Design and DSP Software Implementation of Mobile WiMAX Baseband Transceiver Functions

Hai-wei Wang[1], David W. Lin[1], Kun-Chien Hung[1], and Youn-Tai Lee[2]

[1] Dept. Electronics Engineering and Center for Telecommunications Research
National Chiao Tung University, Hsinchu, Taiwan, R.O.C.
[2] WiMAX Technology Center, Network and Multimedia Institute
Institute for Information Industry, Taipei, Taiwan, R.O.C.
c93jo6@gmail.com, dwlin@mail.nctu.edu.tw,
hkc.ee90g@nctu.edu.tw, lyt@nmi.iii.org.tw

Abstract. This paper considers the orthogonal frequency-division multiple access (OFDMA) physical layer of the IEEE 802.16e standard. We discuss the design of some key baseband signal processing functions, including synchronization, channel estimation, and forward-error-correction (FEC) decoding. Further, we describe a fully software implementation of these functions employing Texas Instruments' digital signal processors (DSPs).

Keywords: Channel estimation, digital signal processor (DSP) software implementation, bit-interleaved coded modulation, IEEE 802.16e, orthogonal frequency-division multiple access (OFDMA), synchronization, WiMAX.

1 Introduction

An increasing amount of programmability is being required of wireless communication devices. For handheld mobile devices, a completely software implementation is still not viable for cost and power consumption reasons, but to some degree one may not be able to fully ignore such a possibility in view of the proliferating wireless communication standards and the envisioned emergence of cognitive radio. For infrastructure devices such as base stations (BSs), on the other hand, the economical balance appears to tip comparatively more towards implementing many of the required signal processing functions in software. In either case, a fully software implementation is at least a useful tool that can serve various prototyping and functional verification purposes.

In this study, we consider the IEEE 802.16e orthogonal frequency-division multiple access (OFDMA) physical layer (PHY), which has been adopted in Mobile WiMAX. We discuss the design of some key baseband signal processing functions and describe a fully software implementation of these functions employing Texas Instruments' digital signal processors (DSPs).

The paper is organized as follows. Section 2 provides an overview of the relevant IEEE 802.16e OFDMA PHY specifications. Section 3 discusses three key

M. Denko et al. (Eds.): EUC Workshops 2007, LNCS 4809, pp. 181–192, 2007.

signal processing functions to some depth, namely, synchronization, channel estimation, and forward-error-correction (FEC) decoding. Section 4 presents some simulation results on their performance. Section 5 describes the DSP software implementation. Finally, Section 6 is the conclusion.

2 Overview of the IEEE 802.16e OFDMA PHY

Orthogonal frequency-division multiplexing (OFDM), by its use of cyclic prefixing (CP), is known to be able to combat multipath interference easily. The OFDMA PHY in IEEE 802.16e further divides the subcarriers into subchannels that can be allocated individually. This provides flexible access control in frequency-selective time-varying channels, but also introduces interesting signal processing problems. One feature of the IEEE 802.16e OFDMA is the selectable discrete Fourier transform (DFT) or fast Fourier transform (FFT) size, from 128 to 2048 in multiples of 2, excluding 256 that is used in the OFDM PHY.

For each FFT size, the subcarriers are divided into three types: null (guard bands and DC), pilot, and data. A data stream can be carried over one or more subchannels depending on its rate. Three basic types of subchannel organization are defined: partial usage of subchannels (PUSC), full usage of subchannels (FUSC), and adaptive modulation and coding (AMC), among which PUSC is mandatory and the others are optional. In what follows, we focus on PUSC.

Figure 1 illustrates the structure of a time-division duplex (TDD) frame that only uses PUSC. The downlink (DL) subfame starts with a preamble, followed by $2n$ OFDMA symbols, and the uplink (UL) subframe contains $3m$ OFDMA symbols, where n and m are integers. The preamble is an OFDM symbol where the used subcarriers are spaced three indices apart and these subcarriers are BPSK-modulated with one of 114 selectable pseudo-noise (PN) sequences.

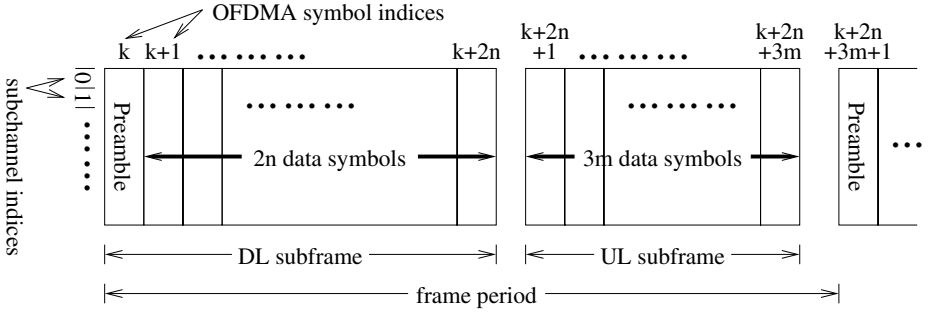

Fig. 1. Structure of IEEE 802.16e OFDMA TDD frame employing PUSC only

In the DL, every two successive OFDMA symbols form one unit in subchannel formation. Each subchannel consists of two "clusters" of subcarriers from each OFDMA symbol, where each cluster contains 12 data subcarriers and two pilot

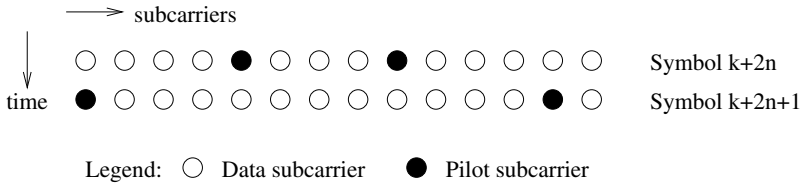

Fig. 2. DL cluster structure (modified from [1])

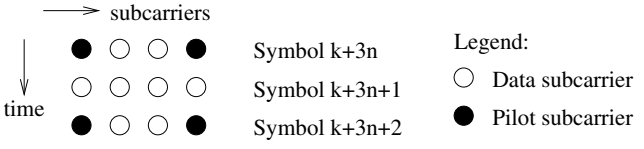

Fig. 3. UL tile structure (modified from [1])

subcarriers. Therefore, a DL subchannel contains at minimum 48 data subcarriers. The DL subchannels are organized into three segments to facilitate use of sectored antennas in a cell (with different segments allocated to different sectors) and the pilots are transmitted on segment basis, not on subchannel basis. Figure 2 illustrates the DL cluster structure.

In the UL, every three successive OFDMA symbols form a unit in subchannel formation. Twelve subcarriers, four from each of the three successive OFDMA symbols, constitute a "tile," and six pseudo-randomly chosen tiles form a subchannel. Figure 3 illustrates the tile structure. In the UL, the pilots are associated with the subchannels, rather than with the segments as in the DL. Thus each UL subchannel contains at least 48 data subcarriers and 24 pilot subcarriers.

3 Some Key Baseband Signal Processing Functions

Figure 4 illustrates the typical OFDMA transceiver structure. The receiver functions are usually more complicated than the transmitter functions. In IEEE 802.16e OFDMA DL, the main baseband receiver functions in the subscriber station (SS) are:

1. Synchronize to the BS. In initial startup, also identify the preamble index.
2. Do channel estimation using the broadcast message signal from the BS.
3. Receive the broadcast messages and obtain the assigned locations of the data bursts for this SS.
4. Do channel estimation for the allocated DL burst locations.
5. Receive signal in the assigned DL burst locations.

For UL transmission, disregard the ranging process (i.e., handshaking) in the present work. The main BS receiver functions in normal signal transmission are:

1. Synchronize to each SS signal.

2. Do channel estimation for the allocated UL burst locations of each SS.
3. Receive the signal from each SS in the allocated UL burst locations.

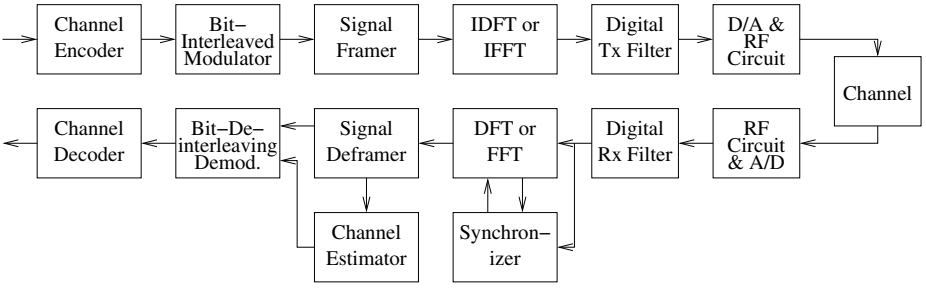

Fig. 4. Typical OFDMA transceiver structure

In the following, we discuss synchronization, channel estimation, and demodulation-decoding in separate subsections.

3.1 Synchronization

Consider initial DL synchronization first. In principle one can consider detecting the carrier frequency offset (CFO), the timing, and the preamble index jointly. But a less complicated approach is desirable for implementation purpose. Our method proceeds as follows:

1. In time domain, perform OFDMA symbol timing and fractional CFO estimation employing blind correlation based on the CP structure.
2. Convert the signal to frequency domain by FFT. Perform joint estimation of integer CFO and preamble index based on differential correlation.

After initial DL synchronization, we enter normal DL synchronization where we keep tracking of any variations in symbol timing and fractional CFO.

Let $r(k)$ be the received signal at time k, N be the FFT size, and L be the CP length. In step 1, the OFDMA symbol timing and the fractional CFO are estimated using the method in [2]:

$$\hat{\tau} = \arg\max_{\tau}\{|\lambda(\tau)|\} \ , \quad \hat{\theta} = -\frac{1}{2\pi}\arctan\frac{\Im\{\lambda(\hat{\tau})\}}{\Re\{\lambda(\hat{\tau})\}} \ , \tag{1}$$

where

$$\lambda(\tau) = \sum_{k=\tau}^{\tau+L-1} r(k)r^*(k+N) \ . \tag{2}$$

In step 3, the differential received signal is given by

$$D_R[k] = \Re\{r(k)r^*(k+3)\} \ . \tag{3}$$

A differential sequence for each of the 114 preambles is also calculated, except that there is no need for the conjugation or the $\Re\{\}$ operation because the preambles are BPSK signals. Let $D_{ID}[k]$ denote the results, where $ID = 0, 1, \ldots, 113$. Then $D_R[k]$ is correlated with each $D_{ID}[k]$ at each candidate integer CFO value. The combination of ID and integer CFO that yields the greatest correlation constitutes the estimator output. The above differential correlation method gives an approximate maximum-likelihood (ML) estimate in multipath channels [3].

In normal UL signal reception after the ranging process, only symbol timing and fractional CFO need to be synchronized. For this we employ the blind CP correlation as in step 1 of the initial DL synchronization.

3.2 Channel Estimation

In OFDM-type systems, channel estimates are needed for FEC decoding, among other things. A frequently considered channel estimation method in such systems is the simple least-square (LS) method that estimates the channel responses at pilot frequencies based on only one received OFDM or OFDMA symbol [4]. The estimates can be interpolated in frequency and in time to obtain channel estimates at non-pilot locations.

For OFDMA systems in slow fading, the received signal in frequency domain can be modeled as

$$r_k = H_k X_k + n_k \tag{4}$$

where k is the subcarrier index, H_k is the channel gain, x_k is the transmitted signal, and n_k is additive noise (assumed white Gaussian). An LS estimator minimizes the following squared error at the pilot locations:

$$\tilde{H}_k = \arg\min_{H_k} |r_k - H_k x_k|^2 \ . \tag{5}$$

With only one observed OFDMA symbol, the solution is simply

$$\tilde{H}_k = r_k / x_k \ . \tag{6}$$

Many frequency- and time-interpolation methods can be conceived [1]. The bilinear interpolation is one of the simplest. It is an appropriate choice when the frequency spacings of the pilot subcarriers are within a fraction of the coherent bandwidth and their temporal spacings are within a fraction of the coherence time. Such conditions are satisfied, for example, in both a DL cluster and a UL tile of an IEEE 802.16e OFDMA system of 10 MHz bandwidth, employing 1024-FFT, with carrier frequency below 3.5 GHz, and with mobile speed not over 120 km/h. When the channel response is subject to much slower time-variation, interpolation or averaging over a longer time period may be considered to yield more accurate channel estimates.

3.3 FEC Decoding

The mandatory coding scheme in IEEE802.16e OFDMA is tail-biting convolutional code (CC) with puncturing, followed by bit-interleaving and QAM. The

punctured bits can be treated as erasures in the CC trellis and thus do not affect the calculation of the path metrics. Both the tail-biting and the bit-interleaving influence the decoder design. We address them separately below.

Dealing with Tail-Biting. Tail-biting makes the encoder start in the state that it will end in. Optimal decoding of a tail-biting CC can be achieved by running as many parallel Viterbi decoders as there are states, with each decoder starting and ending in one different state. Then the best performer of all the decoder outputs gives the optimal solution. However, experience shows that conventional Viterbi decoding is easily a most complicated receiver function already. Since the code in IEEE 802.16e OFDMA has 64 states, the optimal solution is hardly practical for DSP implementation.

A suboptimal decoding method with good performance has been proposed [5], [6]. It employs only one Viterbi decoder that works on a circularly extended input sequence. The idea is that each path through the code trellis whose starting state and ending state are the same can be considered one cycle of an infinitely long periodic sequence. In Viterbi decoding, it is well-known that near-ML performance can be attained with a decoding delay of about 4 to 8 times the constraint length. Although we do not know the initial state of the received code sequence, if we perform Viterbi decoding long enough on the cyclically extended input sequence, we should converge to the optimal solution, but the leading segment and the trailing segment of the decoder output should be discarded.

Therefore, before decoding, we first do cyclic pre- and post-fixing of the received sequence for a sufficient length (e.g., both being 4–8 times the constraint length). After Viterbi decoding over the whole extended sequence, we drop the pre- and post-fixed segments to obtain the final decoder output.

Dealing with Bit-Interleaving. For ML Viterbi decoding in additive white Gaussian noise (AWGN), the error metric is the Euclidean distance between the trellis path and the soft-output of the demodulator. However, by having a bit-interleaver between the CC encoder and the QAM modulator, the combined trellis of the CC and the modulator becomes very entangled. As a result, the ML metrics become hard to calculate. A remedy is to approximate the path metric by the sum of bit metrics [7], [8]. A suitable bit metric for this is the log-likelihood ratio (LLR) [8]. The demodulator outputs the LLRs, which after bit-deinterleaving constitute the soft-decision inputs to the Viterbi decoder. For simplicity, log-sum approximation is applied to LLR calculation.

4 Simulation Results on Algorithm Performance

Table 1 lists the system parameters for many simulations reported below.

4.1 Synchronization

We report results on DL synchronization only. Let the transmission use a segment that is allocated 10 subchannels. Let the preamble index be 33, time offset be

Table 1. Parameters of simulated system

Parameter	Value
Nominal Channel Bandwidth	10 MHz
FFT Size (N_{FFT})	1024
Subcarrier Spacing	10.94 kHz
Useful Symbol Time (T_b)	91.4 μs
Guard Time ($T_g = 1/8 \cdot T_b$)	11.4 μs
OFDMA Symbol Time ($T_s = T_g + T_b$)	102.9 μs
No. of OFDMA Symbols per 5-ms Frame	48*

* 1 preamble + 12×2 DL symbols + 7×3 UL symbols + 2 gaps and unused time.

10 samples, and CFO be 9.35 subcarrier spacings. Two kinds of channel are simulated, namely, AWGN and the Stanford University Interim (SUI) [9].

Figure 5 shows the root-mean-square-error (RMSE) in symbol time estimation in AWGN. The floating-point results are obtained on personal computer (PC) and the fixed-point ones with Texas Instruments (TI)'s CCS tool for the TMS320C6416T DSP using 16-bit computation. Errors are small, and the performance difference between floating-point and fixed-point computations is little.

Figures 6 illustrates the RMSE in fractional CFO estimation in initial DL synchronization in the SUI3 channel at several mobile speeds up to 120 km/h. The RMSE is under 2% of the subcarrier spacing in medium to high signal-to-noise ratio (SNR). The performance in normal synchronization (not shown) is even better because more time has elapsed. The performance difference between floating-point computation and fixed-point computation is reasonable.

Finally, consider integer CFO and preamble index estimation, for which Fig. 7 shows some results for the SUI3 channel. The error rates are low.

4.2 Channel Estimation

We report results on DL channel estimation only, for UL results are characteristically similar. Consider transmitting in segment 0 with the 6 subchannels in "major group 0" allocated to it. Hence there are a total of 144 data subcarriers and 24 pilot subcarriers per symbol. Consider two fixed-point formats in 16-bit computation: Q3.12 and Q2.13. Figure 8 illustrates the mean-square error (MSE) in channel estimation under AWGN with QPSK modulation. The simulation results match well with theory, and the fixed-point results (obtained with CCS) are very close to that using floating-point computation. Figure 9 illustrates the MSE in channel estimation under SUI2 channel at 60 km/h speed. We see that the Q2.13 data format has sufficient dynamic range and, by having one more fractional bit than Q3.12, shows a lower error floor in high SNR.

4.3 FEC

Seven combinations of modulation and coding schemes are defined in the tail-biting CC mode of IEEE 802.16e OFDMA. Figure 10 illustrates the bit error rate

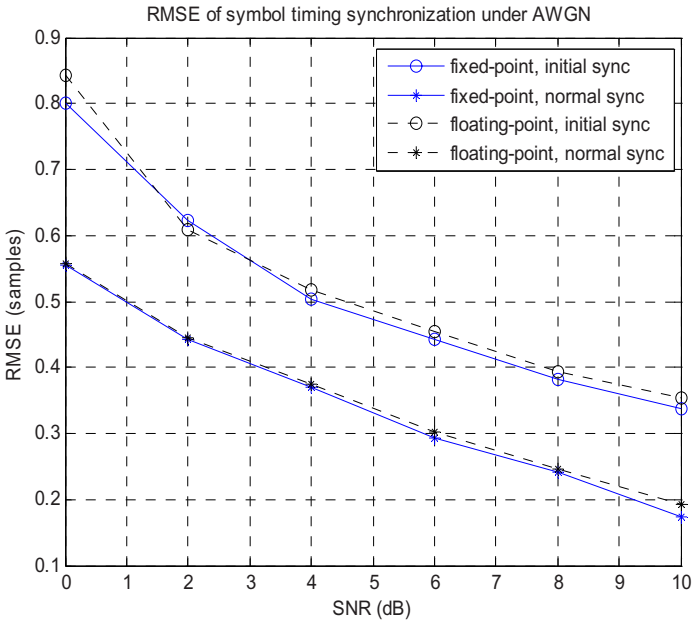

Fig. 5. Performance of symbol time estimation in AWGN

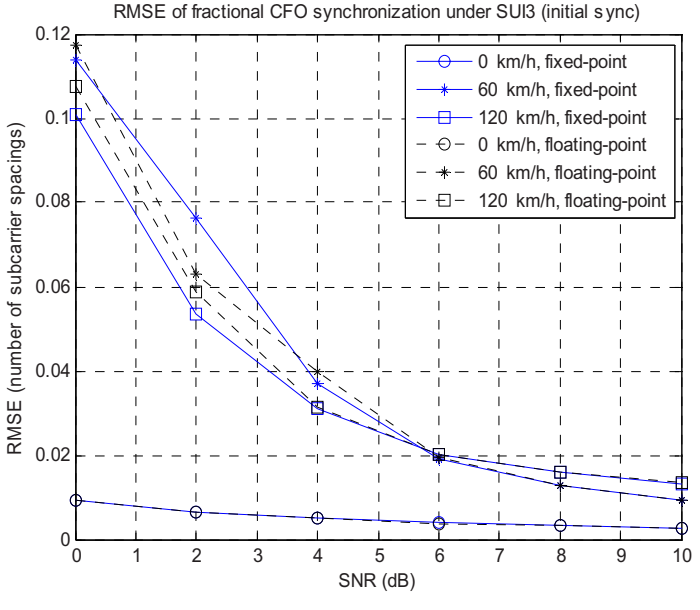

Fig. 6. Performance of fractional CFO estimation in initial DL synchronization in SUI3 channel

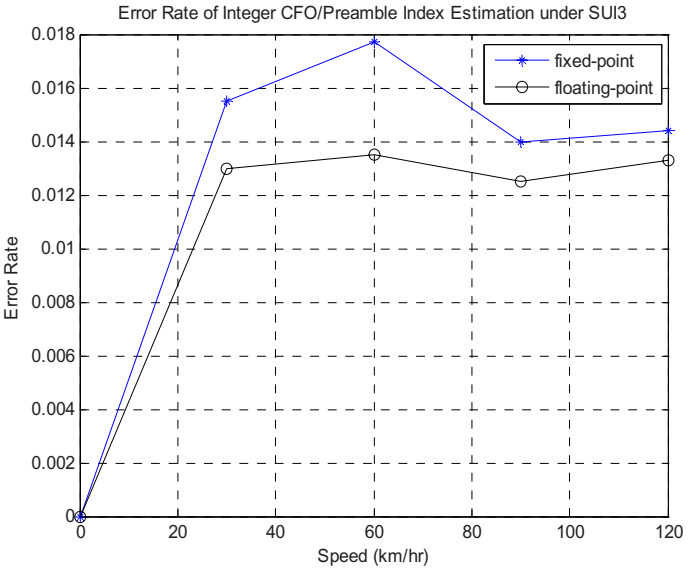

Fig. 7. Performance of integer CFO and preamble index estimation in SUI3 for mobile speed up to 120 km/h

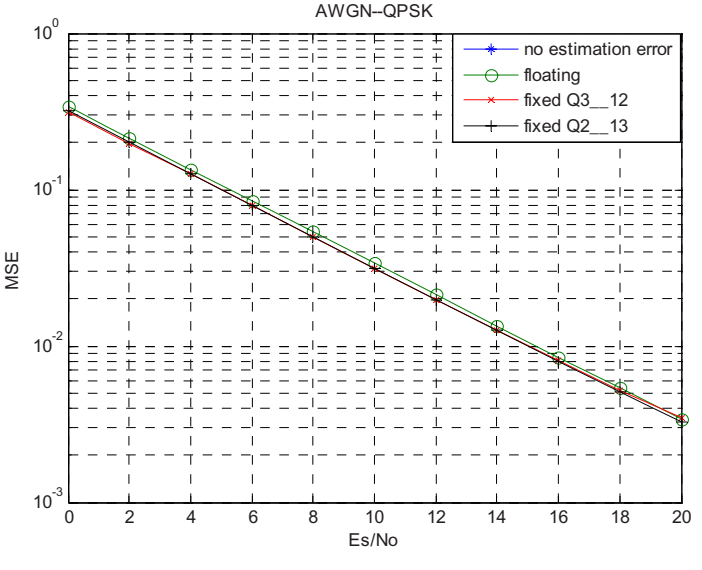

Fig. 8. MSE of DL channel estimation under AWGN

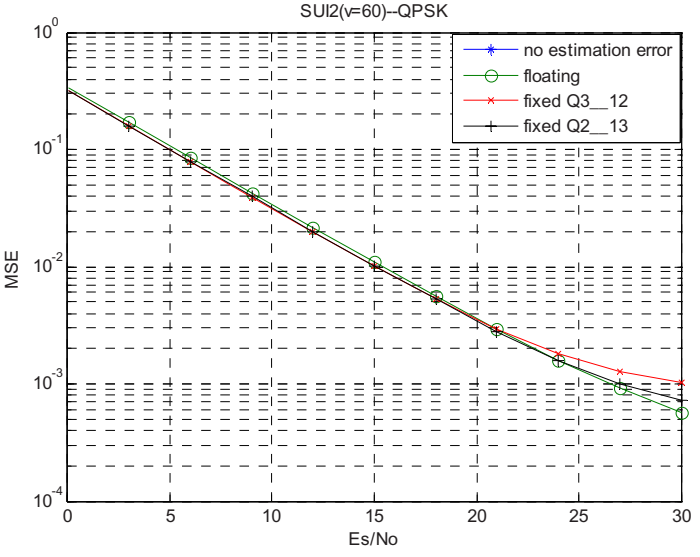

Fig. 9. MSE of DL channel estimation under SUI2 channel at mobile speed 60 km/h

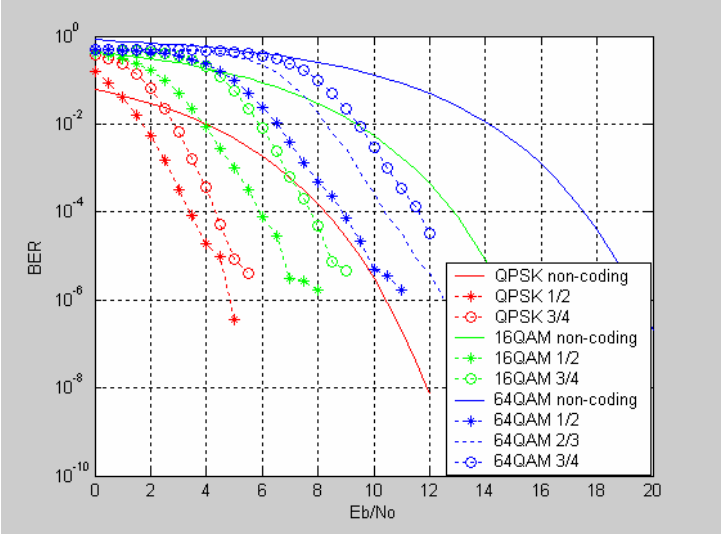

Fig. 10. FEC performance with fixed-point computation

(BER) performance of FEC decoding in AWGN with fixed-point computation, where the number after each modulation type in the legend denotes the code rate. For space reason, we omit a plot of the results with floating-point computation. Suffice it to say that they are in good correspondence.

5 DSP Software Implementation

For DSP implementation, we employ Sundance's SMT8096 platform which is built around TI's TMS320C6416T DSP. The SMT8096 comprises of an SMT310Q PCI PC plug-in board called a "carrier" which can house four SMT395 daughter cards termed Texas Instruments Modules (TIMs). Each TIM houses one TMS320C6416T DSP. The DSP contains 8 parallel function units and runs at 1 GHz. Two TIMs can communicate through the Sundance Digital Bus (SDB). The PCI interface of SMT310Q runs at 33 MHz with a 32-bit data bus. The DSP platform employs the 3L Diamond real-time operating system that supports multiple processor communication and synchronization.

Table 2. Computational performance of the implemented system

Module	DSP Computational Load*
Baseband Transmitter exc. Tx Filter	0.82
Tx Filter (4× oversampled)	0.25
Channel Simulators:	
AWGN	0.82
SUI	1.52
ETSI_A	2.56
Rx Filter (4× oversampled)	0.54
FFT, Sync., & Channel Estimation	0.23
Deframing, Demod., & Decoding	1.16

* As multiple of one DSP's computational capability.

For processing speed and convenience in work division, four TIMs are used: one for the transmitter functions, two for the receiver functions, and one to simulate the equivalent lowpass wireless channel. Table 2 shows the computational speed of the integrated implementation for a 10-MHz bandwidth, 1024-FFT system in DL transmission. Note that channel decoding is a most computation-intensive transceiver function. But interestingly, some "basic" functions, such as the transmitter and the receiver filters, also require much computation. Comparatively, the proposed synchronization and channel estimation methods are less computation-intensive, but they have significant impact on the transmission performance and require a careful design.

6 Conclusion

We considered the OFDMA PHY of the IEEE 802.16e standard and discussed the design of some key signal processing functions. We described a fully software implementation of these functions employing Texas Instruments' DSPs. Acknowledgeably, the implementation still has much room for improvement. However, it demonstrates that a fully software implementation of the baseband transceiver functions for a system as complicated as the Mobile WiMAX is not

beyond reach. In addition, such an implementation is a useful tool that can serve various research, prototyping, and functional verification purposes.

Acknowledgments. This work was supported by the New Generation Broadband Wireless Communication Technologies and Applications Project of the Institute for Information Industry, sponsored by MOEA, R.O.C., under Grant 96-EC-17-A-03-R7-0765 and by the National Science Council of R.O.C. under Grant 95-2219-E-009-003. The following people also played a major role in the reported work: Soon Seng Teo, Yao-Chun Liu, Yi-Ling Wang, and Po-Sheng Wu.

References

1. Hung, K.-C., Lin, D.W., Lee, Y.-T., Loa, K.: WirelessMAN physical layer specifications: signal processing perspective. In: Zhang, Y., Chen, H.-H. (eds.) Mobile WiMAX: Toward Broadband Wireless Metropolitan Area Networks Ch. 3, Auerbach (2007)
2. Lin, J.-C.: Maximum-likelihood frame timing instant and frequency offset estimation for OFDM communication over a fast Rayleigh fading channel. IEEE Trans. Veh. Technol. 52, 1049–1062 (2003)
3. Hung, K.-C., Lin, D.W.: Joint detection of integral carrier frequency offset and preamble index in OFDMA WiMAX downlink synchronization. In: IEEE Wireless Commun. Networking Conf., pp. 1961–1966 (2007)
4. van de Beek, J.-J., Edfors, O., Sandell, M., Wilson, S.K., Börjesson, P.O.: On channel estimation in OFDM systems. In: IEEE 45th Veh. Technol. Conf., vol. 2, pp. 815–819 (1995)
5. Wang, Y.-P.E., Ramesh, R.: To bite or not to bite — a study of tail bits versus tail-biting. In: Proc. IEEE Int. Symp. Personal Indoor mobile Radio Commun., vol. 2, pp. 317–321 (1996)
6. Sung, W., Kim, I.-K.: Performance of a fixed delay decoding scheme for tail biting convolutional codes. In: Proc. IEEE Asilomar Signal Syst. Computers Conf., vol. 1, pp. 704–708 (1996)
7. Zehavi, E.: 8-PSK trellis code for a Rayleigh channel. IEEE Trans. Commun. 40, 837–884 (1992)
8. Tosato, F., Bisaglia, P.: Simplified soft-output demapper for binary interleaved COFDM with application to HIPERLAN/2. In: IEEE Global Telecommun. Conf., vol. 2, pp. 664–668 (2002)
9. Erceg, V., et al.: Channel models for fixed wireless applications. Standards contribution no. IEEE 802.16.3c-01/29r4 (2001)