A Scalable Prototype MR Console Using USB

P. Stang[1], N. Matter[1], S. Conolly[2], J. Pauly[1], G. Scott[1]
[1]Electrical Engineering, Stanford University, Stanford, CA, United States, [2]Bioengineering, UC Berkeley, Berkeley, CA, United States

INTRODUCTION:

Modern MR systems are trending towards large parallel receiver and transmitter arrays. Implementing such systems with efficiency and flexibility requires a scalable modular system architecture. At ISMRM 2005[1], we proposed a USB approach to MRI system design which addressed these requirements. With a prototype console built using our design concept, we are investigating a highly scalable MR system based on digital synthesizer, receiver, and gradient blocks operating as Universal Serial Bus peripherals in a USB peripheral tree. We wish to determine bandwidth and scalability limits of an MR system using a distributed USB data transport, as well as exploring the advantages and issues of nearly all-digital RF transmission and reception in parallel scalable MRI systems.

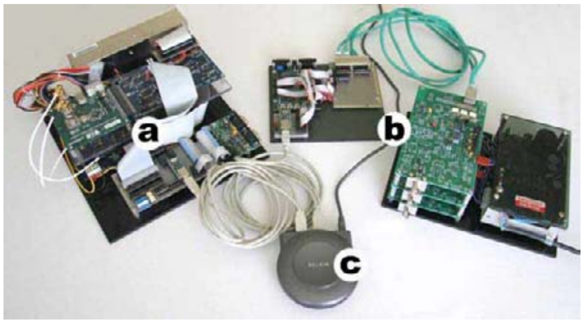

Figure 1: (a) RF and (b) Gradient Modules are connected via USB 1.1 into (c) a USB 2.0 backbone hub with Cypress Semiconductor TetraHub technology.

METHODS:

Our prototype MR console (Figure 1) is built as a distributed set of intelligent RF and Gradient modules with Universal Serial Bus (USB) interfaces. In each module, operations are coordinated by a Philips LPC2106 32-bit ARM microcontroller, or an Atmel 8-bit ATmega128 AVR controller. USB 1.1 connectivity is provided by FTDI FT245BM interface ICs. The RF module adds an exciter and receiver system based on the Analog Devices AD9854 and AD6620 digital RF communication components. The gradient module produces gradient and shim outputs from a bank of LTC1596-1 16-bit D/A converters driven by an optoisolated high-speed serial link. For MRI, all modules are synchronized via common clock and phase reference signals. A host PC connects to the modules via USB and provides data and control. To distribute the full USB 2.0 High-Speed bandwidth of the host PC to multiple USB 1.1 devices, we require a USB hub that incorporates a separate transaction translater for each port. Cypress Semiconductor's TetraHub technology provides this. For system scalability and portability, the host PC control software is layered. A low-level Console Server registers and coordinates the console modules and makes the hardware available via a unified IP network or Dynamic Link Library (DLL) interface. We have developed gradient-echo and spin-echo 2DFT imaging sequences in Matlab and use the Matlab-Executable (MEX) interface to drive the console directly.

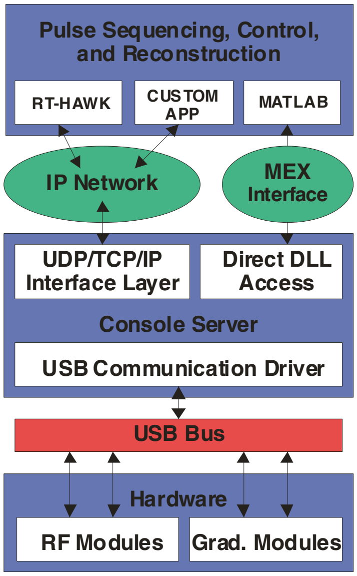

Figure 2: System-level software architecture illustrating modular design using std. open interfaces.

RESULTS:

We have successfully acquired GRE and SE images with the prototype console driving Stanford's Pre-Polarized MRI hardware[2], as shown in Figure 3. We achieve a throughput of 5.4Mbits/sec or 170K complex words/sec through each module's USB interface. The RF module can operate from DC to 66MHz (1.5T) without external RF mixers, and delivers data rates of 50K complex samples/sec. The gradient module yields output sample rates of 35K 16-bit samples/sec per channel when using four DAC channels on one controller. We determined that the RF and gradient sample rate limits are largely due to I/O speed limitations of the microcontrollers used, and are not limited by USB transfer rate. Total hardware cost of our single-channel prototype console is approximately $1400.

DISCUSSION:

To achieve the full potential of this architecture as a parallel MRI system, multiple RF and/or gradient modules may be connected to one host PC to operate together as a multi-channel console. With improved microcontroller integration and the addition of programmable high-speed logic, we expect to achieve sample rates up to the USB 1.1 throughput limit of 170K complex words/sec per module. The USB 2.0 High-Speed backbone allows up to 128 modules and bit rates of 480Mbit/s. The microcontroller in each module handles timing-sensitive tasks, leaving the host PC free to manage data transfer, pulse sequences, and image reconstruction. The console server architecture (Figure 2) simplifies management of multiple modules by providing a unified software interface to all connected console hardware. The server automatically detects, identifies, and interfaces to the modules attached to the host PC. High-level user GUIs, pulse sequencers, and reconstruction algorithms may use the IP network or DLL interfaces to operate the console hardware and recover acquired data. The DLL/MEX interface enables users to create pulse sequences, drive the scanner, and perform on-the-fly image reconstruction all from within Matlab.

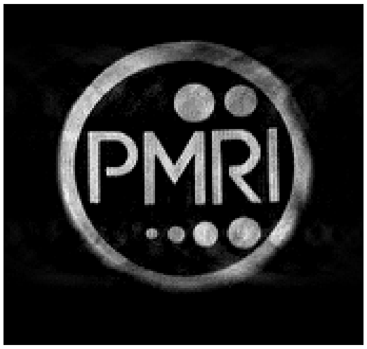

Figure 3: Image of acrylic-cutout phantom in water, acquired using the prototype console shown in Figure 1 on the Stanford PMRI scanner.

CONCLUSIONS:

Our approach to scalable MR system design depends upon digital RF system blocks, a modular architecture with local embedded processing, USB data transport, and open software interfaces. We have demonstrated a prototype single-channel MRI console built using our design approach. Scaling the prototype to multi-channel transmit and receive involves manufacturing additional modules and connecting them to our expandable console infrastructure, thus enabling parallel imaging applications.

REFERENCES:

[1] A USB Approach to Scalable Design of MRI Systems, P. Stang et al., Proc 13th ISMRM, p861, 2005.
[2] A Prepolarized MRI Scanner, G Scott et al., Proc 9th ISMRM p610, Glasgow 2001.