

## A real-time multiple antenna element testbed for MIMO algorithm development and assessment \*

Jon W. Wallace, Brian D. Jeffs, and Michael A. Jensen

Department of Electrical and Computer Engineering,  
459 CB, Brigham Young University, Provo, UT 84602-4099  
wall@ieee.org, bjeffs@ee.byu.edu, jensen@ee.byu.edu

### 1 Introduction

Information theoretic studies show that multiple-input multiple-output (MIMO) wireless systems yield a capacity benefit over single-antenna systems in rich multipath environments, leading to the development of advanced space-time coding techniques [1]. Assessing proposed space-time techniques requires very accurate channel models, detailed channel measurements, or prototyping testbeds.

Approximate channel modeling allows algorithms to be simulated quickly and represents an invaluable tool for initial algorithmic assessment. Detailed channel modeling provides real-world channels, but measured data is limited and may be hard to obtain. Advanced MIMO testbeds not only exercise algorithms in many different environments, but also assess the feasibility of real-time implementation.

We have developed a MIMO prototyping testbed, employing 4 transmit and receive elements (scalable to 8 elements). The architecture is based on a Pentek DSP platform, allowing rapid development of algorithms in C or C++. Embedded PCs simplify the usability of system. This paper details the overall architecture of the system and presents results demonstrating its operation.

### 2 System Overview

Figure 1 depicts a block diagram of the entire system. Photographs of the actual transmit and receive systems on carts is depicted in Figure 2. Details on RF components are identical to those described in [2]. Transmit symbols for four independent data streams are generated on the embedded PC, converted into complex baseband signals, and passed via the VME bus to four separate TI TMS320C6203 fixed-point DSPs. The DSPs perform pulse-shaping and pass the resulting baseband I/Q waveforms to the Pentek 6229 digital upconverter (DUC), which shifts the signals to an 8 MHz IF. A custom RF chassis mixes each IF data stream with the 2.45 GHz local oscillator (LO) from the microwave source and feeds the resulting signal to four separate transmit antennas. After passing through the channel, these signals are intercepted by four receive antennas, amplified, mixed down to the 8 MHz IF, amplified by four automatic gain control (AGC) blocks, and passed to the digital down converters (DDC) in another Pentek VME chassis. The DDCs shift the 8 MHz IF back to complex baseband. Four TI TMS320C6203s receive the I/Q baseband streams, perform matched filtering, and pass decimated matched filter outputs to the embedded PC for final processing. System synchronization is achieved with a common 10 MHz signal, provided via a cable or separate transmit/receive GPS time standards.

\*This work was supported by the National Science Foundation under Wireless Initiative Grant CCR 99-79452 and Information Technology Research Grant CCR-0081476.

## 2.1 DSP Platform

We have chosen a DSP-based platform for our space-time system, combining good performance with a fairly modest development cycle. Figure 2a shows a photograph of the transmit DSP chassis, populated with two Pentek 4292 DSP boards, one of which is used. A single 4292 DSP board has four TI TMS320C6203 200 MHz fixed point DSP processors, each of which is connected to a Pentek 6229 DUC via a high-speed FIFO. The 6229 provides interpolation, upconversion of baseband I/Q signals to a real IF waveform, and 12-bit D/A conversion at sample rates up to 200 MS/s.

Figure 2b shows a photograph of the receive DSP chassis, populated with four Pentek 4292 DSP boards and a single Pentek 4291 floating-point DSP board. Currently, only two 4292 boards are used. Each 4292 board has four TI TMS320C6203 300 MHz DSPs, two of which are connected to a Pentek 6216 DDC, and the other two are available for additional processing. The 6216 provides anti-alias filtering, 12-bit A/D conversion at sample rates up to 65 MS/s, downconversion from a real IF waveform to complex baseband, low-pass filtering, and decimation.

Utility of the system is enhanced by a Concurrent VP CP1/P3x embedded PC in each chassis, housing an Intel Pentium III 1 GHz processor with 256MB RAM, and providing connectivity to standard PC peripherals. The PC runs the Microsoft Windows operating system, allowing immediate data processing with Matlab and the ability to leverage built-in audio/video multimedia support.

## 3 Real-time Software Implementation

Software for the components described in Section 2 has initially been developed with a minimalist approach. The DSP processors perform only rudimentary functions, such as matched-filtering, pulse-shaping, decimation, and initial synchronization. More complex functions, such as data multiplexing and space-time processing are performed on the embedded host PCs, and symbol rates have been scaled to allow real-time processing.

### 3.1 Real-Time Video Streaming

The first application we developed on this platform is a video streaming demonstration employing all four transmit and receive channels to effectively quadruple data throughput. The video stream is obtained with a simple custom Windows DirectShow filter that runs on the embedded PC at the transmitter. The filter can connect to any standard DirectShow components, allowing video frames to be obtained from AVI files, MPEG 1/2 files, Webcams, etc. A simple transmit application accepts these video frames and multiplexes pixel bits to obtain I/Q samples for the desired constellation for each transmitter. A single frame of data consists of a header of Walsh codes for channel estimation, followed by a payload of these I/Q video symbols. Data frames are passed through global shared SDRAM on the VME bus via an interrupt-driven handshake. Each TI DSP processor converts the I/Q data to 8-sample half-sine pulses, which are transferred to each DUC FIFO.

On the receiver, each DDC provides I/Q data samples to a single TI DSP, which performs matched filtering and divide-by-8 decimation. This data is passed through global SDRAM on the VME bus via an interrupt handshake to the embedded PC. The receive host application estimates the channel matrix coefficients from the known training data, performs a pseudo-inverse of the channel matrix, and multiplies the receive streams by the inverted channel to obtain estimates of the I/Q transmit samples. Symbols obtained from an appropriate decision rule are combined to form the original video data, which is fed to another custom DirectShow filter, allowing real-time display or storage in a file.

Figure 3 depicts the output of the system operating in a rich multipath environment. Here, a

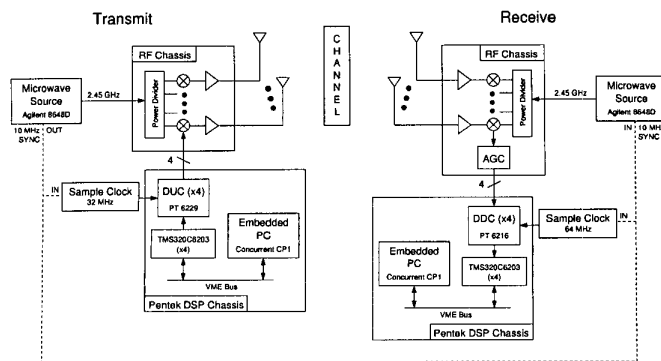


Figure 1: Block diagram of the DSP real-time system.

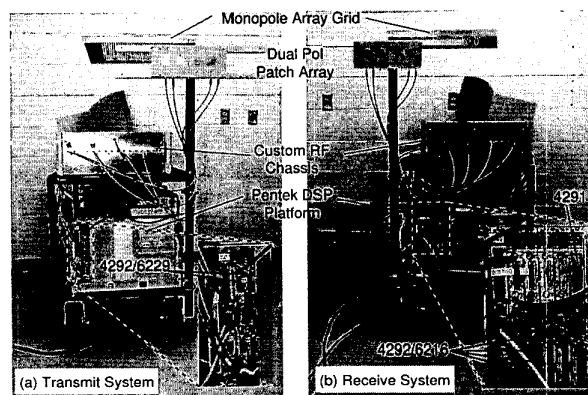


Figure 2: Photo of the transmit and receive subsystems on carts ready for deployment.

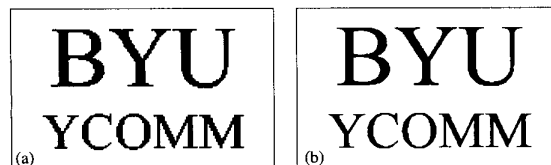


Figure 3: A single frame of a transmitted video stream with the real-time video application. (a) A single transmit/receive antenna yields a resolution of  $180 \times 100$ . (b) Four elements quadruple the throughput, yielding a resolution of  $360 \times 200$ .

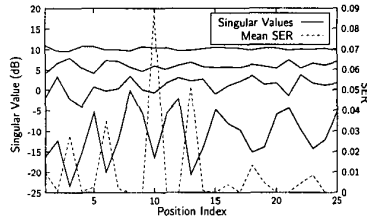


Figure 4: Average SER and channel singular values for the 4-element space-time decoding scheme as the receiver assumes 25 positions along a 10-m path.

black-and-white image was sent 30 times per second. For the single element case, 250 kS/s with a 4QAM constellation provided 500 kbit/s throughput, accommodating a resolution of  $180 \times 100$ . For the 4-element MIMO case, the amplitude on each transmit antenna was cut in half to give the same total transmit power as before. One 250 kS/s stream per transmit antenna provided 2 Mbit/s throughput, doubling the resolution to  $360 \times 200$ . Thus, the MIMO system allows a quadrupling of the system throughput without increasing transmit power, bandwidth, or single-stream constellation size.

### 3.2 SER Measurement/Channel Measurement

Another application allows measurement of symbol error rate (SER) of the coding scheme from Section 3.1. Instead of transmitting video data, a known PN23 code generates the payload symbols. We can see how sensitive the SER of the algorithm is to the conditioning of the channel matrix by storing the singular values of the estimated channel in each frame. Figure 4 plots the SER of the  $4 \times 4$  processing scheme when the transmitter and receiver are in a laboratory. Two dual-polarization patches were used at transmit and receive, and the directional elements were pointed away from each other. The transmitter was fixed and the receiver assumed several points along a prescribed path. The plot shows channel singular values, demonstrating that the algorithm fails when the weakest singular value vanishes.

## 4 Conclusion

We have presented a real-time DSP-based testbed capable of implementing fairly advanced space-time MIMO architectures. Utility of the system was demonstrated by describing a real-time video application and showing SER performance of a simple space-time decoding scheme. In the future, we intend to write software for more advanced space-time algorithms to fully utilize the resources of the platform. Such effort will demonstrate real-world performance of existing algorithms and will allow a feasibility assessment of these proposed algorithms.

## References

- [1] D. Gesbert, M. Shafi, D. shan Shiu, P. J. Smith, and A. Naguib, "From theory to practice: an overview of MIMO space-time coded wireless systems," *IEEE Journal on Selected Areas in Communications*, vol. 21, pp. 281–302, Apr. 2003.
- [2] J. W. Wallace, M. A. Jensen, A. L. Swindlehurst, and B. D. Jeffs, "Experimental characterization of the MIMO wireless channel: data acquisition and analysis," *IEEE Transactions on Wireless Communications*, vol. 2, pp. 335–343, March 2003.