

Altera Nios and TINE Network Queue

Talk Overview

- Altera Nios and Network Queue
 - Introduction
 - Idea and Benefits
 - Design and Implementation
 - Status Report
 - Outlook
 - Use Cases

Altera Nios and TINE Network Queue

Introduction: Hardware fundament

- long term experience in Zeuthen concerning Altera FPGA in custom-built electronic devices (no softcore processor available)
- availability of Altera Nios softcore processor on Altera FPGA
- advantages of FPGA processor design over implementation in pure hardware (e.g. C/C++ code able to run)
- new, advanced Klystron interlock development planned (~ 2002)
- Altera FPGA including Nios processor was chosen (~ 2003)

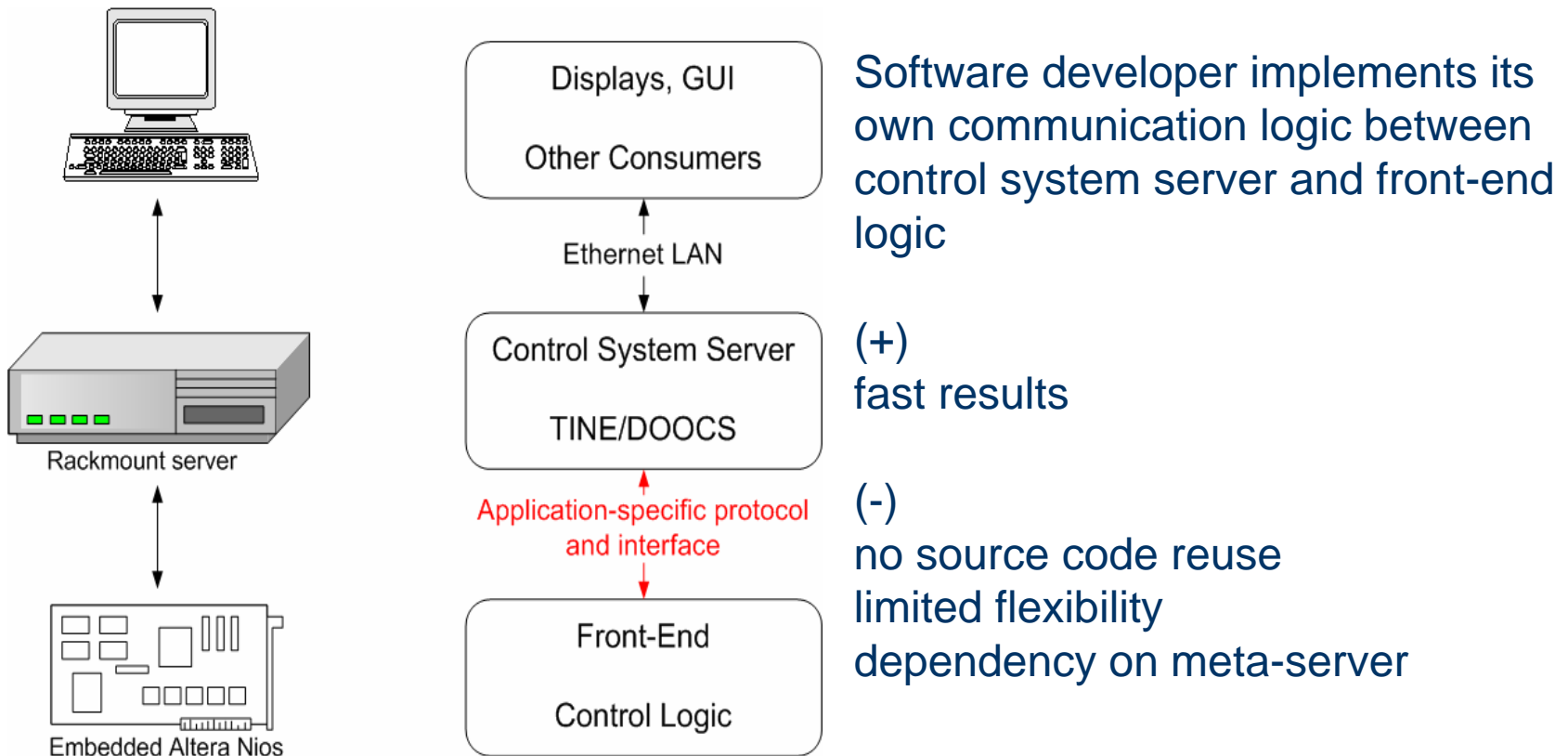
- hardware development using Altera FPGA continuing since 2003
 - updated Klystron Interlock controller boards
 - Unimover device (for heavy mover)
 - Gamma detector control and readout electronics



=> method of communication between Altera front-end logic and control system is required

Altera Nios and TINE Network Queue

Introduction: Straightforward Control System Integration



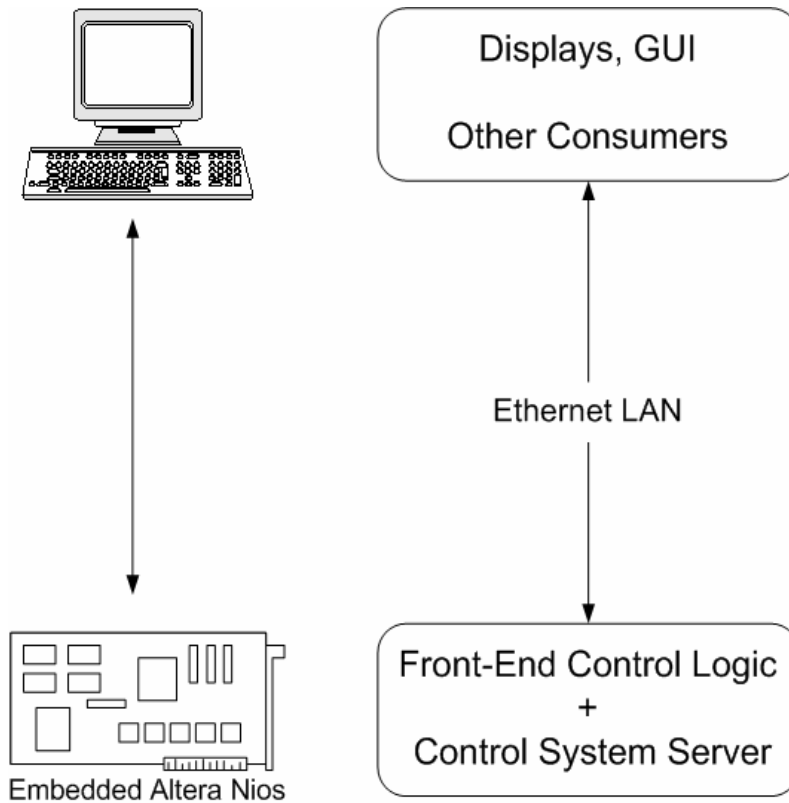
Software developer implements its own communication logic between control system server and front-end logic

(+) fast results

(-) no source code reuse
limited flexibility
dependency on meta-server

Altera Nios and TINE Network Queue

Introduction: Elegant Control System Integration



Port TINE server to Nios platform
 - fundamentals like CPU, C/C++ compiler, C/C++ library, network interface and IDE are available

(+)
 no dependency on meta-server
 direct connection to control system

(-)
 huge effort estimated
 the undiscovered land

Altera Nios and TINE Network Queue

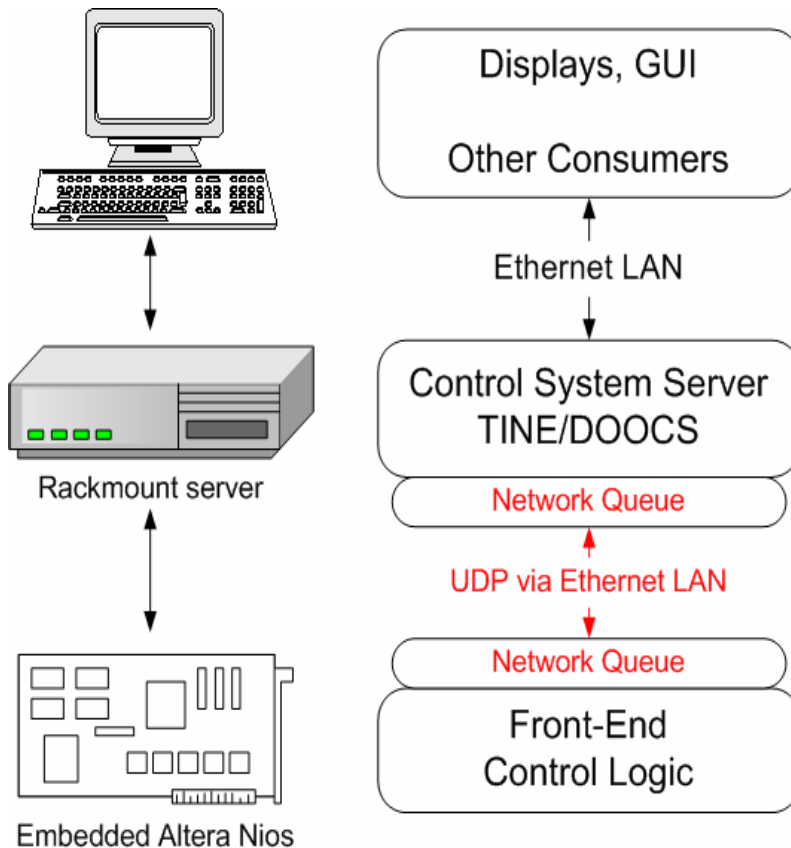
Introduction: Elegant Control System Integration (2)

- development started, some intermediate steps finished
 - special prototype TINE test server is running
- at some point, development was paused
 - very fast evolving hardware, IDE and SDK (no stable platform)
 - big, slow, sometimes buggy IDE for development
 - fragile Nios/Nios2 platform
 - no MMU at Nios2 processor design
 - RISC-based, writing to unaligned memory addresses results in writing to wrong memory location
 - lack of stable, durable TCP/IP stack

=> intermediate step is needed, but straightforward solution is inflexible

Altera Nios and TINE Network Queue

Idea: Network Queue



Limit Demands on Nios Platform

- reduce complexity, code footprint
- being easily able to adopt to new soft- and hardware revisions
- encapsulate internals and provide simple API to exchange data for application developer

(+)

successful implementation can be expected
source code reuse, flexibility
usable for other embedded devices

(-)

still dependent on meta-server architecture

Altera Nios and TINE Network Queue

Benefits

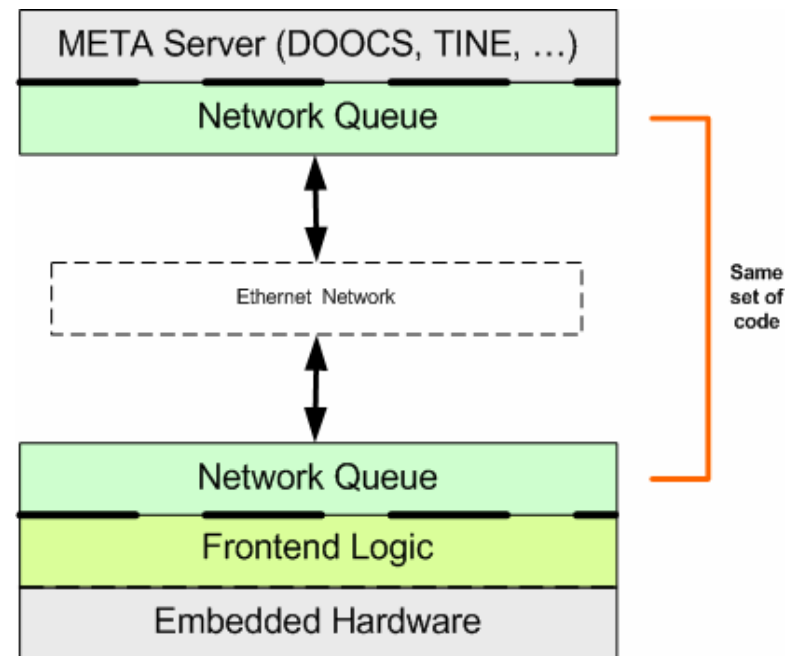
- interconnect networking-capable embedded devices to control system servers
 - on platforms where it is difficult to provide full control system server implementation (lack of resources and/or fundamental APIs)
- encapsulation of general aspects
 - data transport, endian changing, marshalling
- chance to react on/adopt to evolving/new platforms with little extra work
 - wrapped platform-dependent sourcecode

Altera Nios and TINE Network Queue

Design and Implementation (1)

- inspired by SNMP
- Async Telegrams, Requests, Responses, Remote Procedure Call (RPC)
- support of basic data types (Integer 8,16,32 bit, Float, Double, Binary, Text)
- reasonable message contents definition
- fully transparent encoding (endian change handled internally)
- Peer-to-Peer (strict 1:1)
- easy API calls for user code
- (almost) Null Queue foreseen

- + seamless integration of 1-dimensional arrays (any supported data type)
- + acknowledge packets (guaranteed delivery to remote peer)
- + changed strict 1:1 to m:n peer to peer structure



Altera Nios and TINE Network Queue

Design and Implementation (2)

- available platforms
 - 32 and 64bit Solaris and Linux (gcc)
 - Win32 (MS Visual C++ v6)
 - Nios2 + ucOS/II + LWIP TCP/IP stack
 - Nios2 + ucOS/II + Interniche Niche TCP/IP stack
 - shared library is provided on Solaris by Bagrat
- statistics and strong, extensive error checking to detect also potential problems
- one set of platform-independent source code files
- wrapping of platform-dependent functionality for easy adoption to new platforms
- UDP single packet handling (to simplify embedded development process)
- code is designed to compile warning and error free

Altera Nios and TINE Network Queue

Status Report

- implementation of enhancements based on user feedback done
- development takes more time than usual due to fragility of Nios2 RTOS platform and especially the behaviour of TCP/IP Stack solutions provided on it
- looking into and fix issues present in prototype implementation
- working in parallel with Marek to integrate API into Gamma Detector software project

Altera Nios and TINE Network Queue

Outlook

- finalize first release and documentation
- add release to TINE software distribution
- continue to implement queue at approved projects
 - Altera-based Gamma Detector ¹
 - Altera-based Klystron Interlock ¹
 - Tektronix scope readout
- work towards use at other meaningful projects
 - apply of Network Queue to Unimover device
 - use as getterpump <-> server communication at XFEL

¹ details will follow in a second ...