



Published in final edited form as:

*IEEE Trans Med Imaging*. 2012 February ; 31(2): 370–379. doi:10.1109/TMI.2011.2169681.

## Medusa: A Scalable MR Console Using USB

**Pascal P. Stang,**

Magnetic Resonance Systems Research Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA, 94305 USA pstang@mrsrl.stanford.edu

**Steven M. Conolly,**

Department of Bioengineering, University of California, Berkeley, CA, 94720 USA and with the Magnetic Resonance Systems Research Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA, 94305 USA

**Juan M. Santos,**

HeartVista, Inc., 715 Colorado Ave - Suite D, Palo Alto, CA 94306 USA

**John M. Pauly, and**

Magnetic Resonance Systems Research Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA, 94305 USA

**Greig C. Scott**

Magnetic Resonance Systems Research Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA, 94305 USA

### Abstract

MRI pulse sequence consoles typically employ closed proprietary hardware, software, and interfaces, making difficult any adaptation for innovative experimental technology. Yet MRI systems research is trending to higher channel count receivers, transmitters, gradient/shims, and unique interfaces for interventional applications. Customized console designs are now feasible for researchers with modern electronic components, but high data rates, synchronization, scalability, and cost present important challenges. Implementing large multi-channel MR systems with efficiency and flexibility requires a scalable modular architecture. With Medusa, we propose an open system architecture using the Universal Serial Bus (USB) for scalability, combined with distributed processing and buffering to address the high data rates and strict synchronization required by multi-channel MRI. Medusa uses a modular design concept based on digital synthesizer, receiver, and gradient blocks, in conjunction with fast programmable logic for sampling and synchronization. Medusa is a form of synthetic instrument, being reconfigurable for a variety of medical/scientific instrumentation needs. The Medusa distributed architecture, scalability, and data bandwidth limits are presented, and its flexibility is demonstrated in a variety of novel MRI applications.

### Index Terms

Medusa; magnetic resonance imaging (MRI); Universal Serial Bus (USB); direct digital synthesizer; digital receiver; radio-frequency instrumentation; console; synthetic instrumentation

### I. Introduction

Magnetic Resonance Imaging, through its ability to manipulate nuclear spin magnetization, provides exceptional versatility to image anatomy and physiological function. MRI pulse sequences require real-time control of RF and gradient systems, data acquisition, and processing, all coordinated by an intelligent console. Historically, MRI console design has

been the domain of major systems manufacturers, commanding significant engineering time and financial resources. These systems, targeted towards major medical and research markets, are typically closed with proprietary hardware, software, and interfaces, making difficult any expansion or adaptation to new techniques and experiments. FDA regulations on the design process, vendor marketing decisions, and reimbursement issues create political disincentives to rapid development. A confluence of these factors hinders innovations in interventional MRI, MR-guided therapies, and inhibits MR engineering education.

However, with advances in computer infrastructure, and digital and analog RF integrated circuits, investigators can now devise custom console designs targeted to their research needs [1]–[5]. Even so, MRI research systems are trending towards high channel-count receivers [6], [7], transmitter arrays [8], field cameras [9], [10] and even gradient/shim arrays [11], [12]. Meeting modern expectations for high channel counts, customized interfaces and real-time imaging performance poses a considerable design challenge. As one example, direct RF sampling/synthesis with modern wide-band analog-to-digital (ADC) and digital-to-analog (DAC) converters permit typically analog RF operations such as mixing, channel filtering, and modulation to be performed in the digital domain, a concept generally known as *software defined radio* (SDR) [13], [14]. While this has made the RF portion of MRI easier to realize, SDR approaches place additional pressure on the systems which must handle the new large flows of digital data. Implementing such systems with efficiency and flexibility requires a scalable modular system architecture.

Standard multi-core PC's can now handle user interface and image reconstruction duties in real-time [21], yet efficient data handling and real-time control remains a problem. Tightly integrated console hardware, such as PCI cards, provide high-throughput low-latency communication, but sacrifice scalability, cost effectiveness, and design time. Moreover, the PC then becomes directly involved with low-level MRI system operation - an untenable compromise given typical operating system stability and security. The handling of hard real time tasks with PC (or microcontroller) software may provide high flexibility, but imposes harsh restrictions on software design and inefficient use of available processing power and hardware resources. For example, one may expend considerable coding effort and even CPU time to ensure that a controller is simply ready to perform real-time operations. Bus technologies like USB or Ethernet are intrinsically scalable, but place the PC at arms length from system hardware, forcing real-time sampling and pulse sequence tasks to be handled by the local console hardware. Data must be buffered locally, then efficiently communicated to the PC with low latency.

At the 2005 ISMRM [15], we proposed a USB approach to MRI system design that could address these requirements (Fig. 1). This has now evolved into a full synthetic instrument prototype termed Medusa. Synthetic instrumentation refers to the use of reconfigurable building blocks to construct advanced instrumentation, while the name Medusa stems from the system's ever-expanding cables resembling a jellyfish or the head of the character from Greek mythology. In our design concept, we have investigated a highly scalable MR system based on digital synthesizer, digital receiver, and gradient blocks operating as a network of Universal Serial Bus peripherals. The efficacy of this approach will depend upon efficient architecture and design: distributed processing, efficient use of hardware, streamlined efficient data flow, and leveraging of commodity communication buses and components. Here, we present the architectural details of Medusa, its bandwidth and scalability limits using distributed USB data transport, its performance metrics for software-defined all-digital RF transmission and reception for parallel scalable MRI systems. Finally, various configurations are demonstrated, and its suitability for other novel instrumentation applications are discussed.

## II. Methods

### A. System Design Philosophy

The Medusa console aims to deliver the complete set of basic MRI console functions including multi-channel RF excitation and reception, gradient waveform generation, RF coil and amplifier gating, and an open software platform for console control, pulse sequencing, and experimentation. Furthermore, we target a modular architecture for scalability, and a performance level sufficient to enable modern fast imaging with RF and gradient bandwidths of at least 250K samples per second per channel, support for 100% duty cycle operation, and at least 16 bits of RF receiver dynamic range. To satisfy these requirements, hardware development was guided by three dominant priorities: use "digital" RF components to simplify and streamline the RF subsystems, employ a modular architecture with distributed processing for flexible system configuration and scalability to high channel counts, and leverage commodity PC peripheral interfaces for scalability, fast data transfer, interoperability, cost-efficiency, and a modicum of protection against hardware obsolescence. We chose Universal Serial Bus (USB) as our data transport for its support of up to 127 devices per bus, guaranteed data delivery, simple and inexpensive hardware interface options, and the ability to aggregate multiple slower USB devices through a higher speed hub.

**a) Real-Time Tasks**—To best explain our methodology, we define two timing domains, hard real-time and soft real-time. Hard real-time tasks require microsecond or nanosecond-level accuracy and encompass duties such as sample-by-sample RF and gradient pulse playback and echo acquisition, real-time waveform buffer management, and the timing of important pulse sequence parameters, eg. echo time (TE) and repetition time (TR). The high-speed, repetitive, and parallel nature of low-level timing, sampling, and data flow are not well suited to a sequential processor, but better served by dedicated logic, preferably in programmable form as in a Field Programmable Gate Array (FPGA) or Complex Programmable Logic Device (CPLD). By comparison, soft real-time tasks have more relaxed requirements on a millisecond timescale, yet still must be completed on time. These include starting and stopping of the scanner, updating the running pulse sequence each TR as needed, and retrieving the received data from acquisition buffers. The design of Medusa reflects the relative requirements of hard and soft real-time tasks.

**b) Concept Prototype**—The large scope of the proposed development followed from a proof-of-concept console (Fig. 1) first built to demonstrate basic hardware approaches. This prototype networked DDS and digital receiver evaluation boards, with analog gradient waveform generators using FTDI USB 1.1 modules, and a Belkin USB 2 hub employing Cypress hub USB ICs. A critical feature of the Cypress hub IC was the presence of four transaction translators (one per port) allowing full data rate translation to USB 2.0 High-Speed. Most other hubs shared a single transaction translator amongst all ports, that would have created a data-rate bottleneck. The successes and challenges of this first generation were particularly influential in the second generation design discussed here and shown in Figure 2.

### B. Medusa Bus Architecture

Medusa's bus architecture, illustrated in Figure 3, uses a tiered tree structure for configuration flexibility and scalability. At the core, a Medusa System Controller communicates with up to 16 modules over a 16-bit parallel data link called the Backbone Bus. Each module is responsible for a system task such as RF or gradients. The System Controller coordinates the data flow amongst its modules and forwards it via USB to a host PC. When the channel count or throughput limit of a single System Controller is reached,

additional controllers (each with up to 16 modules) can be connected via USB. The USB standard supports up to 127 devices per bus, yet in practice, the data throughput limit of USB or the host PC itself is reached first. When a single host PC is no longer able to accommodate the data or image reconstruction load, the bus design supports multiple host PCs connected through a high speed network (eg. Ethernet). This feature is supported in Medusa drivers but has not been tested since these limits have not yet been reached.

Data throughput concerns aside, there is no design limit to the number of Medusa modules that can be combined into a single system. Modules and controllers require only two signals to be shared system-wide in order to maintain synchronous operation: a reference clock (REFCLK) and a master sync (MSYNC). Medusa scales to the extent that these two signals can be distributed accurately.

### C. Hardware

**1) Logic Core**—A common logic core subsystem (Fig. 4) is incorporated into each Medusa module and is a key element in providing high performance, scalability, and flexibility with relatively few parts. Using an Altera MAX-II EPM1270 CPLD, a Cypress CY7C1061 2Mbyte high-speed SRAM, and an addressable backbone bus interface, the core performs in programmable hardware all the hard real-time module tasks such as sampling, real-time waveform buffer handling, and TR synchronization. The CPLD contains a configuration and identification register block, one or two Direct Memory Access (DMA) engines, a multi-port memory controller, a TR controller, and the backbone bus interface. In addition to these universal features, the logic core is customized with interfaces to serve the specific needs of each module type.

The intellectual-property (IP) cores in the CPLD were developed in Verilog HDL specifically for Medusa, with a focus on design efficiency and timing accuracy. The memory controller and DMA engines are partially-pipelined to increase data throughput per clock cycle, while the module-specific interface logic resynchronizes data transfers for precisely deterministic sample timing as needed in MRI. Many of the components are interconnected using a Wishbone [16] on-chip bus network, but with access and arbitration times tightly controlled for guaranteed real-time execution.

The logic core simultaneously performs several functions crucial to pulse sequence execution, such as timing, sampling, and data flow management. In parallel with pulse sequence execution, the logic core exposes a set of control and data registers through which the module hardware and logic core can be configured and waveform memory can be uploaded and downloaded. The multi-port SRAM memory controller enables concurrent access to the memory from the DMA engines and backbone bus, managing them by using time-division arbitration with access priorities and wait-states. Thus, reconfiguration and pulse sequence updates can be performed in real-time, even during sequence execution.

**2) System Controller**—The Medusa System Controller (Fig. 5) coordinates up to 16 individual modules through the backbone bus, providing configuration, control, and clocking, as well as the USB data link back to the host PC. A NXP/Philips LPC2214 60Mhz 32-bit ARM7 processor is the heart of the controller, accepting high level commands from the host PC and coordinating data transfers to and from the modules. As in the modules themselves, the System Controller includes a logic core alongside the ARM processor for timing, data flow, and synchronization. USB High-Speed (480Mbit/sec) support is implemented using a Cypress Semiconductor CY7C68013A FX2 USB peripheral interface. The internal architecture of the Cypress FX2 is particularly well suited for high throughput, combining an embedded 8051 8-bit processor to perform basic USB enumeration and handshaking, while all USB user data flows through a dedicated hardware-driven 16-bit



First-In First-Out (FIFO) buffer interface. The FIFO interface configures easily for glue-less communication with programmable logic or microcontrollers. The logic core, USB interface, and ARM processor communicate using a 16-bit memory bus capable of 60Mbytes/sec, a considerable improvement over the speed-limited I/O on the first generation Medusa. The system controller also uses a FTDI FT2232L USB Full Speed (12Mbit/sec) dual-interface IC with one interface dedicated to a serial debugging terminal, and the other to support the legacy first-generation USB connection and software.

For maximum flexibility, a programmable Cypress CY22150 PLL clock generator is used to synthesize the main system clock (SYSCLK) which drives local logic as well as the connected modules. The PLL can use an onboard 10MHz crystal, or an external reference clock (REFCLK) to synchronize with outside equipment or other System Controllers. Finally, each System Controller provides some local I/O, all of which can be controlled and interrogated via the host USB connection to facilitate interfacing to auxiliary systems and custom experimental hardware. An addressable multi-drop two-wire I2C bus interface is available for control of intelligent RF multiplexers and programmable-gain front-end amplifiers (PGAs). Also included are eight lines of combined logic I/O with 10-bit A/D for general-purpose control and sensing, and a 10-pin JTAG connector for in-system updating of the logic core CPLD firmware on any module.

**3) RF Module**—The Medusa RF Module (Fig. 6) combines a logic core with digital RF exciter and receiver subsystems. The exciter is based on the Analog Devices AD9854 Quadrature Direct Digital Synthesizer (DDS) and is capable of producing RF output modulated in frequency, phase, and amplitude. The AD9854 DDS includes a complex numerically-controlled oscillator (NCO) with 48-bit tuning word, 14-bit phase control, 12-bit amplitude control, and inverse-sinc interpolation to drive dual 12-bit I/Q output DACs. The DDS operates internally at 200–266 MHz,  $4\times$  the system clock (SYSCLK), and is thus capable of directly synthesizing RF from DC to 100MHz. To limit harmonics from the zero-order hold nature of the DACs, the DDS outputs are followed by a 7th-order elliptical lowpass filter with rolloff at 100MHz.

Similar to the exciter, the RF module receiver digitizes RF directly using an Analog Devices AD6644 14-bit ADC operating at the system clock rate (SYSCLK, 50–66MHz). The sample-and-hold bandwidth of the receiver ADC is 250MHz capturing a frequency range of several Nyquist bands. For carrier frequencies above  $\text{SYSCLK}/2$ , the sampling operation effects a down-conversion via aliasing. As a result, an input lowpass or bandpass filter narrower than  $\text{SYSCLK}/2$  must be used to prevent aliasing of unwanted signals and noise into the band of interest.

The received RF signal exits the ADC as a continuous data stream at 100–132 MB/s which is impractical for direct storage and remains spectrally sparse given the narrowband nature of MRI signals ( $<1\text{MHz}$  typical). To reduce the digital RF to MRI baseband, an Analog Devices AD6620 Digital Receiver (DR) is used to perform further downmixing and channel filtering numerically. The AD6620 receiver includes a complex NCO and multiplier with 32-bit frequency tuning word and 16-bit phase offset for downmixing. This is followed by two stages of cascaded integrator-comb (CIC) filtering and finally a user-defined 128-tap FIR filter/decimator. Baseband signal output is produced as 16-bit interleaved I/Q data.

For each sample clock within a pulse sequence transmit window, the logic core retrieves pulse data from the local waveform memory using DMA and writes the DDS exciter amplitude, phase, and frequency words. The new DDS settings take effect synchronously on the next sample clock. Likewise, during a pulse sequence receive window, receiver I/Q baseband data are captured temporarily by high-speed latches and then stored via DMA to

the local waveform memory. The logic core also provides external gating outputs to signal when transmit or receive operations are taking place. These outputs may be used to sequence RF power amplifiers, T/R switches, and Q-spoiling bias networks for coils.

Together, the exciter and receiver can operate on carriers anywhere in the range from DC to 100 MHz (0–2.4T proton magnetic resonance) with only a change of input filter. To maintain carrier phase synchronization between exciter and receiver as required by MRI, the two subsystems are both clocked from the Medusa system clock (SYSCLK) and the Tx/Rx digital NCO tuning words can be digitally matched. The Tx and Rx NCOs have different word lengths, so step sizes must be chosen to ensure that the same frequency is synthesized. The Tx/Rx phases can also be resynchronized on command. In high-speed ADC and DDS applications, clock phase jitter will influence the noise floor as timing uncertainty maps directly to NCO phase noise and ADC sampling error. An On Semiconductor MC100LVEL16 PECL driver and differential distribution network enables robust ADC and DDS clocks targeting sub-picosecond jitter levels.

**4) Gradient Module**—The Medusa Gradient Module (Fig. 7) implements the same logic core hardware as the RF Module but is instead programmed to handle the playback of gradient waveforms and gating signals. The logic core CPLD renders four channels of digital gradient waveform data as four synchronous high-speed serial links, nominally operating at 10Mbit/sec although signaling speed and data format are reconfigurable. Each of the four serial links consists of four signals (typically clock, data, load, and clear) which are converted to RS-422 100-ohm differential pairs by STMicroelectronics ST26C31 differential drivers. The data format, signaling speed, and purpose of each signal may be reconfigured as needed by the gradient system interface requirements. Differential links were selected to permit robust transmission to remote D/A converters or digitally-controlled gradient amplifiers as far as 50 meters away, as well as for immunity to cabinet-to-cabinet ground offsets. Output connectors are RJ-45 with signal pairing that allows the use of standard twisted-pair Ethernet cables. In addition to the gradient waveform outputs, two 5V and ten 3.3V logic-level gating outputs are provided for the coordination of other experimental hardware. The gating outputs present as a fifth channel and are arbitrarily programmable to a new value at each active sample point in the gradient sequence.

To convert the digital waveform streams for use with analog-input gradient amplifiers, we implemented Medusa DAC boards (Fig. 8) based on the high-precision Linear Technologies LTC1592 16-bit DAC. Avago HPCL-0738 opto-isolators convert the differential serial signals from the Gradient module to single-ended, while also providing 2kV galvanic isolation from the console. The LTC1592 DAC offers a software programmable full scale output range of  $\pm 2.5V$ ,  $\pm 5V$ , or  $\pm 10V$  helping to make the best use of the available 16-bit precision while matching gradient amplifier input requirements. The DAC is governed by a Linear LT1021-5 buried-zener 5V reference with low noise and drift ( $<1\text{ppm}$  0.1Hz to 10Hz, and 5ppm/degree Celsius). Two DAC module analog outputs are provided, a buffered single-ended output for monitoring, and a differential output to drive gradient amplifier inputs. Designed with experimental systems in mind, the DAC module also incorporates an optional output-integrating fault detector which aims to protect the gradient amplifiers and coils from thermal damage in case of gross programming error. The fault detector automatically latches the DAC output to zero when the integral of the output waveform exceeds a pre-set threshold. The circuit uses a lossy R-C integrator to avoid triggering on acceptable persistent low-level shim values.

Additional reference information for components is provided at <http://mrsrl.stanford.edu/~medusa>.

## D. Software Architecture

The host PC software employs a layered plug-in design to complement the modularity and flexibility of the Medusa hardware. A Medusa Server, with architecture shown in Figure 9, simplifies the management of multiple System Controllers and their attached modules by providing a unified software interface for all connected console hardware. The server layer incorporates low-level drivers for both USB High-Speed (Cypress FX2) and Full-Speed (FTDI) communication, and is specifically designed for easy extension to other interfaces in the future. Moreover, the Medusa system uses a standard command and data packet protocol for exchanges between the host PC and console hardware, which can be transported by virtually any communication method. For example, the Medusa protocol can be easily transported over Ethernet or RS-232 serial, in addition to USB.

Once the Medusa Server has identified and established connections to available console hardware, it exposes the functionality to high-level software through either an IP network layer, or a shared dynamically-linked library (DLL). User interfaces, pulse sequencers, and reconstruction engines may then use these interfaces to operate the console and recover acquired data. While the DLL interface provides the fastest possible access, the IP network offers exceptional versatility permitting control and reconstruction software to reside on separate, even remote machines from that running the Medusa Server. By virtue of being purely a communication layer using standard network sockets and leveraging cross-platform USB drivers, namely 'libusb' and FTD2XX, the Medusa Server can be compiled and run easily on any host machine under Microsoft Windows, Apple Mac OS X, or Linux operating systems.

While the Medusa Server provides the communication layer, it depends on higher-level software for pulse sequence design, control, execution, image reconstruction, display and storage. Writing such a suite of software is not a trivial matter. In an effort to harness existing work and provide a familiar powerful development environment, Medusa specifically supports control of the console directly from Mathworks Matlab, illustrated in Figure 10. A Matlab Extension (MEX) driver layer is implemented by the Medusa Server which allows complete access to all hardware features. A substantial library of Matlab functions and tools were written to simplify control of Medusa and facilitate development of pulse sequences. Executing a conventional non-interactive pulse sequence is as simple as supplying the pulse sequence data and parameters in a standardized Matlab structure and passing this to a provided execution script. At the conclusion of the scan, the structure is returned with acquired MRI data filled in. Even though Matlab is not a real-time environment, it can run Medusa successfully, even interactively, because the real-time burden is off-loaded to Medusa hardware.

## E. Sequence Timing & Coordination

Pulse sequence execution typically begins with identification and configuration. The host PC software opens a connection to the Medusa Server and requests a list of all available hardware, for example, the number of RF and gradient modules and channels. The pulse sequence (Fig. 11) is then defined by a set of data blocks, and their associated timing parameters within a TR interval. The data blocks and parameters are transmitted to the local RAM of each module. At the module level, pulse sequence execution is handled by each logic core independently. A master synchronization signal, MSYNC, triggers the beginning of TR, which is then executed to completion on each module. At the rising edge of each MSYNC signal, the internal TR controller registers the beginning of a TR and begins counting sample clocks (SMPCLK). The sample counter is continuously monitored and compared against the start time of the next pulse sequence element, eg. gradient waveform, RF pulse, or acquisition window. When the TR controller determines that a pulse sequence

element is active or "in window", it triggers the DMA engine on each SMPCLK to move data between the waveform SRAM and module devices. For playback, RF or Gradient waveform data is retrieved from waveform memory and stored into the DDS or gradient shift registers. For RF acquisition, data is moved from the receiver I/Q data latches and stored back into waveform memory. When the last pulse sequence element in the TR is executed, the TR controller returns to idle and waits for the next TR start signal (MSYNC).

Medusa can execute a sequence in two operating modes, streaming and batch, depending on the pulse sequence requirements and host PC limitations. Streaming mode is preferred because it maintains continuous execution. Here, the host machine must stream data blocks into each module in advance of MSYNC, while received data and status messages are read out at the completion of each TR. Real-time modification of the scan is possible. In practice, the host must keep a minimum of two TRs loaded (stay two TRs ahead of scan progress). Should the host PC fall behind, the scan will gracefully stop at the last TR that was loaded on-time. Streaming, even at high rate, can be successfully executed with a relatively modest host machine, eg. Intel Core2 Duo at 1.8GHz.

In batch mode, the complete pulse sequence is pre-loaded to Medusa in advance of the scan. The waveforms and parameters of most 2D scans can fit completely in the RF and gradient module memories. Once started, the full scan is executed without requiring communication from the host PC at all. Acquired MR data are downloaded to the host after the scan completes. Batch mode does not support real-time modification of the scan, but is useful for ultra-fast TR, very high data rates, or a slow host PC. Regardless of operating mode, any scan can be paused or halted at any time if user intervention is required.

### III. Results

Initial Medusa imaging tests were performed on the Stanford Pre-Polarized MRI (PMRI) scanner [17]–[20]. PMRI employs an inhomogeneous pulsed resistive electromagnet for polarization, followed by a lower-field uniform electromagnet during signal readout. The Stanford-built scanner served as an excellent testbed with ready access to RF, gradient, and coil control signals. A Matlab-based scan execution environment, as previously illustrated in Figure 10, was developed to drive the scanner along with gradient echo, spin-echo, and fast spin-echo sequences. The PMRI magnet and an early Medusa imaging result are shown in Figure 12.

Medusa was subsequently configured to operate a GE Signa Excite 1.5T scanner, adapting easily to 1.5T (64MHz) with only a change of RF input filters and frequency settings. However, the proprietary nature of some of the signal interfaces on the commercial scanner meant that an in-house RF chain and coils had to be built. Success with simple Matlab-controlled 2DFT scans eventually led to more advanced work such as spiral-based real-time cardiac imaging with Medusa driven by the RT-Hawk real-time scanning software suite [21], results shown in Figure 13. This spawned a Medusa configuration with 8-channel RF transmit/receive and dual System Controllers designed specifically to satisfy the high performance demands of real-time parallel imaging (Figure 14).

Being essentially a generic multi-channel RF instrument, the use of Medusa is not limited to MR imaging. RF network and spectrum analysis functions are easily implemented in the same Matlab programming environment used for scanning. As an example, a 2-ch Medusa console combined with a directional coupler has been used to monitor impedance in a chest surface coil showing variability in-sync with cardiac and respiratory function (Figure 15).

Tables I and II list performance figures for Medusa hardware. Experimentally achieved throughput using Medusa's Full-Speed and High-Speed USB interfaces did not reach

theoretically possible rates (Table I). The transaction, protocol, and bit-stuffing overhead of USB consumed as much as 33% of total bus bandwidth. Furthermore, we observed that usable throughput on USB can vary significantly with host operating system, USB host controller, USB device interface, data packet size, and transfer method. Using large packet sizes, bulk transfer mode, and a device interface with hardware acceleration (DMA) generally yielded the best rates. Failure to achieve theoretically-possible rates is common, even with a modern host PC and high-performance commercial USB products.

The maximum Tx/Rx baseband sample rate in the RF module is limited largely by data transfer speeds between local waveform memory and the DDS and digital receiver. Operating independently, the exciter and receiver can achieve sample rates of 1.8 and 3.2 million complex samples per second, respectively. Yet when transmitting and receiving simultaneously (eg. loopback or calibration), the maximum combined sample rate is 1.1 Msps, which is still in excess of typical MRI baseband requirements of 8–500 Ksps.

In RF performance, the Medusa RF receiver achieves a signal over noise floor of 89 dB (15.8 bits) at 250Ksps baseband rate, a figure made possible in part by process gain from decimation and filtering of the high-rate ADC. The receiver's input-referenced 3rd-order intercept point (IIP3) of 19.7dBm was determined using two-tone tests at  $\pm 1$  kHz and  $\pm 10$  kHz. The RF transmitter produces a spur-free dynamic range (SFDR) of 78 dBc. All measurements were made at a center frequency of 63.9MHz (1.5T). The first version of the RF module was carefully designed on a two-layer PCB for simplicity and economy, yet a recent update derived a mild 0.5dB improvement in RF dynamic range by moving to a four-layer design with full internal power and ground planes. A four-layer PCB has the added advantage of better thermal conductivity for heatsinking of the high-dissipation DDS and ADC chips.

The maximum achievable gradient sample rate when driving Medusa DAC modules is 375K samples per second per channel, the point at which the bandwidth of the 10Mbit serial links is saturated by 24-bit DAC words (8 bits control + 16 bits data). The Medusa Gradient Module is intrinsically capable of 1.2M samples per second when the connected hardware permits higher serial bit rates or a more compact data format. Typical MRI gradient amplifiers are limited to a bandwidth measured in tens of kilohertz, with sample rates of 250Ksps commonly used merely to offer additional temporal resolution.

## IV. Discussion

While Medusa has clearly been successful as a flexible, scalable platform for MRI imaging, it is also a study in tradeoffs. The goal of delivering high performance while making efficient use of resources at every layer in the architecture has been addressed partly through initial design, and partly through iteration.

The proof-of-concept prototype highlighted several design pitfalls including insufficient RF and Gradient sample rates due to underpowered microcontrollers, undersized memory buffers, and the low throughput of USB Full Speed (12Mbit/sec). The loose hardware integration of development kits also made multi-channel scale-up difficult. The second-generation Medusa system addressed many of these problems.

By implementing a common logic core to handle hard real-time tasks common to all Medusa modules, we simultaneously absolved the Medusa Controller and host PC from dealing with precision timing while also simplifying system design, lowering cost, and promoting system modularity and expandability. Hard real-time tasks are contained and remain local to the modules. New or upgraded modules added to the Medusa system can build upon the same logic core (or implement a new compliant one) along with the circuits to support new



functionality. Already, the Medusa Gradient module hardware has been easily reconfigured in firmware to drive several different gradient systems including GE and Varian digitally-driven amplifiers, as well as in-house built RF vector multiplier boards in place of the standard Medusa DAC boards.

Likewise, the use of digital RF components yields some key design advantages and simplifications. Performing down-mixing and channel filtering numerically ensures that multi-channel systems have perfectly matched response without drift due to temperature or component aging. Digitizing RF early also means fewer analog parts in the receive path that must be scrutinized for linearity, dynamic range, and IP3 performance. Direct digital conversion of RF eliminates the need to distribute a local oscillator (LO) common in analog implementations. However, digital systems do not escape the need for quality phase and frequency references. The low phase noise requirement on analog LO signals have an exact corollary in the clock jitter specification levied on high-speed digital sampling clocks. In both cases, a poor quality reference will lead to unwanted spectral broadening and reduced SNR in the baseband, e.g. clock jitter of just 300 femtoseconds is equivalent to the loss of the least-significant bit on Medusa's 14-bit ADC when acquiring a 64MHz signal.

We see from results (Table II) that the Medusa RF receiver closely tracks predicted process gain from oversampling. For sinusoids with random noise, we can expect oversampling by factor  $N$  to yield  $\log_2(N)/2$  additional bits of dynamic range. At lower imaging bandwidths, the receiver's noise-free dynamic range already exceeds the range provided by the 16-bit output data path. A future upgrade of the receiver ADC in speed or bit-depth would need to be accompanied by a 20 or 24-bit digital path to capture the wide dynamic range.

The performance of digital RF components such as DDS and high-speed ADCs are advancing rapidly. Already, the devices used in the Medusa RF module are old, superseded by newer ICs with a broader frequency range, faster sampling and update rates, and higher dynamic range. Far from making Medusa obsolete, this merely reinforces the importance of the modular system architecture which permits the seamless upgrade of components as technology improves and needs grow.

Building the Medusa System Controller separately from the RF and Gradient modules succeeded in minimizing hardware and allowing easy expansion and upgrade of the modules, but this came at the price of introducing a potential data bottleneck in the Medusa backbone bus. Future Medusa designs may need to balance the increasing data demands of high channel counts against ultimate modularity. For example, a single RF Module hosting 4 or 8 channels could produce enough data that it alone would fill a USB 2.0 data link. For such a design, the RF module should integrate System Controller functionality with a dedicated data link.

Moreover, there remains a question of whether USB is the best bus architecture for Medusa. USB is ubiquitous, fast, cost-efficient, and intrinsically supports multiple devices per bus. By virtue of being entirely host managed, USB does not suffer from bus collisions and can provide guaranteed data delivery. While Medusa is now able to saturate a single USB 2.0 480Mb/s link under real-time multi-channel operation, the introduction of USB 3.0 at 4.8Gb/s may provide a compelling upgrade path with backwards compatibility. Multiple Medusa systems working in tandem have already been demonstrated as a method for increasing channel counts. A USB 3.0 hub can aggregate the data from at least ten Medusa systems each with 4 to 8 RF channels, permitting an order of magnitude increase in channel count without any hardware redesign.

Nonetheless, USB is not without flaws. Protocol overhead is 10–15 percent, and host-managed bus scheduling often reduces usable bandwidth an additional 10–25 percent.

Commonly achievable user data rates are only 60% of the raw bus rate, with relatively long round-trip latencies of 300 $\mu$ s to 3ms. The lack of support for broadcast, multicast, and peer-to-peer device communication presents an efficiency and logistics problem when distributing global information amongst a distributed system like the Medusa architecture. Consider for example the need to distribute a time-critical pulse sequence change due to cardiac trigger. On USB, the message must be sent independently to each Medusa Controller, introducing delay and time skew.

The IEEE 1394a/b bus standard (Firewire 400/800 Mbps) was considered early on as an alternative bus transport to USB. Even when running at a bitrate slower than USB 2.0, Firewire achieves higher data throughput with up to 95% bus utilization and guaranteed bandwidth. Yet Firewire began as a closed standard and despite clear technical advantages still has limited market share compared to USB. As a result, there are far fewer components, tools, and resources for Firewire device development, making implementation challenging.

Gigabit Ethernet, although more complex to implement than USB 2.0, presents some advantages for future Medusa hardware generations. Like USB, multiple Ethernet links can be aggregated into a high-bandwidth backbone using network switches. Ethernet network cards and switches are commonly engineered and specified for high network load and are readily available commercially at low cost. Most products can operate to over 90% network utilization. Tests of modern consumer-level PC hardware with built-in gigabit Ethernet achieved 850Mb/s user data throughput when transferring to RAM, and round-trip latencies of less than 50 $\mu$ s, far better than USB 2.0 in both respects. Ethernet also supports broadcast and peer-to-peer communication for more flexible information flow, and significantly longer cable lengths as compared to USB or Firewire.

Medusa may be used as a stand-alone console for novel imaging systems, yet it also integrates easily with most commercial scanners to augment functionality and the number of RF and gradient channels. Only a 10MHz clock reference (REFCLK) and Start-of-TR logic signal (MSYNC) are required to keep Medusa synchronized for tandem operation with a host scanner. This same MSYNC trigger mechanism can also be used to precisely discipline Medusa's pulse sequence execution from an external device such as a cardiac or respiratory monitoring unit. In both cases, Medusa's inputs and outputs are synchronized to the host to within 20 nanoseconds, and can then be used for generic channel expansion, or to monitor or control auxiliary hardware such as for guidewire current monitoring or RF ablation.

When Medusa interacts with live subjects, exceptional care must be taken to ensure that SAR and other physiological safety limits are obeyed. As with commercial scanners, combining pulse sequence safety modeling, monitoring, and hardware interlocks is advisable. One approach to enhancing MR safety in an experimental setting is to use amplifiers which are either physically incapable of exceeding physiological limits, or are failsafe protected against it at the hardware level. For example, we may use 1 kW RF amplifiers yet with time-average RF power limited to 100 watts by the DC power supply.

Thus far, eleven second-generation Medusa systems have been constructed. Demonstrating flexibility and versatility in application, Medusa has played a key role in a variety of imaging experiments from Pre-Polarized MRI, Magnetic Particle Imaging [22], and Overhauser-enhanced MRI [31]. Medusa has also enabled Parallel Transmit research in predistortion methods [24] and Cartesian Feedback for RF power amplifier linearization [27], [28], and impedance characterization of RF amplifiers [23] and coils [25]. Finally, Medusa is used for interventional research addressing therapeutic RF ablation [30] as well as MR guidewire safety through reverse polarization imaging [26] and active current cancellation [29].

## V. Conclusion

MRI systems research has trended to higher channel counts in receiver, transmitter and gradient/shim subsystems. The rapid prototyping and implementation of novel MRI hardware concepts demands a flexible, scalable, open architecture. We have demonstrated Medusa, a highly scalable MR console architecture based on digital synthesizer, digital receiver and gradient building blocks networked by a USB 2.0 transport layer. The Medusa architecture is extensively reconfigurable to synthesize a variety of medical/scientific instruments. To date, eleven Medusa consoles have been built in various configurations to serve diverse research applications including prepolarized and real-time MRI, parallel transmit system development, and magnetic particle imaging.

## Acknowledgments

The authors would like to thank William Overall for his invaluable and continued support of Medusa with the Spin-Bench and RT-Hawk software packages, thereby making rapid sequence development and real-time scanning a reality. We also wish to thank Nate Matter, Sharon Ungersma, and Ross Venook for their kind support and patient assistance in the early testing of Medusa on the Stanford PMRI Scanner.

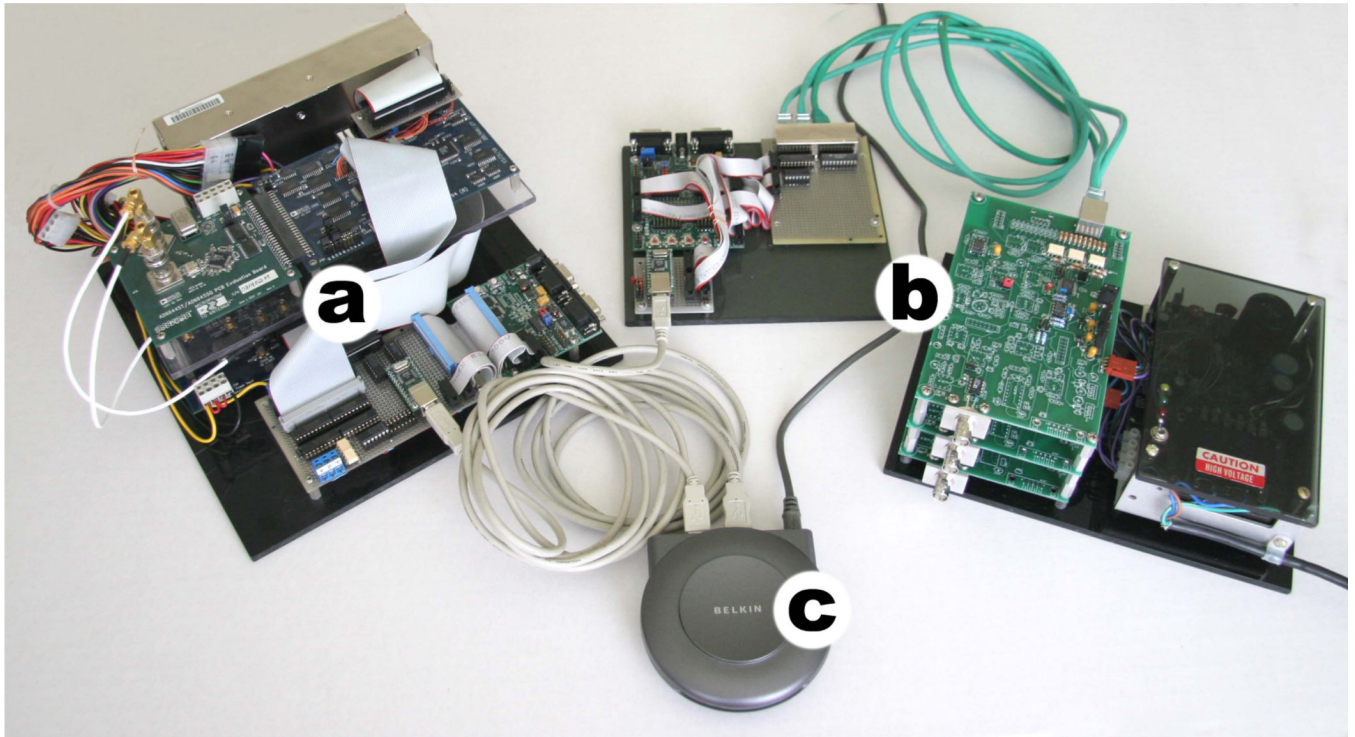
## References

1. Michal, Carl A.; Broughton, Kesten; Hansen, Elsa. A high performance digital receiver for home-built nuclear magnetic resonance spectrometers. *Review of Scientific Instruments*. 2002 Feb; 73(2): 453–458.
2. Takeda, Kazuyuki. A highly integrated FPGA-based nuclear magnetic resonance spectrometer. *Review of Scientific Instruments*. 2007 Mar. 78(3) 033103.
3. Mao, Wenping; Bao, Qingjia; Yang, Liang; Chen, Yiqun; Liu, Chaoyang; Qiu, Jianqing; Ye, Chaohui. A modularized pulse programmer for NMR spectroscopy. *Review of Scientific Instruments*. 2011 Mar. 22(2) 025901.
4. Wright, Steven M.; Brown, David G.; Porter, Jay R.; Spence, David C.; Esparza, Emilio; Cole, David C.; Russell Huson, F. A desktop magnetic resonance imaging system. *Magnetic Resonance Materials in Physics, Biology and Medicine*. 2002 Jan; 13(3):177–185.
5. Bodurka, Jerzy; Ledden, Patrick J.; van Gelderen, Peter; Chu, Renxin; de Zwart, Jacco A.; Morris, Doug; Duyn, Jeff H. Scalable Multi-channel MRI Data Acquisition System. *Magnetic Resonance in Medicine*. 2004 Jan; 51(1):165–171. [PubMed: 14705057]
6. McDougall, Mary P.; Wright, Steven M. 64-channel array coil for single echo acquisition magnetic resonance imaging. *Magnetic Resonance in Medicine*. 2005 Aug; 54(2):386–392. [PubMed: 16032696]
7. Schmitt, Melanie; Potthast, Andreas; Sosnovik, David E.; Polimeni, Jonathan R.; Wiggins, Graham C.; Triantafyllou, Christina; Wald, Lawrence L. A 128-Channel Receive-Only Cardiac Coil for Highly Accelerated Cardiac MRI at 3 Tesla. *Magnetic Resonance in Medicine*. 2008 Jun; 59(6): 1431–1439. [PubMed: 18506789]
8. Zhu, Yudong. Parallel Excitation With an Array of Transmit Coils. *Magnetic Resonance in Medicine*. 2004 Apr; 51(4):775–784. [PubMed: 15065251]
9. Barmet, C.; Wilm, BJ.; Pavan, M.; Katsikatsos, G.; Keupp, J.; Mens, G.; Pruessmann, KP. Concurrent higher-order field monitoring for routine head MRI: an integrated heteronuclear setup. *Proceedings of ISMRM 18th Scientific Meeting in Stockholm; Sweden*. 2010. no.216
10. Sipila, P.; Wachutka, G.; Wiesinger, F. Coherent excitation scheme to operate pulsed NMR probes for real-time magnetic field monitoring. *Proceedings of ISMRM 18th Scientific Meeting in Stockholm; Sweden*. 2010. no.217
11. Hennig, Jürgen; Welz, Anna M.; Schultz, Gerrit; Korvink, Jan; Liu, Zhenyu; Speck, Oliver; Zaitsev, Maxim. Parallel imaging in non-bijective, curvilinear magnetic field gradients: a concept study. *Magnetic Resonance Materials in Physics, Biology and Medicine*. 2008 Mar; 21(1–2):5–14.

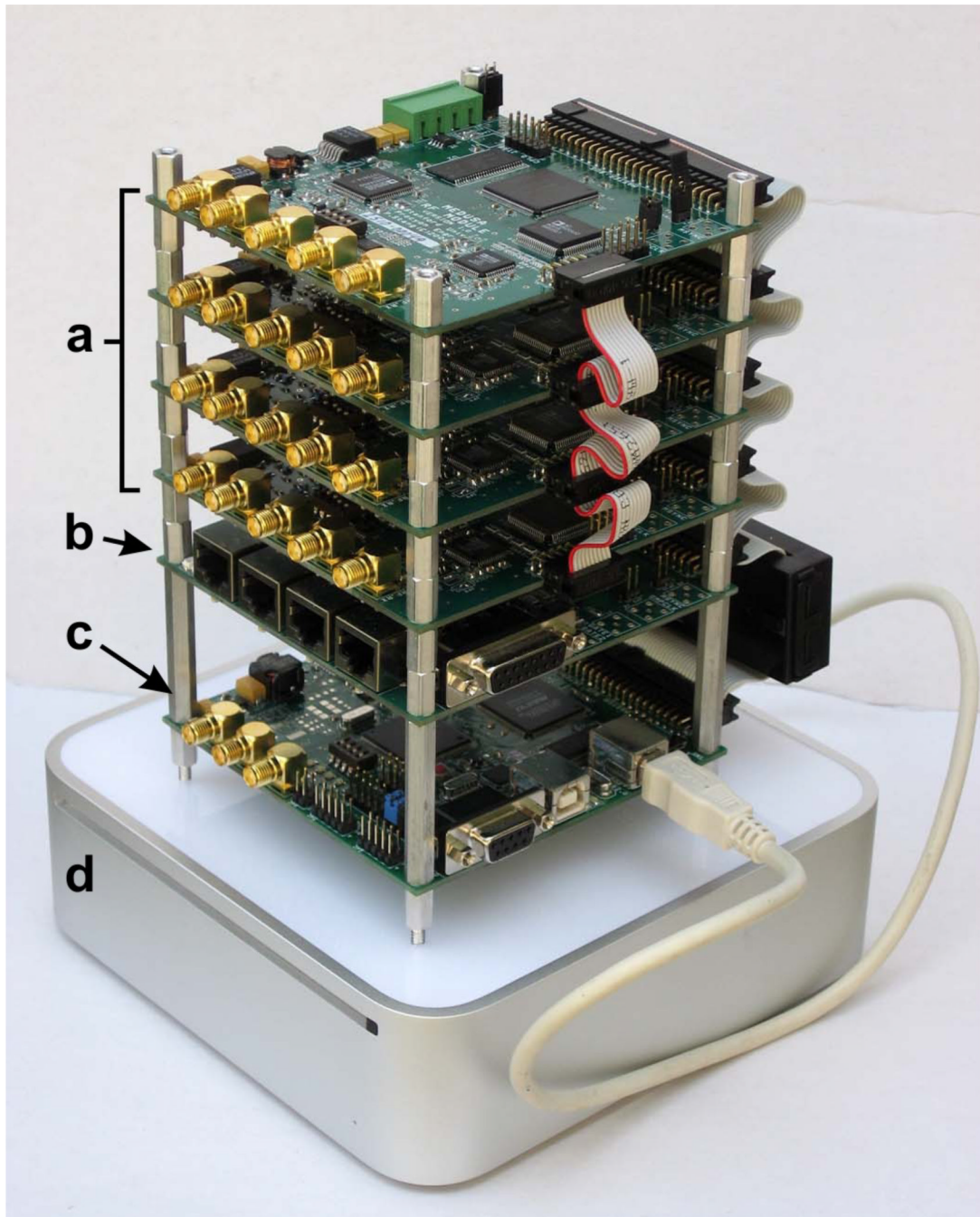
12. Schultz, Gerrit; Ullmann, Peter; Lehr, Heinrich; Welz, Anna M.; Hennig, Jürgen; Zaitsev, Maxim. Reconstruction of MRI data encoded with arbitrarily shaped, curvilinear, nonbijective magnetic fields. *Magnetic Resonance in Medicine*. 2010 Nov; 64(5):1390–1403. [PubMed: 20848635]
13. Kennington, Peter B. *RF and Baseband Techniques for Software Defined Radio*. Boston, Massachusetts: Artech House; 2005.
14. Mitola, Joseph. *Software Radio Architecture*. New York: John Wiley and Sons Inc; 2000.
15. Stang, Pascal P.; Conolly, Steven M.; Pauly, John M.; Scott, Greig C. A USB Approach to Scalable Design of MRI systems. *Proceedings of ISMRM 13th Scientific Meeting in Miami Beach; Florida*. 2005. no.861
16. WISHBONE System-on-Chip (SoC) Interconnection Architecture for Portable IP Cores Rev B.3. [http://opencores.org/downloads/wbspec\\_b3.pdf](http://opencores.org/downloads/wbspec_b3.pdf)
17. Ungersma, Sharon E.; Chronik, Blaine A.; Scott, Greig C.; Albert, Macovski; Conolly, Steven M. Shim coil design using a linear programming algorithm. *Magnetic Resonance in Medicine*. 2004 Aug; 52(3):619–627. [PubMed: 15334583]
18. Matter, Nathaniel I.; Scott, Greig C.; Venook, Ross D.; Ungersma, Sharon E.; Grafendorfer, Thomas; Macovski, Albert; Conolly, Steven M. Three-dimensional prepolarized magnetic resonance imaging using rapid acquisition with relaxation enhancement. *Magnetic Resonance in Medicine*. 2006; 56(5):1085–1095. [PubMed: 17029228]
19. Ungersma, Sharon E.; Matter, Nathaniel I.; Hardy, Jonathan W.; Venook, Ross D.; Macovski, Albert; Conolly, Steven M.; Scott, Greig C. Magnetic resonance imaging with T1 dispersion contrast. *Magnetic Resonance in Medicine*. 2006; 55(6):1362–1371.
20. Venook, Ross; Matter, Nathaniel; Ramachandran, Meena; Ungersma, Sharon; Gold, Garry; Giori, Nicholas; Macovski, Albert; Scott, Greig; Conolly, Steven. Prepolarized MRI around metal orthopedic implants. *Magnetic Resonance in Medicine*. 2006; 56:177–286. [PubMed: 16724303]
21. Santos, Juan; Wright, Graham A.; Pauly, John M. Flexible real-time magnetic resonance imaging framework. *Proceedings of IEEE Engineering in Medicine and Biology Society*. 2004; 2:1048–1051.
22. Goodwill, Patrick W.; Scott, Greig C.; Stang, Pascal P.; Conolly, Steven M. Narrowband Magnetic Particle Imaging. *IEEE Transactions on Medical Imaging*. 2009; 28(8):1231–1237. [PubMed: 19211340]
23. Scott, Greig C.; Stang, Pascal P.; Kerr, Adam; Pauly, John M. A Load Pull/Hot S22 Analyzer for Transmit Array Amplifiers. *Proceedings of ISMRM 17th Scientific Meeting; Honolulu, Hawaii*. 2009. no.3025
24. Stang, Pascal P.; Pauly, John M.; Scott, Greig C. Vector Iterative Pre-Distortion: An Auto-calibration Method for Transmit Arrays. *Proceedings of ISMRM 18th Scientific Meeting in Honolulu; Hawaii*. 2009. no.491
25. Stang, Pascal P.; Pauly, John M.; Scott, Greig C. A Versatile In-Line Sensor For Power Monitoring and Calibration of Transmit Arrays. *Proceedings of ISMRM 18th Scientific Meeting in Honolulu; Hawaii*. 2009. no.494
26. Overall, William R.; Pauly, John M.; Stang, Pascal P.; Scott, Greig C. Ensuring safety of implanted devices under MRI using reversed RF polarization. *Magnetic Resonance in Medicine*. 2010; 64(3): 823–833. [PubMed: 20593374]
27. Zanchi, Marta G.; Stang, Pascal P.; Kerr, Adam A.; Pauly, John M.; Scott, Greig C. Frequency-Offset Cartesian Feedback for MRI Power Amplifier Linearization. *IEEE Transactions on Medical Imaging*. 2011; 30(2):512–522. [PubMed: 20959264]
28. Zanchi, Marta G.; Stang, Pascal P.; Pauly, John M.; Scott, Greig C. Tuning the Output Impedance of RF Power Amplifiers with Frequency-Offset Cartesian Feedback. *Proceedings of ISMRM 18th Scientific Meeting; Stockholm, Sweden*. 2010.
29. Etezadi-Amoli, Maryam; Stang, Pascal; Zanchi, Marta G.; Pauly, John M.; Scott, Greig C.; Kerr, Adam B. Controlling induced currents in guidewires using parallel transmit. *Proceedings of ISMRM 18th Scientific Meeting; Stockholm, Sweden*. 2010.
30. Shultz, Kim; Stang, Pascal P.; Pauly, John M.; Scott, Greig C. Feasibility of RF Ablation at the Larmor Frequency for RF Field Visualization. *Proceedings of ISMRM 19th Scientific Meeting; Montreal, Canada*. 2011. no.1760

31. Ahn, Kang-Hyun; Scott, Greig; Stang, Pascal; Conolly, Steve; Hristov, Dimitre. Multiparametric Imaging of Tumor Oxygenation, Redox Status, and Anatomical Structure Using Overhauser-Enhanced MRI Prepolarized MRI System. *Magnetic Resonance in Medicine*. 2011; 65(5):1416–1422. [PubMed: 21500268]

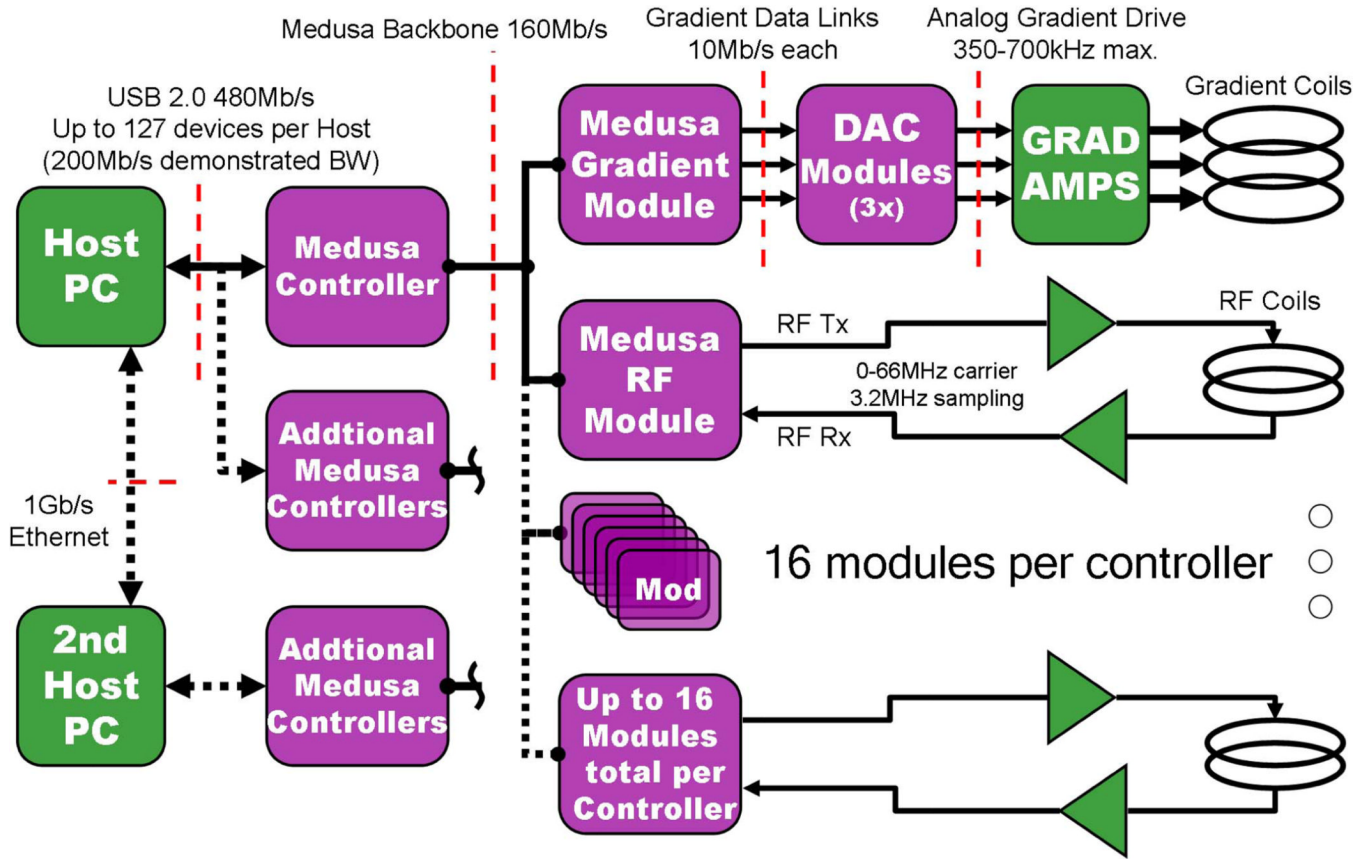




**Fig. 1.** The first proof-of-concept Medusa console hardware with (a) prototype RF transceiver module, (b) prototype gradient control module, and (c) a USB hub with multiple transaction translators.

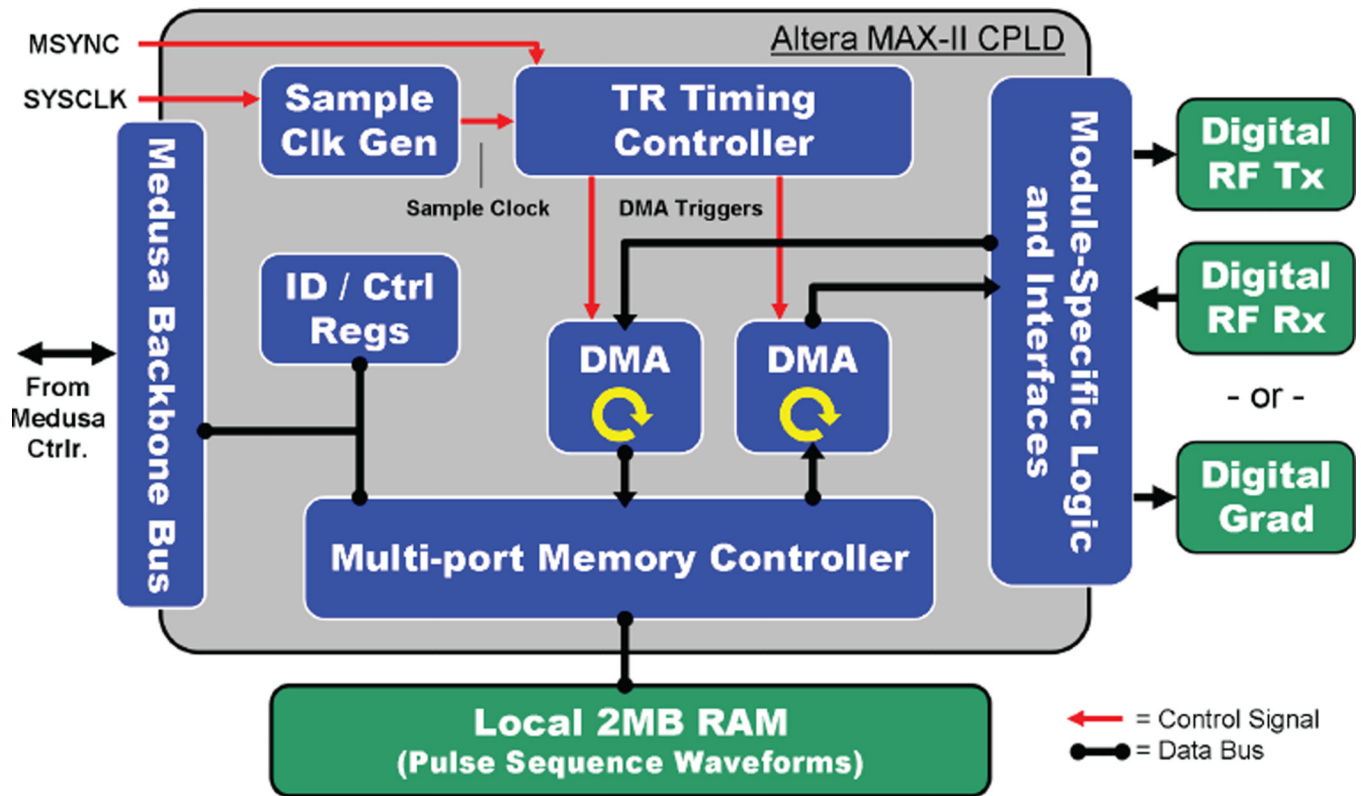


**Fig. 2.** A second-generation Medusa Console illustrating the concept of modular design. The console shown has (a) four RF transmit and receive channels, (b) a four-channel gradient control module, and (c) a system controller. The console is connected by USB to (d) a common host computer shown below the Medusa stack for size reference.



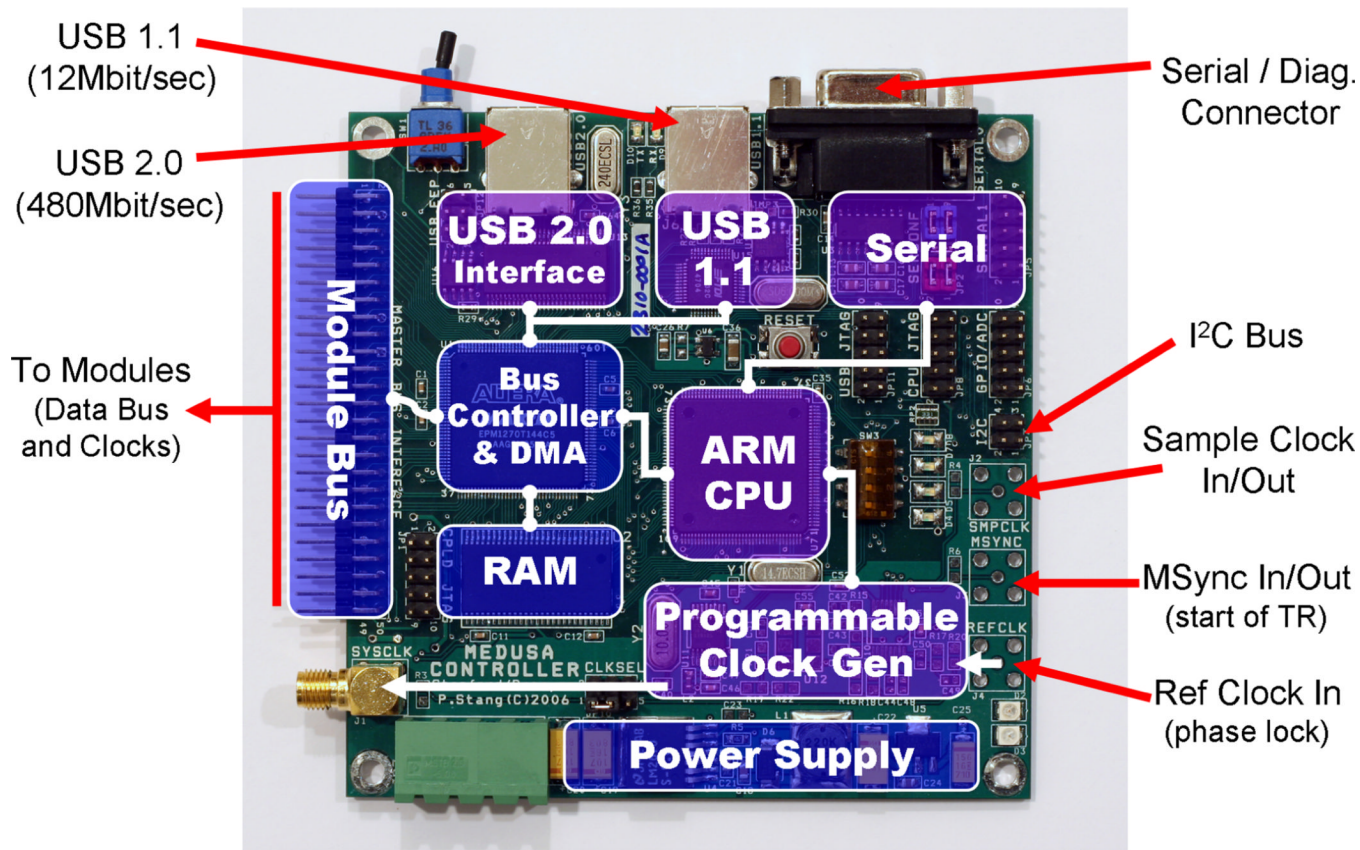
**Fig. 3.** A basic Medusa system is built from one or more RF and Gradient Modules that are driven by a System Controller, in turn connected to a host PC. As application demands grow, the system can be expanded (dashed lines) up to 16 modules per controller. For large channel counts or high-bandwidth applications, additional System Controllers and hosts may be used. The limit of scalability depends only on accurate clock and sync signal distribution.





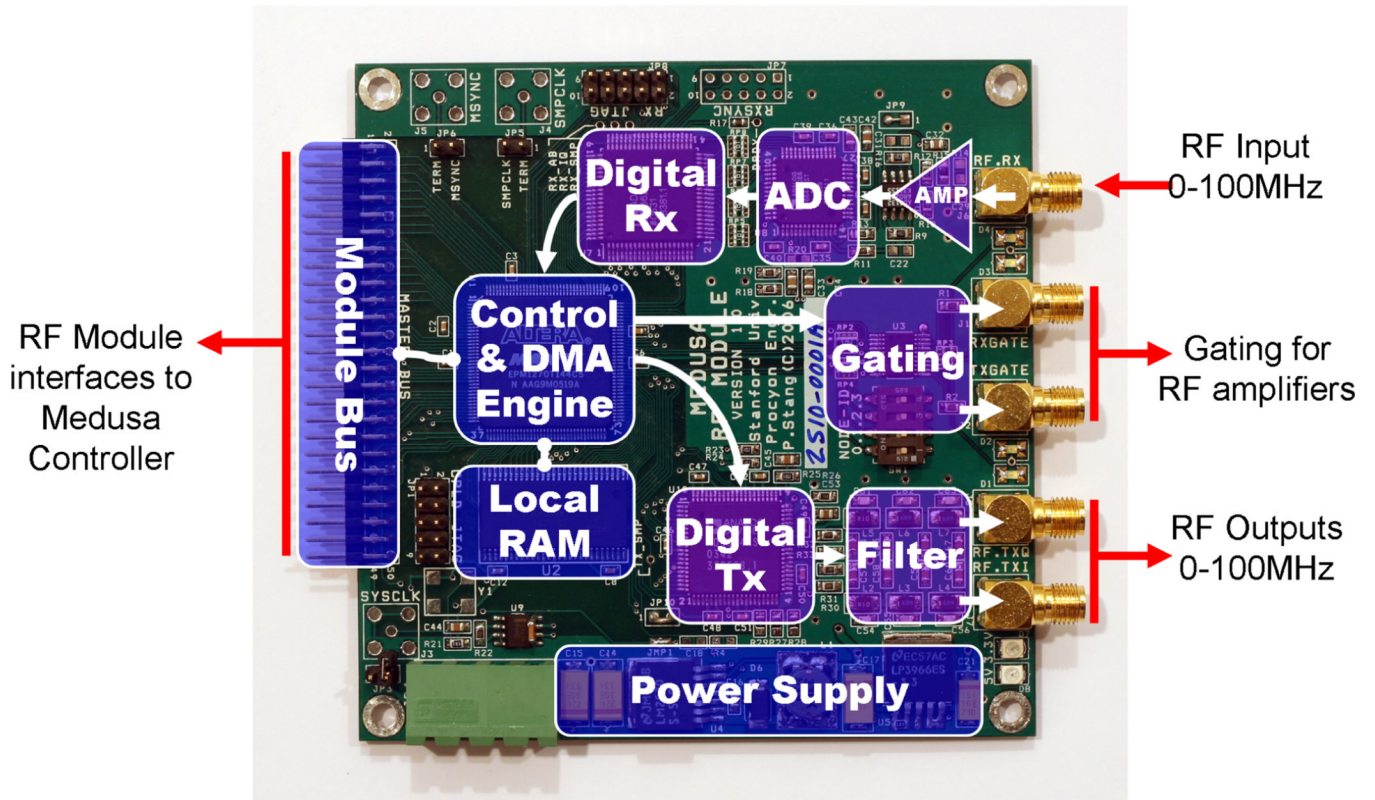
**Fig. 4.**

A common logic core is replicated in each module CPLD to manage timing, data sampling, buffering, and sequence flow control. Many operations must execute simultaneously under tight timing constraints making programmable logic especially well suited to the task.

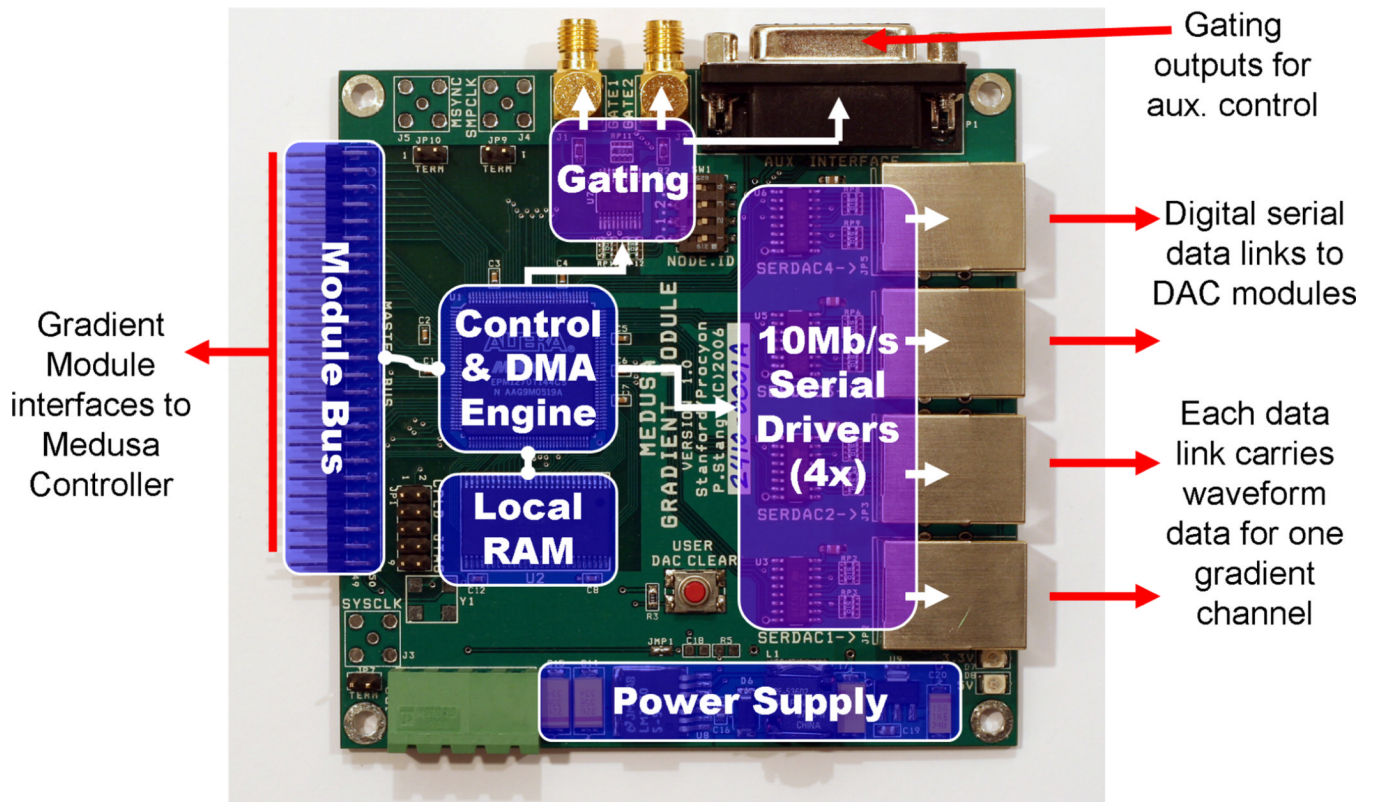


**Fig. 5.** The Medusa System Controller coordinates up to 16 individual modules through a backbone bus, providing configuration, control, clocking, and a USB data link to a host PC. Multiple Medusa Controllers may be used together as a single system by sharing REFCLK and MSYNC signals between them. The Master Controller sets overall system timing by controlling MSYNC.

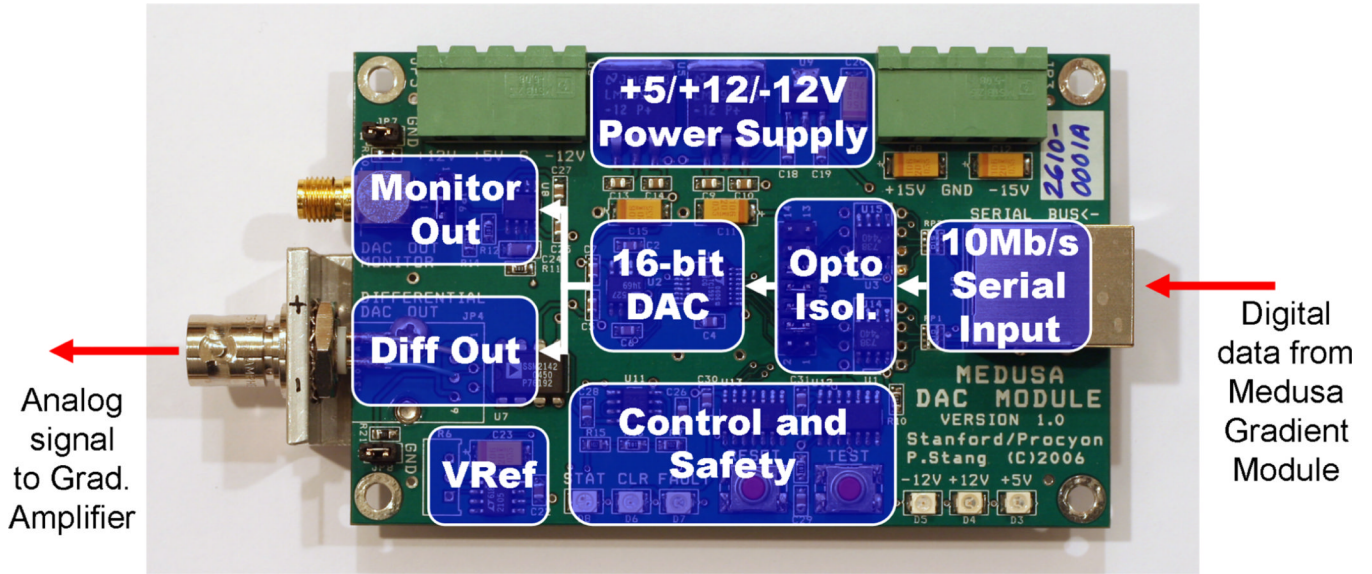




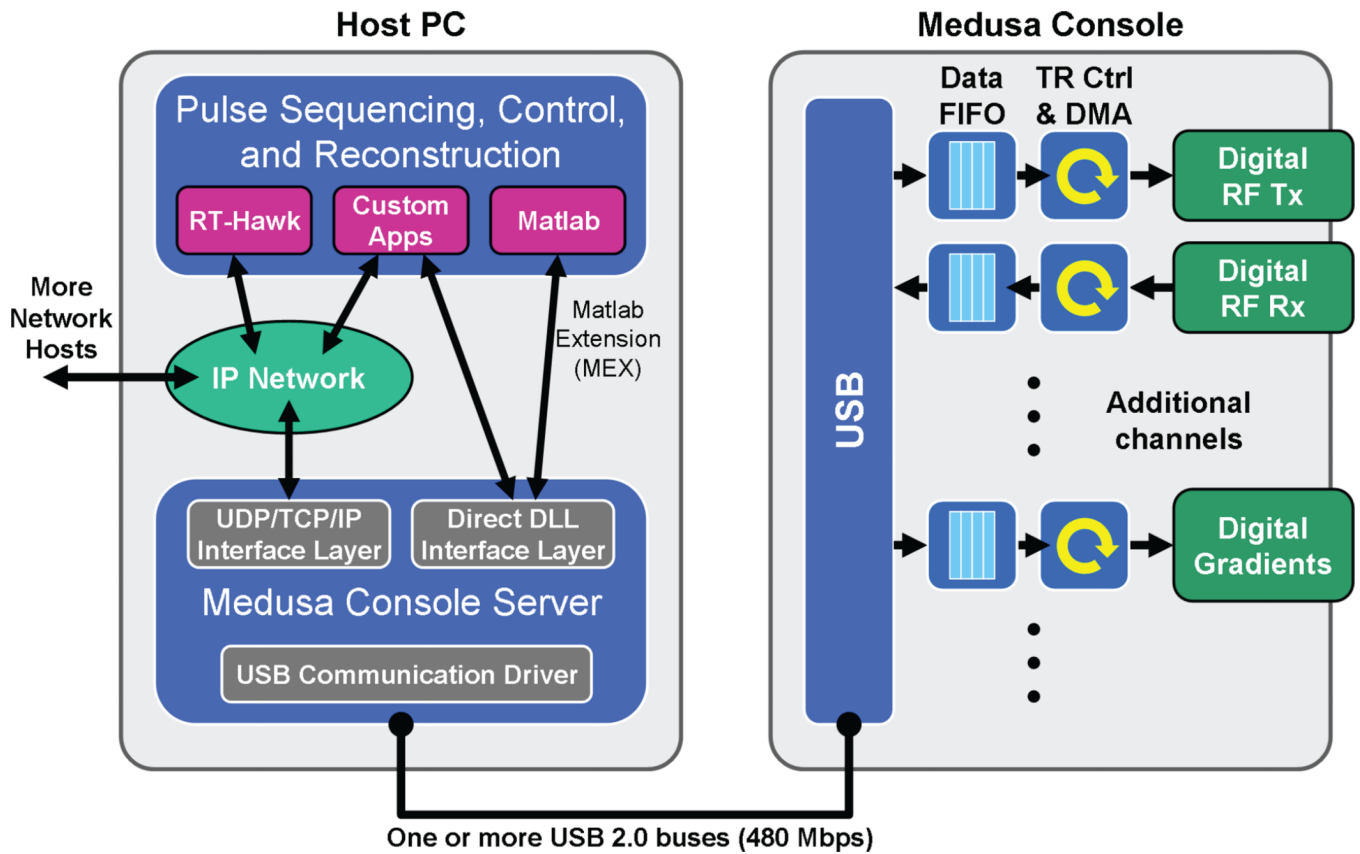
**Fig. 6.** The Medusa RF Module employs direct-conversion methods with high-speed DACs and ADCs to synthesize and acquire RF with relatively few components. Once in the digital domain, the logic core CPLD and RAM perform timing and buffering of the waveforms until they can be serviced by the Medusa Controller. Gating outputs are provided to control RF amplifiers and coil bias.



**Fig. 7.** The Medusa Gradient Module uses logic core hardware to buffer and play gradient waveforms synchronously within the Medusa system. Four channels of digital waveform data are produced via high-speed serial links. A fifth channel is used to control twelve user-defined gating outputs.

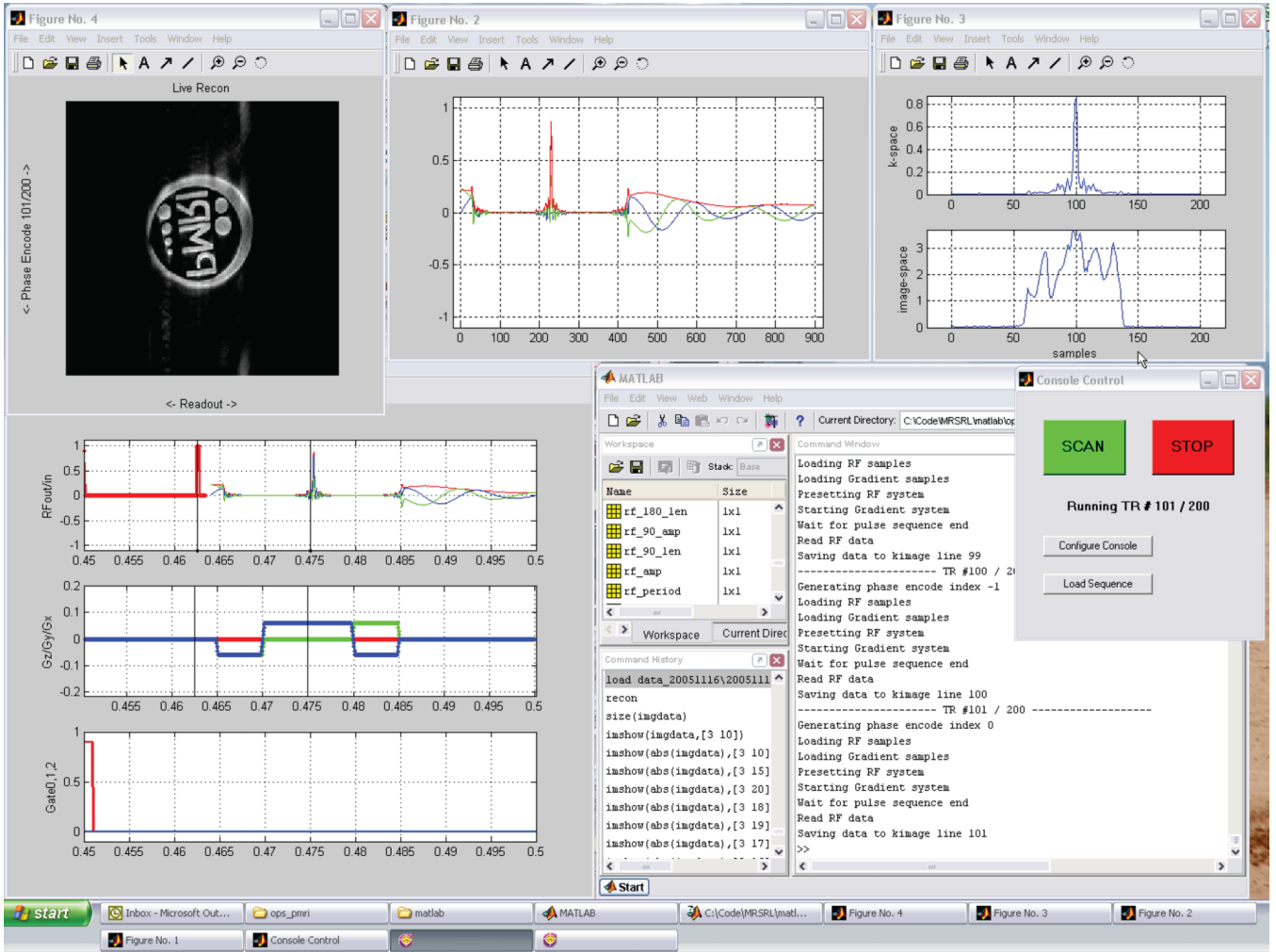


**Fig. 8.** The Medusa DAC Module converts digital waveform data from the Gradient Module into analog voltages suitable for connection to analog gradient amplifiers. The data path is differential and opto-isolated to prevent corruption due to ground offset or common-mode noise typical near gradient amps. The analog output is software-configurable for  $\pm 2.5$ ,  $\pm 5.0$ , or  $\pm 10$ V to match most amplifier input requirements without sacrificing bit-depth.



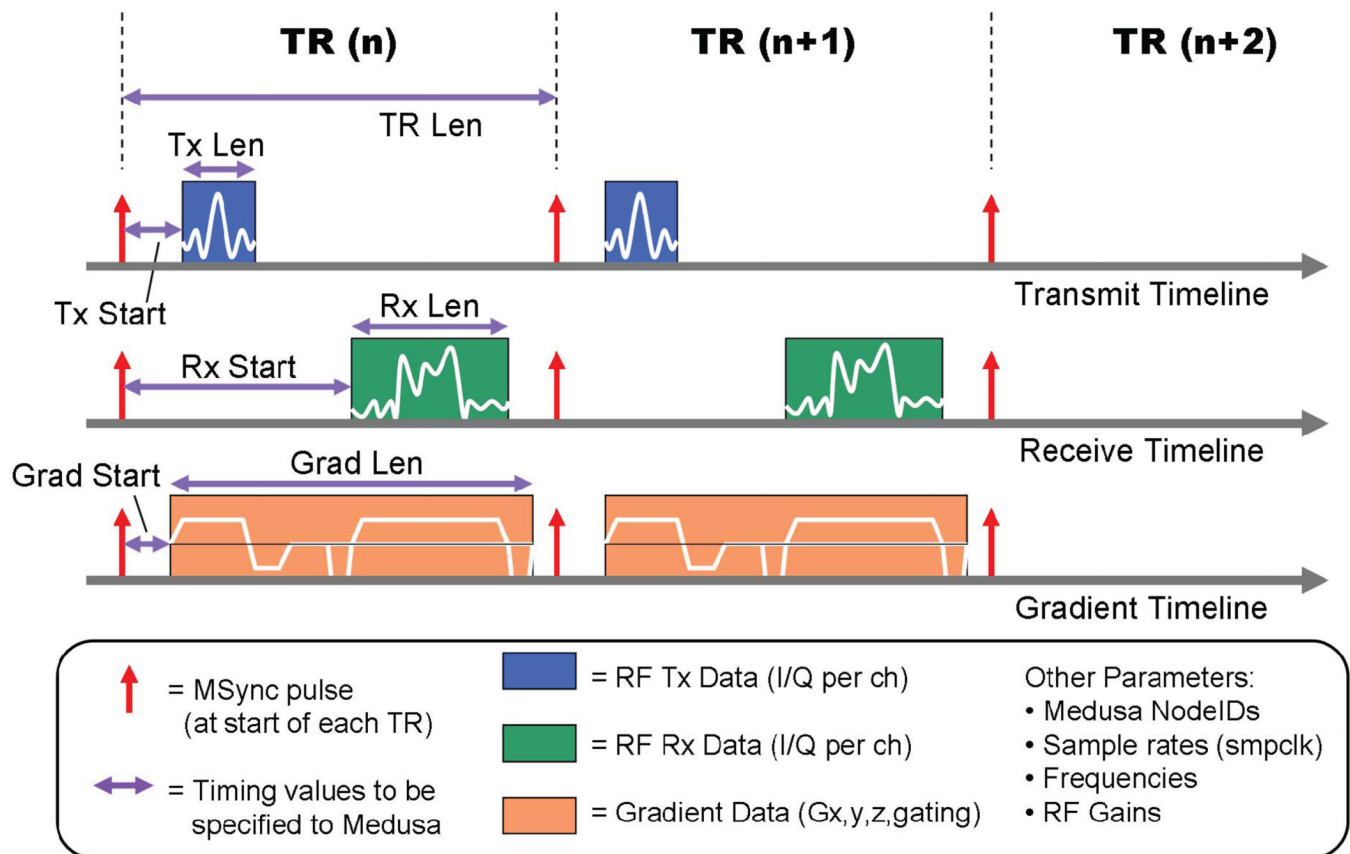
**Fig. 9.** The Medusa software architecture incorporates a Medusa server layer, which simplifies the management of multiple System Controllers and their attached modules by providing a unified software interface for all connected console hardware.



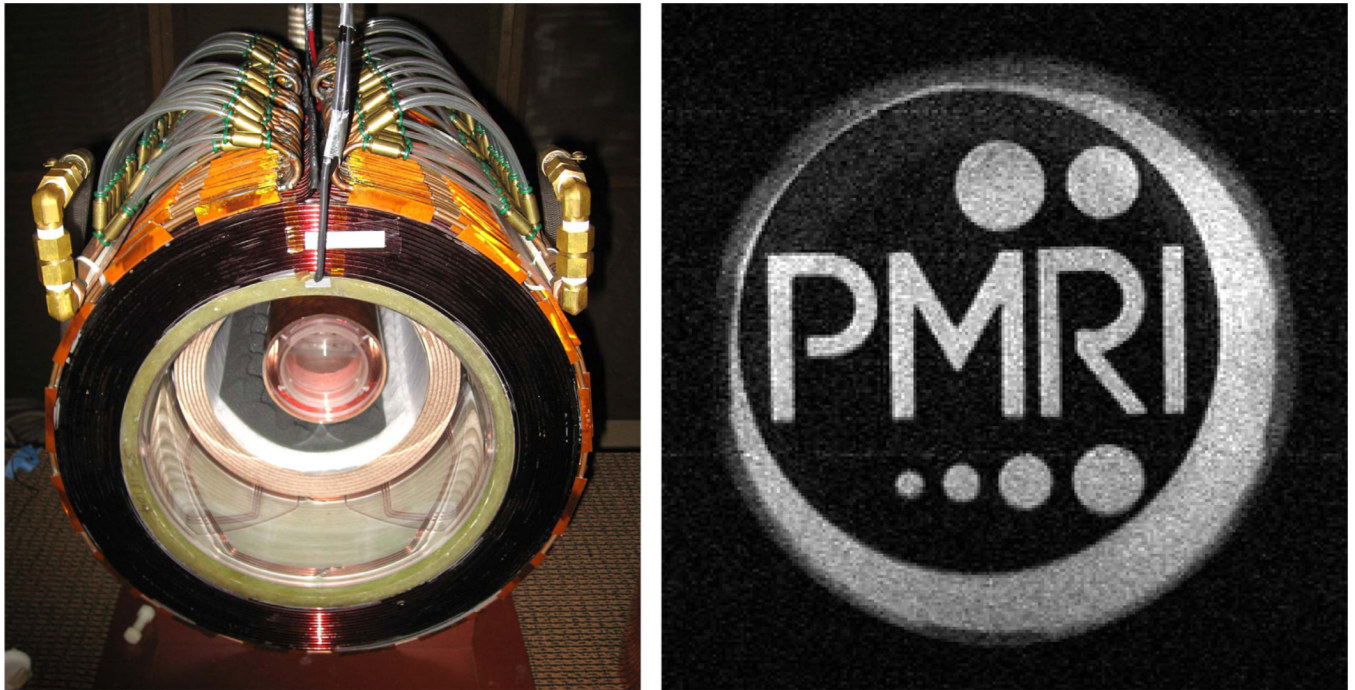


**Fig. 10.** The Medusa console may be driven directly from within Matlab to facilitate pulse sequence and reconstruction algorithm development. Here, a 2DFT spin-echo sequence is used to acquire an image on the Stanford PMRI magnet. Matlab generates the pulse sequence and performs image reconstruction, while the Medusa hardware handles all sampling and timing. The scan shown is half complete at TR=101/200 with the image updated in real-time at the end of each TR.



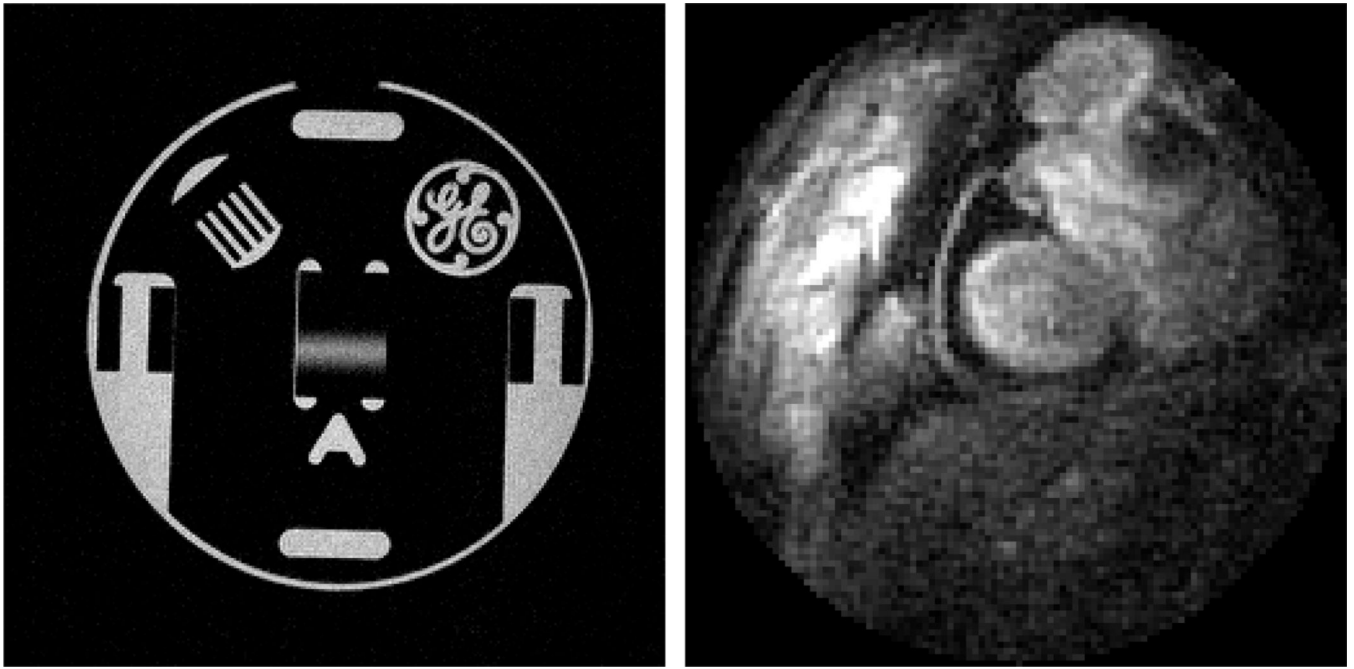
**Fig. 11.**

A Medusa pulse sequence is composed of data blocks and associated timing parameters. A single globally-distributed master synchronization signal (MSYNC) is used to keep the operation of the various modules time-aligned.

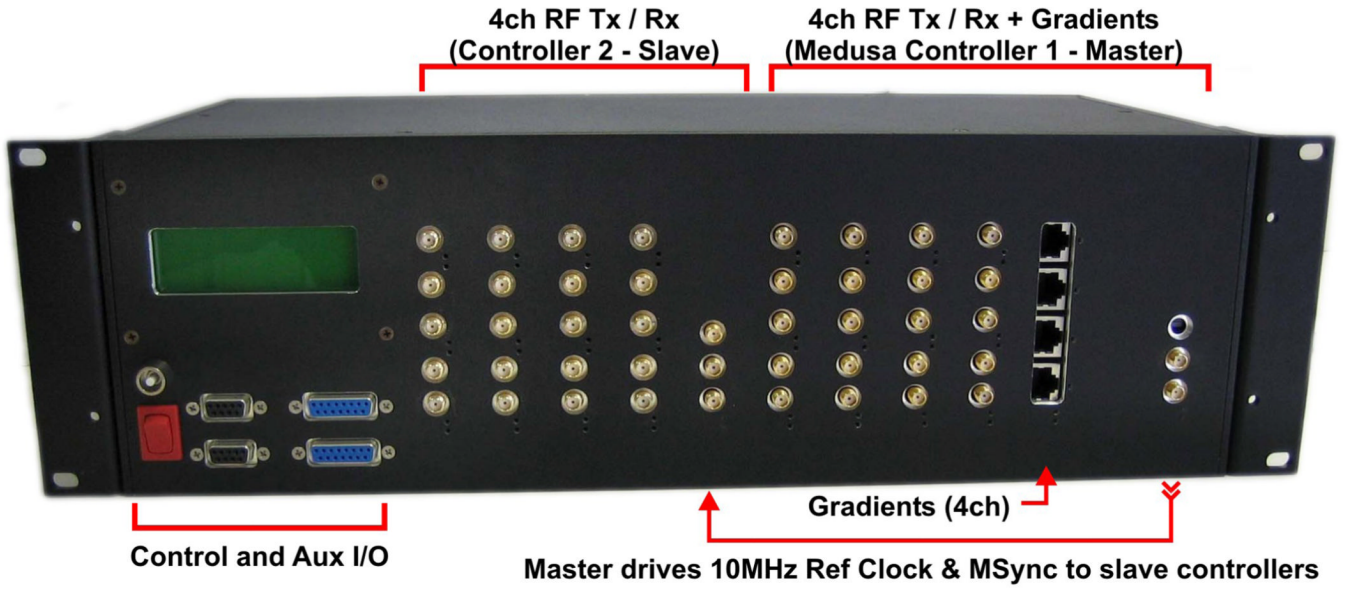


**Fig. 12.**

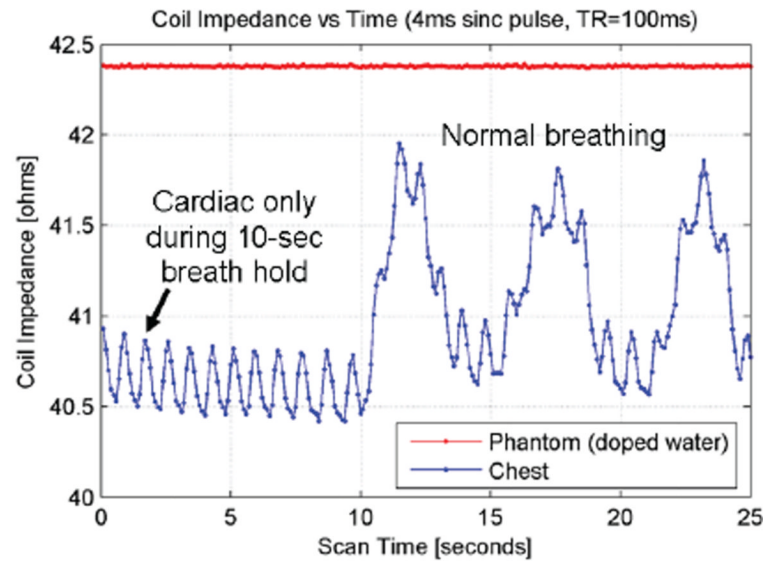
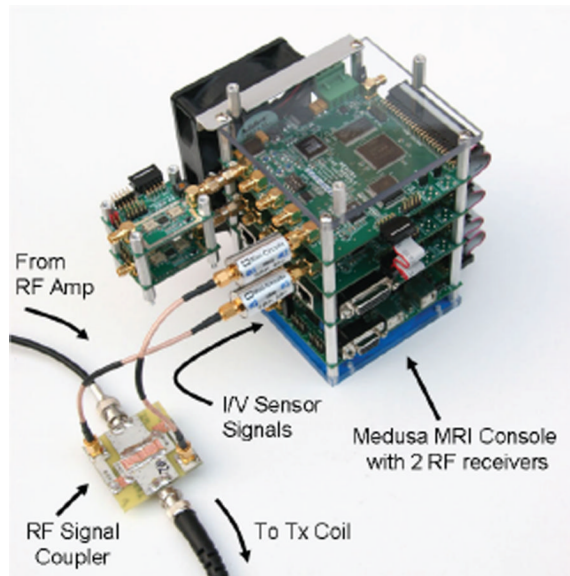
(a) The resistive magnet set of the Stanford Pre-Polarized MRI scanner operates from 0–0.5T and has a usable bore of 50cm. Parts of the gradient coil windings are visible through the acrylic former, and the RF transmit/receive coil is seen in the center. (b) A Medusa-acquired MR image of a 4cm phantom on the Stanford PMRI scanner, using a 2DFT GRE pulse sequence at 50mT field (2.228 MHz). This is one of the first images acquired by the 2nd-gen Medusa console. The features seen on the perimeter of the image are elements of the plastic cup holding the phantom.



**Fig. 13.** Left: MR image of 12cm GE resolution phantom, acquired using Medusa controlling a GE Signa Excite 1.5T scanner. Right: A frame capture of real-time cardiac imaging at 50 frames per second performed using a Medusa Console and RT-Hawk software suite on the 1.5T GE magnet. Acquisition is a 3072-pt spiral with 4× interleave at 20ms TR.



**Fig. 14.** An 8-channel real-time Medusa console demonstrates synchronized operation between two controllers. The controllers work in tandem by sharing a 10 MHz reference clock and an MSYNC pulse which signals the start of each TR. Each controller drives a bank of four RF channels, and one controller also handles the gradients. Inside the enclosure, the USB datalinks from the controllers are combined using a USB hub. This synchronization method enables scaling to very high channel counts, limited only by the practical distribution of the reference clock and MSYNC signal.



**Fig. 15.**

Left: A 2-ch Medusa console monitors transmit coil impedance to detect changes in loading due to motion or even physiological activity. Right: With a chest surface coil we detect both cardiac and respiratory rhythms as impedance changes in the coil.



**TABLE I**

## USB Theoretical and Achieved Throughput

<b>USB Bus (interface device)</b>	<b>Bus Rate (Mbps)</b>	<b>Theoretical (Mbps)</b>	<b>Achieved (Mbps)</b>
USB 1.1 Full-Speed (FT245BM)	12	8.0	5.2
USB 2.0 High-Speed (CY7C68013)	480	320	288

TABLE II

## Medusa Performance Results

RF Module	
Tx/Rx Freq Range	DC-100 MHz
Tx DAC Sample Rate	200–266 MHz
Tx Baseband Sample Rate	0–500 ksps (1.8MSPS max)
Tx SFDR	78 dBc
Tx Inter-channel Phase Error	$< \pm 1$ degree
Rx ADC Sample Rate	50–66 MHz
Rx Baseband Sample Rate	8–500 ksps (3.2MSPS max)
Rx Dynamic Range (250ksps)	89 dB (15.8 bits)
Rx Dynamic Range (62.5ksps)	93 dB (16.4 bits)
Rx Dynamic Range (16ksps)	96 dB (16.9 bits)
Rx Third-order Intercept (IIP3)	19.7 dBm
Rx Inter-channel Phase Error	$< \pm 1$ degree
Gradient Module	
Serial Data Rate (per Channel)	6–16 Mbps (10M nom.)
Sample Rate (Medusa DAC 24bit)	0–375 ksps
Sample Rate (max @ 16bits/smp)	0–1000 ksps