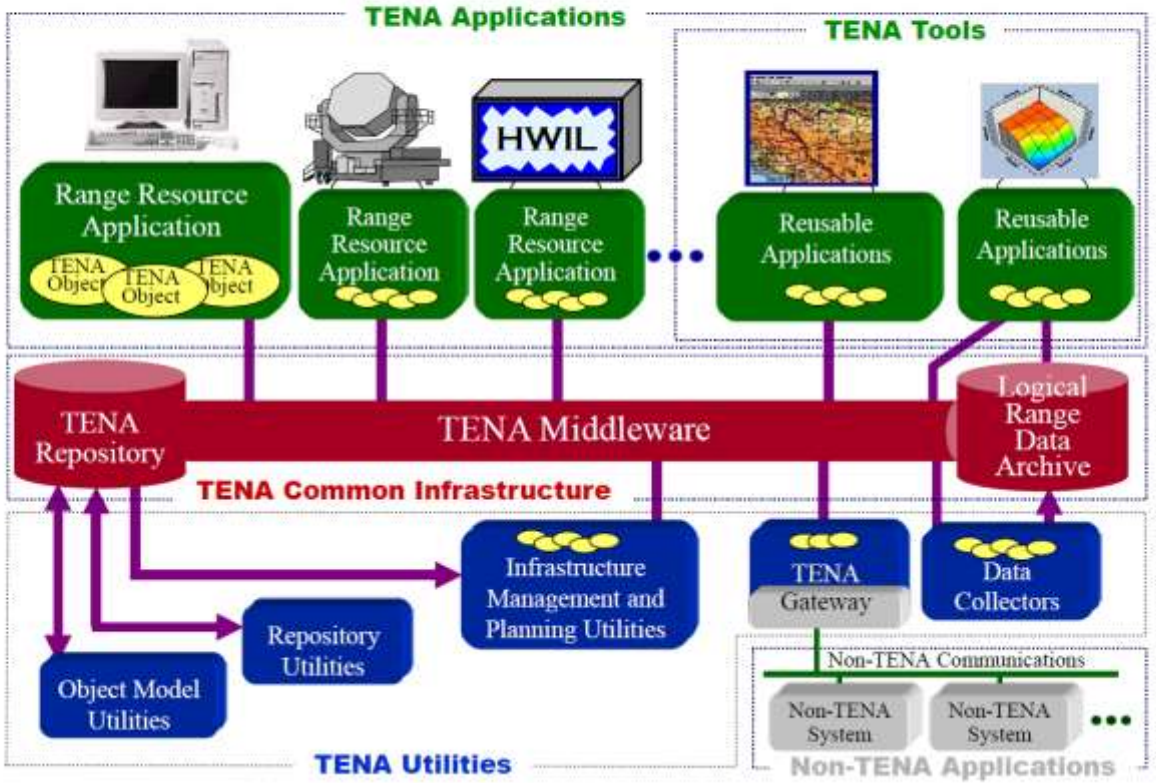
1. **Determine the "in's and out's" of the Particular System**
   - Any system that needs to interoperate with other systems needs to define the data and services shared with these other systems—TENA defines these "ins and outs" as formal data contracts that are easily understood by humans and enforced by computers
   - Determine if existing interfaces (called object models) already exist—TENA Repository has over 1,200 object models that have already been defined by the user community
2. **Auto-Generate Application Source Code**
   - TENA Repository will automatically generate tested and working source code based on the user's particular object models—developers just need to replace the "dummy" behavior for setting/getting attribute values and implementing methods
3. **Integrate Generated Code into Existing System**
   - Working example code simplifies ability to insert the TENA specific code into an existing system, or the example code can be used as the basis for developing a new system
4. **Connect System to Network to begin Collaborating with Others Systems**
   - Publish-Subscribe paradigm makes it easy (no event specific configuration) for multiple participants to share data and services, as well as providing support for redundancy and evolution to new systems

TENA's auto-code generation capability creates tested and proven user specific example applications in minutes!

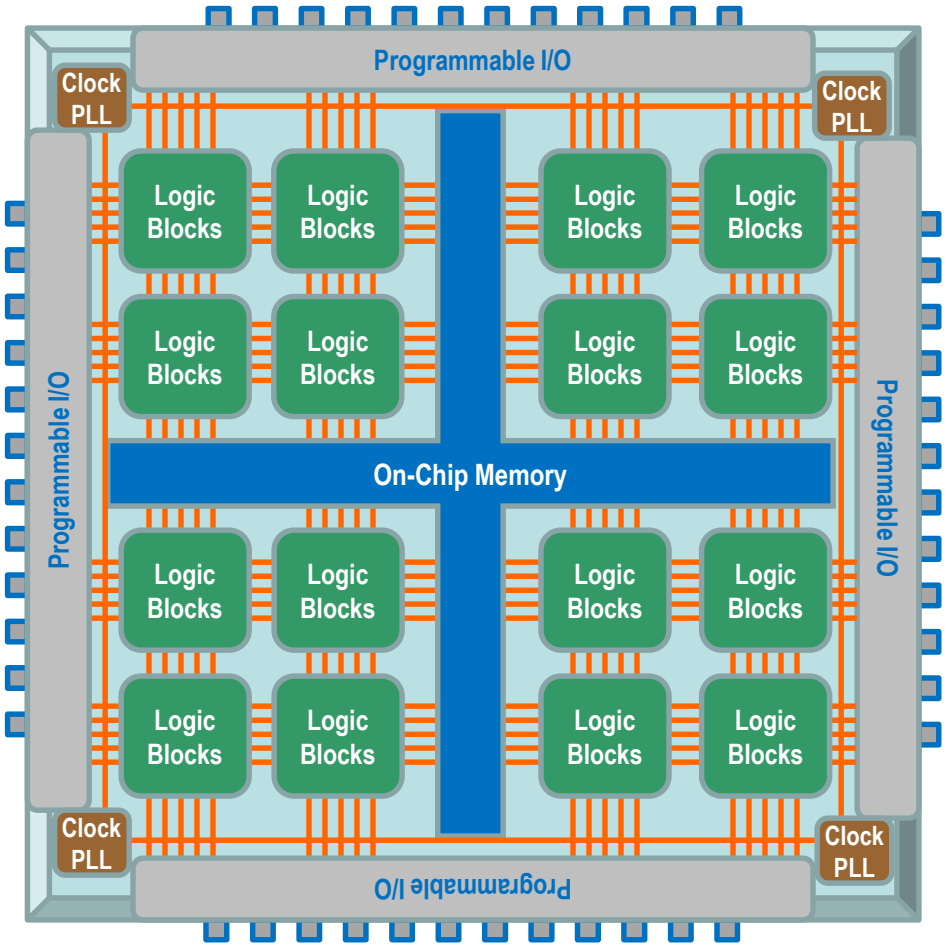
U.S.ARMY

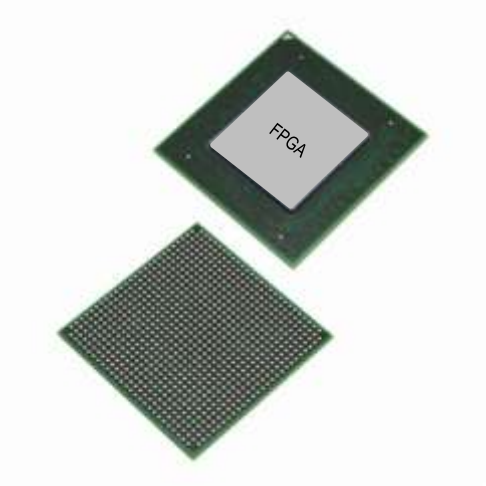
TENA Overview

TENA Architecture Overview



FPGA Overview (Architecture)

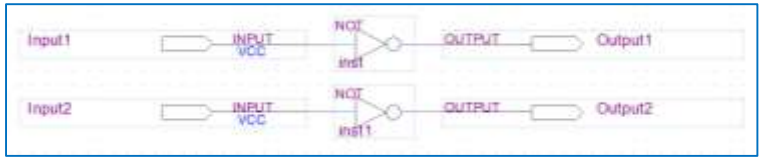




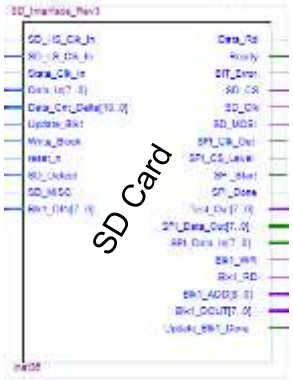
Interconnection scheme, PLL settings, I/O standards, Memory structure and Logic Block functions are defined in the development environment

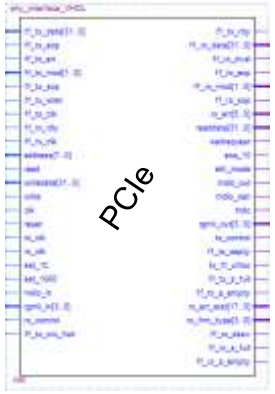
FPGA Overview (Development Environment)

Simple ⟵ ⟶ Complex




```
entity Inverters is
   port (
      Input1   : in std_logic;
      Input2   : in std_logic;
      Output1  : out std_logic;
      Output2  : out std_logic
      );

end entity Inverters;

architecture Behavior of Inverters is

begin

Output1 <= not Input1;
Output2 <= not Input2;

end Behavior;
```





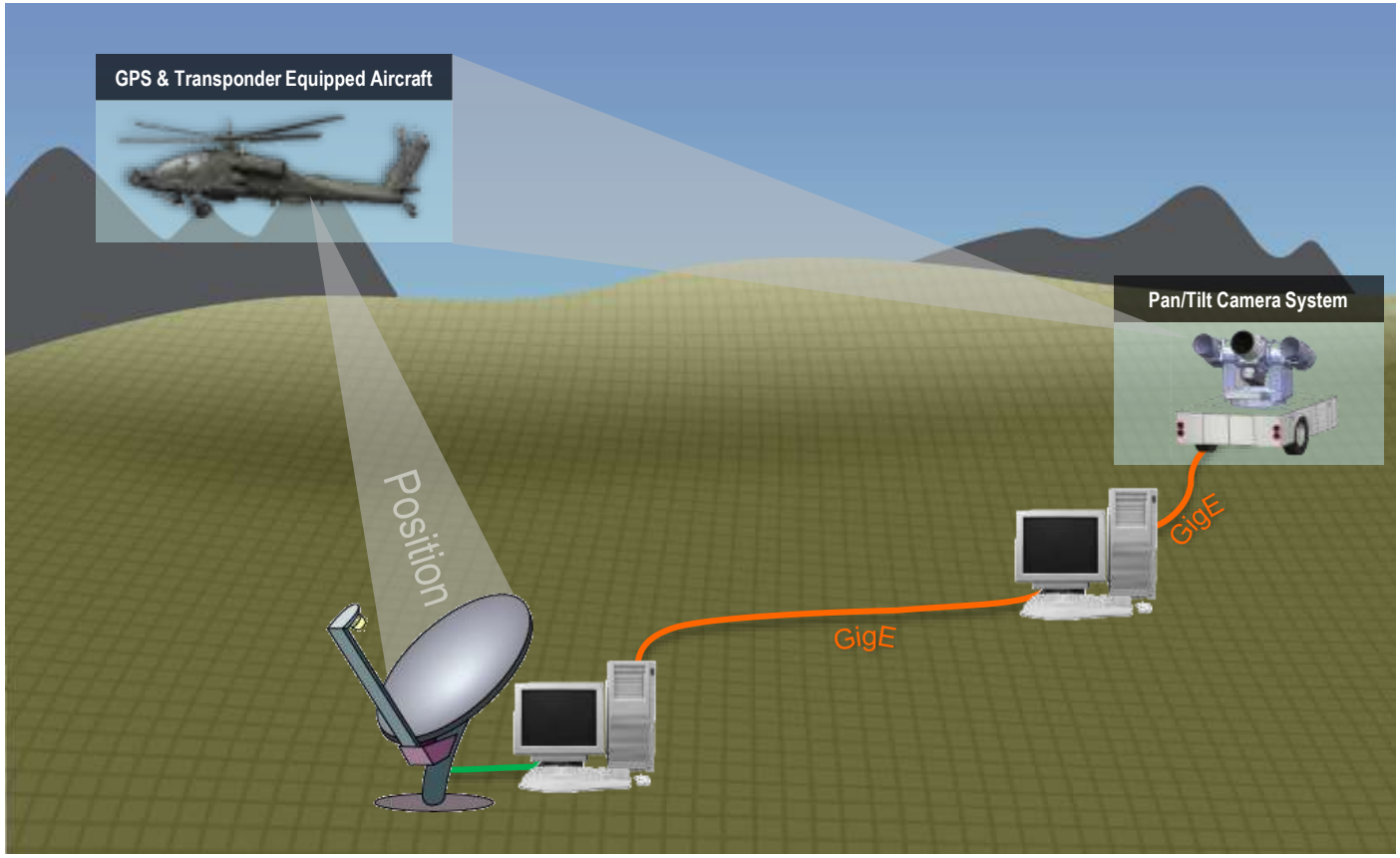
Commercially Available Designs and Interfaces

- Ethernet
- PCIe
- SATA
- Video
- RAM (DDR, etc.)
- Soft processors
- Etc.

- Design graphically or using Hardware Descriptive Language
- Individual Processes execute concurrently versus sequentially

U.S.ARMY

Problem Space (Camera Pointing Example)



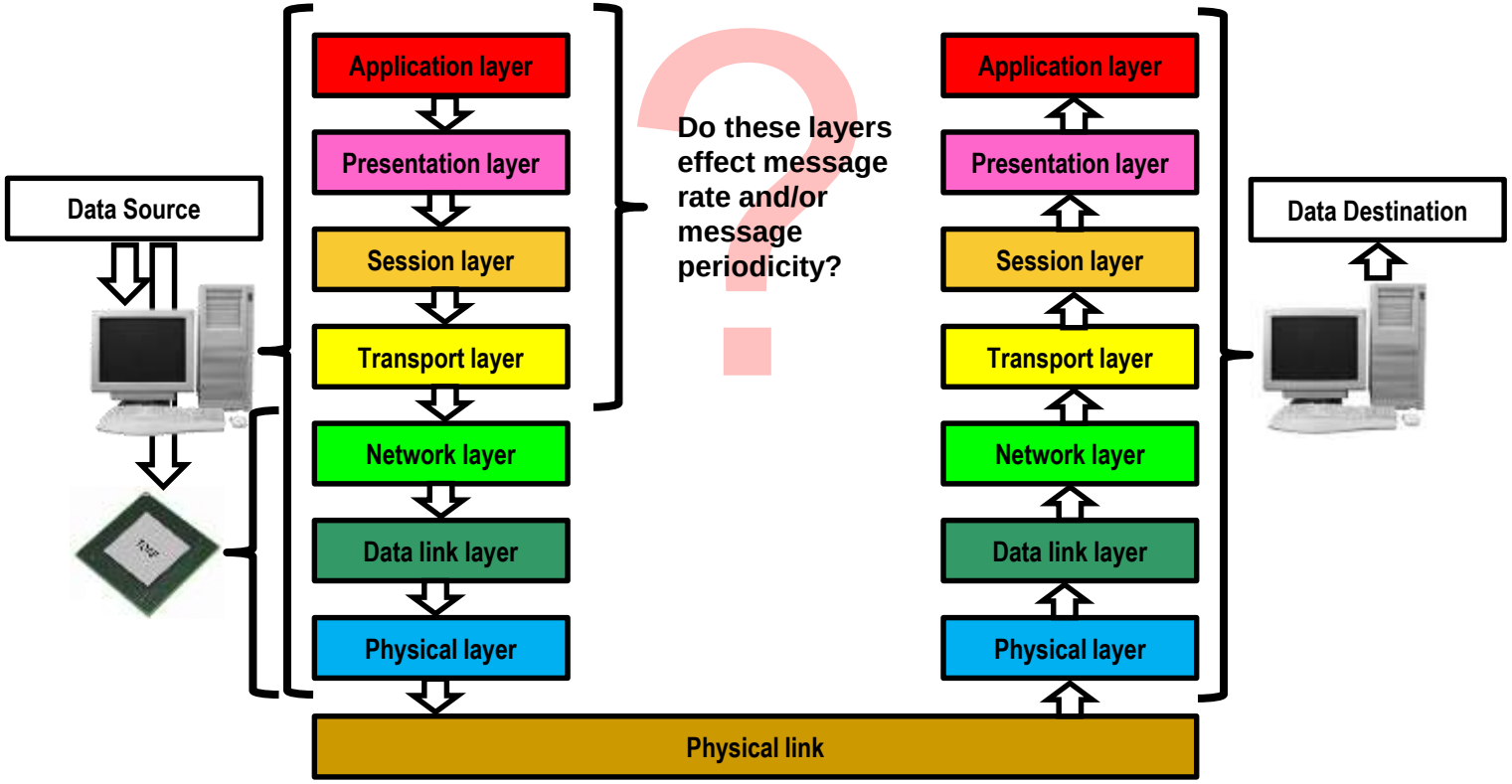
Pan/Tilt Camera System

- Received aircraft position information
- Executes prediction algorithms based on past position/velocity/acceleration information
- Calculates new pan/tilt azimuth/elevation angular velocities
- Transmits information to the pan/tilt system

- **Prediction algorithms are required to account for the system delays and aircraft motion**
- **Position message rate and variations in message periodicity results in prediction algorithm error**

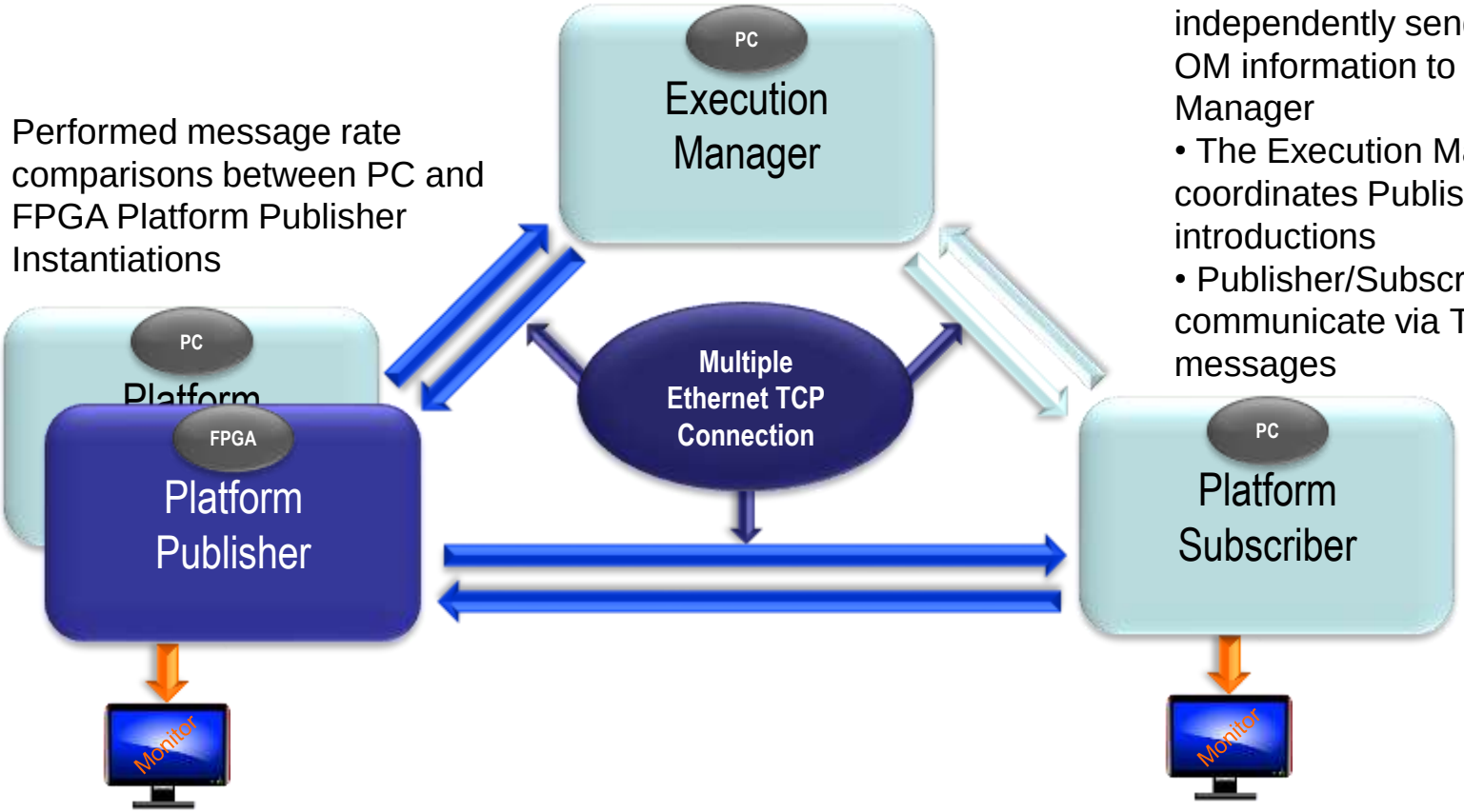
Why TOAF?

- Can TOAF address issues with processor based implementations of TENA?
- How much does sequential code execution and the network stack layer interaction limit message rate and message periodicity?



TOAF Performance Comparison Instantiation Details

Performed message rate comparisons between PC and FPGA Platform Publisher Instantiations



- Publisher and Subscriber independently send IP, MAC and OM information to the Execution Manager
- The Execution Manager coordinates Publisher/Subscriber introductions
- Publisher/Subscriber communicate via TCP or UDP messages

The FPGAs concurrent processing capability should increase the message rate

TOAF Instantiation Details

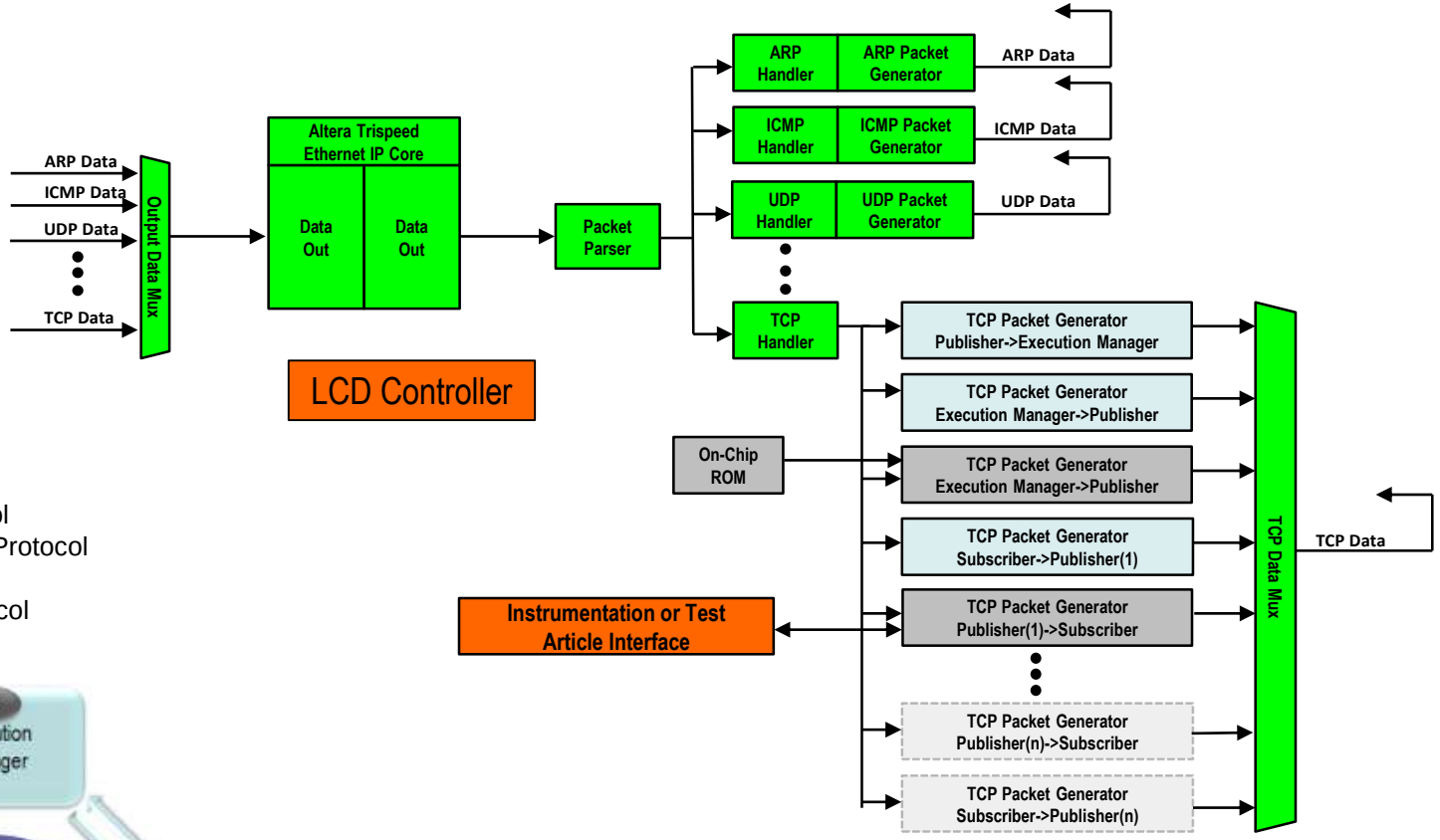
Instrumentation Specific

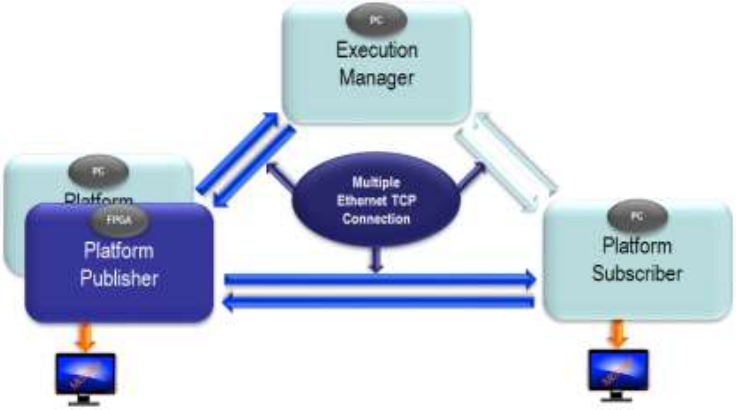
Network Specific

TENA Specific

Unique to TENA OM

ARP – Address Resolution Protocol
ICMP – Internet Control Message Protocol
UDP – User Datagram Protocol
TCP – Transmission Control Protocol





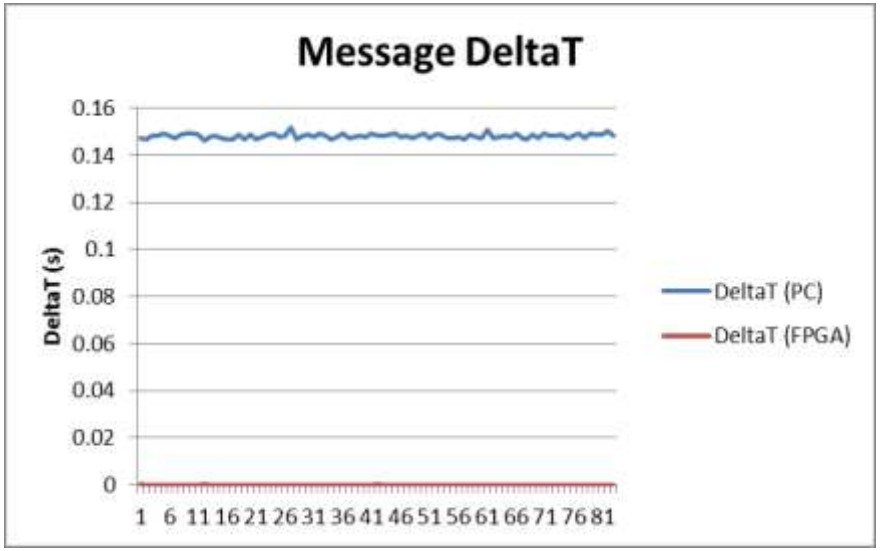
Functional blocks and abstraction layers designed for easy Object Model reconfiguration

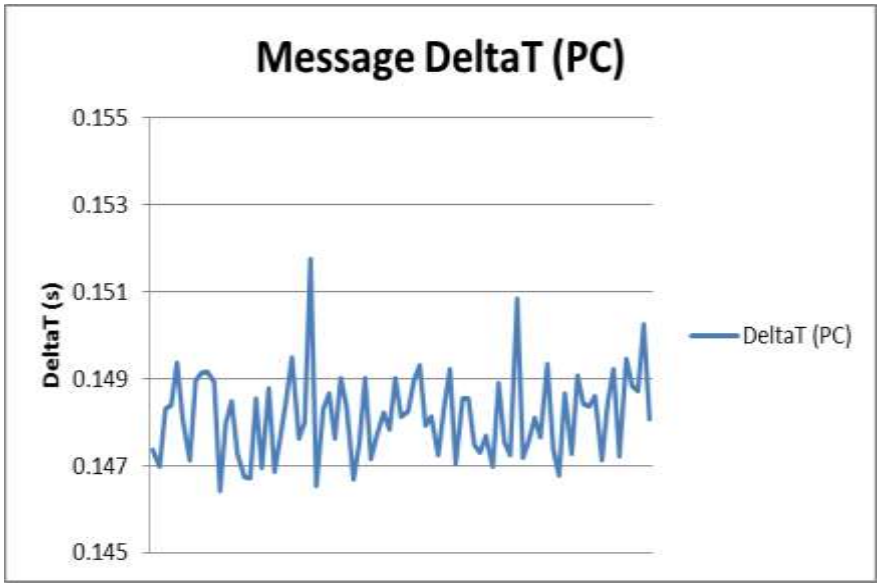
Preliminary Results (Wireshark Comparison)

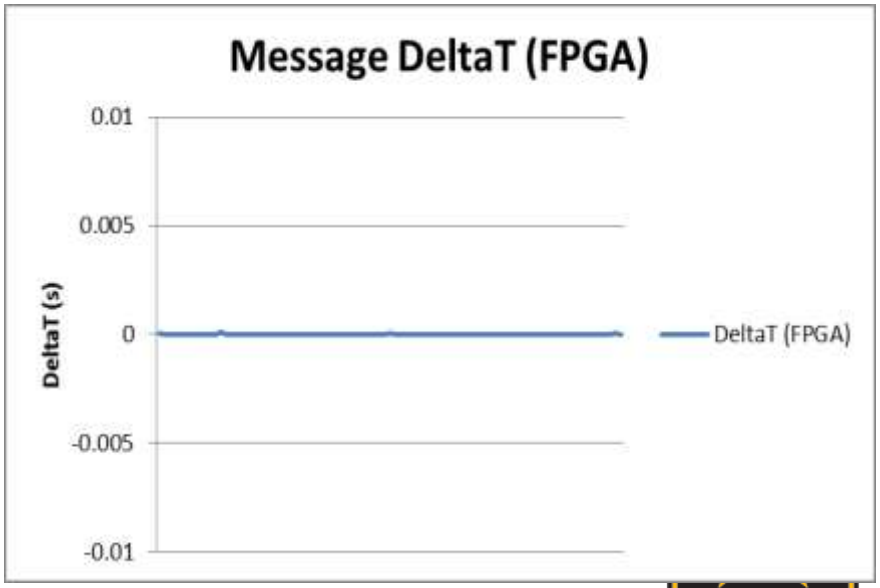
- Significant improvement in message bandwidth (~12000 times faster)
- Significant improvement in message consistence (~75 times)
- Significant reduction in message frequency variations

PC
Mean = 0.148152s
Sdev = 0.000983s

FPGA
Mean = 0.000012s
Sdev = 0.000013s







Future Work

1. Improve from TRL4/5 to a Product
   - Test modular design with multiple TENA object models
   - Perform fault/robustness testing
   - Develop Publisher/Subscriber Intellectual Property (IP) cores
   - Single Publisher -> Multiple Subscribers
2. Beyond GigE
   - 10G to 100G Ethernet IP cores are commercially available