A NEW SOFTWARE APPROACH FOR REAL TIME OFDM

Yahya M. Khawaja, Muhammad U. K. Khan, Abdul Bais, Rizwana Arshad, Muhammad Suleman, Ihsan Ul Haq

NWFP University of Engineering and Technology, Peshawar, Pakistan

ABSTRACT

This paper discusses real time implementation of a new Software Defined Radio (SDR) for Orthogonal Frequency Division Multiplexing (OFDM). The proposed SDR is implemented on a single PC, which does not include any external hardware and solely rely on the capabilities of the PC. A step by step comparison of the proposed SDR with the real world system for OFDM is presented and the errors encountered by the proposed SDR due to sampling offsets are also discussed. Since the proposed SDR does not include any external hardware, its portability is tremendous and upgradation is economical, and the approach can be extended to other communication techniques. The SDR is a multi-standard device and a flexible and economical approach for communication because it incorporates the newer technologies without considerable change of hardware.

Index Terms— Digital communication, OFDM, Sampling offsets, Software radio

1. INTRODUCTION

Referring to Fig. 1, we see that an SDR uses programmable devices (reconfigurable hardware and software) such as DSP and FPGAs [1, 2] to perform the necessary signal processing in order to transmit and receive the information from baseband, modulating a Radio Frequency (RF). The SDR uses digital and an analog subsystems (the analog system being used to perform tasks that cannot be performed by the digital systems like RF filtering, amplification and frequency generation). Compared to SDR, the Hardware Defined Radio (HDR) is with little or no software control [3].

In the HDR, every block functions independently from the other block and only depends on the data from previous block. Incoming data is processed without any regard to previously processed data. In SDR, all the processing is done via a single microprocessor and one command can be executed at a time. Therefore, a block in SDR must wait for a single chunk of data being processed and transmitted to start processing the next input. So the speed of execution in SDR is slower as compared to the HDR. This requires optimization of codes.

If a block of HDR starts to malfunction then replacement of the block is needed. New standards not compatible with the already implemented standards require that blocks in HDR to be replaced. Different countries have different communication standards which automatically requires different hardware. Another issue with the HDR is that once a receiver is designed to receive some specific signal, like 16 PSK, it will not demodulate any other type of RF signal. Upgrading the system could be another problem after installation of HDR.

The SDR tackles all these problems. It can incorporate newer standards easily and can be made to receive any type of modulated signal and demodulate it. It provides multi-standard support. SDR is portable in a sense that the whole system is defined with a high level programming language. Upgrading the SDR is easy as compared to HDR and addition of newer and more efficient algorithms is possible.

Software approach must have low circuit complexity, cost and power consumption because the goal of SDR is to be flexible, economical and adaptable. The difficulty arises when we need to convert the analog RF signal to digital. Down converters are incorporated but the limited bandwidth of converters and jitter produce distortions [4].

The schematic of proposed SDR for OFDM is shown in Fig. 2. It is noted that the data entry to the SDR is the same in both Fig. 1 and Fig. 2 but the transmission differs because we do not hand over the data from the API to an external device for processing, rather a sound card can be used for transmission purposes. Discussions about the diagram follow in the

Fig. 1. A layout of a typical SDR
In OFDM, data stream is transmitted over a number of orthogonal Sub Carriers (SCs) instead of a single carrier as done in other modulation techniques. The orthogonality assures that the maximum of a modulated carrier in the frequency domain occurs at the point where effect of all other carriers is simultaneously zero.

Data stream is used to modulate the SCs that are transmitted in parallel. The frequency difference between adjacent carriers is \( w \). The carriers are modulated by data \( A \) and all the orthogonal carriers are added to make an OFDM symbol \( s(t) \) of duration \( T_o \), i.e.

\[
s(t) = A_1e^{jwt} + A_2e^{j2wt} + A_3e^{j3wt} + \ldots
\]

\[
= \sum_n A_n e^{jnw}(1)
\]

If we sample the signal with the sampling frequency \( \frac{1}{T_s} \) and obtain \( N \) samples in one symbol of OFDM then:

\[
s(kT_s) = \sum_n A_n e^{jnw}(kT_s)
\]

Comparing with the equation of Inverse Discrete Fourier Transform (IDFT):

\[
g(kT_s) = \sum_n A_n e^{j2\pi(nk/N)}
\]

we see that both the equations are equal if:

\[
\frac{w}{2\pi} = \frac{1}{NT_s} = \frac{1}{T_o}
\]

where \( w/2\pi \) is actually the frequency difference between two carriers and each carrier has a whole number of cycles in the one symbol time, which is the basic requirement for orthogonality.

Because IDFT can be easily implemented using Inverse Fast Fourier Transform (IFFT), an added advantage of orthogonality is that we can use IDFT algorithm to generate an OFDM symbol instead of a bank of parallel modulators.

The received OFDM signal \( r(t) \) is given by [5]:

\[
r(t) = \sum_n h(t)s(t-\tau) + \eta(t)
\]

where \( \eta(t) \) is noise added during transmission and \( h(t) \) is the impulse response of the channel. If we sample the signal \( r(t) \) with sampling frequency of \( \frac{1}{T_r} \), then we have:

\[
r(kT_s) = \sum_n h(kT_s)s(kT_s - \tau) + \eta(kT_s)
\]

By taking Fast Fourier Transform (FFT) of \( r(kT_s) \) we get the received data \( z \) as given by (8), along with some noise which degrades Signal to Noise Ratio (SNR).

\[
z(k) = \sum_n r(k)e^{-j2\pi(n/N)k}
\]

There are problems associated with OFDM. InterSymbol Interference (ISI) is common to OFDM as well as other systems. InterCarrier Interference (ICI) is produced due to the multipath and the received signal becomes:

\[
r(kT_s) = e^{j2\pi(f_s-kT_s)} \sum h(kT_s)s(kT_s - \tau) + \eta(kT_s)
\]

where

\[
f_s = f_{rx} - f_{tx}
\]

and \( f_s \) is the transmitted frequency and \( f_{rx} \) is the down converted (or upconverted) frequency by the receiver. The Sampling Frequency Offset (SFO) \( T_\Delta \) between the sampling periods of Digital to Analog Converter (DAC) of the transmitter \( T_{tx} \) and Analog to Digital Converter (ADC) of receiver \( T_{rx} \) contributes to rotation of SCs [6]. This offset is given by:

\[
T_\Delta = \frac{T_{rx} - T_{tx}}{T_{tx}}
\]

The effect of sampling frequency offset in the received signal is expressed by:

\[
R_{l,k} = e^{j2\pi(kl\Delta)\frac{T_{sym}}{T_{use}}} S_{l,k} \text{sinc}(kt\Delta\pi) H_{l,k} + N_{l,k}
\]

where \( l \) is the OFDM symbol index, \( k \) is the subcarrier index, \( T_{sym} \) is the total symbol time or total symbol indices and \( T_{use} \) is the total symbol time excluding Guard Interval (GI) time \( T_G \) [6]. \( N_{l,k} \) simulates the effect of noise. The
rotation of received signal constellation due to this effect is shown in Fig. 3. The rotation experienced by different SCs is due to sampling frequency offset and is given by the term:
\[ e^{i2\pi(kt\Delta f)\frac{T_{sym}}{T_{use}}} \]
in (12), and we note that for large \( l \) the rotation increases. The effect special to OFDM is that the sampling frequency offset causes ICI and the SNR of the signal degrades. By the above discussion, it is clear that it requires an extensive insight of synchronization algorithms in conjunction with OFDM. These offsets and their effects are fully discussed in [6, 7, 8, 5].

The use of GI mitigates the effect of ISI [6]. Pilot carriers are used for synchronization purposes. These are symbols having frequencies known both to the transmitter and receiver. Pilots can be block or comb type[9]. Pilots are send with each data symbol in block type pilot arrangement and comb type arrangement sends pilots at specific intervals of time. The synchronization algorithms are used to obtain start of an OFDM symbol, sampling frequency synchronization, to remove Carrier Frequency Offset (CFO) and Carrier Phase Offset (CPO) [6, 10].

3. THE OFDM SYSTEM

We discuss the implementation of the proposed OFDM system. The goal is to develop a software terminal for transmission and reception of voice packets in real time using OFDM. The test setup includes two Intel PCs (1.7GHz), running Windows XP. Each PC had two sound cards (one as ADC and one as DAC). The model involved the implementation of physical layer and the application layer.

Fig. 4 refers to the transmitters of real world and proposed SDR systems. For this study, an OFDM symbol was made from 20 msec of voice data captured by the sound card at transmitter and sampled at 8000 Hz. It took about 3 msec for 320 bytes of voice data to be processed and converted to an OFDM symbol.

Coding is carried out after the compression. Real world systems use interleaver to feed the modulator (mapper) block in order to tackle burst errors and to produce diversity. Signal mapping can utilize any constellation like QPSK and 64 QAM. A real world system uses a block type or comb type pilot arrangement. The proposed SDR utilizes comb type arrangement.

Pilots are there at start of transmission for about 1 sec in the proposed SDR. The consecutive pilot symbols are at 180° phase shifts so that the receiver can correctly obtain the start of the symbol using GI correlation (details in the OFDM receiver). In the SDR for OFDM, the IFFT is taken in such a way that the output of the IFFT block does not produce any frequency greater than half the sampling frequency of the output DAC, in accordance with the Nyquist criterion. So we can only fill up to half of IFFT indices. To avoid DC and owing to bandpass nature of the channel, the first few indices of IFFT are fed zeros. GI is added to the composite output of IFFT.

Output of the GI Insertion block is the OFDM symbol. The real world system utilizes the I and Q channels to transmit data. The SDR can transmit only the real part because we did not include any external hardware to send real and imaginary data on I and Q channels. Thus we avoided two multipliers, a 90° phase shifter and an adder. Imaginary output of IFFT is generated at the receiver using the Hilbert Transform [11] from the real part.

The receivers of OFDM system are shown in Fig. 5. Both require a signal detector at the front end of the receiver to avoid demodulating noise and save power and processing [6].

Time synchronization algorithms are employed in the real world system to get the start of an OFDM symbol. Software approach utilizes crosscorrelation to detect the start point \( t_{start} \) of the symbol [6, 7]. The correlation is carried out using a correlation window size equal to GI. In the following formula:

\[ m_n = \max \left| \sum r_n r_{n+a} \right| \]  
(13)

\( r \) is the received pilot and the \( a \) is the number of data points (samples) without GI in the symbol. \( m \) is maximum at the starting index of the sampled OFDM symbol because the win-
The comparison of real world and proposed SDR for an OFDM receiver. The lower figure is for the proposed SDR windows have same data of GI. The real world systems use frequency offset correction and sampling clock offset correction algorithms after the signal detection [6, 12, 10]. These algorithms can be carried out post FFT. In proposed SDR, synchronization can be carried out using algorithms presented in [6], after Hilbert Transform.

It can also use resampling algorithms and frequency offset correction algorithms for removing SFO. Both approaches remove the GI and take FFT of the data. Data is remapped. The real world system de-interleaves and decodes the data. Decompression is carried out and we receive the input data to the transmitter. In the proposed SDR, the sound is played on the sound card (DAC) at the receiver.

4. EXPERIMENTAL RESULTS

We compare different aspects of the proposed SDR with its changing parameters. Since the sampling frequencies of ADC and DAC of the transmitter ($T_x$) and receiver ($R_x$) are fixed (because they are just sound cards), therefore, we did not simulate for a Voltage Controlled Oscillator (VCO). The sampling frequency errors, when the receiver’s clock of ADC is faster or slower than transmitter’s clock of DAC, are plotted with different parameters as shown in Fig. 6.

From Fig. 6(a) we notice that the estimation gets poorer with the increasing offset and the algorithm is unable to estimate the offset correctly. Thus the resampling of the data will not produce accurate results if the frequency offset is high. This required that the proposed SDR used good quality sound cards, preferably from the same vendor to reduce SFO. Here, estimation is the ratio between the frequencies of receiver’s ADC and transmitter’s DAC.

Fig. 6(b) shows that the computation of SFO estimation gets closer to the actual offset with the increasing sampling frequency of the receiver.

Fig. 7 shows the spectrum of the pilot symbol. Two consecutive pilot symbols are scanned by the correlation window to detect the starting point of the symbol, after the signal is detected. The maximum of the correlation occurs at the index 20 as shown. This is the starting point of an OFDM symbol.

5. CONCLUSION

The SDR discussed here is flexible because it involves no use of sophisticated hardware and programable chips. Rather, ef-
Table 1. Comparison of various systems employing OFDM

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Proposed</th>
<th>DAB</th>
<th>DV B</th>
<th>802.11a</th>
</tr>
</thead>
<tbody>
<tr>
<td>Carriers</td>
<td>60</td>
<td>1536</td>
<td>1705</td>
<td>48</td>
</tr>
<tr>
<td>( FFT_{length} )</td>
<td>128</td>
<td>2048</td>
<td>4096</td>
<td>64</td>
</tr>
<tr>
<td>GI Ratio</td>
<td>1.5</td>
<td>1.5</td>
<td>1:32 to 1:4</td>
<td>1:5</td>
</tr>
<tr>
<td>( T_G ) (sec)</td>
<td>5.3m</td>
<td>0.25m</td>
<td>7 to 56( \mu )</td>
<td>400 to 800( \mu )</td>
</tr>
<tr>
<td>( T_{sym} ) (sec)</td>
<td>26.7m</td>
<td>1.25m</td>
<td>231 to 280( \mu )</td>
<td>4( \mu )</td>
</tr>
<tr>
<td>Modulation</td>
<td>QPSK</td>
<td>QPSK</td>
<td>QPSK to 64QAM</td>
<td>BPSK to 64QAM</td>
</tr>
<tr>
<td>Data Rate (bps)</td>
<td>6k(^c)</td>
<td>40 to 64k(^c)</td>
<td>20 to 38M</td>
<td>6 to 54M</td>
</tr>
<tr>
<td>BW (Hz)</td>
<td>22.5k</td>
<td>1.7M</td>
<td>6,7 or 8M</td>
<td>16.66M</td>
</tr>
</tbody>
</table>

Fig. 8. Detection of symbol start point with varying sampling frequency offsets

Fig. 8. Detection of symbol start point with varying sampling frequency offsets

sufficient use of sound cards resulted in an economic design. This work requires lesser hardware than given by [1, 2]. The design presented here of SDR can be modified and used for more practical purposes.

Further improvements in the SDR are possible. We can increase the sampling rate of the input ADC of the transmitter in order to represent more faithfully the speech signal. By introducing interleaving, error detection and correction, we can calculate BER. Similarly bigger constellations can be incorporated. Wireless medium and a networking device can be utilized.

6. REFERENCES


