New Embedded Digital Front-End for High Resolution PET Scanner

P. Guerra, *Student Member, IEEE*, J. Espinosa, J. E. Ortuño, G. Kontaxakis, *Senior Member, IEEE*, J. J. Vaquero, *Senior Member, IEEE*, M. Desco, and A. Santos, *Senior Member, IEEE*

***Abstract*——This work describes a new digital front-end for a high-resolution low-cost animal PET scanner which is currently under development. The advances in flexibility and size of modern FPGAs together with the release of new tools enable the integration of most of the front-end electronics in a single FPGA. The implemented system includes a small 32-bit RISC processor, several peripherals attached to the internal buses and a special DSP unit closely attached to the processor which is dedicated to the detection of the gamma events. On top of these, a small footprint real time operating system abstracts the underlying hardware, providing the mechanisms to combine on-chip slow control and data streaming.**

***Index Terms*—Digital front-end electronics, high resolution positron emission tomography, real time system architectures.**

I. Introduction

As programmable logic devices continue to grow in density, designers are increasingly using FPGAs where they previously used ASICs. Compared to ASIC, programmable technologies reduce development time and risk while keeping electronics size comparable [1]. Moreover online reconfiguration allows for area reutilization when considering alternative modes of operation.

In the last few years ASIC-based detectors [2] have been replaced by FPGA-based equivalents, and currently the trend is towards the replacement of the scintillation analog pulse processing by its digital counterpart [3]. However most designs still rely on an external constant fraction discriminator (CFD) or similar circuit for the generation of the event time stamp. There are some exceptions to this last statement [4]–[7] that replace the external circuit by additional digital processing.

In this work we describe the digital front-end currently being developed by our group. Our aim is to build a compact, low-cost and flexible detector for small animal imaging. Currently we assume a detector consisting of multi-layer scintillation crystals attached to a position sensitive photomultiplier (PS-PMT) and Anger readout, although the interface will be flexible enough to accommodate other configurations. In order to reduce space and simplify the design, we have integrated the complete system in a single FPGA; in this way the signal processing block (custom DSP) that computes basic parameters of the pulse (energy, position, time stamp and crystal discrimination) can be treated as another peripheral of a more complex system with direct access to the high speed buses embedded in the device. A small footprint real time operating system (RT/OS) runs on top of the HW architecture, simplifying software development.

This communication is structured as follows: in the first section the software tools that have been used are enumerated and an overview of the complete system under design is provided, next section presents the HW/SW architecture of the embedded digital front-end that has been designed; this communication concludes with results regarding area, speed and streaming bandwidth.

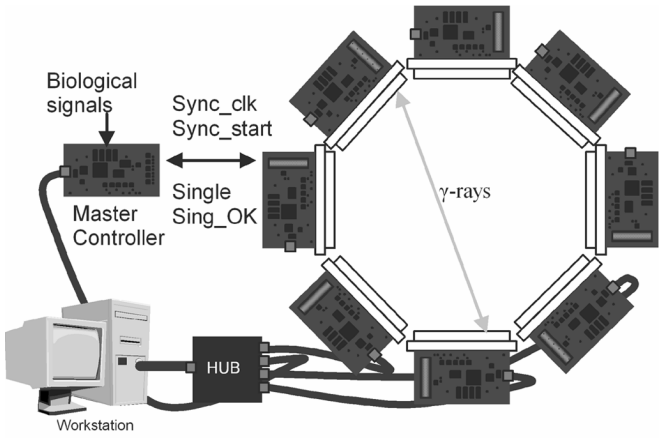

Fig. 1. Tomograph overview: The detectors can be adapted to different geometries. All detectors report to a master controller for first level coincidence resolution and to an external computer for data storage. The master controller integrates other tasks such as time labeling of digitalized biological signals or gantry control.

II. Material and Methods

A. Software Tools

The Embedded Design Kit 6.2 (EDK Xilinx Inc., San José CA, USA) has been used to integrate the complete embedded system-on-chip (SoC), including hardware peripherals (either proprietary or third party cores), RT/OS and different software services.

The implemented digital hardware, as well as the interaction between the developed cores and the low level libraries, has been thoroughly simulated with Modelsim SE (Mentor Graphics, Wilsonville OR, USA).

Manuscript received June 17, 2005; revised March 22, 2006.

This work was supported in part by the FPU Research Grant from the Spanish Education and Science Ministry, by the Spanish Thematic Network IM3 (PI052204) and project TEC2004-07052-C02-02.

P. Guerra, J. Espinosa, J. E. Ortuño, G. Kontaxakis and A. Santos are with the Universidad Politécnica de Madrid, ETS. Telecomunicación, Ciudad Universitaria s/n, E-28040 Spain (e-mail: pguerra@die.upm.es).

J. J. Vaquero and M. Desco are with the Unidad de Medicina y Cirugía Experimental, Hospital G. Gregorio Marañón, c/Dr. Esquerdo, 46. E-28007 Madrid, Spain.

Digital Object Identifier 10.1109/TNS.2006.874800

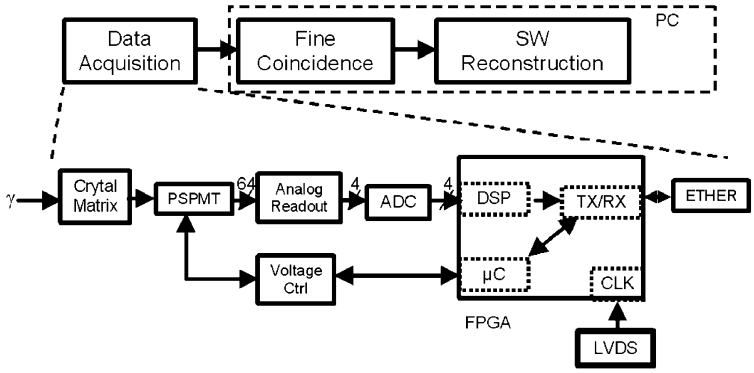

Fig. 2. High Level architecture of the proposed system, and detailed view of the acquisition module.

The DSP core, described in VHDL (Very High Speed Integrated Circuit Hardware Description Language), has been optimized and verified through cosimulation with Modelsim and Simulink 5.0 (The Mathworks, Natick, MA, USA) using the software package *XtremeDSP®* from Xilinx. Simulink has been used to provide realistic input stimuli to the VHDL simulator through the modeling of the analog elements of the front-end (crystal layers, PS-PMT, analog electronics and ADCs)[8].

The real time kernel μC/OS-II (Micrium, Weston FL,USA), a highly portable preemptive real-time multitasking kernel, provides all the resources required by our application. This kernel, written in ANSI C, is suitable for safety critical systems common to aviation and medical products and has been certified in an avionics product by the Federal Aviation Administration (FAA) for use in commercial aircrafts. Moreover μC/OS-II has been ported to the microprocessors currently supported by FPGA vendors (PowerPC, ARM, NiosII and Microblaze), providing an additional level of independence from the final target technology.

The front-end device is interfaced with internal tools developed with the .NET platform (Microsoft Corp., Redmond CA, USA).

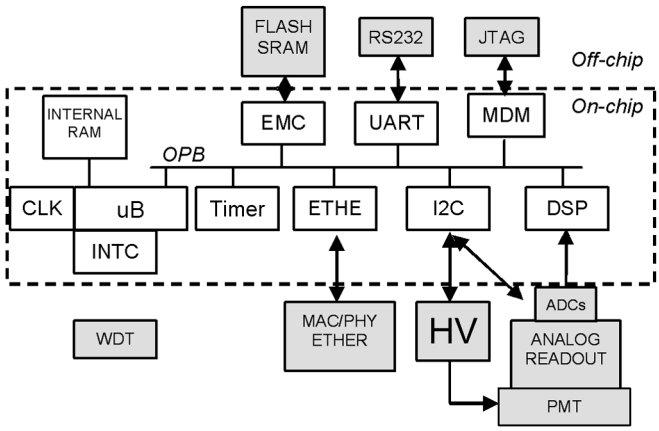

Fig. 3. Block diagram of the acquisition front-end architecture. The FPGA integrates on-chip a microprocessor, a custom DSP for pulse detection and all the required controllers for interface with the outer components and OS support.

B. System Overview

The embedded front-end described in this work will be the main building block of the system, which will consist of:

- A master controller, which will distribute the synchronization signals. These signals are required in order to guarantee that all modules share the same clock and the same value in the internal counters referring to time.
- An even number of acquisition modules, consisting of the scintillation crystals, PS-PMT, readout and acquisition +control, being the latter the object of this work.
- One or several concentrators, depending on the actual number of acquisition modules, which send data to a workstation for off-line fine coincidence resolution and image reconstruction.

The master controller will also include:

- An acquisition interface for the registration of biological signals, such as the cardiac or respiratory cycle. Synchronization of external activity with the internal timestamps is important to ensure the validity of the acquired PET images and helps to improve the reproducibility of PET investigations [9].
- A SW configurable coincidence detector. On every event detection the modules report the single to this unit, which discriminates for coincidences in a clock-cycle wide time window, producing a result before the pulse is completely characterized.
- Gantry control tasks, like controlling the rotating motor or stepping the bed.

The acquisition module includes, as it is shown in Fig. 2, the following blocks:

- A custom DSP core processes the Anger signals generated by the analog readout, which are sampled at a maximum sampling rate of 65 MHz by an ADS5122[10], an 8-channel 10-bit ADC from Texas Instruments (Dallas, TX, USA). When a pulse is detected a programmable number of samples is extracted from the input stream to compute the basic parameters of the scintillation pulse, producing a data packet. Each module will support a maximum count rate of 2 Mcps.
- An Ethernet controller that sends the acquired data to the host computer. In order to simplify HW/SW development, the external Ethernet IC includes the physical (PHY) and medium access (MAC) layers [11].

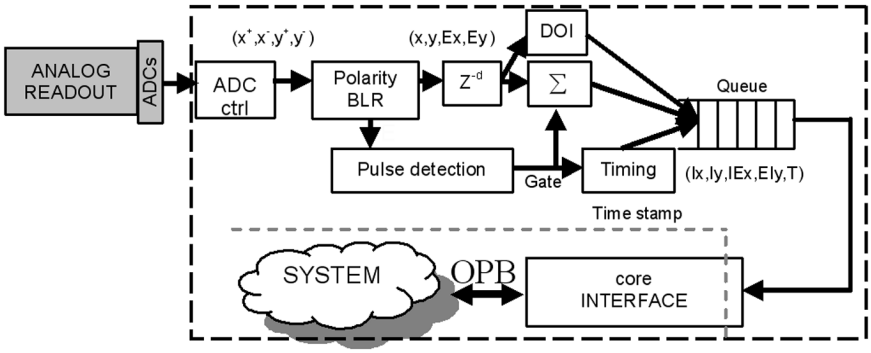

Fig. 4. Detailed block diagram of the DSP block. The pulse detection state machine triggers the acquisition of the pulse. The results are stored in a queue and, through the core interface, read by the embedded processor which immediately sends them to the external computer.

- A microcontroller that handles the communication through the Ethernet as well as slow control.
- Clock management. For reliable coincidence detection, accurate timestamps are needed. For nanosecond accuracy we must be able to synchronize all modules within 1 ns [7].
- This is achieved by a high-speed differential clock distribution and a SW correction after calibration. The master controller distributes a high precision 25 MHz LVDS clock, which is used by each module to generate a synchronized 62.5 MHz clock, making use of the low jitter internal PLLs available in current FPGAs.

In order to reduce space and increase flexibility most subsystem components are integrated in a single FPGA. A very preliminary prototype of the digital part of the acquisition module has been assembled based on development kits from different vendors. This prototype enabled us to test the concept, debug the SW and validate the HW/SW integration.

III. Proposed Architecture for the Embedded System

A. Hardware Overview

The on-chip system designed with the EDK is bus-centric, in the sense that relation between cores is defined by their attachment to system buses. Our design considers 3 internal buses: data and instructions in the internal memory and one for the on-chip peripherals (OPB).

The implemented system, shown in Fig. 3, includes the following cores:

- Application specific DSP block (DSP) and clock management (CLK).
- The Microblaze (uB), a 32-bit RISC processor from Xilinx with Harvard architecture that requires around 1000 logic cells.
- A debug module(MDM) for on-line debug through the JTAG.
- Memory controllers (EMC) for external 2 MB SRAM and 4 MB FLASH and also for the internal memory of 16 KB Serial port (UART) and I2C controller.
- Interrupt controller (INTC).
- Custom made Ethernet controller (ETHE).
- 2 Timers, one of them is required by the RT/OS for context switching.
- General purpose IO, including interface to an external watchdog timer (WDT).

B. DSP Core

This block detects the scintillation pulses generated by the interaction of the gamma-rays with the crystal. The detection and acquisition process is highly pipelined, in such a way that up to a maximum of four consecutive pulses may coexist in the core, each of them in a different phase of processing.

As the block diagram of Fig. 4 shows, the data acquired by the ADC is handled by its controller which corrects the baseline (BLR) and normalizes the pulse, so that the processed pulse has positive amplitude and zero baseline. When the instantaneous energy crosses the programmed threshold, a finite state machine (FSM) is triggered, enabling the integration of the pulse within a certain time window, as well as computing a timestamp for the pulse and a measure of the decay time, which will be used for depth-of-interaction correction in a phoswich system[12].

As a result of the detection a data packet of 15 bytes is generated and stored in a queue waiting to be transmitted. These packets are then transferred into an internal buffer of the IP interface and once enough data has been stored, the core raises an interrupt that invokes the corresponding interrupt handler and the actual transmission to the host computer through the Ethernet interface. The IP interface also includes 8 addressable registers, which are required to configure the acquisition parameters or read the core status.

As it has been earlier mentioned, the acquisition and processing is driven by a 62.5 MHz clock which is synchronized to the external reference. However in order to guarantee the functionality of the system even when the PLL does not succeed in locking to the reference, the microprocessor and cores are driven by a local 50 MHz clock. At the DSP IP interface, where both clock domains meet, data is asynchronously exchanged.

C. Software Services

Following Xilinx's methodology, we have created device drivers for every custom core, so that they automatically integrate with the design tool.

In order to improve design flexibility and reusability we have decided to rely on the synchronization and communication services provided by an RT/OS, in particular the μC/OS-II was selected, which was extended with a new timer service. Addition-

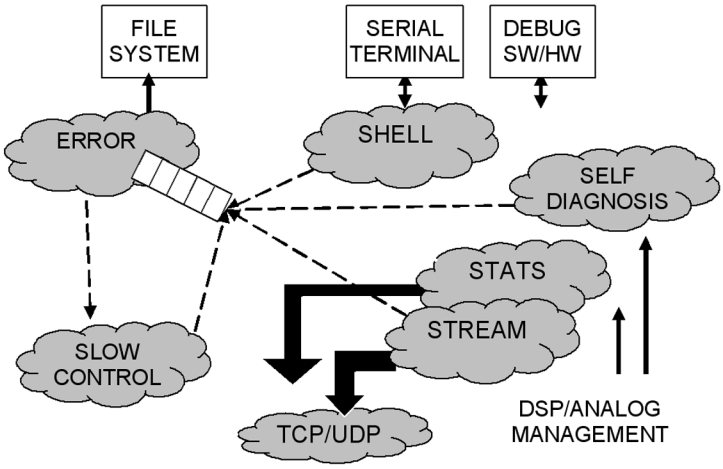

Fig. 5. Application tasks diagram. These tasks are scheduled by the RT/OS according to their respective priority. The OS provides all mechanisms (semaphores, queues, signals) for task communication and synchronization.

ally, during the system boot a filesystem based on Xilinx's libraries is mounted and the LWIP [13] is initialized. This LWIP is a TCP/IP protocol stack optimized for embedded processors and provides a useful interface for efficient data streaming.

D. Application Tasks

The embedded application code is divided into a set of concurrent tasks which are scheduled by the μC/OS-II based on their priority. These tasks, as it is summarized in Fig. 5, are the following:

- Slow Control Task, which is responsible for all steps previous to acquisition, such as calibration or configuration as well as executing control commands generated by the master controller or the host computer.
- Acquisition and streaming task, which is waken up by the DSP interrupt handler and performs the actual data transmission to the computer through a UDP socket.
- DSP statistics task, which is periodically waken up by an OS timer to report the number of detected singles, lost events or transmitted singles together with timer values. This data is useful for a-posteriori sinogram corrections
- Error management,which manages error signals given by tasks creating the corresponding report files and takes the appropriate recovery steps.
- LWIP engine, which keeps the TCP/IP stack alive.
- Self-diagnosis, which performs periodic system self test when the processor is idle.
- Shell,which through the serial port provides a Unix-like command interface that allows the user to log into system given access to the status, report files the system, etc.

E. Data Streaming

The main challenge of the implemented design comes from the fact that we are using the same processor for streaming and slow control. This requires high availability of the processor for streaming and high responsiveness for slow control. As previously stated, the approach has been threefold:

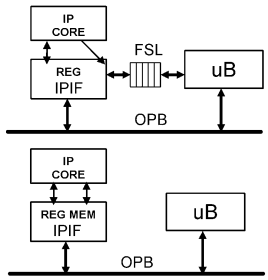

Fig. 6. Two different core interfaces are considered. In one case (top figure) the streaming data is extracted through a FSL channel between the core and the processor, in the other case (bottom figure) the core integrates memory accessible from both sides, by the core and the processor through the OPB bus. In any case the interface includes configuration registers which are accessible through the OPB.

1. Rely on RT/OS priorities to guarantee that streaming tasks do not block control tasks.
2. Implement an efficient interrupt and transmission mechanism to reduce the processor effort.
3. Select an external Ethernet controller with an efficient interface.

Two alternative DSP cores, with different interfaces as shown in Fig. 6, have been designed. The first uses the Fast Simplex Link (FSL) for data streaming while the second uses only the OPB.

The FSL is a unidirectional point-to-point communication channel bus used to perform fast communication between any two design elements on the FPGA when implementing an interface to the FSL bus. The uB accesses the FSL registers as they were part of the register file, making the data transfer between the DSP core and the processor very efficient. However this solution is not that optimum if we consider that this data needs no further processing and the uB will just move it from the register file to a memory buffer, i.e., the processor is being loaded with operations of no added value.

The second approach provides the core with an additional dual-port memory, internal to the FPGA, which is mapped into the processor's address space. This means that the DSP writes the acquired data directly into addressable memory, and no

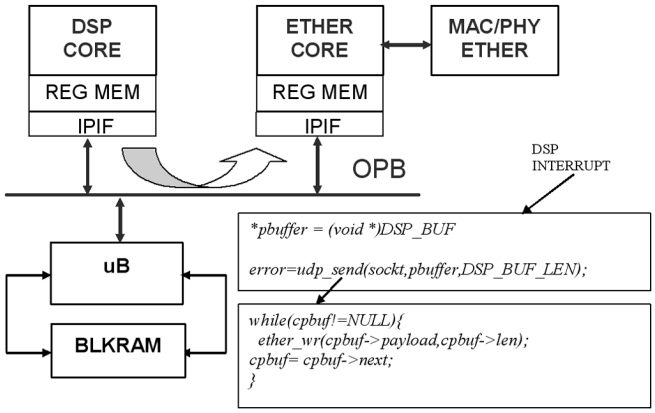

Fig. 7. LWIP zero-buffer mechanism avoids wasting time moving data between memory areas. Low level drivers and interrupt handlers are placed in the device's internal memory (BLKRAM). This way, as it is shown in the figure, when the DSP triggers the handler execution, the code is accessed through independent buses freeing the OPB for data streaming.

access to this data is really required until the actual start of the transmission. This statement becomes really true when the TCP/IP stack does not include any kind of internal buffering. Typical TCP/IP implementations, such as BSD, require copying the data to be sent from the application program to internal buffers in the TCP/IP stack. The reason for this copy operation is that the application and the TCP/IP stack usually reside in different protection domains. In most cases the application program is a user process while the TCP/IP stack resides in the operating system kernel. By avoiding this extra copy, the performance of the API can be greatly improved[14]. That is the case of the LWIP API, which utilizes knowledge of the internal structure of the stack to achieve effectiveness [13].

In order to speed up data streaming we have taken the precaution of placing low level drivers into the internal device memories. As it is shown in Fig. 7, such solution provides a significant improvement in throughput by freeing the OPB to the data transfer between DSP and Ethernet cores. While the data streaming takes places, the execution code is accessed through independent buses, avoiding any contention in the OPB between accesses to external memories and to the on-chip peripherals.

Finally, regarding the external Ethernet controller, most Fast Ethernet controllers are designed for the Peripheral Component Interconnect (PCI) bus which is highly inconvenient, as it would require embedding in the FPGA a PCI controller. However there exist a few Ethernet controllers, specially suited for non-PCI embedded applications, which provide a simple 8/16/32 bit interface and integrate both the MAC and PHY layers, relieving the designer of the need for a MAC controller in the FPGA. Among the available devices the DM9000 from Davicom Semiconductors [11] was selected. It is a fully integrated and cost-effective 10/100 M controller with a general processor interface, an EEPROM interface, designed for low power and high performance which, depending on the processor and the actual bit width of the interface, may drive the Ethernet network to its maximum speed of 100 Mbps [15].

IV. Results and Conclusions

An on-chip hardware/software architecture specially tailored to the needs of low-cost and flexible gamma detection in PET with small animals has been completely specified, synthesized and implemented. The SoC integrates a custom digital pulse processing module with a 32-bit processor running a RT/OS, whose scheduler guarantees that no high priority control task gets blocked by a lower priority acquisition task.

This single embedded processor is used for both slow control, though a reliable TCP channel, and data streaming, through a light-weight UDP channel, where all data is sent through a fast Ethernet link. Moreover HW/SW components have been tuned to achieve efficient and high speed data streaming. Particularly two alternative core interfaces have been considered: direct FSL and a standard OPB. Regarding SW, LWIP library and Ethernet drivers have been optimized to maximize data streaming and a linker script has been tuned to guide the compiler for an efficient placement of the SW in memory.

Although SW development is still going on, we have assessed the streaming bandwidth to be around 40 Mbps ($\sim$300 Kcps). However there is still margin for architectural improvements; as an example we plan to integrate a direct memory access (DMA) controller that will speed up data transfer between custom DSP memory and the Ethernet module. Therefore we are confident of meeting our goal of a sustained maximum count rate of 500 kcps per PS-PMT, which is the expected maximum number of singles for the conventional detector size with small animals.

A Spartan3-400k board has been used for HW/SW integration and development. However the current on-chip system fits quite tightly in our development boards, with the DSP core responsible for half of the area. Therefore we have already designed our own prototype with a Spartan3-1M that mimics the development kits currently assembled and includes new features to improve reliability and flexibility, compared to existing boards. Once our prototypes are up and running with the presented design, we will integrate our analog readout and start doing field acquisitions.

Acknowledgment

The authors thank Xilinx Inc. for the generous donations of software and hardware through the Xilinx University Program. They also acknowledge Vanmicrosistemas for their prototyping board donations.

References

[1] J. W. Young, "FPGA based front-end electronics for a high resolution PET scanner," in *Nuclear Science Symp.*, 1999, vol. 2, pp. 902–906.
[2] D. F. Newport and J. W. Young, "An ASIC implementation of digital front-end electronics for a high resolution PET scanner," *IEEE Trans. Nuclear Sci.*, vol. 40, pp. 1017–1019, 1993.
[3] M. S. Musrock, "Performance characteristics of a new generation of processing circuits for PET applications," *IEEE Trans. Nuclear Sci.*, vol. 50, pp. 974–978, 2003.
[4] K. Ziemons, "The ClearPET LSO/LuYAP phoswich scanner: A high performance small animal PET system," in *Proc. IEEE Nuclear Science Symp.*, 2003, vol. 3, pp. 1728–1732.
[5] P. Stenstrom, "Evaluation of a data acquisition system for SPECT (PET)," *IEEE Trans. Nuclear Sci.*, vol. 47, pp. 1655–1659, 2000.
[6] J.D. Martinez, "High-speed data acquisition and digital signal Processing system for PET imaging techniques applied to mammography," *IEEE Trans. Nuclear Sci.*, vol. 51, pp. 407–412, 2004.

[7] R. Fontaine, "Design of a dual-modality, digital positron emission tomography/computed tomography (PET/CT) scanner for small animal imaging," in *Proc. IEEE Engineering in Medicine and Biology Society Conf.*, 2003, vol. 1, pp. 998–1001.

[8] P. Guerra, "Modeling the acquisition front-end in high resolution gamma-ray imaging," in *Proc. IEEE Nuclear Science Symp.*, 2004, vol. 5, pp. 3089–3093.

[9] R. Lecomte, "Technology challenges in small animal PET imaging," *Nuclear Instruments and Methods in Physics Research Section A*, vol. 527, pp. 157–165, 2004.

[10] "8-Channel, 10-Bit, 65 MSPS, 1.8 V CMOS Analog-To-Digital Converter (Rev. D)," (in Texas instruments) *Datasheet SBAS272D*, 2005.

[11] "ISA to Ethernet MAC Controller with 10/100 PHY," (in Davicom Semiconductor Inc.) *Datasheet DM9000-DS-F03*, 2004.

[12] J. Seidel, "Depth identification accuracy of a three layer phoswich PET detector module," *IEEE Trans. Nuclear Sci.*, vol. 46, pp. 485–490, 1999.

[13] A. Dunkels, "Design and implementation of the lwIP TCP/IP Stack," in *Computer and Network Architectures Laboratory*. Kista: Swedish Institute of Computer Science, 2001.

[14] B. Ahlgren, "The Performance Of A No-Copy API For Communication," in *Proc. Third IEEE Workshop on the Architecture and Implementation of High Performance Communication Subsystems(HPCS '95)*, 1995, pp. 173–176.

[15] *DM9000 Three-in-one Fast Ethernet Controller*, , Davicom Semiconductor Inc., 2004, Application note DM9000-AP-1.22.