PAPER Special Section on Software Defined Radio Technology and Its Applications

Development of MIMO-SDR Platform and Its Application to Real-Time Channel Measurements

Kei MIZUTANI[†a)], ***Student Member***, **Kei SAKAGUCHI**[†], **Jun-ichi TAKADA**[†], ***and*** **Kiyomichi ARAKI**[†], ***Members***

SUMMARY A multiple-input multiple-output software defined radio (MIMO-SDR) platform was developed for implementation of MIMO transmission and propagation measurement systems. This platform consists of multiple functional boards for baseband (BB) digital signal processing and frequency conversion of 5 GHz-band radio frequency (RF) signals. The BB boards have capability of arbitrary system implementation by rewriting software on reconfigurable devices such as field programmable gate arrays (FPGAs) and digital signal processors (DSPs). The MIMO-SDR platform employs hybrid implementation architecture by taking advantages of FPGA, DSP, and CPU, where functional blocks with the needs for real-time processing are implemented on the FPGAs/DSPs, and other blocks are processed off-line on the CPU. In order to realize the hybrid implementation, driver software was developed as an application program interface (API) of the MIMO-SDR platform. In this paper, hardware architecture of the developed MIMO-SDR platform and its software implementation architecture are explained. As an application example, implementation of a real-time MIMO channel measurement system and initial measurement results are presented.

key words: *multiple-input multiple-output (MIMO), software defined radio (SDR), hybrid implementation architecture, real-time channel measurement*

1. Introduction

In past years, a variety of multiantenna technologies have been studied for the evolution of wireless communication systems, e.g. adaptive antenna and direction of arrival estimation [1]. Recently, multiple-input multiple-output (MIMO) systems using multiple antennas both at the transmitter and receiver attract attention due to its high spectral efficiency [2]. Since the MIMO system has high compatibility with orthogonal frequency division multiplexing (OFDM) which has high tolerance to frequency selective channels, MIMO-OFDM is a promissing solution to enhance the system capacity of e.g. IEEE 802.11 wireless local area network (LAN) [3] and IEEE 802.16 wireless metropolitan area network (MAN) [4].

Nowadays interests of MIMO researchers are shifting from theoretical analysis to experimental verification of developed algorithms and systems. Equipment for experiments are desired to be flexible to implement a variety of MIMO systems, e.g. spatial multiplexing and space-time coding, open-loop and closed-loop, single-user and multiuser. One of the solutions for such an equipment is software defined radio (SDR) technology [5]. SDR changes its function by rewriting software on reconfigurable devices such as field programmable gate arrrays (FPGAs) and digital signal processors (DSPs). Recently, system prototyping and performance analysis of MIMO-OFDM systems by using SDR-based testbeds have been carried out by many researchers including authors:

- DSP-based testbeds [6], [7],
- FPGA-based testbeds [8]–[10],
- FPGA/DSP-based testbeds [11]–[13].

Futhermore, additional functions such as network operation, video camera, and spatial scanner enhance the capability of the testbed [14]. Although there exist a variety of testbeds, the optimal architecture of the testbed depends on the target application. Requirements for the testbed are, e.g.

- number of channels,
- duplex,
- signal processing performance,
- throughput of data bus,
- flexible application program interface (API).

It is sufficient to use commercial products, e.g. [15]–[23], if they meet the requirements of the target system, otherwise custom-made testbeds should be designed.

The MIMO-SDR platform in [12], [13] was developed by authors for experimental study on MIMO systems, since there was no commercial product to meet the requirement of 4×4 MIMO transceiver at that time. To the best of the authors' knowledge, it was a first MIMO-SDR transceiver authorized by radio license, and devoted for research and development of MIMO systems practically. The developed MIMO-SDR platform consists of multiple functional boards for baseband (BB) digital signal processing and analog frequency conversion. The BB boards have capability of arbitrary system implementation by rewriting software on FPGAs and DSPs. The developed MIMO system applications are summarized as follows.

- Off-line application
  - MIMO transmission system [13], [24]: The MIMO-OFDM transmission system with a variety of MIMO detection schemes was implemented on the developed MIMO-SDR platform as an off-line software modem. The BB boards are used as memories for transmitting (Tx) and receiving (Rx) signals, and modulation and demodulation are processed off-line on the CPU. Additional

Manuscript received April 6, 2006.

[†]The authors are with the Graduate School of Engineering, Tokyo Institute of Technology, Tokyo, 152-8550 Japan.

a) E-mail: kmiz@ap.ide.titech.ac.jp

DOI: 10.1093/ietcom/e89–b.12.3197

MIMO detection schemes are easily implemented as software modem codes.
  - MIMO channel measurement system [25], [26]: The above system is also applicable for channel measurements. Long training signals are introduced for accurate channel estimation. With a custom-made spatial scanner, MIMO channels in a residential home environment were measured. Performance of a variety of MIMO schemes was analyzed by using the measured channel data.

- Real-time application
  - MIMO transmission system [27]: A space-time block code (STBC) MIMO-OFDM transceiver was implemented on the FPGAs. Frame synchronization, fast Fourier transform (FFT), channel estimation, and STBC decoding are processed in real-time.
  - MIMO channel measurement system [28]–[30]: A real-time MIMO channel measurement system was developed for analysis of time-varying channels. A MIMO channel estimation block was implemented on the FPGAs and processed in real-time. The estimated channel data are continuously stored in a hard disk (HD). Practical performance of closed-loop transmission was analyzed from the propagation point of view. The details of the implemented software will be presented in this paper as an application example of the MIMO-SDR platform.

For the above applications, novel hybrid implementation architecture, i.e. hybrid utilization of off-line and real-time processing, was proposed to achieve a good balance between flexibility and real-time operation on the same MIMO-SDR platform. Features of the architecture are

- functional blocks with the needs for real-time operation are implemented on the FPGAs/DSPs,
- functional blocks without the needs for real-time operation are processed off-line on the CPU,
- a MIMO-SDR driver is developed as an API to control the FPGAs/DSPs as functions of CPU codes.

The MIMO-SDR driver is a key component for the hybrid implementation architecture, which plays a role for creating a flexible interface between real-time and off-line processing by taking advantages of the FPGA, DSP, and CPU.

In this paper, the details of the developed MIMO-SDR platform as well as the application example are explained. The rest of this paper is organized as follows. Section 2 will describe hardware architecture of the developed MIMO-SDR platform and its software implementation architecture. As a representative work, development of the real-time MIMO channel measurement system and initial measurement results will be given in Sect. 3 and Sect. 4 respectively. Finally, concluding remarks will be provided in Sect. 5.

2. MIMO-SDR Platform

2.1 Hardware Architecture

Photographs of the MIMO-SDR platform are shown in Fig. 1. This platform consists of multiple functional boards:

- for BB digital signal processing
  - BB Rx front-end (BRF) board,
  - BB Tx front-end (BTF) board,
  - BB signal processing (BSP) board,
- for frequency conversion
  - radio frequency (RF) board,
  - local oscillator (LO) board,

where a commercial CPU board is used to control the system. All boards are contained in a 6U size compact PCI (cPCI) rack. Advantage of this architecture is board-level scalability. Set of the boards are combined to construct a target MIMO transmitter, receiver, or transceiver.

Circuit schematic of the boards is depicted in Fig. 2. The BRF/BTF board has five 2 Mgates FPGAs and eight

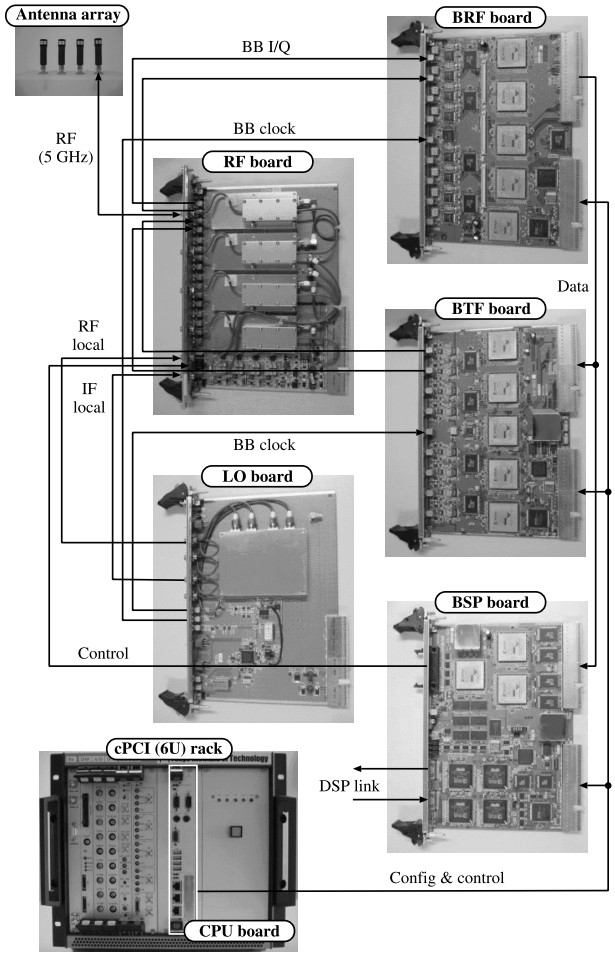


Fig. 1 Photographs of the MIMO-SDR platform.

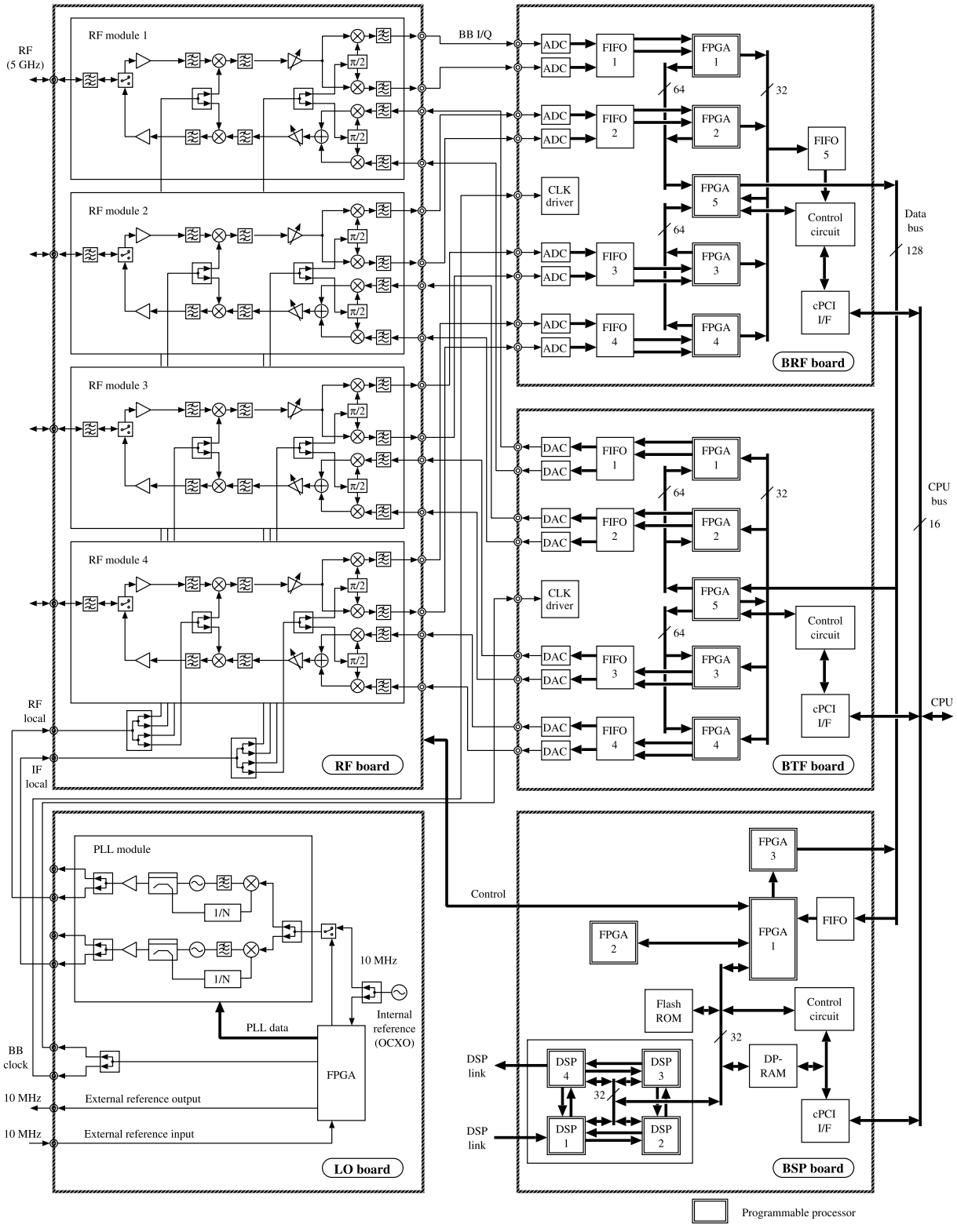


Fig. 2 Circuit schematic of the developed boards.

Table 1 Specifications of the MIMO-SDR platform.

Center frequency	5.04/5.06/5.08 GHz
Bandwidth	20 MHz
Total Tx power	−30 to +16 dBm
Rx noise figure	7 dB
RF local frequency	4470/4490/4510 MHz
IF local frequency	570 MHz
BB clock frequency	40/80 MHz
DAC	14 bit, 105 MHz (× 8)
ADC	14 bit, 80 MHz (× 8)
FPGA	2 Mgates (× 13)
DSP	600 Mflops (× 4)

analog-to-digital converters (ADCs) or digital-to-analog converters (DACs) respectively. Structures of these boards are similar, but the directions of the data flows are opposite. FPGAs 1-4 on the BRF/BTF board perform waveform-level processing of 4-channel I/Q singals. FPGA 5 is a connecting node from/to the parallel streams. The BSP board has three 2 Mgates FPGAs and four 600 Mflops DSPs. The processors on the BSP board operate information-level processing such as coding/decoding, modulation/demodulation, and space-time processing of MIMO systems. The three BB boards are connected via a 128 bit data bus for parallel transmission of Tx or Rx data between the boards. The physical layer software on the BB processors are configured and controlled from the CPU via a 16 bit bus.

On the RF board, four parallel time division duplex (TDD) super heterodyne RF modules are mounted. Center frequency is variable from 5.04 to 5.23 GHz by changing frequency of RF local signal generated in the LO board, whereas it is fixed at 5.04, 5.06, and 5.08 GHz by the limitation of radio license. Tx/Rx switch, Tx/Rx gain, and received signal strength indicator (RSSI) are digitally controlled via 32 bit parallel I/O of the BSP board. Range of total Tx power is from −30 to +16 dBm. The LO board provides RF local, intermediate frequency (IF) local, and BB clock signals. The whole system is synchronized with the oven controlled crystal oscillator (OCXO) on the LO board or an external frequency standard, e.g. a rubidium oscillator. Specifications of the MIMO-SDR platform are listed in Table 1.

2.2 Software Implementation Architecture

Figure 3 shows a reference layout of the functional blocks to implement physical layer software of general MIMO application. In order to explain each functional block, a 4×4 MIMO-OFDM system is assumed, where each Tx stream is modulated in IEEE 802.11a based OFDM signaling [31].

2.2.1 Transmitter

The Tx signal processor block on the BSP board applies coding to the Tx information bits such as forward error correction (FEC) and space-time code (STC), and generates I/Q mapped Tx symbols. The Tx symbols are transferred to the four parallel Tx post-processor blocks on the BTF board via the data bus. In each Tx post-processor block, OFDM modulation is performed, i.e. inverse fast Fourier transform (IFFT), guard interval (GI) insertion, and time window function. The modulated Tx signals are buffered in the first-in first-out (FIFO) memories and then transmitted at allocated Tx frame timing managed by the Tx controller.

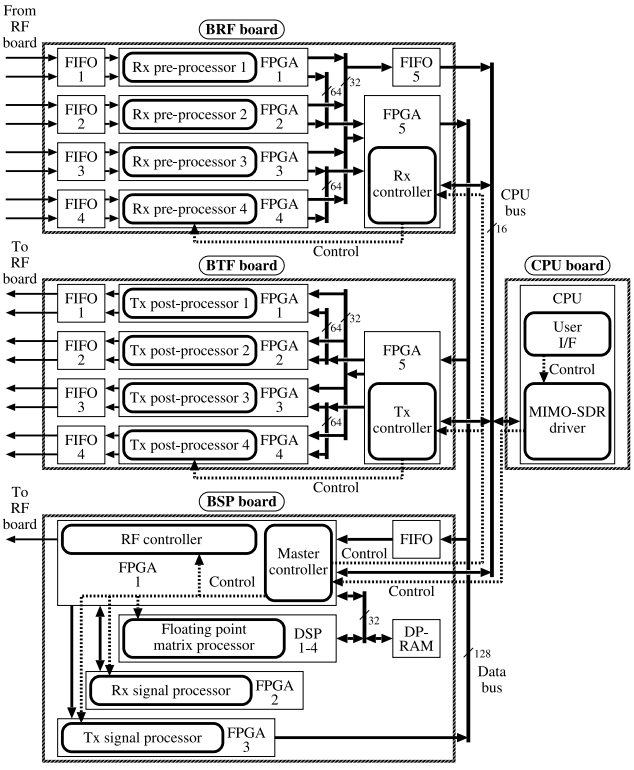


Fig. 3 Layout of functional blocks for implementation of a general MIMO application.

2.2.2 Receiver

On the BRF board, sampled Rx signals are buffered in the FIFO memories and read at Rx frame timing. Frame synchronization to estimate the timing is cooperative task of the Rx pre-processor blocks and the Rx controller. Once the frame synchronization is established, the Rx pre-processor blocks perform OFDM demodulation, i.e. GI removal and FFT. The demodulated Rx symbols are combined by the Rx controller and transferred to the BSP board. The Rx symbols are buffered again in the FIFO on the BSP board. Then the channel estimation, MIMO detection, and decoding are operated in the Rx signal processor block. In linear MIMO detection schemes, weight matrices are required for the channel inversion. The weights are calculated in the floating point matrix processor block implemented on the DSPs.

2.2.3 System Control

The controller on the BSP board is basically the master of the controllers on the BB boards including the Tx, Rx, and

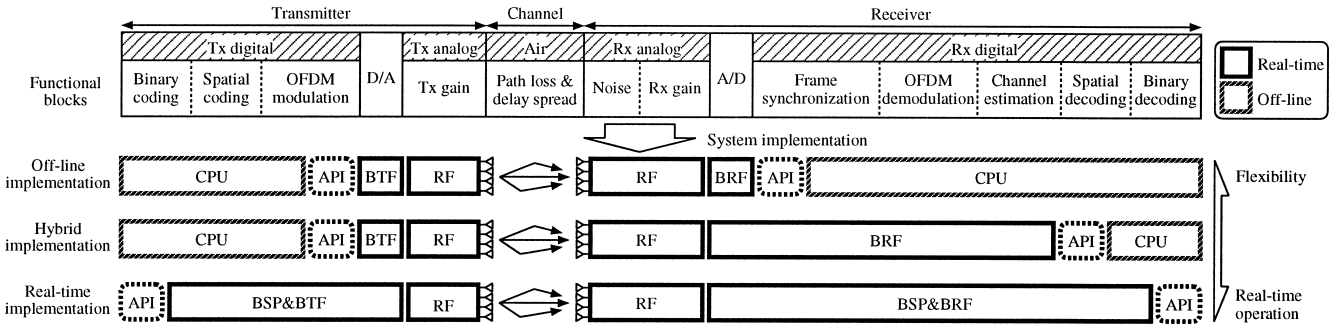


Fig. 4 Hybrid implementation architecture.

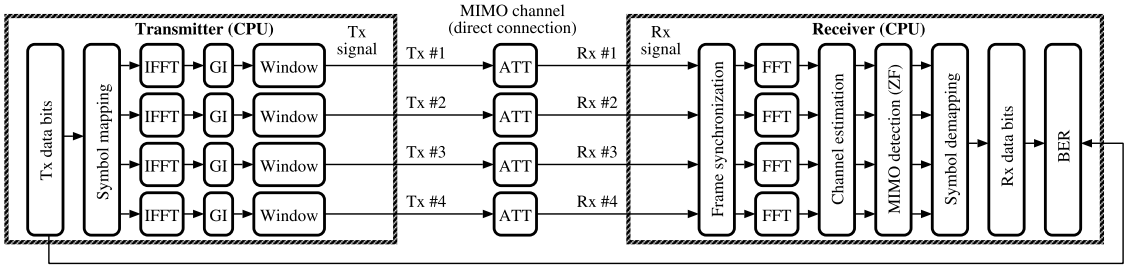


Fig. 5 Experimental setup for BER measurement.

RF controllers. The master controller executes the instructions from the user interface interpreted by the MIMO-SDR driver implemented on the CPU. The MIMO-SDR driver is a wrapper of the hardware driver, which can be customized like an extensible API by adding control functions of the system.

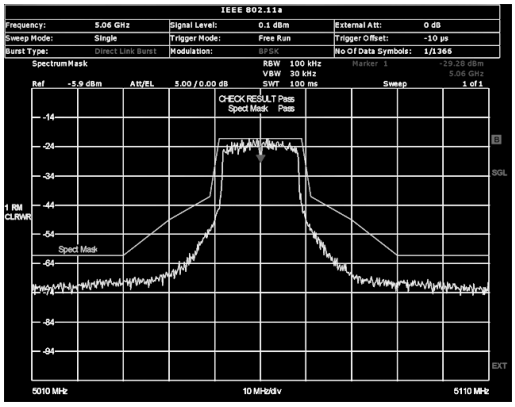
2.3 Hybrid Implementation Architecture

Section 2.2 described implementation of a full-real-time transceiver. If measurements are carried out in static environments, real-time processing is not necessary. Instead, flexibility of implementation is preferable to test different systems, e.g. a variety of MIMO detection schemes. On the other hand, measurements in dynamic environments require real-time processing rather than flexibility. In order to achieve a good balance between flexibility and real-time operation, hybrid implementation architecture, i.e. hybrid utilization of off-line and real-time processing, was employed by using the MIMO-SDR driver.

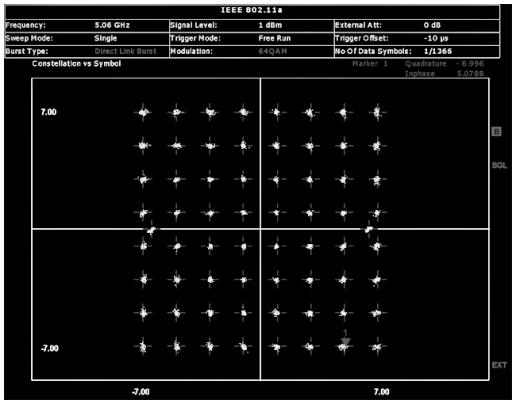
Figure 4 shows an example of the hybrid implementation architecture. Functional blocks to be implemented on the MIMO-OFDM system are shown in the upper part of the figure. In the off-line implementation, the Tx/Rx digital processing parts are written as a software modem on the CPU. API function interconnects the software modem with the memories implemented on the BTF and BRF boards. It is in contrast with the real-time implementation where all of the functional blocks are implemented on the BSP, BTF, and BRF boards in real-time. API function for the real-time implementation works as an interconnection between the hard-coded modem on the BSP board and the user interface on the CPU board. As such, the implemented API can be a flexible interface between the CPU and the BB boards. Therefore, the hybrid implementation between off-line and real-time can be achieved. In the hybrid implementation, Tx/Rx digital processing parts are appropriately arranged on the CPU and the BB boards in accordance with the requirements for real-time operation of the target system. A representative work of the hybrid implementation is a real-time MIMO channel measurement system [28]. At the receiver, frame synchronization, OFDM demodulation, and channel estimation blocks are implemented on the BRF board and processed in real-time. API function continuously store the estimated channel data to the CPU. On the other hand, at the transmitter, the software modem of the off-line implementation is enough to transmit training signals. As such, the MIMO-SDR platform has high flexibility to implement arbitrary systems.

2.4 Experimental Verification

Verification of the developed MIMO-SDR platform was conducted in the off-line transceiver configuration, where modulation and demodulation were processed on the CPU. Experimental setup is given in Fig. 5. At the transmitter, four Tx streams are modulated into the IEEE 802.11a based OFDM signals [31], and transmitted simultaneously at the center frequency of 5.06 GHz. Figure 6(a) shows the Tx signal spectrum with the IEEE 802.11a spectrum mask, and Fig. 6(b) shows the Tx signal constellation in 64QAM signaling. Error vector magnitude (EVM) of the Tx signal was 2.8% (max) and 1.8% (min) at 0 dBm output. To perform bit error rate (BER) measurement in an additive white



(a) Spectrum.



(b) Constellation.

Fig. 6 Tx signal spectrum and constellation (64QAM).

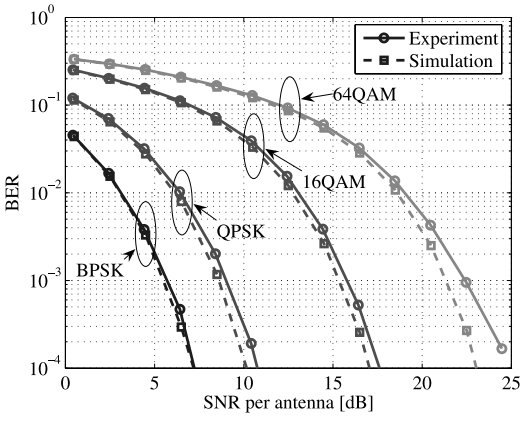


Fig. 7 BER performance in AWGN channel.

Gaussian noise (AWGN) environment, the transmitter and receiver were connected directly through attenuators, where the MIMO channel was considered as parallel single-input single-output (SISO) channels with equal gain. At the receiver, zero-forcing (ZF) was used for MIMO detection of the Rx signals, where it works as parallel coherent detection of SISO signals in this case. In the BER measurement, the Rx signal-to-noise ratio (SNR) was varied by controlling Tx power from −25 to 0 dBm per antenna. Figure 7 shows results of the BER measurement as well as the simulation. Degradation of the measured BER in SNR is less than 1–2 dB than that of theoretical results. The degradation is principally due to the imperfection of RF circuits such as phase noise, I/Q imbalance, and nonlinearity.

3. Real-Time MIMO Channel Measurement System

3.1 System Requirements

A real-time MIMO channel measurement system implemented on the developed MIMO-SDR platform is introduced as an application example in this section. This system continuously estimates MIMO channel responses and stores the results in a HD. The obtained channel data are used for performance analysis of MIMO transmission in dynamic environments. Figure 8 shows the functional blocks of the developed system. As requirements for the system, following issues are considered:

- estimation accuracy,
- sampling interval of channel responses.

To achieve high estimation accuracy, totally 64 OFDM symbols are used for 4×4 MIMO channel measurement. Frame structure of the training signal is based on the preamble of IEEE 802.11a wireless LAN [31] and extended for the 4×4 MIMO system. The 11a preamble consists of two parts: short preamble (SP) for frame synchronization (two OFDM symbols) and long preamble (LP) for channel estimation (two OFDM symbols). The LP signals are extended to be orthogonal among Tx antennas by using 32nd-order Hadamard codes. Theoretically, channel estimation error using 64 OFDM symbols (32nd-order Hadamard codes) becomes less than 1 percent when signal-to-noise (SNR) is higher than 3 dB. In order to guarantee the estimation accuracy, Tx gain is controlled to achieve reasonable SNR in the measurement environment under the consideration of the saturation point of RF circuits. On the other hand, Rx gain is adjusted according to average and maximum Rx power in the environment by monitoring RSSI to minimize quantization noise of the ADC. Furthermore, Rx gain is fixed during a measurement so as not to cause nonlinear noise fluctuations. The timing of the frame synchronization is also fixed during a measurement so as not to cause channel phase fluctuation from frame to frame. The fixed frame synchronization timing is estimated from average frame synchronization timing of the environment measured in advance. Once the average frame synchronization timing is given, the frame synchronization is replaced by a timer-controlled trigger.

Sampling interval of channel responses is designed from maximum Doppler frequency and data storage capability. The target scenario of the channel estimation is 5 GHz-band indoor wireless LAN, where it is assumed that

- positions of the transmitter and receiver are fixed,
- surrounding objects are moving.

It is reported that “Bell” shape Doppler power spectrum is obtained in such an environment, where maximally

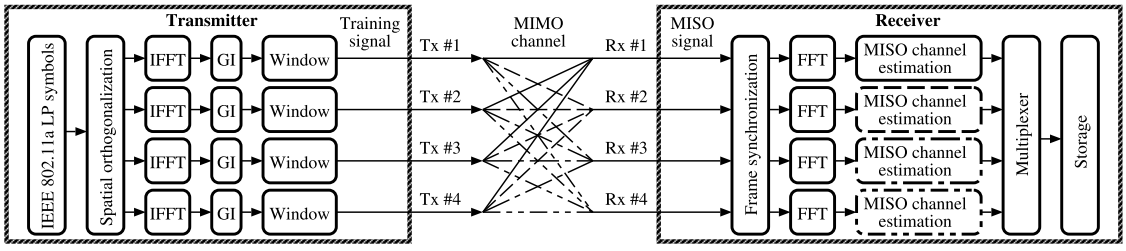


Fig. 8 Functional blocks of the real-time MIMO channel measurement system.

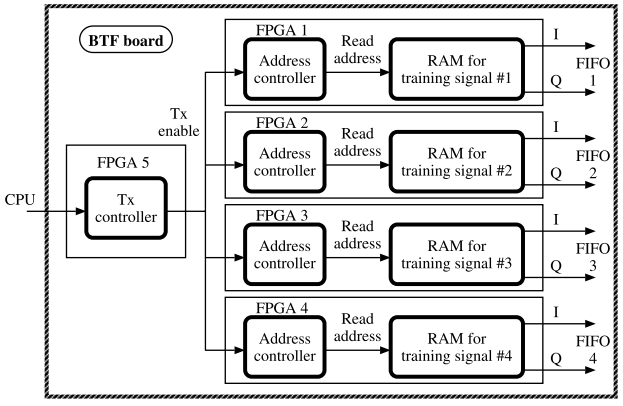


Fig. 9 Block diagram of the transmitter.

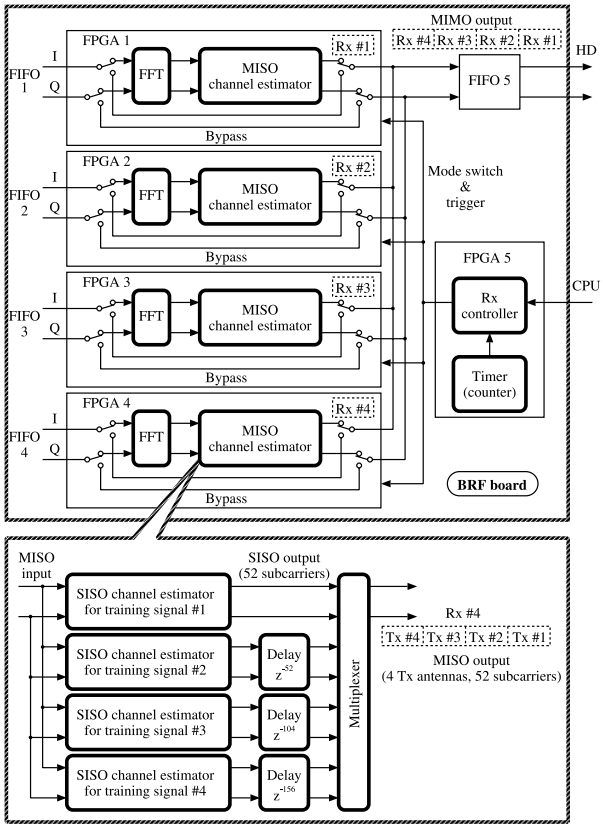


Fig. 10 Block diagram of the receiver.

30 Hz Doppler component is considered at 5.25 GHz in IEEE 802.11n [32]. Sampling interval of channel responses is designed so as to measure the Doppler frequency, which should be shorter than 16 ms to satisfy the sampling theorem for the assumed Doppler frequency. The shorter sampling interval is better to track rapidly moving objects, while the shorter sampling interval requires higher storage capability, i.e. throughput of the CPU bus and stability of writing data to the HD. In this system, the sampling interval is chosen to be 4 ms by considering both the Doppler frequency to be measured and data storage capability of the CPU.

3.2 Transmitter Implementation

The transmitter in this application employs off-line implementation architecture described in Sect. 2. The principal task of this implementation is to generate periodic signal waveforms stored in the memories on the BTF board. Maximum memory size constructed in a single FPGA is about 1 Mbit. For 14 bit I/Q signals, available memory is 32k words. Since the DACs and FPGAs are driven by 80 MHz BB clock, length of an OFDM symbol (4 μs) is 320 samples. Therefore, the memory can store about 100 OFDM symbols for an I/Q Tx channel. Since the length of a training signal (SP and LP) is 66 OFDM symbols, the memory size is sufficient for this application. Figure 9 shows a block diagram of the transmitter. FPGAs 1-4 on the BTF board were configured as random access memories (RAMs) to store the modulated training signals in the CPU. The training signals are cyclically read from the memories and output via DACs according to the enable signal timing managed by FPGA 5.

3.3 Receiver Implementation

At the receiver, functional blocks for the real-time MIMO channel estimation are implemented on the BRF board. Figure 10 shows a block diagram of the receiver. The receiver has two operation phases: frame synchronization and channel measurement. This architecture removes the instability of the frame synchronization as well as reduces the processing load of the BRF board.

In the frame synchronization phase, sampled Rx signal data are not processed on the FPGAs (bypassed) and loaded to the CPU. Frame synchronization is performed off-line on the CPU by taking autocorrelation of the Rx signals. By repeating this process, average frame synchronization timing

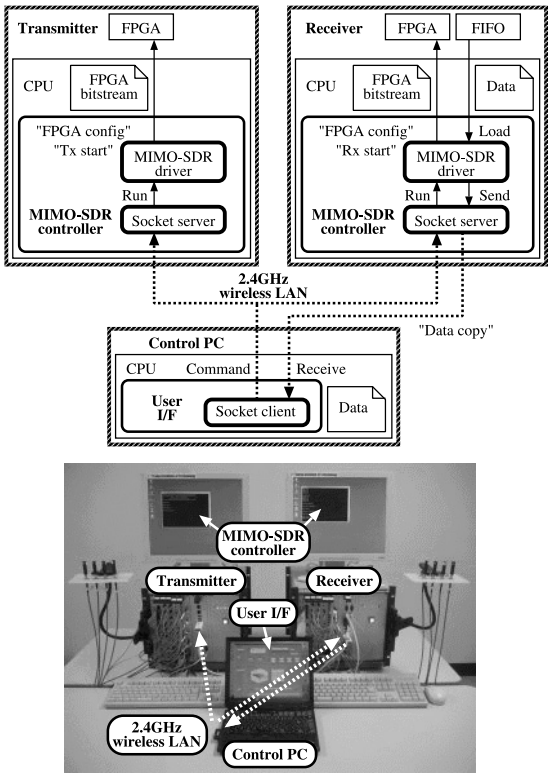


Fig. 11 Block diagram of the system control.

in the environment is estimated. Then estimated frame timing is set to the timer of the trigger control, and the receiver starts the channel measurement in real-time.

In the channel measurement phase, FFT and multiple-input single-output (MISO) channel estimation are done on FPGAs 1-4. The MISO channel estimator consists of four SISO channel estimators for each Tx antenna. All SISO channel estimators run in parallel. Estimated channel responses for 52 subcarriers are multiplexed in the order of Tx antennas. Results of the MISO channel estimation are also multiplexed to complete a MIMO snapshot which consists of 832 complex values (4 Rx antennas, 4 Tx antennas, and 52 subcarriers). Finally, the MIMO snapshot is stored to FIFO 5. Although the processing time of the MIMO channel estimation between signal input and one snapshot output is less than 0.4 ms in this implementation, processing cycle is decimated to meet the capability of data storage.

3.4 System Controller Implementation

The MIMO-SDR controller with the MIMO-SDR driver is implemented on the CPU board to control the whole system including the storage of the measured data to the HD in real-time. Figure 11 shows a block diagram of the system control. The MIMO-SDR controller employs the timer with millisecond order precision to adjust the storage rate and the channel estimation rate. The rates are designed as follows:

- FPGAs 1-4 on the BRF board store one snapshot to FIFO 5 every 4 ms,

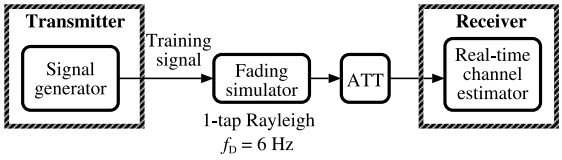
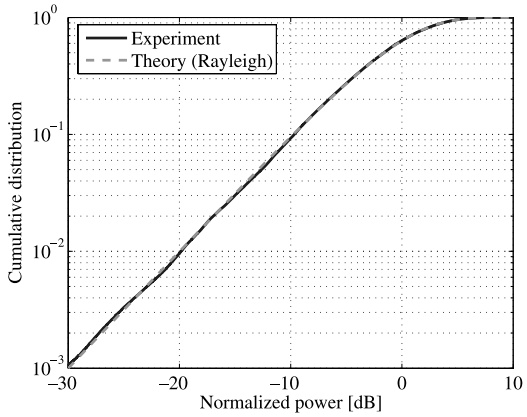
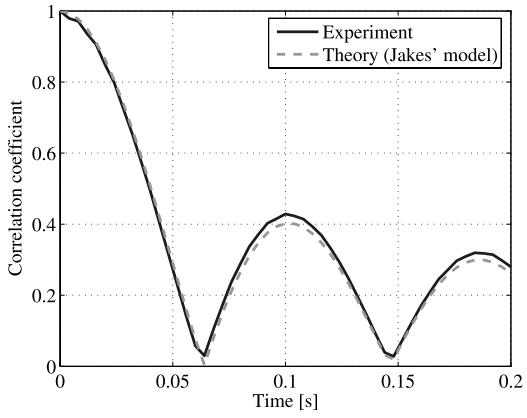


Fig. 12 Experimental setup for system verification.



(a) Cumulative distribution.



(b) Autocorrelation.

Fig. 13 Cumulative distribution and autocorrelation of measured channel responses.

- MIMO-SDR driver loads five snapshots from FIFO 5 and stores them to the HD every 20 ms.

In order to improve the stability of the real-time processing, the MIMO-SDR driver is separated from the user interface. A socket server and a client are implemented in the MIMO-SDR controller and the user interface respectively. The server waits commands from the client, e.g. "FPGA config", "Tx start", "Rx start", "Data copy", etc. After receiving the commands, the MIMO-SDR controller runs the corresponding functions implemented in the MIMO-SDR driver. This architecture improves the stability of the continuous data storage by reducing load of the CPU in the receiver.

3.5 Experimental Verification

The implemented measurement system was verified by using fading simulator. Experimental setup is given in Fig. 12. The transmitter and receiver were directly connected through the fading simulator and attenuator. Configuration of the fading simulator was 1-tap Rayleigh with Jakes' Doppler spectrum spread up to 6 Hz [33]. Cumulative distribution and autocorrelation of the measured channel responses are calculated and shown in Figs. 13(a) and (b) respectively with the theoretical curves. Both results in Fig. 13 are satisfactory to prove the validity of implemented real-time channel measurement system.

4. Example of MIMO Channel Measurement

By using the real-time MIMO channel measurement system described in Sect. 3, a channel measurement was carried out in a dynamic scenario as well as a static scenario. Figure 14 shows the measurement environment where the Tx and Rx antennas were fixed on the tables in the same room. In the dynamic scenario, a man was walking in the room to emulate a moving surrounding object. The number n on the walking route (dashed line in Fig. 14) indicates the position of the man at time $t = n$ s. Measurement parameters are listed in Table 2. MIMO channel matrices for all subcarriers were measured every 4 ms during 10 s. It results in totally 2500 MIMO snapshots.

Figure 15 shows an example of path loss fluctuation. Comparing to the static scenario, the path loss in the dynamic scenario is significantly fluctuating. Since the man was passing through line-of-sight between Tx and Rx antennas at around the time of 7.5 s, a large dip is observed in the measured path loss.

By using the measured channel matrices, instantaneous channel capacity of a 4×4 MIMO and SISO system can be calculated on each subcarrier by using the following equations:

$$C_{\mathrm{MIMO}} = \log_2 \left[\det \left(\boldsymbol{I} + \frac{p}{\sigma^2} \boldsymbol{H} \boldsymbol{H}^{\mathrm{H}} \right) \right], \tag{1}$$

$$C_{\mathrm{SISO},ij} = \log_2 \left[1 + \frac{4p}{\sigma^2} |\boldsymbol{H}_{ij}|^2 \right], \tag{2}$$

where $\boldsymbol{I}$ is 4×4 identity matrix, p is Tx power per antenna per subcarrier†, σ^2 is noise power per subcarrier, $\boldsymbol{H}$ is an estimated 4×4 channel matrix for each subcarrier, $[\cdot]^{\mathrm{H}}$ denotes conjugate transpose, and i and j are the indices of Rx and Tx antennas, respectively. In the calculation, noise power was estimated from the Rx noise figure and temperature in the environment. Figure 16 shows the measured channel capacity of the first subcarrier. In the static scenario, channel capacity of the 4×4 MIMO system is about 3.4 times that of the SISO system. In the dynamic scenario, channel capacity is fluctuating around the case of static. Channel capacity loss of the SISO system was maximally 45 percent from average when the shadowing occurred at around the time of 7.5 s. On the other hand, that of the MIMO system was at most 18 percent. It is another advantage of MIMO systems due to multipath diversity. Further results of analysis are

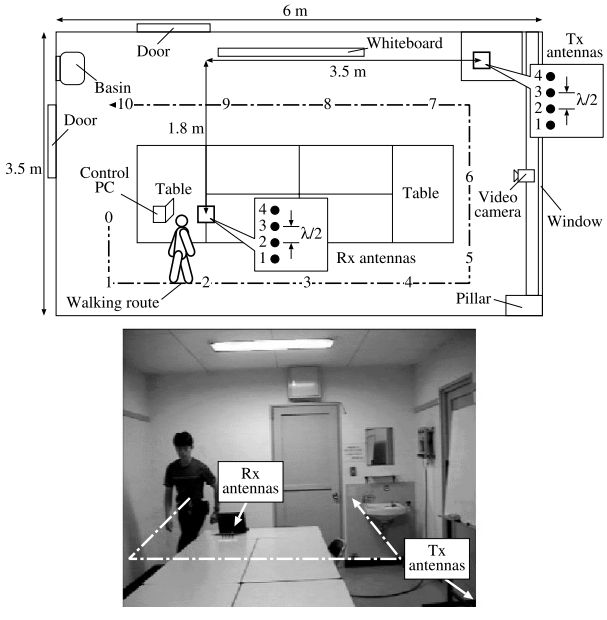


Fig. 14 Measurement environment.

Table 2 Measurement parameters.

Center frequency	5.06 GHz
Bandwidth	16.6 MHz
Tx power	0 dBm per antenna
Tx signal	IEEE 802.11a based OFDM
MIMO configuration	4×4
Antenna array	$\lambda/2$ spacing 4-element ULA
Sampling interval	4 ms
Measurement duration	10 s
Walking speed	1 m/s
Room temperature	28°C

ULA: uniform linear array

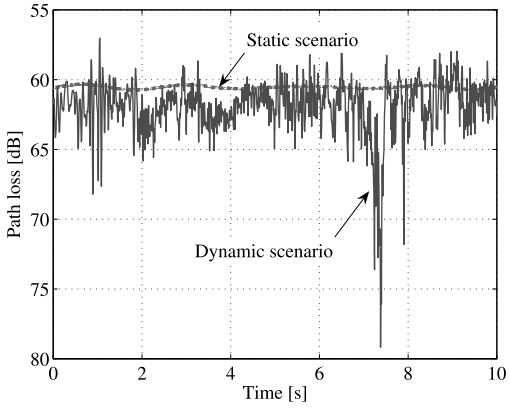


Fig. 15 Measured path loss.

†In the SISO case, 4 times power per antenna than that of the MIMO is allocated for fair analysis.

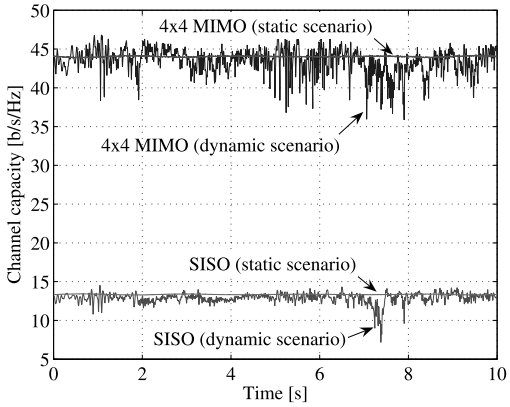


Fig. 16 Measured channel capacity.

given in [29].

5. Conclusion

A MIMO-SDR platform for implementation of MIMO measurement systems was developed in this paper. The MIMO-SDR platform employs hybrid implementation architecture by taking advantages of FPGA, DSP, and CPU, where functional blocks with the needs for real-time processing are implemented on the FPGAs/DSPs, and other blocks are processed off-line on the CPU. The MIMO-SDR driver is a key component for the hybrid implementation architecture, which plays a role for creating flexible interface between real-time and off-line processing. In the paper, the details of hardware architecture of the developed MIMO-SDR platform and its software implementation architecture were explained. As an application example, implementation of a real-time MIMO channel measurement system and initial measurement results were given. The developed MIMO-SDR platform has been devoted to measurement and analysis of real-life MIMO systems in several organizations.

Acknowledgments

The authors would like to thank Shintaro Arata, Kazuhiko Kaiga, and Sakae Manabe from Koden Electronics for cooperation of the baseband signal processing hardware development, and Tsutomu Mitsui, Masao Otani, and Atsuya Yokoi from Samsung Yokohama Research Institute for cooperation of the RF hardware development, and Takahiro Sinke from Addnics for cooperation of the RF hardware implementation and tuning.

References

[1] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," IEEE Signal Process. Mag., vol.13, no.4, pp.67–94, July 1996.
[2] D. Gesbert, M. Shafi, D. Shiu, P. Smith, and A. Naguib, "From theory to practice: An overview of MIMO space-time coded wireless systems," IEEE J. Sel. Areas Commun., vol.21, no.3, pp.281–302, April 2003.
[3] http://www.ieee802.org/11/
[4] http://www.ieee802.org/16/
[5] J. Mitola Ⅲ, Software radio architecture, Wiley-Interscience, New York, 2000.
[6] W. Xiang, T. Pratt, and X. Wang, "A software radio testbed for two-transmitter two-receiver space-time coding OFDM wireless LAN," IEEE Commun. Mag., vol.42, no.6, pp.S20–S28, June 2004.
[7] C. Dubuc, D. Starks, T. Creasy, and Y. Hou, "A MIMO-OFDM prototype for next-generation wireless WANs," IEEE Commun. Mag., vol.42, no.12, pp.82–87, Dec. 2004.
[8] A. van Zelst and T.C.W. Schenk, "Implementation of a MIMO OFDM-based wireless LAN system," IEEE Trans. Signal Process., vol.52, no.2, pp.483–494, Feb. 2004.
[9] S. Nanda, R. Walton, J. Ketchum, M. Wallace, and S. Howard, "A high-performance MIMO OFDM wireless LAN," IEEE Commun. Mag., vol.43, no.2, pp.101–109, Feb. 2005.
[10] H. Yu, M.S. Kim, E.Y. Choi, T. Jeon, and S.K. Lee, "Design and prototype development of MIMO-OFDM for next generation wireless LAN," IEEE Trans. Consum. Electron., vol.51, no.4, pp.1134–1142, Nov. 2005.
[11] V. Jungnickel, A. Forck, T. Haustein, S. Schiffermüller, C. von Helmolt, F. Luhn, M. Pollock, C. Juchems, M. Lampe, S. Eichinger, W. Zirwas, and E. Schulz, "1 Gbit/s MIMO-OFDM transmission experiments," Proc. IEEE VTC 2005-Fall, pp.861–866, Dallas, TX, USA, Sept. 2005.
[12] K. Mizutani, K. Sakaguchi, H. Wang, A. Yamaoka, J. Takada, K. Araki, S. Arata, K. Kaiga, and S. Manabe, "Development of a software defined radio oriented MIMO signal processing hardware," IEICE SR Workshop, SR04-7, May 2004.
[13] K. Mizutani, K. Sakaguchi, J. Takada, and K. Araki, "Development of 4×4 MIMO-OFDM system and test measurement," Proc. 12th European Signal Processing Conference (EUSIPCO 2004), pp.701–704, Vienna, Austria, Sept. 2004.
[14] W. Zhu, D. Browne, and M. Fitz, "An open access wideband multi-antenna wireless testbed with remote control capability," Proc. 1st International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TRIDENTCOM 2005), pp.72–81, Trento, Italy, Feb. 2005.
[15] http://www.pentek.com/
[16] http://www.iaf-bs.de/
[17] http://www.spectrumsignal.com/
[18] http://www.hunteng.co.uk/
[19] http://www.nallatech.com/
[20] http://www.ics-ltd.com/
[21] http://www.sundance.com/
[22] http://www.lyrtech.com/
[23] http://www.signalion.com/
[24] Y. Kakishima, L.H. Doan, K. Mizutani, K. Sakaguchi, and K. Araki, "Implementation of MIMO-OFDM transmission system on MIMO software defined radio," IEICE Technical Report, SR2005-46, July 2005.
[25] D.D. Nguyen, H.Y.E. Chua, K. Sakaguchi, K. Araki, H. Iwai, T. Sakata, and K. Ogawa, "Measurements on area coverage of 5 GHz band MIMO-OFDM system in residential home environment," Proc. IEEE PIMRC 2005, pp.246–250, Berlin, Germany, Sept. 2005.
[26] G.K. Tran, N.D. Dao, K. Sakaguchi, K. Araki, H. Iwai, T. Sakata, and K. Ogawa, "Performance analysis of MIMO-OFDM systems using indoor wideband MIMO channel measurement data," Proc. IEEE VTC 2006-Spring, pp.2868–2872, Melbourne, Australia, May 2006.
[27] A. Yamaoka, K. Mizutani, K. Sakaguchi, J. Takada, and K. Araki, "Prototyping of MIMO-OFDM transceiver based on software defined radio oriented hardware," IEICE Technical Report, AP2004-150, RCS2004-171, Oct. 2004.
[28] K. Mizutani, K. Sakaguchi, J. Takada, and K. Araki, "Real-time MIMO channel measurement using 5 GHz 4×4 MIMO-OFDM hardware," IEICE Technical Report, WBS2004-117, AP2004-298,

RCS2004-385, MoMuC2004-168, MW2004-295, March 2005.
[29] K. Mizutani, K. Sakaguchi, J. Takada, and K. Araki, "Analysis on temporal variation of indoor MIMO channels," IEICE Technical Report, AP2005-118, Dec. 2005.
[30] K. Mizutani, K. Sakaguchi, J. Takada, and K. Araki, "Measurement of time-varying MIMO channel for performance analysis of closed-loop transmission," Proc. IEEE VTC 2006-Spring, pp.2854–2858, Melbourne, Australia, May 2006.
[31] IEEE, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications high-speed physical layer in the 5 GHz band," IEEE Std 802.11a-1999, 1999.
[32] V. Erceg, L. Schumacher, P. Kyritsi, A. Molisch, D.S. Baum, A.Y. Gorokhov, C. Oestges, Q. Li, K. Yu, N. Tal, B. Dijkstra, A. Jagannatham, C. Lanzl, V.J. Rhodes, J. Medbo, D. Michelson, M. Webster, E. Jacobsen, D. Cheung, C. Prettie, M. Ho, S. Howard, B. Bjerke, L. Jengx, H. Sampath, S. Catreux, S. Valle, A. Poloni, A. Forenza, and R.W. Heath, "TGn channel models," doc.: IEEE 802.11-03/940r4, May 2004.
[33] W.C. Jakes, Microwave mobile communications, Wiley, New York, 1974.

Kei Mizutani was born in 1976. He received the B.E. degree in electrical engineering from National Defense Academy, Japan, in 1999, and the M.E. degree in electrical and electronic engineering from Tokyo Institute of Technology, Japan, in 2004. He is currently working toward the D.E. degree at Tokyo Institute of Technology. His research interests are implementation of MIMO systems and MIMO channel measurements. He is a student member of IEEE.

Kei Sakaguchi was born in Osaka, Japan, on November 27, 1973. He received the B.E. degree in electrical and computer engineering from Nagoya Institute of Technology, Japan, in 1996, the M.E. degree in information processing from Tokyo Institute of Technology, Japan, in 1998, and the Ph.D. degree in electrical and electronic engineering from Tokyo Institute of Technology in 2006. From 2000, he is an Assistant Professor at Tokyo Institute of Technology. He received the Young Engineer Awards from IEICE and IEEE AP-S Japan chapter in 2001 and 2002 respectively, and the Outstanding Paper Awards from SDR Forum and IEICE in 2004 and 2005 respectively. His current research interests are MIMO propagation measurements, MIMO communication systems, and software defined radio. He is a member of IEEE.

Jun-ichi Takada was born in Tokyo, Japan, in 1964. He received the B.E., M.E., and D.E. degrees from Tokyo Institute of Technology, Japan, in 1987, 1989, and 1992 respectively. From 1992 to 1994, he was a Research Associate at Chiba University, Japan. From 1994 to 2006, he was an Associate Professor at Tokyo Institute of Technology. Since 2006, he has been a Professor at Tokyo Institute of Technology. He received the Excellent Paper Award and Young Engineering Award from IEICE, Japan, in 1993 and 1994 respectively. His current research interests are wireless propagation and channel modelling, array signal processing, ultra-wideband radio, software defined radio, and applied radio instrumentation and measurements. He is a member of IEEE, ACES, and ECTI Association Thailand.

Kiyomichi Araki was born in 1949. He received the B.E. degree in electrical engineering form Saitama University, Japan, in 1971, and the M.E. and Ph.D. degrees in physical electronics from Tokyo Institute of Technology, Japan, in 1973 and 1978 respectively. From 1973 to 1975, and from 1978 to 1985, he was a Research Associate at Tokyo Institute of Technology. From 1985 to 1995, he was an Associate Professor at Saitama University. In 1979–1980 and 1993–1994, he was a visiting research scholar at University of Texas, Austin and University of Illinois, Urbana, respectively. Since 1995, he has been a Professor at Tokyo Institute of Technology. His research interests are information security, coding theory, communication theory, circuit theory, electromagnetic theory, and microwave circuits, etc. He is a member of IEEE and Information Processing Society of Japan.