Ultrasound Research Scanner for Real-time Synthetic Aperture Data Acquisition

Jensen, Jørgen Arendt; Holm, Ole; Jensen, Lars Joost; Bendsen, Henrik; Nikolov, Svetoslav; Tomov, Borislav Gueorguiev; Munk, Peter; Hansen, Martin; Salomonsen, Kent; Gormsen, Kim; Hansen, Johnny; Pedersen, Henrik Møller; Gammelmark, Kim

Published in:
I E E E Transactions on Ultrasonics, Ferroelectrics and Frequency Control

Link to article, DOI:
10.1109/TUFFC.2005.1503974

Publication date:
2005

Document Version
Publisher's PDF, also known as Version of record

Citation (APA):
Jensen, J. A., Holm, O., Jensen, L. J., Bendsen, H., Nikolov, S., Tomov, B. G., ... Gammelmark, K. (2005). Ultrasound Research Scanner for Real-time Synthetic Aperture Data Acquisition. I E E E Transactions on Ultrasonics, Ferroelectrics and Frequency Control, 52(5), 881-891. DOI: 10.1109/TUFFC.2005.1503974

Ultrasound Research Scanner for Real-time Synthetic Aperture Data Acquisition

Jørgen Arendt Jensen, *Senior Member, IEEE*, Ole Holm, Lars Joost Jensen, Henrik Bendsen, Svetoslav Ivanov Nikolov, Borislav Gueorguiev Tomov, Peter Munk, Martin Hansen, Kent Salomonsen, Johnny Hansen, Kim Gormsen, Henrik Møller Pedersen, and Kim L. Gammelmark, *Student Member, IEEE*

***Abstract*—Conventional ultrasound systems acquire ultrasound data sequentially one image line at a time. The architecture of these systems is therefore also sequential in nature and processes most of the data in a sequential pipeline. This often makes it difficult to implement radically different imaging strategies on the platforms and makes the scanners less accessible for research purposes. A system designed for imaging research flexibility is the prime concern. The possibility of sending out arbitrary signals and the storage of data from all transducer elements for 5 to 10 seconds allows clinical evaluation of synthetic aperture and 3D imaging. This paper describes a real-time system specifically designed for research purposes.**

The system can acquire multichannel data in real-time from multi-element ultrasound transducers, and can perform some real-time processing on the acquired data. The system is capable of performing real-time beamforming for conventional imaging methods using linear, phased, and convex arrays. Image acquisition modes can be intermixed, and this makes it possible to perform initial trials in a clinical environment with new imaging modalities for synthetic aperture imaging, 2D and 3D B-mode, and velocity imaging using advanced coded emissions.

The system can be used with 128-element transducers and can excite 128 transducer elements and receive and sample data from 64 channels simultaneously at 40 MHz with 12-bit precision. Two-to-one multiplexing in receive can be used to cover 128 receive channels. Data can be beamformed in real time using the system's 80 signal processing units, or it can be stored directly in RAM. The system has 16 Gbytes RAM and can, thus, store more than 3.4 seconds of multichannel data. It is fully software programmable and its signal processing units can also be reconfigured under software control. The control of the system is done over a 100-Mbits/s Ethernet using C and Matlab. Programs for doing, *e.g.*, B-mode imaging can be written directly in Matlab and executed on the system over the net from any workstation running Matlab. The overall system concept is presented along with its implementation and examples of B-mode and *in vivo* synthetic aperture flow imaging.

I. Introduction

Modern ultrasound scanners employ digital signal processing to generate high quality images which are dynamically focused in receive. Processing has to be made on data streams in the Gbytes per second range, and this necessitates the use of dedicated chips to perform processing in real time to keep the cost, space, and power consumption moderate. Fundamental changes to the signal processing are, thus, difficult or impossible to make.

The current scanners perform image acquisition sequentially one line at a time, and the frame rate is, thus, limited by the speed of sound. When imaging flow, several emissions have to be made in the same direction, and this limits the frame rate, especially for large depths and large color flow sectors. For a 100-line image at 15-cm depth, the frame rate can be below 6 Hz, which is unacceptable for cardiac imaging.

New imaging techniques based on synthetic aperture (SA) imaging have therefore been suggested and investigated [1]–[3]. The methods can potentially increase both resolution and frame rate, since the images are reconstructed from RF data from the individual transducer elements. Hereby a perfectly focused image in both transmit and receive can be made. Research in real-time 3D imaging is also underway [4], [5]. The purpose is to make systems that in real time can display a pyramidal volume of the heart, where different slices hereafter can be visualized.

It has been intensely discussed whether SA imaging could give better images, and how they will be affected by tissue motion and limited signal-to-noise ratio. It has also been stated that flow imaging cannot be performed with SA methods. A fundamental problem in SA imaging is the poor signal-to-noise ratio in the images, since a single element is used for emission. This gives a much lower emitted energy compared to using the full aperture in conventional imaging and therefore limits the depth of penetration. A potential solution to this is the use of coded excitation, and several groups are working on employing coded signals to enhance the signal-to-noise ratio [6]–[8]. Coded imaging can also be used to further increase the frame rate [9]–[11]. It has also recently been shown that SA imaging can be used for velocity estimation [12], [13]. This shows that SA systems can be made with all of the functionality of conventional systems. There are, thus, good reasons for constructing an experimental system capable of measuring SA data *in vivo* to evaluate the proposed methods. It is the purpose of this paper to describe such a system and give some examples from its use, demonstrating that many of the problems in SA imaging mentioned above can be solved.

Manuscript received February 4, 2003; accepted October 12, 2004. This work was supported by grants 9700883 and 9700563 from the Danish Science Foundation, by B-K Medical A/S, and by grant EF-782 from the Danish Academy of Technical Sciences.

J. A. Jensen, S. I. Nikolov, B. G. Tomov, P. Munk, H. M. Pedersen, and K. L. Gammelmark are with the Center for Fast Ultrasound Imaging, Ørsted•DTU, Technical University of Denmark, DK-2800 Kgs. Lyngby, Denmark (e-mail: jaj@oersted.dtu.dk).

O. Holm, L. J. Jensen, H. Bendsen, M. Hansen, K. Salomonsen, J. Hansen, K. Gormsen, and H. M. Pedersen are with the IO Technologies A/S, Carl Jacobsens Vej 16, DK-2500 Valby, Denmark.

0885–3010/$20.00 © 2005 IEEE

All of the above techniques require digital signal processing on signals from individual transducer elements. The most advanced techniques also use different codes that change as a function of element and emission number, so that a fully flexible emissions system is needed. For research purposes both demands can be difficult to fulfill with commercial scanners, since they are often highly integrated, and it is difficult to access individual signals. Programming commercial scanners for new imaging techniques is often also either cumbersome or impossible. It is, thus, beneficial to develop a dedicated research system that can acquire, store, process, and display ultrasound images from multi-element transducers for any kind of imaging strategy.

This paper describes such a research scanner, its design, implementation, and use *in vivo*. The primary purpose of the scanner is to acquire complete RF data sets, where the signals on the individual elements are stored and can be used for later processing in development of algorithms and imaging schemes. A large memory covering several heart cycles is, thus, needed. It must be possible to implement any kind of imaging, and a fully flexible transmission system is needed. Not all mentioned methods can be implemented in real time, since the resources needed for real-time processing can be determined only after the algorithm has been developed. A key requirement is therefore to store all data and then transport them to general computer clusters for processing. To allow scanning on human volunteers, it is, however, necessary to have real-time processing capabilities for orientation of the scanning, and the various traditional scanning methods should be implemented. The processing should be so flexible that some of the new methods can be implemented or experimented with for real-time implementation. Finally, the scanner should be easy to program, allowing one to implement entirely new scanning methods with minimal efforts. Such a system can assist in the development of scanning techniques and algorithms and facilitate preclinical trials. Also, such a system makes it possible to investigate the role of tissue motion in SA imaging and to quantify the emitted intensities for the advanced imaging methods.

The demands on the system are further elaborated in the next section. A description of the individual units of the system is given in Section III, and the actual implementation is also shown. The programming model for the system, and how phased array imaging can be done in few lines of code, is shown in Section IV. Examples of clinical use of the system on human volunteers is described in Section V, and a summary of experiences with the system and its further development is given in Section VI.

II. System Specification

The purpose of the system is to make the acquisition of multichannel data in real-time from clinical multi-element ultrasound transducers possible, and to enable real-time or near real-time processing of the acquired data. The system must be capable of performing the beam formation for all currently available imaging methods, and this makes it possible to carry out initial trials with new imaging modalities for synthetic aperture imaging, 3D imaging, and 2D and 3D velocity estimation. It must be capable of working in a clinical environment to evaluate the performance of various algorithms on human volunteers.

The system is specifically intended for research purposes, and is not intended for commercial use. The size of the system is, thus, not very important, and it is not necessary to make it easily transportable. Also, it is used only by experts and the user interface to the medical doctor is of minor importance. Most efforts have been put on the flexibility of the system and its ease of configuration for new uses.

The function of the system is defined by the different imaging methods for which it can be used. Each of the imaging types is described below, and the consequences for the system then given.

A. Linear Array Imaging

A linear array image is generated by a multi-element transducer with 128 to 256 elements [14]. The beam is moved by selecting, *e.g.*, 64 adjacent elements and emitting a focused beam from these. The focusing in receive is also done by a number of elements, and dynamic focusing is used. Apodization in both transmit and receive are often applied. The number of active elements is usually 32 to 64. The transducer frequency is from 2 to 10 MHz. Imaging is done down to a depth of 30 cm.

Thus, the demand on the system for 64 channels is simultaneous sampling at 40 MHz. The focusing delay for element i is given by

$$t_d(i) = \frac{|\vec{r}_i - \vec{r}_p| - |\vec{r}_c - \vec{r}_p|}{c}, \tag{1}$$

where c is speed of sound, $\vec{r}_p$ is the imaging point, $\vec{r}_c$ denotes the center element of the active aperture, and $\vec{r}_i$ is the location of the element. Assuming the point to be very close to the transducer gives the maximum delay

$$t_{d\_max} = \frac{|\vec{r}_i| - |\vec{r}_c|}{c}. \tag{2}$$

A linear array often has $\lambda = c/f_0$ pitch, and the maximum delay is then

$$t_{d\_max} = \frac{N_e/2 \cdot c/f_0}{c} = \frac{N_e}{2f_0}, \tag{3}$$

where f_0 is center frequency and N_e is the number of active elements. For 64 elements and $f_0 = 2$ MHz, t_{d_max} equals 16 μs.

The maximum time to sample one line to a depth of d is given by

$$t_s = \frac{2d}{c}. \tag{4}$$

This gives $t_s = 430$ μs, corresponding to 17,200 samples at 40 MHz for a depth of 30 cm.

B. Phased Array Imaging

The beam is electronically swept over the imaging area by using a 64-to-256 element array [15]. The beamforming delays are used both for steering and focusing. The pitch is reduced to $\lambda/2$ to avoid grating lobes, and a very conservative estimate of maximum delay is to assume the imaging point to be at the edge of the array. The maximum delay then is

$$t_{d\_max} = \frac{N_e \lambda/2}{c} = \frac{N_e}{2f_0}, \tag{5}$$

which is equal to the demand for linear array imaging.

The transducer frequency is from 2 to 10 MHz. Investigations are done to a depth of 20 cm.

Thus, the demand on the system for 128 channels is sampling at 40 MHz. The demands on focusing delays, sampling time, and storage are the same as for linear array imaging.

C. Flow Estimation, Spectrum

Beamforming is done in one direction with either a linear or a phased array [16]. The flow signal from blood has 40 dB less power than that from stationary tissue. The dynamic range of the flow signal is 30 dB. The effective number of bits must be 12 or more when the signals from all channels have been combined. The pulse emitted can have from 4 to 16 periods of the center frequency of the transducer or a coded signal can be employed.

D. Flow Imaging

Imaging is done by pulsing repeatedly in one direction and then changing the direction to generate an image [17]. An image can therefore be assembled from up to 1000 pulse emissions.

E. Three-Dimensional Imaging

A matrix element transducer is used with, *e.g.*, 40 × 40 elements [4], [18]. Only some of the elements are used for transmit and receive. The area of the elements is small, and pulsing should be done with 100 to 300 volts. Coded signals should be used. Coded pulses with up to 64 cycle periods must be possible with a high-amplitude accuracy. This corresponds to emission over a period of 32 μs with a sampling frequency of 40 MHz and an accuracy of 12 bits.

The maximum delay conservatively corresponds to the transmission time over the aperture. Assuming $\lambda/2$ pitch gives

$$t_{d\_max} = \frac{\sqrt{2} N_e \lambda/2}{c} = \frac{N_e}{\sqrt{2} f_0}, \tag{6}$$

which for 64 × 64 elements and $f_0 = 2$ MHz gives 23 μs. Parallel lines are generated by using parallel beamformers and reusing data from one pulse emission. The system must be capable of reading the data sampled from one element a number of times, and using different phasing schemes for each cycle through the data. This gives a very high demand on the processing that might not be obtainable by the system.

F. Synthetic Aperture Imaging

A standard linear or phased array multi-element transducer is used. Pulsing is done on a few elements and the received response is acquired for all elements [19], [20]. The image is then reconstructed from only a small number of pulse emissions by using the data from all elements.

This type of imaging needs large amounts of storage and the ability to reuse the data for the different imaging directions. The system must be capable of storing 3 seconds of data for all channels. For a 12-bit resolution at 40 MHz, 64 channels, and 3 seconds of data, this amounts to a total storage demand of more than 14 Gbytes.

The processing of the data cannot be done in the system due to the very large demand on processing. This is solved by storing the data in real time and the transferring them to a multiprocessor system for storage and image reconstruction.

G. Synthetic Aperture Flow Imaging

The data are acquired in the same way as for synthetic aperture imaging, but the focusing is done continuously for all points in the image [21]. Hereby the data for flow estimation are present for all image locations for all time. It is, thus, very important that the image measurement be continuous over the whole data acquisition time.

H. Fast Coded Synthetic Aperture Imaging

Here the image is acquired by sending different codes on the individual elements for different emissions [9]–[11]. The image is then created by processing the received signals using different filters to separate out the individual signals.

The system must, thus, be capable of sending an arbitrary code that is different for the different channels, and that can change from emission to emission.

In general the demands for all of these imaging modes can be condensed into a few generic demands. The transmission system should be capable of sending an arbitrary signal on each element. The signals can be different from element to element and from emission to emission. The transmission system must be capable of focusing the transmission differently for emission to emission. The sampling frequency should be 40 MHz and the resolution is determined by the dynamic range of the system, which should be above 60 dB. A 12-bit digital-to-analog converter (DAC) is, thus, sufficient. Coded waveforms should be emitted for up to 32 μs and different codes should be emitted for different emissions and elements. For 100-image lines this gives 80 ksamples of memory for each transducer element.

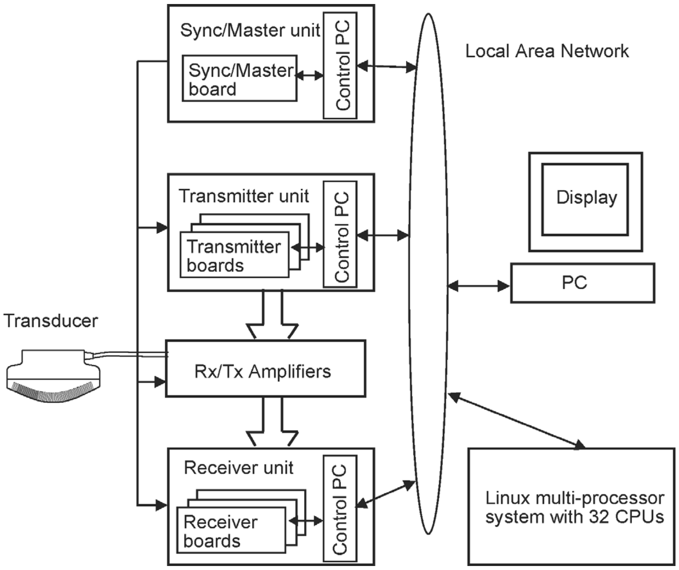

Fig. 1. Overall diagram of system.

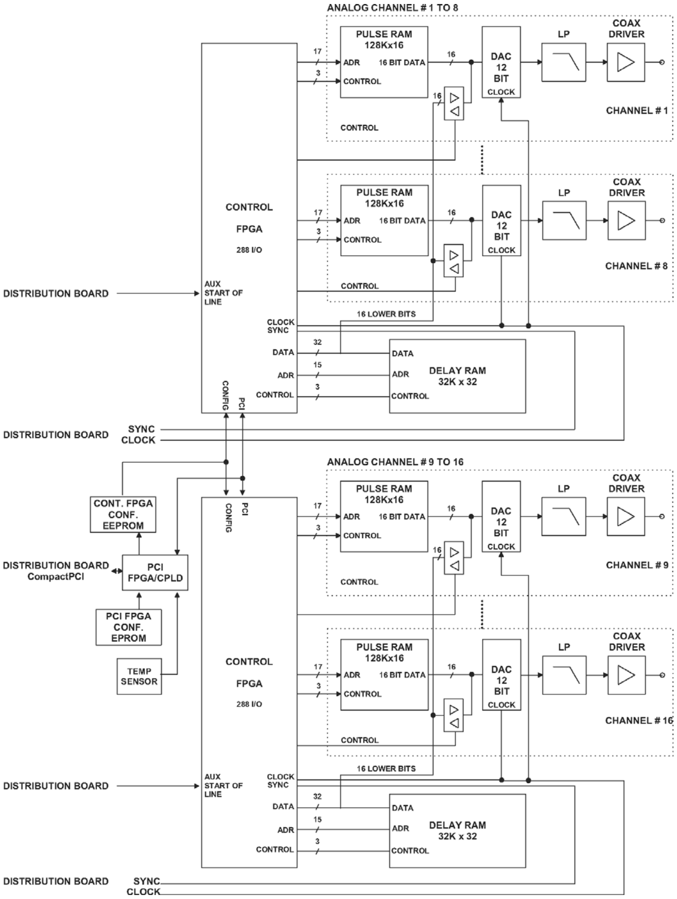

Fig. 2. Main diagram of the transmitter board.

The receiving system should be capable of sampling full RF data from all channels. The sampling frequency should be 40 MHz and the resolution at least 12 bits. Sampling should be possible over a couple of heart beats. Three seconds of real-time data at 40 MHz with 2 bytes per sample gives a demand of 229 Mbytes, so that a RAM of 256 Mbytes per channel is necessary. To process the data it must be possible to dynamically focus the data in real time. A focusing chip must be used for each channel, and this chip must have memory with parameters for the focusing, and the system must be capable of summing the data from the individual channels. Ideally the processing should be easily reconfigurable, and it must be possible to store the raw RF data, the focused data, or the focused and summed data. The beamformed and summed data should also be transported out of the system in real time for further processing and display on a screen.

III. System Realization

The Remotely Accessible Software configurable Multichannel Ultrasound Sampling (RASMUS) system consists of four distinct modules: transmitters, analog amplifiers (Rx/Tx amplifiers), receivers, and a sync/master unit. The main blocks are depicted in Fig. 1. The connection to the transducer is through a 128-wire coaxial cable through the Rx/Tx amplifiers. The transmitter sends the signals through the transmit amplifiers, and the receiver unit samples the amplified and buffered signals from the Rx/Tx amplifiers. The sync/master unit holds a crystal oscillator and controls the central timing of the scanning process. The overall operation of the system is controlled through a number of single-board PCs in the individual units interconnected through a standard 100-Mbits/s Ethernet. The waveforms and focusing delay data are transmitted from the controlling PC to the transmitters and receiver boards. The data from the sampling are processed by field programmable gate arrays (FPGAs) that can be configured for specific signal processing tasks over the net. One Focus FPGA is used for each element and a Sum FPGA is placed for each eight elements. The processed and summed signal can then be routed from Sum FPGA to Sum FPGA through a cascade bus. The resulting signal is read by one or more signal processors (ADSP) that can be connected through serial link channels capable of transmitting 40 Mbytes per second. The beamformed signal is sent via the link channels to the PC for further processing and display.

The following paragraphs detail the overall design of the individual boards.

A. Transmitter

The overall diagram of the transmitter is shown in Fig. 2 and its layout is shown in Fig. 3. Each transmitter board has 16 channels each having a 128-ksample pulse RAM connected to a 40-MHz, 12-bit DAC. The pulse RAM is controlled by two FPGAs, where the individual waveforms are selected as a memory start address and a transmit delay. The delay RAM holds the start address of the waveform in the pulse RAM and the corresponding delay for each line. The delay RAM is implemented as 32 k × 32 bit SRAM. At the start of each line the pulse emission

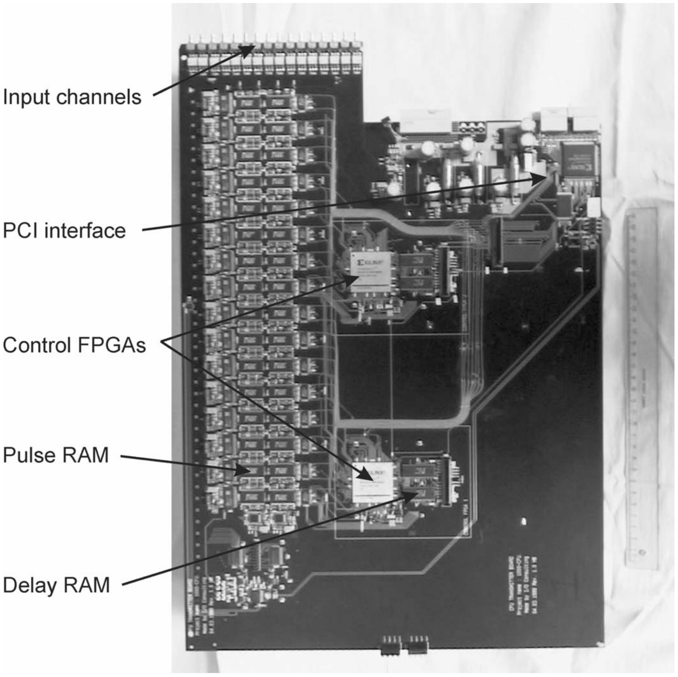

Fig. 3. The transmitter board layout. The board size is 53 cm by 36.5 cm. The size is indicated by the 30-cm ruler on the right.

is delayed according to the delay value for each channel. Both the waveform and the delay can change for each emission, which ensures full flexibility in the transmissions. The length of the waveforms can be set, and waveform durations up to 100 μs can be emitted.

The whole board is controlled through a compact peripheral component interconnect (PCI) interface from which all the memory can be accessed and the FPGAs can be programmed. The system houses 8 boards for a total of 128 independent emission channels.

B. Receiver

The receiver board diagram is shown in Fig. 4. It samples and processes 8 analog signals selected from 16 inputs through a 2-to-1 multiplexer. Each of the input signals is sampled at 40 MHz and 12 bits into one of the two buffer SRAMs. One of the buffer SRAMs is used for sampling and the other one is used for transferring data to the Focus FPGA associated with each channel. The data are then processed in the Focus FPGA using parameters from the 128-Mbyte Focusing RAM, and the result is passed on to the Sum FPGA. Either the processing can be a dynamic receive focusing or the data can be passed unaltered for later storage. The Sum FPGA can either store the data in the 2-Gbyte storage RAM or sum all 8 channels with the result from a cascade bus and pass the data on to the next receiver board through this cascade bus. The individual RF data can be both stored in the RAM and processed by the FPGAs at the same time. The last receiver board in the system then transmits the focused signal to the SHARC ADSP for transmission to the display PC [23]. The storage RAM can contain more than 3 seconds of real-time data for each channel that can be accessed later from the PCs controlling the system.

The actual board layout is shown in Fig. 5. All boards in the system have a size of 53 cm by 36.5 cm, and the receiver is manufactured using a 12-layer printed circuit board.

The receiver boards are accessible from a general-purpose, Linux-based, single-board PC which controls the different boards. All memory can be accessed from this interface, and it is possible to upload code to the FPGAs through the flash memory. A FIFO memory can be programmed to replace the SRAM for sampling and the FIFOs can be used for debugging the system. A mailbox interface to the SHARC ADSP is also accessible from the PCI interface. This is used for sending messages to the ADSP.

1. Focus FPGA: The Focus FPGA controls the initial storing and processing of the sampled data. The Focus FPGA fetches the sampled data from the SRAM and the corresponding focusing parameters from the Focusing RAM and processes the data before transferring the result to the Sum FPGA.

Two independent memory-burst SRAM banks are used to bank switch between the sampled data and the processed data. While the sampled data are being written to one of the two banks, the other bank can be read by the Focus FPGA. Each SRAM is implemented as 256 ksamples, which is equivalent to a line length of 3.3 ms sampled at 40 MHz, since samples are stored twice for making focusing faster.

The basic focusing algorithm currently implemented uses a combination of coarse and fine delays. The coarse delay is in steps of the 25-ns sampling interval, and it is implemented as a general table look-up-address generator. For each sample a 16-bit address index is read from the Focusing RAM. In this way focusing in any order can be implemented. The fine delay is implemented as a linear interpolation between two samples using two signed 16-bit apodization coefficients, which are read from the focusing RAM for each sample. To make one output sample per clock cycle, it is, thus, necessary to read two input samples, which is why the input samples are stored twice in the SRAM. Using this scheme any delay can be implemented, and the data can also be decimated during the focusing process.

The Focus FPGA is implemented using a 300-kgate XILINX device from the Virtex family: XCV300 in a 352-pin BGA package speed grade −4. The simple B-mode beamformer described above uses less than 10% of the logical resources of the chip for real-time B-mode beam formation. This leaves the possibility for investigating hardware implementations of more advanced beamformers including pulse compression, synthetic aperture, and parallel beamforming.

2. Sum FPGA: The Sum FPGA is used to perform digital signal processing on the 8 channels. The most basic operation is to sum the 8 focused channels. Further, it is used as a gateway between the eight independent sampling

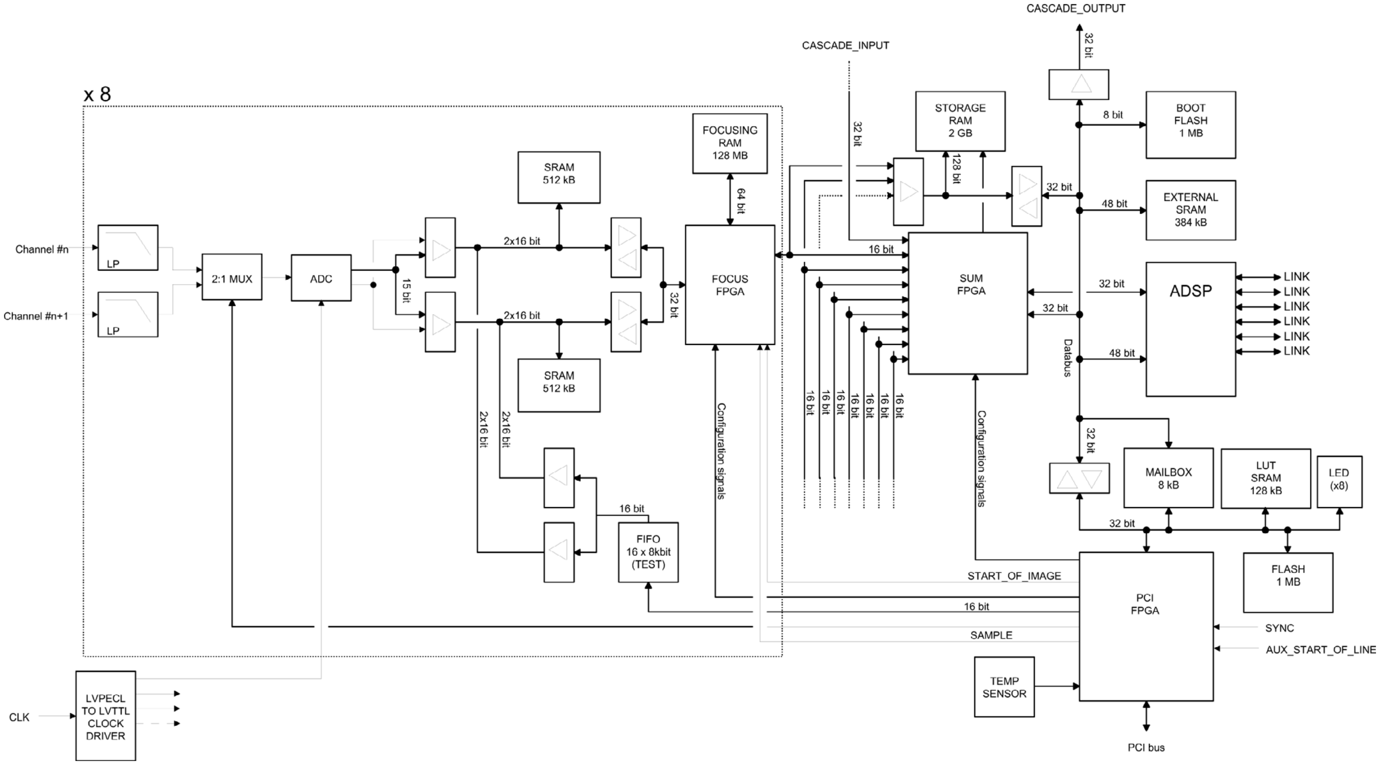

Fig. 4. Main diagram of receiver board (from [22]).

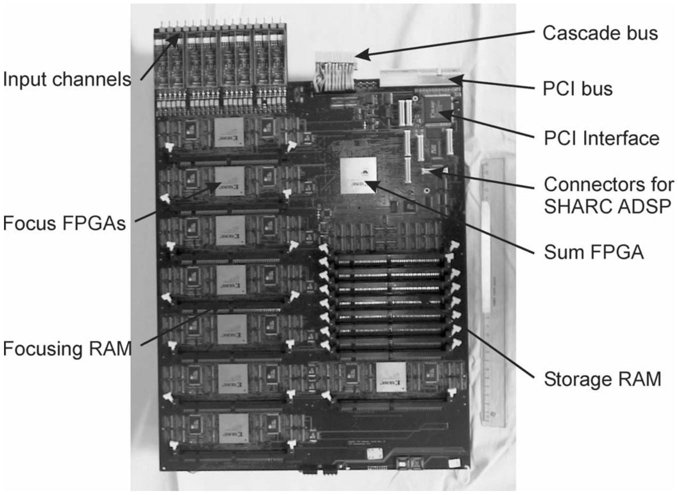

Fig. 5. The receiver board layout without RAM modules to reveal the processing electronics. The board size is 53 cm by 36.5 cm.

channels and the ADSP. The Sum FPGA controls the 2-Gbyte storage SDRAM.

When the focusing is done in the Focus FPGA, the 8 channels are added to the accumulated sum that is passed to the next receiver board using a high-speed cascade bus connecting the Sum FPGAs directly with each other. The last Sum FPGA in the chain uses the ADSP link ports to transfer the final result to an external PC that displays the final image.

The Sum FPGA is implemented using a 1-Mgate XILINX device from the Virtex family: XCV1000 in a 560-pin BGA package speed grade −4. The simple design described above uses less than 5% of the logical resources of the chip for conventional real-time imaging.

C. Sync/Master Unit

The sync/master unit controls the timing of the system. A highly stable oven-controlled crystal oscillator generates the 40-MHz clock frequency. The clock jitter is below 5 ps. The clock is distributed using coax cables and emitter-coupled logic (ECL) in order to minimize jitter. The timing source also transmits a synchronization (SYNC) signal. The receiver and transmitter uses the SYNC signal to start and stop the sampling cycles. An image consists of a number of lines, each with a transmission of a pulse and a reception of the echoes. The transmitter and receiver generate an internal LINE signal from the SYNC signal to control the sampling process for the received signal.

The sync/master board also generates the time-gain compensation (TGC) signal used for the Rx/Tx boards. The sampling clock for the TGC waveform is 1 MHz. The values for the TGC and generation of the synchronization signals are loaded through the board's PCI interface.

D. Current Processing Capabilities

The current system only uses 10% of the available capacity in the FPGAs. With the programs it is possible to perform real-time beam formation for linear, phased, and convex arrays. The beam formation for all conventional ultrasound systems can, thus, be implemented. The postprocessing for image display and flow estimation is today done in PCs, but could be implemented in the Sum FPGA.

It is possible to focus data at a rate of 40 MHz. This gives 8000 beamformed samples for a 200-μs acquisition corresponding to a depth of 15 cm. Sparse output samples could also be generated [24] and normally 512 complex samples per line is sufficient for B-mode display, so that a 1-to-8 parallel beamformer can easily be implemented. The remaining resources can then be used for filtration, code compression, or other processing.

Currently it is not possible to make real-time processing of SA images. Here a full image must be formed after each pulse emission corresponding to more than 100 parallel beamformers. Compression of coded signal should also be done. The main bottlenecks are currently the data transfer rates, since the FPGAs still have room for more processing. The data acquisition can, however, always be done in real time, which is the most important feature for research purposes. Since different image sequences can be intermixed, a conventional B-mode image can be acquired and processed in real time for orientation purposes. The SA image can then be acquired in between B-mode images and full orientation of the scanning can therefore be obtained.

E. Physical Realization

Two 19-inch racks house the 8 transmitter and the 8 receiver boards, respectively, as shown in Fig. 6. The sync/master board is located in the rack cabinet for the receiver boards. Each of these racks also houses a slot PC running Linux, which controls the setup and operation of the boards. A separate enclosure is used for the analog front-end, which is shielded from the digital electronics. Here linear laboratory power supplies (Toellner TOE 8841 and 8852, Herdecke, Germany) are also used to supply the front-end to keep the noise low.

Although the system is quite large and weighs several hundred kilograms, it is still transportable and it can be moved around in the lab by two persons. The system is now fully functional and has been used for acquiring data for several new imaging methods, as described in Section V.

IV. Programming of the System

From the software point of view, the system consists of several computers that are directly connected to a number of transmitter and receiver boards and a single sync/master board. The computers are linked by a LAN, and use Linux as the operating system and TCP/IP as the underlying communication protocol.

The software was designed to meet the following requirements:

- Flexibility and ease of use. It is of prime importance that new imaging methods can be quickly implemented with a minimal amount of programming even for new users.

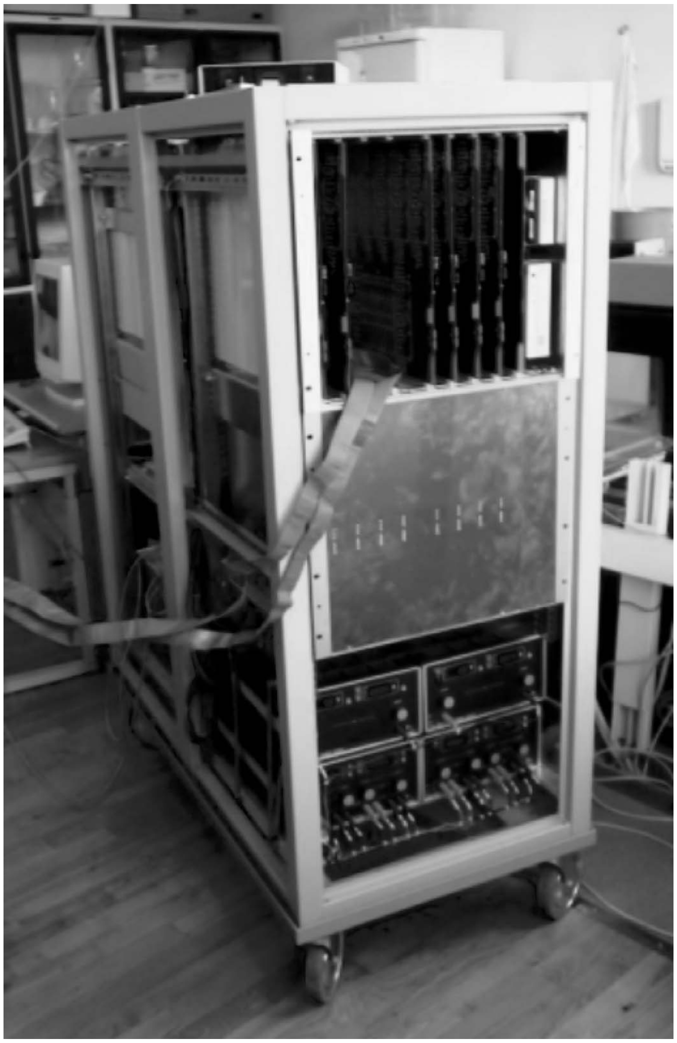

Fig. 6. The RASMUS scanner seen from the front with the receivers on the top, the transmitters in the middle, and the power supplies in the bottom. The analog front-end and transducer connection along with a control PC is placed in the other rack cabinet.

- Data encapsulation. All data for the imaging algorithm are stored in the boards of the scanner.
- Distributed computations. The computers work independently of each other, and each calculates only those imaging parameters for boards that are connected to that particular computer.
- Portability. The software is written in ANSI C and is platform-independent.

The client/server communication model was adopted for the software. The computers controlling the boards run a *server* program. The server waits for requests coming from the LAN and processes them. The requests can be sent by any *client* program running on a computer connected to the LAN using the TCP/IP communication protocol.

Fig. 7 shows the client-server model of the software. At start-up the server detects which boards are in the PCI enclosures. The computers can handle any combination of transmitter and receiver boards plugged into the same PCI back-plane. The server is in idle mode until an event

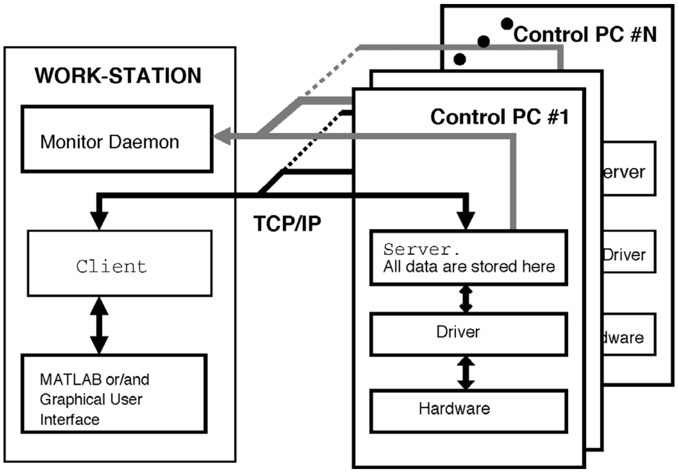

Fig. 7. Client-server model of software.

occurs. In the case of a hardware malfunction, the server sends an emergency message to a program, called *monitor daemon.* Another event is a request by the *client.* The request can be for transferring parameters to the boards, for performing some calculations, or for retrieving data from the boards to the computer with the user.

The interface to the client program is implemented as a Matlab (The MathWorks Inc., Natick, MA) tool box. The function calls are implemented to be as close to the functions in the simulation program Field II [25] as possible. Algorithms created using Field II can thereby easily be tested on the scanner with only minor changes in the Matlab program. An example for setting the system to perform phased array B-mode imaging is shown below:

```
% Auto-detect and initialize the system
sys_init('auto');
% Set the pulse repetition frequency
sys_set_fprf(fprf);
% Set the sampling range gate in receive
sys_set_sampling_interval(start_depth, end_depth);
% Set the number of scan-lines per frame
sys_set_no_lines(no_lines);
% Define the transducer. Necessary for the delay calculations
tr_linear_array(no_elements, width, height, kerf);
% Do for all lines in the image:

for line_no = 1: no_lines

 % Set the pulse and the apodization for the current line
 xmt_excitation(waveform(line_no));
 xmt_apodization(line_no,xmt_apo(line_no, : ));
 rcv_apodization(line_no,times, rcv_apo(line_no, : , : ));

 % Set the focus, defined in 3D coordinates
 xmt_focus(line_no,focus(line_no));
 rcv_dynamic_focus(line_no, theta(line_no), fi(line_no));
end

% Set the time-gain compensation curve
tmg_tgc(tgc_vector);
% Start the continuous imaging process
tmg_start
```

In order to make the system perform linear array imaging, only one line needs to be added, which changes the origin of the individual scan lines.

V. Examples of Use

The initial design of the system was started in October 1998 and the first images using the full system were acquired in the spring of 2001. The full real-time capabilities of the system were completed in 2002 and the RASMUS system is now capable of real-time focusing for conventional imaging and real-time data acquisition for all modes. This is illustrated in the following examples.

The first example shows a comparison between a traditional linear array image and a synthetic aperture image. A commercial 7-MHz probe is used and the system is programmed to alternate between linear array imaging and synthetic aperture imaging. Therefore, both images are acquired at the same time on the same patient and with the same view, and a direct comparison can be made. The linear array image uses 64 elements in transmit and 128 in receive, and the active aperture is moved across the physical aperture to perform the image. The synthetic aperture image is acquired using a number of spherical wave emissions. Normally one element is used in transmit, but to ensure sufficient transmitted energy here, a number of elements are used in transmit. A set of 33 elements are used in transmission to form a spherical wave, and this emission aperture is moved over the physical aperture. This is done for 96 emissions. A 20-μs chirp signal is also used during emission, and two emissions are used for sampling the whole 128-element aperture using the 2-to-1 receive multiplexing. More details on this temporally encoded multi-element SA imaging (TMS) method and the corresponding intensity levels can be found in [8].

The resulting images are shown in Fig. 8, with the linear array image to the left and the synthetic aperture image to the right. It can be seen that the SA image attains a 40% increase in depth of penetration and that the resolution for the deeper-lying wires are improved [8].

The second example demonstrates the system's ability to estimate flow *in vivo.* The image is made using synthetic aperture techniques with a 7-MHz linear array. Here 11 elements are grouped for the emission. Eight emissions are used over the full 128-element aperture to form an image. A 20-μs chirp is used as excitation to ensure a good signal-to-noise ratio. The 64 elements closest to the center of the emission are sampled during reception and hereby signals from all 128 elements are sampled for the whole emission sequence. The sequence is repeated 16 times for a total of 128 emissions with a pulse repetition frequency of 3 kHz. The velocity is then estimated by focusing along the direction of the flow and these signals are cross-correlated to estimate the movement and thereby the velocity. Details about the method can be found in [13]. The resulting image taken of the carotid artery (bottom) and jugular vein (top) in the neck of a male volunteer is shown in Fig. 9.

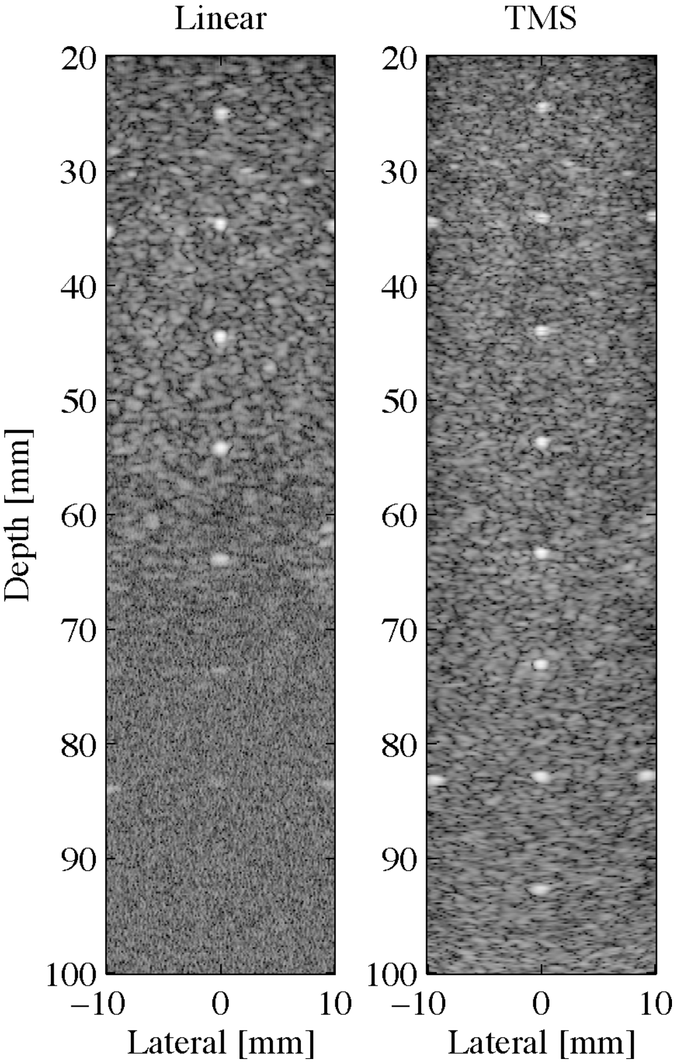

Fig. 8. Linear array (left) and synthetic aperture scan (right) of nylon wires in a tissue mimicking phantom with an attenuation of 0.5 dB/(MHz cm) (from [8]).

It should be noted that the raw estimates are displayed, and the velocity is, thus, accurately estimated. The estimates start at a depth of only 13 mm due to the use of the chirp excitation. The white areas at the bottom of the artery are artifacts from the velocity estimation process. This is due to the selection of a wrong peak in the cross-correlation function [26], probably due to an insufficient signal-to-noise ratio.

In the last example, data are sampled to a depth of 32 mm, although the pulse repetition frequency is 3 kHz, so that sampling is only done in 12.5% of the time. The system is then capable of storing a sequence of image data for 26.9 seconds.

Several other image types have also been acquired, such as convex synthetic aperture images [27], 3D synthetic aperture data [28], and transverse flow data [29] and pre-clinical trials of convex array SA imaging [30]. The system has, thus, shown that it is a very flexible tool that is able to sample data for nearly any imaginable application.

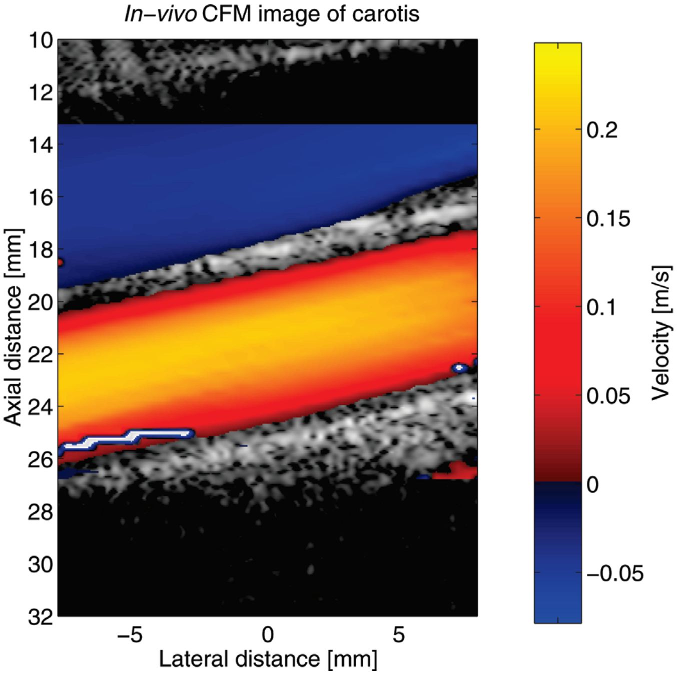

Fig. 9. *In vivo* color flow map image at a 77-degree flow angle for the jugular vein and carotid artery. The color scale indicates the velocity along the flow direction: red hues indicate forward flow and blue reverse flow (from [13]).

VI. Summary and Future Extensions

The RASMUS system has been described in this paper. The system is currently fully operable and gives high quality *in vivo* data that can be used for devising and testing new imaging schemes such as synthetic aperture imaging, directional flow techniques, and real-time three-dimensional imaging. The main method for operation has been real-time data acquisition with storage of all the data in the system's RAM. The data are then transferred for later processing on a Linux cluster with 32 CPUs. The model of control under Matlab has shown to be very efficient, since the user can work at a very high level of control without being buried in the system's details. More than 15 people have currently been using the system without a very long introduction to its facilities. The software model has also greatly helped in debugging the system, since small scripts for testing can easily be written. The network feature has also eased the use. Most phantom experiments for debugging have been set up in the laboratory, and then the script debugging and data acquisition is done from the office.

By far the most important features are the flexible transmission system and the real time storage facility that are nearly always used. The main annoyance in the use of the system has been the slow transfer of data out of the system's memory. This is currently done through the controlling PCs, and this is limited by the 100-MBit/s Ethernet and the PCI interface of the boards. The PCI interface is based on an FPGA, and it is currently slow and limited to a transfer rate of 2–8 Mbytes/s. A full experiment storing 16 Gbytes of data ideally takes nearly one hour to transfer to external disk at best, and often it takes longer. The slow transfer should be addressed in an updated version of the system, which could be based on a 1-Gbit/s Ethernet and an optimized PCI interface design.

The real-time capability has been the most difficult to get working. It relies on the global timing of the whole system and the coordination of all 72 FPGAs in the processing chain, and has been the last feature to work. Some of the examples mentioned in Section V have therefore been made by scanning with a normal B-K Medical 2102 scanner (B-K Medical A/S, Herlev, Denmark) and then switching the probe to the RASMUS system for data acquisition, which is very cumbersome. The real-time processing can now make dynamic receive focusing for all transducer models. This makes real-time orientation possible, and the system can be programmed to acquire several different images in the same sequence for comparison. Also, the potential of the FPGA processing solution needs to be determined. The current processing fills 10% of the FPGAs. Many more options exist and are currently being investigated.

The system has, however, already been capable of acquiring high-quality data for linear, phased, and convex SA imaging and for SA flow imaging, and the first preclinical trials comparing conventional and SA imaging have been conducted.

Acknowledgment

The anonymous reviewers are thanked for their many useful comments.

References

[1] J. T. Ylitalo and H. Ermert, "Ultrasound synthetic aperture imaging: Monostatic approach," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 41, pp. 333–339, 1994.

[2] C. R. Hazard and G. R. Lockwood, "Theoretical assessment of a synthetic aperture beamformer for real-time 3-D imaging," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 46, pp. 972–980, 1999.

[3] S. I. Nikolov, K. Gammelmark, and J. A. Jensen, "Recursive ultrasound imaging," in *Proc. IEEE Ultrason. Symp.*, 1999, pp. 1621–1625.

[4] S. W. Smith, H. G. Pavy, and O. T. von Ramm, "High-speed ultrasound volumetric imaging system—Part I: Transducer design and beam steering," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 38, pp. 100–108, 1991.

[5] G. R. Lockwood, J. R. Talman, and S. S. Brunke, "Real-time 3-D ultrasound imaging using sparse synthetic aperture beamforming," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 45, pp. 980–988, 1998.

[6] P. Li, E. Ebbini, and M. O'Donnell, "A new filter design technique for coded excitation systems," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 39, pp. 693–699, 1992.

[7] T. X. Misaridis, K. Gammelmark, C. H. Jørgensen, N. Lindberg, A. H. Thomsen, M. H. Pedersen, and J. A. Jensen, "Potential of coded excitation in medical ultrasound imaging," *Ultrasonics*, vol. 38, pp. 183–189, 2000.

[8] K. L. Gammelmark and J. A. Jensen, "Multielement synthetic transmit aperture imaging using temporal encoding," *IEEE Trans. Med. Imag.*, vol. 22, pp. 552–563, 2003.

[9] T. Misaridis, "Ultrasound imaging using coded signals," Ph.D. dissertation, Ørsted•DTU, Technical University of Denmark, Lyngby, Denmark, 2001.

[10] T. Misaridis and J. A. Jensen, "A new coding concept for fast ultrasound imaging using pulse trains," in *Proc. SPIE: Prog. Biomed. Optics Imag.*, vol. 3, 2002, pp. 68–78.

[11] F. Gran and J. A. Jensen, "Multi element synthetic aperture transmission using a frequency division approach," in *Proc. IEEE Ultrason. Symp.*, 2003, pp. 1942–1946.

[12] S. I. Nikolov and J. A. Jensen, "Velocity estimation using synthetic aperture imaging," in *Proc. IEEE Ultrason. Symp.*, 2001, pp. 1409–1412.

[13] J. A. Jensen and S. I. Nikolov, "Transverse flow imaging using synthetic aperture directional beamforming," in *Proc. IEEE Ultrason. Symp.*, 2002, pp. 1488–1492.

[14] A. Macovski, "Ultrasonic imaging using arrays," *Proc. IEEE*, vol. 67, pp. 484–495, 1979.

[15] F. L. Thurstone and O. T. von Ramm, "A new ultrasound imaging technique employing two-dimensional electronic beam steering," in *Acoustical Holography.* vol. 5, P. S. Green, Ed. New York: Plenum Press, 1974, pp. 249–259.

[16] D. W. Baker, "Pulsed ultrasonic Doppler blood-flow sensing," *IEEE Trans. Sonics Ultrason.*, vol. SU-17, pp. 170–185, 1970.

[17] C. Kasai, K. Namekawa, A. Koyano, and R. Omoto, "Real-time two-dimensional blood flow imaging using an autocorrelation technique," *IEEE Trans. Sonics Ultrason.*, vol. SU-32, pp. 458–463, 1985.

[18] O. T. von Ramm, S. W. Smith, and H. G. Pavy, "High-speed ultrasound volumetric imaging system—Part II: Parallel processing and image display," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 38, pp. 109–115, 1991.

[19] S. Bennett, D. K. Peterson, D. Corl, and G. S. Kino, "A real-time synthetic aperture digital acoustic imaging system," in *Acoust. Imaging*, vol. 10, P. Alais and A. F. Metherell, Eds. 1982, pp. 669–692.

[20] M. Karaman, P. C. Li, and M. O'Donnell, "Synthetic aperture imaging for small scale systems," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 42, pp. 429–442, 1995.

[21] S. I. Nikolov and J. A. Jensen, "In-vivo synthetic aperture flow imaging in medical ultrasound," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 50, pp. 848–856, 2003.

[22] J. A. Jensen, O. Holm, L. J. Jensen, H. Bendsen, H. M. Pedersen, K. Salomonsen, J. Hansen, and S. Nikolov, "Experimental ultrasound system for real-time synthetic imaging," in *Proc. IEEE Ultrason. Symp.*, vol. 2, 1999, pp. 1595–1599.

[23] S. I. Nikolov, J. P. G. Gonzaléz, and J. A. Jensen, "Real time 3D visualization of ultrasonic data using a standard pc," *Ultrasonics*, vol. 41, pp. 421–426, 2003.

[24] B. G. Tomov and J. A. Jensen, "A new architecture for a single-chip multi-channel beamformer based on a standard FPGA," in *Proc. IEEE Ultrason. Symp.*, 2001, pp. 1529–1533.

[25] J. A. Jensen, "Field: A program for simulating ultrasound systems," *Med. Biol. Eng. Comp.*, 10th Nordic-Baltic Conf. Biomed. Imag. vol. 34, suppl. 1, pt. 1, pp. 351–353, 1996b.

[26] J. A. Jensen, "Artifacts in velocity estimation using ultrasound and cross-correlation," *Med. Biol. Eng. Comp.*, vol. 32/4, suppl., pp. s165–s170, 1994a.

[27] K. L. Gammelmark and J. A. Jensen, "Experimental study of convex coded synthetic transmit aperture imaging," in *Proc. IEEE Ultrason. Symp.*, 2002, pp. 1573–1576.

[28] S. I. Nikolov, R. Dufait, A. Schoisswohl, and J. A. Jensen, "Three-dimensional real-time synthetic aperture imaging using a rotating phased array transducer," in *Proc. IEEE Ultrason. Symp.*, 2002, pp. 1545–1548.

[29] J. A. Jensen and R. Bjerngaard, "Directional velocity estimation using focusing along the flow direction. II: Experimental investigation," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 50, pp. 873–880, 2003.

[30] M. H. Pedersen, K. L. Gammelmark, and J. A. Jensen, "Preliminary in-vivo evaluation of convex array synthetic aperture imaging," in *Proc. SPIE—Ultrason. Imag. Signal Process.*, 2004, pp. 33–43.

Jørgen Arendt Jensen (M'93–SM'02) earned his Master of Science in electrical engineering in 1985 and the Ph.D. degree in 1989, both from the Technical University of Denmark. He received the Dr.Techn. degree from the university in 1996. He has published a number of papers on signal processing and medical ultrasound and the book "Estimation of Blood Velocities Using Ultrasound", Cambridge University Press in 1996. He is also developer of the Field II simulation program. He has been a visiting scientist at Duke University, Stanford University, and the University of Illinois at Urbana-Champaign. He is currently full professor of Biomedical Signal Processing at the Technical University of Denmark at Ørsted•DTU and head of Center for Fast Ultrasound Imaging. He has given courses on blood velocity estimation at both Duke University and University of Illinois and teaches biomedical signal processing and medical imaging at the Technical University of Denmark. He has given several short courses on simulation, synthetic aperture imaging, and flow estimation at international scientific conferences. He has received several awards for his research. He is also the co-organizer of a new biomedical engineering education offered by the Technical University of Denmark and the University of Copenhagen. His research is centered around simulation of ultrasound imaging, synthetic aperture imaging and blood flow estimation and constructing systems for such imaging.

Svetoslav Ivanov Nikolov received M.Sc. in electrical engineering and MBA in international business relations from the Technical University of Sofia in 1996 and 1997, respectively. In 2001 he got a Ph.D. degree from the Technical University of Denmark, Lyngby. His dissertation explored approaches for synthetic aperture tissue and flow imaging, and possibilities for real-time 3D imaging.

After completing his doctoral work, he stayed on at Ørsted•DTU, Technical University of Denmark as an assistant professor, where he teaches digital design, software development, and digital signal processing. His current research interests are focused on real-time systems and signal processing, and methods for high-resolution tissue and flow imaging.

Borislav Gueorguiev Tomov was born on Nov. 28, 1973 in Montana, Bulgaria. He earned a M.Sc. degree in electronics from the Technical University of Sofia, Bulgaria, in 1996, and Ph.D. degree form the Danish Technical University, Denmark, in 2003. He is currently an Assistant Professor at the latter. His research interests include ultrasound imaging and digital signal processing.

Peter Munk received a B.Sc. degree in electrical engineering in 1987 from the University College of Aarhus, and M.Sc. and Ph.D. degrees in 1996 and 2000, respectively, from the Technical University of Denmark (DTU) in Copenhagen.

He joined Brüel & Kjaer A/S, Denmark, later B-K Medical A/S, from 1987 to 2002, where he worked with development and research for medical ultrasonic imaging. He is currently employed at NATO SACLANT Underwater Research Centre, Italy, working on mine-hunting sonar systems on AUVs. His areas of interest are acoustics, statistics, navigation, and multi-dimensional signal processing for imaging and velocity estimation.

Kim Løkke Gammelmark (S'02) was born in Fakse, Denmark on May 1, 1975. He received his M.S. degree in electrical engineering from the Technical University of Denmark, Kgs. Lyngby, Denmark, in August 2001. He completed his Ph.D. project in Biomedical Engineering at the Ørsted•DTU, the Technical University of Denmark in 2004. His major research interests are the application of synthetic aperture techniques in medical ultrasound imaging, and medical ultrasound beamforming.